Tri Rhena Tech

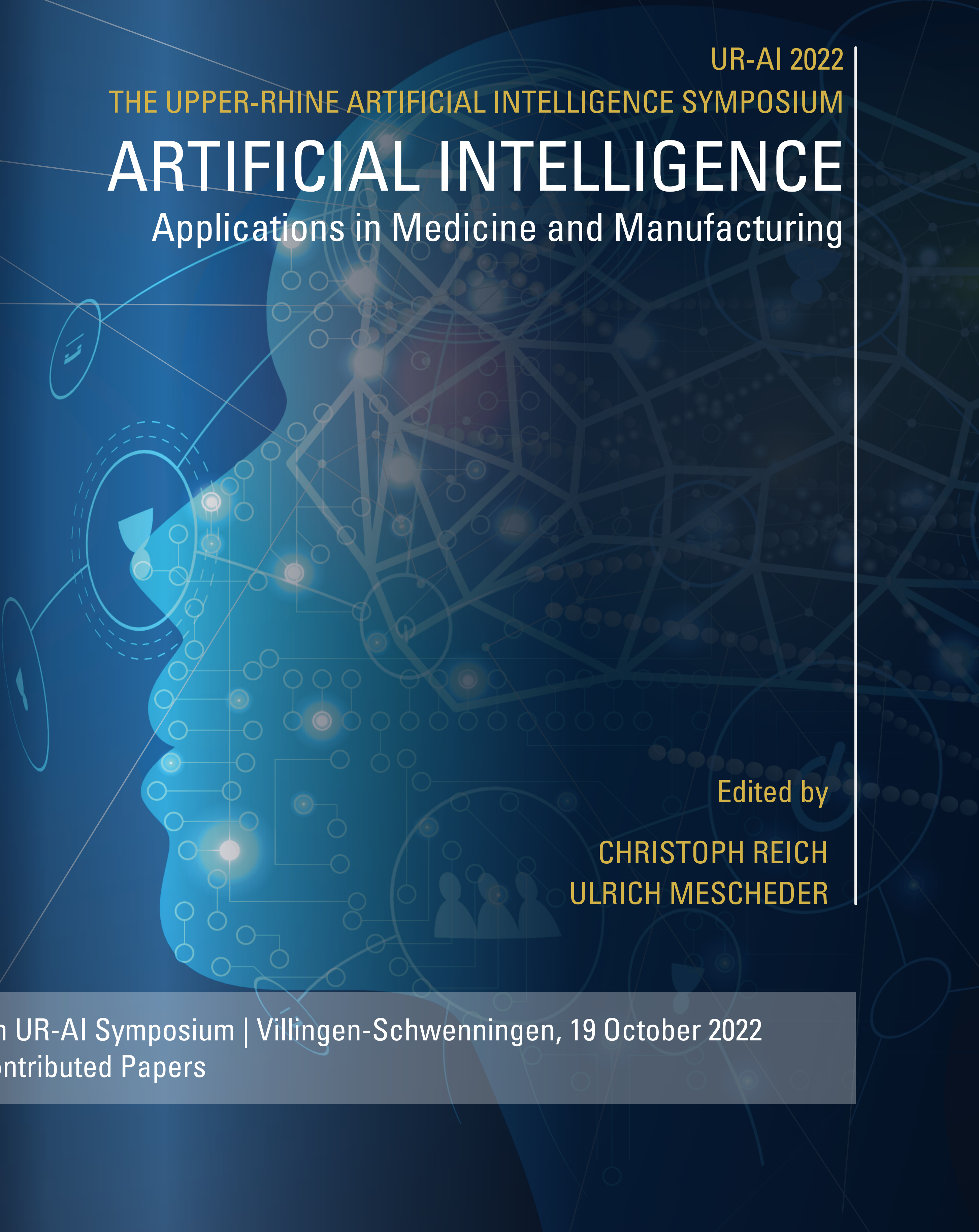
HOCHSCHULE FURTWANGEN UNIVERSITY HFU

UR-AI 2022

THE UPPER-RHINE ARTIFICIAL INTELLIGENCE SYMPOSIUM

ARTIFICIAL INTELLIGENCE

Applications in Medicine and Manufacturing



Edited by

CHRISTOPH REICH
ULRICH MESCHEDER

4th UR-AI Symposium | Villingen-Schwenningen, 19 October 2022
Contributed Papers

The Upper-Rhine Artificial Intelligence Symposium
UR-AI 2022

AI Applications in Medicine and Manufacturing

Christoph Reich, Ulrich Mescheder (eds.)

ISBN   978-3-00-073637-7
e-ISBN   978-3-00-073638-4

Published by: Furtwangen University
Cover illustration by: vs148/shutterstock
Cover by: Caroline Armbruster
Printed by: Druckerei Leitz GmbH, Furtwangen

The Upper-Rhine Artificial Intelligence Symposium
UR-AI 2022

AI Applications in Medicine and Manufacturing

Conference Chairs

Ulrich Mescheder, Furtwangen University
Christoph Reich, Furtwangen University

Program Committee

Andreas Christ, Offenburg University of Applied Sciences
Klaus Dorer, Offenburg University of Applied Sciences
Thorsten Fitzon, Furtwangen University
Thomas Lampert, Télécom Physique Strasbourg
Jörg Lohscheller, Trier University of Applied Sciences
Ulrich Mescheder, Furtwangen University
Enkelejda Miho, University of Applied Sciences and Arts Northwestern Switzerland
Knut Moeller, Furtwangen University
Franz Quint, Karlsruhe University of Applied Sciences
Christoph Reich, Furtwangen University
Karl-Herbert Schäfer, Kaiserslautern University of Applied Sciences
Ulf Schreier, Furtwangen University
Holger Ziekow, Furtwangen University

Organising Committee

Anna Dister, TriRhenaTech
Thorsten Fitzon, Furtwangen University
Manav Madan, Furtwangen University
Ulrich Mescheder, Furtwangen University
Christoph Reich, Furtwangen University

Table of contents

FOREWORD iii

1 MANUFACTURING 1

AI-Powered Defect Segmentation in Industrial CT Data 2
Tim Schanz, Robin Tenscher-Philipp and Martin Simon

Deep Learning based classification of vocal folds' vibration dynamics 12
Mona Kirstin Fehling, Maximilian Linxweiler, Bernhard Schick and Jörg Lohscheller

Feasibility study to distinguish mundane movements automatically by analysing the pressure distribution on a seat 19
Alparslan Babur, Nicolas Dockwiller, Yacine Belguermi, Ali Moukadem, Alain Dieterlen and Katrin Skerl

Image Processing and Neural Network Optimization Methods for Automatic Visual Inspection 25
Kawther Aboalam, Christoph Neuswirth, Florian Pernau, Stefan Schiebel, Fabian Spaethe and Manfred Strohrmann

Learning based Model Predictive Control of a High-Altitude Simulation Chamber . 35
Arsema Derbie Chekol, Maurice Kettner and Eyassu Woldesenbet

Modelling of a large-format lithium-iron-phosphate-based lithium-ion battery cell with neural ordinary differential equations 41
Jennifer Brucker, Wolfgang G. Bessler and Rainer Gaspe

Object Classification with a Robot Gripper equipped with Force Sensitive Fingertips using Convolutional Neural Networks 51
Christoph Uhrhan

Predicting critical machining conditions using time-series imaging and deep learning in slot milling of titanium alloy 57
Faramarz Hojati and Bahman Azarhoushan

Searching for Feature Sets for Misalignment Classification Using Experimental Data and Data Mining 64
Sebastian Bold and Sven Urschel

Silage Bale Detection for the «Cultivable Area» Update of the Cantonal Agricultural Office, Thurgau 66
Adrian Meyer and Denis Jordan

Value-Sensitive Design for AI Technologies: Proposition of Basic Research Principles Based on Social Robotics Research 74
Theresa Schmiedel, Janine Jäger and Vivienne Jia Zhong

2 MACHINE LEARNING 81

Analyzing sequential Graph Generation with Graph Convolutional Policy Networks 82
Ruxandra Lasowski

Explainable AI: A key driver for AI adoption, a mistaken concept, or a practically irrelevant feature? 88
Julia Dvorak, Tobias Kopp, Steffen Kinkel and Gisela Lanza

Improved e-mail forensic using dynamic graphs and change-point detection 98
Christian Hiller and Andreas Wagner

Machine Learning Models in Industrial Blockchain, Attacks and Contribution . . . 106
Fatemeh Ghovanlooy Ghajar, Axel Sikora, Jan Stodt and Christoph Reich

Tackling Key Challenges of AI Development – Insights from an Industry-Academia Collaboration 112
Alexander Melde, Paul Gavrikov, Manav Madan, David Hoof, Astrid Laubenheimer, Janis Keuper and Christoph Reich

3 MEDICAL TECHNOLOGY 123

Breast cancer classification methods for augmented reality microscopes 124
Robin Heckenauer, Jonathan Weber, C ́edric Wemmert, Michel Hassenforder, Pierre-Alain Muller and Germain Forestier

Generative Adversarial Network for Facial Emotion Recognition: A Feasibility Study 132
Herag Arabian and Knut Moeller

German Medical Natural Language Processing – A Data-centric Survey 137
Torsten Zesch and Jeanette Bewersdorff

Risk-based Assessment of ML-based Medical Devices 146
Martin Haimerl

Specification of neck muscle dysfunction through digital image analysis using machine learning 151
Filip Paskali, Angela Dieterich and Matthias Kohl

Spatial-temporal Modelling for Surgical Tool Classification in Cholecystectomy Videos 158
Tamer Abdulbaki Alshirbaji, Nour Aldeen Jalal, Thomas Neumuth and Knut Moeller

FOREWORD

To deliver better healthcare to patients and advance healthcare solutions as well as to increase the efficiency of the manufacturing process and thus reduce material and energy consumption in production, more and more artificial intelligence (AI) methods are applied in the field of both, Medicine and Manufacturing. Some of the exciting applications in these areas are remote patient treatment, transcription and storage of digitalized medical data, new drug development, support tools for fastened disease diagnosis, visual quality control in production processes, intelligent supply chain, and logistics solutions, and machine parameter optimization. The applications of artificial intelligence are manifold and therefore, many experts call AI a disruptive and cross-sectional technology. In general, AI is mainly used to optimize internal processes or to develop new business models. However, AI for Medicine and Manufacturing faces obstacles such as a shortage of high-dimensional data, privacy concerns regarding the data collected, and the risks of the creation of biased algorithms if data are not collected over a representative population.
The articles presented in this conference proceedings report are selected from the oral presentations and poster presentations from the conference held on 19 October 2022 at the University Furtwangen, Campus Schwenningen, and assigned to the chapters MANUFACTURING, MACHINE LEARNING, and MEDICAL TECHNOLOGY.
The URAI 2022 is the fourth conference on artificial intelligence organized by the tri-national alliance TriRhenaTech, the alliance of universities of applied sciences in the Upper Rhine region. Starting in 2019 in Offenburg, the TriRhenaTech universities have been cooperating on AI for many years. 2021 the conference was held at Hochschule Kaiserslautern and the focus was on applications of AI in life sciences. This year, 2022, the conference moved to Hochschule Furtwangen. The conference's focus on AI in Manufacturing and Medicine was chosen to emphasize the role and the strength of Manufacturing and Medicine in and for our region.

Prof. Dr. Christoph Reich
Institute of Data Science,
Cloud Computing, and IT Security
Furtwangen University

Prof. Dr. Ulrich Mescheder
Vice-President Research and Transfer
Furtwangen University

Chapter 1

MANUFACTURING

Keynote Abstract: 1, Introducing Keynote Speaker Prof. Dr. Marco Huber

Cognitive Production Systems – Machine Learning in Industrial Manufacturing

Machine learning (ML) methods have recently led to enormous progress in the field of artificial intelligence. It allows the automatic recognition and exploitation of correlations and patterns in complex data. The application of ML is particularly useful in applications where cause-effect relationships are very difficult or impossible to describe analytically using mathematical methods, but instead extensive data is available. This situation is encountered in many places in industrial manufacturing. Production facilities are continuously monitored by various sensors so that ML processes can be triggered, action plans can be generated and then executed, resulting in continuous optimizations of production processes. In this talk, first a brief introduction to the topics of artificial intelligence and ML is given, together with highlighting the benefits and limitations. This will be followed by an introduction of basic ML principles. This is combined with providing insights to a large number of real-world use cases solved at Fraunhofer IPA together with different manufacturing companies.

Figure 1.1: Prof. Dr. Marco Huber (University of Stuttgart/IPA Fraunhofer)

AI-Powered Defect Segmentation in Industrial CT Data

Tim Schanz[1], Robin Tenscher-Philipp[2], Fabian Marschall[3], Martin Simon[4]

[1] Tim Schanz (M.Sc)
tim.schanz@h-ka.de
[2] Robin Tenscher-Phlipp (M.Sc)
robin.tenscher-philipp@h-ka.de
[3] Fabian Marschall (B.Eng)
fabian.marschall@h-ka.de
[4] Martin Simon (Prof. Dr.-Ing.)
martin.simon@h-ka.de

Hochschule Karlsruhe - Technik und Wirtschaft
University of Applied Sciences
Fakultät für Maschinenbau und Mechatronik
Moltkestr. 30
76133 Karlsruhe

Abstract. Non-destructive quality testing using CT plays an important role in industrial quality assurance. However, manual analysis of the large voxel data sets is not efficient. AI-powered processes have already shown that successful segmentation of defects in industrial voxel data is possible. In this paper, we show an AI-solution to detect small defects in industrial CT data. Therefore, we propose two new network architectures, PU-Net and PCU-Net based on an U-Net architecture. For this purpose, we first conducted a parameter study to determine the parameters with the greatest impact on the segmentation performance and incorporated them into the new architecture. In addition, we improved a reference dataset by introducing a data augmentation and also improved the annotation of the real data in this dataset. The evaluation of the new architectures showed very good results.

Keywords: AI; CT; INDUSTRIAL; CNN; DEEP LEARNING

1 Introduction

With ever increasing demand for resource-saving and thus more environmentally friendly production, the need for optimized processes and rising quality is growing. In modern production systems for highly stressed or safety-relevant components, the internal condition of the part is of high importance for quality assurance. The method of choice is Computed Tomography (CT), which is one of the most important non-destructive testing methods. The CT provides 3D image data of the scanned component, consisting of volumetric pixels (voxels). The gray values of the voxels correlate with the density of the part. Thus, defective areas can be determined based on the gray value. Until now, this process has required a trained worker to inspect the components individually. There are also algorithmic software solutions which lack in reliability because of the user influence on manually set thresholds. In this process, artifacts and pixel defects complicate the work. Particularly in inline applications, where the CT is integrated into a production process, the quality of the images suffers. As a result, algorithmic methods for detecting defects on the basis of gray value distributions are not an option.

With the help of artificial intelligence, it is possible to automize these tasks in order to save personnel in production. This is particularly important in view of the shortage of skilled workers, which will become even more acute in the coming years. The progress of image processing with neural networks in medical technology can be adapted to industrial applications. With the U-Net architecture [1] [2], a promising possibility for the segmentation of medical CT data was created, which is used in a modified way. This u-shaped architecture was the baseline for several new derivatives for image segmentation. For industrial CT data analysis researchers [3] [4] also used the classical U-Net architectures and applied changes to fulfill the task. In this paper, besides the architecture, the data set will be considered in detail. Here, the representation and distribution of the pores, the variance of the data as well as the generation of the synthetic data are important. Because training such a neural network requires a very large amount of data, which is difficult to obtain for various reasons. A major challenge is that real industrial CT data is not publicly available because companies don't distribute their CT data and even if they make them public the data must be annotated by an expert. In addition, the effort for annotation is enormous and requires trained personnel. It is very likely that some defects will be incorrectly annotated or even completely overlooked.

In a previous work, we have already published different approaches for the segmentation of CT data, which yielded promising results [5]. The goal of this work is to further improve the performance of the Network. In the following, these methods are presented.

2 CT data for training and evaluation

In this work we consider the detection of defects described as pores. These are gas inclusions that are perceptible as dark spots in the 3D volume data. Their gray value is therefore lower than the surrounding material. In this research pores can have a minimum extent of 2^3 voxel and up to of 10^3 voxel, depending the resolution of the CT scanner in microns per voxel. The very small size of the defects leads to a problem with class balance, because the number of voxels assigned to the background is very large compared to the voxels of the defect class. For example, in a training dataset of ~10.000 samples, a resolution of 64^3 per sample, an average of 35 pores per sample and an average size of 3^3 voxel per pore, the pores take an average of 0.36% of one whole volume. Thus, loss functions as well as metrics must be used to account for this problem. In addition to this, care can be taken when generating the synthetic data to oversample the errors in order to address this problem. For the reasons already mentioned, only a small amount of annotated real data is available for the training. To enable the training, a considerably larger number of synthetically generated data are added to the real data. The approach for the generation is algorithmic and is approximated as closely as possible to the real data. Where the main feature is the grey scale gradient from material to the inner of the pore as well as the shape. During the data development and the data generation process, it is necessary to have an eye on the network, the network architecture and the application, which the network must fulfill, to further improve the quality and the authentic representation of synthetic data.

To further increase the variation in the industrial CT data set used in [5] (hereafter referred to as reference data set refICTDS) [5], Table 1), various augmentation methods were used. With data augmentation, the variation of the 3D data could be significantly increased by randomly applying rotation, flipping, cropping and elastic deformation on the complete dataset. The ratio between real and synthetic data stays the same but the overall amount of data is increased with higher variation to address the data issue. This will also variationally multiply our real data to a useful amount. With these methods we were able to create a dataset (ICTDS) (Table 1) with around 16000 samples where 156 available samples of real data are augmented to receive 329 samples, and 7300 synthetic samples are augmented to receive 15000 samples. The number of needed samples can be chosen, as well as split in training data, validation data and test data. Additionally, the annotation of the real data samples was improved.

Table 1 Overview of used datasets

Name	Res	No. of samples	No. of real samples included	No. of training samples	No. of evaluation samples	No. of test samples
refICTDS	64^3	7405	156	6334	703	368
ICTDS	64^3	16190	318	12666	1408	2116

The following Figures show a comparison of real and synthetic samples of the two datasets. Figure 1 and Figure 2 are from the refICTDS dataset. Figure 3 and Figure 4 belong to the augmented ICTDS dataset where the massive impact of elastic deformation can be seen. The deformation leads especially in the synthetic dataset to a much more realistic visualization of the pores and guarantee a higher variation in geometric shapes.

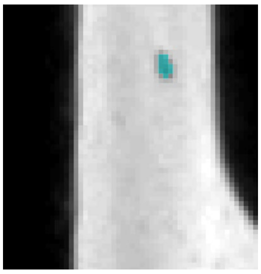


Figure 1: Real data symple refICTDS

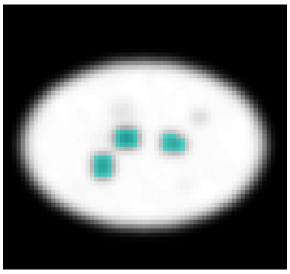


Figure 2: Syntehtic data sample refICTDS

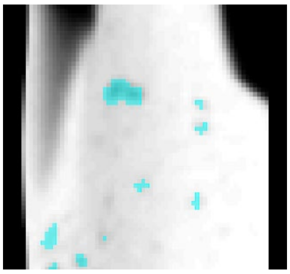


Figure 3: Real data refICTDS with elastic deformation

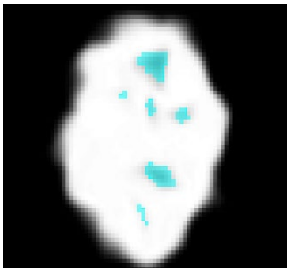


Figure 4: Synthetic data ICTDS with elastic deformation

3 Hardware and software setup

Training our models with different parameters and large 3D datasets requires specific hardware. In our case we are using a PC system with a Nvidia RTX Titan (24GB VRAM), 64 GB system memory and an Intel Core i9-10940X (14 cores). To train a model we used Tensorflow and needed to guarantee, that the model and data fits into the VRAM. To reserve the GPU memory for the model, we used the Tensorflow dataset API. This allows us to provide the path to the data and load it batchwise on the fly from HDD into VRAM when the previous batch is processed. For this purpose, an own separate nested load function has to be developed where the sample and groundtruh volume could be loaded together.

4 Neural network architecture

In our publication [5] different classical neural network architectures like U-Net [2] and V-Net [6] were compared with respect to object segmentation. Based on the earlier shown results, this project was started and developed on the basis of the best architecture U-Net-Gdata [5] (hereafter referred to as refU-Net). Thus, the models in this work are mainly based on the U-Net architectures. As described in [1] [2] [7], the encoder consists of several convolutional layers followed by down sampling, which can be performed in several different ways. This results in the input image being encoded into feature representations at several different levels. The deeper a model is, the more relevant features are extracted and the less significant ones are discarded. But if small details matter to finally distinguish between the classes this could lead to misclassification/-segmentation. The decoder semantically projects the low-resolution features learned by the encoder onto the higher-resolution pixel space to obtain dense pixel-wise segmentation. The decoder consists of up sampling and concatenation followed by regular convolution operations. The up sampling is also needed to obtain a segmentation with the resolution of the input image. The concatenation in the decoding path is important to marry classification with the localization obtained from the encoder at the corresponding level. The number of convolutions per depth level determine combined the filter kernel size the field of view for every voxel. Which means we are able to summarize larger features to a single value with more convolutions per layer and larger kernel sizes. To extend the results and progress shown, the classic U-Net architecture is used and significantly developed. The optimization is based purely on performance and detection probability. So, we investigate different parameters like encoding and decoding steps, number of convolutions per step, kernel size and filter amount, with and without batch normalization and dropout as well as activation functions [8] [9] and losses. The number of filters in the decoder is normally doubled every encoding step. A higher amount of feature maps could lead to a higher variation in separable features. The size of the filter kernels is adjusted depending on the spatial size of the features to learn. To train a model the model needs to be compiled where the optimizer algorithm, the loss function and evaluations metrics are set. These are some of the most important parameters, but the sheer number of variations of all these parameters is enormous.

Target ranges could be defined for most of the hyperparameters by performing some restrictive examinations. Thus, a theoretical parameter field can be spanned, in which optimal combinations of the parameters should be settled. With the help of the theoretical preliminary work, the enormous variety can be counteracted. The final solution is a combination of the theoretical parameter field and an hyperparameter optimization. The hyperparameter tuning is targeted against the evaluation metric which is the binary mean IoU. Additionally, to plain hyperparameter optimization we evaluated several hyperparameter to analyze which parameter has the highest influence in reaching a high evaluation score. For that we evaluated every single parameter. We also evaluated different loss functions based on a survey for image segmentation [10] [11]. The higher the score achieved with a specific hyperparameter variation, the higher the influence for our application and with our data. Also, the factor of score change depending on changing to another hyperparameter value within the evaluated set is taken into account. In addition, the correlation of the parameter change in dependence on the change of the detection probability must be correlated. In the following diagram (Figure 5), the most important parameters are shown. The farer outside the respective parameter is, the larger is the change in this parameter in terms of the achievable score.

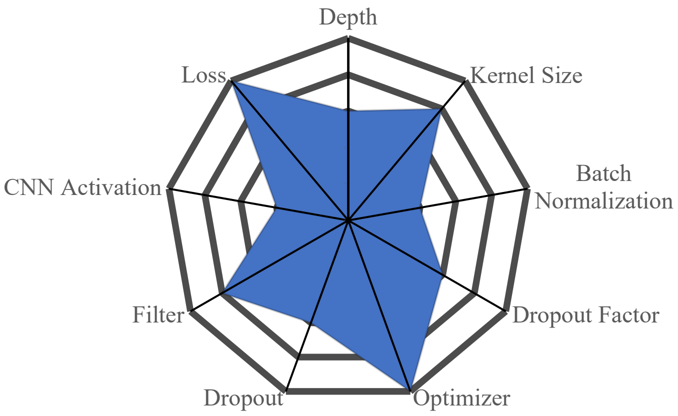


Figure 5: Impact of different hyperparameters on achievable IoU score

With this evaluation we were also able to determine specific values for every evaluated parameter which led us to the following values shown in Table 2.

Table 2: Determined hyperparameter set

Depth	**Filter base**	**Kernel size**	**Layer activation**	**Dropout**	**Dropout factor**	**BatchNorm**	**Optimizer**	**Loss function**
3	16	3 and 5	ReLU	False	0.1	True	Adam	Dice

In the following Table 3 we show a parameter comparison between the refU-Net and the two best models developed in this work. We moved the activation from the convolutional layer to a separate activation layer after normalization and created a parallel U-Net (PU-Net) model where we used the shown layer order per depth level in parallel with the kernel sizes 3 and 5 and concatenated the feature maps before down pooling. It has been shown [12] [13] [14] [15] that a parallelization of convolutions with different kernel sizes help to train features appearing in different sizes where as these mechanisms often used for natural language processing and training the feature maps for classification tasks. We made use of this technique to take into account that pores have volumetric sizes between 2^3 and 10^3 voxels. In the parallel paths we applied 2 convolutions to enlarge the receptive field of view for the resulting feature maps.

Table 3: Hyperparameters layer order of U-Net of previous work and new developed models

Model versions	**Down steps**	**Conv layers per depth lvl**	**Kernel size**	**Layer activation**	**Layer order per depth lvl**	**Loss function**	**Params**
refU-Net	3	2	3	ELU	Conv (incl. Act), Conv (incl. Act), MaxPool	BCE	1.4Mio
PU-Net	3	2	3+5	ReLU	Conv, BN, Act, Conv, BN, Act, Concat, MaxPool	Dice	6.3Mio
PCU-Net	3	2	3+5	ReLU	Conv, BN, Act, Concat, Conv, BN, Act, Concat, MaxPool	Dice	7.1Mio

The second model we developed, parallel cross U-Net (PCU-Net) (Figure 6), we concatenated the parallel generated feature maps after the first convolution for feature map mixture before leading into the second parallel convolutions. The parallelization is only used in the encoder path.

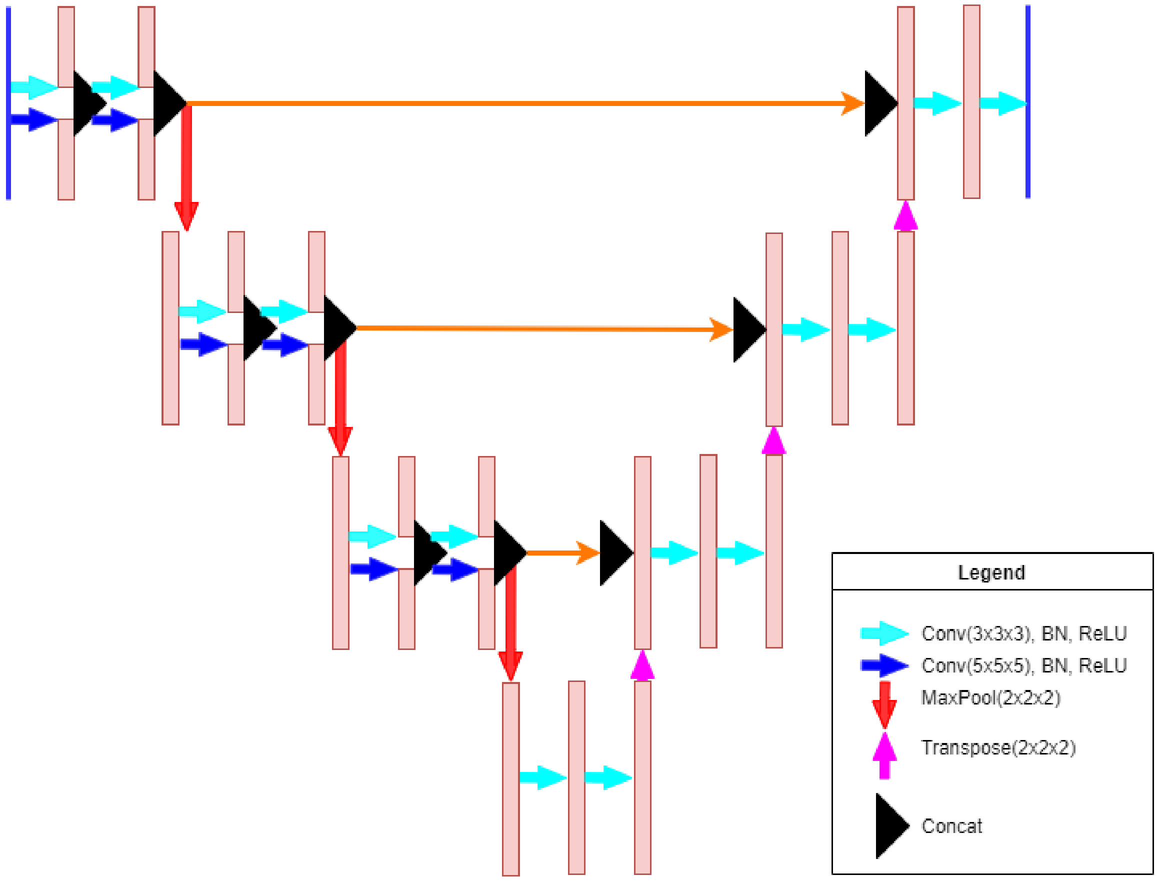


Figure 6: Schematic of our proposed PCU-Net

5 Training and evaluation of the neural networks

After defining the most impacting parameters, evaluating the best values for each parameter and building up the model architectures, the models were trained with different datasets. The datasets and their properties are described in chapter 2. During the training callbacks for early stopping, checkpointing, learning rate reduction and logging are used. At the beginning of the trainings, a learning rate of 0.001 was used.

Table 4 shows our results for the different models trained on different data sets (mixed and synthetic) in comparison to the -Net. First, the networks were trained with the reference dataset refICTDS to verify the changes and differences to the reference model. Both of our two new models PU-Net and PCU-Net achieved a higher BIoU score on the reference dataset. The PCU-Net achieved the highest value of 99.876% [BIoU]. In the confusion matrix (Figure 7) we can see that the false predicted background (419 voxel) and the false predicted pores (207) voxel is vanishig small compared to correct predicted background an pore voxels which shows an even more distictive good result. Consequently, an increase of 7.301% [BIoU] could be reached just by the new architectures and their parallelization of the convolutions. Thus, a first confirmation of the improvement of the new architecture has been achieved.

Table 4: Model evaluation on all types of samples of reference data set refICTDS. Tag in brackets describe which part of the dataset was used.

Model	**Training dataset**	**Evaluation dataset**	**BIoU [%]↑**
refU-Net	refICTDS (synth)	refICTDS (mixed)	92.575
PU-Net	refICTDS (mixed)	refICTDS (mixed)	99.838
PCU-Net	refICTDS (mixed)	refICTDS (mixed)	**99.876**

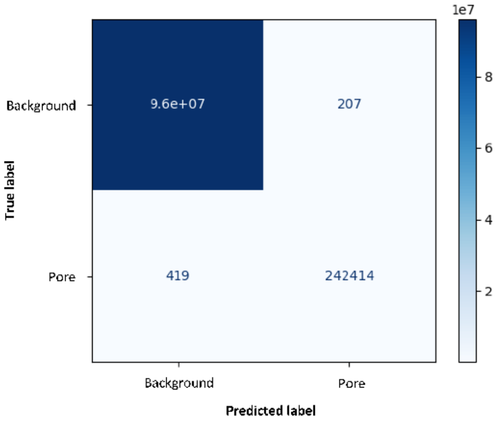


Figure 7: Confusion matrix of PCU-Net evaluation on refICTDS (mixed)

In the second step we trained the models on our new improved dataset ICTDS. With our new paralell architectures we achieved reasonably higher IoU scores where as models trained on the new dataset have slightly lower scores. This behavior can be traced back to the more extensive and much more complex dataset (ICTDS) dataset. The results (Table 5) varify that even on the new dataset the architectures performe better that refU-Net with 93.017% for PU-Net and 93.567% for PCU-Net. In the confusion matrix (Figure 8) we can see again vanishing less false predictions for both classes.

Table 5: Model evaluation on all types of samples of dataset ICTDS. Tag in brackets describe which part of the dataset was used.

Model	**Training dataset**	**Evaluation dataset**	**BIoU [%]↑**
refU-Net	ICTDS (mixed)	ICTDS (mixed)	90.762
PU-Net	ICTDS (mixed)	ICTDS (mixed)	93.017
PCU-Net	ICTDS (mixed)	ICTDS (mixed)	**93.567**

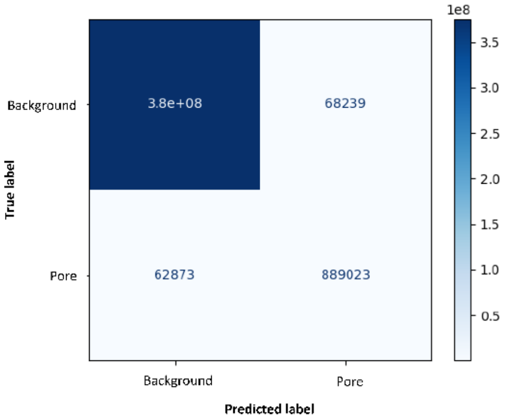


Figure 8: Confusion matrix of PCU-Net evaluation on ICTDS (mixed)

We also evaluated the models against refICTDS dataset (Table 6) and against ICTDS dataset (Table 7) with only real samples . We can see that there is still a reality gap between synthetic and real data but our improved architectures, which can be deducted from Table 6 and data, which can be deducted from Table 7, achieved higher scores than we achieved in our prevous work [5].

Table 6: Model evaluation on real samples of dataset refICTDS. Tag in brackets describe which part of the dataset was used.

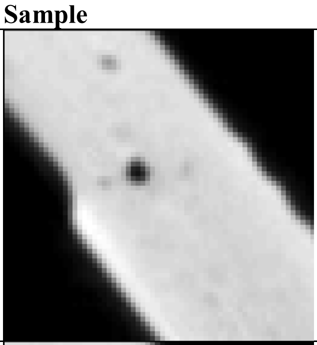
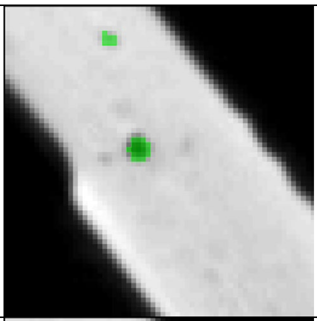
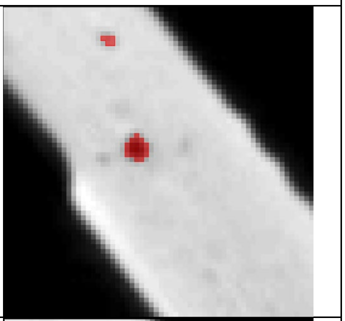
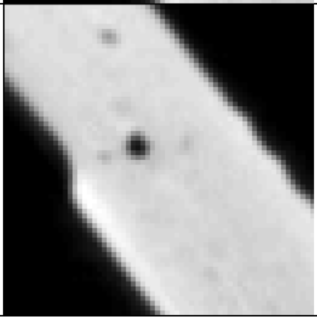
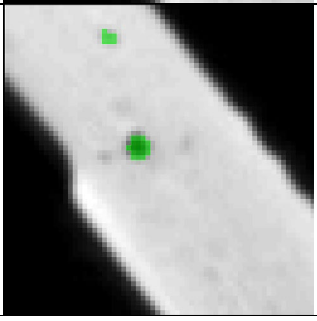
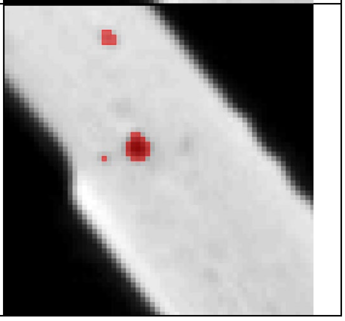
Model	**Training dataset**	**Evaluation dataset**	**BIoU [%]↑**
refU-Net	refICTDS (mixed)	refICTDS (real)	55.285
PU-Net	refICTDS (mixed)	refICTDS (real)	61.881
PCU-Net	refICTDS (mixed)	refICTDS (real)	**62.504**

Table 7: Model evaluation on real samples of dataset ICTDS. Tag in brackets describe which part of the dataset was used.

Model	**Training dataset**	**Evaluation dataset**	**BIoU [%]↑**
refU-Net	ICTDS (mixed)	ICTDS (real)	59.958
PU-Net	ICTDS (mixed)	ICTDS (real)	62.305
PCU-Net	ICTDS (mixed)	ICTDS (real)	**63.899**

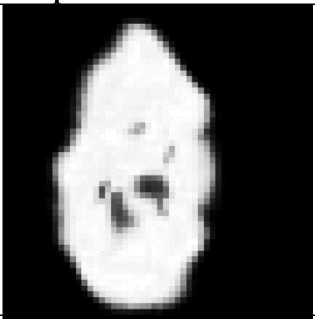
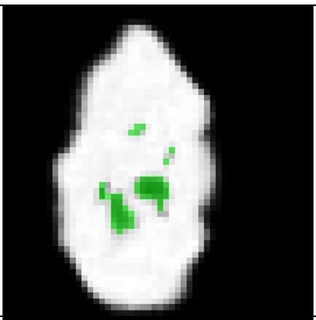
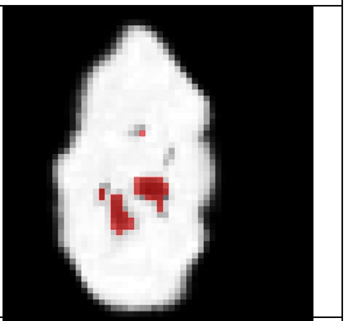
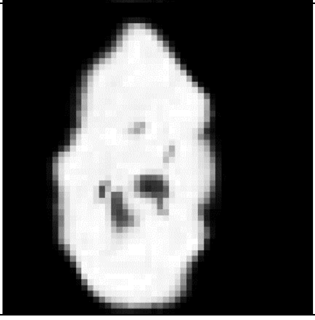
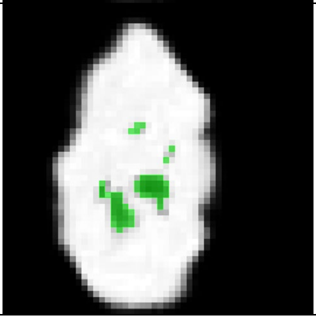
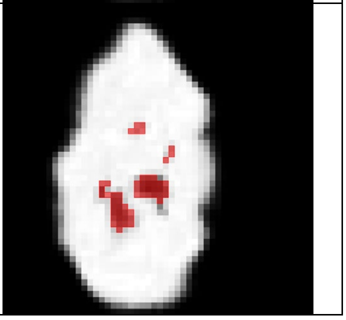
In order to make the pure numerical results more tangible, individual sections through samples are shown in the following overview (Table 8). This makes the results of refU-Net and PCU-Net visually comparable. The samples are taken from the evaluation set of ICTDS. In the comparison on the first sample which is real data we can see that our PCU-Net has a slightly more accurate segmentation prediction. As mentioned in the introduction, human accuracy in annotating data is not always consistent. This is because the decision whether a defect is present and which voxels belong to it are subjective. The PCU-Net model found an aditional pore right next to the large one which was not annotated but seems as valid segmentation. This is because the model learns from a large variety of data which is in case of real data an average of experts decission.

Table 8: Visual comparison of prediction results of refU-Net and PCU-Net taken from ICTDS evaluation dataset. Real data sample.

Model	**Sample**	**Groundtruth**	**Prediction**
refU-Net			
PCU-Net			

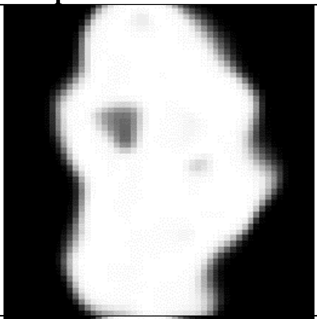
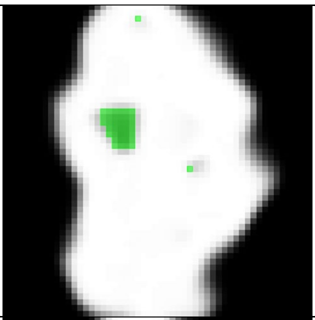
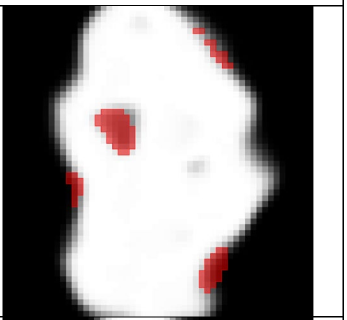
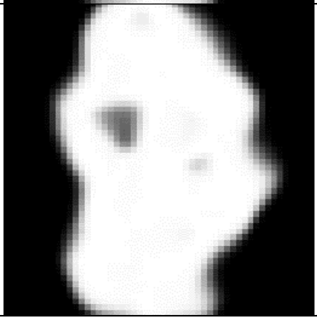
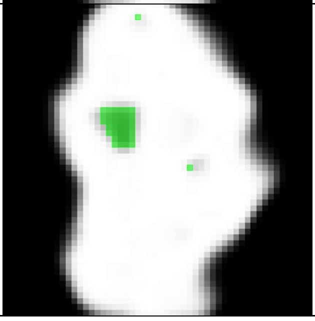
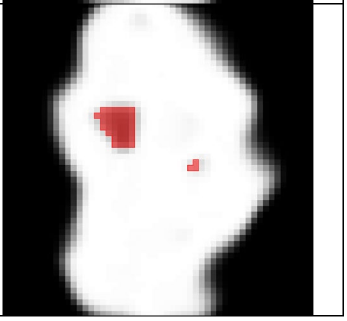
The second sample (Table 9) has highly deformed pores. What seems very unrealistic here is in reality often found in the overlaying of pores in pore nests or due to the deformation of pores. Additionally the visualisation of small structures could be influenced by noise in CT data, artifacts during reconstruction or cannot be resolved cleanly. Also here both models were able to segment most of the pores correctly but again slightly better results achieved by PCU-Net.

Table 9: Visual comparison of prediction results of refU-Net and PCU-Net taken from ICTDS evaluation dataset. Synthetic data sample.

Model	Sample	Groundtruth	Prediction
refU-Net			
PCU-Net			

In the last sample (Table 10) we can see that refU-Net has massive false predictions at the surface of the object which does not occur with our new model. Both models achieved reasonable results in the evaluation of over 90% but relying on just a metric could be dangerous because outliers in large evaluation sets vanisch in the metric. If the model refU-Net would be used in an application for example in a production processes it would be important to manualy inspect the quality. If such massive segmentation faults are automaticaly evaluated this could lead to waste of supposedly defective components in production. In some cases a basic algorithmic plausibility check could avoid wasting parts. Nevertheless, AI-based pre-processing of data can significantly reduce the effort required for manual quality assurance.

Table 10: Visual comparison of prediction results of refU-Net and PCU-Net taken from ICTDS evaluation dataset. Synthetic data sample with artifacts predicted from refU-Net.

Model	Sample	Groundtruth	Prediction
refU-Net			
PCU-Net			

6 Conclusion

In this work, we proposed two neural network models PU-Net and PCU-Net that are able to predict pore segmentation in industrial CT data. To determine the optimization possibilities, we evaluated the most important parameters and defined the most promising parameter values. These results allowed us to further develop a reference model to derive two new architectures. To verify our new models, we trained them on a reference dataset, which showed that the improvements were valid (BIoU of 99.876 %). In the next step, we created a new dataset ICTDS to further improve the prediction quality of PU-Net (BIoU of 93.017 % mixed data and BIoU of 62.305 % real data) and PCU-Net (BIoU of 93.567 % mixed data and BIoU of 63.899 % real data). Unfortunately, there is no publicly available industrial CT dataset with this type of defects to evaluate our results against other proposals, but with our stepwise evaluation and improvement strategy, we have shown that one way to successfully improve AI networks on specific data is to first improve the model by training with an existing dataset to verify the model improvements, and then use a new dataset. This brings us one step closer to AI-based defect segmentation of small defects in industrial CT data. With an achievable defect segmentation accuracy of ~63%, our model already provides good pre-segmentation for CT application engineers evaluating production data and saving time for quality assurance. The accuracy can be further enhanced with more realistic synthetic and additional real data. In technical tasks which should solved with AI unbalanced data is often the case. Using a methodology like proposed here could be beneficial on solving these problems. Furthermore, our method could be applied on similar industrial segmentation tasks or even for 2D analysis. Our method could be applied on similar industrial segmentation tasks or even for 2D analysis. In summary this paper shows that our AI solution achieved promising results solving automated non-destructive quality inspection needs.

7 References

[1] O. Ronneberger, P. Fischer und T. Brox, „U-Net: Convolutional Networks for Biomedical Image Segmentation,“ 18 May 2015.

[2] Ö. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox und O. Ronneberger, *3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation,* arXiv, 2016.

[3] P. Fuchs, T. Kröger und C. S. Garbe, „Self-supervised Learning for Pore Detection in CT-Scans of Cast Aluminum Parts,“ *International Symposium on Digital Industrial Radiology and Computed Tomography, 2 – 4 July 2019 in Fürth, Germany (DIR 2019),* November 2019.

[4] P. Fuchs, T. Kröger, T. Dierig und C. S. Garbe, „Defect Detection in CT Scans of Cast Aluminum Parts: A Machine Vision Perspective,“ *9th Conference on Industrial Computed Tomography (iCT) 2019, 13-15 Feb, Padova, Italy (iCT 2019),* March 2019.

[5] T. Schanz, R. Tenscher-Philipp und M. Simon, „AI-Powered Analysis of Industrial CT Data,“ 2020.

[6] F. Milletari, N. Navab und S.-A. Ahmadi, „V-Net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation,“ 15 June 2016.

[7] N. Siddique, P. Sidike, C. Elkin und V. Devabhaktuni, „U-Net and Its Variants for Medical Image Segmentation: A Review of Theory and Applications,“ *IEEE Access,* Bd. PP, pp. 1-1, June 2021.

[8] D. P. Kingma und J. Ba, „Adam: A Method for Stochastic Optimization,“ 22 December 2014.

[9] C. Nwankpa, W. Ijomah, A. Gachagan und S. Marshall, „Activation Functions: Comparison of trends in Practice and Research for Deep Learning,“ 8 November 2018.

[10] S. Jadon, „A survey of loss functions for semantic segmentation,“ *2020 IEEE International Conference on Computational Intelligence in Bioinformatics and Computational Biology,* June 2020.

[11] S. S. M. Salehi, D. Erdogmus und A. Gholipour, „Tversky loss function for image segmentation using 3D fully convolutional deep networks,“ 18 June 2017.

[12] X. Chen, B. Xu und H. Lu, „Effects of Parallel Structure and Serial Structure on Convolutional Neural Networks,“ in *Journal of Physics Conference Series*, 2021.

[13] H. W. Fentaw und T.-H. Kim, „Design and Investigation of Capsule Networks for Sentence Classification,“ *Applied Sciences,* Bd. 9, p. 2200, May 2019.

[14] L. Jia, H. Zhai, X. Yuan, Y. Jiang und J. Ding, „A Parallel Convolution and Decision Fusion-Based Flower Classification Method,“ *Mathematics,* Bd. 10, p. 2767, August 2022.

[15] A. Krizhevsky, I. Sutskever und G. E. Hinton, „ImageNet Classification with Deep Convolutional Neural Networks," in *Advances in Neural Information Processing Systems*, 2012.

[16] S. R. Dubey, S. Chakraborty, S. K. Roy, S. Mukherjee, S. K. Singh und B. B. Chaudhuri, „diffGrad: An Optimization Method for Convolutional Neural Networks," *IEEE Transactions on Neural Networks and Learning Systems, 2019,* 12 September 2019.

Deep Learning based classification of vocal folds' vibration dynamics

Mona Kirstin Fehling[1,2,3], Maximilian Linxweiler[2], Bernhard Schick[2], and Jörg Lohscheller[1]

[1]Department of Computer Science, University of Applied Sciences Trier, Schneidershof, Trier, Germany
[2]Department of Otorhinolaryngology, Head and Neck Surgery, Saarland University Hospital, Homburg/Saar, Germany
[3]Department of Otorhinolaryngology, Head and Neck Surgery, University Hospital Mannheim / Medical Faculty Mannheim of Heidelberg University, Mannheim, Germany

Abstract. Vocal fold (VF) dynamics can be captured in real-time using high-speed videolaryngoscopy, laying the basis for quantitative assessment of the VFs vibration properties. A compact representation of the vibrational behavior as captured in these high-speed videos (HSV) is provided by the so-called Phonovibrogram (PVG). The PVG encodes the VFs vibrational behavior by characteristic spatial and temporal patterns in a three-dimensional representation. Based on these characteristic PVG patterns, this work realizes a fully automatic classification of different voice disorders. For this purpose, a Convolutional Neural Network (CNN) was trained and evaluated using a stratified 10-fold cross-validation strategy on PVGs from 220 subjects to solve two different classification tasks: (a) Classification of the vibrational behavior as *physiologic* or *pathologic* and (b) classification of the PVGs according to the subjects actual clinical diagnosis as *healthy*, *muscle tension dysphonia (MTD)*, *paresis*, or *polyp*. The trained CNN distinguished with an average classification accuracy of 0.82 ± 0.07 between *physiologic* and *pathologic* VF vibration (sensitivity: 0.81 ± 0.12, specificity: 0.82 ± 0.12) and achieved an average classification accuracy of 0.85 ± 0.07 across all classes (sensitivity: 0.71 ± 0.19, specificity: 0.91 ± 0.07) for classification according to the clinical diagnoses. Based on the PVG representation, the presented approach reliably differentiates between physiologic and pathologic VF vibration and is even eligible to distinguish types of voice disorders without user interaction. However, to further increase the method's performance, a larger amount of training data is required.

Keywords: vocal fold vibration, voice disorders, high-speed video, Phonovibrogram, classification, deep neural network

1 Introduction

Voice disorders emerge from disturbances in the vocal folds (VFs) vibrational behavior [1]. At any time, about 7.6% of the adult population in the USA is affected by any sort of voice disorder, with a lifetime prevalence of voice disorders being 30% [2, 3].

Quantitative assessment of the VFs' vibrational behavior can be done in real time using high-speed videolaryngoscopy [4, 5]. High-speed videolaryngoscopy captures the vibrating VFs at frame rates of 2,000 - 20,000 fps [6], allowing for detection of even slight variations in VF vibration. However, the huge amount of recorded data makes evaluation of the captured high-speed videos (HSVs) time-consuming and thus challenging during voice assessment in clinical routine [7]. Moreover, visual assessment and rating of the HSVs, demand an experienced clinician to make a proper diagnosis [8]. Therefore, various approaches were presented to assist the clinicians with voice assessment providing a compact and clinically meaningful representation of the relevant information on VF vibration as contained in the HSVs [9–16].

One of the so far most comprehensive approache is the so-called Phonovibrogram (PVG). The PVG provides a compact and clinically meaningful three-dimensional representation of the VFs vibrational behavior contained in an HSV sequence [11]. Figure 1(a) shows the construction process of the PVG; a detailed description of the PVG construction can be found in Lohscheller et al. [11]. To build the PVG, the glottal area is initially segmented in the subsequent HSV frames. Based on this segmentation result, the glottal symmetry axis is defined between the posterior (P) and the anterior (A) end of the glottis. Afterwards, the glottal symmetry axis is used to split the retrieved glottis contour into two halves representing the left and the right VF edge. That following, the distances between the contours and the glottal symmetry axis are computed for all positions along the glottal symmetry axis. Color-coding the computed distances results into one color stripe per frame. The PVG visualization is finally built by temporal concatenating all these color stripes. Spatio-temporal patterns in the PVG encode the VFs vibrational behavior and show characteristic patterns for physiologic as well as pathologic VF vibrations. Exemplarily, PVGs from a healthy subject, a subject with muscle tension dysphonia (MTD), a subject with unilateral VF paresis, and a subject diagnosed with a VF polyp are shown in Figure 1(b).

PVG-based classification of voice disorders was so far done by training Support Vector Machines (SVMs). Suitable features were retrieved from the PVG and partly supplemented with additional features from the acoustical

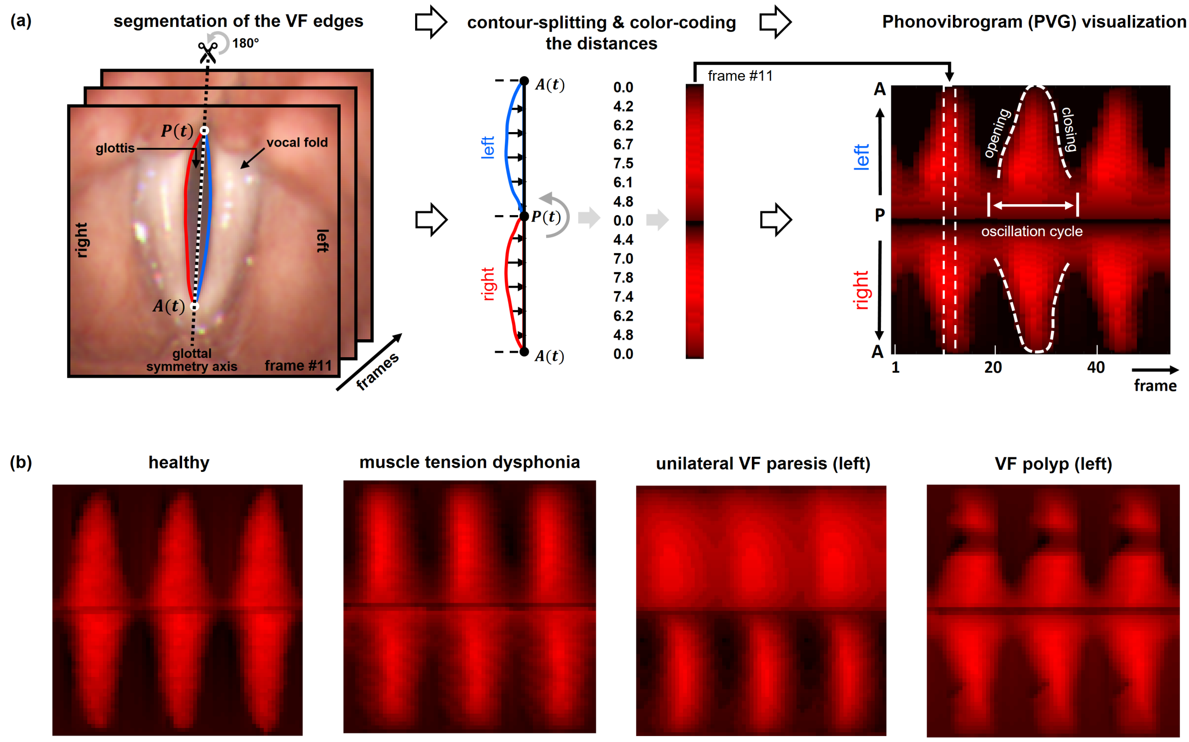


Fig. 1: Phonovibrogram (PVG) visualization of VF vibration. (a) Computation of the PVG. The VF edges are retrieved from each HSV frame based on the segmented glottis. Then, the contour of the left VF edge is rotated at 180°, followed by computation of the distances between the respective VF edge and the glottal symmetry axis and color-coding them. Concatenation of the resulting color stripes results in the PVG representation of the VFs spatiotemporal vibrational behavior. (b) Examples of PVGs computed from clinical HSVs: Healthy subject, subject with muscle tension dysphonia (MTD), subject with unilateral VF paresis (left VF), and subject with a polyp on the left VF.

voice signal.

By employing such an SVM-based classification approach, Voigt et al. achieved an average classification accuracy of 0.744 on a 2-class discrimination task by retrieving features solely from the PVG (healthy vs. muscle tension dysphonia patients) [17].

In another work, Voigt et al. achieved an average classification accuracy of 0.93 on a SVM-based classification of PVGs as either healthy or paralytic [18].

Unger et al., on the other hand, achieved an accuracy of 0.69 ± 0.02 on a four-class classification task (healthy vs. MTD vs. paresis vs. polyp) by training an SVM on a 12-dimensional feature set [8]. In that study, the PVG-based feature set was retrieved by using a combined wavelet- and PCA-based approach for dimensionality reduction on the initially 1000 frames long PVG sequences. The features retrieved from the PVG describe the glottal closure type, the phase information, the asymmetry, and the irregularity of VF vibration.

In this work, a fully automatic classification of voice disorders based on characteristic PVG patterns is presented, which for the first time is realized by using a deep Convolutional Neural Network (CNN).

2 Material and Method

Figure 2 provides an overview of the conducted study by showing the dataset used, the employed CNN architecture, and the performed training and evaluation procedure. Details on the clinical data and the trained CNN are provided in the following.

2.1 Data

Clinical data were collected from the *Department of Otorhinolaryngology, Head and Neck Surgery* at the *Saarland University Medical Center* (Homburg/Saar, Germany) within a previous study [8]. Ethical approval was obtained from the local ethics committee (*Ethik-Kommission bei der Ärztekammer des Saarlandes*, reference number:

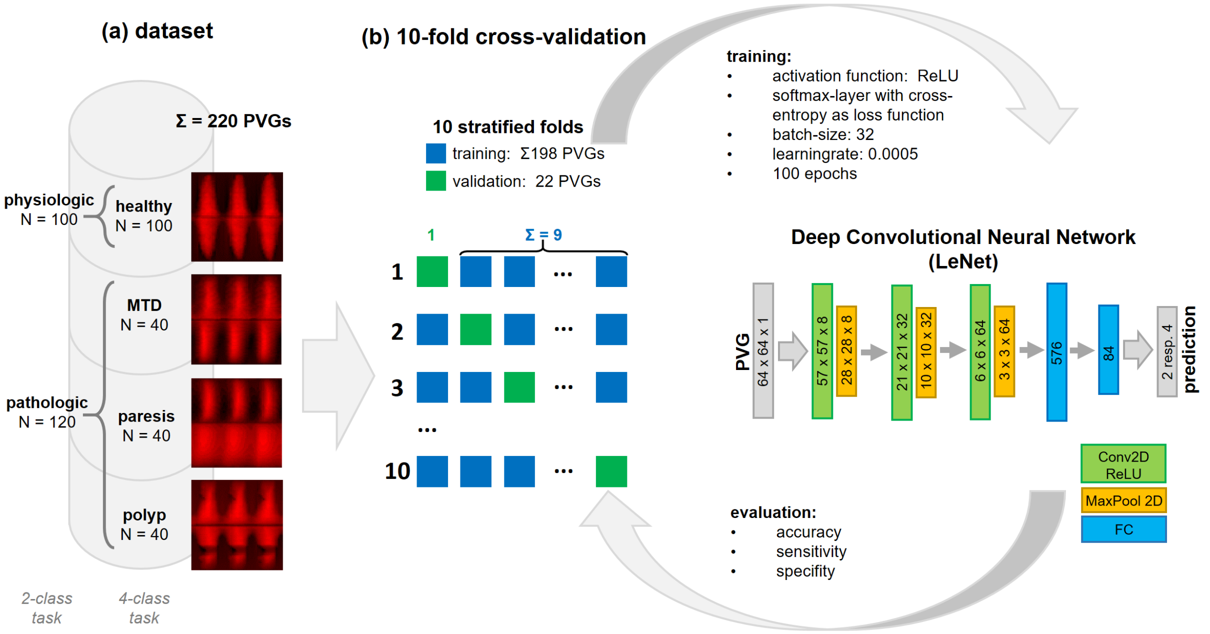


Fig. 2: Overview on the conducted study. (a) Composition of the dataset used, and (b) schematic representation of training and evaluation procedure. The PVGs were classified with a CNN (based on the LeNet) using a stratified 10-fold cross-validation strategy.

103/12), and the participants gave their consent prior to participation. Laryngeal HSVs were recorded with a 90° tip and in color using the rigid endoscopy system *HRES ENDOCAM 5562* from *Richard Wolf GmbH* (Knittlingen, Germany). The HSVs were captured with a spatial resolution of 256 × 256 px and at a frame rate of 4,000 fps. All participants were asked to perform a sustained phonation of the vowel /ae/ at comfortable pitch and loudness for at least 1 s during the examination.

In total, PVGs from clinical HSV recordings acquired from 220 individual subjects comprising four clinical groups *healthy*, *muscle tension dysphonia (MTD)*, unilateral VF *paresis*, and unilateral VF *polyp* were included in the dataset. The dataset comprises $N_{healthy} = 100$ recordings from healthy subjects, $N_{MTD} = 40$ recordings from subjects diagnosed with muscle tension dysphonia, $N_{paresis} = 40$ subjects with paresis, and $N_{polyp} = 40$ subjects with polyp.

To avoid any bias caused by the subjects' individual fundamental frequencies, three complete oscillation cycles were extracted from each PVG. The VF-deflections encoded in each PVG extract were normalized to the interval $[0; 1]$ and the PVGs were futher rescaled to $64px \times 64px$.

2.2 Classification Tasks

A deep CNN architecture is used to classify the PVGs. This study investigated two different classification scenarios: (a) First, a CNN was trained to classify the VFs vibrational behavior as represented by the PVG as either *physiologic* or *pathologic* (2-class task). For differentiation between physiologic and pathologic VF vibration, the patients diagnosed with MTD, paresis, and polyp were combined to the class *pathologic* ($N_{pat} = 120$), while the healthy subjects compose the class *physiologic* ($N_{phys} = 100$). (b) In a second step, a CNN was analogously trained to classify the PVGs according to their actual clinical diagnosis as *healthy*, *MTD*, *paresis*, or *polyp* (4-class task).

2.3 Architecture

This study used a CNN based on the LeNet-architecture [19], a comparatively small and elementary CNN architecture for computer vision tasks. Contrary to the original LeNet, we used as input single PVGs of size $64px \times 64px$, ReLU activations instead of tanh and sigmoid activations [20]. Moreover, dropout layers were used to avoid overfitting and improve classification performance despite the small dataset [21].

As depicted in Fig. 3, the CNN used here consists of seven layers in total: the input layer, three convolutional layers, two fully connected layers, and an output layer. These layers build two functional parts: a feature extractor

and a classifier.
The feature extractor is built from the convolutional layers, where each convolutional layer consists of convolutions with varying kernel sizes and dropout, ReLU as a nonlinear activation function, and 2D-max-pooling. The extracted features are then classified by the classifier that is composed of two fully connected layers which are activated with ReLU and a final linear softmax layer that returns the class probabilities for the respective number of classes as $out\_features$. The properties of the consecutive CNN layers are as follows (Fig. 3):

```
LeNetOpt64DropOut(
  (feature_extractor): Sequential(
    (0): Conv2d(1, 8, kernel_size=(8, 8), stride=(1, 1))
    (1): Dropout(p=0.2, inplace=False)
    (2): ReLU()
    (3): MaxPool2d(kernel_size=2, stride=2, padding=0,
         dilation=1, ceil_mode=False)
    (4): Conv2d(8, 32, kernel_size=(8, 8), stride=(1, 1))
    (5): Dropout(p=0.2, inplace=False)
    (6): ReLU()
    (7): MaxPool2d(kernel_size=2, stride=2, padding=0,
         dilation=1, ceil_mode=False)
    (8): Conv2d(32, 64, kernel_size=(5, 5), stride=(1, 1))
    (9): Dropout(p=0.2, inplace=False)
    (10): ReLU()
    (11): MaxPool2d(kernel_size=2, stride=2, padding=0,
          dilation=1, ceil_mode=False)
  )
  (classifier): Sequential(
    (0): Linear(in_features=576, out_features=84, bias=True)
    (1): ReLU()
    (2): Linear(in_features=84, out_features= {2 or 4}, bias=True)
  )
)
```

Fig. 3: Properties of the consecutive CNN layers.

2.4 Training & Evaluation.

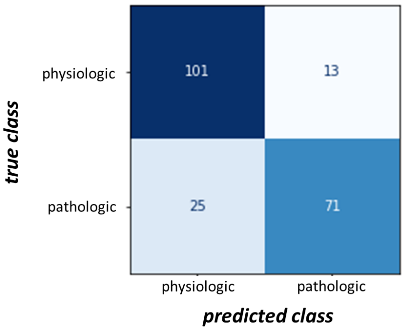
In order to find suitable hyperparameters, a train-validation-split pre-experiment with randomly but stratified selected 198 PVGs as training data and 22 PVGs as validation data was conducted on the 2-class as well as on the 4-class task.
The CNN was then trained and evaluated using a 10-fold cross-validation strategy (c.f. Fig. 2(b)). Each fold comprises a stratified training set of 198 PVGs and a stratified validation set of 22 PVGs. Based on the findings of the pre-experiment, for both classification tasks, the CNN was trained with a $batch\_size$ of 32 and over 100 $epochs$. As loss function, an according to the class frequencies weighted cross-entropy was used, with an Adam Optimizer and a $learning\_rate = 0.0005$. The dropout probability for the convolutions was set to 0.2.
The predictive power of the CNN was assessed after each fold and for the individual classes using accuracy, sensitivity, and specificity. The overall predictive power of the CNN was finally evaluated by considering all folds.

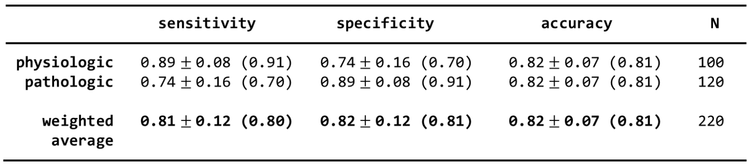
3 Results

This work realized a classification of Phonovibrograms using a CNN. We investigated two different classification tasks: (a) Differentiation between *physiologic* and *pathologic* VF vibration (2-class task) and (b) PVG Classification according to the actual diagnosis as *healthy*, *MTD*, *paresis*, or *polyp* (4-class task).

(a) Differentiation between *physiologic* and *pathologic* VF-vibration

The CNN trained to distinguish between physiologic and pathologic VF-vibration achieved over all 10 folds an average accuracy of $ACC_{2-class}^{avg} = 0.82 \pm 0.07$ (0.81) with an average sensitivity of $SEN_{2-class}^{avg} = 0.81 \pm 0.12$ (0.80) and an average specificity of $SPEC_{2-class}^{avg} = 0.82 \pm 0.12$ (0.81) [all values indicated as: mean ± standard deviation (median)]. The respective confusion matrix, as well as the detailed results, are depicted in Figure 4.





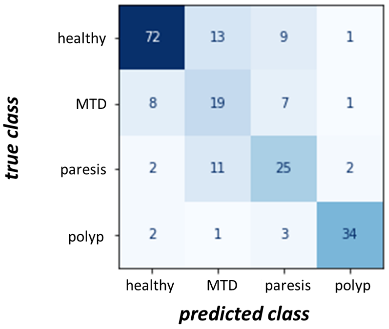
	sensitivity	specificity	accuracy	N
physiologic	0.89 ± 0.08 (0.91)	0.74 ± 0.16 (0.70)	0.82 ± 0.07 (0.81)	100
pathologic	0.74 ± 0.16 (0.70)	0.89 ± 0.08 (0.91)	0.82 ± 0.07 (0.81)	120
weighted average	**0.81 ± 0.12 (0.80)**	**0.82 ± 0.12 (0.81)**	**0.82 ± 0.07 (0.81)**	220

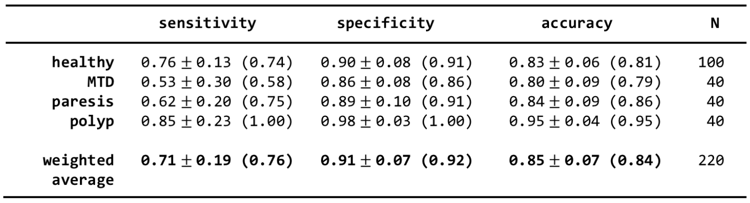
Fig. 4: Results on the differentiation between physiologic and pathologic VF vibration. Classification results were retrieved from all 10 folds and incorporated the entire dataset. The confusion matrix on the left visualizes the frequency of predicted classes for the two classes, *physiologic* and *pathologic*, individually. Detailed information on classification performance as measured by the parameters sensitivity, specificity, and accuracy is given inside the table. Moreover, overall performance is given by the weighted average of the respective measures, where class-individual measures were weighted with class frequencies to avoid any bias by unequal class sizes. Values are stated as: mean ± standard deviation (median), N states the class frequency.

Physiologic VF vibration patterns were classified correctly with a sensitivity of $SEN_{phys} = 0.89 \pm 0.08\ (0.91)$, while the specificity for these subjects was $SPEC_{phys} = 0.74 \pm 0.16\ (0.70)$. On the other hand, sensitivity for the pathologic VF-vibration patterns was reduced to $SEN_{pat} = 0.74 \pm 0.16\ (0.70)$ with a specificity of $SPEC_{pat} = 0.89 \pm 0.12\ (0.81)$.
In order to systematically investigate the misclassifications and to analyze whether these confusions might be related to individual pathologies, in a second step a CNN was trained on the classification of PVGs according to their actual clinical diagnosis.

(b) PVG Classification according to the actual diagnosis

When classifying the PVGs according to their actual diagnosis, the CNN achieved over all 10 folds an average accuracy of $ACC^{avg}_{4-class} = 0.85 \pm 0.07\ (0.84)$ with a sensitivity of $SEN^{avg}_{4-class} = 0.71 \pm 0.19\ (0.76)$ and an average specificity of $SPEC^{avg}_{4-class} = 0.91 \pm 0.07\ (0.92)$. Detailed results and the respective confusion matrix are shown in Figure 5.





	sensitivity	specificity	accuracy	N
healthy	0.76 ± 0.13 (0.74)	0.90 ± 0.08 (0.91)	0.83 ± 0.06 (0.81)	100
MTD	0.53 ± 0.30 (0.58)	0.86 ± 0.08 (0.86)	0.80 ± 0.09 (0.79)	40
paresis	0.62 ± 0.20 (0.75)	0.89 ± 0.10 (0.91)	0.84 ± 0.09 (0.86)	40
polyp	0.85 ± 0.23 (1.00)	0.98 ± 0.03 (1.00)	0.95 ± 0.04 (0.95)	40
weighted average	**0.71 ± 0.19 (0.76)**	**0.91 ± 0.07 (0.92)**	**0.85 ± 0.07 (0.84)**	220

Fig. 5: Results on PVG classification according to the actual clinical diagnosis. Classification results were retrieved from all 10 folds and incorporated the entire dataset. The confusion matrix on the left visualizes the frequency of predicted classes for each of the four classes *healthy*, *MTD*, *paresis*, and *polyp*. Detailed information on classification performance for the four classes as measured by the parameters sensitivity, specificity, and accuracy are given inside the table. Moreover, overall performance is given by the weighted average of the respective measures, where class-individual measures were weighted with class frequencies to avoid any bias by unequal class sizes. Values are stated as: mean ± standard deviation (median), N states the class frequency.

On this classification task, the class *polyp* outperformed all other classes and in all three measures ($ACC_{polyp} = 0.95 \pm 0.04$ (0.95), $SEN_{polyp} = 0.85 \pm 0.23$ (1.00), $SPEC_{polyp} = 0.98 \pm 0.03$ (0.92)). Organic lesions on the VFs, like polyps, lead to characteristic local alterations in the PVG pattern (c.f. Fig. 1(b)). Moreover, as pathologies with organic lesions are represented in this study only by the class *polyp*, high sensitivity and high specificity for this class indicate that these local PVG alterations caused by the polyp might help to detect this pathology.

On the other hand, PVGs for the remaining three classes *healthy*, *MTD*, and *paresis* mainly differ in their degrees of symmetry and regularity of the VFs vibrational behavior (c.f. Fig. 1(b)), making differentiation between these classes harder. This fact also becomes clear from the classification results, as confusions often occurred between the classes *healthy* and *MTD* as well as between the classes *MTD* and *paresis* (c.f. confusion matrix in Fig. 5). Nevertheless, when comparing these three classes, the class *healthy* outperformed both pathologic classes, in both sensitivity and specificity ($SEN_{healthy} = 0.76 \pm 0.13$ (0.74), $SPEC_{healthy} = 0.90 \pm 0.08$ (0.91)). This is likely due to the different sized classes where much more cases were available from healthy subjects than from pathological cases ($N_{healthy} = 100$ vs. $N_i = 40$ for all pathologic groups *i = MTD, paresis, polyp).*

4 Discussion

This work presents a fully automatic classification of the VFs vibrational behavior from PVGs using a CNN. Based on the PVG representation, the presented approach has shown to differentiate between physiologic and pathologic vibrational behavior reliably and has also demonstrated reliable classification of the PVG according to the four considered clinical diagnoses.

On a 2-class discrimination task between physiologic and pathologic VF vibration from PVG, Voigt et al. achieved an average classification accuracy of 0.744 when the pathologic subjects were MTD patients [17] and average classification accuracy of 0.93 when the pathologic subjects were paralytic [18]. While the here presented CNN-based approach with an average accuracy of $ACC^{avg}_{2-class} = 0.82 \pm 0.07$ outperforms the SVM-based classification of healthy vs. MTD patients, it shows clearly reduced performance when compared to the SVM-based classification of healthy vs. paralytic patients. These differences in classification performance might be due to the fact that in the here presented work, the pathologic class consisted of subjects with three different diagnoses. And as these diagnoses present themselves totally differently in the PVG, making the classification task somewhat harder. Moreover, by only 220 PVGs, the dataset is relatively small compared to datasets usually used to train a CNN comprising thousands of samples. This is also limiting the achievable classification performance.

However, when considering the actual clinical diagnosis, on the 4-class classification task, the here presented Deep Learning based approach with an average accuracy of $ACC^{avg}_{4-class} = 0.85 \pm 0.07$ outperformed the previous approach by Unger et al., which achieved an average accuracy of $ACC^{avg}_{SVM} = 0.69 \pm 0.02$ [8]. These results are directly comparable as both works are based on the identical data set.

Confusions between the classes occured mainly between *healthy* and *MTD* subjects as well as between *MTD* and *pareses*. All these three classes share that there are no organic lesions located at the VFs altering the PVG representation. Instead, the pathologies (*MTD* and *paresis*) are mainly characterized by asynchronities and asymmetries in the VFs vibrational behavior. These asynchronities and asymmetries are presumably harder to detect as irregularities marked by characteristic alterations in the PVG patterns caused by organic lesions, i.e. by a polyp (c.f. 1 (b)). This is especially relevant for such a comparatively small dataset as the one used in this study.

Despite the small dataset, the CNN showed overall promising results. However, much more data are needed to extend this approach further and increase classification performance. Acquisition of additional clinical data and refinement of the here presented approach will be a focus of future works.

5 Conclusion

Due to its time-consuming nature, visual assessment of the VFs vibrational behavior as comprised in the HSVs is not feasible in clinical routine and moreover demands an experienced clinician.

In this work, we presented for the first time an approach that employs a deep Convolutional Neural Network to fully automatically classify different types of voice disorders. The presented approach is based on the VFs vibrational behavior as encoded in the PVG representation and has shown to reliably differentiate between physiologic and pathologic vibrational behavior. It is also eligible to classify different types of voice disorders without the need of any user interaction.

More training data will be required to increase the classification performance further. With an increased amount of training data, in the future it will be possible to realize more complex and thus even more powerful CNNs.

Acknowledgments

Computational resources were provided by the High Performance Compute Cluster 'Elwetritsch' at the University of Kaiserslautern, which is part of the 'Alliance of High Performance Computing Rheinland-Pfalz' (AHRP). We kindly acknowledge the support.

References

1. Titze, I.R.: Principles of Voice Production. National Center for Voice and Speech (2000)
2. Roy, N., Merrill, R.M., Gray, S.D., Smith, E.M.: Voice Disorders in the General Population: Prevalence, Risk Factors, and Occupational Impact. The Laryngoscope **115**(11) (nov 2005) 1988–1995
3. Bhattacharyya, N.: The prevalence of voice problems among adults in the United States. The Laryngoscope **124**(10) (may 2014) 2359–2362
4. Mehta, D.D., Hillman, R.E.: Voice assessment: updates on perceptual, acoustic, aerodynamic, and endoscopic imaging methods. Current Opinion in Otolaryngology & Head & Neck Surgery **16**(3) (June 2008) 211–215
5. Hertegård, S.: What have we learned about laryngeal physiology from high-speed digital videoendoscopy? Current Opinion in Otolaryngology & Head and Neck Surgery **13**(3) (#jun# 2005) 152–156
6. Deliyski, D.D., Hillman, R.E.: State of the Art Laryngeal Imaging: Research and Clinical Implications. Current Opinion in Otolaryngology & Head and Neck Surgery **18**(3) (#jun# 2010) 147–152
7. Lohscheller, J., Eysholdt, U.: Phonovibrogram Visualization of Entire Vocal Fold Dynamics. The Laryngoscope **118**(4) (#apr# 2008) 753–758
8. Unger, J., Schuster, M., Hecker, D.J., Schick, B., Lohscheller, J.: A multiscale product approach for an automatic classification of voice disorders from endoscopic high-speed videos. 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (jul 2013)
9. Timcke, R., von Leden, H., Moore, P.: Laryngeal Vibrations: Measurements of the Glottic Wave: Part I. The Normal Vibratory Cycle. A.M.A. Archives of Otolaryngology **68**(1) (1958) 1–19
10. Švec, J.G., Schutte, H.K.: Videokymography: High-Speed Line Scanning of Vocal Fold Vibration. Journal of Voice **10**(2) (1996) 201–205
11. Lohscheller, J., Eysholdt, U., Toy, H., Döllinger, M.: Phonovibrography: Mapping High-Speed Movies of Vocal Fold Vibrations Into 2-D Diagrams for Visualizing and Analyzing the Underlying Laryngeal Dynamics. IEEE Transactions on Medical Imaging **27**(3) (#mar# 2008) 300–309
12. Deliyski, D.D., Petrushev, P.P., Bonilha, H.S., Gerlach, T.T., Martin-Harris, B., Hillman, R.E.: Clinical Implementation of Laryngeal High-Speed Videoendoscopy: Challenges and Evolution. Folia Phoniatrica et Logopaedica **60**(1) (#nov# 2007) 33–44
13. Karakozoglou, S.Z., Henrich, N., d'Alessandro, C., Stylianou, Y.: Automatic glottal segmentation using local-based active contours and application to glottovibrography. Speech Communication **54**(5) (jun 2012) 641–654
14. Unger, J., Meyer, T., Herbst, C.T., Fitch, W.T.S., Döllinger, M., Lohscheller, J.: Phonovibrographic wavegrams: Visualizing vocal fold kinematics. The Journal of the Acoustical Society of America **133**(2) (2013) 1055
15. Chen, G., Kreiman, J., Alwan, A.: The glottaltopogram: A method of analyzing high-speed images of the vocal folds. Computer Speech & Language **28**(5) (2014) 1156–1169
16. Herbst, C.T., Unger, J., Herzel, H., Švec, J.G., Lohscheller, J.: Phasegram Analysis of Vocal Fold Vibration Documented With Laryngeal High-speed Video Endoscopy. Journal of Voice **30**(6) (nov 2016) 771.e1–771.e15
17. Voigt, D., Döllinger, M., Braunschweig, T., Yang, A., Eysholdt, U., Lohscheller, J.: Classification of functional voice disorders based on phonovibrograms. Artificial Intelligence in Medicine **49**(1) (may 2010) 51–59
18. Voigt, D., Döllinger, M., Yang, A., Eysholdt, U., Lohscheller, J.: Automatic diagnosis of vocal fold paresis by employing phonovibrogram features and machine learning methods. Computer Methods and Programs in Biomedicine **99**(3) (sep 2010) 275–288
19. LeCun, Y., Boser, B., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W., Jackel, L.D.: Backpropagation Applied to Handwritten Zip Code Recognition. Neural Computation **1**(4) (1989) 541–551
20. Shuang, Y., Gang, W.: Research on Meter Image Recognition Based on Improved LeNet-5. In: Proceedings of the 2020 4th International Conference on Electronic Information Technology and Computer Engineering, ACM (nov 2020)
21. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: A Simple Way to Prevent Neural Networks from Overfitting. The Journal of Machine Learning Research **15**(1) (2014) 1929–1958

Feasibility study to distinguish movements automatically by analysing the pressure distribution on a seat

Alparslan Babur[1,2], Nicolas Dockwiller[2], Yacine Belguermi[2], Ali Moukadem[2], Alain Dieterlen[2] and Katrin Skerl[1]

[1] Furtwangen University
A.Babur@hs-furtwangen.de
[2] Université de Haute Alsace, IRIMAS – Research Institute in Informatics, Mathematics, Automation and Signal

Abstract:

Fatigue is an important factor in the occurrence of car accidents. In Germany, there are an average of almost 2000 car accidents with personal injuries due to tiredness of the driver per year [1]. Automatic detection of fatigue by constantly monitoring a person's condition allows the initiation of emergency braking and therefore reduces the number of car accidents. In this work, a pressure mat was used to record the movements of a driver, simulated by a male healthy volunteer. 18 sitting positions were defined and performed by the volunteer. In total, 103 measurements were evaluated. The results show, that it is feasible to detect movements, when the torso is moving. Movements of the arms without moving the torso were not clearly detectable. However, small differences in the quantitative measurements were detected. Using innovative artificial intelligence algorithms might enable the classification even if there is no torso movement included.

Keywords: pressure mat, position recognition, movement detection

1. Introduction

Sitting is quite a common action in our daily life. Especially lorry and train drivers are sitting almost their entire working time on the driver's seat. According to the German Federal Statistical Office, the number of car accidents in Germany went up by more than 11 % during the years from 2010 to 2019. Tiredness of the driver causes almost 2000 car accidents with personal injuries per year. This results in costs of more than 80 million € [1]. Studies show, that accidents can be prevented by constantly monitoring the driver's health condition [2,3]. Generally, there are 3 classes of parameters that can generate risks: the state of the vehicle, the environment (traffic, weather, etc.) and the physiological state of the driver. Conventional methods of monitoring the driver's activity and movements usually require the installation of a camera into the driver's cab. This might lead to a situation of discomfort by the driver. Analysing the movements using a pressure mat to record the pressure distribution overcomes that problem. In this study, we focus on the evaluation and monitoring of the driver's physiological state. For that, we tried to recognize mundane movements automatically by analysing pressure changes on a pressure mat, without limiting the user's movements or field of vision. The objective was to find out how a person's inclination, posture and additional weight affected pressure distribution on the pressure mat.

1.1 Related Works

With recent technical advancements, pressure sensors are increasingly used in various devices. Pressure mats can measure the interface pressure applied to the subject continuously. The approach to distinguish movements is based on continuous pressure measurements at the hips and thighs of a person.

In our research, we could not find any studies, which dealt with distinguishing movements on a pressure mat for drivers in a seated position. That's why we looked for similar research areas. Elsharif et al. proposed in 2021 a monitoring system that provides a comprehensive system to avoid pressure ulcers and provide proper care. The authors obtained promising results, where sleep posture classification achieved 99.6 % overall accuracy using feed-forward artificial neural networks [4]. Authors in [5] achieved 97.9% accuracy for three posture classifications using artificial neural network. Diao et al. proposed a smart mat system that recognizes sleep posture using deep residual networks. They reached an accuracy up to 95.08 % for the short-term test and up to 86.35 % for posture classification [6]. Channa et al. monitored sleeping postures of sleep apnea patients using pressure mats. They used supervised machine learning algorithms for the collected data and used it for posture identification [7]. Lima et al.

used unsupervised machine learning algorithms for vital sign monitoring. This type of setup has the advantage of reacting only to movement changes, and no offset signal due the person's weight is measured [8].

All the works described above focused on the classification of sleeping postures, which is not applicable to our system that focusses on seated postures.

1.2 Paper Organization

The remaining of this paper is as follows: Section 2 describes the system design and the test scenarios. Section 3 presents the obtained results. Finally, the discussion and conclusion are given in Section 4 and Section 5.

2. Method

This work evaluates the feasibility to detect small body movements on seat on a pressure mat. In this first exploratory work, data was collected on one adult male volunteer. He was 26 years old and weight 90 kg. During the tests, he was asked to take different seating positions on the pressure mat with various additional weights. In addition, the volunteer was asked to maintain normal breathing during measurements.

The measurements were obtained using the TexiMat system (TexiSense, Rue Evariste Galois, 71210 Torcy, France) which is a bi-dimensional pressure mapping sensor, that was placed under the upper torso. The two outer layers are composed of artificial silk (row or column) of conductive fiber while the intermediate has piezoresistive properties. A voltage is applied to one of the outer layers and a voltage reading is applied to the other. The voltage deviation at each zone gives the resistivity of the intermediate layer which is an image of the pressure. TexiMat is composed of three layers (see Fig. 1) and consists of 32 x 32 pressure sensors given a total of 1024 measuring points obtaining a measurement up to a frequency of 10 Hz. The maximum pressure allowed is 34.4 kPa while the resolution is 1/11 Pa. Each measuring point is written into a 32 x 32 Matrix and then applied to a colour scale (see Fig. 1). The resulting image is the pressure distribution [9].

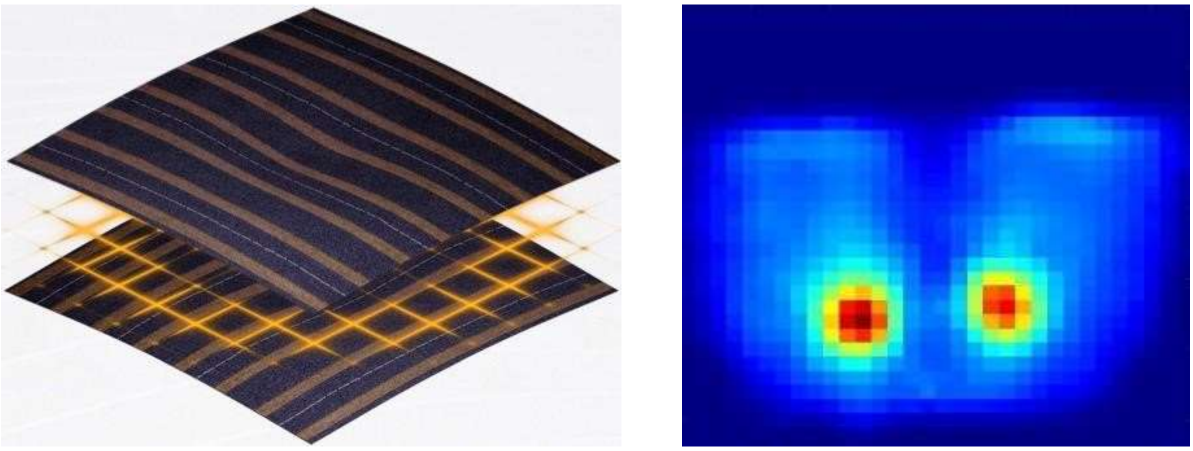


Figure 1. Structure of TexiMat [10] (left) and an example of a pressure distribution (right)

2.1 Influence of inclination

The first examined parameter was the influence of the inclination of a person on the pressure mat measurements. For this, seven positions were tested: no inclination, 3 levels of inclination to the side (slight, strong and very strong inclination), 2 levels of forward inclination (slight and strong) and slight inclination backwards. The scale of the different imprints is not the same, because it is normalized. The normalization ensures, that the maximum pressure is always the greatest value on the scale. This way pressure differences can be evaluated better.

2.2 Influence of additional weight

The weight of a person varies over time. There are differences between morning and evening and the impact of clothes. Thus, in the next test series additional weight was added to the volunteer and the pressure distribution was observed again. The additional weight was increased from 0 kg to 0.5 kg, 1.0 kg, 1.5 kg, 2.0 kg, 3.0 kg, 5.0 kg and finally 10.0 kg by a weighted vest worn by the volunteer.

2.3 Influence of specific postures without the movement of the torso

Last but not least different postures solely by the arms without a clear movement of the torso were examined. These are right arm stretched out, left arm stretched out, both arms stretched forward and both arms stretched upwards.

3. Results

Measurements were taken in real time. In total, 103 measurements were evaluated on the subject and compared with another. It is clear, that movements of the entire upper part of the body are clearly visible by the change of the pressure distribution.

3.1 Influence of inclination

The pressure distributions of the various test scenarios are shown in Fig. 2. In Fig. 2a the volunteer has no inclination. He sits upright and his hands are on his lap. The maximum pressure is in the area of the hips with two centre points coloured in red. The pressure distribution is symmetrical to the vertical axis. The maximum pressure is 49 kPa. Fig. 2b shows a slight inclination to the right side. The pressure on the right side increases and at the same time it decreases on the left side. The pressure is distributed over the surface of the right leg. The maximum pressure is 47 kPa. Stronger inclinations ensure that the pressure distribution shifts even further to the right (Fig. 2c & 2d). The maximum pressure is 46 kPa (Fig. 2c) and 44 kPa (Fig. 2d). A very strong inclination ensures that nearly the total weight is on the right side. A forward inclination ensures, that the pressure distribution is less punctual, but more homogenous. The persons weight is distributed more on the thighs then on the hips. This phenomenon can be clearly seen in Fig. 2f, where the maximum pressure is 19 kPa and occurs on the thighs. Even with a slight inclination the maximum pressure reduces to 22 kPa. A backward inclination behaves similar like a forward inclination. The pressure distribution is less punctual, but more homogenous. The maximum pressure reduces to 22 kPa.

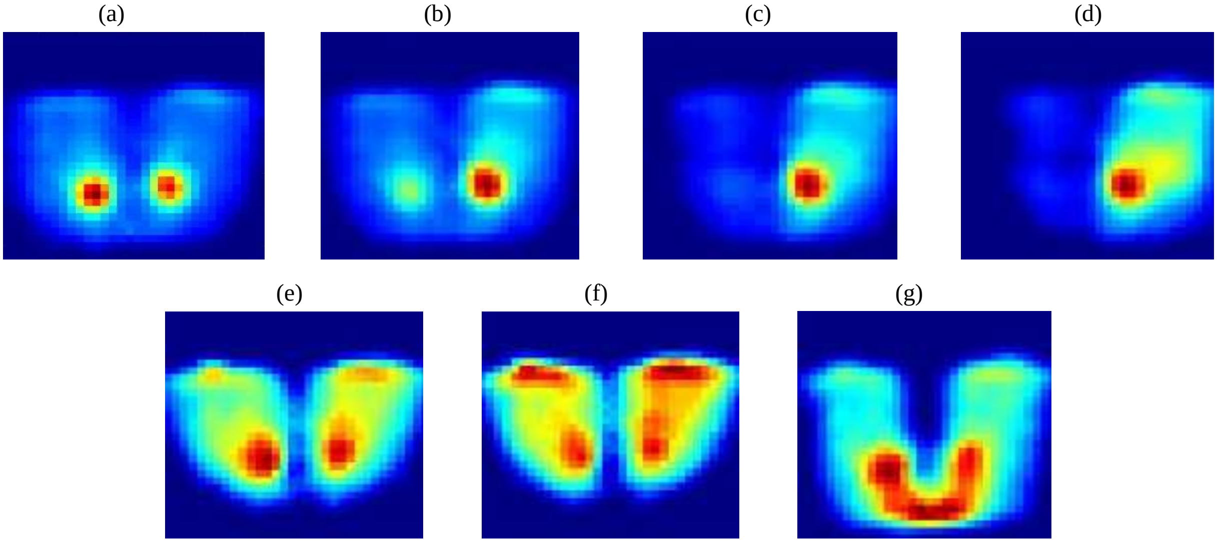


Figure 2. impression of pressure distributions at different inclinations: (a) no inclination, (b) slight inclination to the side, (c) strong inclination to the side, (d) very strong inclination to the side, (e) slight inclination forward, (f) strong inclination forward, (g) slight inclination backwards

3.2 Influence of additional weight

By adding additional weights, the pressures on the entire pressure mat are increased. The pressure distribution on the entire mat becomes more homogeneous and the pressure is again divided over the entire surface (Fig. 3). The red areas, i.e. the maxima, do not change their positions like they did before. It is interesting to see, that the maximum pressure is decreasing, when additional weight is added from 0 kg to 10.0 kg. That could be caused by the more homogenous pressure distribution. The maximum pressure reduces from 49 kPa with 0 kg additional weight to 37 kPa with 10.0 kg additional weight. An addition of 10 kg, could be interpreted as a slight forward inclination, since both scenarios reduce the maximum pressure and shift the pressure distribution more to the thighs.

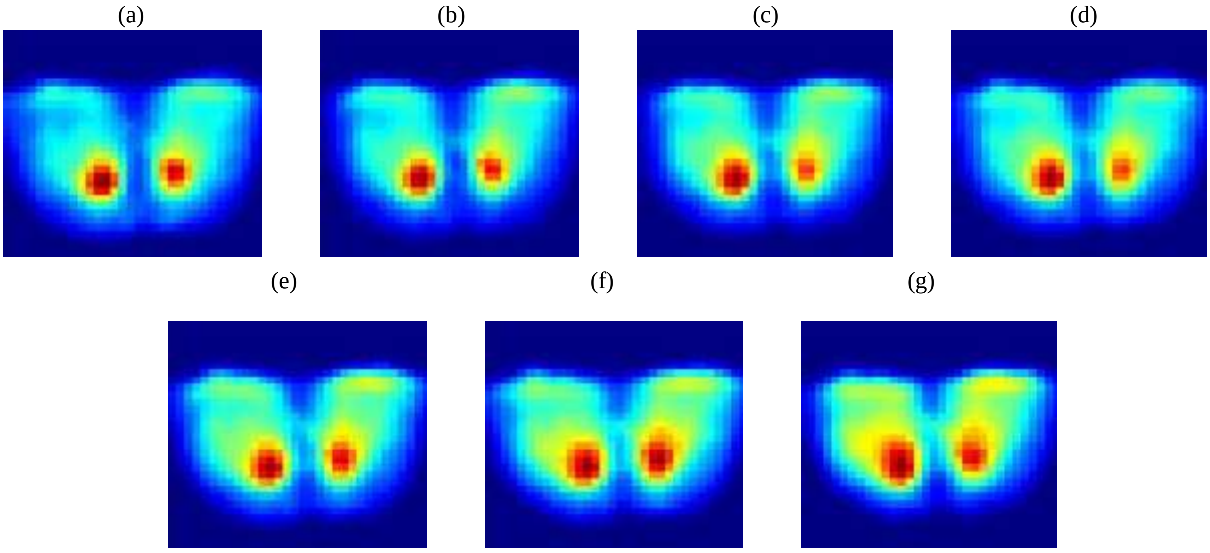


Figure 3. impression of pressure distributions at different additional weights: (a) 0.5 kg, (b) 1.0 kg, (c) 1.5 kg, (d) 2.0 kg, (e) 3.0 kg, (f) 5.0 kg, (g) 10.0 kg

3.3 Influence of specific postures

A movement of the arms without movement of the torso does not lead to an obvious change in the pressure distribution on the mat. Figures 4a, 4b and 4d are nearly identical. However, a difference can be observed by looking at the measured values. When both arms are stretched upwards, the pressure forward (thighs) gets reduced (16 kPa), but the pressure in the centre of the mat (hips) increases (47 kPa). The extension of the arms forward causes the pressure to be shifted forward, like when leaning slightly forward (compare Fig. 4c and Fig. 2e).

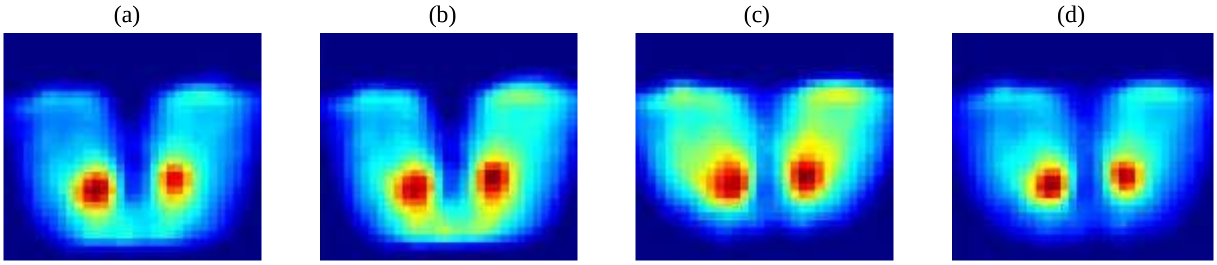


Figure 4. impression of pressure distributions while doing specific postures: (a) right arm stretched right, (b) left arm stretched left, (c) both arms stretched forward, (d) both arms stretched upward

4. Discussion

In this paper we have shown that it is feasible to distinguish movements automatically by analysing the pressure distribution on a pressure mat. We collected data on one adult male volunteer, who was asked to take different seating positions on the pressure mat with various additional weights. For the measurements we used the TexiMat system [10]. During the measurements, the volunteer was asked to take following positions: no inclination, 3 levels of inclination to the side (slight, strong and very strong inclination), 2 levels of forward inclination (slight and strong) and slight inclination backwards. Furthermore, additional weight was added to the volunteer, which was increased from 0 kg to 10.0 kg. Finally, different postures solely by the arms without a clear movement of the torso were examined. The results show, that an inclination to the side ensures a shift in the pressure distribution to the same side. The pressure is distributed over the surface and therefore the maximum pressure decreases. A forward inclination ensures, that the pressure distribution is less punctual, but more homogenous. The persons weight is distributed more on the thighs then on the hips. A backward inclination does the same like a forward inclination, only that the weight shifts to the back instead of the thighs. Adding additional weight increases the pressure on the entire pressure mat. The pressure distribution on the entire mat becomes more homogeneous and the pressure is again divided over the entire surface. A movement of the arms without movement of the torso does not change the pressure distribution significantly. In our research, we could not find many studies, which dealt with distinguishing movements on a pressure mat for drivers in a seated position. That's why we looked for more similar research areas. Several studies focus on classification of sleeping postures on a pressure mat using machine learning algorithms [4-7]. In this first exploratory work, our goal was to find out, if it is possible to find out the seating position with a pressure mat. So, we didn't include any machine learning algorithms yet but aim to include this analysis in the future. In addition, there are other difficulties that may bias the measurements. One of them are the volunteer's feet. By pressing or removing his feet to/from the ground the volunteer can change the pressure distribution on the mat. If the driver presses the gas pedal, the pressure on the right side of the pressure mat increases. As a result, the system detects an inclination to the right. Especially for the evaluation of the driver's health condition who presses the gas and braking pedal regularly, the influence of the foot position needs to be investigated further. The second difficulty is the angle of the spine during an inclination. For every inclination an angle has to be defined as a fix value. At the same angle of inclination, the absolute position of two people of diverse sizes is different. In the case of a tall person, the pressure distribution shift more to the side, than it shifts in the case of a small person. As a result, an inclination to the side can be evaluated incorrectly. This aspect must be considered for future work

5. Conclusion

The results of this study are very promising to automatically classify mundane movements. Although movements of the arms without moving the torso were not clearly detectable in this qualitative evaluation, quantitative differences in the measurements were recorded. The pressure distribution on the mat has symmetries that can be described and define descriptors for further quantitative analyses. Furthermore, using machine learning methodology might overcome this issue and enable the classification of defined movements. Improvements need to be made for future measurements like more data needs to be collected with a larger study group. In addition, more postures need to be included. In the future it would be of interest to record measurements of an entire day period to classify 'normal' situations.

6. References

1. Federal Statistical Office (Destatis). "Verkehr, Verkehrsunfälle, 2020" (2021).
2. Wilkinson, Cassandra M., et al. "Predicting stroke severity with a 3-min recording from the Muse portable EEG system for rapid diagnosis of stroke." Scientific Reports 10.1 (2020): 1-11.
3. Nguyen, Duy-Linh, Muhamad Dwisnanto Putro, and Kang-Hyun Jo. "Eye state recognizer using lightweight architecture for drowsiness warning." Asian Conference on Intelligent Information and Database Systems. Springer, Cham, 2021.
4. Elsharif, Eman, Nabil Drawil, and Salaheddine Kanoun. "Automatic Posture and Limb Detection for Pressure Ulcer Risk Assessment." 2021 IEEE 1st International Maghreb Meeting of the Conference on Sciences and Techniques of Automatic Control and Computer Engineering MI-STA. IEEE, 2021.
5. Matar, Georges, Jean-Marc Lina, and Georges Kaddoum. "Artificial neural network for in-bed posture classification using bed-sheet pressure sensors." IEEE journal of biomedical and health informatics 24.1 (2019): 101-110.
6. Diao, Haikang, et al. "Deep residual networks for sleep posture recognition with unobtrusive miniature scale smart mat system." IEEE Transactions on Biomedical Circuits and Systems 15.1 (2021): 111-121.
7. Channa, Asma, Muhammad Yousuf, and Nirvana Popescu. "Machine Learning Algorithms for Posture Identification of Obstructive Sleep Apnea Patients using IoT Solutions." 2020 International Conference on e-Health and Bioengineering (EHB). IEEE, 2020.
8. Lima, Frederico GC, Almothana Albukhari, and Ulrich Mescheder. "Machine Learning for Contactless Low-Cost Vital Signs Monitoring Systems." From Research to Application (2019): 95.

9. Bui, He Thong. *Modélisation et optimisation de l'assise d'un fauteuil roulant pour handicapé afin d'améliorer le confort d'un point de vue médical*. Diss. Reims, 2018.

10. Texisense, OUR TECHNOLOGY IS EXCLUSIVE https://www.texisense.com/#technologie, Torcy 2022 (12.09.2022 12:00 Uhr)

Image Processing and Neural Network Optimization Methods for Automatic Visual Inspection

Kawther Aboalam, Christoph Neuswirth, Florian Pernau, Stefan Schiebel,
Fabian Spaethe, Manfred Strohrmann

Faculty of Electrical Engineering and Information Technology – Karlsruhe University of Applied Sciences
kawther.aboalam@h-ka.de
manfred.strohrmann@h-ka.de

Abstract. In intelligent production lines, methods of automatic visual inspection are used to continuously record process parameters and production results. Using the quality control of thin film solar modules as an example, this paper shows how visual inspections can be automated by using neural networks. The starting point of this automation is an image of a manufactured solar module generated by the inverse operation of the solar cell and the associated electroluminescence.

It turns out that the amount of data generated with every image is very high despite truncation of irrelevant areas and a reduction of the resolution from 1024 x 1024 pixels to 256 x 128 pixels. Without further preprocessing of the data, A neural network would be built that would have 32768 input nodes due to the pixels acquired.

The Convolutional Neural Networks (CNNs) are usually considered for such image classification problems. However, their use increases the complexity of the architecture of the network and thus the number of parameters to be optimized. Therefore, in addition to automated visual inspection, this work addresses the question of how image processing methods can be used for high-quality and efficient implementation.

The Fast Fourier Transform is used for data preprocessing to enable the use of a multilayer perceptron (MLP) rather than a CNN. It is shown that the computation time can be reduced by a factor of 13 by image preprocessing and using the Fast Fourier Transform to compress the dataset. The reduced computation time is a prerequisite for optimizing the neural network hyperparameters. Particle-Swarm Optimization and Genetic Algorithms are implemented and compared to perform a Neural Architecture Search (NAS) for a MLP and to optimize the other hyperparameters. The methods lead to architectures with which an accuracy of more than 99 % is achieved.

The high accuracy of the presented method recommends it for further projects of automatic visual inspection. The method thus allows digitalization in statistical process control and thus contributes to the implementation of the Quality 4.0 vision.

Keywords: Genetic Algorithms, Image Processing, Particle-Swarm, Automatic Visual Inspection

1 Introduction

Many companies are currently adopting systematic approaches to data analysis with the aim of efficiently implementing comprehensive quality management. Following the term Industry 4.0, which refers to the intelligent networking of machines and processes in industry with the help of information and communication technology, the term Quality 4.0 has been introduced by some companies. The aim is to plan and implement intelligent production lines that monitor themselves and constantly provide information about the current production quality. In this context, the term predictive quality refers to measures that forecast decision-making bases for action measures on the basis of continuously logged process parameters and production results. Statistical evaluations of this logged data lead to statistical process control, with which manufacturing deviations can be immediately detected and eliminated.

An essential element of statistical process control is the continuous evaluation of production quality. In many industries, this evaluation is done by manual visual inspection. If this monotonous and tiring work is carried out over a longer period of time, the concentration of the employees decreases and the reliability of the inspection suffers. For some time, therefore, attempts have been made to automate manual visual inspections. Using the quality control of thin-film solar modules as an example, this paper shows how visual inspections can be automated by using neural networks to classify the quality of the solar modules. The starting point for this automation is the image of a manufactured solar module, which is generated by the reverse operation of the solar cell and the associated electroluminescence. Employees decide on the basis of the image what quality the solar module has and classify it into the categories "OK", "PREMIUM" and "NOK". Bright areas in the image indicate high current densities, dark areas indicate interruptions. Fig.1 compares the images of a solar module of high quality and a solar module with significant manufacturing defects.

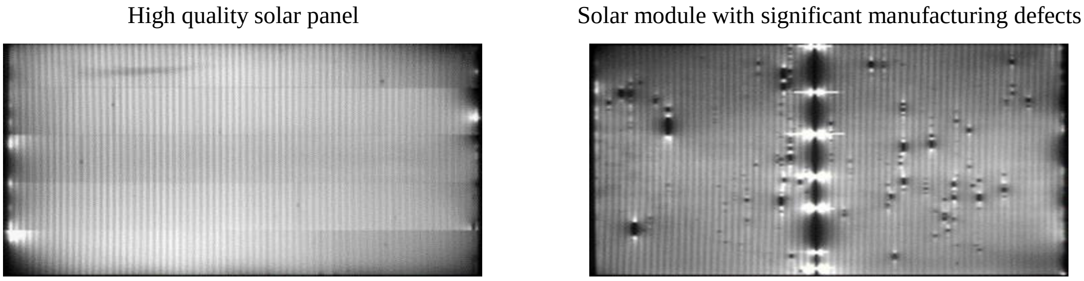


Fig. 1. Electroluminescence of two solar modules of different quality

Employees evaluate the images and classify the manufactured solar modules into classes. Over a longer manufacturing period, images were linked to a quality assessment and stored. The resulting dataset represents the knowledge about this quality assessment and is the starting point for its automation with neural networks.

The paper represents a Multi-Layer-Perceptron (MLP) structure that is optimized using three methods. First, space-filling design, a method of statistical experimental design, is considered. In addition, two methods from statistical optimization are employed, particle swarm optimization and genetic algorithms. The particle swarm optimization is explained in section 4.1

The space-filling design (SFD) - often also called Latin hypercube sampling - is a procedure from statistical experimental design, which is explained below for a two-dimensional experimental space with two hyperparameters (λ_1 ,λ_2). Each hyperparameter is uniformly distributed and has a distribution function F(λ_1) and F(λ_2). In this application, all distribution functions are uniform distributions. If the simulation is to be performed for j = 1 ... J different values, the range of values of the distribution function from 0 ... 1 is divided into J equal intervals for all variables. [1]

Genetic algorithms (GA), like PSO, are nature-inspired algorithms. They are based on Charles Robert Darwin's theory of evolution, which states that life on Earth is subject to constant change. These changes are a result of mutation and selection, with only individuals surviving that are best adapted to their environment. GAs have been developed by John Holland in 1975. [2]

Different representation schemes of GA exist, each leading to different performances in terms of accuracy and computation time. In GA, two common representation methods for numerical optimization problems are distinguished. One is the binary string representation method with a set of binary bits representing a single parameter. The second representation method is to use a vector of real numbers, where each real number represents a single parameter. The GA with real numbers is best suited for optimization in a continuous search space. [3] In this application, each chromosome consists of the hyperparameters described in Table 3.

The structure of the rest of the paper is as follows: After this introduction, Section 2 introduces the image processing and the preparation of the data. Section 3 describes the methodology that has been implemented. Section 4 illustrates the optimization methods for the network architecture. The results and discussions are reported in Section 5. Finally, Section 6 concludes this work.

2 Image Processing and Data preparation

The images of the solar modules generated by electroluminescence are the starting point for the quality assessment. The resolution of these images determines the dimension of the input data for the neural network. The resolution and the dimension of the input data have a significant influence on the quality assessment on the one hand, and on the size of a neural network on the other hand [4]. The aim of image processing and data preparation is to reduce the data to essential information by means of suitable image processing in order to provide the neural network with the required information with a small dimension of input variables.

2.1 Normalization, truncation of irrelevant areas and reduction of resolution

The captured images of the solar modules show strong differences in brightness, which impaired the generalization ability of the neural network. [5] For this purpose, the images are normalized into a number range of 0 ... 255 using a linear scaling.

The resolution of the images acquired is 1024 x 1024 pixels. Some regions in the images show the module frame. These irrelevant areas are selected and removed. The active areas consist of individual strips where individual interruptions or short circuits must be reliably detected. The images can be compressed especially in this direction of these strips. After this first step of preprocessing the resolution is to 256 x 128 pixels.

2.2 Data compression with the Fast Fourier Transform

The image recordings are composed of pixels of different brightness, which form a visible grid structure. Due to the periodicity of the structures, it is convenient to transform the image into the frequency domain using the Fast Fourier Transform. [6]

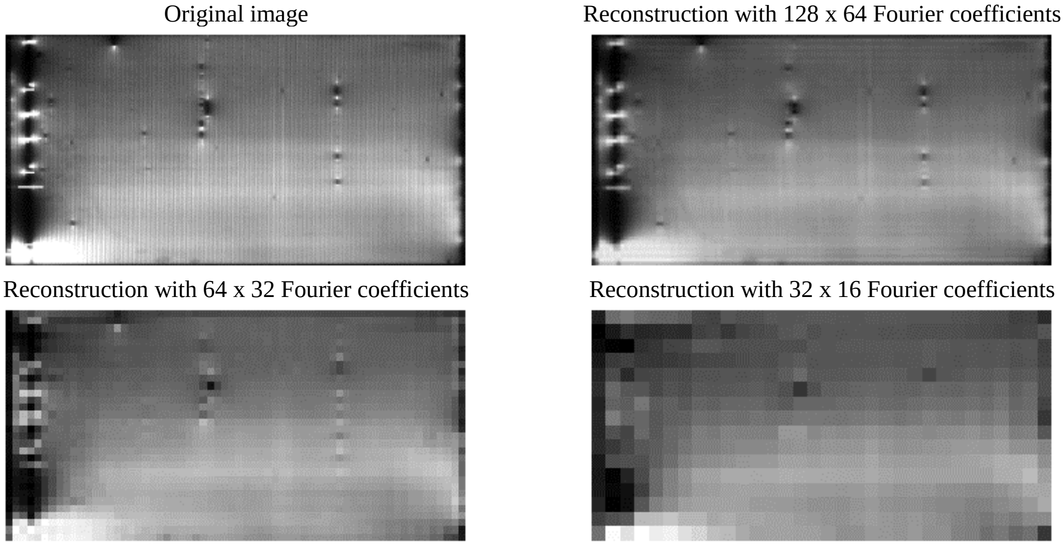
If the captured image has imperfections, the uniform grid structure of the image is disturbed. Small defects with steep light/dark changes cause high-frequency disturbances that lead to high-frequency spectral components. Larger defects with flat light/dark changes, on the other hand, lead to low-frequency spectral components. With the help of the Fast Fourier Transform, these spectral components can be determined with their respective frequencies. Since the image a(m,n) is a two-dimensional signal, the Fourier coefficients A(k,l) of the Fast Fourier Transform are also two-dimensional. If the image has a width of M and a height of N pixels, M x N Fourier coefficients result in the frequency domain.

$$A(k,l)=\sum_{m=0}^{M-1}\sum_{n=0}^{N-1}a(m,n)\cdot e^{-j\cdot 2\cdot\pi\cdot\left(\frac{m\cdot k}{M}+\frac{n\cdot l}{N}\right)} \tag{2.1}$$

The Fast Fourier Transform results in complex Fourier coefficients A(k,l) with magnitude and phase. The magnitude describes the intensity and the phase refers to the spatial orientation of the corresponding harmonic oscillation. The reconstruction of the image is calculated with the inverse Fast Fourier Transform.

$$a(m,n)=\sum_{k=0}^{M-1}\sum_{l=0}^{N-1}A(k,l)\cdot e^{j\cdot 2\cdot\pi\cdot\left(\frac{m\cdot k}{M}+\frac{n\cdot l}{N}\right)} \tag{2.2}$$

Up to M x N complex amplitudes can be used for this purpose. In the present case, however, it is shown as in Fig. 2 that the structures to be identified are already reproduced sufficiently well with 64 x 32 complex coefficients.



Original image

Reconstruction with 128 x 64 Fourier coefficients

Reconstruction with 64 x 32 Fourier coefficients

Reconstruction with 32 x 16 Fourier coefficients

Fig. 2. Original image and reconstructed image with Fourier transforms of different resolution

For quality assessment, the location of defects is of subordinate interest. Therefore, the magnitude of the Fourier coefficients is used for quality assessment only. The number of input nodes is reduced by the transition from 256 x 128 pixels to 64 x 32 real Fourier coefficients.

2.3 Data augmentation

Each image in the dataset was assigned to one of the categories NOK (not okay), OK (okay) and PREMIUM during a manual visual inspection by a staff member. The result was a classified dataset. The classification problem belongs to the category of so-called supervised machine learning. Here, the neural network is repeatedly trained with training data in order to determine a meaningful prediction model for future. If the existing dataset is composed of few, hardly distinguishable images, this impairs the generalization ability of the neural network and leads to so-called overfitting [7]. In overfitting, the network memorizes certain features in the data, making it more intolerant of image data that is not included in the training set. Therefore, the dataset for a neural network should be composed of many different images. In addition, a balanced dataset should be used. By having the same number of image data per category, a uniform recognition rate is achieved for all categories during training. [5]

Another factor for determining the required data volume is the complexity of the neural network used, in particular the number of parameters to be determined. The training of the neural network serves to optimally represent the dataset to be examined by a suitable selection of parameters. For the determination of each parameter, at least one sample is required. The dataset on which the training is based must therefore be correspondingly extensive.

In order to extend the dataset, the existing image data were modified by random but realistic operations and simultaneously duplicated without degrading the informative reliability in the subsequent classification. The original dataset is unbalanced, images with "OK" category are dominant compared to the categories "PREMIUM" and "NOK". Therefore, image augmentations techniques are not applied equally to the three categories. Fewer operations are applied to the category "OK", because it has much more images than the other two categories. Table 1 gives an overview of the concrete operations used.

Table 1. Overview of the operations used for data augmentation

Quality	Operation
NOK Quality insufficient	Shifting the image by two pixels to the right / left and up / down
	Rotation of the image by 3° to the right / left
	Horizontal reflection
	Combination of individual measures
OK Quality sufficient	Shifting the image by two pixels to the right / left
PREMIUM Premium quality	Shifting the image by two pixels to the right / left
	Rotation of the image by 3° to the right / left
	Horizontal reflection
	Combination of individual measures

There is some similarity between the original images and the generated images, but the neural network perceives them as new input data. The described procedure is called data augmentation and allowed an extension of the dataset from originally 83647 images to 490295 images.

3 Methodology

For the visual pattern recognition of images, so-called convolutional nets are usually used. These types of networks use the discrete convolution operations known from signal processing for feature extraction. This reveals patterns in the images, which are used to classify the data [8], their use increases the complexity of the network structure and thus the number of parameters to be determined. Furthermore, due to the complex convolution operations, the training and testing phases are time and computationally intensive. For this reason, this project uses the Fast Fourier Transform for data preprocessing explained in Section 2.2 to enable the use of a MLP network.

3.1 Network architecture and resulting hyperparameters

The dimension of the input data (input) is given by the number of magnitudes of the Fourier coefficients passed as a 1 x 2048 vector of the network input. Since the neural network is to divide the images of the solar modules into the three categories NOK, OK and PREMIUM, three output nodes (output) are required. The number and size of the hidden layers in between (HL: Hidden-Layer) determine the quality of the classification and have to be chosen depending on the task. Based on empirical investigations, a network structure with 5 hidden layers is determined. This results in the network structure shown schematically in Fig. 3. [9]

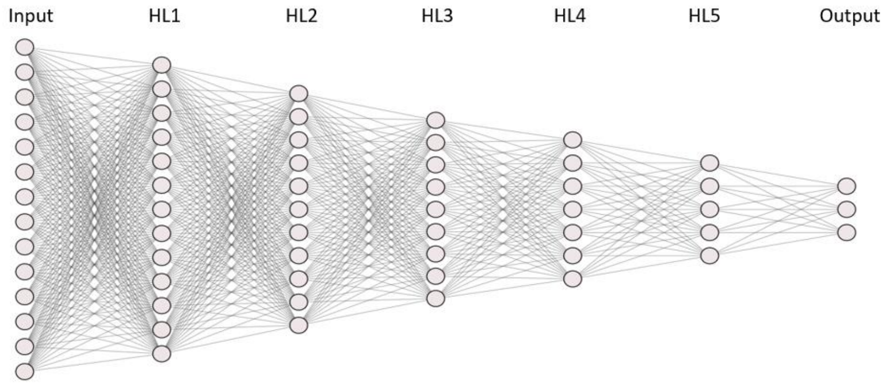


Fig. 3. MLP network structure used with 5 hidden layers

The number of neurons in each hidden layer (HL1 - HL5) has a decisive influence on the quality of the classification [10]. Therefore, they are included as hyperparameters for the optimization of the network.

The Softmax activation function is used for the output layer and the Rectified Linear Unit activation function (ReLU) is used for the hidden layers. The function does not adapt the weights for negative score values because of the vanishing gradient in the training. On the other hand, negative scores are also multiplied by zero and thus excluded from further processing. In [11] it is shown that this implicitly involves regularization. Trials with different activation functions have confirmed that the ReLU function is the best choice for the hidden layers of the network used in this project.

Due to the change of the network parameters, a shift of the output distributions in each layer can occur during the training. As a result, individual weights w_n are no longer updated because they lie outside the functional range of the activation function. This can lead to a worse performance of the neural network. To overcome this problem, so-called batch normalization layers are used. These measure the mean and variance at the output of all hidden layers during the training of the network in each batch. Using adaptive scaling and shifting parameters, the distributions are centered, thus ensuring updating during training. [9]

The selection of a suitable cost function depends on the respective classification problem. In this project, it is a multi-class classification problem with more than two classes, which is why the so-called categorical cross entropy function (Keras) is used. [12]

The optimizer is used to adjust the weights w_n during training so that the loss function reaches its minimum. Depending on the choice of optimizer, different parameters are available for adjustment. Each optimizer has a learning rate α as a hyperparameter, which has a decisive influence on the training process. The learning rate can be optimized during training in order to achieve a faster convergence of the loss function. In preparation for the optimization, optimizers with this adaptive function were tested in advance and the so-called Adadelta optimizer was selected based on the results. In addition to the adaptation of the learning rate, this optimizer has another important hyperparameter, the decay factor ρ. The Decay Factor is used to define which fraction of the previously computed gradients is used in the current computation to update the weights. The information from previous directional derivatives increases the probability of achieving a faster convergence of the error function in the direction of the steepest descent. Learning rate α and decay factor ρ of the chosen Adadelta optimizer have a crucial impact on the performance of the neural network, which is why they are included as hyperparameters to be optimized.

3.2 Training and testing of the neural network

The training of the neural network is divided into epochs. In an epoch, all training images are analyzed once. To keep memory usage and computational capacity efficient during training, all image data in an epoch are divided into batches. The batch size determines after which number of images the weights are updated. After each epoch, the validation dataset is used to check the accuracy of the classification with the currently computed weights and to continue the training if necessary. Finally, in order to evaluate the performance of the network, the network is evaluated with the test dataset at the end of the training. To do this, a dataset must be used that the neural network has not yet run through during training. For these steps, the dataset is divided into training, validation and test datasets (Table 2). Both the number of training epochs and the batch size influence the convergence behavior of the network. For this project, after validation runs, the number of epochs has been set to 100 and the batch size to 128.

Table 2. Division of data into training, test and validation data, total 490 295 images

	Training	**Test**	**Validation**
Distribution	83.3 % of the total	16.7 % of the total	10 % of the test patterns
Number of images	408563	81732	8173

4 Optimization methods for an adequate network architecture

The hyperparameters of a neural network can be freely selected. Different hyperparameters lead to different results for the same dataset. The different hyperparameters can be combined in a vector λ. Each element of the vector affects the cost function.

Only vague hints can be found in the literature about which hyperparameters are suitable for the task. Therefore, the hyperparameters in this project are determined using various optimization techniques. The goal is to find a solution vector of hyperparameters λ that gives an optimal result of the cost function. Table 3 gives an overview of the hyperparameters that are determined in the following optimization step. The parameters HL1 ... HL5 refer to the neural network architecture, while the learning rate hyperparameter α and the decay factor ρ refer to its training. The definition of the limits in Table 3 is based on the validation runs with the neural network represented in Section 3.1.

Table 3. Limitation of the experimental space for the optimization procedures

Hyperparameter	**Lower limit**	**Upper limit**
Number of neurons in the 1st hidden layer, HL1	100	9000
Number of neurons in the 2nd hidden layer, HL2	50	5000
Number of neurons in the 3rd hidden layer, HL3	30	3000
Number of neurons in the 4th hidden layer, HL4	20	2000
Number of neurons in the 5th hidden layer, HL5	10	1000
Learning rate, α	0.0001	1.0
Decay factor, ρ	0.0001	0.9999

Three different methods are used for optimization. On the one hand, the space-filling design, a method of statistical experimental design, is explored. In addition, two methods from statistical optimization are used, the Particle-Swarm-Optimization and the Genetic Algorithms. The Particle-Swarm Optimization (PSO) algorithm is illustrated below.

The Particle-Swarm Optimization (PSO) algorithm is one of the most widely considered and applied algorithms in the literature for stochastic optimization problems. Mathematically, its objective is to find the global minimum of a multidimensional and nonlinear cost function. It is based on the flocking behavior of birds. It uses three main interactions of the bird flock: separation, alignment and cohesion. Due to the flocking behavior and the associated large number of test points, there is a high probability that the algorithm will find a global optimum. [13]

The first step in the PSO is the random initialization of the particles. Each individual particle with index j represents a random selection of a combination of all hyperparameters as a numerical vector λ_j. In addition, the initial velocity v_j of each particle is randomly determined. For each particle, the cost function is calculated.

Based on the function values of all particles at time k, new particle positions are determined for all particles. For this purpose, the individual velocity $\underline{v}_j$ at time k+1 is determined for each particle.

$$\underline{v}_j[k+1] = w \cdot \underline{v}_j[k] + c_1 \cdot r_1 \cdot \left(\underline{p}_{j,BEST}[k] - \underline{\lambda}_j[k]\right) + c_2 \cdot r_2 \cdot \left(\underline{g}_{BEST}[k] - \underline{\lambda}_j[k]\right) \tag{4.1}$$

The first summand consists of the weighted velocity at the last time point k. The weighting factor w describes which part of the velocity is retained. The term is also called the inertia term because of the analogy to mechanical mass inertia. The larger the weighting factor w is chosen, the higher is the velocity and thus the distance the particle travels in the experimental space.

The second summand stands for the individual cognitive component and evaluates the previous best position $\underline{P}_{j,\mathrm{BEST}}$ of particle j (Particle Best) as well as its current position $\underline{\lambda}_j[k]$. The difference moves the particle in a new direction and is weighted by the factors c_1 and r_1.

The third summand represents the social swarm component. It consists of the weighted difference of the best result of all particles of the swarm g_{BEST} (Global Best) and the current position $\underline{\lambda}_j[k]$. The difference moves the particle in a new direction and is weighted by the factors c_2 and r_2.

The weighting of the components is done using the factors c_1 and c_2, the so-called cognitive and social speedup coefficients. To add a stochastic component to the search algorithm, the factors r_1 and r_2 are used, which are random and uniformly distributed in the range between 0 and 1. This ensures that a particle can also move randomly into new areas.

Using the individual velocity $\underline{v}_j[k+1]$, the future position $\underline{\lambda}_j[k+1]$ of the individual particle j is determined.

$$\underline{\lambda}_j[k+1]=\underline{\lambda}_j[k]+\underline{v}_j[k+1] \tag{4.2}$$

For graphical visualization, this update is shown in Fig. 4 for one particle.

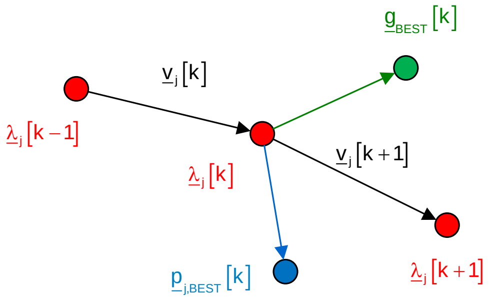

$\underline{g}_{\mathrm{BEST}}[k]$
$\underline{v}_j[k]$
$\underline{\lambda}_j[k-1]$
$\underline{v}_j[k+1]$
$\underline{\lambda}_j[k]$
$\underline{p}_{j,\mathrm{BEST}}[k]$
$\underline{\lambda}_j[k+1]$

Fig. 4. Visualization of Particle-Swarm-Optimization

The optimization ends when the search function reaches a defined accuracy with the determined hyperparameters or the maximum number of iterations is reached.

5 Results and discussions

The optimization methods are applied to a 7-dimensional test space according to Table 3. The parameterization of the methods is largely independent of each other. Only for the particles and chromosomes identical parameter configurations are used as starting points in order to achieve a better comparability of the results between PSO and GA. Furthermore, the number of particles and chromosomes is chosen to be the same with J = 40. In order to keep the scope of the experiment approximately the same for all methods, the number of iterations for PSO and GA is limited to 14. This results in a total of 560 parameter configurations to be examined. For the space-filling design, there are 500 parameter configurations. Based on the selected initialization for each method, the following results are obtained, which will be discussed in more detail.

5.1 Space-Filling Design

The space-filling design uses the entire range of values of each hyperparameter to achieve the best possible coverage of the experimental space. This idea can be well understood in Fig. 5. As an example, the accuracies are shown here as a function of two of the seven hyperparameters visualized in three-dimensional space. For the remaining hyperparameters, which are not listed, this arrangement applies analogously. The resulting accuracies lie between 96.99 % and 99.08 %. Low accuracies occur except for a few outliers for learning rates of $\alpha < 0.2$.

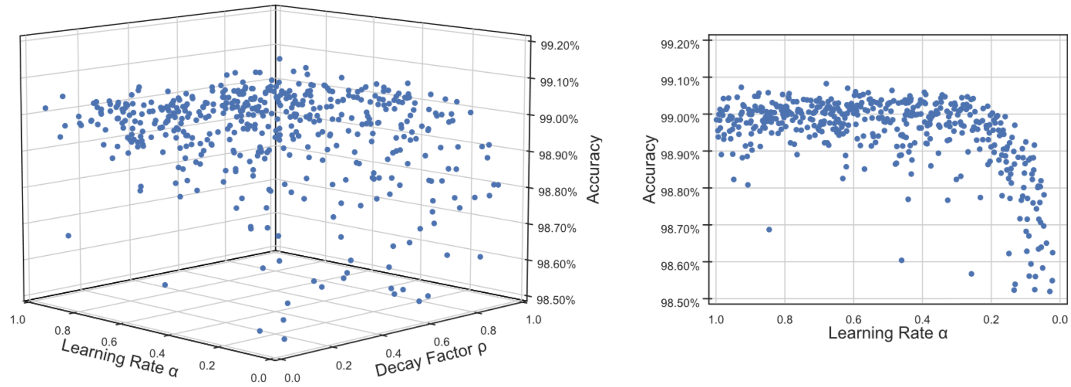


Fig. 5. Space-filling design - Accuracy as a function of the parameters learning rate and decay factor

5.2 Particle-Swarm-Optimization

The results from PSO are shown in Fig. 6. As with the SFD, the PSO delivers poor accuracies for learning rates α < 0.2. Due to the selected starting points, the resulting accuracies lie in a larger scatter range between 80.71 % and 99.11 % compared to the SFD.

The vertical gradients in sections in the two-dimensional view characterize the swarm behavior during the optimization. Likewise, the denser arrangement of the samples describes the convergence of the swarm towards the identified optimum, which is here located at α = 0.3686. Based on Fig. 6, it is also clear that the PSO algorithm does not include the complete range of values of the learning rate in the optimization. Instead of an exploratory search in all levels of the experimental space, the determination of the optimum is locally limited. The reasons for this lie in the selected parameterization of the PSO algorithm and the position of the starting points in the experimental space.

Although the maximum accuracies differ only marginally, the PSO achieves a slightly higher accuracy of 99.11 % than the SFD with 99.08 %, despite incomplete coverage of the entire test space.

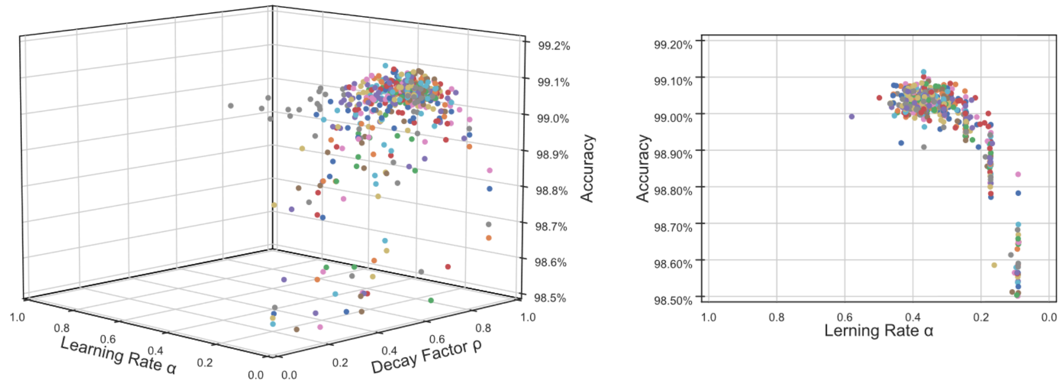


Fig. 6. Particle-Swarm-Optimization – Accuracy as a function of the parameters learning rate and decay factor

5.3 Genetic algorithms

The focus on samples in the optimal region is even more pronounced when genetic algorithms are used. The results obtained with this optimization method are shown in Fig. 7. The individual chromosomes quickly concentrate to values close to α = 0.38 and ρ = 0.75, despite identical initial conditions as in particle swarm optimization. A larger mutation rate could be used to parameterize the algorithm so that it also covers the experimental space further.

Using the Genetic Algorithms, a maximum accuracy of 99.10 % is achieved. It is between the accuracy of 99.11 % achieved with the PSO and the accuracy of 99.08 % achieved with the SFD.

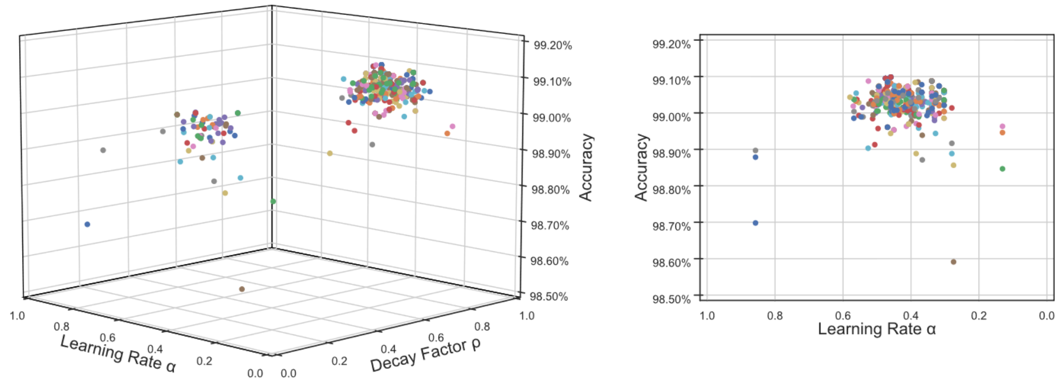


Fig. 7. Genetic algorithms - accuracy as a function of the parameters learning rate and decay factor

5.4 Comparison of the optimization methods

Due to the different optimization methods, a direct comparison of the results between the methods used is not possible. The PSO and the GA use an iterative and population-based approach to optimization, respectively. The SFD, on the other hand, is based on covering the experimental space as uniformly as possible.

In order to evaluate the performance of the methods, the values of the highest accuracy and the mean value of the accuracies per iteration or generation are used. They are shown in the two diagrams in Fig. 8. The results from the SFD are listed as a constant value.

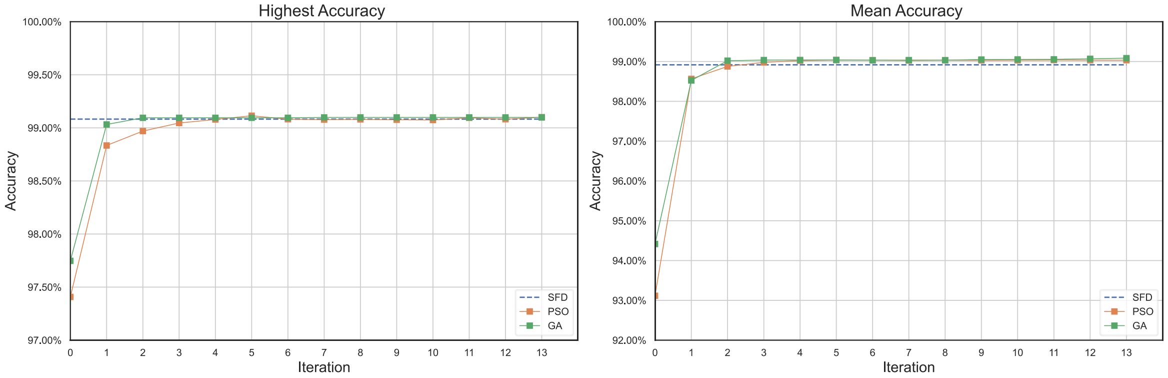


Fig. 8. Comparison of results

Already after the first iteration the accuracies show very high values of 98.83 % (PSO) and 99.03 % (GA). It becomes clear that the optimization with the GA converges faster than that with the PSO. The best values in each case are achieved with the parameter combinations given in Table 4.

Table 4. Tabular comparison of the optimum settings for the different methods and the achieved accuracies

Hyperparameter	SFD	PSO	GA
Number of neurons in the 1st hidden layer, HL1	7073	4728	5562
Number of neurons in the 2nd hidden layer, HL2	3513	3719	5000
Number of neurons in the 3rd hidden layer, HL3	1561	2274	2808
Number of neurons in the 4th hidden layer, HL4	866	1800	827
Number of neurons in the 5th hidden layer, HL5	866	1000	888
Learning rate, α	0.6795	0.3686	0.4593
decay factor, ρ	0.8084	0.8426	0.9272
accuracy	99.08 %	99.11 %	99.10 %

The values show that the best results were produced with different settings. This suggests that achieving minimum requirements results in high accuracy and that the maximum accuracies are local maxima that are achieved by chance. The results depicted in Fig. 8 have always been calculated with the same seed to improve comparability, thus the learning process has always started with the same initial situation. In order to test whether the differences between the optimization methods can be generalized, the neural networks were trained for 40 different seed values. The results were reproducible, all accuracies lie in the narrow range of 99 - 99.1%. The median of the Genetic Algorithms (99.05 %) is larger than the median of the Particle Swarm Optimization (99.04 %) and the median of the Space Filling design (99.03 %).

6 Conclusion

This paper presents a method for automatic visual inspection of solar modules. It uses a supervised learning method with a labeled dataset to build a neural network that assigns the labels NOK, OK and PREMIUM to the modules with an accuracy of more than 99 %.

A prerequisite for the application of neural networks was the preprocessing of the data, which consisted of normalization, truncation of irrelevant areas and reduction of resolution. Fast Fourier Transform was used to compress the data. These measures reduced the original input data shape form 1024 x 1024 down to the applied data shape of 64 x 32, which results in a reduction of computation time by a factor of 13. Data augmentation was used to expand and balance the dataset.

The shortened computation time was a prerequisite for optimizing the hyperparameters of the neural network. Space-filling design, particle warm optimization and genetic algorithms were used and compared for optimization. All optimizations resulted in architectures that achieved over 99% accuracy. The Genetic Algorithms showed slight advantages in terms of the achieved accuracy

The high accuracy of the presented method and the use of genetic algorithms demonstrate the performance of the method and recommend it for further projects of automatic visual inspection. The method thus allows digitalization in statistical process control and thus contributes to the implementation of the Quality 4.0 vision.

References

1. Siebertz, K.: Statistical Design of Experiments, Berlin: Springer- Verlag GmbH, 2017.
2. Holland, J.: Adaptation in natural and artificial systems, The University of Michigan Press, 1975.
3. Jong-Wook K., Sang Woo K., PooGyeon P. and Tae Joon P.: On the similarities between binary-coded GA and real-coded GA in wide search space, Proceedings of the Congress on Evolutionary Computation. CEC'02, Honolulu, HI, USA, 2002, pp. 681-686 vol.1.
4. Sabottke C. and Spieler B.: The Effect of Image Resolution on Deep Learning in Radiography, Radiology: Artificial Intelligence, Vol. 2, No. 1, 2020.
5. Schiebel, S., Spaethe, F.: Solar cell classification, Project work, Karlsruhe University of Applied Sciences, Karlsruhe, 2019.
6. Strohrmann, M.: Systems Theory Online: www.hs-karlsruhe.de/mesysto, accessed on 07.04.2021.
7. YANG, S., et al. Image Data Augmentation for Deep Learning: A Survey. arXiv preprint arXiv:2204.08610, 2022.
8. Keiron O., Ryan N., An introduction to convolutional neural networks. arXiv preprint arXiv:1511.08458, 2015.
9. Pernau, F.: Visual Quality Assurance with Neural Networks Master's thesis, Karlsruhe University of Applied Sciences, 2020.
10. Brownlee J., Machine Learning Mastery, Available: https://machinelearningmastery.com/how-to-configure-the-number-of-layers-and-nodes-in-a-neural-network/. [Zugriff am 14 März 2020].
11. Vardi, G., Shamir, O.: implicit Regularization in ReLU Networks with the Square Loss, Weizmann Institute of Science, arXiv:2012.05156v2 [cs.LG] on 15.12.2020.
12. Keras: Keras API Refrence - Probabilistic losses, https://keras.io/api/losses/probabilistic_losses/, accessed on 16.04.2021.
13. Neuswirth, C.: Optimization of Hyperparameters of Neural Networks with a Particle Swarm Optimization, Project Thesis, Karlsruhe University of Applied Sciences, 2020.

Learning based Model Predictive Control of a High-Altitude Simulation Chamber

Arsema Derbie[1], Maurice Kettner [1] and Eyassu Woldesenbet [2]
[1] IEEM-Institute of Energy Efficient Mobility, Karlsruhe University of Applied Sciences
arsema.derbie2@h-ka.de, maurice.kettner@h-ka.de
[2] Addis Ababa institute of Technology
dreyassu@gmail.com

Abstract. The limitation of conventional methods to explicitly model and monitor physical systems emanates from system complication, uncertainties, and so forth. Artificial intelligence approaches, Artificial Neural Networks in particular, resolve this difficulty by efficiently capturing the pattern of physical systems and exploring key relationships of determinant parameters effectively. The development of an artificial neural network model to catch the interrelation of the input and output variables was successful. Time series data collected using a variety of combinations of control variables were used to train a sequential model which predicted the chamber temperature from the inputs of control variables. From such a black box model, developing a model predictive controller that predicts upcoming events and sets control actions accordingly was developed. Optimization is a major essence of Model Predictive Control as each suggested step by the controller must be optimized to the required control law. Such optimization is better realized in mathematically modeled systems. But, for non-linear and non-convex relationships as in neural networks, this is cumbersome. This difficulty is addressed by the use of input convex neural networks which relate model outputs with the inputs in a convex relationship. Optimization is relatively simpler when convexity is granted. Finally, a 3 layers input convex neural network that represent the system specifications was developed and optimized control steps were generated using COBYLA (Constrained Optimization by Linear Approximation) solver.

Keywords: Input convex layers, Model Predictive control, convexity

1. Introduction

In Bruchsal Germany, there exists an environment chamber that simulates global temperature and altitude conditions to test the performance of hand-held power tool engines to date [1]. The pressure control, that provides the altitude feature, is monitored and controlled by Raspberry operated actuator manipulating the throttle that controls the amount of conditioned air that flows to the chamber. The matter, air and refrigerant, together with the data flow to and out of the chamber is shown in Fig. 1.

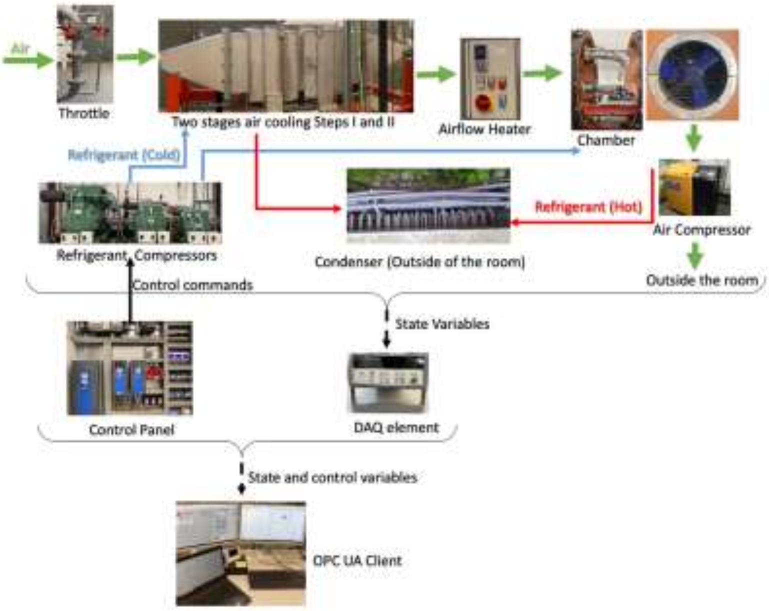


Fig. 1. Matter and data flow to and out of the chamber

The system monitory and control were enhanced by a digitalization solution which allowed the field devices (low level components) to be connected to high level supervision station making the setup a Cyber Physical System (CPS)[2].

For such a digitized system of non-linearly related multiple inputs and multiple outputs with time delays, the classic control mechanisms are limited since they disregard the knowledge of the process, possess constant parameters and solely rely on the measurements from sensors [3]. MPC provides optimal, predictive, and adaptive control with a simple structure and dynamic performance [4]. It's maximum potential of commands in order to impose temperature bounds makes it highly recommended for thermal (conditioning) control applications like heating, ventilation and air conditioning (HVAC) units and environment chambers [4].

MPC predicts future system behavior based on a system model considering it in the optimization that determines the optimal trajectory of manipulated variables [5]. A typical MPC consists mainly of a system model and a controller design which comprises an objective function and a control law.

This paper reports the results of a learning-based Model predictive control by first stating about the data acquisition process. Then, it discusses the system modeling and Control design before concluding.

2. Data Acquisition

System variables are basically divided into three depending on their role from the entire relationship: state variables, control variables and parameters. Table 1 summarizes the variables of the system accordingly.

Table 1. Variable types

State variables	**Control Variables**	**Parameters**
Step-I T, Step-II T, Heater T, Fan speed, Mass flow, Chamber T, Motor T	Step-I T_{set}, Step-II T_{set}, Heater T_{set}, Fan $speed_{set}$, Chamber cooler T_{set}	Room Temperature, Humidity, Season

Parametric variables do not affect state variables directly as control variables do. But, their influence on the system is significant. For the parameters of the chamber, there are two datasets depending on the relatively long seasons of the year, summer and winter, which dictate the room temperature and humidity. Both the winter and the summer datasets are collected with varying pressure and temperature values with different starting points to make the data of a high volume and variety. The data was recorded every 2 seconds with suffices to the velocity requirements for such a slow system. Hence, the three basic features of big data acquisition: volume, variety and velocity are fulfilled [6].

Multiple recordings were taken within the upper bounds and lower bounds of attainable temperature and pressure values [1]. 11 temperature points ranging between 30 ℃ and -20 ℃ with a step of 5 ℃ and four pressure values of 990 mbar, 900 mbar, 800 mbar and 700 mbar for the two datasets with a total file size of more than 300MB having millions of data points were collected.

3. System Modeling

One of the basic features of MPC include making optimized decisions depending on the system dynamics. The accuracy and effectiveness of an MPC are highly dependent on the identified model. Modeling and identification are the most difficult and time-consuming parts of automation processes. The basic conditions that each model intended for MPC usage should satisfy are reasonable simplicity, well estimated system dynamics and steady-state properties as well as satisfactory prediction properties [7].

System modeling techniques can be broadly classified into three: physical modeling (or white box, mathematical, forward), data-driven (or black box/empirical/inverse), and gray box (or hybrid). System models can be dynamic or static (steady) depending on the variability of parameters with time. Depending on the linearity of the system, the models can be linear or nonlinear. Most physics-based models fall under inductive types of system models while data- based models are mostly deductive. Other classifications include explicit and implicit, discrete or continuous, deterministic or probabilistic/stochastic models [8].

To represent the environment chamber, a data-based approach was selected. The concept of this approach is to fit a transfer function model to the input/output real model data to yield coefficient polynomials that can be factored to provide resonance frequencies and characterization of damping coefficients without knowledge of the internal working [9]. The strengths of such system identification technique are lower engineering cost because it follows a data-in-data-out approach; less domain knowledge because it is based on the mapping of input and output data, and greater adaptability because the model will evolve itself with new data. Some of its drawbacks include high demand for data quality: missing, wrong, or biased data lead to low quality models [10].

For most thermal systems that are dynamic, nonlinear, and very high order due to physical properties such as high thermal inertia, real lag time, uncertain disturbance factors, etc. black-box models provide better accuracy without a comprehensive knowledge of the operations [9]. Deep neural networks have proven to be successful in many identification tasks, however, from a model-based control perspective, these networks are difficult to work with because they are typically nonlinear and nonconvex. To bridge the gap between model accuracy and control tractability faced by neural networks, networks that are convex concerning their inputs are con- structed explicitly [11].

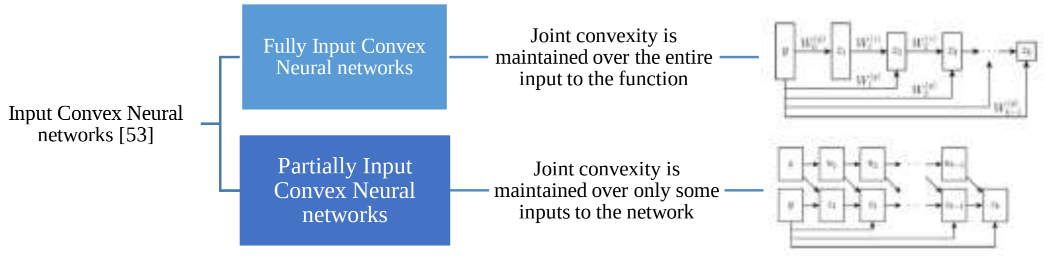

Input Convex Neural networks [53]
Fully Input Convex Neural networks
Joint convexity is maintained over the entire input to the function
Partially Input Convex Neural networks
Joint convexity is maintained over only some inputs to the network

Fig. 2. Classification of Input Convex Neural networks

From the numerous variables in the environment chamber, a three-layered partially input convex neural network where only the control variables are convex to the output was constructed. Their development is highly dependent on the selection of convex and non-decreasing activation functions and non-negative weights. The proof of convexity in these networks is given by the fact that non-negative sums of convex functions are also convex and that the composition of a convex and non-decreasing function is also convex [12]. The system representation capacity is compensated as can be seen in Fig. 3.

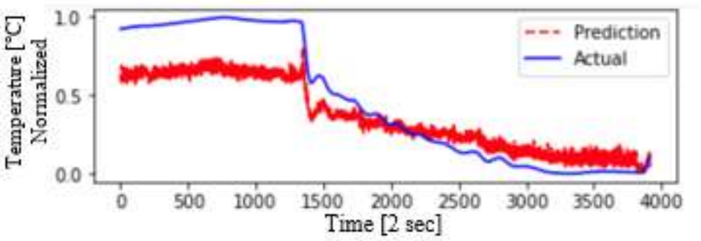


Fig.3. PICNN representation of Chamber Temperature at 700mbar (Temperature [°C] Vs Time [sec]-Normalized)

4. Control Design

Defining an optimization problem is an important task in the controller design step of an MPC. This problem is again constrained by the system representation equation and the boundary conditions of the input and output, together with state and control variables. System representation equations of many thermal systems (discrete-time linear time-invariant system that evolves in time) are given in equation 1.

$$X_{t+1} = AX_t + BU_t \qquad \text{Eqn 1a}$$

$$y_t = CX_t + BU_t \qquad \text{Eqn 1b}$$

where x, y and u are input state variables, output state variables and control actions while A, B, C and D are System Matrices.

A general optimization problem can be stated as shown in equation 2.

$$\min_{n} \sum_{t=0}^{\infty} y_t^T Q_{x_t} + u_k^T R u_k \qquad \text{Eqn 2a}$$

Subject to:

$$X_{t+1} = AX_t + BU_t \qquad \text{Eqn 2b}$$

$$y_t = CX_t + BU_t \qquad \text{Eqn 2c}$$

$$y_{min} \leq y_t \leq y_{max} \qquad \text{Eqn 2d}$$

$$x_{min} \leq x_t \leq x_{max} \qquad \text{Eqn 2e}$$

$$u_{min} \leq u_t \leq u_{max} \qquad \text{Eqn 2f}$$

The system model predicts chamber temperature and the respective motor temperature which makes these variables output variables (y). The customized optimization problem is given in equation 3.

$$\min \sum \left(\sqrt{(\mathrm{T_{set}} - \mathrm{T_{chamber}})^2} + \sqrt{(\mathrm{T_{set}} - \mathrm{T_{motor}})^2}\right) \qquad \text{Eqn 3}$$

This optimization was carried out by COBYLA (Constrained Optimization by Linear Approximation) solver. The output of the controller hence becomes a series of optimized control actions when a certain temperature in the chamber is desired to be achieved. An example of the changes in the chamber temperature with respective changes in the control actions is shown in Fig. 4.

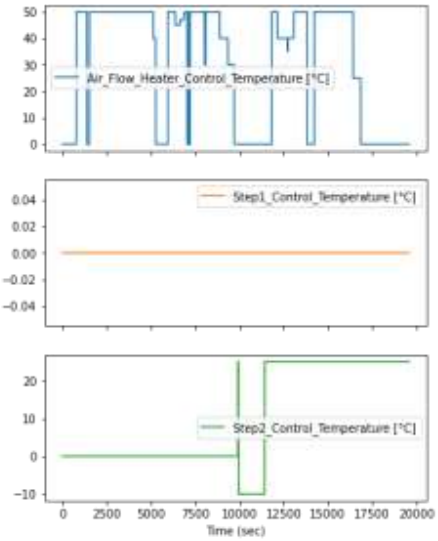


Fig. 4a. Control actions of Air heater, Step-I and step-II coolers at 700mbar

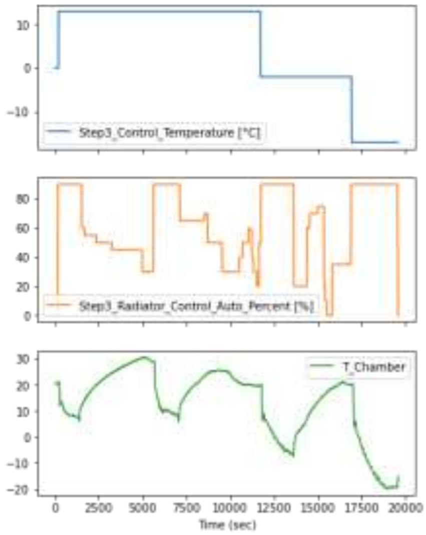


Fig. 4b. Changes in the chamber temperature according to the control variables at 700mbar with control actions of chamber cooler and fan

5. Conclusion

The high-altitude simulation chamber with unpredictable ways to reach desired chamber and motor temperature values was represented with neural networks to find the optimum control actions of its thermal components. The biggest challenge of optimization on non-linear and non-convex models was addressed by the implementation of partially input convex neural networks. MPC parameters that correspond to the slow thermal process were defined and the optimum control actions were generated by a linearization solver. Hence, the chamber can be conditioned with the most efficient control steps in terms of time and resource.

References

1. A. Martel, F. Scholl, and D. Weierter, “Development of a Climate and Altitude Simulation Test Bench for Handheld Power Tools,” 2018, doi: 10.4271/2018-32-0033.
2. A. Derbie, P. Nenninger, C. Hadamek, and M. Renner, “Digitalization of a Climate and Altitude Simulation Test Bench for Handheld Power Tools to Automate Its Thermal Management System,” *SAE Technical Paper*, 2022, doi: 10.4271/2022.
3. M. Tesfay, F. Alsaleem, P. Arunasalam, and A. Rao, “Adaptive-model predictive control of electronic expansion valves with adjustable setpoint for evaporator superheat minimization,” *Build Environ*, vol. 133, no. December 2017, pp. 151–160, 2018, doi: 10.1016/j.buildenv.2018.02.015.
4. P. Hameed, N. Bin, M. Nor, P. Nallagownden, I. Elamvazuthi, and T. Ibrahim, “A review on optimized control systems for building energy and comfort management of smart sustainable buildings,” *Renewable and Sustainable Energy Reviews*, vol. 34, pp. 409–429, 2014, doi: 10.1016/j.rser.2014.03.027.

5. M. Schwenzer, M. Ay, T. Bergs, and D. Abel, “Review on model predictive control: an engineering perspective,” *International Journal of Advanced Manufacturing Technology*, vol. 117, no. 5–6. Springer Science and Business Media Deutschland GmbH, pp. 1327–1349, Nov. 01, 2021. doi: 10.1007/s00170-021-07682-3.
6. H. Singh, “Big data, industry 4.0 and cyber-physical systems integration: A smart industry context,” *Mater Today Proc*, 2020, doi: 10.1016/j.matpr.2020.07.170.
7. S. Prívara, J. Cigler, Z. Váňa, F. Oldewurtel, C. Sagerschnig, and E. Žáčeková, “Building modeling as a crucial part for building predictive control,” *Energy Build*, vol. 56, pp. 8–22, Jan. 2013, doi: 10.1016/j.enbuild.2012.10.024.
8. Z. Afroz, G. M. Shafiullah, T. Urmee, and G. Higgins, “Modeling techniques used in building HVAC control systems: A review,” *Renewable and Sustainable Energy Reviews*, vol. 83. Elsevier Ltd, pp. 64–84, Mar. 01, 2018. doi: 10.1016/j.rser.2017.10.044.
9. R. Z. Homod, “Review on the HVAC System Modeling Types and the Shortcomings of Their Application,” *Journal of Energy*, vol. 2013, no. May, pp. 1–10, 2013, doi: 10.1155/2013/768632.
10. Y. Li, Z. O’Neill, L. Zhang, J. Chen, P. Im, and J. DeGraw, “Grey-box modeling and application for building energy simulations - A critical review,” *Renewable and Sustainable Energy Reviews*, vol. 146. Elsevier Ltd, Aug. 01, 2021. doi: 10.1016/j.rser.2021.111174.
11. Y. Chen, Y. Shi, and B. Zhang, “Optimal control via neural networks: A convex approach,” *7th International Conference on Learning Representations, ICLR 2019*, pp. 1–25, 2019.
12. F. Bünning, A. Schalbetter, A. Aboudonia, M. H. de Badyn, P. Heer, and J. Lygeros, “Input Convex Neural Networks for Building MPC,” pp. 1–11, 2020, [Online]. Available: http://arxiv.org/abs/2011.13227

Modelling of a large-format lithium-iron-phosphate-based lithium-ion battery cell with neural ordinary differential equations

Jennifer Brucker*, Wolfgang G. Bessler, and Rainer Gasper

Institute of Sustainable Energy Systems – Offenburg University of Applied Sciences
jennifer.brucker@hs-offenburg.de

Abstract. Lithium-ion batteries show strongly nonlinear behaviour regarding the battery current and state of charge. Therefore, the modelling of lithium-ion batteries is complex. Combining physical and data-driven models in a grey-box model can simplify the modelling. Our focus is on using neural networks, especially neural ordinary differential equations, for grey-box modelling of lithium-ion batteries. A simple equivalent circuit model serves as a basis for the grey-box model. Unknown parameters and dependencies are then replaced by learnable parameters and neural networks. We use experimental full-cycle data and data from pulse tests of a lithium iron phosphate cell to train the model. Finally, we test the model against two dynamic load profiles: one consisting of half cycles and one dynamic load profile representing a home-storage system. The dynamic response of the battery is well captured by the model.

Keywords: neural ordinary differential equations; grey-box model; equivalent circuit model; lithium-ion battery.

1 Introduction

Lithium-ion batteries play an important role in our everyday life: They supply portable devices such as smartphones with electrical energy, are a key technology for electromobility, and are used in stationary applications such as home-storage systems. Battery models are used to predict the dynamic voltage and current behaviour and to monitor internal states. There are many different types of models with different accuracy and complexity [1,2]. We summarize a grey-box (GB) modelling approach that uses an equivalent circuit model (ECM) as a basis and was first introduced in Ref. [3].

GB modelling is the combination of white-box (WB) and black-box (BB) modelling techniques. BB models learn relations between inputs and outputs of systems based on data. Neural networks belong to the BB modelling techniques. WB models use mathematical equations derived from prior physical, chemical or engineering knowledge to describe the behaviour of the underlying system [4,5,6,7].

Many approaches in current research use neural networks to model lithium-ion batteries. The authors of Ref. [8] and Ref. [9] use neural networks to predict the state of charge (SOC) of a battery whereas the authors of Ref. [10] focus on the state of health (SOH) prediction. In Refs. [11] and [12] neural ordinary differential equations (NODEs), a special form of neural network, are used for GB modelling of lithium-ion batteries. The authors of Ref. [11] model aging effects such as solid electrolyte interface formation, lithium plating, and active material isolation as well as the increase in the internal resistance. The final deviation of the physical model and measurement results is approximated with NODEs. In contrast to that, we built a GB model of a lithium-ion battery based on a simple ECM in our previous work [12]. Herein, the voltage drop across the included resistor–capacitor (RC) circuit is represented by NODEs.

We continued our previous work from Ref. [12] by considering the battery dynamics. The methods and results can be found in Ref. [3] in detail and are summarized here. We used additional data from charging and discharging with pulsed currents during the training of the GB model to identify the time constant of the fast battery dynamics. Neither temperature dependencies nor aging effects have been considered.

We applied our GB modelling approach to a large-format 180 Ah prismatic commercial lithium-ion cell with lithium iron phosphate (LFP)/graphite chemistry. The experimental properties of the cell have been investigated in great detail in Ref. [13]. The state diagnosis of LFP cells is challenging because of their flat, plateau-like open circuit voltage curve and charge-discharge voltage hysteresis [14]. One of our goals is to investigate the applicability of our GB modelling approach to this type of cell.

2 Methodology

In this section, we give a short introduction to NODEs. Furthermore, we present a simple ECM and show how to derive a GB model from it. The section is complemented with initialization, normalization, and training techniques. Finally, we give insights into the experimental basis.

2.1 Background: Neural Ordinary Differential Equations

The interested reader can find a detailed overview of neural networks in Ref. [15].

In Ref. [16] residual neural networks (ResNets) are introduced. They overcome problems with the degradation of the training loss with an increasing number of hidden layers in deep neural networks by adding additional short-cut connections to feedforward networks. These short-cut connections allow the direct addition of the input of a neuron to its output. ResNets can be used for time series prediction.

The state transformation from layer t to layer $t+1$ in a ResNet follows the recursive formula [16]

$$\boldsymbol{z}_{t+1} = \boldsymbol{z}_t + \boldsymbol{f}\left(\boldsymbol{z}_t, \boldsymbol{\theta}_t\right), \quad t = 0, ..., T-1 \tag{1}$$

where, $\boldsymbol{z}_t \in \mathbb{R}^d$ is the vector of the hidden states at layer t, $\boldsymbol{\theta}_t$ the learned parameters of layer t and $\boldsymbol{f} : \mathbb{R}^d \to \mathbb{R}^d$ a learnable function. Herein, the vector $\boldsymbol{\theta}_t$ of learned parameters summarizes the learned weights and biases. The explicit Euler discretization of the initial value problem [17,18,19,20,21,22,23]

$$\frac{\mathrm{d}\boldsymbol{z}\left(t\right)}{\mathrm{d}t} = \boldsymbol{f}\left(\boldsymbol{z}\left(t\right), t, \boldsymbol{\theta}\right), \quad \boldsymbol{z}\left(0\right) = \boldsymbol{z}_0 \tag{2}$$

can be derived through parameter sharing across the layers ($\boldsymbol{\theta}_t = \boldsymbol{\theta}$ for $t = 0, ..., T-1$). The right-hand side of the differential equation according to Equation (2) is represented by the neural network $\boldsymbol{f}$. Therefore, it is called NODE. Starting from the initial state $\boldsymbol{z}\left(0\right)$ a differential equation solver delivers the output state $\boldsymbol{z}\left(T\right)$ [17,20,21,23].

The differential equation according to Equation (2) is generalized to consider external variables $\boldsymbol{u}\left(t\right)$ [12]:

$$\frac{\mathrm{d}\boldsymbol{z}\left(t\right)}{\mathrm{d}t} = f\left(\boldsymbol{z}\left(t\right), \boldsymbol{u}\left(t\right), t, \boldsymbol{\theta}\right), \quad \boldsymbol{z}\left(0\right) = \boldsymbol{z}_0. \tag{3}$$

As stated in Refs. [12] and [24] one can combine NODEs with differential equations derived from prior physical knowledge in one equation system to build a GB model.

2.2 Equivalent circuit modelling

Besides physical modelling, it is a common approach to use ECMs to describe the dynamics of lithium-ion batteries. Due to their simplicity, ECMs are often used for SOC and SOH predictions [25,26,27].

As in Ref. [12], we used a simple ECM as a basis for GB modelling of the battery. The chosen ECM is shown in Figure 1. It is composed of an SOC-dependent voltage source, a hysteresis voltage drop, a serial resistor, and one RC circuit. We include parameter dependencies on battery current and SOC. The ECM can be described by the following equation system:

$$\frac{\mathrm{dSOC}}{\mathrm{d}t} = -\frac{1}{C_{\mathrm{bat}}} i_{\mathrm{bat}} \tag{4a}$$

$$\frac{\mathrm{d}v_{\mathrm{RC1}}}{\mathrm{d}t} = \frac{1}{C_1} \cdot \left(i_{\mathrm{bat}} - \frac{1}{R_1\left(\mathrm{SOC}, i_{\mathrm{bat}}\right)} \cdot v_{\mathrm{RC1}} \right) \tag{4b}$$

$$v_{\mathrm{bat}} = v_{\mathrm{OC}}\left(\mathrm{SOC}\right) - v_{\mathrm{hys}} \cdot \mathrm{sgn}\left(i_{\mathrm{bat}}\right) - R_{\mathrm{S}} \cdot i_{\mathrm{bat}} - v_{\mathrm{RC1}}, \tag{4c}$$

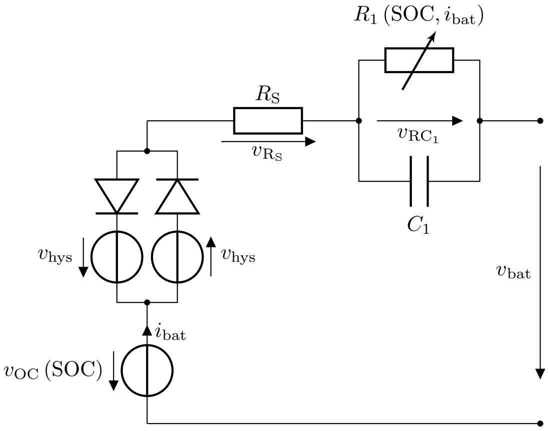


Fig. 1: ECM of a battery consisting of an SOC-dependent voltage source, a hysteresis voltage drop, a serial resistor, and an RC circuit. [3]. CC BY 4.0 https://creativecommons.org/licenses/by/4.0/

with the battery capacity C_{bat}, the hysteresis voltage drop v_{hys}, the serial resistance R_{S}, the charge-transfer resistance $R_1\left(\mathrm{SOC}, i_{\mathrm{bat}}\right)$ in the RC circuit depending on SOC and battery current, the double-layer capacitance C_1 and the SOC-dependent open-circuit voltage (OCV) $v_{\mathrm{OC}}\left(\mathrm{SOC}\right)$. The battery voltage v_{bat} is the output of the dynamic system and the battery current i_{bat} serves as the external input. The current for battery discharge is defined as positive, and the current for battery charge is defined as negative.

2.3 Grey-box modelling

The ECM according to equation system (4) served as a basis for GB modelling. We replaced unknown parameters and dependencies with learnable parameters and neural networks. As they are unknown or only approximately known, we considered the battery capacity C_{bat} in Equation (4a), the double-layer capacitance C_1 in Equation (4b), the hysteresis voltage drop v_{hys}, and the serial ohmic resistance R_{S} in Equation (4c) as learnable parameters. The charge-transfer resistance and its dependency on battery current and SOC are also unknown. As observed experimentally [13], it may have different characteristics during charging and discharging. Due to this fact, we represented the charge-transfer resistance R_1 through two learnable functions or rather two neural networks. Depending on the sign of the battery current, one of these learnable functions is chosen; at zero current ($i_{\mathrm{bat}} = 0\,\mathrm{A}$) the mean is taken.

In the output equation 4c we had to establish a link between OCV and SOC. Therefore, we derived $v_{\mathrm{OC}}\left(\mathrm{SOC}\right)$ from dedicated quasi-OCV measurements (cf. [13]). Overall, we derived the following GB model:

$$\frac{\mathrm{dSOC}}{\mathrm{d}t} = -\frac{1}{\omega_0} i_{\mathrm{bat}} \tag{5a}$$

$$\frac{\mathrm{d}v_{\mathrm{RC1}}}{\mathrm{d}t} = \frac{1}{\omega_1} \cdot \left(i - \frac{1}{R_1\left(\mathrm{SOC}, i_{\mathrm{bat}}\right)} \cdot v_{\mathrm{RC1}} \right) \tag{5b}$$

$$R_1\left(\mathrm{SOC}, i_{\mathrm{bat}}\right) = \begin{cases} f\left(\mathrm{SOC}, i_{\mathrm{bat}}, \boldsymbol{\theta}_f\right) & \forall\, i_{\mathrm{bat}} < 0 \\ g\left(\mathrm{SOC}, i_{\mathrm{bat}}, \boldsymbol{\theta}_g\right) & \forall\, i_{\mathrm{bat}} > 0 \\ \frac{1}{2}\left(f\left(\mathrm{SOC}, i_{\mathrm{bat}}, \boldsymbol{\theta}_f\right) + g\left(\mathrm{SOC}, i_{\mathrm{bat}}, \boldsymbol{\theta}_g\right)\right) & \text{else} \end{cases} \tag{5c}$$

$$v_{\mathrm{bat}} = v_{\mathrm{OC}}\left(\mathrm{SOC}\right) - \omega_2 \cdot \mathrm{sgn}\left(i_{\mathrm{bat}}\right) - \omega_3 \cdot i_{\mathrm{bat}} - v_{\mathrm{RC1}} \tag{5d}$$

with the learnable parameters ω_0, ω_1, ω_2, and ω_3. The functions f and g represent feedforward networks with their respective learnable parameters $\boldsymbol{\theta}_f$ and $\boldsymbol{\theta}_g$. We chose neural networks with one hidden layer and rectified linear unit (ReLU) activation for f and g. We varied the number of neurons in the hidden layer between 10 and 300. The SOC and the battery current serve as inputs to the neural networks and the ohmic resistance R_1 is the output.

The GB model combines physics-based ordinary differential equations (ODEs) and machine-learning-based NODEs in one equation system. This equation system is solved numerically within a single framework.

2.4 Experiments

We applied the proposed GB modelling approach to a single lithium-ion battery cell of the Chinese manufacturer CALB, model CA180FI. The cell is shown in Figure 2. The large-format prismatic cell with a nominal capacity of $180\,\mathrm{Ah}$ uses LFP at the positive electrode and graphite at the negative electrode. Experimental measurements were performed under a controlled laboratory environment (climate chamber CTS $40/200$ Li) using a battery cycler (Biologic VMP3). Details can be found in Refs. [3] and [13].

We used the measurement data from Ref. [3]. In detail, we used the constant current constant voltage (CCCV) charge and discharge curves with different C-rates of $0.1\,\mathrm{C}$, $0.28\,\mathrm{C}$, and $1\,\mathrm{C}$ (corresponding to $18\,\mathrm{A}$, $50\,\mathrm{A}$, and $180\,\mathrm{A}$, respectively) during the CC phase. The upper and lower cutoff voltages were $3.65\,\mathrm{V}$ and $2.5\,\mathrm{V}$, respectively, and a cut-off current of the CV phase of $\mathrm{C}/20$ was used. Additionally, one charge and one discharge curve with a pulsed current were recorded: During $50\,\mathrm{A}$ CC operation, every two SOC-percent the current was reduced to $25\,\mathrm{A}$ for $30\,\mathrm{s}$. Furthermore, two independent measurements that we used for model testing were carried out. Firstly, the cell was cycled with $50\,\mathrm{A}$ between $25\,\%$ and $75\,\%$ SOC, in the following referred to as half cycles. Secondly, measurements with a dynamic load profile over $48\,\mathrm{h}$ representing a home storage battery in a single-family house were performed. This synthetic load profile was taken from Ref. [28] and downscaled to the energy of the present cell. During all measurements the ambient temperature was kept at $T = 25\,^{\circ}\mathrm{C}$. To reduce the number of measurement points per data series, measurement values were deleted if the current only changed by $|\Delta i_{\mathrm{bat}}| \leq 0.5\,\mathrm{A}$ and the measured voltage changed by $|\Delta u_{\mathrm{bat}}| \leq 0.5\,\mathrm{mV}$. The measurement data were recorded and used as voltage versus time and current versus time series. The experiments are described in more detail in the original contribution [3].

Fig. 2: Photograph of the CALB cell

2.5 Normalization and initialization

It is recommended to scale the inputs of neural networks to simplify the training process. The input variables should be transformed in a way that their average over the complete training data set is close to zero. Additionally, the input variables should have similar value ranges [29].

We scaled all inputs to values between -1 and 1 to achieve a similar value range as for the SOC which is in the range of 0 to 1. The output values of the neural networks were normalized to the same value range. As we did not use different learning rates for different parameters, we also scaled the learnable parameters. For this reason, we had to estimate their order of magnitude. Additionally, the learnable parameters have to be initialized at the beginning. Therefore, we took a closer look at the training data to estimate the unknown parameters.

To find for example a good initial value for the battery capacity or rather the learnable parameter ω_0, one can calculate the charge throughput for a whole charging or discharging process by integrating the measured current over time. The other parameters can be estimated by analyzing the voltage response of the battery following a current step or by analyzing the overpotential. The interested reader is referred to Ref. [3] for more details.

2.6 Simulation and Optimization Methodology

We implemented the GB model in Python (version 3.7.6). Tensor computing and automatic differentiation were performed with the open-source machine learning framework PyTorch (version 1.9.0) [30]. We used the torchdiffeq library (version 0.2.1) [31] which builds on PyTorch to solve the ODEs and to backpropagate through the solutions. In detail, we used the Dopri8 method with an absolute tolerance of 10^{-5} and a relative tolerance of 10^{-3} to solve the differential equations and the standard odeint method from torchdiffeq for backpropagation. Finally, the loss minimization was carried out with an Adam optimizer.

2.7 Training

Only a small data base was available for training the GB model. Therefore, we decided to split the training into two consecutive steps: First, we used the CCCV data to train a simplified, static version of the GB model. Afterward, the battery dynamics were taken into account by considering the data from charging and discharging with a pulsed current.

In detail, in the first step we converted the differential equation (5b) into an algebraic equation by neglecting the double-layer capacitance:

$$v_{\mathrm{RC1}} = R_1\left(\mathrm{SOC}, i_{\mathrm{bat}}\right) \cdot i_{\mathrm{bat}}. \tag{6}$$

The resulting simplified GB model was trained with the time series data from CCCV charging and discharging with different C-rates. We initialized and scaled the learnable parameters ω_0, ω_2, and ω_3 and the neural networks f and g of the simplified model as described above and in Ref. [3] in detail. To solve the differential equation (5a) we had to provide an initial value for the SOC. We assumed that the battery was in equilibrium at the beginning of each time series and therefore the battery voltage was equal to the OCV. We used the inverted OCV(SOC) curve to obtain the respective SOC value. The learning rate of the Adam optimizer was decaying between 10^{-2} and 10^{-3}. The loss function was defined as the root mean squared error (RMSE) between the simulated battery voltage and the measured battery voltage. Additionally, approximated SOC values lower than 0 or higher than 1 were penalized. The total number of training epochs was varied as a hyperparameter of the training process. The optimization steps were carried out with stochastic gradient descent. We stored the parameters when the total loss of all considered training data sets decreased.

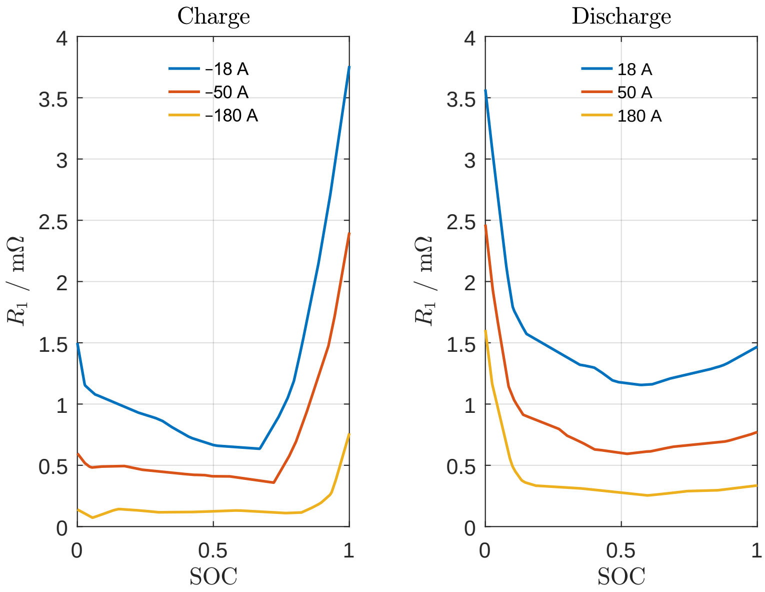


Fig. 3: Simulation results: R_1 as a function of SOC for different battery currents; (**left**): charging, (**right**): discharging. [3]. CC BY 4.0 https://creativecommons.org/licenses/by/4.0/

In the second step, we completed the training with the whole GB model according to Equations (5a) to (5d). Therefore, we initialized ω_1 as stated previously. The other parameters were taken from the pre-trained model and the initial SOC was determined as before. The initial value of the voltage drop v_{RC1} across the RC circuit was needed as well. We assumed $v_{\mathrm{RC1}}\,(t=0) = 0\,\mathrm{V}$. The loss function was implemented as before. However, a constant learning rate of 10^{-3} was chosen. During the first ten training epochs of the second part, we only considered the data from charging and discharging processes with a pulsed battery current. Afterward, we also took the other training data sets into account. Additionally, we only considered ω_1 as a changeable parameter during the first 20 of 30 training epochs in this second part of training. The other parameters were frozen. Here, we chose batch gradient descent for parameter optimization.

As mentioned before, the number of hidden neurons in f and g was varied between 10 and 300. The number of training epochs in the first training step served as a second hyperparameter. It was varied between 100 and 1000 while training part two was not changed. The detailed evaluation of the results in Ref. [3] finally made us choose the trained model with 100 hidden neurons in f and g and 300 training epochs in training step one as the final GB model.

2.8 Test

We tested the final GB model against the data sets covering the half cycles and the synthetic load profile. For the half cycles, the integration of the differential equation system resulted in a step size underflow. Therefore, we had to increase the absolute tolerance of the solver to 10^{-3} for the half cycles. Otherwise, we proceeded the same way as for training.

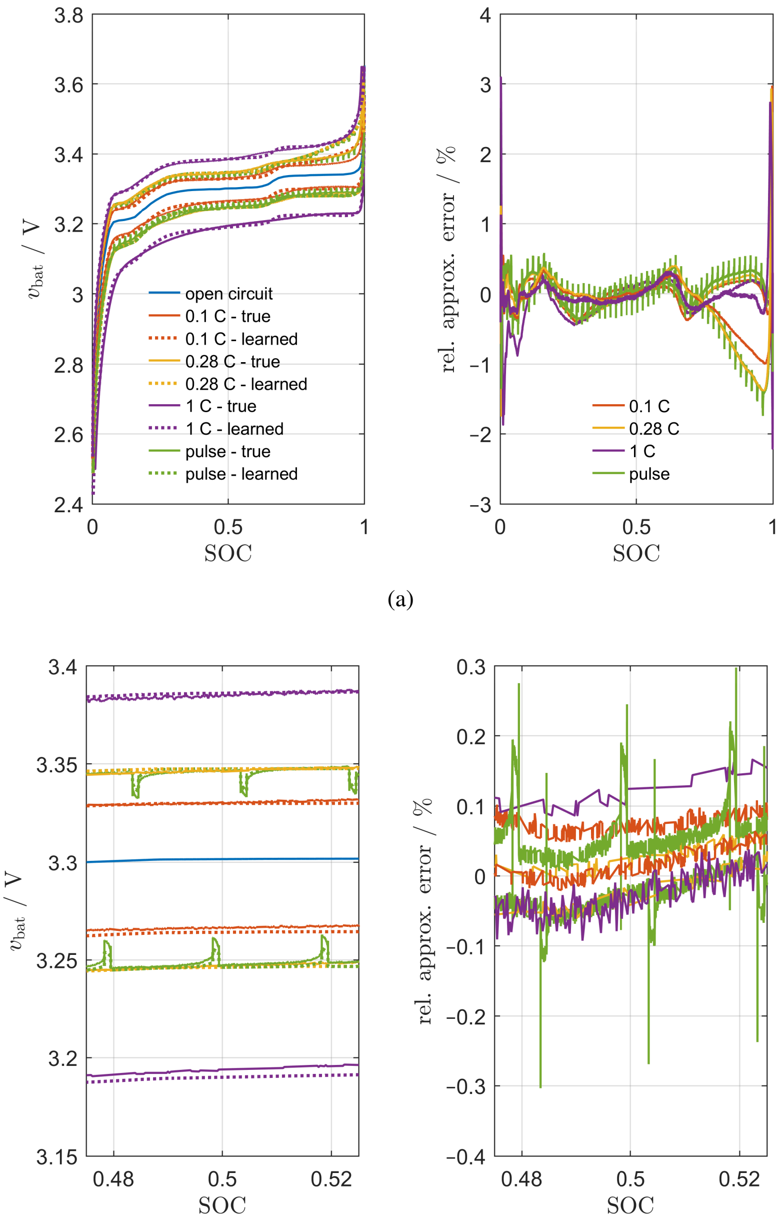
3 Results and Discussion

The training and test results can be found in detail in Ref. [3]. Here, we discuss the most important findings.

3.1 Training

After finishing the training, we took a closer look at the learned parameters and neural networks. The evaluation of the learned parameters with their respective scaling factors resulted in the following values:

$$\begin{aligned} C_{\mathrm{bat}} &= 191.5\,\mathrm{Ah} \\ C_1 &= 50.69\,\mathrm{kF} \\ v_{\mathrm{hys}} &= 11.25\,\mathrm{mV} \\ R_{\mathrm{S}} &= 281.4\,\mu\Omega. \end{aligned}$$



(a)

(b)

Fig. 4: Simulation results using NODEs for grey-box modelling of a lithium-ion battery in comparison to experimental data; left: charge and discharge curves for different C-rates at $T = 25\,°\mathrm{C}$. The lower branches represent discharge (time progresses from right to left), while the upper branches represent charge (time progresses from left to right); right: relative approximation error; (**a**) the whole SOC range (**b**) focus on medium SOC. [3]. CC BY 4.0 https://creativecommons.org/licenses/by/4.0/

The charge-transfer resistance is represented by the two neural networks f and g. Figure 3 illustrates the final results for R_1. The left panel shows the charge-transfer resistance for charging while the right panel shows the results for discharging as a function of SOC for different current values. The charge-transfer resistance is in the range of up to several milliohms.

When the absolute battery current is increased, the charge-transfer resistance decreases for both charging and discharging. At a medium SOC, the resistance is lower than at a low or a high SOC. During charge, the highest values occur when the cell is (nearly) full, during discharge, the highest values occur when the battery is (nearly) empty. This is typical for lithium-ion batteries with LFP cathode [13]. However, one has to keep in mind that the model is based on a simple equivalent circuit. It is therefore difficult to derive detailed electrochemical properties of the battery from the results.

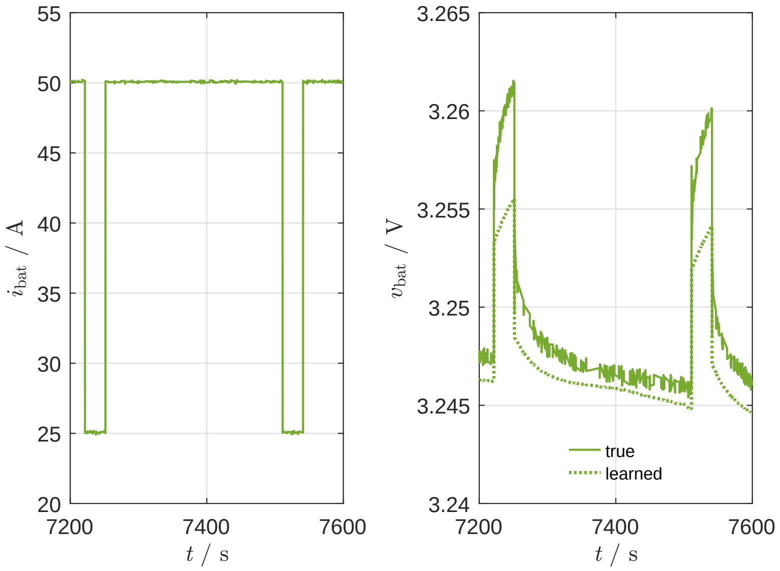


Fig. 5: Simulation results using NODEs for grey-box modelling of a lithium-ion battery in comparison to experimental data at $T = 25\,°\mathrm{C}$. The focus is on discharging with a pulsed current at a medium SOC; (**left**): battery current versus time; (**right**): battery voltage versus time

3.2 Comparison of Model against Training Data

Figure 4 shows the training results in comparison to the measured battery voltage. Here we chose a representation in the form of voltage versus SOC. The measurement data was given as voltage versus time series. However, the graphical display versus SOC allows a better comparison for different C-rates. The measured and the learned battery voltage are shown in the left panel. The right panel shows the relative approximation error concerning the measured battery voltage. Figure 4a shows the complete SOC range while Figure 4b focuses on a medium SOC. Overall, the simulation results are well in accord with the experiments for all investigated C-rates. The absolute value of the relative approximation error is smaller than 1% for a wide range of SOC. Only for (nearly) full and (nearly) empty batteries, the approximation error reaches an absolute value of up to around $3\,\%$. As the OCV(SOC) curve (shown in blue in Figure 4a,b) is very steep for high and low values of SOC, higher approximation errors are expected. Especially, for this reason, the results are acceptable.

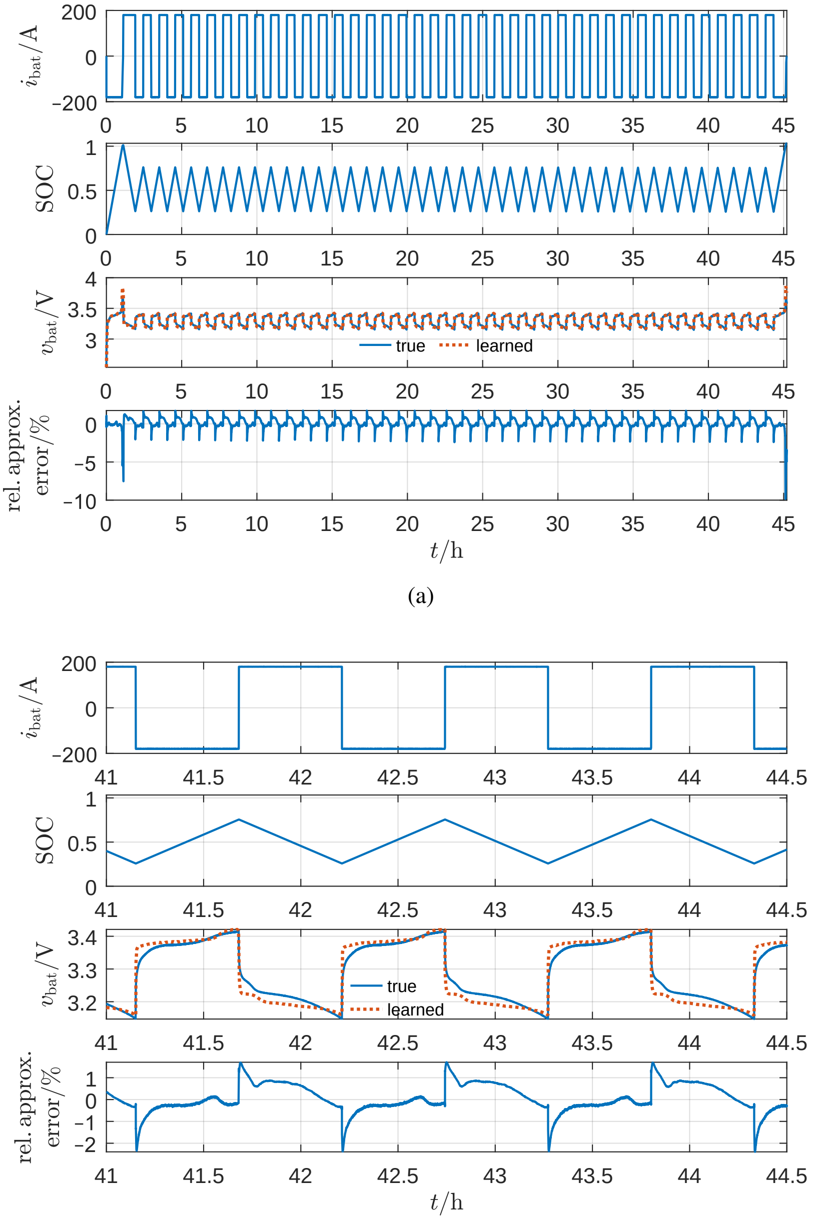
Figure 5 shows the training results for a pulsed current discharge in comparison to the measured battery voltage. In the original contribution [3] we have shown an excerpt of the charge branch for comparison. In contrast to Figure 4 we chose a temporal representation here. Figure 5 shows the voltage curve for a medium SOC. We achieved good results for the voltage response of the battery following a current step. However, the absolute voltage drop is underestimated by the model. The simulation shows an exponential behaviour resulting from the first-order dynamics of the RC element (Equation (5b)) which differs slightly from the experiment. Here, one has to keep in mind that we have chosen a rather simple ECM as a basis for GB modelling. The results are similar for other SOC values and the charge branch.

In conclusion, the training results are in good agreement with the experiment.

3.3 Comparison of Model against Test Data

We used data not included in the training data set to test the final GB model. The measurement data from several consecutive half cycles served as the first test data set. The approximation results are shown in Figure 6 in comparison to the experimental data. Figure 6a shows the complete time series. Overall, the results are in good agreement with the measured battery voltage. Figure 6b focuses on the last three half cycles of the time series. Particularly at the beginning of each half cycle, some deviations occur between simulation and experiment. However, overall the results are good.

The synthetic load profile of a home-storage battery was used as a second test data set. The results are shown in Figure 7. While Figure 7a covers the complete time series Figure 7b focuses on a section in the middle with fast dynamics. The test results are in good agreement with experimental data for the complete load profile. The highest relative approximation errors occur in the area of high SOC values. Note that the training loss was also high at high



(a)

(b)

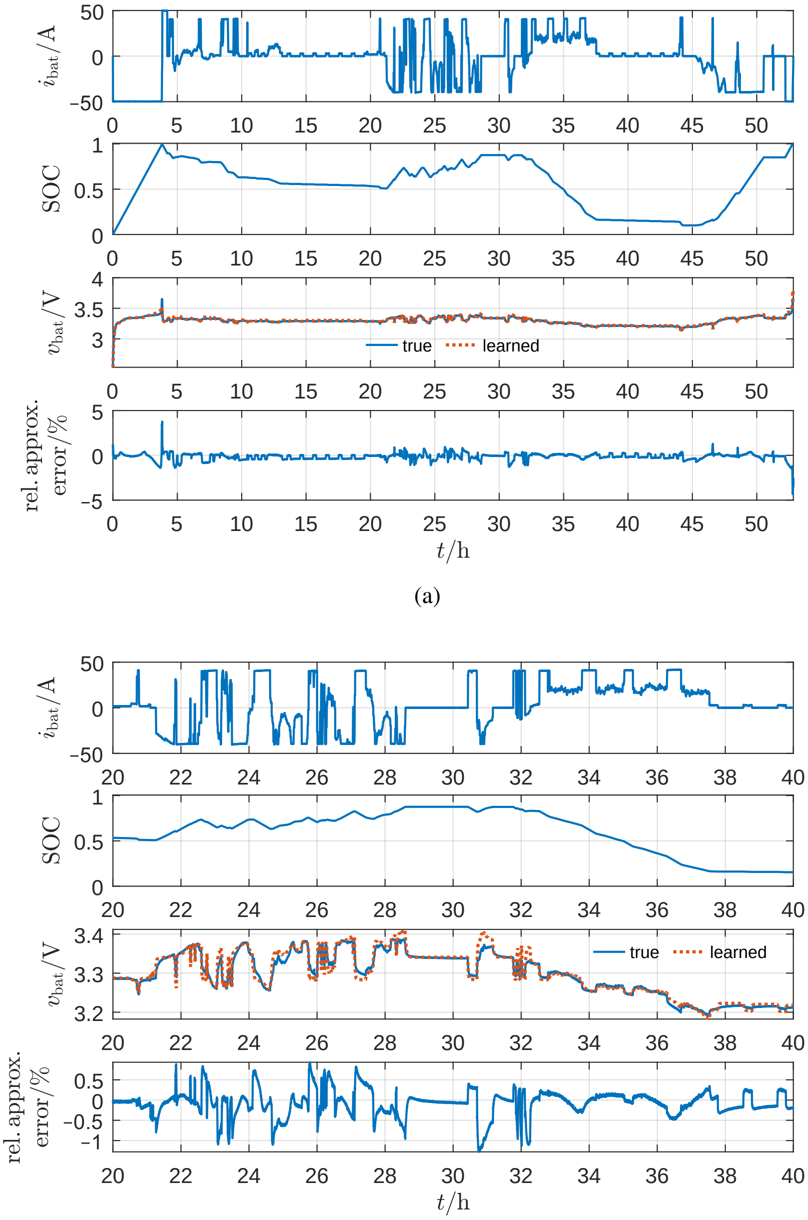
Fig. 6: Test results in comparison to experimental data at $T = 25\,°\mathrm{C}$ for half cycles; (**a**) the complete time series; (**b**) focus on the last three half cycles. [3]. CC BY 4.0 https://creativecommons.org/licenses/by/4.0/

values of SOC. The synthetic load profile covers the longest measuring time which is around 4.5 times as long as the longest training time series. Nevertheless, the test results are good for the whole load profile.

4 Summary and Conclusions

We have shown how to couple physics-based ODEs and NODEs for GB modelling of lithium-ion batteries. The derived model was trained and tested with experimental data of an LFP battery cell that is used in home-storage applications.

As there was only little training data available, we split the training into two steps: first, a simplified static model with neglected double-layer capacitance was trained with CCCV training data. In the second step, the fast dynamics of the battery were taken into account.



(a)

(b)

Fig. 7: Test results in comparison to experimental data at $T = 25\,°\text{C}$ for a synthetic load profile; (**a**) the complete time series (**b**) focus on the segment in the middle. [3]. CC BY 4.0 https://creativecommons.org/licenses/by/4.0/

The final GB model can reproduce the complete set of training data with good accuracy. For the test data, the simulations also show good agreement with the experiments. The highest approximation errors occur where the OCV curve is very steep.

It would be beneficial to have more training data. Especially, the usage of training data from pulse tests with different current steps would be of interest. A bigger data set would also allow a better model validation. A k-fold cross validation could be used to evaluate the robustness of the model against the chosen training data.

Summing up, we have shown how to combine NODEs and physics-based ODEs for GB modelling of lithium-ion batteries.

References

1. Franco, A.A., Doublet, M.L., Bessler, W.G., eds.: Physical Multiscale Modeling and Numerical Simulation of Electrochemical Devices for Energy Conversion and Storage: From Theory to Engineering to Practice. 1 edn. Green Energy and

Technology. Springer, London, London (2016)

2. Seaman, A., Dao, T.S., McPhee, J.: A survey of mathematics-based equivalent-circuit and electrochemical battery models for hybrid and electric vehicle simulation. Journal of Power Sources **256** (2014) 410–423
3. Brucker, J., Behmann, R., Bessler, W.G., Gasper, R.: Neural ordinary differential equations for grey-box modelling of lithium-ion batteries on the basis of an equivalent circuit model. Energies **15**(7) (2022)
4. Estrada-Flores, S., Merts, I., de Ketelaere, B., Lammertyn, J.: Development and validation of “grey-box” models for refrigeration applications: A review of key concepts. International Journal of Refrigeration **29**(6) (2006) 931–946
5. Oussar, Y., Dreyfus, G.: How to be a gray box: dynamic semi-physical modeling. Neural Networks **14**(9) (2001) 1161–1172
6. Duarte, B., Saraiva, P.M., Pantelides, C.C.: Combined mechanistic and empirical modelling. International Journal of Chemical Reactor Engineering **2**(1) (2004)
7. Hamilton, F., Lloyd, A.L., Flores, K.B.: Hybrid modeling and prediction of dynamical systems. PLoS computational biology **13**(7) (2017) e1005655
8. Almeida, G.C.S., de Souza, A.C.Z., Ribeiro, P.F.: A neural network application for a lithium-ion battery pack state-of-charge estimator with enhanced accuracy. Proceedings **58**(1) (2020)
9. Jiménez-Bermejo, D., Fraile-Ardanuy, J., Castaño-Solis, S., Merino, J., Álvaro-Hermana, R.: Using dynamic neural networks for battery state of charge estimation in electric vehicles. Procedia Computer Science **130** (2018) 533–540
10. Yang, D., Wang, Y., Pan, R., Chen, R., Chen, Z.: A neural network based state-of-health estimation of lithium-ion battery in electric vehicles. Energy Procedia **105** (2017) 2059–2064
11. Bills, A., Sripad, S., Fredericks, W.L., Guttenberg, M., Charles, D., Frank, E., Viswanathan, V.: Universal battery performance and degradation model for electric aircraft. ChemRxiv (2020)
12. Brucker, J., Bessler, W.G., Gasper, R.: Grey-box modelling of lithium-ion batteries using neural ordinary differential equations. Energy Informatics **4**(S3) (2021) 1–13
13. Yagci, M.C., Behmann, R., Daubert, V., Braun, J.A., Velten, D., Bessler, W.G.: Electrical and structural characterization of large–format lithium iron phosphate cells used in home–storage systems. Energy Technology **9**(6) (2021)
14. Dreyer, W., Jamnik, J., Guhlke, C., Huth, R., Moskon, J., Gaberscek, M.: The thermodynamic origin of hysteresis in insertion batteries. Nature materials **9**(5) (2010) 448–453
15. Goodfellow, I., Bengio, Y., Courville, A.: Deep learning. Adaptive computation and machine learning. MIT Press, Cambridge, Massachusetts (2016)
16. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: 29th IEEE Conference on Computer Vision and Pattern Recognition, Piscataway, NJ, IEEE (2016) 770–778
17. Chen, R.T.Q., Rubanova, Y., Bettencourt, J., Duvenaud, D.: Neural ordinary differential equations. CoRR **abs/1806.07366** (2018)
18. Haber, E., Ruthotto, L.: Stable architectures for deep neural networks. Inverse Problems **34**(1) (2017)
19. Ruthotto, L., Haber, E.: Deep neural networks motivated by partial differential equations. Journal of Mathematical Imaging and Vision **62** (2020) 352–364
20. Dupont, E., Doucet, A., Teh, Y.W.: Augmented neural odes. In: Advances in Neural Information Processing Systems 32, Red Hook, NY, USA, Curran Associates, Inc. (2019) 3140–3150
21. Zhang, T., Yao, Z., Gholami, A., Keutzer, K., Gonzalez, J., Biros, G., Mahoney, M.W.: Anodev2: A coupled neural ode evolution framework. CoRR **abs/1906.04596** (2019)
22. Haber, E., Ruthotto, L., Holtham, E., Jun, S.H.: Learning across scales - multiscale methods for convolution neural networks. In: Proceedings of the AAAI Conference on Artificial Intelligence. Volume 32., Palo Alto, California USA, AAAI Press (2018)
23. Gholami, A., Keutzer, K., Biros, G., Gholaminejad, A.: Anode: Unconditionally accurate memory-efficient gradients for neural odes. In: Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence. (2019) 730–736
24. Rackauckas, C., Ma, Y., Martensen, J., Warner, C., Zubov, K., Supekar, R., Skinner, D., Ramadhan, A., Edelman, A.: Universal differential equations for scientific machine learning. CoRR **abs/2001.04385** (2020)
25. He, H., Xiong, R., Fan, J.: Evaluation of lithium-ion battery equivalent circuit models for state of charge estimation by an experimental approach. Energies **4**(4) (2011) 582–598
26. Wang, Y., Fang, H., Zhou, L., Wada, T.: Revisiting the state-of-charge estimation for lithium-ion batteries: A methodical investigation of the extended kalman filter approach. IEEE Control Systems **37**(4) (2017) 73–96
27. Braun, J.A., Behmann, R., Schmider, D., Bessler, W.G.: State of charge and state of health diagnosis of batteries with voltage-controlled models. Journal of Power Sources **544** (2022)
28. Weißhar, B., Bessler, W.G.: Model-based lifetime prediction of an lfp/graphite lithium-ion battery in a stationary photovoltaic battery system. Journal of Energy Storage **14** (2017) 179–191
29. LeCun, Y., Bottou, L., Orr, G.B., Müller, K.R.: Efficient backprop. In Orr, G.B., Müller, K.R., eds.: Neural Networks: Tricks of the Trade. Springer Berlin Heidelberg, Berlin, Heidelberg (1998)
30. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S.: Pytorch: An imperative style, high-performance deep learning library. In: Advances in Neural Information Processing Systems 32, Red Hook, NY, USA, Curran Associates, Inc. (2019) 8024–8035
31. Chen, R.T.Q.: torchdiffeq (version 0.2.1) (2021) `https://github.com/rtqichen/torchdiffeq`

Object Classification with a Robot Gripper equipped with Force Sensitive Fingertips using Convolutional Neural Networks

Christoph Uhrhan[1], Philipp Triebold[1] and Orion Franz Lorenz Salas[2]

[1]Furtwangen University
{uc,philipp.triebold}@hs-furtwangen.de
[2]Frickly Systems GmbH
info@frickly.systems

Abstract. In this paper we present a solution for the identification and classification of grasped objects with an electrical robot gripper with force sensor arrays in its fingertips. The solution is based on a convolutional neural network (CNN). The CNN is trained with relatively few examples but gives already reasonable results. Objects to be detected are of geometrical shape like ring, pen, sphere. The challenges in such applications are the generation of random training data and interfaces between the different components such as gripper, sensor array fingertips and robot. The trained CNN is ported to a Raspberry Pi for real-time execution and communication between the gripper and the robot.

Keywords: force sensing robot gripper; convolutional neural network, object classification; bin picking.

1 Introduction

The flexibility of robots depends not at least on the flexibility of the robot gripper. Beside versatile finger kinematics [1] universal grippers need appropriate sensors to interact with the environment. The mostly used sensors are cameras to detect the correct grasp position [2], [3], [4]. As more sophisticated the gripper become as more time consuming is the classical programming of grasping procedures. Several approaches are proposed using Artificial Intelligence to learn the correct grasping [5], [6]. Standard electrical grippers offer already the possibility to adjust and measure position and the applied force of the finger. Additional sensitive fingertips with a force sensor array give additional information about the location size and shape of the grasped object.

Nevertheless, the interpretation of this additional information is not always unique and/or difficult to program. [7] presents two approaches to determine the elasticity of the grasp object using force sensing in the fingertips. The following presented approach interprets fingertip force sensor data to classify objects by its shape using convolutional neural networks (CNN). This reduces the programming effort to identify specific objects. This approach is a contribution to make fingertip sensors easier to use in industrial robot applications, for example for bin picking applications.

2 Experimental Setup

The base of our experiments is the electrical gripper WSG50 from Weiss Robotics (Figure 1) equipped with forces sensor matrix in the fingertips.

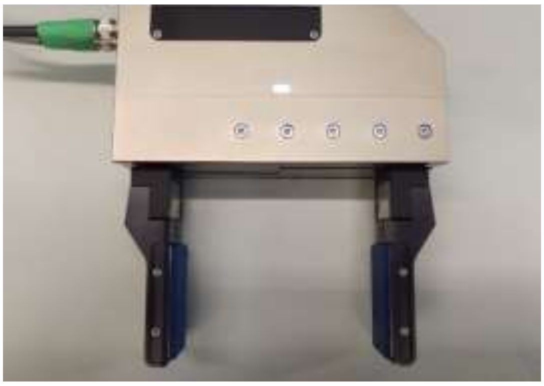


Fig. 1. Electrical Gripper from Weiss Robotics

2.1 Fingertip Sensor

The fingertip sensor provides a force distribution over the fingertip area (Figure 2). The output is a 6 x 14 matrix with force/pressure values in the range of 20 to 250 kPa for each force pixel with a pixel size of 3.4 x 3.4 mm².

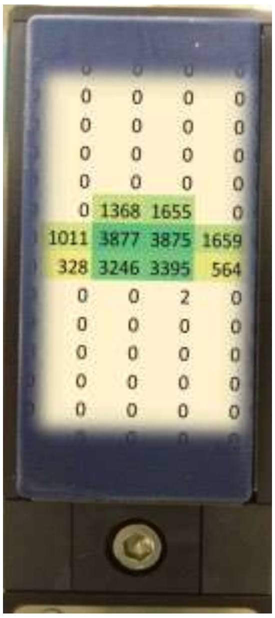


Fig. 2. Sensor-Array in the Fingertips

2.2 Data Organisation

To make the handling of the sensor data more universal, the two arrays of the fingers are put together to one matrix which represents the force distribution of both finger matrices in a 6 x 28 matrix (Figure 3). Like this the data can easily be used within Python and the Keras library.

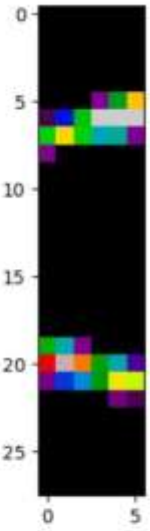


Fig. 3. Combined Sensor-Data-Array from 2 Fingers

2.3 Data Generation

A big challenge to use Neural Networks is the generation of sufficient data sets. In this approach we used an automated randomized grasping sequence where the objects are swinging between the fingertips and the gripper closes randomly (Figure 4). Additionaly the length of the pendulum has been varied. Like this about 950 data sets has been generated per object. The selected objects are shown in Figure 5.

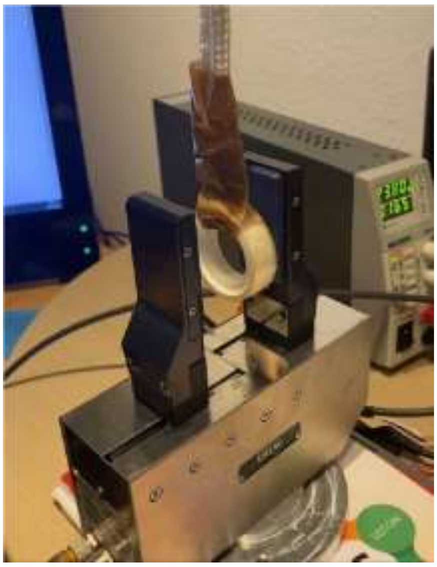


Fig. 4. Setup for the data generation

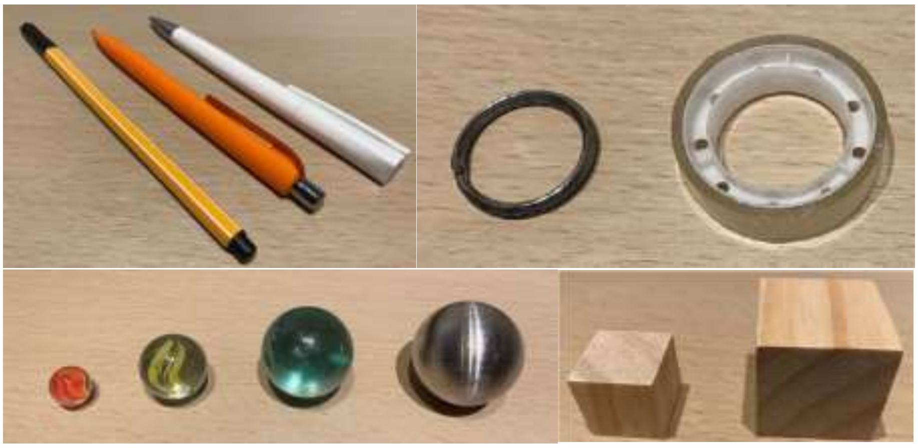


Fig. 5. Objects to be identified

3 Object Classification using Convolutional Neural Networks

Convolutional Neural Networks (CNN) has been selected because these are known that they need relative few training data sets for the classification. Several convolutional neural network structures have been tested from very simple one to more advanced ones [8].

3.1 Simple CNN with 2 Convolutional Layers

The first approach was a very simple CNN with 2 convolutional layers and 1 max pooling layer only. With this first reference approach an accuracy of 50% could be achieved. This value could be optimized by manually eliminating obvious wrong training data sets, e. g. where the objects has only touched very few force pixels. Like this the accuracy could be increased to 87% already.

3.2 Improved CNN

To further increase the accuracy, several CNN has been implemented and tested. The final CNN has 21 layers and 11.779 parameters (Table 1). This network achieved an accuracy of 98,65% for the test data and 99,47% for the training data. This is a quite good result for only less than 1000 training sets. The forecast of the network shows the confusion matrix in Table 2.

Using the test data sets the network did only one wrong forecast. One pen has been classified as a ring. The reason of the miss-interpreted set shows Figure 6. For a bigger ring the image is very similar to a pen which is slightly inclined within in the grippers.

Table 1. Final CNN structure

Layer Type	Result Values/Dimensions	No. of Parameters
Input	-	-
Convolutional 2D	(None, 358, 78, 10)	280
Convolutional 2D	(None, 368, 76, 10)	910
MaxPoolling2D	(None, 178, 38, 10)	0
Dropout	(None, 178, 38, 10)	0
Convolutional 2D	(None, 176, 36, 10)	910
Convolutional 2D	(None, 174, 34, 10)	910
MaxPoolling2D	(None, 78, 17, 10)	0
Dropout	(None, 78, 17, 10)	0
Convolutional 2D	(None, 85, 15, 10)	910
Convolutional 2D	(None, 83, 13, 10)	910
MaxPoolling2D	(None, 41, 6, 10)	0
Dropout	(None, 41, 6, 10)	0
Convolutional 2D	(None, 39, 4, 10)	910
Convolutional 2D	(None, 37, 2, 10)	910
MaxPoolling2D	(None, 18, 1, 10)	0
Dropout	(None, 18, 1, 10)	0
Flatten	(None, 180)	0
Dense	(None, 64)	11584
Dense	(None, 3)	195
Output	(1)	-

Table 2. Confusion Matrix

	Forecast of the Network		
Correct Value	**Sphere**	**Ring**	**Pen**
Sphere	60	0	0
Ring	0	80	1
Pen	0	0	41

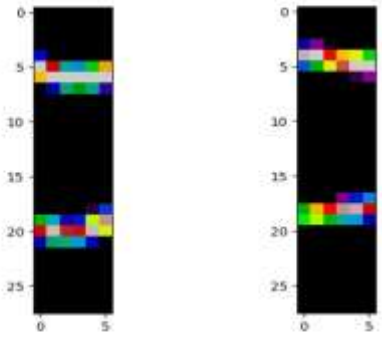


Fig. 6. Comparison ring (left) vs. pen (right)

4 Real-Time Robot Setup

The data generation and training of the network has been done on a standard computer with a NVIDIA GPU. For the final robot application, the trained network has been ported to a Raspberry Pi. The Raspberry Pi functions as the interface between the gripper and a robot (Universal Robot UR5). On the one hand the basic functions of the gripper WSG50 should be made accessible to the UR5 robot. On the other hand, the sensor data, read from the gripper fingers, should be evaluated and the grasped object classified (Figure 7). To reach both goals an interface between the robot and the gripper is made using a Raspberry Pi. The Raspberry Pi communicates with the gripper over TCP/IP and with the robot over Modbus TCP.

Additional to the communication between the components the Raspberry Pi should do the analysis of the sensor data, means the trained CNN should be executed on the Raspberry Pi itself. A special function has been implemented that reads and classifies the data from the gripper fingers. This classification is done with the Tensorflow-Lite module. The trained model file has then just to be stored on Raspberry Pi.

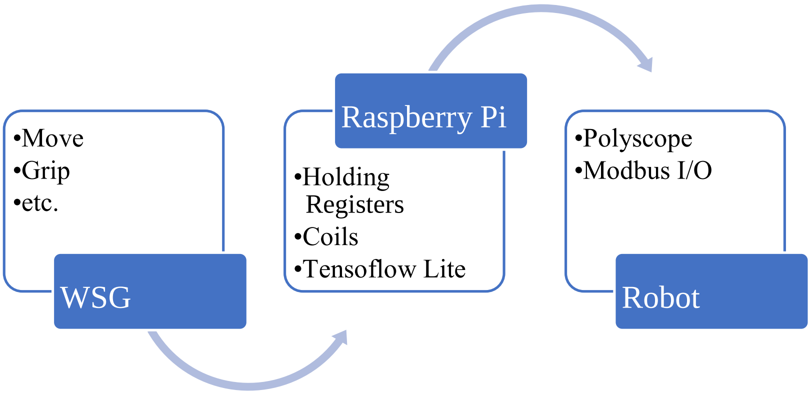


Fig. 7. Communication between Gripper, Raspberry Pi and Robot

5 Conclusion

In this paper we present a solution for the identification and classification of grasped objects with an electrical gripper with force sensor arrays in its fingertips. The solution is based on a convolutional neural network (CNN). The CNN is trained with relatively few examples but gives already reasonable results. Objects to be detected are of geometrical shape like ring, pen, sphere. The challenges in such applications are the generation of random training data and interfaces between the different components such as gripper, sensor array fingertips and robot. The trained CNN has been ported to a Raspberry Pi for real-time execution and communication between the gripper and the robot.

Further research will be to use the finger positions and motor currents of the gripper additionally to the force sensor matrix to expand the data sets. Like this even more details of the grasped object should be trainable. In combination with a classical approach to detect the object location (position and orientation) inside the gripper which uses the centre of gravity and axis of inertia [9], the grasped objects can not only be classified but be moved by the robot the goal position with the required orientation.

With these approaches, new solutions for the bin-picking-problem without camera or with cameras with less resolution might be possible.

References

1. Tadaaki Hasegawa, et. al: Powerful and Dexterous Multi-Finger Hand Using Dynamical Pulley Mechanism, 39th IEEE International Conference on Robotics and Automation, ICRA 2022, 23-27 May 2022
2. Xinghao Zhu, et. al.: Learn to Grasp with Less Supervision: A Data-Efficient Maximum Likelihood Grasp Sampling Loss, 39th IEEE International Conference on Robotics and Automation, ICRA 2022, 23-27 May 2022

3. Xibai Lou, Yang Yang, Changhyun Choi: Learning Object Relations with Graph Neural Networks for Target-Driven Grasping in Dense Clutter, 39th IEEE International Conference on Robotics and Automation, ICRA 2022, 23-27 May 2022
4. Mayer, V., et. al.: FFHNet: Generating Multi-Fingered Robotic Grasps for Unknown Objects in Real-Time, 39th IEEE International Conference on Robotics and Automation, ICRA 2022, 23-27 May 2022
5. Xinghao Zhu, et. al.: Learn to Grasp with Less Supervision: A Data-Efficient Maximum Likelihood Grasp Sampling Loss, 39th IEEE International Conference on Robotics and Automation, ICRA 2022, 23-27 May 2022
6. Xibai Lou, Yang Yang, Changhyun Choi: Learning Object Relations with Graph Neural Networks for Target-Driven Grasping in Dense Clutter, 39th IEEE International Conference on Robotics and Automation, ICRA 2022, 23-27 May 2022
7. Gabler, V., Huber, G., Wollherr, D.: A Force-Sensitive Grasping Controller Using Tactile Gripper Fingers and an Industrial Position-Controlled Robot, 39th IEEE International Conference on Robotics and Automation, ICRA 2022, 23-27 May 2022
8. Lorenz Salas, F.: Verbesserung des SMARTGRIPPERs im Robotik Labor mit Fokus auf Automatisierung, Bachelor-Thesis, Fakultät Wirtschaftsingenieurwesen, Hochschule Furtwangen, Wintersemester 2020/21
9. Schmidt, Ch.: Intelligentes Greifen für flexible Roboteranwendungen, Bachelor-Thesis, Fakultät Wirtschaftsingenieurwesen, Hochschule Furtwangen, Wintersemester 2017/18

Predicting critical machining conditions using time-series imaging and deep learning in slot milling of titanium alloy

Faramarz Hojati, Bahman Azarhoushang

Institute of Precision Machining (KSF) – Furtwangen University
hofa@hs-furtwangen.de
aza@hs-furtwangen.de

Abstract. Tool wear and tool breakage cause product damage in terms of low surface quality and undesired geometrical and dimensional tolerances, followed by a dramatic increase in the production cost. In this study, an Artificial Intelligence (AI) model has been developed to predict the critical machining conditions concerning surface roughness and tool breakage in the slot milling of titanium alloy. The signals recorded from the main spindle and different axes through the Siemens SINUMERIK EDGE Box integrated into a CNC machine tool were converted into images using Gramian Angular Field (GAF). Further, the converted images were used for training Convolutional Neural Network (CNN). The combination of GAF and trained CNN model indicates good performance in predicting critical machining conditions, particularly in the case of an imbalanced dataset.

Keywords: Artificial Intelligence, Gramian Angular Field, Convolutional Neural Network, Slot-milling, Edge Box, Imbalanced dataset.

1 Introduction

The cutting tool breakage or severe tool wear causes low product quality and damage to machine components. Therefore, the tool must be changed before these unwanted events. However, the tool cost increases with earlier tool changes. To avoid quality failure by changing the cutting tool at an appropriate time interval, the tool and process condition should be monitored using different sensors integrated into the machine tool [1]. In recent years, extensive research works have been conducted to monitor the cutting tool condition during the process for optimizing tool lifespan [2], early detection of tool wear, and prevention of tool breakage [3]. Direct monitoring of the cutting tool, which measures the tool geometry using vision or optical apparatus, requires expensive equipment and cannot be applied in real-time due to the presence of coolant and the contact between the tool and material [4]. Therefore, the focus of research activities was mainly on indirect approaches of tool and machining condition monitoring, which benefit from the fact that a variation in cutting tool condition changes certain variables such as cutting forces, vibration, and surface finish. In these methods, the current or power signals from the machine elements (like spindle or axis motors) [5,6] or signals from the sensors integrated into the machine tool (like piezosensor, accelerometer, strain gauge, thermocouple or acoustic emission sensor) [7–9] are analyzed to recognize the possible correlations with the cutting tool and machining condition.

The tool and process condition monitoring methods, generally need to extract the features of measured signal to use them for the model training. In this regard, the application of the utilized algorithm in the extraction of features varies from signal to signal, and the feature extraction requires a well-experienced person. Moreover, some information on the signal through the feature extraction would be lost. To solve this issue, imaging the signals using different approaches like Gramian Angular Field rather than feature extraction can be helpful. As an example, Arellano et al. [10] applied GAF for tool wear classification. The recorded cutting force signals were encoded to several images that have been used for training Convolutional Neural Network (CNN) classification model. A percentage of accuracy over 80% was reported for different groups corresponding to different states of tool wear (break-in, steady-state, and failure).

A need for analyzing the signal for process monitoring is growing with a new generation of machine tools capable of recording different types of data. The Siemens SINUMERIK EDGE (SE) Box that can be integrated into the machine tool record the signals from different axes and main spindles and fuse all measured data into a JSON file. The signals for recording are selected in the MindSphere Capture4Analysis application, which is connected with SINUMERIK EDGE Box. The current study aims to predict the machining condition using different types of signals recorded by the SINUMERIK EDGE Box integrated into the five axes CNC machine tool (Haas-Multigrind® CA). The experimental tests were conducted in milling a titanium alloy (Ti6Al4V) as a difficult-to-cut material. Severe tool wear and even tool breakage due to Built-Up Edge (BUE), particularly at high-speed machining of the titanium alloy, followed by low workpiece surface quality, was detected. After measuring signals, GAF, as one of the time series imaging methods, was applied for encoding the signals into images. Further, images were used for training the CNN classification model to predict the critical machining condition. Eventually, the combination of the GAF and CNN model was evaluated.

2 Experiment

In this investigation, Ti6Al4V was selected as the workpiece material. Slot-milling was used as a milling strategy. The milling tests were carried out with different process parameters. In all tests, the axial depth of cut a_p was kept constant and equal to 1 mm, and the radial depth of cut a_e was 3 mm. The feed per tooth, f_z, and cutting speed v_c were considered as varying parameters. Figure 1a illustrates the experimental setup. The milling tool moved from the right side to the left side of the workpiece (according to the milling direction shown in Figure 1a), and for the creation of slots with a depth of 6 mm, six slot-milling passes each with axial depth of cut, a_p, of 1 mm were conducted. 161 tests (each test including 6 slot milling passes) were carried out with different combinations of varying process parameters. The tests were conducted with and without coolant lubricant. In the presence of cooling, no tool wear and tool breakage were observed. Therefore, several tests were also carried out without coolant lubricant (dry cutting) at the higher range of feeds and cutting speeds to increase the tool wear rate and reduce the required experimental time. Table 1 provides the range of milling parameters.

Table 1. Process parameters

Cutting speed v_c [m/min]	**Feed per tooth f_z [µm/tooth]**	**Radial depth of cut a_e [mm]**	**Axial depth of cut a_p [mm]**	**Coolant**
50-113	17-50	3	1	Oil / Dry

Figure 1b indicates the integration of the whole data acquisition system into the utilized milling machine. A five axes CNC machine tool (Haas-Multigrind® CA) with a Siemens controller was used in this study. For the data acquisition, the machine tool is equipped with a so-called Siemens SINUMERIK EDGE (SE) Box. Siemens CNC controls supply data, and SE makes it possible to record data and states of the control in a resolution of 1 ms (1kHz) parallel to the process. The SE box is, in principle, an industrial computer and has the corresponding resources to store the data. The MindSphere Capture4Analysis application enables the selection of the signals to be recorded and the trigger time from which a signal is to be recorded. The types of signals that Edge-Box for the main spindle and each axis can record are categorized into current, load, torque and power. At each milling pass, the Edge-Box started to automatically record the signals slightly prior to the slot milling process and the recording was automatically stopped after the cutting process. According to a defined system, the data is written to a JSON file on the hard disk of the SE box with the execution of the NC program. With a correspondingly high number of milling attempts, hundreds or thousands of files can be created. Further, the JSON File for each test is processed through a written ETL program (Extract-Transform-Load) to obtain the tabular data in CSV format. The CSV data and post-process information from the measurement system collected in tabular data are imported directly into Artificial Intelligent (AI) Model. Moreover, the CSV data can also be visualized using an external computer for the machine tool user.

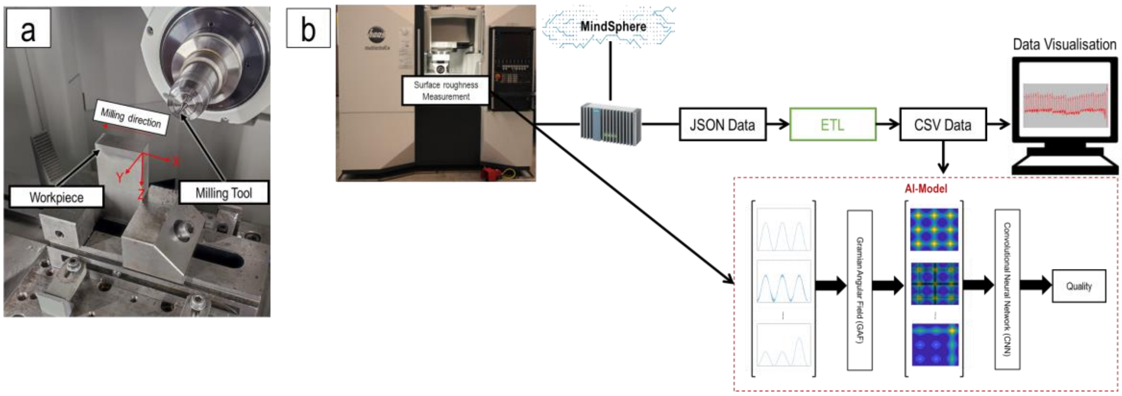
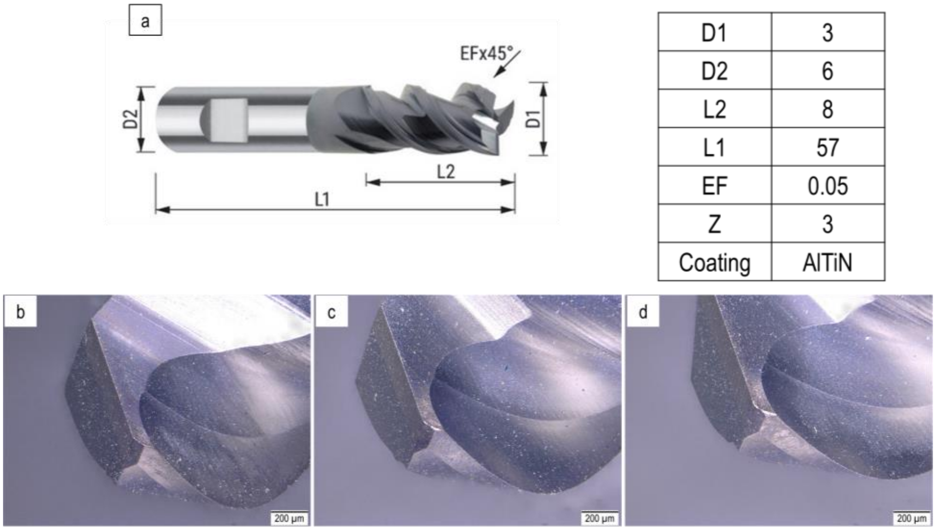


Fig. 1. (a) experimental setup (b) schematic representation of the integration of the ETL program and AI Model

The concept of the utilized AI Model is shown in Figure 1b. Instead of using the signals features such as mean, peak and standard deviation, each signal was converted into an image by Gramian Angular Field (GAF), which contains all the features as well as the relationships between different points of the signal. In the next step, the images were used as input parameters for training the CNN model. Finally, the model can predict the quality of the process concerning the critical machining condition in terms of machined surface roughness and tool breakage with respect to new images obtained from new signals (even with different process parameters).



D1	3
D2	6
L2	8
L1	57
EF	0.05
Z	3
Coating	AlTiN

Fig. 2. (a) specification of utilized cutting tool (b) edge 1 (c) edge 2 (d) edge 3

Figure 2a shows the geometrical properties of the utilized tools. The milling tools were coated with AlTiN. D1, D2, L1, L2, EF and Z are cutting tool and shank diameter, total length, cutting edge length, corner chamfer and the number of teeth, respectively. Figure 2 (b-d) also demonstrated three cutting edges of a new tool.

3 Signal selection

According to the recorded signals, it was concluded that the signals in the z-axis for current, load and torque are the best candidates for the model training. In detail, the signals in other axes showed no remarkable change before tool breakage, while the mentioned types of signals in the z-axis considerably altered. Figure 3 illustrates an example of these signals before the tool breakage. After the second pass, a considerable fluctuation in all types of the signal can clearly be observed. This can be associated with considerable Built-Up Edge (BUE) at the dry milling of titanium alloys that was eventually followed by tool breakage. Since the response of three different types of signals (current, load and torque) are similar, using all of them for training the model is not required. Therefore, the load signal was used for further analysis. Figure 4 shows a considerable BUE at v_c = 80 m/min and f_z = 20.6 µm/tooth that led to cutting edge and tool breakage.

Tool	3 tooth / Φ = 3 mm / AlTiN
Workpiece	Ti6Al4V; Extruded
Parameters	Slot-milling, v_c = 80 m/min , f_z = 20.6 µm/tooth , a_p = 1 mm, a_e = 3 mm
Coolant	Oil

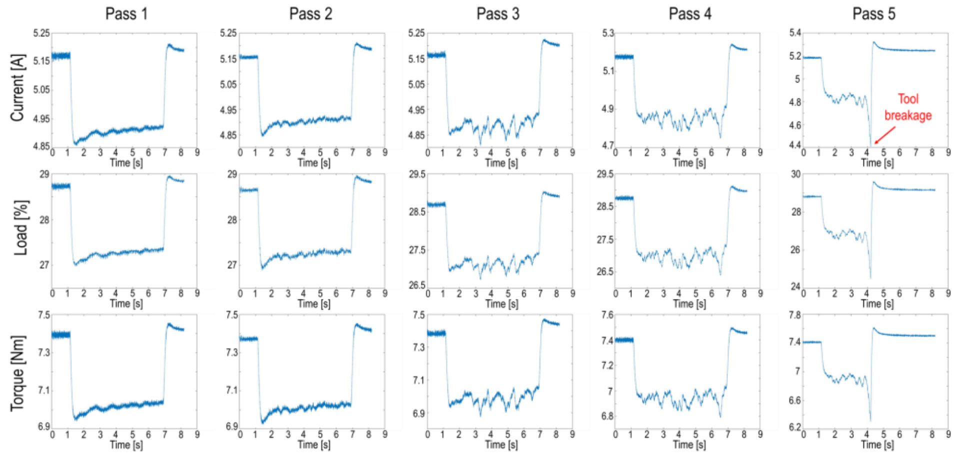


Fig. 3. Signals of Current, Load and Torque in different passes before tool breakage at v_c = 80 m/min and f_z = 20.6 µm/tooth

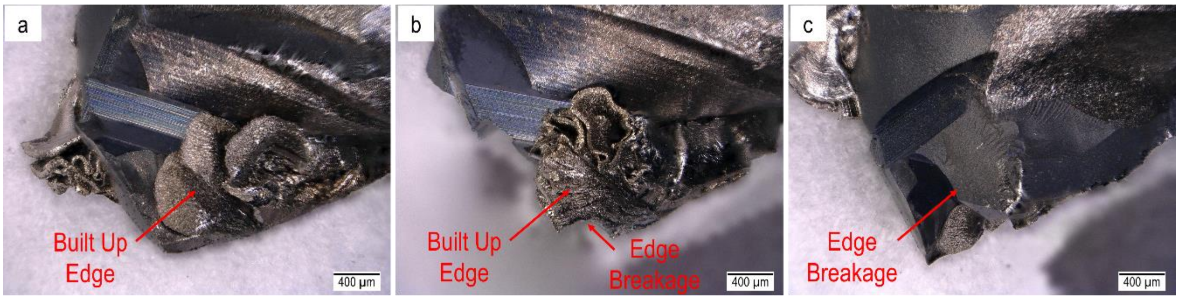


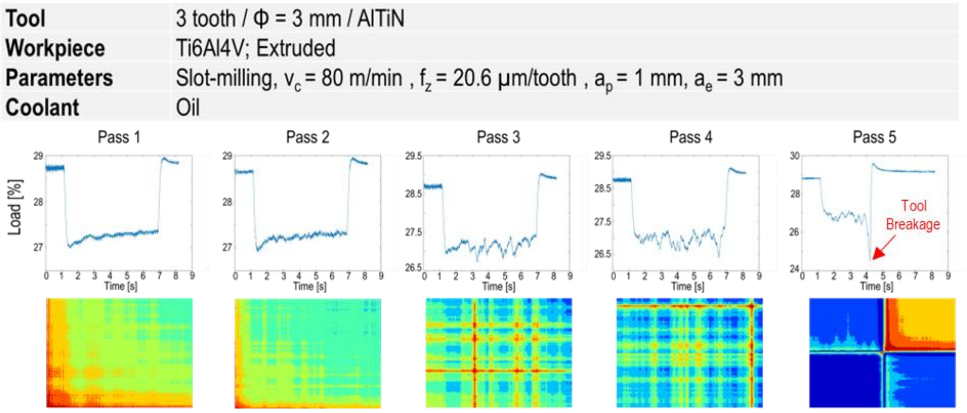
Fig. 4. BUE and Tool breakage at v_c = 80 m/min and f_{z} = 20.6 µm/tooth (a) edge1 (b) edge2 (c) edge3

4 Gramian Angular Field (GAF)

The **GAF** algorithm encodes the time-series signal into an image, resulting in transferring the signal to a polar coordinate space. Using Gramian Angular Summation Field (**GASF**), the trigonometric sum is applied to each couple of angular positions of each time series point and the rest of the signal points for generating each row of the GASF matrices. Therefore, the temporal correlation between each point of the signal and the rest of the signal is calculated at each row as below:

$$\mathrm{GASF} = \begin{bmatrix} \cos(\emptyset_1 + \emptyset_1) & \cdots & \cos(\emptyset_1 + \emptyset_n) \\ \cos(\emptyset_2 + \emptyset_1) & \cdots & \cos(\emptyset_2 + \emptyset_n) \\ \vdots & \ddots & \vdots \\ \cos(\emptyset_n + \emptyset_1) & \cdots & \cos(\emptyset_n + \emptyset_n) \end{bmatrix} \tag{1}$$

Figure 5 provides the corresponding GASF images of the signals for a series of 5 pass-milling before the tool breakage. At the first two passes, no remarkable change in the image (and correspondingly the row signals) can be observed, while afterwards, a dramatic change in images (alteration in the colors and pattern of images) associated with the signal fluctuation can clearly be seen that is followed by the tool breakage in the 5th pass.

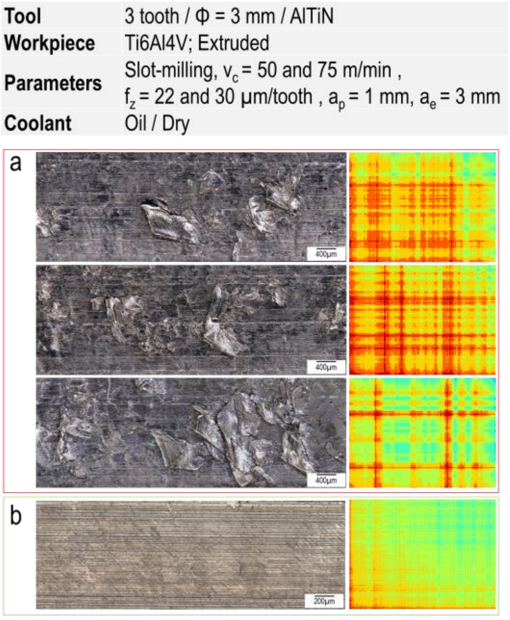


Tool	3 tooth / Φ = 3 mm / AlTiN
Workpiece	Ti6Al4V; Extruded
Parameters	Slot-milling, v_c = 80 m/min , f_z = 20.6 µm/tooth , a_p = 1 mm, a_e = 3 mm
Coolant	Oil

Fig. 5. GASF images at five different milling passes with a new tool at constant process parameters

5 Clustering

Before training the CNN model, the images were clustered into two main groups (A and B). Group A includes images of the tests where no tool breakage occurred, and acceptable surface quality was induced. Group B contains all images belonging to the experimental tests where tool breakage occurred, or the surface quality deteriorated dramatically. Figure 6a and Figure 6b indicate respectively the quality of milled surface associated with group A and group B and their corresponding GASF images. 858 GASF images were produced in this study. Based on the assumption regarding the clustering of images in two groups, 726 and 132 images corresponded to group A and group B, respectively.



Tool	3 tooth / Φ = 3 mm / AlTiN
Workpiece	Ti6Al4V; Extruded
Parameters	Slot-milling, v_c = 50 and 75 m/min , f_z = 22 and 30 µm/tooth , a_p = 1 mm, a_e = 3 mm
Coolant	Oil / Dry

Fig. 6. Exemplary milled surfaces and their corresponding GASF images (a) at v_c = 75 m/min and f_z = 22 µm/tooth in dry machining and (b) at v_c = 50 m/min and f_z = 30 µm/tooth with oil

6 Convolutional Neural Network (CNN)

A Convolutional Neural Network (CNN), a commonly applied method for machine learning of images, was used as a classification model in this study. Figure 7 illustrates the architecture of the developed CNN model. Two main components in this model are the convolution layer and the pooling layer. In the convolution layer, multiple filters are applied to the imported image. Each filter scans the entire image, and at each position, the similarity of the filter is compared with that area of the image. The output of the Convolution Layer results in several images that are smaller than the original image. The number of images corresponds to the number of applied filters. In the next step, the Pooling Layer is applied to reduce the size of the images and, correspondingly, the number of parameters to be learned as well as the number of computations performed in the network. Moreover, the pooling layer extracts the most important features of the image. Therefore, the images are imported from the convolution layer to the pooling layer, which results in a dimension reduction while preserving the important features of the images. Finally, the images are flattened and imported into the neural network for training.

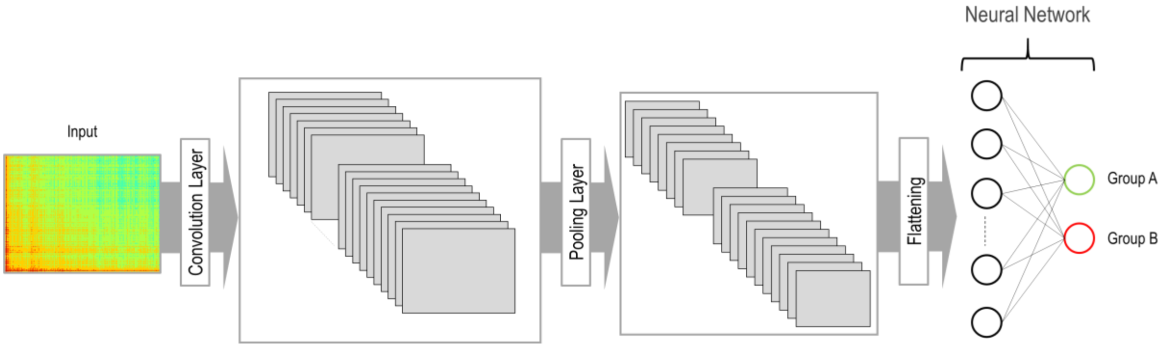


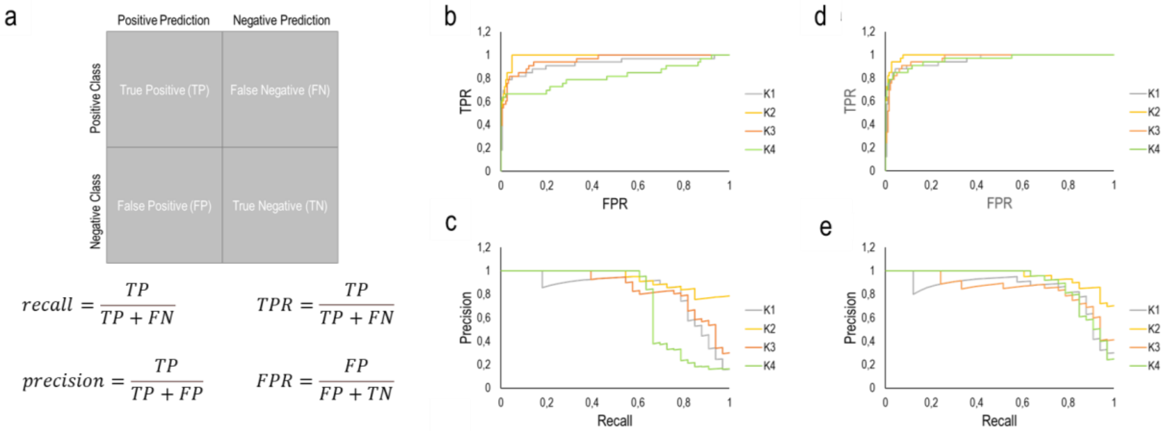
Fig. 7. Architecture of the CNN model

After creating the images using the GASF method, they are resized to 224×224 for importing into the CNN model. In this study, two convolution layers were determined. The number and size of filters for each convolution layer accounted for 128 and 3×3, respectively. After each convolution layer, a pooling layer is placed with a pool size of 2×2. Therefore, the imported images (with the size of 224×224) after the first convolution layer are reduced to 222×222. Further, 128 images obtained from the first convolution layer are further subjected to the pooling layer, which results in 128 images with the size of 111×111. Afterwards, the output of the second convolution layer results in images with the size of 109×109. By applying the second pooling layer, the size of 128 images is eventually reduced to 54×54. The Rectified Linear Unit (ReLU) function is applied as an activation function at both convolution and pooling layers. In the next step, two layers in a neural network are determined. The number of neurons at the first and second layers accounted for 256 and 2, respectively. Due to binary classification, the sigmoid function is used as an activation function in the last layer.

7 Model evaluation

To evaluate better the trained model, the K-fold cross-validation approach was used. In this method, the dataset was divided into K groups, where K-1 groups are used for the training model and one group is used for the test. Again, the test group is changed, and the rest of the dataset is used for training. This procedure is repeated K times, and the average accuracy is considered as a good indicator for the future prediction of the model. In this study, K is set to 4 so that the data set for group A and group B has been divided into four groups. As mentioned before, the number of images in group A and B are 726 and 132, respectively. For each round of training, 545 and 181 images for group A are used for training and testing, respectively. In the case of group B, the number of images for training and testing accounted for 99 and 33, respectively.

Due to this fact that the accuracy is not a good indicator for model evaluation in the case of imbalanced dataset, other approaches such as Receiver Operator Characteristic (ROC) and Precision-Recall curves were used to evaluate a binary classifier for testing each group. For generating these curves, True Positive Rate (TPR) against False Positive Rate (FPR) and Precision versus Recall are plotted at various threshold values determined for sigmoid function in the last layer of the neural network. The calculation of Precision, Recall, TPR and FPR with respect to the confusion matrix are provided in Figure 8a. The classifier that provides a curve close to the top-left corner indicates a better performance based on the ROC curve. According to Figure 8b, the trained model at K = 2 shows better performance compared to others. The lowest performance concerning the ROC curve can be seen for K = 4. According to Figure 8c, it is desired that the classifier model has higher precision and recall. Therefore, the Precision-recall curve closer to the top-right corner shows better performance. Correspondingly, the trained model classifier at K = 1, 2, and 3 performed better than that at K = 4. A variation in performance of the trained model in different groups (K = 1, 2, 3 and 4) shown in Figure 8b and Figure 8c is related to the imbalanced dataset issue that the model is mainly trained by group A rather than group B. Figure 8d and Figure 8e illustrate the ROC curve and Precision-Recall curve, respectively, after oversampling the dataset through the image augmentation technique by ImageDataGenerator from the KERAS library at Python. Accordingly, a variation between curves has been reduced, and the Classifier at different groups indicates good performance. This highlighted the issue of the imbalanced dataset in the training of the model. Therefore, the extension of the dataset, particularly for group B, is important to assist the model training by reducing a degree of imbalance in the dataset.



$$recall = \frac{TP}{TP + FN} \qquad TPR = \frac{TP}{TP + FN}$$

$$precision = \frac{TP}{TP + FP} \qquad FPR = \frac{FP}{FP + TN}$$

Fig. 8. (a) confusion matrix **(b)** ROC curve **(c)** Precision-Recall curve **(d)** ROC curve after oversampling **(e)** Precision-Recall curve after oversampling

8 Conclusion

In this study, the Gramian Angular Field (GAF) method was applied after collecting the process signals through an Edge computing device (Siemens SINUMERIK EDGE (SE) Box) integrated into the machine. The captured signals were encoded into images using GAF. Additionally, a CNN classification model was developed to predict the critical process conditions in milling a titanium alloy (Ti6Al4V). The developed AI model was able to predict the critical process conditions in terms of tool breakage and low surface quality in the slot milling of Ti6Al4V even in the presence of an imbalanced dataset. The improvement of the model in the future can be carried out using expanding the dataset, particularly for collecting more experimental data associated with the critical machining condition.

References

1. Ambhore, N., Kamble, D., Chinchanikar, S., Wayal, V.: Tool Condition Monitoring System: A Review. Materials Today: Proceedings 2(4-5) (2015) 3419–28
2. Teti, R., Jemielniak, K., O'Donnell, G., Dornfeld, D.: Advanced monitoring of machining operations. CIRP Annals 59(2) (2010) 717–39
3. Mohanraj, T., Shankar, S., Rajasekar, R., Sakthivel, NR., Pramanik, A.: Tool condition monitoring techniques in milling process — a review. Journal of Materials Research and Technology 9(1) (2020) 1032–42
4. Nouri, M., Fussell, BK., Ziniti, BL., Linder, E.: Real-time tool wear monitoring in milling using a cutting condition independent method. International Journal of Machine Tools and Manufacture 89 (2015) 1–13
5. Patra, K., Jha, AK., Szalay, T., Ranjan, J., Monostori, L.: Artificial neural network based tool condition monitoring in micro mechanical peck drilling using thrust force signals. Precision Engineering 48 (2017) 279–91.
6. Drouillet, C., Karandikar, J., Nath, C., Journeaux, A-C., El Mansori, M., Kurfess, T.: Tool life predictions in milling using spindle power with the neural network technique. Journal of Manufacturing Processes 22 (2016) 161–8
7. Zhang, XY., Lu, X., Wang, S., Wang, W., Li, WD.: A multi-sensor based online tool condition monitoring system for milling process. Procedia CIRP 72 (2018) 1136–41
8. Hesser, DF., Markert, B.: Tool wear monitoring of a retrofitted CNC milling machine using artificial neural networks. Manufacturing Letters 19 (2019) 1–4
9. Siddhpura, M., Paurobally, R.: A review of chatter vibration research in turning. International Journal of Machine Tools and Manufacture 61 (2012) 27–47
10. Martínez-Arellano G, Terrazas G, Ratchev S. Tool wear classification using time series imaging and deep learning. Int J Adv Manuf Technol 104(9-12) (2019) 3647–62

Searching for Feature Sets for Misalignment Classification Using Experimental Data and Data Mining

Sebastian Bold and Sven Urschel

Working Group for Electrotechnical Systems of Mechatronics − Kaiserslautern University of Applied Sciences,
Kaiserslautern, Germany
{sebastian.bold,sven.urschel}@hs-kl.de

Abstract. Misalignment causes heat, which leads to increased wear, stator-rotor friction, and, in the worst case, fatigue cracks. Misalignment can occur in any motor-driven assembly unless the motor is integrated into the enclosure, such as in pumps, fans, or gearboxes. In order to avoid short operation times or even unexpected down-times, companies spend a lot of effort on avoiding misalignment. But even with high precision devices such as optical alignment systems, it is not always possible to avoid the occurrence of misalignment, especially during processes resulting from heat expansion or vibration.

These problems can be overcome with a monitoring approach. Besides optical systems based on high-speed cameras, approaches based on the analysis of the motor current are also promising. By comparison, currentbased approaches are cheaper due to the simpler equipment. Approaches like motor current signature analysis (MCSA) are well described in the literature. To execute this approach, the spectra of the motor current and the fault-related frequencies are calculated. By comparing the magnitude at the specific frequency with a limit, an alarm or warning can be sent. In [1] and [2], it is shown that misalignment has an influence on the motor current. The main disadvantage of MCSA is the load dependence, which was investigated in [3]. This means that it is not possible to distinguish between load variation and an increase of misalignment.

More sophisticated classifiers such as k-nearest neighbors or artificial neural networks may solve these problems by using more sources of information. To apply such a classifier, a new feature set needs to be identified. In order to find the feature set, this work focuses on data mining, which includes two steps. Since no public data for misalignment was available, an experiment was needed to create such data. The first step in data mining is to conduct an experiment to create a database that includes variation of the misalignment and distortion such as load variation and motor size. In the second step, the data is processed by data extraction, followed by the application of a feature selection algorithm.

The experiment must be adapted to the following data mining. To extract valuable features from the data, measurement of 3-phase motor current and one line-to-line voltage was performed in 42 states. The states result from the variation of the target (misalignment) and two distortions (load and motor size). Regarding misalignment four levels were chosen from zero to 1.5 times the alarm level defined by an optical alignment system producer. Regarding load, 100%, 75%, and 50% of the rated load were selected. Regarding size, a 1.1kW and a 7.5kW motor were investigated. All magnitudes were sampled with 10kHz to assure that all natural distortions are included. The time of the measurement cycle was five times the fundamental period. The number of cycles, which led to the number of feature samples, is 2000 for each state and is needed to guarantee the stability of the feature selection as examined in [4].

For the data mining, the feature extraction is followed by a feature selection. The approaches used for data mining are described in [5]. In the feature extraction step, the raw data from the experiment were processed to create physically interpretable values. This is necessary to allow discussing the results of the data mining. Since the extraction process compresses the data, as many features from different domains as possible must be calculated to avoid the loss of important information. In practice, the signals from the experiment were used to calculate the output of additional virtual sensors, in this case the spectra, the space vector represented by its length and angle components, and the spectra of the space vector. All signals were used to calculate values such as the *root mean square*, the *maximum value*, or the *signal-to-noise ratio*. Besides generally used values, values from MCSA theory were also calculated, for example the magnitude of the frequency that correlates with air gap eccentricity (ECC). All the calculated values, based on different signals, create the feature space for the feature selection. A wrapper type approach as described in [6] was chosen for the selection. The wrapper approach uses a learning model to find information rich features, which in this case was a k-nearest neighbor classifier. To ensure generalization of the classifier, it was embedded into a 10-fold cross-validation with random selection of the samples. For the data mining, the measured states were combined into groups, which differ in terms of the target being either parallel or angular misalignment. In addition, all groups contain data with a variation in load and size. A problem with feature selection is stability, which is the sensitivity of the algorithm to perturbation in the training data. In [4], the dependence of stability is examined, including data distribution. Since it is possible to find redundant features because of the input data, the data processing was repeated several times. In the first round, the full feature space was used for data mining. In the second round, previously found features were excluded, and in the last round, only MCSA features were used. The results from all turns lead to a better understanding of the affects which misalignment has on the electrical signals and avoid confusion due to redundant feature sets.

The results show that in the first round, which used the full feature space for the search, five features are needed to reach an error rate below 0.1%. In the second round, nine features are needed to reach the threshold, and in the last round, the threshold could not be reached.

Keywords: data mining; misalignment; feature extraction; feature selection.

References

1. Verma, A.K., Sarangi, S., Kolekar, M.H.: Shaft misalignment detection using stator current monitoring
2. Popaleny, P., Antonino-Daviu, J.: Electric motors condition monitoring using currents and vibrations analyses. In: 2018 XIII International Conference on Electrical Machines (ICEM), IEEE (9 2018)
3. Obaid, R., Habetler, T.: Effect of load on detecting mechanical faults in small induction motors. In: 4th IEEE International Symposium on Diagnostics for Electric Machines, Power Electronics and Drives, 2003. SDEMPED 2003., IEEE (2003)
4. Alelyani, S., Liu, H., Wang, L.: The effect of the characteristics of the dataset on the selection stability. In: 2011 IEEE 23rd International Conference on Tools with Artificial Intelligence, IEEE
5. U.R., A., Paul, S.: Feature selection and extraction in data mining. In: 2016 Online International Conference on Green Engineering and Technologies (IC-GET), IEEE
6. Kaur, A., Guleria, K., Trivedi, N.K.: Feature selection in machine learning: Methods and comparison. In: 2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), IEEE

Silage Bale Detection for the «Cultivable Area» Update of the Cantonal Agricultural Office, Thurgau

Adrian F. Meyer[1] and Denis Jordan[1]

[1]Institute Geomatics – Fachhochschule Nordwestschweiz FHNW
adrian.meyer@fhnw.ch

Abstract. In Switzerland direct subsidies are paid to farms for sustainable agricultural practice. The cultivable agricultural area layer (German: Landwirtschaftliche Nutzfläche, LN) serves as an annual basis for the calculation of these contributions at the Swiss cantonal agricultural offices. Material deposits like silage bale stacks are usually excluded from the LN. Therefore, the canton of Thurgau could profit from a spatial vector layer indicating locations and area consumption extent of silage bale stacks intersecting with the LN perimeter.

To ease the monitoring process, we propose a Mask-RCNN based prototypical Deep Learning framework which was trained on 10cm SWISSIMAGE orthophoto datasets (swisstopo, Bern). Embedded in an efficient python-based geodata workflow the model boasts a high F1-Score of 92% on evaluation data. This approach allows robust and accurate inference detections over the whole cantonal area. Having the silage bale stack detections at hand reduces the manual workload of the responsible official by directing the eyes to the relevant hotspots.

Keywords: Agriculture; Object Detection; Monitoring; Administration; Subsidy Payments; Cadastral; Mask-RCNN; Aerial Imagery; Remote Sensing

1 Introduction

Switzerland's direct payment system is the basis for sustainable, market-oriented agriculture. The federal government supports local farms in the form of various types of subsidies such as biodiversity contributions, landscape quality contributions, or food supply security contributions.

Subsidies are often calculated by area and the agricultural offices of the respective cantonal administration are responsible for monitoring agricultural areas in order to approve the requested amounts. Only certain land usage profiles are eligible for subsidies payment. The cultivable agricultural area layer (German: Landwirtschaftliche Nutzfläche, LN) is a GIS product maintained by the cantonal agricultural offices and serves as the key calculation index for the receipt of contributions.

Major adjustments of the LN are part of the periodic update (German: Periodische Nachführung, PNF) which is carried out within the framework of the official cadastral survey (German: Amtliche Vermessung, AV) [1][2], while smaller updates are performed annually. Its correct determination is of immense importance, because if the LN vector polygons derived from the cadastral survey data deviate largely from the actual conditions on site, the monitoring effort during the annual farm structure data survey process (German: Betriebsstrukturdatenerhebung) [10][11] increases.

Farm areas that are not usable for effective productive agriculture are to be excluded from the LN. This includes material deposits such as silage hay bales storage plots which are constantly changing due to the high degree of mechanization in agriculture and can sometimes fall within the perimeter of the registered LN. The tracking of these areas with conventional surveying such as repeated field visits or the visual interpretation of current aerial imagery proves to be very time-consuming and costly. Therefore we propose an automatized workflow to predict areas currently in use for silage bale stack deposits.

Artificial convolutional neural networks (CNN) based on deep learning (DL) have been used for automated detection and classification of image features for quite some time. Reliable detection from aerial imagery using applications of DL would enable cost-effective detection of these storage areas and provide added value to agricultural office of the Canton of Thurgau (German: Landwirtschaftsamt, LWA) but also in other cantons.

In the context of the publicly financed project "Swiss Territorial Data Lab" the applicability of CNNs to generate a localized silage bale stack inventory was investigated. The delivered dataset should consist of vector polygons which are compatible with the LWA's webGIS workflow and should be made available together with new acquisitions of aerial imaging campaigns. This project therefore aims at the development of an efficient and flexible algorithm which offers a highly accurate performance and can be quickly deployed over the complete cantonal area of Thurgau (approx. 992 km²). For the LWA it is important that the detections are precise, relevant in size, and do not contain a large number of false positives.

2 Method

2.1 Overview

Silage bale stacks as target objects are clearly visible on the newest 2019 RGB layer of the 10cm SWISSIMAGE dataset [15]. A few hundred of each of these objects were manually digitized as vector polygons ("annotations") with QGIS [14] in separate workflows using a semi-automatic approach.

In order to limit computational load, the current LN extent of the canton of Thurgau was defined as Area of Interest (AoI) and tiled into smaller quadratic images (tiles). Those tiles containing an intersecting overlap with a annotation were subsequently presented to a neural object detection network for training in a process known as Transfer Learning. A random portion of the dataset was kept aside from the training process in order to allow an unbiased evaluation of the detector performance.

Multiple iterations were performed in order to find out near-optimal input parameters such as tile size, zoom level, or network- and training-specific variables termed «hyperparameters» for each of the above-mentioned target objects. All detector models were evaluated for their prediction performance on the reserved test dataset. For each target object the best model was chosen by means of its overall performance measured by maximizing the F1-Score [4] on an independent reserved evaluation dataset. This model was used in turn to perform a prediction operation («Inference») on all tiles comprising the AoI – thereby detecting the target objects over the whole canton of Thurgau.

Postprocessing included filtering the resulting polygons by a high confidence score threshold provided by the detector for each detection in order to reduce the risk of false positive results (misidentification of an object as a silage bale stack). Subsequently adjacent polygons on separate tiles were merged by standard vector operations. A spatial intersection with the known LN layer was performed to identify the specific areas occupied by the objects which should not receive contributions but potentially did in last year's rolling payout. Only intersections covering more than 50m^2 of LN area are considered «relevant» for the final delivery. For completeness, all LN-intersecting polygons of detections covering at least 20m^2 are included in the final delivery. Filtering can be undertaken easily on the end user side by sorting the features with along a precalculated area column.

2.2 Aerial Imagery

The prototypical implementation uses the publicly available SWISSIMAGE dataset of the Swiss Federal Office of Topography swisstopo [15]. It was last flown for Thurgau in spring 2019 and offers a maximum spatial resolution of 10cm Ground Sampling Distance (GSD) at 3-year intervals. As the direct subsidies are paid out yearly the periodicity of SWISSIMAGE in theory is insufficient for annual use. The challenge of low aerial image frequency remains for manual and automatic methods alike. In this case the high-quality imagery on the one hand can serve as a proof of concept though. On the other hand, the cantons have the option to order own flight campaigns or satellite data to increase the periodicity of available aerial imagery if sufficient need can be shown from several relevant administrative stakeholders.

For our approach aerial images need to be downloaded as small quadratic subsamples of the orthomosaic called tiles to be used in the DL process. The used tiling grid system follows the "Slippy Map" standard [12] with an edge length of 256 pixels and a zoom level system which is derived from a quadratic division tree on a Mercator-projected world map. The whole world equals zoom level = 0 with a GSD at equator ~156 km/px, a zoom level = 18 in this system would approximate to a GSD of ~60 cm/px.

2.3 Dataset: Silage Bale Stacks

Silage hay bales are one of several features of interest specifically excluded from the subsidized cultivable LN area. These bales are processed, and compacted fermenting grass cuttings wrapped in plastic foil. They often roughly measure 1 - 2 cubic meters in volume and are weighed in at around 900kg. They are mainly used as animal food during winter when no fresh hay is available. Farmers are required by regulation to compactly stack them in regular piles at few locations rather than scattered collections consuming large areas.

As no conducive vector dataset for silage bale locations exists in Thurgau, the annotations for this use case had to be created manually. A specific labeling strategy to obtain such a dataset was therefore implemented (see Fig. 1). Using SWISSIMAGE as a WMS bound basemap in QGIS, a few rural areas throughout the canton of Thurgau were selected and initially approximately 200 stacks of silage bales were manually digitized as polygons. Clearly disjunct stacks were digitized as two separate polygons. For partially visible stacks only visible parts were included. Loose collections of bales were connected into one common polygon if the distances between the single bales were not exceeding the diameter of a single bale. Ground imprints where silage bales were previously stored were not included. Also shadows on the ground were not part of the polygon. Plastic membrane rests were not included unless they seemed to cover additional bales. Most bales were of circular shape with an approximate diameter of 1.2 – 1.5 m, but also smaller rectangular ones were common. Colors ranged from mostly white or green tinted over still common dark green or grey to also more exotic variants such as pink, light blue and yellow (the latter three are related to a specific cancer awareness program) [18].

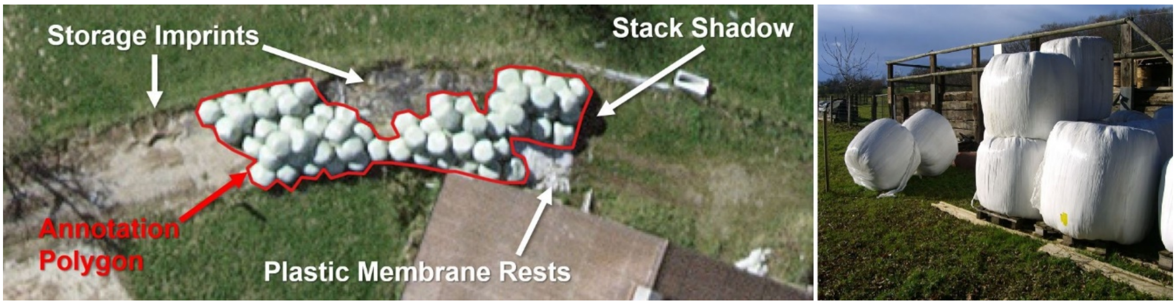


Fig. 1. Example of the annotation rules (left), example photo of Swiss silage bales (right)[16].

With these initial 200 annotations a preliminary detector was trained on a relatively high zoom level (18, 60cm GSD, tiling grid at about 150m) and predictions were generated over the whole cantonal area (See section «Training» for details). Subsequently, the 300 highest scoring new predictions (all above 99.5%) were checked visually in QGIS, precisely corrected, and then transferred into the training dataset. All tiles containing labels were checked visually again at full zoom and missing labels were created manually. The resulting annotation dataset consists of approximately 700 silage bale stacks.

2.4 Deep Learning

DL was performed with the Swiss Territorial Data Lab's Object Detection Framework [3]. The technology is based on a Mask RCNN architecture [6], an extension of Fast R-CNN [5], implemented with the High-Level API Detectron2 [17] leveraging the Deep Learning framework PyTorch [13]. Parallelization is achieved with CUDA-enabled GPUs on the High-Performance Computing cluster at the FHNW server facility in Muttenz. The Mask RCNN Backbone is formed by a 50 layer deep residual neural network (ResNet-50) [7] implementation and is accompanied by a Feature Pyramid Network (FPN) [8]. This combination of code elements results in a neural network leveraging more than 40 Mio. parameters. The model weights are obtained pretrained on the COCO dataset [9] and are modified through transfer learning. The model accepts three channel images and feature regions represented by pixel masks superimposing the imagery in the shape of the target object vector polygons.

Training is performed iteratively by presenting subsets of the tiled dataset to modify the edge weights in the network graph. Input images are not augmented as RGB aerial imagery relies on consistent northing and shadow angles. Progress is measured step-by-step through statistically minimizing the loss functions. The process is aborted if validation loss is not decreasing further after each 250 step iterations. Typically, less than 10 000 step iterations are sufficient to reach this point. Only tiles containing masks (labels) can be trained. Two smaller subsets of all labeled tiles are reserved from the training set (TRN), so a total of 70% of the trainable tiles are presented to the network for loss minimization. The validation set (VAL, 15%) and the test set (TST, 15%) are pseudo-randomly distributed and statistically independent from the TRN set. The VAL set is used to perform recurrent evaluation during training. Training can be stopped if the loss function on the validation set has reached a minimum since after that point further training would push the model into an overfitting scenario. The TST set serves as an unbiased reserve to evaluate the detector performance on previously unseen data. Tiles not containing a label yet were classified into a separate class called "other" (OTH, see Fig. 2). This dataset was only used for generating predictions (inference).

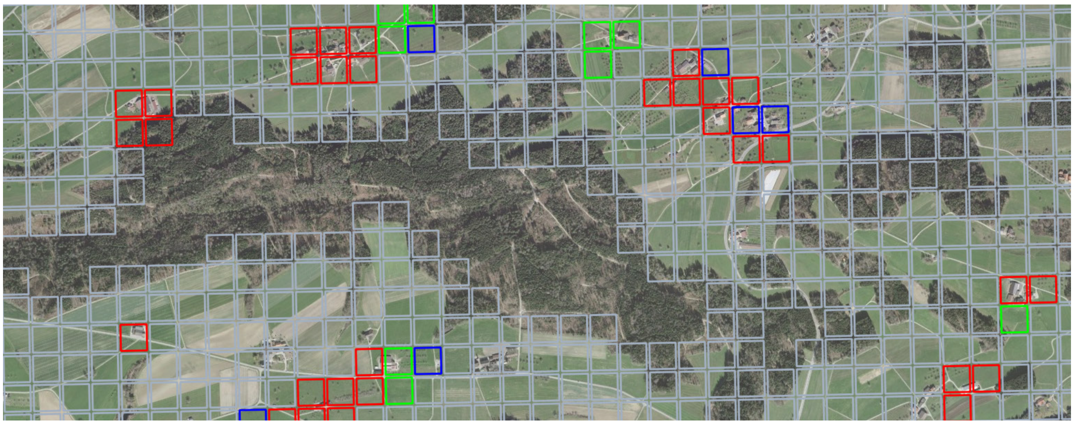


Fig. 2. Dataset Split – Grey tiles are only used in prediction (OTH); they do not contain any labels during training. The colourful tiles contain labels, but are scattered relatively sparsely. Red tiles are used for training the model weights (TRN); green tiles validate the learning progress during training to avoid overfitting (VAL) and blue tiles are reserved for unbiased post-training evaluation (TST).

Multiple training runs were performed separately to manually optimize the network-specific hyperparameters such as batch size, learning rate or momentum. Also, multiple zoom levels (spatial resolution, quadratic subdivision of tiles, see Fig. 2) were tested as a hyper-parameter variable in this manner. Learning rate was scheduled over the iterations using the "WarmupMultiStepLR" system.

2.5 Prediction and Assessment

For the TRN, VAL and TST subset, confusion matrix counts, and classification metrics calculations can be performed since they offer a comparison with the digitized «ground truth» reference. For all subsets (including the rest of the cantonal LN as OTH), predictions are generated as vectors covering those areas of a tile that the detector algorithm identifies as target objects and therefore a confidence score is attributed. In case of the tiles containing annotation polygons, the overlap between the predictions and the labels can be checked. Is any overlap found between a label and a prediction this detection is considered a true positive (TP). If the detector missed a label entirely this label can be considered as false negative (FN). Did the detector predict a target object that was not present in the labelled data it is considered false positive (FP). On the unlabeled OTH tiles, inference predictions cannot be checked against reference data.

The counting of TPs, FPs and FNs on the TST subset allows the calculation of standard metrics such as precision (user accuracy), recall (producer accuracy) and F1 score (as a common overall performance metric calculated as the harmonic mean of precision and recall) [4]. The counts, as well as the metrics can be plotted as function of the minimum confidence score threshold which can be set to an acceptable filter percentage for a certain detection task. A low threshold should generally yield fewer FN errors, while a high threshold should yield fewer FP detections. The best performing model by means of maximum F1 score was used to perform a prediction run over all tiles intersecting with the cantonal LN surface area.

2.6 Post-Processing

In order to obtain a consistent result dataset, detections need to be postprocessed. Firstly, the confidence score threshold operation is applied. Here, a comparatively high threshold can be used for this operation. «Missing» a detection of a target object (FN) is not as costly for the analysis of the resulting dataset at the agricultural office as analyzing large numbers of FP detections would be. Also missing single individual small target objects is much less problematic than missing whole large areas. These larger areas are typically attributed with higher confidence scores though and are therefore less likely to be missed.

In some cases, silage bale stacks can cross the tiling grid and are therefore detected on multiple images. This results in edge artefacts along the tile boundaries intersecting detections that should be unified. For this reason, adjacent polygons need to be merged into a single polygon. This is achieved by first buffering all detections with a 1.5m radius (roughly the radius of a single typical bale). Then all touching polygons are dissolved into single features. Afterwards, negative buffering with -1.5m radius is applied to restore the original boundary (see Fig. 3).

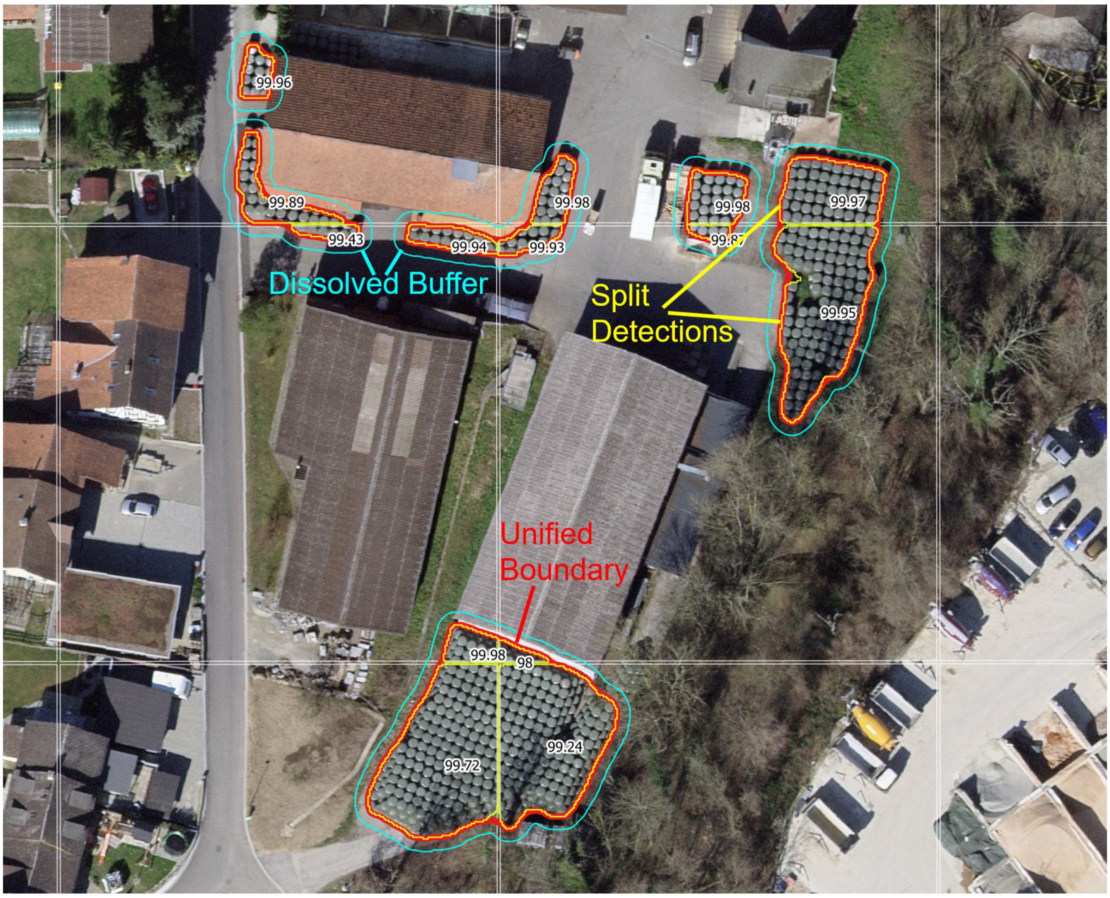


Fig. 3: Example of silage bale detection polygons (red) from raw detections (yellow) dissolved because they are crossing the tile boundary (light blue).

This process also leads to an edge smoothing of the pixel step derived vector boundary into curves containing a high number of vertices. A simplification operation reducing the number of vertices can be performed without the loss of relevant spatial accuracy. For all remaining detection polygons, the confidence score is reattributed as a merged area-weighted average of the input values. With a threshold operation on the resulting area all target objects with an area cover below 20 m^2 are filtered out of the dataset to provide only economically relevant detections.

3 Results

Silage Bale Stacks as a target object generally resulted in successful and robust models achieving high prediction performance. The detections were considered deliverable to the LWA.

Tab 1: Performance of silage bale detector models at several zoom levels evaluated by maximum F1-Score.

	Zoom Level 16	Zoom Level 17	Zoom Level 18	Zoom Level 19	Zoom Level 20
GSD	~ 240 cm/px	~ 120 cm/px	~ 60 cm/px	~ 30 cm/px	~ 15 cm/px
# Tiles Trained	~ 600	~ 1 000	~ 1 600	~ 3 000	~ 5 000
# Tiles Inference	~ 8 000	~ 25 000	~ 84 000	~ 310 000	~ 1 310 000
Duration of Run	~ 0.6 h	~ 2 h	~ 4 h	~ 15 h	~ 100 h
TST Max F1 RGB	52.5 %	74.7 %	87.2 %	92.3 %	90.9 %

The model trained with tiles at zoom level 19 (every pixel approx. 30cm GSD) showed the highest performance with a maximum F1 Score of 92.3% (see Tab. 1). Increasing the resolution even further by using 15 cm/px GSD did not result in a gain in overall detection performance while drastically increasing storage needs and computational load. The detector model at zoom level 19 is performing very well on the independent TST dataset detecting the largest portion of silage bale stacks at any given confidence threshold. The number of FP reaches very low counts towards the higher end of the threshold percentage, increasing precision while decreasing recall (see Fig. 4).

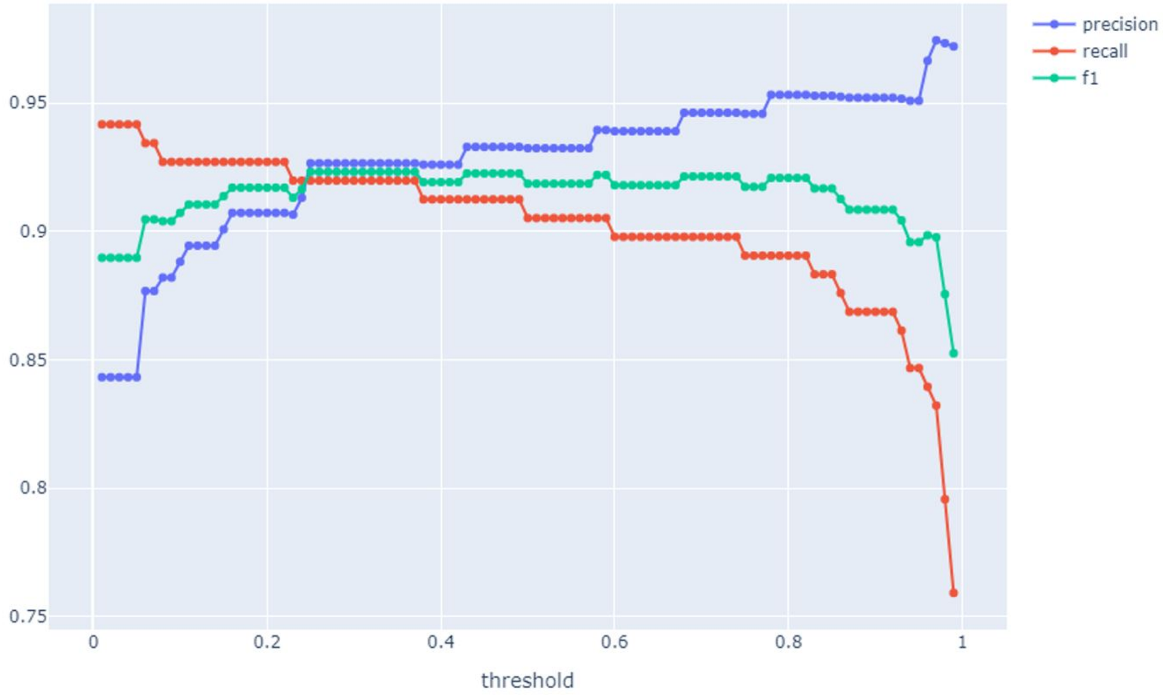


Fig. 4: Performance metrics of the Zoom Level 19 model on the TST dataset as a function of the minimum confidence score threshold.

For delivery of the dataset a detector was subsequently used at a threshold of 96% minimizing FP errors resulting in a conscious bias on precision, see Fig. 4 and 5. At this value 809 silage bale stacks were rediscovered in the TRN, TST and VAL subset. Just 10 FP detections were found in these subsets. 97 silage bale stacks were not rediscovered (FN). The model precision (user accuracy) on the TST set was found to reach approx. 98% and the recall (hit rate, producer accuracy) was acceptable at approx. 85%.

In the applied inference run the model detected a total of 2 473 additional silage bale stacks after post-processing over the rest of the LN area of the canton of Thurgau (OTH subset), of which 288 stacks cover more than 20 m^2 and were prepared for delivery. The relevant total intersection area of the final vector polygon dataset with the LN layer amounts to approx. 8 000 m^2.

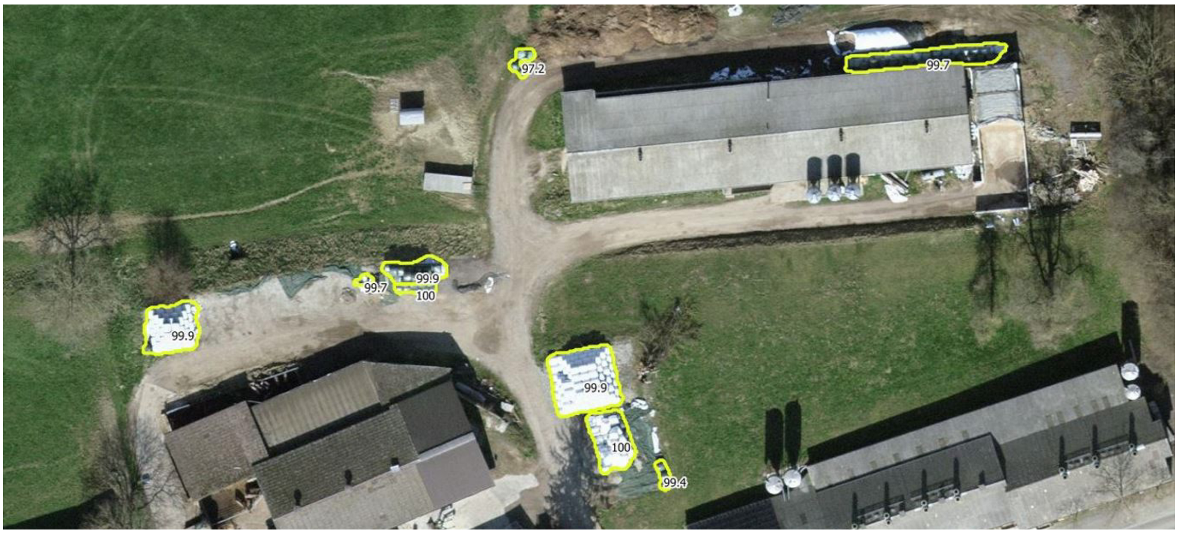


Fig. 5: Raw inference detections (yellow) of silage bale stacks displaying very high confidence scores outside of the TRN/VAL/TST subsets.

4 Conclusion

The agricultural office describes the detections of silage bales as very accurate with only a small percentage of actual FP detections. Clearly delineated objects such as silage bales are generally less demanding to detect than more complex target objects. The high F1 score surpassing 90% suggests a productively usable result. Especially larger stacks are detected with very high confidence scores and can be targeted first by area-filtering in the monitoring process. The Mask-RCNN approach proved to be a viable deep learning kernel architecture.

The highest zoom level 20 (15cm GSD) requires enormous computational resources especially for the prediction run while performing suboptimal on evaluation metrics. Hence, RGB winter imagery resampled from SWISSIMAGE at a resolution of 30cm GSD proved to be sufficient in resolution and quality while still maintaining a reasonable effort on computational resources for inference runs over the complete cantonal agricultural surface.

Very few false positive samples such as animal shelters, material deposits or white-colored vehicles remained in the final prediction dataset. Options to automatically tackle this challenge in the future include new models distinguishing multiple classes, the choice of larger (higher parametric) model architectures, larger training datasets or a revised and improved post-processing workflow.

On an economical scale the extra effort for the LWA resulting from misplaced silage bale stacks in the LN areas is not negligible but also not extremely critical. In the scope of this study, silage bale stacks did serve as an accessible initial proof of concept regarding the usability of the detector. The new detections allow the professionals at the agricultural office to direct their eyes more quickly at relevant hotspots and spare them some aspects of the long and tedious manual search on aerial imagery that was performed in the past.

For the future, extending the range of target objects to larger and more complex areas such as complete farm yards or land usage patterns such as grazed pastures on steep slopes could provide strong additional benefits for the monitoring process at the agricultural office.

5 Acknowledgments

We want to thank the cantonal agricultural office of Thurgau and the team at the Swiss Territorial Data Lab for the detailed review of our results. Especially we would like to show our immense gratitude to Alessandro Cerioni at the Système d'Information du Territoire à Genève (SITG) who contributed large efficient and powerful sections of code to the object detection framework used in this study. Furthermore, we are very grateful to Pascal Salathé (FHNW) for his input on the Swiss direct subsidy system and to Natalie Lack (FHNW) for her internal review of the article.

References

1. Amt für Geoinformation des Kanton Thurgau: Handbuch Amtliche Vermessung. Kanton Thurgau (2022) 25-26, 158-171, 185-188
2. Bundesamt für Landwirtschaft BLW: Direktzahlungen an Schweizer Ganzjahresbetriebe. BLW, Bern (2021)
3. Cerioni, A. & Meyer, A.: Object Detection Framework. Swiss Territorial Data Lab (2021)
4. Chinchor, N.: MUC-4 Evaluation Metrics, Proceedings of the Fourth Message Understanding Conference (1992) 22-29
5. Girshick, R.: Fast r-cnn. Proceedings of the IEEE international conference on computer vision (2015) 1440-1448
6. He, K., Gkioxari, G., Dollár, P., & Girshick, R.: Mask r-cnn. Proceedings of the IEEE international conference on computer vision (2017) 2961-2969
7. He, K., Zhang, X., Ren, S., Sun, J.: Deep Residual Learning for Image Recognition. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas (2016) 770-778
8. Lin, T., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature Pyramid Networks for Object Detection. Computer Vision and Pattern Recognition (2016)
9. Lin, T., Maire, M., Belongie, S., Bourdev, L., Girshick, R., Hays, J., Perona, P., Ramanan, D., Zitnick. C.L., Dollár, P.: Microsoft COCO: Common Objects in Context, Computer Vision and Pattern Recognition (2014)
10. Meier, T. & Menzel, S.: Bundesamt für Landwirtschaft BLW, Agrarbericht 2020 – Politik, Einführung (2020) 2-4
11. Meyer, D.: Bundesamt für Landwirtschaft BLW, Agrarbericht 2020 – Politik, Direktzahlungen (2020) 11-12
12. OpenStreetMap Foundation: Slippy Map. https://wiki.openstreetmap.org/wiki/Slippy_Map (2021)
13. Paszke, A., Gross, S., Chintala, S., Chanan, G.: PyTorch. Facebook AI Research / Meta AI (2016)
14. QGIS Association: QGIS Geographic Information System. https://qgis.org/en/site/ (2021)
15. Swisstopo: «SWISSIMAGE 10 cm», The Digital Color Orthophotomosaic of Switzerland. Federal Office of Topography swisstopo (2021)
16. Tschudin, P.: Photo Silageballen; https://www.flickr.com/photos/35637563@N00/80239710/ (2006)
17. Wu, Y., Kirillov, A., Massa, F., Lo, W. Y., & Girshick, R.: Detectron2. Facebook AI Research / Meta AI (2019)
18. Zindel, N.: Siloballen-Aktion für den guten Zweck. Pink Ribbon Schweiz (2018)

Value-Sensitive Design for AI Technologies: Proposition of Basic Research Principles Based on Social Robotics Research

Theresa Schmiedel, Vivienne Jia Zhong and Janine Jäger

FHNW University of Applied Sciences and Arts Northwestern Switzerland
theresa.schmiedel@fhnw.ch
viviennejia.zhong@fhnw.ch
janine.jaeger@fhnw.ch

Abstract. Artificial intelligence (AI) technologies, such as social robots, increasingly influence economic and social life, with both opportunities and risks. The added value of such technologies depends enormously on how they are designed. In this conceptual paper, we report on our research experience and call for developing AI technologies including human values relevant to the particular stakeholder groups of such technologies as suggested by value-sensitive design (VSD) – an established design approach. To facilitate the application of VSD for designing AI technologies, we propose three basic research principles for VSD and call for more research particularly covering the design of AI technologies, such as social robots and beyond.

Keywords: Value-sensitive Design, Artificial Intelligence, Human Values, AI Technologies, Social Robots.

1 The importance to design AI technologies considering human values

Artificial Intelligence (AI) technologies are increasingly part of all areas of social and economic life (e.g., smart home applications, self-driving cars). AI systems can mimic or even surpass human intelligence and decision-making. They are able to learn independently, interact with their environment, and make decisions. Thus, with increasing autonomy, AI becomes an independent actor in social and economic life. While AI bears innovation potential with high societal benefits in various areas, AI also comes along with unforeseeable risks and challenges for economy and society.

Social robots are a good example to demonstrate the benefits and challenges of AI. Social robots are physically embodied robots designed to engage in social interactions with humans in a socially acceptable way [1]. They engage with humans using various interaction modalities, for example, conversing with users in natural language, recognizing user's face and emotion and reacting accordingly. To enable such humanlike, natural interactions, a wide range of AI technologies (e.g., spoken dialog system, face recognition, motion planning, etc.) are employed and aligned with each other. Social robots are used for various contexts (e.g., in education [2], healthcare [3–5], tourism [6–9]). Across the applications, numerous benefits are reported. Research reports on several positive outcomes (e.g., improvement of social behaviors in children) involving social robots in autism therapy [10] or diverse benefits from using social robots for mental health (e.g., improved mood and loneliness reduction) . Despite these benefits, researchers also raise numerous concerns with regard to the deployment of social robots. For example, researchers have identified several ethical and societal issues related to the use of social robots for therapeutic applications in mental health services (e.g., loss of patient autonomy, lack of guidance on development, emotional dependency on robots) and developed corresponding recommendations [12]. In the same vein, findings from a series of transdisciplinary workshops involving social robot experts indicate further concerns from ethical, social and legal aspects (e.g., replacement of human interaction) [13].

As social robots and other AI technologies have the potential to influence human relationships [12, 14] and shape social practices [15], we need to mitigate potential negative impacts connected to the use of AI. Thus, we need to design AI with human values in mind that reflect what is important to the relevant user groups of each AI technology in its particular use context [16]. Only by employing a value-sensitive approach to AI, the use of AI is sustained in the society.

2 Value-sensitive design

Value-Sensitive Design (VSD) is an established approach that incorporates various user-centered design methods and aims to design technologies in such a way that human values are reflected in the technology design and its applications [17, 18]. Values are subconscious needs, that members of a group share (e.g., autonomy, security, or collaboration) [19], and which guide their behavior (e.g., to use or avoid certain technologies) [20]. The advantage of VSD is that technology adapts to human needs and not the other way around. In other words, humans shape the technology in a sustainable and proactive way and are not passively influenced or controlled by the technology [21–23]. VSD is thus a proactive approach that translates human values into technological requirements and thus influences the design of technologies early in the development process to incorporate stakeholder-relevant requirements in a fundamental and timely manner [21]. Particularly, VSD consists of a tripartite methodology of empirical, conceptual, and technical investigations. In the conceptual investigation, stakeholders are identified. Moreover, human values from literature and those important to relevant stakeholder groups are identified and conceptualized through analytic, theoretical and philosophical investigation. In the empirical investigation, methods from social science research are used to examine how stakeholders experience values in the particular sociotechnical context. Specially, competing values are prioritized and value conflicts are solved. Design requirements are then formulated. In the technical investigation, one can employ retrospective analyses to evaluate the compliance of an existing technology to the identified values in the conceptual investigation, or design the technology according to the identified values in a proactive way [21, 24]. It is important to stress that the application of the tripartite methodology does not follow a specific order. Researchers can start with any of the three types of investigation. Indeed, it is recommended to apply the three types of investigation in an iterative and integrative manner [21].

A wide range of technologies have benefited from VSD (e.g., [25–27]). In the field of social robotics, several works have advanced the design of value-sensitive human-robot interaction. In her works, van Wynsberghe [28, 29] focuses the application of VSD on care robotics — research on robots that are used related to the care of persons — and develops the "care centered value sensitive design methodology" as well as the "care centered framework" that allow one to design and evaluate care robots from an ethical perspective. Building upon on her works, Umbrello and his colleagues [24, 30] proposed a multi-tiered approach VSD-AI for Social Good (AI4SG) for care robotics that extend the scope of values to be considered (e.g., including UN Sustainable Development Goals as value sources). Beyond the healthcare context, VSD has been applied to social robots in education to uncover parents' moral values with respect to the introduction of social robots in primary schools [31]. Furthermore, various works have been conducted in the field of child-robot interaction to gain an understanding of children's perception of and attribution of moral and social standing to social robots . Finally, using VSD as a research background, Zhong et al. [32] investigated factors that affect robot likeability.

The existing literature shows that VSD research traditionally has a strong background in ethics. However, recent discussion calls on broadening the view of values by giving more attention to the context, which affects stakeholders' view on values [30, 33]. In particular, the context plays a vital role in how stakeholders perceive, interpret and prioritize values relevant to the design of technology [33]. Complementary to the ethical perspective, we take a managerial and contextual viewpoint in our application of VSD.

3 Proposition of three basic research principles for VSD

Based on our research experience on VSD in social robotics, we propose three basic research principles, which we outline next.

3.1 Employ a multi-iterative VSD approach

The first principle concerns the iterative characteristic of VSD, which essentially ensures a technology design that respects stakeholders' values. With regard to VSD as an iterative approach of three types of investigation (conceptual, empirical, technical), a recent review reveals a lack of iteration in most VSD projects and calls for considering this aspect to enhance technology design [34]. While existing research (e.g., [24]) suggests that each design process once contains the three types of investigation of VSD, we argue that the iterative nature of VSD calls for a more frequent iteration of the three investigation types within a design process. Centering values in the core of the technology design, we propose a multi-iterative VSD approach. This multi-iterative approach should follow the identification of stakeholders and their benefits and harms in the particular sociotechnical context. Based on our research on human-robot interactions, we distinguish three main phases of VSD for specifying the multi-iterative approach: 1) value identification, 2) value embedding, and 3) value evaluation. Fig 1 displays the multi-iterative VSD approach we propose as the first basic research principle of applying VSD for designing AI technologies, and particularly social robots.

Value identification. This phase focuses on the identification, conceptualization, and prioritization of values relevant to the stakeholders not only in general but especially in the particular use case context of the particular technology. All three types of investigations can be part of this phase. For instance, the identification of values in HRI can be done *conceptually* by drawing on ethical frameworks identified in the literature [28], but also by conducting *empirical* investigations with stakeholders for the stake of contextual values. Furthermore, *technical* artifacts such as mockups can be used for the elicitation of values as well. Particularly, we learned that mockups are useful for identifying values in HRI as most people do not have any experience in interacting with social robots yet [32]. The outcomes of the empirical and technical investigation inform the conceptual investigation, for instance, with regards to the conceptualization of values. Another important aspect of this phases is addressing potential tensions between values, also through conceptual, empirical, and potentially technical investigations.

Value embedding. In this phase, values are embedded into technology design through translating the values into design requirements and implementing them into the design. Deriving design requirements can be *conceptually* guided by existing frameworks (e.g., [35]), complementing *empirical* investigations involving stakeholders. The operationalization of the design requirements can then draw from theoretical *concepts* (e.g., design patterns [40] for specifying values-sensitive HRI), followed by the *technical* realization in form of prototypes. Before moving on to the next phase of value evaluation, the developed prototype should be tested not only with regard to functional aspects but also with regard to usability aspects. The latter is important to ensure a potential misperception of values in the evaluation phase is not due to a lack of usability. We suggest that the technical prototype shall ensure good usability before conducting the value evaluation. This requires an iterative cycle of *technical* prototyping and *empirical* usability testing. Furthermore, from past projects, we learned that during prototyping, there might be unforeseen technical limitations that require adaptions of the operationalization of design requirements, which requires further iterations within this phase.

Value evaluation. This phase focuses on the *empirical* assessment whether the values embedded into the developed prototype are actually perceived by stakeholders as intended. The results of the evaluation can either confirm the design or indicate two types of required improvement. The first type concerns the insufficient technical specification of the developed prototype (e.g., misguiding robot behaviors), which consequently needs a revision of design requirements. In this case, researchers shall revisit the second phase, value embedding. The second type concerns unanticipated values, which stakeholders become aware of to be relevant, and value tensions, which usually become apparent after the deployment of the prototype [36]. The evaluation ideally spans a period of several weeks to be able to uncover this type of required improvement. If such improvements are necessary, researchers are advised to revisit the first phase, value identification. To avoid serious deviations from stakeholders' expectations over time that lead to a major redesign of the technology, we propose to revisit the three phases frequently.

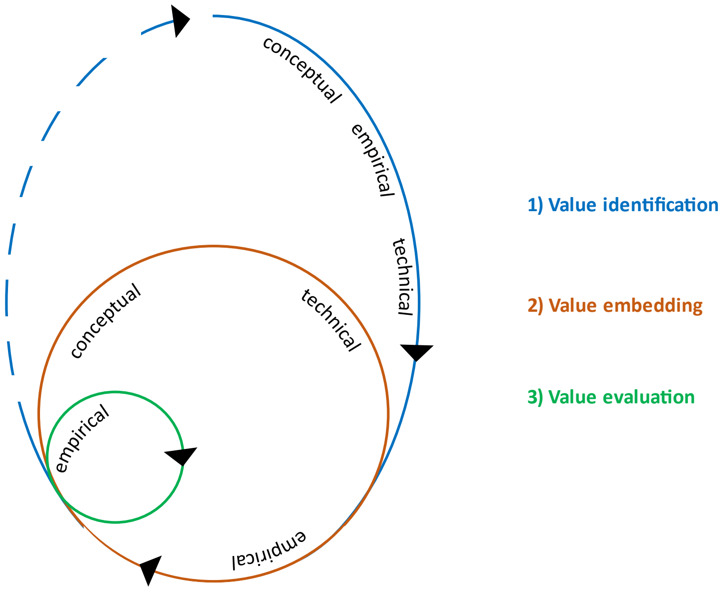


Fig. 1. Specification of the multi-iterative VSD approach

3.2 Start with the use case

The second principle addresses the beginning of a VSD project. Traditionally, VSD does not prescribe how to start a VSD project. Instead, Friedman et al. [36] suggest beginning with the most relevant aspect to the researchers. Therefore, over the past years, we started our VSD projects with the context of use and the technology — social robots. For example, we investigated the use of social robots in public spaces such as a campus library [38], a reception area of a company [37], and the application of social robots in the healthcare sector. We learned that it is highly beneficial to focus on specific social robot use cases to identify values that are relevant to the various stakeholders of a social robot such as in the healthcare context. Even though it is possible to investigate which generic values are important to the application of social robots in the healthcare field, we found that such values are relevant but insufficient for specifying design requirements that foster acceptance and usage of a social robot. For example, in a survey (N = 127) investigating values important in the healthcare context, we found that some universal values such as power and universalism as identified by Schwatz [39] have no importance as opposed to context-specific values such as compassion and respect for person as specified in prior research [40]. Furthermore, participants prioritized the values in a use case-dependent way. Using a robot as a companion in the case of a longer hospital stay, the top three values are compassion, respect for person, hedonism, while the values of security, benevolence and self-direction share the fourth rank. Using a robot as a distraction during needle injection, the top five values are compassion, respect for person, security, hedonism, benevolence and conformity. Our explanation is that the technology can be used for a very diverse set of applications. For example, social robots may serve as assistants in the registration process of patients or they may support staff through providing medical information to patients. In each particular use case, the set of values relevant to the specific stakeholders of a social robot can differ largely. In the example, efficiency may be one of the dominant values relevant in the registration process of patients, while empathy may be especially important when giving medical information to the patients. Thus, we suggest establishing the research principle of a use case-guided specification of stakeholder-relevant values for a particular technology.

3.3 Sustain the knowledge of value-sensitive design

The third principle focuses on perpetuating the findings of VSD projects, so both researchers and engineers can build upon the already generated design knowledge. Some researchers have already generated an overview of VSD projects, such as Friedman and Hendry [21] who categorized VSD projects by application domain, values, and technology. To foster the exchange of the VSD community, in particular, with regard to design practices and knowledge, we suggest to go beyond such categorizations and extend the overview into a technology-specific catalogue of use cases, stakeholder-relevant values, and particularly also the related design requirements, and technical specifications. Such a catalogue can then develop into a reference frame for researchers and engineers for 1) leveraging current best practices and 2) further developing value-sensitive AI technology. Thus, we propose developing a reference frame that shall serve as a basis for technology design, especially shaping the interaction of humans and technology, such as through specifying typical design patterns.

4 Conclusion

Various AI technologies are currently in their infancies and, thus, naturally a predominant focus lies on their technical performance and development. In this context, the end users and affected social stakeholder groups are not necessarily sufficiently considered. Yet, to develop AI technologies in a responsible and sustainable manner requires integrating what these stakeholder groups deem important. While user-centered design approaches focus on integrating the perspectives of the end user, VSD goes beyond in that it involves all stakeholders of the technology and focuses on a value perspective rather than a user requirements perspective.

To ease the design of AI technologies, we propose three basic research principles of VSD. The first research principle employs the multi-iterative VSD approach that consists of 1) value identification, 2) value embedding, and 3) value evaluation. The second research principle starts with the use case to understand and identify contextual values relevant to the stakeholders. The third research principle sustains knowledge of value-sensitive design. The proposed research principles are our first attempt to generalize our learnings from the past VSD projects. Currently, we are refining and further developing the proposed principles, in particular, using social robots as an exemplary technology.

We intend to create awareness for the relevance of values in AI design. Our overall goal is to trigger interdisciplinary exchange and collaboration on the important topic of designing AI with a focus on stakeholder relevancy and societal wellbeing.

References

1. Schmiedel, T., Jäger, J., Zhong, V.J.: Social Robots in Organizational Contexts: The Role of Culture and Future Research Needs. In: Dornberger, R. (ed.) New Trends in Business Information Systems and Technology: Digital Innovation and Digital Business Transformation. pp. 163–177. Springer International Publishing, Cham (2021). https://doi.org/10.1007/978-3-030-48332-6_11.
2. Belpaeme, T., Kennedy, J., Ramachandran, A., Scassellati, B., Tanaka, F.: Social robots for education: A review. Science Robotics. 3, eaat5954 (2018). https://doi.org/10.1126/scirobotics.aat5954.
3. Hung, L., Liu, C., Woldum, E., Au-Yeung, A., Berndt, A., Wallsworth, C., Horne, N., Gregorio, M., Mann, J., Chaudhury, H.: The benefits of and barriers to using a social robot PARO in care settings: a scoping review. BMC Geriatr. 19, 232 (2019). https://doi.org/10.1186/s12877-019-1244-6.
4. Arent, K., Kruk-Lasocka, J., Niemiec, T., Szczepanowski, R.: Social robot in diagnosis of autism among preschool children. In: 2019 24th International Conference on Methods and Models in Automation and Robotics, MMAR 2019. pp. 652–656 (2019). https://doi.org/10.1109/MMAR.2019.8864666.
5. Riek, L.D.: Healthcare Robotics. Communications of the ACM. 60, 68–78 (2017).
6. Murphy, J., Hofacker, C., Gretzel, U.: Dawning of the age of robots in hospitality and tourism: Challenges for teaching and research. European Journal of Tourism Research. 15, 104–111 (2017).
7. Ivanov, S.H., Webster, C., Berezina, K.: Adoption of Robots and Service Automation by Tourism and Hospitality Companies. Social Science Research Network, Rochester, NY (2017).
8. Chung, M.J.-Y., Cakmak, M.: "How was Your Stay?": Exploring the Use of Robots for Gathering Customer Feedback in the Hospitality Industry. In: 2018 27th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN). pp. 947–954. IEEE, Nanjing (2018). https://doi.org/10.1109/ROMAN.2018.8525604.
9. Germak, C., Lupetti, M.L., Giuliano, L., Ng, M.E.K.: Robots and Cultural Heritage: New Museum Experiences. Journal of Science and Technology of the Arts. 7, 47–57 (2015).
10. Pennisi, P., Tonacci, A., Tartarisco, G., Billeci, L., Ruta, L., Gangemi, S., Pioggia, G.: Autism and social robotics: A systematic review. Autism Res. 9, 165–183 (2016). https://doi.org/10.1002/aur.1527.
11. Rabbitt, S.M., Kazdin, A.E., Scassellati, B.: Integrating socially assistive robotics into mental healthcare interventions: Applications and recommendations for expanded use. Clinical Psychology Review. 35, 35–46 (2015). https://doi.org/10.1016/j.cpr.2014.07.001.
12. Fiske, A., Henningsen, P., Buyx, A.: Your Robot Therapist Will See You Now: Ethical Implications of Embodied Artificial Intelligence in Psychiatry, Psychology, and Psychotherapy. Journal of Medical Internet Research. 21, e13216 (2019). https://doi.org/10.2196/13216.
13. Fosch-Villaronga, E., Lutz, C., Tamò-Larrieux, A.: Gathering Expert Opinions for Social Robots' Ethical, Legal, and Societal Concerns: Findings from Four International Workshops. Int J of Soc Robotics. 12, 441–458 (2020). https://doi.org/10.1007/s12369-019-00605-z.
14. Volpe, G., Schulte-Althoff, M., Dillmann, D., Maurer, E., Niedenzu, Y., Schließer, P., Fürstenau, D.: Humanoid Social Robots and the Reconfiguration of Customer Service. In: Bandi, R.K., C. R., R., Klein, S., Madon, S., and Monteiro, E. (eds.) The Future of Digital Work: The Challenge of Inequality. pp. 310–325. Springer International Publishing, Cham (2020). https://doi.org/10.1007/978-3-030-64697-4_23.
15. Samani, H., Saadatian, E., Pang, N., Polydorou, D., Fernando, O.N.N., Nakatsu, R., Koh, J.T.K.V.: Cultural Robotics: The Culture of Robotics and Robotics in Culture Regular Paper. International Journal of Advanced Robotic Systems. 10, 400 (2013). https://doi.org/10.5772/57260.
16. Umbrello, S., De Bellis, A.F.: A Value-Sensitive Design Approach to Intelligent Agents. In: Yampolskiy, R. (ed.) Artificial Intelligence Safety and Security (2018). CRC Press, Rochester, NY (2018).
17. Friedman, B.: Value-sensitive design. interactions. 3, 16–23 (1996). https://doi.org/10.1145/242485.242493.
18. Friedman, B., Kahn, P.H., Borning, A., Huldtgren, A.: Value Sensitive Design and Information Systems. In: Doorn, N., Schuurbiers, D., van de Poel, I., and Gorman, M.E. (eds.) Early Engagement and New Technologies: Opening up the Laboratory. pp. 55–95. Springer Netherlands, Dordrecht (2013). https://doi.org/10.1007/978-94-007-7844-3_4.
19. Ros, M., Schwartz, S.H., Shoshana, S.: Basic Individual Values, Work Values, and the Meaning of Work. Applied psychology. 48, 49–71 (1999).
20. Maio, G.R.: The Psychology of Human Values. Routledge, London (2016).
21. Friedman, B., Hendry, D.G.: Value Sensitive Design: Shaping Technology with Moral Imagination. MIT Press, Cambridge, MA (2019).

22. Orlikowski, W.J.: The Duality of Technology: Rethinking the Concept of Technology in Organizations. Organization Science. 3, 398–427 (1992).
23. Šabanović, S.: Robots in society, society in robots: Mutual shaping of society and technology as a framework for social robot design. International Journal of Social Robotics. 2, 439–450 (2010). https://doi.org/10.1007/s12369-010-0066-7.
24. Umbrello, S., van de Poel, I.: Mapping value sensitive design onto AI for social good principles. AI Ethics. 1, 283–296 (2021). https://doi.org/10.1007/s43681-021-00038-3.
25. Maathuis, I., Niezen, M., Buitenweg, D., Bongers, I.L., van Nieuwenhuizen, C.: Exploring Human Values in the Design of a Web-Based QoL-Instrument for People with Mental Health Problems: A Value Sensitive Design Approach. Sci Eng Ethics. 26, 871–898 (2020). https://doi.org/10.1007/s11948-019-00142-y.
26. Thornton, S.M., Lewis, F.E., Zhang, V., Kochenderfer, M.J., Christian Gerdes, J.: Value Sensitive Design for Autonomous Vehicle Motion Planning. In: 2018 IEEE Intelligent Vehicles Symposium (IV). pp. 1157–1162 (2018). https://doi.org/10.1109/IVS.2018.8500441.
27. Gazzaneo, L., Padovano, A., Umbrello, S.: Designing Smart Operator 4.0 for Human Values: A Value Sensitive Design Approach. Procedia Manufacturing. 42, 219–226 (2020). https://doi.org/10.1016/j.promfg.2020.02.073.
28. van Wynsberghe, A.: Designing Robots for Care: Care Centered Value-Sensitive Design. Science and Engineering Ethics. 19, 407–433 (2013). https://doi.org/10.1007/s11948-011-9343-6.
29. van Wynsberghe, A.: Service Robots, Care Ethics, and Design. Ethics and Information Technology. 18, 311–321 (2016). https://doi.org/10.1007/s10676-016-9409-x.
30. Umbrello, S., Capasso, M., Balistreri, M., Pirni, A., Merenda, F.: Value Sensitive Design to Achieve the UN SDGs with AI: A Case of Elderly Care Robots. Minds & Machines. (2021). https://doi.org/10.1007/s11023-021-09561-y.
31. Smakman, M., Jansen, B., Leunen, J., Konijn, E.: Acceptable social robots in education: A value sensitive parent perspective. In: INTED2020 Proceedings. pp. 7946–7953. International Academy of Technology, Education and Development (IATED), Valencia, Spain (2020). https://doi.org/10.21125/inted.2020.2161.
32. Zhong, V.J., Mürset, N., Jäger, J., Schmiedel, T.: Exploring Variables That Affect Robot Likeability. In: Proceedings of the 2022 ACM/IEEE International Conference on Human-Robot Interaction. pp. 1140–1145. IEEE Press, Sapporo, Hokkaido, Japan (2022).
33. Saket, M.: Putting Values in Context: an augmentation of Value Sensitive Design (VSD). J. Eth. Emerg. Tech. 31, 1–9 (2021). https://doi.org/10.55613/jeet.v31i2.86.
34. Winkler, T., Spiekermann, S.: Twenty Years of Value Sensitive Design: A Review of Methodological Practices in VSD Projects. Ethics and Information Technology. (2018). https://doi.org/10.1007/s10676-018-9476-2.
35. van de Poel, I.: Translating Values into Design Requirements. In: Michelfelder, D.P., McCarthy, N., and Goldberg, D.E. (eds.) Philosophy and Engineering: Reflections on Practice, Principles and Process. pp. 253–266. Springer Netherlands, Dordrecht (2013). https://doi.org/10.1007/978-94-007-7762-0_20.
36. Friedman, B., Hendry, D.G., Borning, A.: A Survey of Value Sensitive Design Methods. HCI. 11, 63–125 (2017). https://doi.org/10.1561/1100000015.
37. Zhong, V.J., Schmiedel, T.: A User-Centered Agile Approach to the Development of a Real-World Social Robot Application for Reception Areas. In: Companion of the 2021 ACM/IEEE International Conference on Human-Robot Interaction. pp. 76–80. Association for Computing Machinery, Boulder, CO, USA (2021). https://doi.org/10.1145/3434074.3447132.
38. Sabbioni, G., Zhong, V.J., Jäger, J., Schmiedel, T.: May I Show You the Route? Developing a Service Robot Application in a Library Using Design Science Research. In: Ahram, T. and Taiar, R. (eds.) Human Interaction, Emerging Technologies and Future Systems V. pp. 306–313. Springer International Publishing, Cham (2021). https://doi.org/10.1007/978-3-030-85540-6_39.
39. Schwartz, S.H.: Are There Universal Aspects in the Structure and Contents of Human Values? Journal of Social Issues. 50, 19–45 (1994). https://doi.org/10.1111/j.1540-4560.1994.tb01196.x.
40. Rider, E.A., Kurtz, S., Slade, D., Longmaid, H.E., Ho, M.-J., Pun, J.K., Eggins, S., Branch, W.T.: The International Charter for Human Values in Healthcare: An interprofessional global collaboration to enhance values and communication in healthcare. Patient Education and Counseling. 96, 273–280 (2014). https://doi.org/10.1016/j.pec.2014.06.017.

Chapter 2

MACHINE LEARNING

Keynote Abstract: 2, Introducing Keynote Speaker Mr. Bogdan Penkovsky

Towards autonomous API synthesis with deep reinforcement learning

Reinforcement learning (RL) is a general learning and decision making paradigm based on interaction with the environment. Despite being challenging in practice, implementing RL for real world tasks could improve the safety and efficiency of autonomous processes. In this work we apply RL for an autonomous molecule synthesis with continuous flow chemistry. In our preliminary experiments, we demonstrate that our agent trained by deep reinforcement learning is capable of responding to real-time challenges, such as changes in the environment, sensor noise, and perturbations in order to ensure the optimal chemical synthesis conditions. The ultimate goal of this work is to conceive an autonomous chemical production unit for active pharmaceutical ingredients (API).

Figure 2.1: Mr. Bogdan Penkovsky (Alysophil SAS)

Analyzing sequential Graph Generation with Graph Convolutional Policy Networks

Ruxandra Lasowski

Furtwangen University
ruxandra.lasowski@hs-furtwangen.de

Abstract. One property of Graph Convolutional Neural Networks (GCN) is to be permutation equivariant with respect to the node ordering. When GCN are used in combination with Reinforcement Learning (RL) and the action space depends on the node labeling then the action space is not equivariant by default. In this work we empirically show on very small graphs that the Graph Convolutional Policy Network introduced in [1] cannot generalize to symmetries introduced by node permutations. We extend the method to deal with permutation symmetries by using representatives of an isomorphic class of valid constructed subgraphs for a desired graph structure.

Keywords: Graph Generative Method, Graph Convolutional Networks, Deep Reinforcement Learning, Permutation Symmetries

1 Introduction

Symmetries in the data are a challenge for deep learning systems if the underlying neural network architecture doesn't account for it. In [2] the authors unify deep learning architectures under the hood of geometric deep learning and show how those deal with symmetries inherent in our world. Moreover they emphasize the success of several architectures because they are able to deal with certain kind of symmetries. For example, graph convolutional neural networks are by design permutation equivariant when computing node embeddings. This property ensures that if the labeling is permuted then the node embeddings will permute in the same way. When the graph is represented by a diagram, it means that the embeddings will be attached to the same position in the graph regardless of the label/ordering. We illustrate the equivariance property on a path graph with three nodes in Fig. 1.

In a recent work [1] the authors propose a graph convolutional policy network for drug generation and optimization towards a desired chemical property. The method is able to start generating graph molecules from scratch, i.e. from one node, or to continue from a subgraph making the method very attractive also for other applications like mesh generation or mesh completion. But before trying to use the method in other applications we need to understand it's generalization capabilities. In this work we focus first on how this method will generalize to the symmetries introduced by the symmetric group S_n, i.e. all possible permutations of node labelings. All those graphs are isomorphic. That means that there is a bijection between the vertex sets of the involved graphs that preserves the adjacency of the vertices from one graph to the other. The permutations can be represented as matrices and the mapping between two isomorphic graphs G_1, G_2 can be written as $G_1 = P^T G_2 P$. We consider a reinforcement learning setting where the agent hasn't seen all possible permutations and our research question is whether the method is capable to account for them. The involved complexity is in general n factorial and therefore we analyze a toy problem which consists of the sequential generation of a 2×2 grid graph. As an introductory example lets consider that the agent is presented a path graph of four nodes as illustrated in Fig. 2. By taking into account the topology of a path graph, there are $4!/2 = 12$ distinct node labelings that the agent has to deal with. All those 4-node path graphs with different labeling represented as diagrams are isomorphic. In this representation the graphs look to us humans equivalent and we would connect the ends regardless of the labelings to obtain a 2×2 grid graph. This visual equivalence disappears also for humans when the graph is represented by an adjacency matrix or node embeddings (there are also graphs represented by diagrams that are isomorphic yet difficult to recognize as isomorphic by visual inspection). All those isomorphic path graphs have different adjacency matrices and also permuted node embeddings that are generated by a graph convolution. So for an agent in a reinforcement learning setting these are different states. This means that we need to ensure that either the action space is also equivariant or all equivalent states, i.e. isomorphic graphs, can be mapped to one state, i.e. a representative. Homomorphic Markov Decision Process Networks(Homomorphic MDP Networks) [3] achieve equivariance under actions. In the latter work the authors propose to construct equivariant layers in the input and action space by identifying the involved group structured symmetries. Equivariance is thus hardwired in the network architecture. This approach would work for our application for very small examples due to the involved complexity of n factorial. Path graphs and grid graphs do not have $n!$ distinct adjacency matrices however all those permutations that generate distinct adjacency matrices do not form in general a subgroup of S_n because they are not closed under composition (see Appendix A for the number of distinct adjacency matrices of different graphs and the discussion related to the only subgroup of S_4 with 12 elements called A_4). Here we propose to use representatives of a valid subgraph that would lead to the construction of the desired grid graphs. Our approach shifts the complexity that is present on the algorithmic side by performing a graph isomorphism test between the constructed subgraph and representatives that we identify. We map then valid constructed subgraphs to those representatives. We use the algorithm that is implemented in NetworkX [4] that is based on the paper [5]. The graph isomorphism is a problem for which no polynomial algorithm is known. And it is still open if it is NP-complete. The latest achievement [6] in this regard showed that it can be solved in quasi-polynomial time. For certain classes of graphs like planar graphs, as are grid graphs, there exist polynomial time algorithms [7], [8]. Instead of using representatives of valid subgraphs we can cast the problem as subgraph isomorphism: Given a grid graph and a constructed subgraph test if the given grid graph contains a subgraph that is isomorphic to the constructed subgraph. Unlike graph isomorphism, subgraph isomorphism is proven to be an NP-complete problem.

Our contribution is to explore and go towards a direction that considers special classes of graphs where the intractable symmetry permutations on the nodes could be handled by a tractable graph isomorphism test.

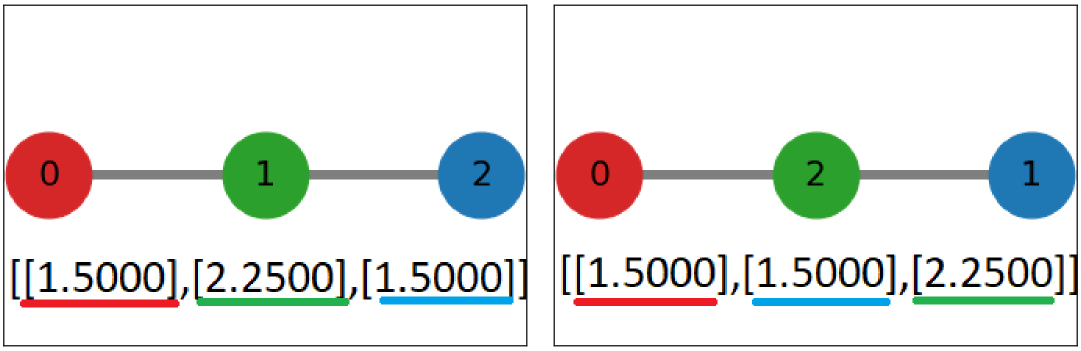


Fig. 1. Equivariance property illustrated on a path graph with three nodes. Nodes 1 and 2 are permuted. Node features are here scalar values. Visually, the node features (red, green, blue) stay at the same position in the diagram regardless of the labeling. The internal representation in terms of a vector will permute the values according to the node permutations. Notice that when only the topology is considered for the embeddings, the end nodes have the same embedding values.

2 Background

2.1 Deep Reinforcement Learning

In a reinforcement learning setting an agent learns how to act in an environment by receiving a positive or negative feedback called reward. The agent is trained such that it maximizes the cumulative reward. Formally, the environment is represented by a Markov Decision Process (MDP) which is a tuple $M = (S, A, R, T, \gamma)$, where $s \in S$ is a Markov state, $a \in A$ an action that an agent can take, $R : S \times A \to \mathbb{R}$ is a reward function that returns a scalar, γ is a discount factor to weigh recent rewards more than future rewards and $T : S \times A \times S \to [0, 1]$ is a transition function that assigns a probability for a pair of states and action of transitioning from the first to the second state. The goal of an agent is to find a policy $\pi : S \times A \to [0, 1]$, a function assigning probabilities for taking an action in a state, that maximizes the cumulative reward over an infinite time horizon. In deep reinforcement learning the policy is parametrized by a deep neural network.

2.2 MDP Homomorphism

In RL many tasks exhibit various types of redundancies and symmetries [9], [10] and researchers formalized these symmetries through the concept of Markov Decision Process Homomorphism (MDP Homomorphism) [10] that captures equivalent state-action pairs. Informally, two states are equivalent if for every action in the first state there is some valid possibly different action in the other state that produces similar results. Going back to our application, consider again two isomorphic 4-node path graphs $0-1-2-3$ and $1-0-3-2$. A valid action in the first graph is to connect node 0 and node 3 and the result is a 2×2 grid graph. A different action in the second graph connecting node 1 and node 2 leads also to the construction of a 2×2 grid graph isomorphic to the first result. The goal of an MDP homomorphism is to find a minimal number of admissible state-action pairs, solve the problem in this reduced MDP and transfer the solution back to the original MDP. This is possible since under an MDP Homomorphism the equivalent state-action pairs share the same optimal Q-value and optimal value function [10]. When an MDP exhibits symmetries these can be treated as a special case of MDP homomorphism [10], [3]. In GCPN the action space is directly connected to the state space (see 3) so we are only interested in state equivalence (and not any more in equivalent state-action pairs). As the method is graph generation, the number of nodes and/or edges increases over time, so we need to identify for each number of nodes and edges valid subgraphs of the desired graph structure.

3 Graph Environment

We adopt the design of the environment as in [1]. The state of the MPD is represented by the adjacency matrix of the graph to be constructed. The agent can add a node with an id greater than the actual number of nodes and connect it to the actual graph or it can connect two nodes in the actual graph. During training it learns to respect the topology of a grid graph. Thus the action space consists of a pair denoting the node ids that we vectorize. The size is $maxnode * maxnode$ where maxnode is 4 for 2×2 grid graph. We note that for example if the state consists of a subgraph containing a node with id id then actions $[id, id1]$ and $[id, id2]$ with $id1 \neq id2$, $id1, id2 > id$ and $id1, id2 \notin$ subgraph then both actions map to the same state. This invariance, as we will see in the results section, has some impact on what actions an agent learns.

3.1 Reward design 2×2 grid graph

We adapt the reward design according to our 2×2 grid graph: a positive step reward is assigned for constructing a path graph and a final positive reward is assigned when the grid is constructed. A negative step reward is assigned if there is a deviation from a path graph, like a star graph, and a negative final reward is assigned for not connecting the ends of the path graph.

3.2 Learning

For training the agent we use the Proximal Policy Optimization (PPO)[11] algorithm as provided in [12]. The state, i.e. the adjacency matrix is used some agents to perform an one layer GCN and pass the node embeddings to the policy model. Input features for the GCN layer are vectors of ones for each node of the size of the nodes of the graph. The embeddings and policy are trained jointly. We train following agents:

- **Agent1** is trained for constructing a 2×2 grid graph where the state is represented as an adjacency matrix and is passed directly to PPO.
- **Agent2** is trained for constructing a 2×2 grid graph where an one layer GCN that computes 2-dimensional node embeddings is introduced and the features/node embeddings are passed to PPO.
- **Agent3** works like **Agent2** but states belonging to valid subgraphs are mapped to a representative. This state is going through the GCN layer and then passed to PPO.

The agents are trained with default hyperparameters.

What are the valid subgraph representatives? For the 2×2 grid graph, valid subgraphs are $2, 3, 4$-nodes path graphs.

4 Results

Our test scenarios for the agents are to build the desired grid graphs from one node and from any valid subgraph. We summarize the results in Table 1:

4.1 Results Agent1

Agent1 could construct a 2×2 grid graph starting from one node. When starting from a 4-node path graph with $4!/2 = 12$ states only 4 have been seen during training and only those could be reconstructed.

Table 1. Results table.

Agents	Grid construction from one node	Generalization to S_4 for path graphs
Agent1	1.0	0.33
Agent2	1.0	0.42
Agent3	1.0	1.0

4.2 Results Agent2

Agent2 could also construct a 2×2 grid graph starting from one node. This agent could identify one additional unseen state compared to Agent1 when starting from a 4-node path graph due to the graph convolutional layer that introduced an invariance induced by the topology of a path graph. Path graph $0 - 1 - 2 - 3$ has identical node embeddings with path graph $0 - 2 - 1 - 3$. We illustrate it in Fig. 2.

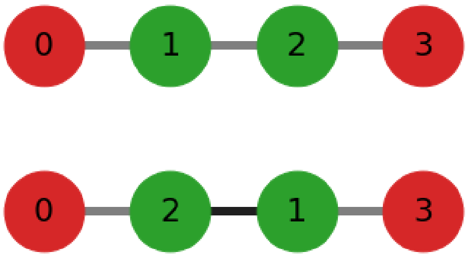


Fig. 2. Invariance introduced by GCN: 4-node path graphs with different adjacency matrices but equal embeddings. Upper graph: state that has been seen during training. Down graph: state that has not been seen during training. Node 1 and node 2 are permuted. Equal features or node embeddings are colored with the same color. The optimal action is to connect node 0 with node 3 to obtain a 2×2 grid graph.

4.3 Results Agent3

Agent3 learned to build the 2×2 grid graph with only one action $[0, 3]$ starting from a graph with one node, id = 0. This is because of the invariance that we mentioned in section 3 and the mapping of the valid path graphs to one representative. We lay out the sequence and we omit the action since it is the same:

1. Node id 1 and path graph $0 - 1$ is created.
2. Node id 2 and path graph $2 - 0 - 1$ is created.
3. Path graph is mapped to representative $0 - 1 - 2$.
4. Node id 3 and path graph $3 - 0 - 1 - 2$ is created.
5. Path graph is mapped to representative $0 - 1 - 2 - 3$.
6. Node id's 0 and 3 are connected and episode ends successfully.

When starting from a 4-node path graph all 12 states have been recognized due to the mapping to one representative of an isomorphic class.

5 Discussion and Conclusion

In this toy examples we have seen that in the context of RL also the action space has to be taken into account if the considered application needs to generalize to all equivalent state-action pairs that are not seen during training. We enumerated all valid subgraphs of equivalence classes that could occur during the sequential construction of the grid graphs. We think that even for the small graphs considered it was a better option to perform isomorphism graph tests than to incorporate the equivariance in the action space through the permutation group. For the 2×2 grid graph there were only 2 representatives to check against compared to 24 different permutations. In the analyzed method for Graph Convolutional Policy Networks [1] the authors used imitation learning and generated valid subgraphs of the desired distribution. This approach could be used to generate representatives of an equivalence class of a subgraph. This may not include all representatives but it would scale.

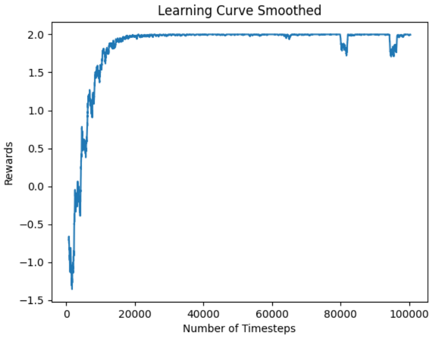


Fig. 3. Rewards for Agent1 training.

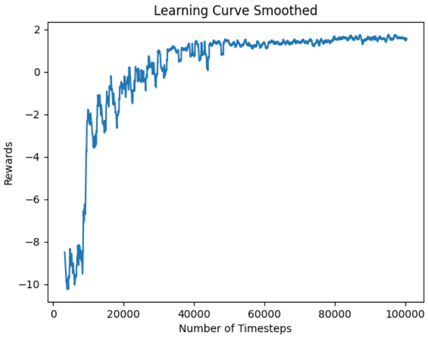


Fig. 4. Rewards for Agent2 training.

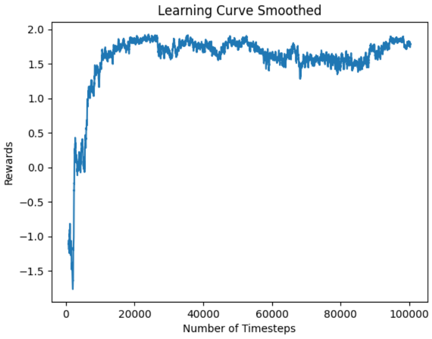


Fig. 5. Rewards for Agent3 training.

References

1. You, J., Liu, B., Ying, R., Pande, V.S., Leskovec, J.: Graph convolutional policy network for goal-directed molecular graph generation. CoRR **abs/1806.02473** (2018)
2. Bronstein, M.M., Bruna, J., Cohen, T., Velickovic, P.: Geometric deep learning: Grids, groups, graphs, geodesics, and gauges. CoRR **abs/2104.13478** (2021)
3. van der Pol, E., Worrall, D.E., van Hoof, H., Oliehoek, F.A., Welling, M.: MDP homomorphic networks: Group symmetries in reinforcement learning. In: Advances in Neural Information Processing Systems. (2020)
4. Hagberg, A.A., Schult, D.A., Swart, P.J.: Exploring network structure, dynamics, and function using networkx. In Varoquaux, G., Vaught, T., Millman, J., eds.: Proceedings of the 7th Python in Science Conference, Pasadena, CA USA (2008) 11 – 15
5. Cordella, L.P., Foggia, P., Sansone, C., Vento, M.: An improved algorithm for matching large graphs. In: In: 3rd IAPR-TC15 Workshop on Graph-based Representations in Pattern Recognition, Cuen. (2001) 149–159
6. Babai, L.: Graph isomorphism in quasipolynomial time [extended abstract]. In: Proceedings of the Forty-Eighth Annual ACM Symposium on Theory of Computing. STOC '16, New York, NY, USA, Association for Computing Machinery (2016) 684–697
7. Hopcroft, J.E., Wong, J.K.: Linear time algorithm for isomorphism of planar graphs (preliminary report). In: STOC '74. (1974)
8. Datta, S., Limaye, N., Nimbhorkar, P., Thierauf, T., Wagner, F.: Planar graph isomorphism is in log-space. In: 2009 24th Annual IEEE Conference on Computational Complexity. (2009) 203–214
9. Zinkevich, M., Balch, T.: Symmetry in markov decision processes and its implications for single agent and multi agent learning. In: In Proceedings of the 18th International Conference on Machine Learning, Morgan Kaufmann (2001) 632–640
10. Ravindran, B.: An Algebraic Approach to Abstraction in Reinforcement Learning. PhD thesis, University of Massachusetts, Amherst MA (2004)
11. Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O.: Proximal policy optimization algorithms. CoRR **abs/1707.06347** (2017)
12. Raffin, A., Hill, A., Gleave, A., Kanervisto, A., Ernestus, M., Dormann, N.: Stable-baselines3: Reliable reinforcement learning implementations. Journal of Machine Learning Research **22**(268) (2021) 1–8

A Appendix

Due to the symmetric topology of path and grid graphs, not all permutations of the labeling generate distinct adjacency matrices. The question for a graph with n nodes is: Do those restricted set of permutations form a subgroup of S_n? Group structure is required for hardwiring equivariant action spaces in Homomorphic MDP's [3]. By Lagrange theorem, the order of a subgroup divides the order of the group. We identified computationally that the number of distinct adjacency matrices that are generated by a restricted set of permutations is a divisor of the group order and the cardinality matches the order of the subgroup. However as we will see the restricted set does not always form a subgroup. The intermediate steps leading to a 2×2 grid graph are path graphs with $2, 3, 4$ nodes. Starting from 3 nodes, permutations of the labeling generate isomorphic graphs with different adjacency matrices. There are $n!/2$ permutations that generate distinct adjacency matrices of path graphs. For S_3 we can find $3!/2 = 3$ of them forming a subgroup: $\{(1,2,3),(2,3,1),(3,1,2)\}$. For S_4 there exists only one subgroup of order $4!/2 = 12$, the alternating group A_4, in cycle notation:

$$\{e,(12)(34),(13)(24),(14)(23),(123),(132),(124),(142),(134),(143),(234),(243)\}$$

The identity permutation $e = (1,2,3,4)$ and $(14)(23) = (4,3,2,1)$ generate the same adjacency matrix. The missing permutation to generate the 12-th distinct adjacency matrix would destroy the group structure by violating the closeness property.

Explainable AI: A key driver for AI adoption, a mistaken concept, or a practically irrelevant feature?

Julia Dvorak[1], Tobias Kopp[2], Steffen Kinkel[2] and Gisela Lanza[1]

[1] Institute of Production Science (wbk), Karlsruhe Institute of Technology
julia.dvorak@kit.edu
[2] Institute for Learning and Innovation in Networks (ILIN), Karlsruhe University of Applied Sciences
tobias.kopp@h-ka.de

Abstract. Explainable artificial intelligence (xAI) has become a popular subject of research amongst AI scholars in the last years. Some scholars consider xAI a significant driver of AI adoption in practice. However, at date, only a few studies investigated the conditions under which xAI solutions provide benefits in practice. Additionally, there is still a lot of controversy and inconsistency about related terminology revealing large conceptual differences between the understanding of explanations from a theoretical social science viewpoint and from a technological viewpoint. In this article, we strive to contribute to a more realistic picture of the potential and practical application scenarios of xAI. Thereby, we clarify the question whether xAI is a key driver for AI adoption, a mistaken concept from a theoretical point of view or perhaps a practically irrelevant feature and bridge the gap between different disciplines.

Keywords: explainable AI; artificial intelligence; trust; manufacturing; real-life applications.

1 Introduction

Along with the growing interest in the research community, the number of publications in the field of explainable artificial intelligence (xAI) has increased sharply in the last approximately five years [1]. xAI is attributed the potential to significantly drive AI adoption in practice [1], which is supposedly hindered by the black box nature of many AI models [2]. Thereby, xAI refers to the capacity of an AI system to explain either the model itself to the developer to achieve intrinsic interpretability or a specific AI outcome to a user as part of some kind of post-hoc reverse engineering process [3–5]. In that sense, xAI aims at explaining “the way in which an algorithm works in order to understand how and why it has delivered particular outcomes” [4].

The assumption that xAI can drive AI adoption in practice leads to the question of its practical benefits. In recent years, several researchers have strived to identify and collect possible motivations to implement xAI solutions. Explanations can be used to evaluate the AI application, to justify its reliance, to control its outcomes, to discover and to learn from it, i.e., to serve educational purposes [1, 5, 6]. In newer publications, the motivations for xAI applications are enriched with the purpose to manage AI applications [6]. Apart from social and practical reasons, some of these purposes address legal issues such as the demand for transparency [7], which has gained importance since it has been defined as a key criterion for trustworthy AI by the EU [8], or the right to “obtain meaningful information about the logic involved” [9] as granted by the European’s General Data Protection Regulation (GDPR) [1, 2].

Despite the multitude of possible motivations to use xAI, the actionability of its output remains unclear, i.e., there is a lack of empirical studies investigating whether recipients of explanations are able to derive beneficial practical implications [10]. Additionally, there is still a lot of controversy and inconsistency about the terminology. This leads to a variety of co-existing definitions and large conceptual overlaps with related terms such as interpretability or transparency, which also triggers diverse expectations towards xAI by relevant user groups [7, 11]. Analogous to the term intelligence, explanation refers to a concept that has its roots in human and social sciences but is also present in everyday language. Consequently, these concepts are loaded with associations and expectations. Simply transferring them to technical contexts can easily result in exaggerated expectations.

Figure 1 visualizes the elements relevant to a xAI system. An AI application is applied to a specific data set and generates an output. The output of the machine learning algorithm and its quality thereby depend on the quality of the respective input data. Since the output of an AI algorithm is often opaque, a xAI can be used to gain further insights into the mechanisms that have led to this particular output or the model of the AI itself. A human-machine interface makes these insights accessible to the user in a specific context. The human feedback may in turn be used as an input for the xAI.

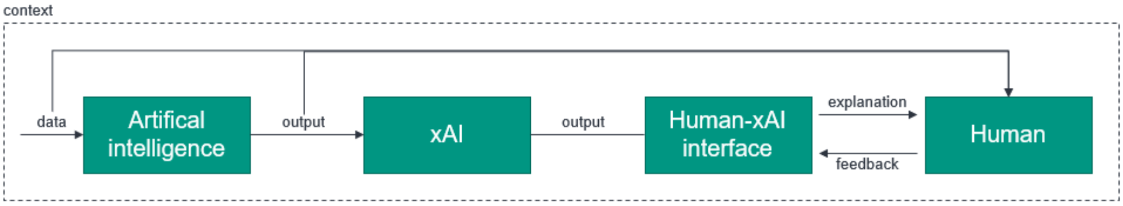


Figure 1 AI, xAI and Human embedded in a specific context following [12]

In this article, we aim to critically reflect on the opportunities and the range of practical use cases of xAI from two perspectives to contribute to a more realistic understanding and more adequate expectations regarding xAI. First, we highlight some conceptual limitations of xAI from a theoretical viewpoint mainly driven by arguments from scholars of philosophy and the social sciences. Thereby, we want to answer the question whether and in what sense xAI might be a mistaken and sometimes misunderstood concept. Second, we will adopt a practical viewpoint to analyse whether and in what application contexts xAI might be either a beneficial or a practically irrelevant feature. With this in mind, we will narrow and specify the potential application domains of xAI from two thrusts, while also highlighting relevant potential use cases in appropriate contexts. Finally, we will bring theoretical and practical considerations together in order to derive practical implications as well as future research avenues in the final chapter.

2 Theoretical viewpoint: Explainable AI as a mistaken concept?

Human beings are familiar with explanations since they represent an essential part of everyday communication. We regularly ask why a certain event (the explanandum) happened and hope to be provided with an answer (the explanans) that satisfies our information needs regarding reasons and causes. After having received an explanation, we might be asked whether we feel that we have understood a certain phenomenon. However, in many cases, the answer to this question is gradual in a sense that we have understood something “a bit more”, but our information needs are not entirely satisfied. At the contrary, the provided explanation let further questions arise, so that explaining becomes a complex, iterative, and interactive process [13–15].

Due to our familiarity with explanations, we can quite easily develop a vision of what is supposedly meant by xAI or by AI-generated explanations. However, this also imposes the danger that we subliminally expect AI explanations to be of a similar kind than explanations we know from our everyday communication amongst human beings. Unfortunately, despite being a common term in everyday language and a concept of folk psychology, there is no consensus on the criteria a certain piece of information has to meet in order to be conceived as an explanation, let alone on the criteria for a *good* explanation [1]. Accordingly, it is challenging to develop objective evaluation metrics for xAI approaches, an issue only addressed by a comparably low number of studies so far [16]. Hence, in the first place, xAI algorithms provide certain information, but do they qualify to be regarded as explanation in a narrower sense? In the remainder of this chapter, we provide four theses about the limits of xAI, which contradict common assumptions about the similar nature and explanatory power of xAI-generated and human explanations.

2.1 xAI does not provide explanations that resemble human explanations

Whereas scholars with technical backgrounds tend to consider explainability as a feature of the AI system, social scientists consider explanations as audience-dependent involving cognitive as well as social processes [15, 17]. The explainer exchanges some information regarding the explanandum with the explainee. However, when someone provides information about the conditions under which a certain outcome arises, this does not imply that the outcome is sufficiently explained. In contrast, the information recipient ultimately decides whether the information had explanatory power and supported him/her in interpreting the outcomes or in the understanding of the model itself. Put in other words, the definition of an explanation is relative to the explainee and its individual expectations, prior knowledge on the explanandum, information needs and so on [17]. Hence, the output of an xAI represents a necessary but not a sufficient piece of information which serves as an input for an explanatory cognitive process on the user side [17]. Ideally, this process should result in a feeling that someone has understood certain phenomena and its formation conditions. Hence, whereas explainability by definition focusses on whether some outcome is explainable per se, the focus should be shifted to the user side and its understanding of a phenomenon after having processed an explanans [18].

This argumentation draws on an interactional and processual understanding of explanations, which corresponds with the way humans explain things to each other as part of social interaction and conversation [14, 17]. Processing explanations is not an entirely rational and optimized process, which can be best supported by a large amount of exact data. Quite the contrary, those processes usually suffer from cognitive biases and are shaped by social expectations, which might prevent someone from revealing one's true reasons [14]. Furthermore, people prefer pragmatic, functional explanations, which can guide future behaviour, instead of mechanistic, data-driven explanations [18]. From a social sciences perspective, most explanations are contrastive, selective, social and do not refer to probabilities [14]. Accordingly , people usually do not seek for a complete list of causes when they ask for an explanation, but rather ask why a certain event happened instead of another plausible event [14]. Seemingly, most existing xAI approaches fail to take these criteria sufficiently into account. However, some approaches like counterfactual explanations or approaches to equip explanation engines such as SHAP (Shapley Additive exPlanations) with interactive features [19] can be considered as a step towards explanation processes that more closely resemble interactive human-to-human explanation processes. In particular, counterfactual explanations, which have a long tradition of research in analytic philosophy [20], resemble human explanations and thus might be positively evaluated in user studies, although they provide its user with only a limited set of information about possible causes [21]. Hence, whilst there are approaches to provide more human-like explanations, the conceptual differences seem considerably large at the moment. However, this does not mean that human-like explanations are the best possible explanations from a normative viewpoint, but they are at least the type of explanations which we are familiar with and which we might intuitively expect from xAI applications. Consequently, the conceptual differences at least underpin the relevance of expectation management.

2.2 xAI does not necessarily lead to correct expectations and higher trust in an AI system

Increasingly complex organisational styles of leadership as well as seemingly non-deterministic and unpredictably acting modern technology have given rise to the relevance of trust in human-technology interaction [22]. The described opaque, intangible, and complex nature of AI systems, their novelty as well as the manifold expectations and associations make them difficult to comprehend. Users often struggle to anticipate outcomes and to adequately assess the ability of AI systems, which in turn affects the evaluation of its trustworthiness [23]. Consequently, calibrating users' trust in AI and thereby influencing users' willingness to rely on an AI system is an important practical issue.

Basically, trust is about expectations regarding the unknown course of the future [24]. Hence, trust becomes relevant in situations in which persons perceive a lack of knowledge about the future while possessing some relevant information about the past from which they derive their expectations [24]. By enhancing users' knowledge about the working principles of AI systems, xAI applications strive to support users in making predictions about the future output of AI and thereby affect their trust in AI systems. Accordingly, fostering users' trust in an AI system is often considered as a major goal of xAI [7, 25] and measuring real-time trust as an opportunity to assess the quality of generated explanations [26].

However, research in related contexts of human-machine interaction have highlighted the significant relevance of a large variety of contextual as well as individual factors on trust evolution [27–29]. Preliminary results from the few available user studies in the human-AI interaction context seem to confirm this finding by indicating a complex and far from straightforward relationship between the characteristics of the xAI system and perceived trust [18]. For instance, [30] found that visual explanations increased perceived observability of an AI's working principles, but did not lead to an increase in trust. Additionally, meaningless placebic explanations without any explanatory power resulted in comparable levels of perceived trust than real explanations [31], just to name two examples. Consequently, much more theoretical and empirical work is required in order to analyze the relationship between perceived interpretability of AI output and human-AI trust under consideration of relevant contextual factors. Meanwhile, the widespread assumption that xAI implementations necessarily foster users' trust in any situation remains doubtable. An interdisciplinary perspective incorporating existing literature from sociology, psychology, philosophy of technology, and human-robot interaction seems promising to close this research gap [32].

2.3 xAI does not address the need for meaningful narratives

Explanations are usually provided in response to a why-question. Many reasons can drive someone to ask such a question [14]. For instance, if an AI application in manufacturing suggests replacing a certain part of a machine, one could ask "Why should I do this?". Or if an AI system refuses to grant creditworthiness, the applicant might ask "Why am I refused?".

Whilst the relation between why-questions and explanations seems intuitive and straightforward, a closer look reveals different notions and meanings of this question. As philosopher Daniel Dennett states, why-questions sometimes ask for a retrospective process narrative precisely answering the question "How did you come to this decision?" [33]. In case of the application for evaluating creditworthiness, the system could present a series of causes, mathematical calculations, and correlations to describe the process of generating its final decision. However, this kind of explanation is unlikely to satisfy the applicant and make him/her accept the decision. Indeed, automated decision-making systems often lack acceptance, because they fail to satisfy humans' need to understand decisions [2]. Here, it is crucial to have a closer look at the possible notions of the application's why-question. The latter may not only ask for a causal series of events, but also refer to intentions and meaning. Hence, they precisely ask the question "What did this happen for?" [33] and thereby request "justifications of the system's actions or recommendations *(why)"* as opposed to "rule-oriented explanations of *how* the system reasoned" [6]. This is in line with humans' tendency to build stories around their reasons for certain decisions to provide a deeper meaning and to foster trust. They "explain themselves by referring to post-factum coherent stories" [2] including additional environmental information. Oftentimes, humans also expect such stories as explanations. One's request for an explanation is not necessarily driven by the need for exact mathematical calculations, but by humans' condemnation to make-sense even in a "world of data, correlations, and probabilities" [34]. This process of sense-making requires convincing narratives instead of pure information. If a good friend betrayed you, you most likely would not be willing to forgive him/her, if he/she would only explicate some factors and correlations that made it quite probable that he/she would ever betray you. In this context, [34] concludes that humans have a narrative responsibility which is actually challenged by the rise of AI applications. The latter can provide a series of possible causes but fail to generate stories and narratives and to answer the important *what for*-question. In that sense, not every need for an explanation in a practical scenario can be addressed by xAI, because the ones asking for an explanation might indeed seek for a narrative.

2.4 xAI does not sufficiently take into account diverging addresses and their needs

The arguments brought forth so far have stressed an interactional understanding of explanation processes strongly involving the explainee's cognitive processes. This hints at intersubjective differences in these processes. However, many articles about xAI do neither specify their target audience nor name the motives for which their xAI can be usefully applied. Thereby, they overlook that the requirements and information needs of different target groups for explanations substantially differ [2], especially if they have varying levels of expertise with respect to AI technology and domain-specific knowledge [2, 6]. This can have various implications in terms of how they understand and emotionally evaluate the explanations. Whereas novices might require very detailed and easy-to-comprehend explanations, AI-experts might feel offended by easy explanations which disregard their experience and knowledge [2]. Additionally, laypersons might tolerate a less accurate but understandable explanation whereas experts might prefer more accurate but very technical and complex explanations [1]. It has to be taken into account that the real-life end users of xAI systems can also encompass non-technical users [10].

Whereas most publications omit to address this issue, some scholars have tried to raise attention for variety of possible user groups and their diverging demands in recent years [35–37]. For instance, [37] proposed to differentiate stakeholders into regulators, users, developers, and persons affected by AI outputs. Still, the knowledge about the influence of user characteristics on users' perception of explanations and the attitudinal, affective, and behavioral implications is limited [5, 11, 38]. However, these are key questions in terms of xAI adoption in practice as highlighted by a recent meta-analysis on the future of AI at work [39]. In that sense, apart from the explanandum (what to explain?) and the explanans (how to explain?), the explainee (to whom to explain?) should be considered as a third essential building block of an explanation [2]. This would also stress the value-added by personalized explanations [6]. However, since some of the most popular xAI approaches like LIME (Local Interpretable Model Agnostic Explanation) or SHAP are model- and application-agnostic, they lack methods of personalization. Hence, this advantage in terms of wide applicability of these techniques backfires in terms of an appropriate consideration of application-specific user needs [40].

Surprisingly, although companies represent the key application domain for AI and xAI solutions, an organizational and workplace-oriented perspective on xAI is underrepresented in related literature and in public discourse [11].

3 Practical viewpoint: Explainable AI as a practically irrelevant feature?

Until this point, we discussed xAI with a strong focus on the nature of explanations and the differences between human and xAI-generated explanations. The forthcoming section focusses on the practical usefulness of xAI in an organizational context, without considering technical details of these applications. Highlighting the relation between AI, xAI and the user, several criteria to assess existing use cases regarding their potential of an xAI application will be discussed. The identification of use cases goes thereby in line with a set of design recommendations for an xAI. We use the domain of production as an example. However, we believe that the criteria identified are rather associated with the nature of xAI than with a specific application domain and it is the goal to develop criteria that can identify xAI use cases on a domain-independent level. .

3.1 Applications of AI and xAI in company

Use cases for AI are found across a broad range of domains as for example production, medical, mobility, but also education or scientific work. In a study conducted by Fraunhofer IAO potential fields of application for AI are identified, including autonomous robots and transport, but also cognitive assistants and smart devices [41]. Production as it is considered in the course of this publication is not a “primary” domain such as for example autonomous driving, but there remains a broad range of AI use cases. In [42], a variety of such use cases are worked out abstractly, including maintenance, quality management and control, automation technology, and product and process development. In particular in small and medium enterprises, the assumed potential of AI applications is far from being identified or even realized, mainly due to a lack of competence, obstacles for the actual implementation and data problems [43].

xAI use cases in industry include cases on predictive maintenance, business management, anomaly detection and modeling [16]. However, it remains open what the potential for xAI actually is across all levels from shop floor to management level. For practical applications, the motivation to use an xAI is essential to justify its application, as it has been discussed beforehand in the introduction [44]. Ranging from shop floor to management level, the availability of high-quality data is an essential basis to make decisions. Confidence in how the systems work must thus be created, but at the same time it must also be ensured that the database is of sufficient quality [45]. In addition to strategic decisions, other possible applications for xAI are, for example, the training of employees, product development and the sharpening of process understanding [46].

3.2 Derivation of criteria for use case analysis and xAI application

So far, we have highlighted some typical application domains, but the question how to identify future use cases to assess the potential of xAI applications remains open. In the course of this section, several criteria will be presented that could serve practitioners as a first indication for an xAI potential. Thereby, neither exhaustiveness nor completeness is claimed.

The first aspect to mention is the need for an explanation [5], although from the authors' point of view this aspect can also be well justified via the objectives of an xAI. As a second criterion the criticality of for example a decision is to mention [5, 31]. The criticality can therefore refer to the fault tolerance of a use case, the criticality in time [41] and the criticality of data itself, resulting in criticality of the use case itself and the resulting implications [47]. Along with time criticality comes the decision if an explanation must be provided in real time or if the post-hoc provision of an explanation is sufficient for evaluation and knowledge enhancement purposes [48]. This question is particularly relevant in the context of manufacturing work, which is often associated with time pressure. Furthermore, the time window for a reaction to a decision and the explanation provided by an AI or xAI must be considered. For decisions of the strategic management in companies which are associated with a high impact, time criticality is not an important aspect, however, since those may be based on simulations, explanations are of crucial importance [46]. Also from a practical point of view the prior knowledge of stakeholders of a practical use case or system must be subject of consideration [49].

3.3 xAI design for practical application

In line with the previous aspects, it should be noted that the user of the xAI plays a central role and that the design of an xAI must be determined to its user group [50]. In their study, [50] contrast different explanations to different user groups and conclude that also especially the application context of an xAI is of great importance. This is also confirmed in other publications (e.g. [44]), particularly more recent publications. In addition to the type of information that is provided to the user, it is also recommended to consider the level of detail of information that should be provided to the user to ensure a communication appropriate to the target group [50]. It is emphasized that the user should be included in the early stages of an xAI development to ensure a fulfillment of expectations [12]. In current developments, this aspect has not been paid significant attention to [6]. If one considers further the relation between AI-xAI-human as shown in Figure 1, the user interface and the way of information visualization is an additional critical aspect [46]. From an entrepreneurial perspective, further challenges can be identified apart from the actual application. In [46], it is emphasized that its economic evaluation of xAI is difficult. If we look in particular at small and medium enterprises, we already find barriers to the adoption of AI approaches [51]. For example, in smaller companies it may be the case that human intervention is required along the information chain [52] and that an insufficient amount of data is provided for an AI application [53]. This may be the case due to several reasons, for example, if a number of different systems is used and data availability to even train an AI is not guaranteed [52].

Moreover, cost may be a critical criterion for the decision if an AI or xAI is even considered within the scope of potential solutions. Usually, before a product is fully developed, a number of prototypes can be created. However, in the case of an AI, this is not possible without major effort, including financial effort [51] and it is assumed that this applies analogously to xAI which also leads to financial risk at this point. Based on the previous chapters, it can also be stated that explanations that appeal to the user and are thus truly perceived as explanations offer the greatest added value. [49] underpins this approach by focusing on current demand and thus user requirements. Nevertheless, according to [47], these so-called application-grounded explanations, including real humans and real tasks, are also the explanations that are associated with the highest costs, which again represents a barrier for companies. However, it is conspicuous that even for the adoption of AI cost is not a primary factor, but major reasons range around competencies, data, or infrastructure [43].

4 Bridging the gap between theory and practice

In this section, we try to bring together the human-centric and the application-specific perspective on explanations and xAI. Alike for any other application within a company or production, a tool or technology must fulfill a function. The first statement refers to the human perception of explanations that are provided by an xAI, but that do not resemble human explanations. Thinking in terms of a practical use case, this aspect is to be considered as controversial, since sometimes human beings provide an inadequate amount of information than it is actually needed in a specific use case. This can on the one hand cause problems in time critical environments, but on the other hand contribute to trust building. From a practitioner's point of view, trust in a system is an essential aspect. Trust is hampered by the fact that AI system's behavior is not to be anticipated, but the question arises to which degree this must be the case in practical use cases. Explanations do not seem relevant in any cases, since we oftentimes accept decisions based on human reasoning, which is also a black box, and humans' explanations, which might not detail causes but only represent ex-post rationalizations [2]. xAI is often assigned the quality to foster users' trust in a system by unfolding the AI's working principles and decision mechanisms. However, if the underlying problem was so deterministic and obvious, would there be a need for an AI at all? AI is working with data, whilst human beings often share a broader view on a phenomenon as a whole including observations and implicit knowledge [54]. Thinking of use cases on the shop floor and in higher management, different context and settings are found with respect to time, consequences of decisions, and so on. The actual need for trust might be context-specific and there might even be cases, in which trust in an explanation is not an essential requirement (e.g. in cases without any human-AI interaction). However, in many cases, trust might be an essential success factor, but xAI applications might fail to modulate trust appropriately due to other overshadowing contextual factors.

Human beings do not only expect the transmission of an information, but rather information embedded in a story that provides a full picture with potentially more information that are required from a merely rational stance. Nevertheless, human beings do not act merely rationally and thereby desire and expect explanations in the form of narratives. Such explanatory stories can be shaped by subjective impressions. Thinking about production and for example predictive maintenance, the information itself may be of greater interest, but its visualization format may be of lower interest. In any case, the provided information and its visual representation should contribute to an adequate expectation management addressing the fact that human's expectations differ. However, since algorithms tend to work the same, a different set of algorithms may be needed for different people and different use cases.

This leads to the last aspect of consideration, the use specificity. From a practical perspective, there is a difference in talking to a developer or talking to a worker on the shop floor. Both will have different requirements, whilst an engineer might want to understand how the implemented algorithm works, the higher management wants to know why a specific decision should be taken based on a simulation. The claim for user specific explanations is addressed from a theoretical and also practical viewpoint alike. Information must be provided with respect to the context, the stakeholder, and the prior knowledge of a person.

Interdisciplinarity, therefore, fosters shedding light on relevant issues from different perspectives. Thereby, a holistic picture of current limitations, obstacles and challenges regarding xAI and its use in practice is created. Nonetheless, the importance of each aspect is determined by the context in which an AI and an xAI is applied. We therefore highly stress the importance of research focusing more on the actual adoption of xAI and system design, but also the transfer of xAI to real use cases and not solely on the algorithmic development. In addition, proper expectation management towards xAI is crucial. This may then again transfer xAI from a theoretical construct into a practically relevant feature.

5 Conclusion and implications for research and practice

In this contribution we approached xAI from a twofold perspective trying to answer the question if xAI is either a key driver for AI adoption, a mistaken concept, or practically irrelevant feature. First, the term explanation itself has been challenged and contrasted with human explanations. Second, the term trust within the scope of AI and xAI has been considered as a subject of interest, which is not necessarily influenced by the provision of AI-generated explanations. Third, the human need for storytelling instead of factual mathematical explanations was highlighted. This goes along with the fourth aspect pointing out the that current xAI implementations tend to provide general explanations, which are neither adapted to a specific situation nor a specific person. From a practical perspective, insights were provided into criteria that may promote or hamper the application of xAI. The widespread use of AI is a prerequisite for xAI adoption, but to use AI, data must be provided which is seen as a major obstacle by small and medium enterprises. Criticality of time and decision, control, speed of response and acceptance are identified as first criteria to assess AI-use cases for their potential of xAI. For sure, this list of criteria should be extended in further work.

In order to answer the research question, theoretical as well as practical issues and challenges were discussed. Thereby, we analyzed some important issues from different perspectives. Based on the theoretical-philosophical and the practical view the following implications for further research and practice can be derived:

- **Context specificity and user centeredness**. We highly emphasize the consideration of the context of an xAI application and the specific characteristics and needs of users. We therefore suggest not to neglect any of the relations indicated in Figure 1.
- **Empirical context-specific user studies.** The demand for a user-centered design of xAI approaches in practice requires a precise understanding how users perceive different explanation types in their typical usage contexts. Much more empirical findings are needed to be able to adequately design xAI approaches.
- **Expectation management**. Since the terms intelligence and explanation are common in everyday language and loaded with associations, practitioners will most likely expect something that explains in similar way than humans when they are told about xAI approaches. This can easily lead to false expectations and misleading ability attributions, which can negatively influence appropriate trust calibration. Mechanisms for expectation management and trust calibration should be developed and empirically tested to tackle this issue.
- **Promotion of AI for small and medium-sized enterprises (SMEs)**. xAI should not be considered as a universal solution to foster AI adoption in any company and usage context. Instead, it can be helpful for example when the AI adoption is hindered by transparency requirements. However, other typical obstacles such as missing technical competencies or a lack of high-quality data, especially in SMEs, must first be overcome to enable the use of xAI.

- **Criteria for xAI identification and heat map**. From a practical perspective, there is a need to develop and provide a guide for practitioners that empowers them to recognize beneficial xAI use cases based on a set of context- and use case-specific criteria.
- **Empirical analysis of overall potential of xAI in practice**. Apart from technical progress and single-case studies, empirical studies should also evaluate the potential of xAI in practice from a macro perspective. Thereby, they should address the research question, which practical use cases indeed fulfill the basic criteria for xAI to be feasible and advantageous. Such a study is foreseen within the context of the research project KARL.

Acknowledgements

This research and development project is funded by the German Federal Ministry of Education and Research (BMBF) within the "The Future of Value Creation – Research on Production, Services and Work" program (funding number 02L19C250) and managed by the Project Management Agency Karlsruhe (PTKA). The authors are responsible for the content of this publication.

References

[1] R. Confalonieri, L. Coba, B. Wagner, and T. R. Besold, "A historical perspective of explainable Artificial Intelligence," *WIREs Data Mining Knowl Discov*, vol. 11, no. 1, 2021, doi: 10.1002/widm.1391.

[2] N. Burkart and M. F. Huber, "A Survey on the Explainability of Supervised Machine Learning," *jair*, vol. 70, pp. 245–317, 2021, doi: 10.1613/jair.1.12228.

[3] A. B. Arrieta *et al.*, "Explainable Artificial Intelligence (XAI): Concepts, Taxonomies, Opportunities and Challenges toward Responsible AI," Oct. 2019. Accessed: Nov. 12 2021.

[4] D. Shin, "The effects of explainability and causability on perception, trust, and acceptance: Implications for explainable AI," *International Journal of Human-Computer Studies*, vol. 146, 2021, doi: 10.1016/j.ijhcs.2020.102551.

[5] A. Adadi and M. Berrada, "Peeking Inside the Black-Box: A Survey on Explainable Artificial Intelligence (XAI)," *IEEE Access*, vol. 6, pp. 52138–52160, 2018, doi: 10.1109/ACCESS.2018.2870052.

[6] C. Meske, E. Bunde, J. Schneider, and M. Gersch, "Explainable Artificial Intelligence: Objectives, Stakeholders, and Future Research Opportunities," *Information Systems Management*, vol. 39, no. 1, pp. 53–63, 2022, doi: 10.1080/10580530.2020.1849465.

[7] A. Brennen, "What Do People Really Want When They Say They Want "Explainable AI?" We Asked 60 Stakeholders," in *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems*, Honolulu HI USA, 2020, pp. 1–7.

[8] High-Level Expert Group on Artificial Intelligence, "Ethics Guidelines for Trustworthy AI," 2019. Accessed: Jun. 7 2022.

[9] Parliament and Council of the European Union, "General data protection regulation.," 2016. [Online]. Available: https://ec.europa.eu/info/law/law-topic/data-protection/data-protection-eu_de

[10] S. Verma, A. Lahiri, J. P. Dickerson, and S.-I. Lee, "Pitfalls of Explainable ML: An Industry Perspective," Jun. 2021. Accessed: Jun. 7 2022. [Online]. Available: https://arxiv.org/pdf/2106.07758.pdf

[11] E. Hafermalz and M. Huysman, "Please Explain: Key Questions for Explainable AI research from an Organizational perspective," *Morals & Machines*, vol. 1, no. 2, pp. 10–23, 2021, doi: 10.5771/2747-5174-2021-2-10.

[12] T. A. Schoonderwoerd, W. Jorritsma, M. A. Neerincx, and K. van den Bosch, "Human-centered XAI: Developing design patterns for explanations of clinical decision support systems," *International Journal of Human-Computer Studies*, vol. 154, p. 102684, 2021, doi: 10.1016/j.ijhcs.2021.102684.

[13] A. Vultureanu-Albisi and C. Badica, "Recommender Systems: An Explainable AI Perspective," pp. 1–6, 2021, doi: 10.1109/INISTA52262.2021.9548125.

[14] T. Miller, "Explanation in artificial intelligence: Insights from the social sciences," *Artificial Intelligence*, vol. 267, pp. 1–38, 2019, doi: 10.1016/j.artint.2018.07.007.

[15] P. Madumal, “Explainable Agency in Intelligent Agents,” in *AAMAS 2019*, 2019. Accessed: Feb. 12 2021.

[16] M. R. Islam, M. U. Ahmed, S. Barua, and S. Begum, “A Systematic Review of Explainable Artificial Intelligence in Terms of Different Application Domains and Tasks,” *Applied Sciences*, vol. 12, no. 3, p. 1353, 2022, doi: 10.3390/app12031353.

[17] K. O’Hara, “Explainable AI and the philosophy and practice of explanation,” *Computer Law & Security Review*, vol. 39, p. 105474, 2020, doi: 10.1016/j.clsr.2020.105474.

[18] A. Páez, “The Pragmatic Turn in Explainable Artificial Intelligence (XAI),” *Minds & Machines*, vol. 29, no. 3, pp. 441–459, 2019, doi: 10.1007/s11023-019-09502-w.

[19] M. Chromik, “Making SHAP Rap: Bridging Local and Global Insights Through Interaction and Narratives,” in *Springer eBook Collection*, vol. 12933, *Human-Computer Interaction – INTERACT 2021: 18th IFIP TC 13 International Conference, Bari, Italy, August 30 – September 3, 2021, Proceedings, Part II*, C. Ardito et al., Eds., 1st ed., Cham: Springer International Publishing; Imprint Springer, 2021, pp. 641–651.

[20] S. Wachter, B. Mittelstadt, and C. Russell, “Counterfactual Explanations Without Opening the Black Box: Automated Decisions and the GDPR,” *Harvard Journal of Law & Technology*, vol. 31, no. 2, pp. 841–887, 2018.

[21] G. Warren, M. T. Keane, and R. M. J. Byrne, “Features of Explainability: How users understand counterfactual and causal explanations for categorical and continuous features in XAI,” 2022.

[22] J. D. Lee and K. A. See, “Trust in automation: designing for appropriate reliance,” *Hum Factors*, vol. 46, no. 1, pp. 50–80, 2004, doi: 10.1518/hfes.46.1.50_30392.

[23] R. C. Mayer, J. H. Davis, and F. D. Schoorman, “An Integrative Model of Organizational Trust,” *The Academy of Management Review*, vol. 20, no. 3, pp. 709–734, 1995.

[24] P. Sumpf, *System Trust: Researching the Architecture of Trust in Systems*: Springer Fachmedien Wiesbaden, 2019.

[25] S. Laato, M. Tiainen, A. Najmul Islam, and M. Mäntymäki, “How to explain AI systems to end users: a systematic literature review and research agenda,” *INTR*, vol. 32, no. 7, pp. 1–31, 2022, doi: 10.1108/INTR-08-2021-0600.

[26] R. Setchi, M. B. Dehkordi, and J. S. Khan, “Explainable Robotics in Human-Robot Interactions,” *Procedia Computer Science*, vol. 176, pp. 3057–3066, 2020, doi: 10.1016/j.procs.2020.09.198.

[27] S. Marsh and M. R. Dibben, “The role of trust in information science and technology,” *Ann. Rev. Info. Sci. Tech.*, vol. 37, no. 1, pp. 465–498, 2003, doi: 10.1002/aris.1440370111.

[28] K. A. Hoff and M. Bashir, “Trust in automation: integrating empirical evidence on factors that influence trust,” *Human factors*, vol. 57, no. 3, pp. 407–434, 2015, doi: 10.1177/0018720814547570.

[29] P. A. Hancock, T. T. Kessler, A. D. Kaplan, J. C. Brill, and J. L. Szalma, “Evolving Trust in Robots: Specification Through Sequential and Comparative Meta-Analyses,” *Human factors*, 1-34, 2020, doi: 10.1177/0018720820922080.

[30] T. Schrills and T. Franke, “Color for Characters - Effects of Visual Explanations of AI on Trust and Observability,” in *Springer eBook Collection*, vol. 12217, *Artificial Intelligence in HCI: First International Conference, AI-HCI 2020, Held as Part of the 22nd HCI International Conference, HCII 2020, Copenhagen, Denmark, July 19–24, 2020, Proceedings*, H. Degen and L. Reinerman-Jones, Eds., 1st ed., Cham: Springer International Publishing; Imprint Springer, 2020, pp. 121–135.

[31] M. Eiband, D. Buschek, A. Kremer, and H. Hussmann, “The Impact of Placebic Explanations on Trust in Intelligent Systems,” in *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems*, Glasgow Scotland Uk, 2019, pp. 1–6.

[32] T. Kopp, “Facets of trust and distrust in collaborative robots at the workplace: Towards a multidimensional and relational conceptualisation,” *International Journal of Social Robotics*, under review.

[33] D. C. Dennett, *From bacteria to Bach and back: The evolution of minds*. New York, London: W.W. Norton & Company, 2017.

[34] M. Coeckelbergh, “Narrative responsibility and artificial intelligence,” *AI & Soc*, 2021, doi: 10.1007/s00146-021-01375-x.

[35] A. Kirsch, “Explain to whom? Putting the User in the Center of Explainable AI,” *Proceedings of the First International Workshop on Comprehensibility and Explanation in AI and ML 2017 colocated with 16th International Conference of the Italian Association for Artificial Intelligence (AI*IA 2017), 2017, Bari, Italy.*, 2018.

[36] R. Tomsett, D. Braines, D. Harborne, A. Preece, and S. Chakraborty, “Interpretable to Whom? A Role-based Model for Analyzing Interpretable Machine Learning Systems,” *in Proceedings of the 2018 ICML Workshop on Human Interpretability in Machine Learning (WHI 2018), Stockholm, Sweden*, pp. 8–14, 2018.

[37] M. Langer *et al.*, “What do we want from Explainable Artificial Intelligence (XAI)? – A stakeholder perspective on XAI and a conceptual model guiding interdisciplinary XAI research,” *Artificial Intelligence*, vol. 296, p. 103473, 2021, doi: 10.1016/j.artint.2021.103473.

[38] L.-V. Herm, J. Wanner, F. Seubert, and C. Janiesch, “I Don’t Get It, but It Seems Valid! The Connection Between Explainability and Comprehensibility in (X)AI Research,” in *European Conference on Information Systems (2021)*, 2021.

[39] M. Langer and R. N. Landers, “The future of artificial intelligence at work: A review on effects of decision automation and augmentation on workers targeted by algorithms and third-party observers,” *Computers in Human Behavior*, vol. 123, p. 106878, 2021, doi: 10.1016/j.chb.2021.106878.

[40] C. Panigutti, A. Beretta, F. Giannotti, and D. Pedreschi, “Understanding the impact of explanations on advice-taking: a user study for AI-based clinical Decision Support Systems,” in *CHI Conference on Human Factors in Computing Systems*, New Orleans LA USA, 2022, pp. 1–9.

[41] D. Hecker, I. Döbel, U. Petersen, A. Rauschert, V. Schmitz, and A. Voss, “Zukunftsmarkt Künstliche Intelligenz: Potenziale und Anwendungen,” 2019.

[42] B. Hatiboglu, S. Schuler, A. Bildstein, and M. Hämmerle, “Einsatzfelder von künstlicher Intelligenz im Produktionsumfeld,” Fraunhofer-Institut für Produktionstechnik und Automatisierung IPA, Stuttgart; Fraunhofer-Institut für Arbeitswirtschaft und Organisation IAO, Stuttgart, 2019.

[43] P. Ulrich, V. Frank, and M. Kratt, “Adoption of artificial intelligence technologies in German SMEs — Results from an empirical study,” in *Corporate governance: A search for emerging trends in the pandemic times*, 2021, pp. 76–84.

[44] B. Wickramanayake, C. Ouyang, C. Moreira, and Y. Xu, “Generating Purpose-Driven Explanations: The Case of Process Predictive Model Inspection,” in *Lecture Notes in Business Information Processing, Intelligent Information Systems*, J. de Weerdt and A. Polyvyanyy, Eds., Cham: Springer International Publishing, 2022, pp. 120–129.

[45] N. Tanwar and Y. Hasija, “Explainable AI; Are we there yet?,” *2022 IEEE Delhi Section Conference (DELCON)*, 2022.

[46] C. J. Turner and W. Garn, “Next generation DES simulation: A research agenda for human centric manufacturing systems,” *Journal of Industrial Information Integration*, vol. 28, p. 100354, 2022, doi: 10.1016/j.jii.2022.100354.

[47] F. Doshi-Velez and B. Kim, “Towards A Rigorous Science of Interpretable Machine Learning,” *arXiv: Machine Learning*, 2017, doi: 10.48550/arXiv.1702.08608.

[48] C. H. Park, “Anomaly Pattern Detection on Data Streams,” in *2018 IEEE International Conference on Big Data and Smart Computing (BigComp)*, Shanghai, 2018, pp. 689–692.

[49] T. Weber, H. Hußmann, and M. Eiband, “Quantifying the Demand for Explainability,” in *Springer eBook Collection*, vol. 12933, *Human-Computer Interaction – INTERACT 2021: 18th IFIP TC 13 International Conference, Bari, Italy, August 30 – September 3, 2021, Proceedings, Part II*, C. Ardito et al., Eds., 1st ed., Cham: Springer International Publishing; Imprint Springer, 2021, pp. 652–661.

[50] X. Wang and M. Yin, “Effects of Explanations in AI-Assisted Decision Making: Principles and Comparisons,” *ACM Trans. Interact. Intell. Syst.*, 2022, doi: 10.1145/3519266.

[51] G. Dove, K. Halskov, J. Forlizzi, and J. Zimmerman, “UX Design Innovation: Challenges for working with machine learning as a design material,” in *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, Denver Colorado USA, 2017, pp. 278–288.

[52] B. Pokorni, M. Braun, and C. Knecht, “Menschenzentrierte KI-Anwendungen in der Produktion: Praxiserfahrungen und Leitfaden zu betrieblichen Einführungsstrategien,” Fraunhofer-Institut für Arbeitswirtschaft und Organisation IAO, 2021.

[53] G. Siaterlis *et al.*, “An IIoT approach for edge intelligence in production environments using machine learning and knowledge graphs,” *Procedia CIRP*, vol. 106, pp. 282–287, 2022, doi: 10.1016/j.procir.2022.02.192.

[54] X. Li, Y. Koren, and B. I. Epureanu, “Complementary learning-team machines to enlighten and exploit human expertise,” *CIRP Annals*, vol. 71, no. 1, pp. 417–420, 2022, doi: 10.1016/j.cirp.2022.04.019.

Improved e-mail forensic using dynamic graphs and change-point detection

Christian Hiller and Andreas Wagner

Hochschule Karlsruhe - Technik und Wirtschaft
andreas.wagner@h-ka.de

Abstract. Fraudulent behavior costs the German healthcare system an estimated 14 billion euros per year. Reasons are, amongst others, criminal networks of nursing services, doctors and patients. To investigate such cases, authorities often examine the e-mail communications of suspects. This still requires very high effort in practice, as often all e-mail communication is actually read manually. This work proposes algorithms based on graph metrics and change-point-detection to automatically identify changes in the communication structure of e-mail accounts over time. This can speed up investigations, as it enables authorities to reduce the amount of data to evaluate manually. The starting point for the proposed method is a dynamic graph modeling of e-mail communication. Then graph metrics are calculated and the resulting time-series of graph metrics are analysed using change-point detection methods. An evaluation of the methods on the infamous ENRON data set shows the potential to support forensic investigations.

Keywords: fraud investigations; healthcare; e-mail forensic; dynamic graph metrics, density; average clusterin; change-point-detection.

1 Introduction

Billing fraud and corruption in the health care sector causes estimated costs of about 14 billion euros per year for the German social system [1]. In this context, corruption is often organized in network structures. Only recently, a German newspaper reported three cases of large-scale billing fraud by networks of doctors, nursing services and patients in the German cities of Munich and Augsburg. Each case resulted in damages between two and three million Euros [2]. When working on such cases, investigating authorities need to analyze large amounts of communications data in order to understand the structure of the network. They are interested in the actors involved, need to reconstruct the hierarchy of the network, try to identify important events, and, most importantly, gather legally tenable evidence.

In practice, mainly e-mail communication data is available to the investigators. In a typical fraud case there can be as many as 400.000 e-mails, which need to be taken into account. From our experience, the evaluation is often done manually, which means that a police officer is actually reading the e-mail content in order to determine valuable information. The analysis includes the evaluation of text, the temporal classification of messages, and the identification of groups, communities, and patterns in communication behavior. These activities require high capacities and lead to long investigation times. In the publicly funded research project *Kriminelle Netzwerke* [3] the authors work with German police authorities in order to develop tools and methods from mathematical graph theory to support the investigations with algorithms for data evaluation.

In this paper we introduce an approach based on dynamic graph interpretation and change-point analysis as a decision support system for investigators. It is based on the intuitive idea that suspicious elements of a criminal case are reflected in the communication behavior of participants and thus in the dynamic graph model. Automated identification of such changes can narrow down the time period and thus the amount of e-mails for investigators to examine. This frees up investigators' capacities. We propose to interpret the e-mail communication as a dynamic graph. Based on this dynamic graph representation we select graph metrics for particular use-cases of the investigator and automatically detect change-points in the structure of the network. This helps to identify conspicuous periods of time. We evaluate our approach on the famous ENRON e-mail corpus [4].

The paper is structured as follows. In Section 2 we give a summary of research on the application of graph-theory in police investigations. Section 3 introduces our approach using graph metrics and change-point detection (CPD). In a case study in Section 4 we validate the proposed approach on ENRON-data. We conclude in Section 5 with a summary and an outlook.

2 Related Work

In the context of e-mail forensics, (static) graph representations, as shown in Figure 1, have been used to analyze communication structures in previous research. There is also previous research on temporal changes in e-mail

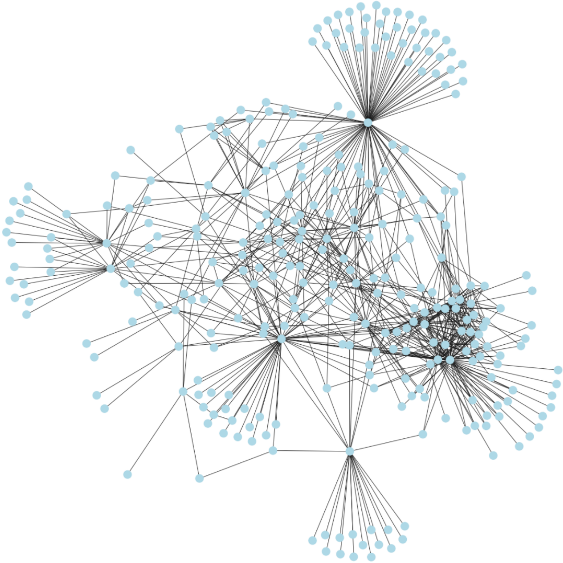


Fig. 1: Exemplary illustration of a communication network as a graph

users' communication behavior. In the following, we summarize the current state of research on those topics. The e-mail mining toolkit developed by [5], [6], and [7] is a modular framework for analyzing e-mail accounts through statistical analysis and graph representation of e-mail communications. [8], [9] and [10] present similar frameworks. They add functionalities for the classification and clustering of e-mails based on their content. The analysis of such graph models by computing graph metrics has been done in [11]. In [12] an application for dynamic graph visualization of e-mail data is presented. Although a temporal distribution of e-mails is also visualized here, a dynamic graph model is not used.

In general, *dynamic* graph models are rarely used and CPD is not applied in any of the previously cited research. In contrast, in research on social network analysis, which we evaluate in the following, dynamic graph representations and CPD are already used to identify conspicuous periods. This research concludes that structural changes in the dynamic graph representation of a social network can be indicators of changes in the real network. [13] model the e-mail communications of students and employees at a university and discuss the change in average node degree, clustering coefficient, length of the shortest path, and size of the largest connected component. [14] model the Bitcoin block-chain transactions as a graph and analyze the change of the graph over time on its adjacency matrix through principal component analysis and eigenvalue decomposition. [15] model communication through short messages from employees of a hedge fund as a dynamic graph and discuss the effects of internal and external events, such as drops in stock prices, on the structure of the graph and employee communication. It is concluded that negative events make communication more intense and the graph more densely connected. All approaches have in common a modeling of the dynamic graph in terms of discrete, successive static graphs. Studies show that real events are reflected in the structure of dynamic graph representations of social networks and that these can be captured by analyzing the time course of graph metrics. CPD approaches can identify such changes in an automated manner [16]. Advances in the fields of social network analysis and graph representation learning are continually yielding new approaches for identifying changes in the structure of graphs. The following are examples of some methods for CPD on graphs. [16] explore the identification of change points by analyzing the distribution function of the node degree of a dynamic graph. Change points are identified by computing a distance metric between the distributions of successive graphs. A hypothesis test is then used to determine the probability that each graph is a change point. [17] propose CPD on generative graph models. In addition, there are approaches to CPD on changes in the Laplacian spectrum of a dynamic graph [18] and an approach by clustering snapshots [19].

A study by [20] compares the results of CPD algorithms on traditional graph metrics with complex metrics derived from generative graph models. The authors conclude that analysis of traditional graph metrics yields only slightly worse results with significantly less complexity and computational power required. The most precise change points can be identified by the number of active nodes and edges. In contrast, the density, the average clustering coefficient and the average shortest path are robust to structural changes in a graph.

This literature review shows that graph-based methods for representing e-mail data are used in e-mail forensics. However, such frameworks rarely use dynamic graph models. Even further, work in the area of social network analysis shows that CPD approaches can be used to deduce real-world events from dynamic graph models. The goal of this work is to transfer such an approach to the field of e-mail forensics in order to support the work of police investigations.

3 Method

To identify periods or points in time of particular interest for investigation, an approach using CPD on a time series of a graph metric is designed. This approach requires the development of a model for representing the communication structure of e-mail networks, with special consideration of the evolution of these networks over time. Based on such a dynamic model, metrics and their changes over time are computed. These time-series then serve as an input for CPD algorithms.

(1) Preparation of data
To model e-mail data as a dynamic, structural communication graph, e-mail metadata is required in a format similar to Table 1. During preprocessing, e-mails with multiple recipients are split into multiple e-mails with one recipient. The information to how many recipients the e-mail was sent is stored in the 'No. of receivers' feature.

Table 1: Features used to describe e-mail communication.

Timestamp	Sender	Receiver	No. of receivers
1997-07-10 14:05:57	student1@h-ka.de	student2@h-ka.de	2

(2) Modelling a dynamic graph
To represent the network structure of e-mail data, it is modeled as an undirected, weighted graph. Nodes in the graph correspond to e-mail addresses. An edge exists between two nodes if there is at least one e-mail exchange between the e-mail addresses. The weight of an edge corresponds to the number of e-mails between the addresses. We do not use directed graphs, which would also represent the sender / receiver structure. This is a possible extension of our work.

To account for the number of recipients of an e-mail, an e-mail contributes with $1/\textit{total number of receivers}$ to the weight of an edge. To analyze the change over time, the graph is modeled dynamically. The dynamic is represented using a series of static graphs. An element of the series, i. e. a single static graph, is referred to as a *snapshot* in the following. The snapshots are build from the e-mail corpus such that the e-mails are divided into temporally consecutive blocks of equal length. The blocks not necessarily need to be disjoint. Changes between these snapshots represents changes in the communication structure over time. Note that we keep the set of nodes the same across all snapshots. It includes all e-mail addresses in the data-set. Nodes with no edges in a snapshot are called inactive.

The snapshot approach is chosen because it has been proven effective in the reviewed work on CPD on social network graphs. The time period from the first to the last e-mail is discretized and modeled as a sorted set T, where for each $t \in T$ the e-mails of the corresponding time period are aggregated. The length of the time interval Δt covered by a snapshot is a hyper-parameter. Short time intervals allow for precise detection of changes, whereas long-term structural changes and features may not be detected due to the smaller graphs that would result. One way to circumvent this issue is to consider *sliding* snapshots. In this case, a snapshot is taken for each t, although a snapshot still summarizes a period Δt. One possible modeling approach, illustrated in Figure 2, may be to discretize the observation period to day-level and choose $\Delta t = 7$. Accordingly, one snapshot covers one week. This approach has a smoothing effect on metrics over time that are analyzed on the graph. It remains to be noted that each e-mail is contained in Δt snapshots.

(3) Calculation of graph metrics
The next step is to determine graph metrics on the snapshots. The metrics along with their meaning shown in Table 2 are proposed.

The metric is calculated on each snapshot. These values form a time series that describes the change in the graph structure over time. The choice of the metric determines which information about the communication structure is represented.

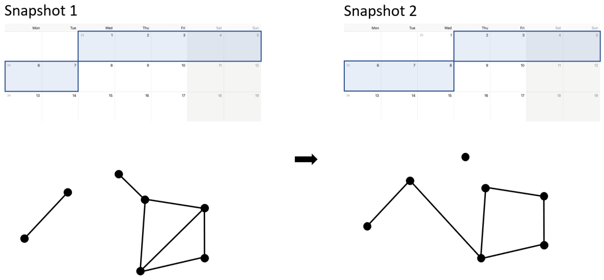


Fig. 2: Dynamic Graph Model

Table 2: Proposed graph metrics and interpretations for snapshot analysis.

Graph metric	Interpretation
Density	How closely connected are the accounts?
Average Clustering Coefficient	How strong is the clique formation in the graph?
Highest Centrality Account	Which account is particularly central to the communication?

The *density* of a graph can be interpreted as the degree of interconnection of the nodes. In the case of a communication graph, a structural change in density indicates a change in the communication behavior of the corresponding e-mail accounts. In the context of a police investigation, it is assumed that such a change is suspicious and e-mails in the corresponding period should be examined more closely.

The *average clustering coefficient* of a graph describes how strongly nodes tend to form cliques. A clique is a set of nodes, all of which are interconnected. Studies such as [15] show that individuals tend to communicate particularly intensely within their close network in response to unusual externalities. It is assumed that changes in clique formation behavior may reveal such externalities to investigators.

Centrality metrics can be used to evaluate the importance of a node within a graph. The *Highest Centrality Account* is the node, which takes the maximum centrality. In a communication graph, the centrality of a node can provide information about how important a certain actor is within the graph. For example, one possible interpretation of an important actor may be that it controls or significantly influences the flow of information in a group. Accordingly, a change of the node with the highest centrality can be an indicator for structural changes within a dynamic communication graph. For example, in the context of police investigations of criminal networks, such a change may indicate a shift in the hierarchy of the network. Since there are different ways to evaluate the centrality of nodes, various metrics can be tested using this approach. Exemplary metrics are *degree centrality* and *betweenness centrality*.

At this stage, even a simple visualizing of the time series can already identify time points and periods of particular interest, for example times of particularly low or intensive communication, or periodicity in the communication. However, we propose an approach to automatically detect structural changes in the following step.

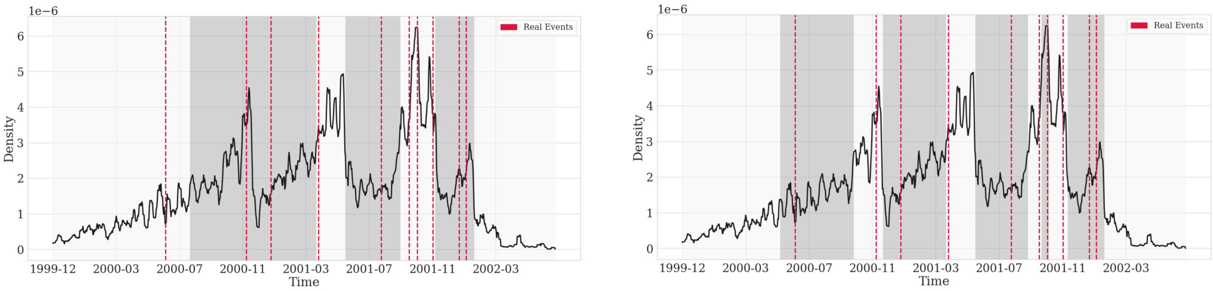
(4) Identification of anomalies

For the automated identification of structural changes in the time-series of a graph metric we propose change-point-detection (CPD). CPD algorithms divide a time series in sub-segments that are as homogeneous as possible. The boundaries between these segments are called change points. After determining these change points, investigators can examine the e-mail communication around the corresponding time in order to detect interesting events.

To design a CPD, we follow the approach of [21]. According to this approach, a CPD consists of a search strategy, a cost function and, if known, the number of change points to be determined. If the number of change points is known, this approach uses the algorithm of [22] based on dynamic programming. As a search strategy when the number of change points is unknown, [21] proposes the Pruned Exact Linear Time (PELT) algorithm of [23]. In this case, a linear penalty function $p_{linear}(k) = \beta|k|$ is used to limit the number of change points k. The penalty coefficient β is calculated depending on the cost function. The cost function is part of a CPD algorithm and chosen depending on the change to be detected. Typical cost functions are change-in-mean or change-in-variance (and others). We suggest change-in-mean for our application, so we use the cost function in Equation 1.

$$c_\mu(y_{a..b}) = \sum_{t=a+1}^{b} \|y_t - \bar{y}_{a..b}\|_2^2 \tag{1}$$

Hereby, $y_{a..b}$ are sub-segments of variable length of the time-series under investigation $y = \{y_t\}_{t=1}^{T}$.



(a) Unknown no. of change points

(b) Known no. of change points

Fig. 3: Change points of the density of the dynamic graph of Enron e-mail communication. The identified change points are located at the transition of the colors.

4 Case Study

We test our method on the famous Enron e-mail data set [4]. As part of the data cleaning process, e-mails sent to multiple recipients are transformed into multiple e-mails, each with a single recipient. After cleaning, the data set consists of 1,115,923 e-mails during the period from December 1999 to July 2002. A period of seven days per snapshot is chosen to model the dynamic graph. On the dynamic graph model, the previously mentioned graph metrics are calculated for each snapshot. Then, the resulting time series are segmented by the described approach to CPD using the `ruptures` library in Python 3 of [21]. The segment boundaries are interpreted as change points. The automatically determined change points are compared with real events from the environment of the Enron Group in the period under investigation showed in Table 3. These events are taken from chronicles of the Enron scandal from an article in a German business magazine [24] and a website of the UMKC School of Law [25].

Table 3: Real events from the environment of the Enron Group.

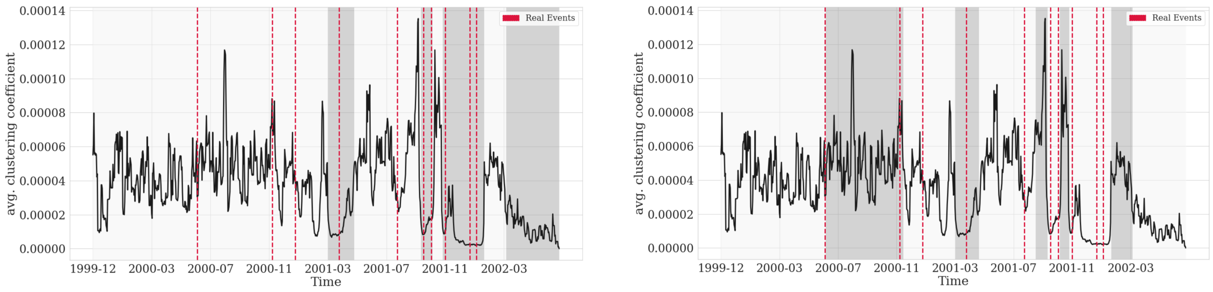
Date (dd.mm.yyyy)	Event
01.07.2000	Cooperation with Blockbuster
01.12.2000	Announcement of Jeffrey Skilling taking over as CEO
17.01.2001	Start of the electricity crisis in California
17.04.2001	*Asshole* call
14.08.2001	Skilling resigns / Kenneth Lay takes over as CEO
16.10.2001	Publication of high losses in the last quarter
22.-31.10.2001	First investigations against Enron become public
20.-26.11.2001	Drastic drop in Enron share price
09.01.2002	Announcement of investigations into accounting fraud
23.01.2002	Kenneth Lay resigns as CEO

In the following we discuss the analysis of the *density* and *average clustering coefficient* for the ENRON dataset. We use the PELT algorithm and chose a penalty coefficient $\beta = max(y_t)^2$, where y_t describes the time series of the density. In an intuitive sense, this penalty coefficient means that an additional change point will penalize the value of the cost function as much as the highest value of the time series.

In our figures, which we introduce in the following, we use two different grey-scale colors to mark the identified segments. The identified change points are consequently located at the transition of the colors. The real events cited in the table above are market by dotted vertical lines.

Density

Figure 3a shows the automatically identified change points on the time series of the density by the PELT algorithm. Six change points are identified by the PELT algorithm. Since the algorithm divides the time series into sub-segments that are as homogeneous as possible, the change points do not necessarily correspond to points in time that are particularly conspicuous visually. The plot shows that one real event in April 2001 corresponds exactly to one change point. Three other change points are close to other real events that are in the period of increased density at the end of 2001. The remaining real events cannot be predicted by this approach. Due to the unknown number of change points, obviously not each change point can be assigned to each real event.



(a) Unknown no. of change points

(b) Known no. of change points

Fig. 4: Change points of the average clustering coefficient of the dynamic graph of Enron e-mail communication. The identified change points are located at the transition of the colors.

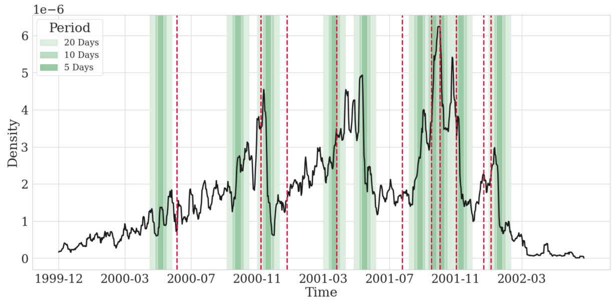


Fig. 5: Curve of the density of the dynamic graph of the Enron data set with real events and periods of investigation of 5, 10 and 20 days before and after ten change points

We also fix the number of change points and use [22]. It shall be tested how close the proposed method does predict real events. Therefore, according to the number of real events, the number of change points searched is set to ten[1]. The change points determined in this way are visualized in Figure 3b. As with the testing of the first approach, it is observed from Figure 3b that some change points are close to real events. In addition to the change points identified by the PELT algorithm, change points in late 2000 and late 2001 can be identified that are close to real events. Some other real events can again not be predicted (however, these events may not even be contained in the communication structure).

Average Clustering Coefficient

The analysis of the time series of the average clustering coefficient yields a similar result. By the PELT algorithm, illustrated in Figure 4a, six change points are identified. Three of them are very close to real events at the end of 2001. By specifying a searched number of change points, shown in Figure 4b, a total of five real events can be identified with sufficient precision. Also in this case, two further events in July 2001 and January 2002 are located near change points.

Recommendation for investigators

When applying the method presented here as part of an investigation, it is recommended that e-mails from periods before and after each change point are examined in detail, as these periods contain possible interesting content for investigations. This is why an exact determination of the real event is not mandatory. However, when narrowing down the time period for a detailed investigation of e-mails around a change point, it is necessary to balance between the least possible effort for the investigator and the thoroughness of the investigation.

Figure 5 visualizes the curve of the *density* of the dynamic graph. As in the previous plots, the real events are marked by dotted lines. In addition, in this representation, the ten determined change points are color-coded by their periods of investigation of lengths 5, 10 and 20 days. Some change points are close enough to each other that their observation periods may even overlap. Especially in the period from August to December 2001, some time periods already overlap for a threshold of 10 days before and after each change point, while this can be observed for 20 days in several cases. Because of these overlaps, there is no need to examine e-mail communications of, say, 20 days

[1] To identify ten change points, the time series must be divided into eleven segments as the end of the last segment is not considered a change point

before and after each change point. In this example, if we compare the proportion of days in the total time horizon and the share of e-mails that must be examined by the corresponding observation periods, the proportions are as shown in Table 4. This shows that in the example, 30 % of the real events are already covered from an examination of only 12 % of the entire time horizon. These may be periods of particularly intensive communication, during which an above-average number of e-mails must be examined. Still, however, for 30 % of real events only 20 % of e-mails must be examined. This speeds up the investigation and can lead to more evidence being seized more quickly. These can speed up the entire investigation process.

Table 4: Proportion of time periods and e-mails to be investigated in the total observation horizon as a percentage of the observation period per change point

Period under investigation	share time period	share e-mails	Precision
+/- 5 Days	12 %	20 %	30 %
+/- 10 Days	21 %	37 %	40 %
+/- 20 Days	38 %	59 %	60 %

The decisive added value of this approach for investigators thus lies in the identification of change points that mark important structural shifts in a communication network. By prioritizing the investigation of e-mails around the corresponding points in time, scarce capacities can thus be used efficiently.

5 Conclusion

We propose an algorithm for automatic detection of important events in criminal cases using only e-mail metadata. The method is based on a graph representation of the data and an automated detection of change points on time-series of graph metrics. Using the Enron data set, the case study shows that structural anomalies in a dynamic graph are indicative of real events. If an investigator is provided with the identified change points during the exploration of an e-mail data set, she can examine the e-mail communication around the identified time periods in detail. This decisively narrows down the volume of e-mails to be examined and capacities are freed up. Since the method developed here is only tested on a single data set as part of the case study, no general statement can be made about the potential of this method in police investigations. This requires testing by police investigators and testing on additional e-mail data sets. It should also be examined how the method behaves on larger or smaller data sets. In addition, the comparison of subjectively selected, real events and automatically determined change points does not represent a reliable form of evaluation that allows a statement about the applicability of this method for the unsupervised exploration of e-mail data.

Regardless of the automatically determined change points, insights such as time periods of special interest for further investigations can also be drawn from the visualization of the time-series of graph metrics. For this purpose, the investigation of further graph metrics can be part of future research, in particular graph metrics on node level. Also different cost function in the CPD can be considered. Due to the unsupervised nature of the problem it may be useful to implement an ensemble of parameter variations in practical application.

References

1. Bavarian State Ministry of the Interior, f.S., Integration: Betrug im Gesundheitswesen (2018)
2. SueddeutscheZeitung: Millionenbetrug bei pflegediensten in münchen und augsburg. (2021)
3. Federal Ministry of Education and Research: Bekämpfung von Abrechnungsbetrug und Korruption im Gesundheitswesen (Kriminelle Netzwerke) (2021)
4. Cohen, W.: Enron email dataset (2015)
5. Stolfo, S.J., Hershkop, S., Wang, K., Nimeskern, O., Hu, C.W.: Behavior profiling of email. In Chen, H., ed.: Intelligence and security informatics. Volume 2665 of Lecture Notes in Computer Science. Springer, Berlin and Heidelberg (2003) 74–90
6. Stolfo, S.J., Li, W.J., Hershkop, S., Wang, K., Hu, C.W., Nimeskern, O.: Detecting viral propagations using email behavior profiles. ACM transactions on internet technology (TOIT) (2004) 128–132
7. Stolfo, S.J., Hershkop, S.: Email mining toolkit supporting law enforcement forensic analyses. In: Proceedings of the 2005 national conference on Digital government research. (2005) 221–222
8. Hadjidj, R., Debbabi, M., Lounis, H., Iqbal, F., Szporer, A., Benredjem, D.: Towards an integrated e-mail forensic analysis framework. Digital Investigation **5**(3-4) (2009) 124–137
9. Meng, F., Wu, S., Yang, J., Yu, G.: Research of an e-mail forensic and analysis system based on visualization. In: 2009 Asia-Pacific Conference on Computational Intelligence and Industrial Applications, Piscataway, NJ, IEEE (2009) 281–284

10. Sobiya R. Khan, Smita M. Nirkhi, R. V. Dharaskar: E-mail data analysis for application to cyber forensic investigation using data mining. IJAIS Proceedings on 2nd National Conference on Innovative Paradigms in Engineering and Technology (NCIPET 2013) **NCIPET**(3) (2013) 1–4
11. Haggerty, J., Haggerty, S., Taylor, M.: Forensic triage of email network narratives through visualisation. Information Management & Computer Security **22**(4) (2014) 358–370
12. Stadlinger, J., Dewald, A.: A forensic email analysis tool using dynamic visualization. Journal of Digital Forensics, Security and Law (2017)
13. Kossinets, G., Watts, D.J.: Empirical analysis of an evolving social network. Science (New York, N.Y.) **311**(5757) (2006) 88–90
14. Kondor, D., Csabai, I., Szüle, J., Pósfai, M., Vattay, G.: Inferring the interplay between network structure and market effects in bitcoin. New Journal of Physics **16**(12) (2014) 125003
15. Romero, D.M., Uzzi, B., Kleinberg, J.: Social networks under stress. (2016)
16. Miller, H., Mokryn, O.: Size agnostic change point detection framework for evolving networks
17. Peel, L., Clauset, A.: Detecting change points in the large-scale structure of evolving networks. Proc. of the 29th International Conference on Artificial Intelligence (AAAI) (2015)
18. Huang, S., Hitti, Y., Rabusseau, G., Rabbany, R.: Laplacian change point detection for dynamic graphs. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, New York, NY, USA, ACM (2020) 349–358
19. Zhu, T., Li, P., Yu, L., Chen, K., Chen, Y.: Change point detection in dynamic networks based on community identification. IEEE Transactions on Network Science and Engineering **7**(3) (2020) 2067–2077
20. Kendrick, L., Musial, K., Gabrys, B.: Change point detection in social networks—critical review with experiments. Computer Science Review **29** (2018) 1–13
21. Truong, C., Oudre, L., Vayatis, N.: Selective review of offline change point detection methods. Signal Processing **167**(4) (2020) 107299
22. Bellman, R.: On a routing problem. Quarterly of Applied Mathematics **16**(1) (1958) 87–90
23. Killick, R., Fearnhead, P., Eckley, I.A.: Optimal detection of changepoints with a linear computational cost. Journal of the American Statistical Association **107**(500) (2012) 1590–1598
24. Frentz, C.: Enron: Chronik einer Rekord-Pleite. Manager Magazin **2003** (2003)
25. Linder, D.: The enron trial: A chronology (2022)

Machine Learning Models in Industrial Blockchain, Attacks and Contribution

Fatemeh Ghovanlooy Ghajar[1], Axel Sikora[1], Jan Stodt[2], Christoph Reich[2]

[1] Institute of Reliable Embedded Systems and Communication Electronics (ivESK), Offenburg University
{Fatemeh.Ghovanlooy,Axel.Sikora}@hs-offenburg.de
[2] Institute for Data Science, Cloud Computing and IT Security – Furtwangen University
{Jan.Stodt,Christoph.Reich}@hs-furtwangen.de

Abstract. The importance of machine learning has been increasing dramatically for years. From assistance systems to production optimisation to support the health sector, almost every area of daily life and industry comes into contact with machine learning. Besides all the benefits that ML brings, the lack of transparency and the difficulty in creating traceability pose major risks. While there are solutions that make the training of machine learning models more transparent, traceability is still a major challenge. Ensuring the identity of a model is another challenge. Unnoticed modification of a model is also a danger when using ML. One solution is to create an ML birth certificate and an ML family tree secured by blockchain technology. Important information about training and changes to the model through retraining can be stored in a blockchain and accessed by any user to create more security and traceability about an ML model.

Keywords: Machine learning; Blockchain; Traceability; Security

1 Introduction

Machine learning has successfully found its way into almost every area of everyday life. ML makes our everyday lives easier with the help of assistance systems, makes production processes more efficient, detects errors before they occur and helps to develop medicines faster than ever possible by humans. However, it must also be said that at the beginning of the increased use of ML, many people were dazzled by its capabilities and refrained from using documentation methods that were already established in other areas. And even if documentation exists, it usually has gaps that make it difficult or impossible to follow up at a later stage.

For a number of reasons, organizations wanted to stay ahead of the curve in consistently protecting ML assets. In recent years, huge firms are investing directly in machine learning for the first time. Second, standard-setting organizations such as ISO are proposing certification guidelines to assess the security of machine learning (ML) systems [1], whose recommendations have historically been sought by the industry. The European Union has even developed a detailed checklist to assess the dependability of ML systems [2]. Lastly, machine learning is rapidly becoming a company's primary value proposition.

The lack of traceability not only makes verification by external auditors difficult, but also internal auditing. Even if there is documentation of the process, it can be very difficult to find a source of error hidden in poorly documented preprocessing of the training data. One problem that is noticeable when looking at the documentation of preprocessing is that it usually only talks about using "a subset of dataset X". Which subset it is, is usually not documented. One technology that is particularly suitable for traceability is the blockchain. Information that is recorded in the blockchain can no longer be changed, and updates to the information can be traced. The decentralized storage of data also ensures that the stored data is still available even if a local copy of the data is lost. To address the traceability problems that exist in machine learning in the area of documentation, this paper introduces two concepts: the ML birth certificates and the ML family tree. Both concepts aim to clarify the ancestry of the models and to document decisions in the area of the model's origin clearly and semi-automatically.

Of course, the security and privacy problems for data (transactions) exchange and transmission in such a novel network environment are taken into account in the architecture of blockchain [3]. Several recent studies have indicated that (1) utilizing AI to improve the performance of a blockchain system [4–7] and (2) using blockchain to improve the security and privacy of data and model transfer on an AI system, [8–10] are two attractive research areas.

The study is structured as follows: part two describes the current state of the art, and section three discusses the security challenges associated with machine learning models. In the fourth part, we provide a strategy for maintaining the model's security, and in the last section, we reach a conclusion.

2 Related Work

McGraw et al. [11] identified 78 risks in ML security and highlighted the top 10 of them. Most of these attacks target the data used to train the model. Attacks also exist when the model is retrained: online learning, in which the model is trained while in use with new data, and attacks that occur during transfer learning, in which an existing model is trained (tuned) on new data. A detailed taxonomy of machine learning attacks was described by Pitropakis et al. [12]. In addition to the theory-focused reviews, Cheatham et al. [13] provide a detailed overview of the resulting consequences in the areas of individuals, organisations and society.

In this industrial use scenario, in an adversarial ML setting, transparency may need to testify across three modalities: that the ML platform is implemented securely, that the MLaaS fulfills basic security requirements, and that the ML model embedded in an edge device meets basic security objectives. Providing test harnesses to increase the security assurance of products developed on top of formal verification, such as [14] to address large-scale ML models used in industry, is an intriguing approach.

In order to capture some information about the creation of the ML model and its creation, Mitchell et al. [15] introduced the concept of ML Model Report Cards. However, it must be clearly stated that the aim of the model report cards is to create transparency in the area of fairness, not traceability. Although the model report cards (may) contain information about the creation, the level of detail to be achieved is not specified. In addition, evolution of models via online learning and especially in transfer learning, it is difficult to determine the ”base model” used in the process. Over several generations of transfer learning, the problem becomes even more acute; in retrospect, traceability is factually no longer given.

To address some of the problems mentioned, Arnold et al. [16] developed FactSheets 360, which collects more information in the area of traceability and versioning of the ML model. But even here, the quality/granularity of the collected information is not defined in detail. In examples of FactSheets 360, such vague statements as ”The test data consists of a subset of data set X” [17] are made. This coarse granularity does not allow outsiders or even internal staff to check the creation of the model afterwards. In addition, there is no automatic fact collection in the area of data pre-processing. In addition to the approaches mentioned, there are now best practices and tools in the field of ML DevOps [18] that attempt to solve the traceability problem mentioned above. For example, there are model registries that record the processing steps that have been carried out [19]. Examples of tools used for this are MLFlow[1] and Comet[2]. One problem with these tools, however, is the centralised storage approach of these solutions. This creates a large number of data silos, which makes it difficult to keep track of the data and leads to a lack of traceability when the centralised storage approach is switched off.

3 Issues

Issues of ML models exist in the areas of security against attacks, the insufficient documentation of the model creation, and tracing the lineage of ML models. In this section that follows covers the security challenges that arise throughout the construction of an ML system, as well as while the system is under assault and being readied for deployment.

3.1 Security Attacks

Attacks on ML models can sometimes have dramatic consequences [20]. Attacks on process optimization models can be used to cause financial damage by slowing down or stopping production. However, not every attack is only financially damaging. It is easy to imagine that human damage can be inflicted when decisions are made about people; making attacks on machine learning models only more critical. The following are the most significant assaults that may occur against machine learning models and software.

1. False data training model: a training model based on datasets with erroneous data. In the ML development phase, vulnerability might occur. The majority of ML developers are unaware that attackers may breach the repository of ML training datasets, poisoning the dataset [21].
2. Malicious code: typically, programmers don’t start from scratch when creating a model; instead, they utilize existing code. If an attacker uploads modified scripts to websites like library or sample code, he or she can simply manipulate the model and result. Vulnerabilities might occur at this level due to the absence of automated tools for secure developers as well as a transparency center for machine learning systems.

[1] https://mlflow.org/

[2] https://www.comet.com/site/

3. If an attacker constructs a model and utilizes it as spy software or to influence model output, they are acting as model programmers. The danger of an attacker posing as a machine learning (ML) developer exists for apps that small businesses without detection and monitoring security measures seek to utilize for minimal cost. This kind of machine learning software may be built to spy on and exploit client data.
4. Model poisoning: An attacker degrades the model performance in order to obtain a different model decision [22].
5. Manipulation of a running program and modification of its output, when the machine learning system is under assault, and the objective of the attack is something that is important to the business of the industry.

3.2 Insufficient Documentation of Model Creation

As mentioned in related work, there are approaches to improve the documentation of ML model creation and use. However, both of the approaches presented, model report cards and FactSheets 360, still have the problem that the documentation of individual model quality-determining steps (e.g., data preprocessing) are not automatically documented. Lack of information on how the data set is pre-processed makes it difficult or even impossible to check the processes at a later stage. However, this is very important in order to investigate errors in the model and to find the origin of the error. This could be due to faulty preprocessing or deliberate attack on the data.

3.3 Tracing Lineage of ML Model

Lack of information about how the data set was pre-processed makes it difficult or even impossible to check the processes later. Although there are approaches to versioning data in the area of data preprocessing, there is usually no traceability of how the data was processed, e.g. which methods, which parameters. However, this is very important to investigate errors in the model and find the origin of the error. This could be due to faulty pre-processing or a deliberate attack on the data.

4 Method

The proposed method to address the challenges comprises two parts. Create the birth certificate before adding the hash and signature to the distributed ledger. Due to the immutability of blockchain, it cannot be altered or controlled. However, it will be required to have the model on blockchain as a reference for determining the method's dependability.

4.1 First Step

To realize detailed traceability of the ML model, we introduce the idea of ML birth certificates as well as an ML family tree to trace the changes of the base ML model and the changes made. A graphical example can be seen in Fig. 1.

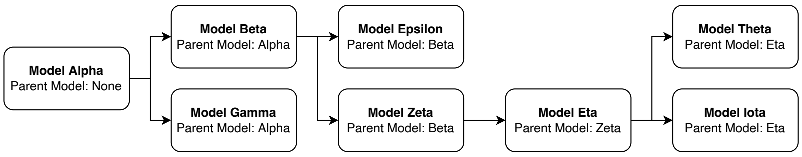


Fig. 1. ML Birth Certificates and ML Family Tree

ML birth certificates contain detailed information about the creation process, in order to make the creation process verifiable later in the life cycle of the model, if required. This information is similar to the concept of ML Model Report Cards [15] introduced by Mitchell et al. and the FactSheets 360 by Arnold et al. [16], our ML birth certificates contain a much finer granularity of information that is captured semi-automatically. For example, it logs exactly which preprocessing steps and which commands were used in which software version.

In order to make the descent of models from previous models (e.g., through transfer learning or online learning) clearly comprehensible, the concept of the ML family tree is introduced. Here, the ML birth certificate refers to the parent model. An example of the blockchain entries for the model birth certificates can be seen in Listing 1.1.

```
[
  {
    "record-type": "model",
    "model-name": "Model Alpha",
    "model-id": "8ad53dd3-4b02-4af5-a17f-e03d2fabee1c",
    "parent-model-name": "None",
    "parent-model-id": "None",
    "training-data-hash": "3cf33bd19e5001e9c151fe8127632e9...",
    "training-data-url": "ipfs://hl99h884pkragfmnlgxl",
    "payload": "exde4UOIQvV/f4bwZ59bhg=="
  },
  {
    "record-type": "model",
    "model-name": "Model Beta",
    "model-id": "402441d4-677f-4363-84aa-6f5a400a179cc",
    "parent-model-name": "Model Alpha",
    "parent-model-id": "8ad53dd3-4b02-4af5-a17f-e03d2fabee1c",
    "training-data-hash": "960f0bac5d1740c8ef0924442bc31ea8...",
    "training-data-url": "ipfs://9i0xd83ucroffgp739x4",
    "payload": "BNuEnBslKzpDk1nCwWys2A=="
  }
]
```

Listing 1.1. Example of blockchain blocks of the Model Birth Certificates

4.2 Second Step

To ensure traceability, the data used for the training is either stored as hash in the blockchain (for confidential data) or stored in IPFS [23] to ensure optimal traceability. To ensure about originality and address the concerns in Section 3.1, it is advised that the model be placed in a Docker container and stored it in IPFS. It also introduces a smart contract that verifies and signs the output based on the model's data stored in the blockchain. The process of data verification and sign by the smart contract can be seen in Fig. 2. An overview of the architecture can be seen in Fig. 3

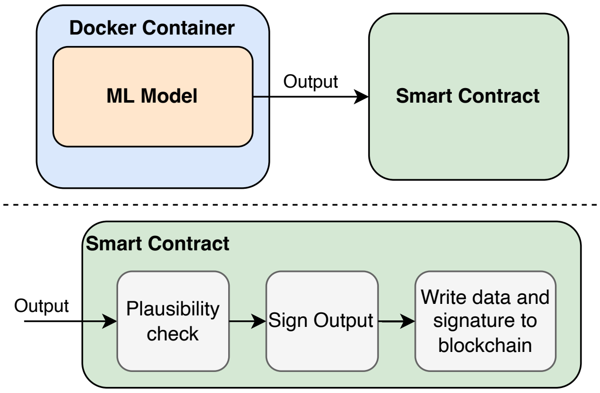


Fig. 2. Data verification and sign process of model output.

5 Conclusion

The vast majority of machine learning engineers and incident responders working in the sector do not lack the abilities essential to safeguard enterprise-grade ML systems from being attacked by malicious actors. Semi-automated data capture and blockchain technologies can help trace the lifecycle of ML models to mitigate the risks of ML model attacks and provide a more comprehensive trace of the development process. The blockchain and the concepts of ML birth certificates and ML family tree presented here also help to record the lineage of models in more detail. This means that it is always clear on which data or model basis a model was trained or retrained.

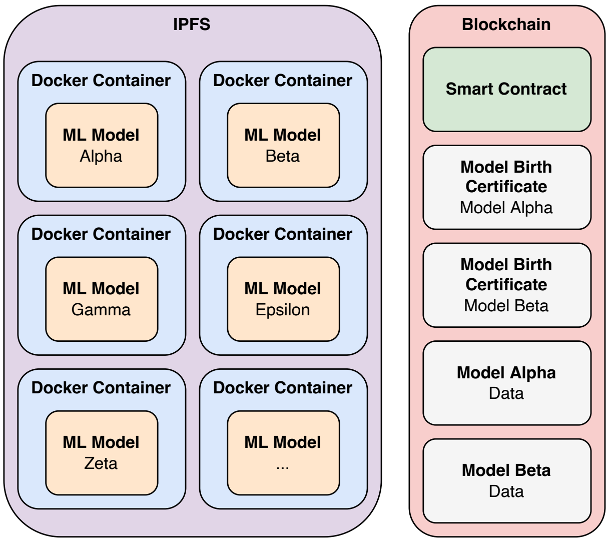


Fig. 3. Architecture.

References

1. ISO: ISO/IEC JTC 1/SC 42 - Artificial intelligence `https://www.iso.org/cms/render/live/en/sites/isoorg/contents/data/committee/67/94/6794475.html`.
2. Smuha, N.A.: The eu approach to ethics guidelines for trustworthy artificial intelligence. Computer Law Review International **20**(4) (2019) 97–106
3. Stodt, J., Schönle, D., Reich, C., Ghovanlooy Ghajar, F., Welte, D., Sikora, A.: Security audit of a blockchain-based industrial application platform. Algorithms **14**(4) (2021) 121
4. Marwala, T., Xing, B.: Blockchain and artificial intelligence. arXiv preprint arXiv:1802.04451 (2018)
5. Singh, S.K., Rathore, S., Park, J.H.: Blockiotintelligence: A blockchain-enabled intelligent iot architecture with artificial intelligence. Future Generation Computer Systems **110** (2020) 721–743
6. Ghovanlooy Ghajar, F., Sikora, A., Welte, D.: Schloss: Blockchain-based system architecture for secure industrial iot. Electronics **11**(10) (2022) 1629
7. Tagde, P., Tagde, S., Bhattacharya, T., Tagde, P., Chopra, H., Akter, R., Kaushik, D., Rahman, M., et al.: Blockchain and artificial intelligence technology in e-health. Environmental Science and Pollution Research **28**(38) (2021) 52810–52831
8. Lo, S.K., Liu, Y., Lu, Q., Wang, C., Xu, X., Paik, H.Y., Zhu, L.: Towards trustworthy ai: Blockchain-based architecture design for accountability and fairness of federated learning systems. IEEE Internet of Things Journal (2022)
9. Ghovanlooy Ghajar, F., Salimi Sratakhti, J., Sikora, A.: Sbtms: Scalable blockchain trust management system for vanet. Applied Sciences **11**(24) (2021) 11947
10. Lo, S.K., Liu, Y., Lu, Q., Wang, C., Xu, X., Paik, H.Y., Zhu, L.: Blockchain-based trustworthy federated learning architecture. arXiv preprint arXiv:2108.06912 (2021)
11. McGraw, G., Bonett, R., Shepardson, V., Figueroa, H.: The Top 10 Risks of Machine Learning Security. **53**(6) 57–61
12. Pitropakis, N., Panaousis, E., Giannetsos, T., Anastasiadis, E., Loukas, G.: A taxonomy and survey of attacks against machine learning. **34** 100199
13. Cheatham, B., Javanmardian, K., Samandari, H.: Confronting the risks of artificial intelligence. `http://ceros.mckinsey.com/unintended-consequences-desktop`.
14. Katz, G., Barrett, C., Dill, D.L., Julian, K., Kochenderfer, M.J.: Reluplex: An efficient smt solver for verifying deep neural networks. In: International conference on computer aided verification, Springer (2017) 97–117
15. Mitchell, M., Wu, S., Zaldivar, A., Barnes, P., Vasserman, L., Hutchinson, B., Spitzer, E., Raji, I.D., Gebru, T.: Model cards for model reporting. In: Proceedings of the conference on fairness, accountability, and transparency. (2019) 220–229
16. Arnold, M., Bellamy, R.K.E., Hind, M., Houde, S., Mehta, S., Mojsilovic, A., Nair, R., Ramamurthy, K.N., Reimer, D., Olteanu, A., Piorkowski, D., Tsay, J., Varshney, K.R.: FactSheets: Increasing Trust in AI Services through Supplier's Declarations of Conformity
17. IBM Research: AI FactSheets 360 `https://aifs360.mybluemix.net/examples/aifs360.mybluemix.net/examples`.
18. Rubasinghe, I., Meedeniya, D., Perera, I.: Traceability management with impact analysis in devops based software development. In: 2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI), IEEE (2018) 1956–1962
19. Kreuzberger, D., Kühl, N., Hirschl, S.: Machine learning operations (mlops): Overview, definition, and architecture. arXiv preprint arXiv:2205.02302 (2022)
20. Cheatham, B., Javanmardian, K., Samandari, H.: Unintended Consequences

21. Goldblum, M., Tsipras, D., Xie, C., Chen, X., Schwarzschild, A., Song, D., Madry, A., Li, B., Goldstein, T.: Dataset security for machine learning: Data poisoning, backdoor attacks, and defenses. IEEE Transactions on Pattern Analysis and Machine Intelligence (2022)
22. Panda, A., Mahloujifar, S., Bhagoji, A.N., Chakraborty, S., Mittal, P.: Sparsefed: Mitigating model poisoning attacks in federated learning with sparsification. In: International Conference on Artificial Intelligence and Statistics, PMLR (2022) 7587–7624
23. Benet, J.: Ipfs-content addressed, versioned, p2p file system. arXiv preprint arXiv:1407.3561 (2014)

Tackling Key Challenges of AI Development – Insights from an Industry-Academia Collaboration

Alexander Melde[1]*, Manav Madan[3]*, Paul Gavrikov[2], David Hoof[2], Astrid Laubenheimer[1], Janis Keuper[2], and Christoph Reich[3]

[1] Karlsruhe University of Applied Sciences
{alexander.melde, astrid.laubenheimer}@h-ka.de
[2] Offenburg University
{paul.gavrikov, janis.keuper}@hs-offenburg.de, dhoof@stud.hs-offenburg.de
[3] Furtwangen University
{manav.madan, christoph.reich}@hs-furtwangen.de

Abstract. Harnessing the overall benefits of the latest advancements in artificial intelligence (AI) requires the extensive collaboration of academia and industry. These collaborations promote innovation and growth while enforcing the practical usefulness of newer technologies in real life. The purpose of this article is to outline the challenges faced during cross-collaboration between academia and industry. These challenges are also inspected with the help of an ongoing project titled "Quality Assurance of Machine Learning Applications" (Q-AMeLiA), in which three universities cooperate with five industry partners to make the product risk of AI-based products visible. Further, we discuss the hurdles and the key challenges in machine learning (ML) technology transformation from academia to industry based on robustness, simplicity, and safety. These challenges are an outcome of the lack of common standards, metrics, and missing regulatory considerations when state-of-the-art (SOTA) technology is developed in academia. The use of biased datasets involves ethical concerns that might lead to unfair outcomes when the ML model is deployed in production. The advancement of AI in small and medium sized enterprises (SMEs) requires more in terms of common standardization of concepts rather than algorithm breakthroughs. In this paper, in addition to the general challenges, we also discuss domain specific barriers for five different domains i.e., object detection, hardware benchmarking, continual learning, action recognition, and industrial process automation, and highlight the steps necessary for successfully managing the cross-sectoral collaborations between academia and industry.

Keywords: Artificial Intelligence, Machine Learning Lifecycle, MLOps, Collaboration of Academia and Industry, Challenges in Action Recognition, Model Search

1 Introduction

In recent years, applications of AI in the industry have made significant gains. This highlights a novel trend in ML-based data-driven products which are quickly substituting their traditional counterparts. However, investing in AI product development without contemplating the exact deployment of these products for business value creation can lead to a variety of problems in future which can also be summed up as technical debt [1].

The complexity and vast amount of new technological advancements in AI can quickly become overwhelming for enterprises starting their own AI transition journey. These challenges can be solved with a successful and productive collaboration of universities and industrial partners. When universities and enterprises collaborate to solve complex tasks in AI, the motivational aspects are different for individuals. Enterprises provide real-life input and high-quality data, to refine more generic research into specific problem definitions whereas academia focuses on generating breakthroughs in research. Overall their preferences are not always aligned. This paper highlights some of these differences based on five different use cases.

Additionally, there have been various studies performed in the past outlining the challenges of ML development and deployment. However, most of these are based on conducted interviews with ML experts from different fields and lack comparability [2,3,4]. As ML problems can be highly unique it is only natural that not all challenges can be covered by these different studies.

In this work, first, challenges encountered in collaborations between industry and academia for AI projects will be presented. Following, the challenges related to each use case are highlighted, including ethical issues and explaining general conflicts of interests in this context. Finally, solutions to the most common challenges will be proposed.

* These authors contributed equally.

1.1 Related Work

Despite many different approaches proposed in the past for tackling issues for successful collaboration between the industry and academia, not many include AI as the center of the development.

Some of the studies and their results can be summarized as follows: Aykol et al. [5] mention that tackling challenges in applying ML for the development of complex material systems such as batteries requires successful collaboration between academia and industry. They also refer to the need for experimental battery data which is required for the further growth of research in academia. This is similar to the lack of industry-standard datasets that are observed on a large scale in multiple AI research fields. Additionally, Fursin et al. [6] mention the difficulties of co-designing efficient software and hardware which have emerged in recent times due to the growing technology transfer gap between academia and industry. The use case that is targeted in this work is autotuning, which focuses on automatically exploring optimization spaces such that the efficiency of computer systems could be improved. The key challenges that are highlighted in this work are the lack of a common experimental framework and the lack of practical knowledge exchange between academia and industry. Their viewpoint targets the point that practitioners spend more time adapting already developed tools for their use case of multi-objective autotuning rather than innovating solutions for new problems such as more complex search spaces. They develop a framework and present a study in which researchers and students are taught to use reusable customizable workflows. In their framework, the outputs such as research artifacts can be shared easily with a unified API to further encourage reusability. Furthermore, Garousi et al. [7] target an overall perspective on industry-academia collaboration for the successful development of software engineering. The authors mention three main challenges that should be tackled. These three challenges are (1) conserving and staying on a common goal; (2) engaging in mutual understanding (between all partners) and teamwork, and (3) identifying managerial bottlenecks such that the whole project could be tracked.

None of the studies mentioned above have yet provided a comprehensive overview of the challenges of AI-based products, especially describing the division based on different domains and their dependency on key challenges.

1.2 The Q-AMeLiA project

The motivation behind this paper originates in the project "Quality Assurance for Machine Learning Applications" (Q-AMeLiA). The consortium aims to support small and medium enterprises (SMEs) in the special machine learning software development life cycle (ML-SDLC) by developing tools to evaluate data quality indicators (e.g. in terms of representative coverage of the feature space), as well as to evaluate the quality of the learned AI model achieved in the learning process. Further, to serve the business needs of the industry partners, the consortium aims to reduce and make measurable the product risk of AI-based products, enabling manufacturers to assure a quantified performance of their products (with respect to AI decisions) to their customers (see Fig. 1).

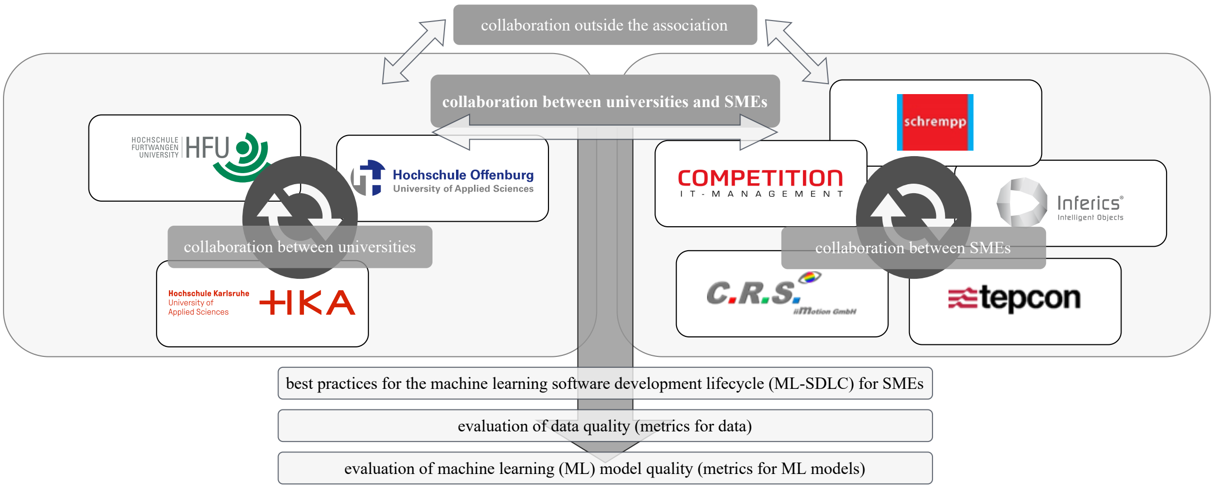

collaboration outside the association
collaboration between universities and SMEs
HOCHSCHULE FURTWANGEN UNIVERSITY HFU
Hochschule Offenburg University of Applied Sciences
collaboration between universities
Hochschule Karlsruhe University of Applied Sciences HKA
schrempp
COMPETITION IT-MANAGEMENT
Inferics Intelligent Objects
collaboration between SMEs
C.R.S. iiMotion GmbH
tepcon
best practices for the machine learning software development lifecycle (ML-SDLC) for SMEs
evaluation of data quality (metrics for data)
evaluation of machine learning (ML) model quality (metrics for ML models)

Fig. 1. Consortium: 5 SMEs collaborate with 3 universities.

To achieve these goals, three research topics have been defined. First, universally applicable and domain-independent best practices for the ML-SDLC should be developed by combining research results with use cases and lessons learned from industry. Results regarding this question are shown in this paper and previous work [8]. Second, methods, metrics and tools for qualitative and quantitative assessment of data quality should be developed. An architecture to quantify the risk of AI models has been presented in [9]. Third, these results should be adapted exemplary for a set of real-world use cases provided by the industry partners.

2 Challenges of AI Development

Collaborating on AI products comes with its challenges. Despite the differences in interest, academia and industry have competencies that can enable an interdisciplinary approach to innovation in AI. Indeed, studies have shown that such collaborations have increased in the recent past [10]. This section aims to provide an overview of common challenges faced when collaborating for AI projects.

First, *domain independent* organizational challenges are outlined. These are sometimes non-technical but play an important role in value generation from ML algorithms in the real world. Next, we outline challenges based on five individual domains which we refer to as *domain dependent* challenges.

2.1 Domain-Independent AI Challenges

During this and previous collaborations, the partners of the Q-AMeLiA project encountered common challenges independent from the specific projects domains. In this section, a selection of these challenges will be presented based on both the perceived importance and coverage in related publications [11,12,13,14].

Varying Mindsets In general, academia and industry have disjoint paths to AI. While academia focuses on research and development of new algorithms, seeking to find generalizing solutions for common challenges, industrial interest is tailored to their often highly-specific problems.

There is a vast amount of hype around AI, which in turns causes unrealistic expectations from enterprises. The involvement of multiple stakeholders might further complicate the product as joining business metrics to AI is not a straightforward task. The varying mindsets could result in a conflict of interests and might also lead to compromise on ethical aspects of AI.

Ownership of Intellectual Property From a ML perspective, a product consists of a combination of training data, an algorithm, a model trained on the data, and the output it generates. These all could come from different sources and could be owned by different stakeholders, leading to conflicts regarding the ownership of the final product as it cannot be created without the individual components. Collaborations should therefore discuss intellectual property (IP) early-on and should decide how the IP and usage rights are apportioned throughout the consortium. An example may be whether an enterprise who provides a dataset should be awarded the usage rights of the model trained with the respective data.

Data Quality and Quantity A sufficient amount of high quality training data can be considered a major factor for the success of AI projects and projects fail due to low data quality. Enterprises tend to overlook this aspect due to a target-driven approach, which focuses on generating business value with ML without really understanding the requirements of these approaches. A better approach would be to put the data and the model first [13]. Disregarding this advice may lead to expensive iterations of data-ingestion and curation. AI researcher Andrew Ng proposed that "the focus has to shift from big data to good data" [15]. Hence for any collaboration, the dataset and use case are the first points on which consensus should be reached before any effort is applied.

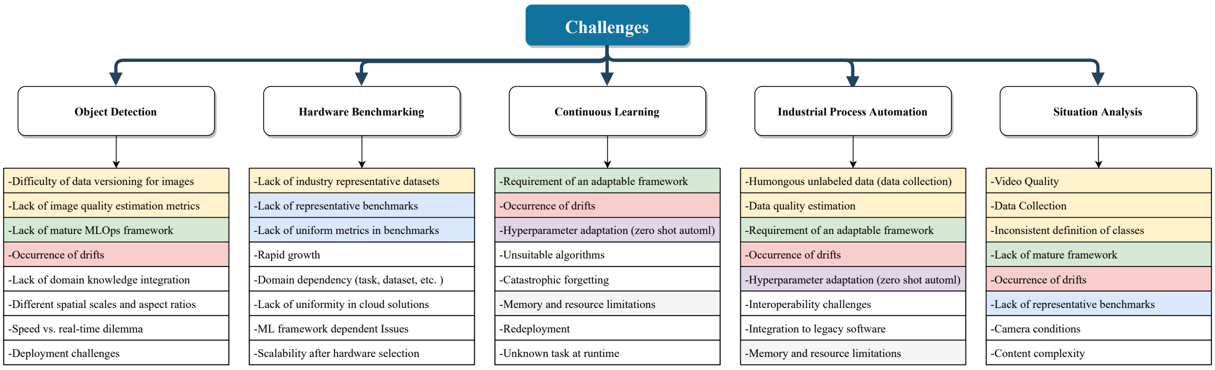
Dataset Licensing Industry-oriented research requires tailored datasets, as tackling industrial problems based on research-oriented datasets is difficult due to the domain shift. Unfortunately, such datasets are rare because companies are often interested in keeping their datasets closed source, to prevent various privacy issues and to allow for a competitive advantage. To create a representative dataset, the full complexity of a given context should be covered, however, it is impossible to represent the full range of external influence and it is difficult to define where to abstract and simplify. In previous work, data is often recorded in laboratory conditions that are not suitable for real life applications [16], but sometimes even this is a tedious process, as for example recordings of people can be difficult to obtain for privacy reasons. Furthermore, some important events may only occur rarely and therefore be underrepresented in collected datasets. This problem can be seen in anomaly detection systems: Real video recordings capturing all possible events may not be feasible in practice, and on the other hand simulation-based approaches are not guaranteed to behave naturally.

Overchoice in Source Models Transfer learning exploits knowledge learned by a model on one problem to help solve a different but related problem. For this, a model pretrained on a large dataset is fine-tuned on a significantly smaller dataset, considerably cutting data acquisition costs. However, there is a variety of pretrained ML models available for various domains, making it difficult for experts to choose the optimal fit for a particular problem. Furthermore, the optimal model may be different for each problem due to requirements such as latency, performance, and license considerations.

Integration in Legacy Systems Enterprises typically already operate and maintain software solutions. New ML models should then be able to communicate with these legacy systems and the effort required to integrate ML models into them should not be underestimated. Due to fast advancements in ML, there is a requirement for greater interoperability and automation of ML solutions compared to the software stack already in use.

Further Challenges In addition to the challenges faced in collaboration, there are additional barriers that need to be tackled when applying AI to real-life product, e.g. minimizing bias (see section 3), handling legal requirements like copyright and GDPR and choosing the right deployment infrastructure (regarding compute architectures and platforms and energy consumption). In addition, models should be as efficient as possible, explainable (providing the possibility to explore the reasoning behind AI decisions) and sometimes transferable, e.g. from cloud to edge [12].

2.2 Domain-Dependent Challenges



Object Detection	Hardware Benchmarking	Continuous Learning	Industrial Process Automation	Situation Analysis
-Difficulty of data versioning for images	-Lack of industry representative datasets	-Requirement of an adaptable framework	-Humongous unlabeled data (data collection)	-Video Quality
-Lack of image quality estimation metrics	-Lack of representative benchmarks	-Occurrence of drifts	-Data quality estimation	-Data Collection
-Lack of mature MLOps framework	-Lack of uniform metrics in benchmarks	-Hyperparameter adaptation (zero shot automl)	-Requirement of an adaptable framework	-Inconsistent definition of classes
-Occurrence of drifts	-Rapid growth	-Unsuitable algorithms	-Occurrence of drifts	-Lack of mature framework
-Lack of domain knowledge integration	-Domain dependency (task, dataset, etc.)	-Catastrophic forgetting	-Hyperparameter adaptation (zero shot automl)	-Occurrence of drifts
-Different spatial scales and aspect ratios	-Lack of uniformity in cloud solutions	-Memory and resource limitations	-Interoperability challenges	-Lack of representative benchmarks
-Speed vs. real-time dilemma	-ML framework dependent Issues	-Redeployment	-Integration to legacy software	-Camera conditions
-Deployment challenges	-Scalability after hardware selection	-Unknown task at runtime	-Memory and resource limitations	-Content complexity

Fig. 2. Overview of challenges for each domain.

In addition to the challenges mentioned in the previous section, some challenges only occur in particular domains. The domains discussed in this paper are chosen due to their relevance in the research community and the importance for our industry partners. A more complete list of challenges can be seen in Fig. 2.

The uniqueness of such challenges arises from either the kind of data required or the use case itself. A total of five different domains of industrial use cases are targeted in the Q-AMeLiA project and are introduced in this section.

Object Detection using ML Q-AMeLiA collaboration partner *C.R.S. iiMotion GmbH* is providing solutions and services in the field of image and video processing. Their current research interest is the construction of an automated object detection pipeline for anomaly detection in optical paths of camera systems.

Anomaly detection is a prominent use case where the identification of defects or frauds is widely implemented in the industry [17]. The aim is to build an automated ML workflow utilizing machine learning operations (MLOps) such that the delivery time of the new model for production can be minimized [8]. This involves building an ML workflow automatically with steps such as automated data quality assessment and model testing.

Hardware Benchmarking for ML Software and hardware solutions for ML are growing at a rapid pace. Unlike other types of computational workloads, ML based workloads vary in compute intensity. A ML workflow is typically divided into two parts: training and inference.

In the training phase, the algorithm is iteratively updated till some predefined metric is reached and hence a lot of resources are required for this process. The use of resource-intensive hardware like graphics processing units (GPUs) and machines accelerated with tensor processing units (TPUs) is crucial for training models faster.

In the inference phase the trained and deployed model is used for making predictions. This is a less resource-intensive task, enabling the deployment to embedded devices with restricted resources.

Choosing the correct machine learning hardware for training is a complicated process. It is known that different workloads might require different specialized hardware [18,19]. Evaluating the suitability of hardware is the Q-AMeLiA use case for the company *competition it-management GmbH* that offers IT consulting services, project management, and support for setting up hardware and software Infrastructure.

Continuous Learning under Drift Traditional ML workloads presume that the data is static and will remain constant also in the production, which is not necessarily the case. The need for Continuous Learning (CL) stems from the fact that data never remains static and neither does the environment. Due to the various drift scenarios, such as data drift, concept drift, and model drift, CL is required to address these challenges and update the workload accordingly. The notion of CL is relatively new in the field of ML. Nevertheless, it is well known that neural networks suffer from the fundamental problem of forgetting what they have learned in the past when trained on new data. This is referred to as catastrophic forgetting. This problem makes the continuous improvement of a static ML workload a difficult task [20]. This is the use case for the company *tepcon GmbH* who encountered these challenges while providing machine learning based digitization solutions for predictive maintenance of industrial machinery.

Industrial Process Automation Regarding industrial process automation, Q-AMeLiA collaboration partner *schrempp edv GmbH* explores intelligent user interfaces (UI) that can predict and provide shortcuts for user interactions from the perspective of enterprise resource planning systems (ERP). ERPs help to track and store different processes and transactions in an organization. However, customer needs may be highly specific and require manual optimization of the UI to speed-up common routines. The rise of data analytics and big data, allows for such systems to become more intelligent such that they can automatically adjust the UI [21] for each individual user. The company collects anonymized telemetry data that it then uses to train predictive ML models. This allows customer-tailored UIs at a fraction of the cost of manual adjustments that can often exceed customers expectations.

Situation Analysis in Health Care Health care can benefit greatly from AI-assisted solutions, as demonstrated by our industry partner *Inferics GmbH*, which provides an AI platform used for access control systems based on face recognition and situation analysis for assisted living based on 3D poses. Detecting swiftly when, for example, elderly people are in dangerous situations and need assistance can be lifesaving, however identifying human posture and activity can be difficult to implement in real use cases for a variety of reasons, which are explained and grouped in more detail below.

For situation analysis, video data is needed for training. It is possible to record this data using varying camera conditions: the camera position influences the scale of an action and the viewpoint [16], possibly leading to (partial) occlusion. In the wild, lighting, shadows and internal camera parameters will vary. In some scenarios, the camera will even move. Varying video quality is encountered in the form of different resolutions, framerates, compression artefacts and blur. In addition, the complexity of the content filmed is in itself high, as a video may contain multiple activities performed by multiple actors either subsequently or in parallel. Every actor has it's own anthropometric appearance [16], just as the background scenes are varying and might distract from the activity. In addition, weather conditions and seasons might change how entire scenes look and how activities are performed. Any action can have varying temporal and spatial boundings, leading to videos (or bounding boxes) that have to be both cutted to time frames and cropped to a certain image area to represent only one action at a time.

Another challenge is the variety in definitions of situations, actions, activities, interactions, events and tasks in previous publications and datasets. These terms are closely related and sometimes used interchangeably. There are attempts to use activities as a general term that can be categorized into other terms based on their complexity, however this approach is not exhaustive [22]. Other work differentiates terms based on the number of humans involved [16]. Even if these definitions are used, an ambiguous term like “running” can be used to describe an action performed by either a single person or a group with diverse contexts and intentions. Some activities involve sub-activites, creating a hierarchy. Both intraclass variety and interclass similarity are high, as tasks can be completed using various methods and some activities are very similar to each other. Due to the potentially infinite number of activities, no dataset can cover the entire spectrum, and no naming convention has yet been established. Existing datasets suffer from label noise and poor label quality caused by online clickers with different cultural backgrounds.

Once the data is collected, high computational complexity is encountered, caused by multi-dimensional data and the involvement of temporal context. Depending on the implementation, high resolution 2d or 3d videos annotated with multi-dimensional labels and additional meta data are used. Unfortunately, the state of the art for situation analysis is far behind the one in e.g. image classification. No de-facto standard libraries, break-through datasets or benchmarks have been established yet. This scattering makes it difficult to compare previous research, whose approaches also vary widely, e.g. in the network architecture used, the temporal context strategy, the feature extraction (separate or end-to-end), the fusion strategy (if required), and the embedding used in the network.

To summarize, there is an insufficient amount of data for training action recognition models (given the high complexity of the task), datasets lack scene variety and their classes are biased, unbalanced and unreliably labeled. Due to the computational complexity and the difficulty in comparing previous approaches, efficient research can only be conducted if sufficient training time and computational resources are available [23].

3 Ethical Issues and Conflicts of Interest

Hagendorff and Meding [24] presented an important study on the ethical impact of the industry in ML research. They mainly set the focus on ethical concerns, conflicts of interest, innovation, and gender equality and conclude the following: (1) even with the growth of collaboration there are rarely any mentions of the conflict of interest between academia and industry, (2) research papers by industry often target trending ML topics earlier than academia on average and (3) the focus of research from the industry often falls short on social aspects such as gender diversity and gender equality.

In this work, first, the ethical concerns regarding AI will be considered, and then, the conflicts of interests between academia and industry will be described separately.

3.1 Ethical AI

Previous research has shown how biased AI software can harm society [25]. Such software might cause damage to under-represented groups in the source data, including discrimination based on race, gender, or otherwise unfair decision-making. With the rising number of AI products, the potentially negative impact of AI may be amplified, creating the need to integrate ethical considerations at the heart of product development. Unfortunately, addressing ethical concerns can result in a loss of business value for the industry and is therefore often skipped. Yet, the effectiveness of laws and regulations tackling ethical issues of AI is limited.

To tackle these problems, the European Commission introduced ethics guidelines for trustworthy artificial intelligence in 2019 [26]. These guidelines specify different principles for general AI ethics but are not an European law with any legal framework. Due to the rapid development of AI, standards are quickly outdated or not common yet. This gap in terms of legal consequences from different regulators leads to a need for self-regulation. The final goal should be to make AI responsible, transparent and accountable, such that the products based on AI could become sustainable in the future.

3.2 Conflict of Interest: Academia vs. Industry

Commercial product development and early-stage research have traditionally been regarded as the major difference between the industry and the academic world. Academic collaboration with industry takes many forms, but specifically for AI, collaboration efforts between the two have not been well coordinated [27]. There is a possibility of conflict between academia and industry in the development and monitoring of the use of AI systems.

In the last decade, companies have hired experienced researchers such as professors from prestigious universities for full-time positions to work on AI on a large scale. This implies that there is a strong motivation for creating applied solutions from the research however in the long run this would amplify the lack of experienced professionals for students at the universities. Among other reasons, researchers are opting out of academia due to the lack of resources. In spite of the above-mentioned challenges, the increased presence of industry could help stimulate the search for innovative solutions to real-world problems, thereby fueling further growth. On the other hand, some in the industry, in general, might strive to develop high-precision systems that may not be developed in accordance with ethical considerations [28]. For example, many of the initial datasets used in building commercial products by different industry giants had exhibited bias in some sense and were still used until publicly criticized [27]. There is also a lack of incentive for companies to act responsibly. They invest far more in AI than academia has done so far and might set their own (profit optimizing) goals higher. A solution to the above challenges lies in successful cooperation between both areas [27].

4 Paths forward

Over the past decade, large enterprises like Netflix, Google, and Amazon have benefited massively from the adoption of ML, but SMEs often fail in their efforts to adopt AI in their businesses. We believe the solution to this problem lies in successful cooperation between academia and industry. The challenges have been categorized in the previous sections as domain-dependent and domain-independent and an overview of domain-dependent challenges is presented in Section 2.2 and Fig. 2. In this section, we further discuss the predominant challenges of chapter 2 and propose solutions based on the collaboration of academia and industry.

4.1 Drift Adaptation

The environment into which a ML model is deployed to can change over time, leading to a change in the models input data and the degradation of a model's performance over time. To counter these so-called data drifts, the deployed model should be updated continuously, e.g. by building an automated MLOps pipeline. MLOps pipelines are closed-loop systems in which either a new model is trained or an old one is retrained, to counter the drop in performance indicated by a specified metric [8]. Drift adaptation should be an internal block of MLOps. In ML, drift is categorized into two parts: covariate drift and concept drift. The ML model is often developed in a static environment, tested, and finally deployed in a dynamic environment. This change can result in to drop in performance as the model might encounter data it has not seen previously or the relationship between the input data and predicted label might also change with time. The former case describes the situation in which there is a change in the distribution of incoming data. This is referred to as the covariate drift. The latter use-case describes the concept drift, where the patterns extracted from the data change over time. Methods to handle concept drift can be divided into two groups: Implicit (blinding) methods update the models in regular intervals, independently of the occurrence of the concept drift, whereas explicit methods update models only if a concept drift occurred [29]. There is a need to establish a general drift adaptation framework that targets all different types of drift.

4.2 Data Collection & Data Quality Estimation

Data is a key aspect of ML because models extract patterns based on the acquired data. Usually, data gets exposed to multiple sources of error at different stages of development. Challenges such as missing data, range miss-match and type miss-match pose specific threat to data integrity. These errors require integrity checks which are placed at different stages of an ML workflow. A standard process of handling data for ML starts with a storage engine (e.g. a warehouse or a data lake) which is then integrated into a feature store. The data from the feature store is then cleaned and transformed and finally integrated into the MLOps pipeline. This end-to-end process from curation to ML model is not standardized and organizations define their own tool stacks based on their requirements. Choosing the right data models and storage engines is challenging due to the high number of options [30][31]. Additionally, data quality is a subjective and a multi-dimensional concept, therefore good quality data cannot be defined with a single standard definition. The quality that differs from context to context is highly dependent on the targeted business process [32].

Another aspect of data preparation for using ML in the computer vision domain is *image labeling*. This step is immensely important as the label quality decides the quality of ML models in supervised learning scenarios. It is common to observe mistakes found in labels of popular benchmark datasets like ImageNet due to sloppiness or imprecise instructions [33]. Given their huge size, such datasets are heavily used in academia for benchmarking and pretraining, however due to labeling mistakes, precautions should be taken when research is applied in real-world settings, creating the need for better datasets. Building new datasets requires labeling which in turn requires domain expertise with multiple iterations for the construction of a high quality dataset. Often, the construction of new datasets is done by individuals with limited domain expertise resulting in non-representative datasets. The goal should be to fill this gap by industry-standard benchmark datasets created by industry experts. The challenges of data labeling are also highlighted in the work by Denton et.al. [34]. They discuss how the quality of labels can be improved with extended feedback loops between the annotators and the experts. Researchers should be incentivized to use more appropriate datasets based on industrial standards for scientific studies. There is a need to provide more encouragement to do more qualitative analysis with results based on the accuracy of the performed tasks. For example, the explainable AI could flow into the results with the interpretability of the trained model. Additional tests could also be carried out to check for fairness and energy efficiency. This is also illustrated by the model cards created in the work by [35]. Furthermore, the academia can help create better metrics for data quality estimation [9].

4.3 Automation and MLOps

Automation aims to boost productivity and make everyday tasks easier. For the ML domain, this is offered by machine learning operations (MLOps) [8]. The emphasis of MLOps is on reducing the friction between the different departments involved in producing business value through the use of ML. It advocates for teamwork and cutting down on waste in terms of the artifacts generated in the ML lifecycle. Many of the challenges with ML arise once the model has been trained and is put into use in production. For example, there are scenarios where the model's performance degrades with time as described in section 4.1 (drifts).

Without MLOps, local manual development workflows are data-scientist-centric. The model and the data both lie with the data scientists who do the training until they transfer the model to the operations department responsible for deployment. This static workflow has a lot of challenges, can introduce many different errors and vulnerabilities and deployed models cannot be easily replaced with new ones.

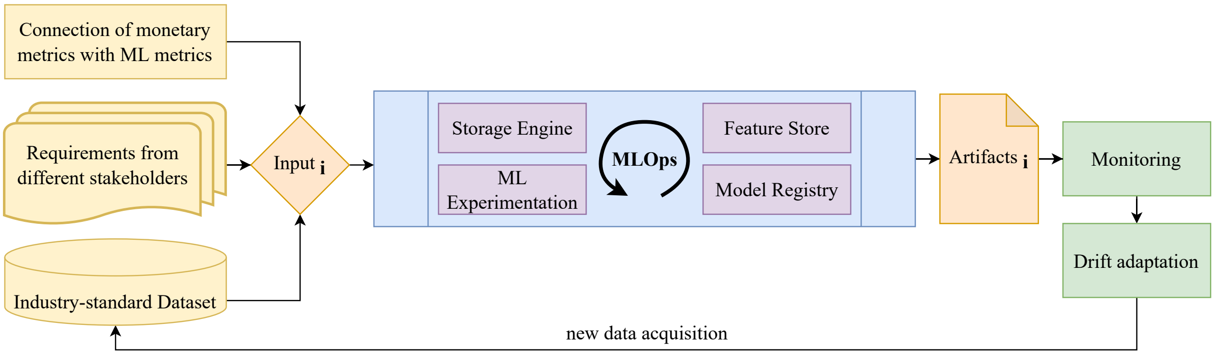


Fig. 3. Overview of an MLOps workflow.

MLOps provides a set of tested guidelines for reliably and automatically resolving such issues. An exemplar workflow is portrayed in Fig. 3. From the experience gathered through this project, it is important to understand that there is not one single correct MLOps workflow. There could be additional blocks for different processes in MLOps as this is not standardized for every domain. But the important part is to build the workflow with the option of integrating input from different stakeholders with industrial standard datasets to reduce the time spent on experimentation on the non-representative datasets. The workflow displayed also shows that the process of monitoring can be connected with an intelligent drift detection block to keep the quality of the ML model deployed high. There should also be additional support for data acquisition for the next iteration of the development. Through this, the ML models could be continuously adapted in the future automatically.

The MLOps approach to building a machine learning lifecycle focuses on automation and scalability. It provides the necessary stack of tools for rapid deployment of new models and helps in versioning and logging the success of deployments, creating a governance loop.

However, it can be challenging to know where to start and how to adopt MLOps practices. For businesses that are not performing the same operations manually, quantifying the benefits of MLOps is another challenge. For this reason, a survey of open-sourced tools was conducted, laying the foundation for adopting MLOps depending on the maturity of data science processes in a company [8]. The survey [8] serves as an introduction to various actors, roles, and tools involved in MLOps workflows. It provides a thorough comparison of supportive tools for the different phases to simplify the process of MLOps adoption. After MLOps has been integrated, a system for continuous learning that is capable of retrieving fresh training data, creating new features, developing new models, testing, and registering can easily be discovered.

4.4 Q-AMeLiA Search Engine

Due to the lack of high quantities of high-quality training data, ML engineers oftentimes opt for already trained models as a starting point for training a model on the new dataset. This process is referred to as transfer learning which typically yields considerable performance gains, especially for cases where data is scarce, however, in computer vision, it is specifically difficult as the data is of a high dimension and training from scratch can be an expensive process. For different domains, numerous pretrained models are available publicly, but choosing the best one is not straightforward.

As a solution to these problems regarding finding the right model, a search engine for pretrained computer vision models was developed. It helps to look up models by leveraging user-provided basic meta information such as the training task, visual domain, architecture, input size, or number of parameters. This first version of the search engine is available open-source under the following URL:

`https://github.com/Q-AMeLiA/searchengine`

Ongoing research is conducted to automate the process of model selection, and, a second version of the search engine is planned, which will suggest pretrained models based on a small, user-provided data subset, and a backend-sided evaluation of model suitability based on multiple metrics such as quality of the learned representations [36,37].

5 Conclusion

Usually, industry experts and researchers from academia have different backgrounds and objectives. With the ongoing project Q-AMeLiA it is showcased how bottlenecks in different domains of AI can be identified by merging the interests of academia and industry. The project has resulted in approaches for (1) linking theoretical research with creative solutions, (2) capturing challenges of individual domains, and (3) highlighting the bottlenecks in the automation of ML workflows for continual improvement in production. In the future the project will focus on creating further solutions with the help of the industrial expertise of the partners involved. In general, such collaborations encourage innovative solutions to real-world problems. When universities are linked to the business world, they are inundated with problems requiring solutions which in feedback provides new research directions to the universities. This is also helpful for the universities in the sense of curriculum design as the project highlighted the gap between applied machine learning (ML) and its research. The knowledge learned can be used to train students how to tackle the challenges of applying ML at scale in the real world. Students working on the project were also able to understand the relevance of both theoretical and practical approaches with the aid of such collaboration.

References

1. Sculley, D., Holt, G., Golovin, D., Davydov, E., Phillips, T., Ebner, D., Chaudhary, V., Young, M., Crespo, J.F., Dennison, D.: Hidden technical debt in machine learning systems. Advances in neural information processing systems **28** (2015)
2. Baier, L., Jöhren, F., Seebacher, S.: Challenges in the deployment and operation of machine learning in practice. In: ECIS. (2019)
3. Paleyes, A., Urma, R.G., Lawrence, N.D.: Challenges in deploying machine learning: a survey of case studies. arXiv preprint arXiv:2011.09926 (2020)
4. Lwakatare, L.E., Raj, A., Bosch, J., Olsson, H.H., Crnkovic, I.: A taxonomy of software engineering challenges for machine learning systems: An empirical investigation. In: International Conference on Agile Software Development, Springer, Cham (2019) 227–243
5. Aykol, M., Herring, P., Anapolsky, A.: Machine learning for continuous innovation in battery technologies. Nature Reviews Materials **5**(10) (2020) 725–727
6. Fursin, G., Lokhmotov, A., Savenko, D., Upton, E.: A collective knowledge workflow for collaborative research into multi-objective autotuning and machine learning techniques. arXiv preprint arXiv:1801.08024 (2018)
7. Garousi, V., Felderer, M., Fernandes, J.M., Pfahl, D., Mäntylä, M.V.: Industry-academia collaborations in software engineering: An empirical analysis of challenges, patterns and anti-patterns in research projects. Proceedings of the 21st International Conference on Evaluation and Assessment in Software Engineering (EASE) (2017)
8. Ruf, P., Madan, M., Reich, C., Ould-Abdeslam, D.: Demystifying mlops and presenting a recipe for the selection of open-source tools. Applied Sciences **11**(19) (2021)
9. Melde, A., Laubenheimer, A., Link, N., Schauer, C.: An architecture to quantify the risk of ai-models. Upper-Rhine Artificial Intelligence Symposium UR-AI 2021 (2021) 84–93
10. Michel-Schneider, U.: Challenges for University - Industry Collaboration - a Stakeholder View. Proceedings of Business and Management Conferences 12713398, International Institute of Social and Economic Sciences (October 2021)

11. Mikhaylov, S.J., Esteve, M., Campion, A.: Artificial intelligence for the public sector: opportunities and challenges of cross-sector collaboration. Philosophical transactions of the royal society a: mathematical, physical and engineering sciences **376**(2128) (2018) 20170357
12. German AI Association (KI-Bundesverband): Large European AI Models (LEAM) als Leuchtturmprojekt für Europa (Konzeptpapier). LEAM:AI (6 2022)
13. Nahar, N., Zhou, S., Lewis, G., Kästner, C.: Collaboration challenges in building ml-enabled systems: Communication, documentation, engineering, and process. Organization **1**(2) (2022) 3
14. Saghiri, A.M., Vahidipour, S.M., Jabbarpour, M.R., Sookhak, M., Forestiero, A.: A survey of artificial intelligence challenges: Analyzing the definitions, relationships, and evolutions. Applied Sciences **12**(8) (2022) 4054
15. Strickland, E.: Andrew ng: Unbiggen ai. IEEE Spectrum (2022)
16. Turaga, P., Chellappa, R., Subrahmanian, V.S., Udrea, O.: Machine recognition of human activities: A survey. IEEE Transactions on Circuits and Systems for Video Technology **18**(11) (2008) 1473–1488
17. Tao, X., Zhang, D., Ma, W., Liu, X., Xu, D.: Automatic metallic surface defect detection and recognition with convolutional neural networks. Applied Sciences **8**(9) (2018) 1575
18. Mattson, P., Cheng, C., Diamos, G., Coleman, C., Micikevicius, P., Patterson, D., Tang, H., Wei, G.Y., Bailis, P., Bittorf, V., Brooks, D., Chen, D., Dutta, D., Gupta, U., Hazelwood, K., Hock, A., Huang, X., Kang, D., Kanter, D., Kumar, N., Liao, J., Narayanan, D., Oguntebi, T., Pekhimenko, G., Pentecost, L., Janapa Reddi, V., Robie, T., St John, T., Wu, C.J., Xu, L., Young, C., Zaharia, M.: Mlperf training benchmark. In Dhillon, I., Papailiopoulos, D., Sze, V., eds.: Proceedings of Machine Learning and Systems. Volume 2. (2020) 336–349
19. Madan, M., Reich, C.: Comparison of benchmarks for machine learning cloud infrastructures. CLOUD COMPUTING 2021 (2021) 50
20. van de Ven, G.M., Tolias, A.S.: Three scenarios for continual learning. ArXiv **abs/1904.07734** (2019)
21. Tallón-Ballesteros, A.: The design of erp intelligent sales management system. Fuzzy Systems and Data Mining VI: Proceedings of FSDM 2020 **331**(2020) (2020) 413
22. Vrigkas, M., Nikou, C., Kakadiaris, I.A.: A review of human activity recognition methods. Frontiers in Robotics and AI **2** (2015)
23. Gowda, S.N., Rohrbach, M., Sevilla-Lara, L.: Smart frame selection for action recognition. In: Proceedings of the AAAI Conference on Artificial Intelligence. Volume 35/2. (2021) 1451–1459
24. Hagendorff, T., Meding, K.: Ethical considerations and statistical analysis of industry involvement in machine learning research. AI & SOCIETY (2021) 1–11
25. Giuffrida, I.: Liability for ai decision-making: some legal and ethical considerations. Fordham L. Rev. **88** (2019) 439
26. Larsson, S.: Ai in the eu: Ethical guidelines as a governance tool. In: The European Union and the Technology Shift. Springer (2021) 85–111
27. Horvitz, E.: One hundred year study on artificial intelligence (2016)
28. Lauer, D.: You cannot have ai ethics without ethics. AI and Ethics **1**(1) (2021) 21–25
29. Mehmood, H., Kostakos, P., Cortes, M., Anagnostopoulos, T., Pirttikangas, S., Gilman, E.: Concept drift adaptation techniques in distributed environment for real-world data streams. Smart Cities **4**(1) (2021) 349–371
30. Huyen, C.: Designing Machine Learning Systems. " O'Reilly Media, Inc." (2022)
31. Idreos, S., Callaghan, M.: Key-value storage engines. In: Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data. SIGMOD '20, New York, NY, USA, Association for Computing Machinery (2020) 2667–2672
32. Panahy, P.H.S., Sidi, F., Affendey, L.S., Jabar, M.A.: The impact of data quality dimensions on business process improvement. In: 2014 4th World Congress on Information and Communication Technologies (WICT 2014), IEEE (2014) 70–73
33. Vasudevan, V., Caine, B., Gontijo-Lopes, R., Fridovich-Keil, S., Roelofs, R.: When does dough become a bagel? analyzing the remaining mistakes on imagenet. arXiv preprint arXiv:2205.04596 (2022)
34. Denton, E., Díaz, M., Kivlichan, I., Prabhakaran, V., Rosen, R.: Whose ground truth? accounting for individual and collective identities underlying dataset annotation. arXiv preprint arXiv:2112.04554 (2021)
35. Mitchell, M., Wu, S., Zaldivar, A., Barnes, P., Vasserman, L., Hutchinson, B., Spitzer, E., Raji, I.D., Gebru, T.: Model cards for model reporting. In: Proceedings of the conference on fairness, accountability, and transparency. (2019) 220–229
36. Gavrikov, P., Keuper, J.: Cnn filter db: An empirical investigation of trained convolutional filters. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). (June 2022) 19066–19076
37. Gavrikov, P., Keuper, J.: Adversarial robustness through the lens of convolutional filters. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops. (June 2022) 139–147

Acknowledgements

We thank our industry partners for helpful comments and discussions. This work was funded by the Ministry of Science, Research and Arts of Baden-Württemberg (MWK) as part of the project "Quality Assurance of Machine Learning Applications" (Q-AMeLiA).

Chapter 3

MEDICAL TECHNOLOGY

Keynote Abstract: 3, Introducing Keynote Speaker Dr. Lars Mündermann

AI in medicine – overrated or groundbreaking?

Developments in the last decade have shown that technology is the most disruptive factor in healthcare. This trend is expected to continue in the upcoming decades as Health & Care is one of the sectors with highest levels of investment in new technologies, treatment options, and drugs. Medical treatment is expected to be supported by a range of diagnostic tools, and real-life data on treatment success rates may influence the outcome for patients. Patients may no longer rely on a single and local physician but may access platforms whenever they have a specific medical need. Health might no longer be defined by the absence of a disease but rather be seen in the more holistic context of a person's well-being embracing the concept of P4 medicine (predictive, preventive, personalized and participatory). This keynote will illustrate what health and medicine might look like in 2050, introduce research activities in the fields of Cognitive Surgery and Surgical Data Science, discuss potential for AI applications in areas such as digital monitoring, prevention and AI-assisted diagnostics, and allude to prerequisites and restrictions for (successfully) introducing AI in medicine.

Figure 3.1: Dr. Lars Mündermann (Karl Storz SE & Co. KG)

Breast cancer classification methods for augmented reality microscopes

Robin Heckenauer[1], Jonathan Weber[1], Cédric Wemmert[2],
Michel Hassenforder[1], Pierre-Alain Muller[1], and Germain Forestier[1]

[1] IRIMAS, Université de Haute-Alsace, France
name.surname@uha.fr
[2] ICube, Université de Strasbourg, France
wemmert@unistra.fr

Abstract. An Augmented Reality Microscope (ARM) displays additional information about the tissues being analyzed by a pathologist. The image analysis methods used by these microscopes must be robust to changes in magnification level in order to follow the displacements in the glass slides. In this paper, we propose to take advantage of features present in some key magnification levels that improve the results of other magnification levels. We propose a real-time method robust to changes in magnification levels by training deep neural networks on certain key levels of Whole Slide Images (WSI) and testing them on the levels of a microscope. We show that our approach outperforms or equals naive methods on a breast cancer dataset for the Inception-ResNet-v2 deep learning architecture.

Keywords: Real-time breast cancer classification; Digital pathology; Augmented reality microscope; Deep convolutional networks.

1 Introduction

1.1 Digital pathology

The introduction of slide scanners in the late 1990s enabled the advent of digital pathology [1]. Indeed, scanners have made it possible to digitize glass slides by creating high resolution tissue images (Whole Slide Image - WSI). An example of such images is shown in Figure 1.

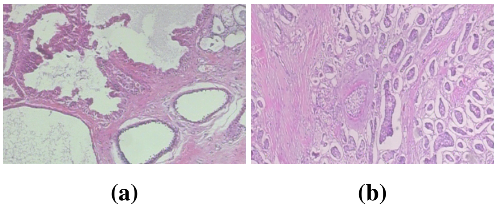


Fig. 1. Patches of benign and malignant tumors. Examples of benign tumor (a) and malignant tumor (b) as seen in a WSI at 40X magnification level, in the case of breast cancer (H&E stain).

The digitization of the slides has made it possible to solve many problems related to microscopic study. First, unlike glass slides, it is possible to preserve the tissues indefinitely and without deterioration. Consequently, the archiving of medical images makes it possible to create large databases around particular pathologies, which facilitates research work concerning these diseases. Then it becomes possible to share WSIs for peer review purposes to obtain multiple remote diagnoses of the same glass slide. Alternatively, microscope virtualization and WSI sharing can be used for training purposes. Indeed, to train specialists in pathology, it is no longer necessary to have glass slides, nor even a microscope available. A virtual microscope with WSI is sufficient, thus allowing easy access to education. These technological advances come with new challenges related to the size of WSIs. Because of their high resolution, WSIs weigh several gigabytes. Therefore, appropriate IT systems are needed to store, process and share WSI.

1.2 Digital pathology & AI

Following the encouraging results shown by artificial intelligence (AI) in the field of computer vision [2], these methods were quickly used on medical images. In the field of digital pathology, AIs have the potential to provide solutions in education, quality control, clinical diagnoses, image analysis [3–5]. However, faced with these new opportunities, the challenges are numerous: the lack of annotated data; high consumption of computing resources; the lack of transparency and explainability of AI predictions; ethical and legal issues.

Among the methods of AI, we find in particular deep learning methods. Deep neural networks were quickly applied to various pathologies, including breast cancer, which is the deadliest for women [6, 7].

Thus, studies have proposed methods for classifying tumor patches in the case of breast cancer [8–12]. These works have shown the superiority of deep learning approaches compared to machine learning approaches.

1.3 Virtual microscope & augmented reality

Driven by the evolution of digital pathology, virtual microscopes have appeared, thus making it possible to emulate the functions of an optical microscope on a computer [13, 14]. In addition to not needing a microscope, virtual microscopes allow tissues to be zoomed in and out without delay, unlike optical microscopes, which only offer a few levels of magnification and require focusing of the tissue observed at each level change.

Despite these advantages, in a context of redundant work such as glass slide diagnosis, using a virtual microscope is not suitable, because it leads to a loss of time during the digitization of the glass slide, then during the diagnosis phase [15]. An approach based on augmented reality tools integrated into microscopes (ARM), where additional information is displayed during the microscopic study seems better suited to help pathologists in their routine.

During a microscopic study carried out with an optical microscope, the pathologist analyzes a glass slide in search of structures of interest to establish a diagnosis. The glass slide is studied in particular at different levels of magnification. To change the level of magnification, optical microscopes have a limited number of objectives. We believe that the intermediate magnification levels unattainable with a light microscope are rich in features and can provide insight into tissue at the magnification levels accessible by a microscope.

Following previous work [16] on real-time object detection in WSIs, we propose to:

- Classify in real-time patches of benign and malignant tumors in the case of breast cancer using the Inception-ResNet-v2 [17] architecture.
- Use the intermediate magnification levels of a WSI to improve the performance of an ARM method that uses standard magnification levels of an optical microscope (20X, 40X, 100X).
- Compare three tumor prediction approaches at different magnification levels: Train networks at multiple levels; Train networks on each level; Train networks on an intermediate level.

For the sake of reproducibility, the experiments were performed on the public dataset BreakHis [18], and the source code is available here [3].

The paper is organized as follows. First, we introduce three methods to process images of multiple magnification levels in the case of an ARM application in section 2.1. Then, we introduce the public dataset BreakHis [18] and datasets created from BreakHis in section 2.2. We compare the methods presented in 2.1 in section 3, before providing an analysis on the experiments performed in section 4.

2 Materials and methods

2.1 Methods

In recent years, the rapid evolution of the computational capabilities of graphics processing units (GPUs) has largely contributed to the emergence of new approaches that improve the overall performance of deep learning methods, allowing them to be considered more seriously for real-time applications such as augmented reality microscopy [19, 20]. One of the challenges of using the augmented reality microscope (ARM) is the frequent changes in magnification level during diagnosis sessions. Indeed, changing the magnification alters the structures, shapes and borders of the observed tissues. To overcome this problem, we consider two naive approaches to develop a robust real-time method that performs well at different magnifications.

The first approach consists of training a single network on several magnifications, so that it generalizes its understanding of structures at several levels. This method requires only one model to be trained, however, the performance is largely related to the ability of the architecture to generalize over multiple magnifications. The second approach consists in training one network per level. In this case, several models are required, depending on the range of magnification levels studied. We then propose a third approach where we train a network on a level near to the level used for the test phase.

[3] https://github.com/RobinHCK/Breast-cancer-classification-methods-for-augmented-reality-microscopes

Among the architectures that have shown the best results on the task of classifying benign and malignant tumor patches in the case of breast cancer are: Inception-v3 [21], ResNet [22], GoogleNet [23], VGG [24]. Inception-v3 seems to show slightly better results than the other methods although the difference may be minimal depending on the dataset used. The Inception-ResNet-v2 architecture, which is an evolution of the Inception-v3 architecture, proposes to add residual connections to the Inception modules in order to speed up training and outperform Inception architectures without residual connections [17].

We compare the three approaches by studying the results of different Inception-ResNet-v2 models, over a wide range of magnification levels to be able to select the best networks over a given range.

2.2 Datasets

BreakHis The public dataset BreakHis [18] was used because it contains many images at different magnification levels. It consists of 7,909 patches stained in H&E (Hematoxylin and Eosin), size 700x460 pixels, at 40X, 100X, 200X and 400X magnification levels. This dataset is destined for the breast cancer classification task. 2,480 benign and 5,429 malignant tumor patches were collected from 82 patients.

Creation of intermediate magnification levels To cover a wider range of magnification levels, we interpolate patches of 20 intermediate magnification levels from the original dataset.

To do this, each original patch is cropped according to Equation 1 where *crop size* is the size of the patch to be kept when cropping, *patch size* is the size of the patch that will be used by the deep neural network, and *nearest level* is the nearest lower magnification level that exists in the patches of the original dataset.

$$crop\ size = \frac{patch\ size\ *\ nearest\ level}{desired\ level} \tag{1}$$

Next, we choose bicubic interpolation [25], which is a good compromise between interpolation quality and processing time [26], to size the patches to the dimension *patch size*. Thus, we obtain a dataset containing patches of size 350x230 pixels[4] for each of the 24 magnification levels in our range (from 20X to 400X). We create a last balanced dataset named allX from the 40X, 100X, 200X and 400X patches.

Finally, we apply data augmentation methods to the datasets to improve the diversity of the data and counteract the lack of data that is a recurrent problem when using medical data. Patches are randomly rotated, horizontally flipped, and vertically flipped.

3 Experimentations

In section 2.1, we presented three approaches that are robust to changes in magnification levels. To study and compare these approaches, we train Inception-ResNet-v2 networks on datasets built from BreakHis over a wide range of magnification levels.

Before starting training, we transfer weights from a pre-trained model on ImageNet [27] (ImageNet is the largest image database, commonly used for classification tasks, containing 14,197,122 images organized into 21,841 classes) to our model. We freeze the model weights, before re-training the last layers of the network on our data. This method is a standard procedure as reported in [28] for refining the training of models with little data. This saves training time and reduces overfitting.

We train the networks with the same parameters and hyperparameters to facilitate comparison of results: learning rate = 0.01; momentum = 0.9; epochs = 30; batch size = 64; optimizer = SGD; cost function = softmax; regularization = dropout. In addition, we use EarlyStopping to stop training when no improvement occurs on the validation loss in the last 15 epochs to avoid overfitting. To take full advantage of the data we have and to tackle overfitting, we perform a 5-fold (K=5) cross-validation for each magnification level. Patches are distributed in each fold respecting the organization given by BreakHis. Each fold is used to test once the results obtained on a network trained on all other folds. Thus, each patch of the dataset is classified once.

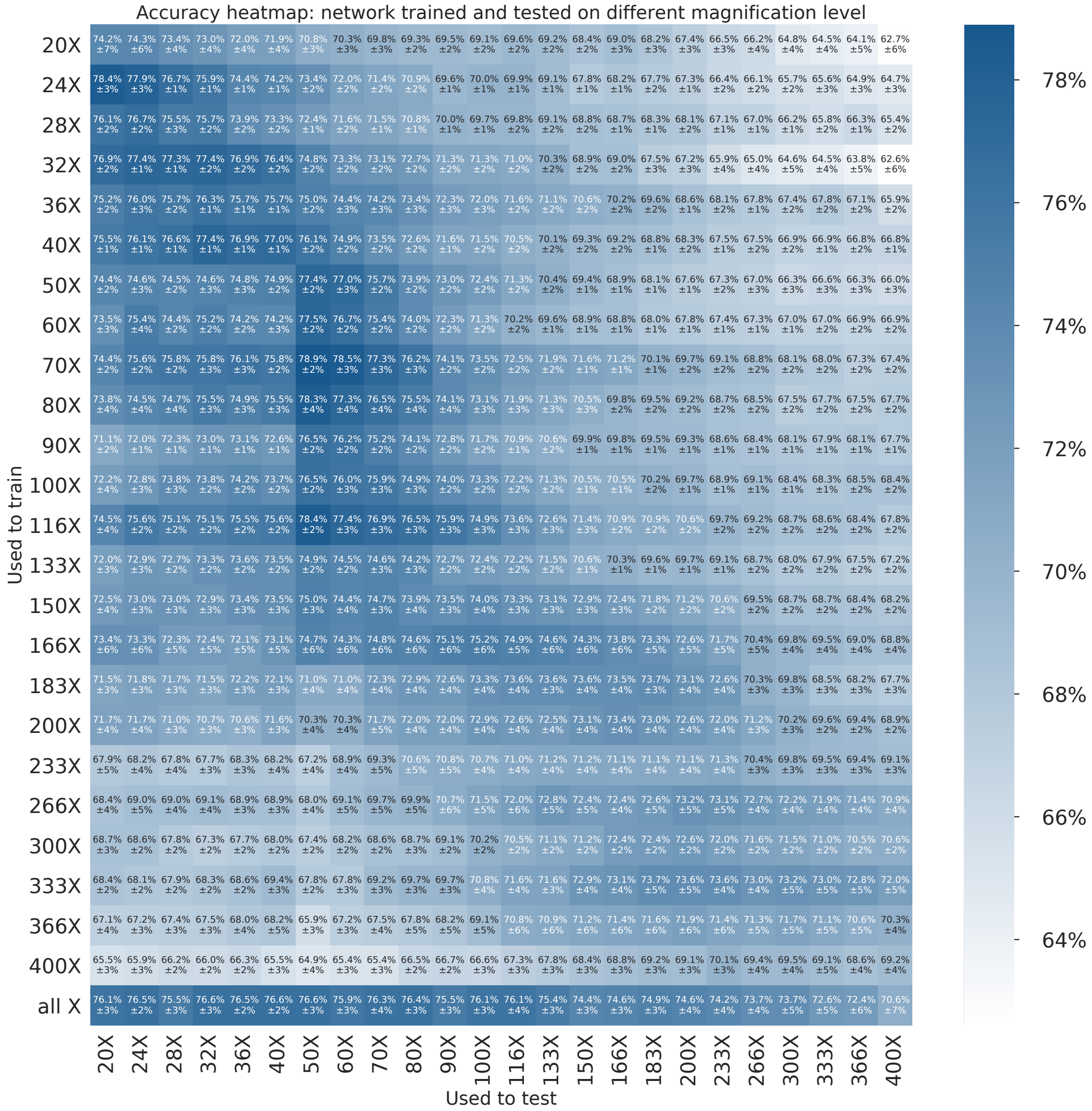
We perform all the computational processing on GPUs provided by the computing center of the University of Strasbourg. In total, at a rate of 10 minutes of training per network, it took about 1 day of cumulative computation to train the 125 networks (cross-validation K=5x25 datasets) that this experiment counts.

Finally, we test each model on all magnification levels. For this step, we choose to use the accuracy defined in Equation 2 as the evaluation metric since our classes are balanced.

[4] The size of the patches in the BreakHis dataset is 700x460 pixels. In order to obtain patches with a lower magnification level, the original patches must be cropped and resized. Thus we lose some of the context of the original patch

$$accuracy = \frac{well\ classified\ patches}{total\ number\ of\ patches} \tag{2}$$

We build a heatmap illustrated in Figure 2 from the accuracies obtained by the networks.



	20X	24X	28X	32X	36X	40X	50X	60X	70X	80X	90X	100X	116X	133X	150X	166X	183X	200X	233X	266X	300X	333X	366X	400X
20X	74.2% ±7%	74.3% ±6%	73.4% ±4%	73.0% ±4%	72.0% ±4%	71.9% ±4%	70.8% ±3%	70.3% ±3%	69.8% ±3%	69.3% ±2%	69.5% ±2%	69.1% ±2%	69.6% ±2%	69.2% ±2%	68.4% ±2%	69.0% ±3%	68.2% ±3%	67.4% ±3%	66.5% ±3%	66.2% ±4%	64.8% ±4%	64.5% ±4%	64.1% ±5%	62.7% ±6%
24X	78.4% ±3%	77.9% ±3%	76.7% ±1%	75.9% ±1%	74.4% ±1%	74.2% ±1%	73.4% ±2%	72.0% ±2%	71.4% ±2%	70.9% ±2%	69.6% ±1%	70.0% ±1%	69.9% ±1%	69.1% ±1%	67.8% ±1%	68.2% ±1%	67.7% ±2%	67.3% ±1%	66.4% ±2%	66.1% ±2%	65.7% ±2%	65.6% ±3%	64.9% ±3%	64.7% ±3%
28X	76.1% ±2%	76.7% ±2%	75.5% ±3%	75.7% ±2%	73.9% ±2%	73.3% ±2%	72.4% ±1%	71.6% ±2%	71.5% ±1%	70.8% ±1%	70.0% ±1%	69.7% ±1%	69.8% ±2%	69.1% ±2%	68.8% ±2%	68.7% ±1%	68.3% ±1%	68.1% ±2%	67.1% ±1%	67.0% ±1%	66.2% ±1%	65.8% ±2%	66.3% ±1%	65.4% ±2%
32X	76.9% ±2%	77.4% ±1%	77.3% ±1%	77.4% ±2%	76.9% ±2%	76.4% ±2%	74.8% ±2%	73.3% ±2%	73.1% ±2%	72.7% ±2%	71.3% ±2%	71.3% ±2%	71.0% ±2%	70.3% ±2%	68.9% ±2%	69.0% ±2%	67.5% ±3%	67.2% ±3%	65.9% ±4%	65.0% ±4%	64.6% ±5%	64.5% ±4%	63.8% ±5%	62.6% ±6%
36X	75.2% ±2%	76.0% ±3%	75.7% ±2%	76.3% ±1%	75.7% ±1%	75.7% ±1%	75.0% ±2%	74.4% ±3%	74.2% ±3%	73.4% ±3%	72.3% ±3%	72.0% ±3%	71.6% ±2%	71.1% ±2%	70.6% ±2%	70.2% ±2%	69.6% ±2%	68.6% ±1%	68.1% ±2%	67.8% ±1%	67.4% ±2%	67.8% ±2%	67.1% ±2%	65.9% ±2%
40X	75.5% ±1%	76.1% ±1%	76.6% ±1%	77.4% ±1%	76.9% ±1%	77.0% ±1%	76.1% ±2%	74.9% ±2%	73.5% ±2%	72.6% ±2%	71.6% ±1%	71.5% ±2%	70.5% ±2%	70.1% ±2%	69.3% ±2%	69.2% ±2%	68.8% ±1%	68.3% ±2%	67.5% ±1%	67.5% ±2%	66.9% ±2%	66.9% ±1%	66.8% ±2%	66.8% ±1%
50X	74.4% ±2%	74.6% ±3%	74.5% ±2%	74.6% ±3%	74.8% ±3%	74.9% ±2%	77.4% ±2%	77.0% ±3%	75.7% ±2%	73.9% ±2%	73.0% ±2%	72.4% ±2%	71.3% ±2%	70.4% ±2%	69.4% ±1%	68.9% ±1%	68.1% ±1%	67.6% ±1%	67.3% ±2%	67.0% ±3%	66.3% ±3%	66.6% ±3%	66.3% ±3%	66.0% ±3%
60X	73.5% ±3%	75.4% ±4%	74.4% ±2%	75.2% ±2%	74.2% ±2%	74.2% ±3%	77.5% ±2%	76.7% ±2%	75.4% ±2%	74.0% ±2%	72.3% ±2%	71.3% ±2%	70.2% ±2%	69.6% ±1%	68.9% ±1%	68.8% ±1%	68.0% ±1%	67.8% ±1%	67.4% ±1%	67.3% ±1%	67.0% ±1%	67.0% ±2%	66.9% ±2%	66.9% ±2%
70X	74.4% ±2%	75.6% ±2%	75.8% ±2%	75.8% ±3%	76.1% ±3%	75.8% ±2%	78.9% ±2%	78.5% ±3%	77.3% ±3%	76.2% ±3%	74.1% ±2%	73.5% ±2%	72.5% ±2%	71.9% ±2%	71.6% ±1%	71.2% ±1%	70.1% ±1%	69.7% ±2%	69.1% ±2%	68.8% ±2%	68.1% ±2%	68.0% ±2%	67.3% ±2%	67.4% ±2%
80X	73.8% ±4%	74.5% ±4%	74.7% ±4%	75.5% ±3%	74.9% ±3%	75.5% ±3%	78.3% ±4%	77.3% ±4%	76.5% ±4%	75.5% ±4%	74.1% ±4%	73.1% ±3%	71.9% ±3%	71.3% ±3%	70.5% ±3%	69.8% ±2%	69.5% ±2%	69.2% ±2%	68.7% ±2%	68.5% ±2%	67.5% ±2%	67.7% ±2%	67.5% ±2%	67.7% ±2%
90X	71.1% ±2%	72.0% ±1%	72.3% ±1%	73.0% ±1%	73.1% ±1%	72.6% ±1%	76.5% ±2%	76.2% ±2%	75.2% ±2%	74.1% ±2%	72.8% ±2%	71.7% ±2%	70.9% ±1%	70.6% ±2%	69.9% ±1%	69.8% ±1%	69.5% ±1%	69.3% ±1%	68.6% ±1%	68.4% ±1%	68.1% ±1%	67.9% ±1%	68.1% ±1%	67.7% ±1%
100X	72.2% ±4%	72.8% ±3%	73.8% ±3%	73.8% ±2%	74.2% ±2%	73.7% ±2%	76.5% ±2%	76.0% ±3%	75.9% ±3%	74.9% ±3%	74.0% ±2%	73.3% ±2%	72.2% ±2%	71.3% ±2%	70.5% ±1%	70.5% ±1%	70.2% ±2%	69.7% ±1%	68.9% ±1%	69.1% ±1%	68.4% ±1%	68.3% ±1%	68.5% ±2%	68.4% ±2%
116X	74.5% ±4%	75.6% ±2%	75.1% ±2%	75.1% ±2%	75.5% ±2%	75.6% ±2%	78.4% ±2%	77.4% ±3%	76.9% ±3%	76.5% ±3%	75.9% ±3%	74.9% ±3%	73.6% ±3%	72.6% ±3%	71.4% ±3%	70.9% ±2%	70.9% ±2%	70.6% ±2%	69.7% ±2%	69.2% ±2%	68.7% ±2%	68.6% ±2%	68.4% ±2%	67.8% ±2%
133X	72.0% ±3%	72.9% ±3%	72.7% ±2%	73.3% ±2%	73.6% ±2%	73.5% ±2%	74.9% ±2%	74.5% ±2%	74.6% ±3%	74.2% ±3%	72.7% ±2%	72.4% ±2%	72.2% ±2%	71.5% ±2%	70.6% ±1%	70.3% ±1%	69.6% ±1%	69.7% ±1%	69.1% ±1%	68.7% ±2%	68.0% ±2%	67.9% ±2%	67.5% ±2%	67.2% ±2%
150X	72.5% ±4%	73.0% ±3%	73.0% ±3%	72.9% ±3%	73.4% ±3%	73.5% ±3%	75.0% ±3%	74.4% ±4%	74.7% ±3%	73.9% ±4%	73.5% ±3%	74.0% ±4%	73.3% ±3%	73.1% ±3%	72.9% ±3%	72.4% ±3%	71.8% ±2%	71.2% ±2%	70.6% ±2%	69.5% ±2%	68.7% ±2%	68.7% ±2%	68.4% ±2%	68.2% ±2%
166X	73.4% ±6%	73.3% ±6%	72.3% ±5%	72.4% ±5%	72.1% ±5%	73.1% ±5%	74.7% ±6%	74.3% ±6%	74.8% ±6%	74.6% ±6%	75.1% ±6%	75.2% ±6%	74.9% ±5%	74.6% ±6%	74.3% ±6%	73.8% ±6%	73.3% ±5%	72.6% ±5%	71.7% ±5%	70.4% ±5%	69.8% ±4%	69.5% ±4%	69.0% ±4%	68.8% ±4%
183X	71.5% ±3%	71.8% ±3%	71.7% ±3%	71.5% ±3%	72.2% ±3%	72.1% ±3%	71.0% ±4%	71.0% ±4%	72.3% ±4%	72.9% ±4%	72.6% ±4%	73.3% ±4%	73.6% ±4%	73.6% ±3%	73.6% ±4%	73.5% ±4%	73.7% ±3%	73.1% ±4%	72.6% ±4%	70.3% ±3%	69.8% ±3%	68.5% ±3%	68.2% ±3%	67.7% ±3%
200X	71.7% ±4%	71.7% ±4%	71.0% ±3%	70.7% ±3%	70.6% ±3%	71.6% ±3%	70.3% ±4%	70.3% ±4%	71.7% ±5%	72.0% ±4%	72.0% ±4%	72.9% ±4%	72.6% ±4%	72.5% ±4%	73.1% ±4%	73.4% ±4%	73.0% ±4%	72.6% ±4%	72.0% ±4%	71.2% ±3%	70.2% ±3%	69.6% ±2%	69.4% ±2%	68.9% ±2%
233X	67.9% ±5%	68.2% ±4%	67.8% ±4%	67.7% ±3%	68.3% ±3%	68.2% ±4%	67.2% ±4%	68.9% ±4%	69.3% ±5%	70.6% ±5%	70.8% ±5%	70.7% ±4%	71.0% ±4%	71.2% ±4%	71.2% ±4%	71.1% ±4%	71.1% ±4%	71.1% ±4%	71.3% ±4%	70.4% ±3%	69.8% ±3%	69.5% ±3%	69.4% ±3%	69.1% ±3%
266X	68.4% ±4%	69.0% ±5%	69.0% ±4%	69.1% ±4%	68.9% ±3%	68.9% ±3%	68.0% ±4%	69.1% ±5%	69.7% ±5%	69.9% ±5%	70.7% ±6%	71.5% ±5%	72.0% ±6%	72.8% ±5%	72.4% ±5%	72.4% ±4%	72.6% ±5%	73.2% ±5%	73.1% ±5%	72.7% ±4%	72.2% ±4%	71.9% ±4%	71.4% ±4%	70.9% ±4%
300X	68.7% ±3%	68.6% ±2%	67.8% ±2%	67.3% ±2%	67.7% ±2%	68.0% ±2%	67.4% ±2%	68.2% ±2%	68.6% ±2%	68.7% ±3%	69.1% ±2%	70.2% ±2%	70.5% ±2%	71.1% ±2%	71.2% ±2%	72.4% ±2%	72.4% ±2%	72.6% ±2%	72.0% ±2%	71.6% ±2%	71.5% ±2%	71.0% ±2%	70.5% ±2%	70.6% ±2%
333X	68.4% ±2%	68.1% ±2%	67.9% ±2%	68.3% ±2%	68.6% ±2%	69.4% ±3%	67.8% ±2%	67.8% ±3%	69.2% ±3%	69.7% ±3%	69.7% ±3%	70.8% ±4%	71.6% ±4%	71.6% ±3%	72.9% ±4%	73.1% ±4%	73.7% ±5%	73.6% ±5%	73.6% ±4%	73.0% ±4%	73.2% ±5%	73.0% ±5%	72.8% ±5%	72.0% ±5%
366X	67.1% ±4%	67.2% ±3%	67.4% ±3%	67.5% ±3%	68.0% ±4%	68.2% ±5%	65.9% ±3%	67.2% ±3%	67.5% ±4%	67.8% ±5%	68.2% ±5%	69.1% ±5%	70.8% ±6%	70.9% ±6%	71.2% ±6%	71.4% ±6%	71.6% ±6%	71.9% ±6%	71.4% ±6%	71.3% ±5%	71.7% ±5%	71.1% ±5%	70.6% ±5%	70.3% ±4%
400X	65.5% ±3%	65.9% ±3%	66.2% ±2%	66.0% ±2%	66.3% ±2%	65.5% ±3%	64.9% ±4%	65.4% ±3%	65.4% ±3%	66.5% ±2%	66.7% ±2%	66.6% ±3%	67.3% ±3%	67.8% ±3%	68.4% ±3%	68.8% ±3%	69.2% ±3%	69.1% ±3%	70.1% ±3%	69.4% ±4%	69.5% ±4%	69.1% ±5%	68.6% ±4%	69.2% ±4%
all X	76.1% ±3%	76.5% ±2%	75.5% ±3%	76.6% ±3%	76.5% ±2%	76.6% ±2%	76.6% ±3%	75.9% ±3%	76.3% ±4%	76.4% ±3%	75.5% ±3%	76.1% ±3%	76.1% ±4%	75.4% ±3%	74.4% ±3%	74.6% ±3%	74.9% ±3%	74.6% ±4%	74.2% ±4%	73.7% ±4%	73.7% ±5%	72.6% ±5%	72.4% ±6%	70.6% ±7%

Fig. 2. Inception-ResNet-v2 Accuracy Heatmap. Shows the accuracies (%) obtained by individually tested trained networks on all datasets in a range of magnification levels from 20X to 400X. The higher the accuracy of a network on a dataset, the darker the box, and vice versa.

In parallel, we measure the number of patches processed per second during the testing step on a subsample of 10 networks. Then, assuming that this calculation is linear, we convert this number of patches per second into the number of 720p and 1080p definition frames tested per second (FPS) to obtain the Table 1. These measurements were carried out using a processor (CPU) [5] then two GPU [6].

[5] i5-11600KF 3.90GHz 3.91 GHz

[6] NVIDIA GeForce GTX 1080 & RTX 3080

Table 1. Number of FPS processed by Inception-ResNet-v2. The average number of frames per second (FPS) tested by a network depending on the type of resource used (GPU or CPU) and on the image definition (720p or 1080p).

Image definition	720p	1080p
CPU	1.3	0.6
GPU 1080	3.5	1.7
GPU 3080	7.3	3.2

4 Results and discussion

4.1 Real-time

Displaying augmented reality information on a microscope requires a real-time method capable of tracking a pathologist's movements. We define the notion of real-time based on [29]. Thus, for the tumor patch classification task, we speak of real time to mean that our method is able to predict the field of view observed under a microscope several times per second as shown in Table 1. Indeed, the Inception-ResNet-v2 architecture processes an average of 7.3 FPS at 720p resolution with a GPU, which can classify images quite quickly for a smooth result.

With 55,873,736 parameters and a depth of 572, Inception-ResNet-v2 is one of the computationally intensive networks. Despite its size, it is possible to perform real-time tasks using a single, latest-generation GPU. Thanks to recent advances in computational capacity, it is now possible to run deep learning networks in real time by embedding a commercial GPU on a microscope.

4.2 Heatmap

First, it is clear that networks trained on one magnification level have the best accuracies on nearby magnification levels, as shown in Figure 2. It is obvious that a network will generally be better on a magnification level near to the one used in the training step. However, by looking at the central diagonal, we can see that the best accuracy at one level of magnification is not always found by the trained network on that level of magnification. In other words, a slight zooming in or out during training improves the network's results. It appears that near context contains rich features that improve the results of a network. The results also show that at a given magnification level, networks trained at that level and nearby levels perform best. Moreover, these networks do not necessarily misjudge on the same cases. This means that each level of magnification does not provide the same features.

The second observation concerns the gradient on both sides of the diagonal of the darkest boxes. The further the magnification level of a tested dataset is from the magnification level of the dataset used for training, the lower the accuracy. This seems to confirm the intuition that the near context contains the most useful features for a network.

Third, we note that the higher the magnification level, the worse the accuracies of networks trained on a single magnification level. At higher magnification levels, the spatial context for understanding the structure of a tumor is lost. Therefore, observing tissue at high magnification is not relevant for tumor classification in breast cancer. We note that magnification levels between 20X and 80X are more rich in features for our networks in our case. In practice, this coincides with the behavior of the pathologist, who spends more time in the lower magnification levels.

Fourth, note that the best networks give accuracies close to 80%. This performance is lower than the state of the art in tumor patch classification in breast cancer on BreakHis [30]. This is due to the way we build our datasets from BreakHis. Indeed, in order to study a large range of magnification levels, we build patches of new levels, so part of the context has been removed, which impacts the performance of the networks. We also notice that in the worst case, the lowest accuracy obtained is 62.6%. This means that in the case of breast cancer, the features to distinguish benign and malignant tumors are present at all magnification levels of the study range.

4.3 Comparison of approaches

From the heatmap in Figure 2, we can compare our approach with two naive methods as reported in Table 2.

First, we find that the first approach, where we train the networks on the magnification level used for the test, manages to obtain the best accuracies on some levels in the 20X to 70X range.

Second, the networks trained on all magnification levels achieve best accuracies over the 100X to 300X range. This second approach succeeds in generalizing the learned features over multiple levels, thus obtaining better accuracies over higher magnification levels. However, this method fails to outperform specialized networks on lower magnification levels (20X to 100X).

Table 2. Comparison of approaches. Comparison of the accuracies (%) of three tumor prediction approaches on different magnification levels: Training networks on several levels; Training networks on each level; Training networks on an intermediate level.

Levels of	network trained	network trained on	network trained	
magnification tested	& tested on one level	40X, 100X, 200X, 400X	on an intermediate level	The best network
		(allX)	(Our method)	
20X	0.742	0.761	**0.784**	24X
24X	**0.779**	0.765	**0.779**	24X
28X	0.755	0.755	**0.773**	32X
32X	**0.774**	0.766	**0.774**	32-40X
36X	0.757	0.765	**0.769**	32-40X
40X	**0.770**	0.766	**0.770**	40X
50X	0.774	0.766	**0.789**	70X
60X	0.767	0.759	**0.785**	70X
70X	**0.773**	0.763	**0.773**	70X
80X	0.755	0.764	**0.765**	116X
90X	0.728	0.755	**0.759**	116X
100X	0.733	**0.761**	0.752	allX
116X	0.736	**0.761**	0.749	allX
133X	0.715	**0.754**	0.746	allX
150X	0.729	**0.744**	0.743	allX
166X	0.738	**0.746**	0.738	allX
183X	0.737	**0.749**	0.737	allX
200X	0.726	**0.746**	0.736	allX
233X	0.713	**0.742**	0.736	allX
266X	0.727	**0.737**	0.730	allX
300X	0.715	**0.737**	0.732	allX
333X	**0.730**	0.726	**0.730**	333X
366X	0.706	0.724	**0.728**	333X
400X	0.692	0.706	**0.720**	333X
Average	0.740	0.751	**0.754**	

Third, we find that the proposed approach, where we select the best networks trained on any magnification level, achieves the best accuracies for the 20X to 90X range. A magnification level contains features that sometimes yield better results on nearby magnification levels.

On average, the proposed approach (75.4%) outperforms or equals the naive approaches (74% & 75.1%) over the range of magnification levels studied with the Inception-ResNet-v2 architecture on the BreakHis dataset. Our approach shows better results on low magnification levels (20X to 90X). From 100X, the allX approach becomes better. Beyond a certain magnification level, the tumor features present in the image become less relevant. However, the features at low magnification levels significantly help the understanding of the features at high magnification levels.

Looking at the best networks by magnification levels shown in Table 2, we observe that the 24X, 32X, 40X, 70X, 116X, 333X, and allX networks are able to cover the entire range of magnification levels studied, while achieving the best results.

In the context of an ARM method, we want to display additional information about the observed tissue at the few magnification levels offered by the optical microscope. Let's take for example a microscope with 20X, 40X and 100X magnification levels. In order to obtain the best results in our case, we must use: the 24X model on the 20X level; the 40X model on the 40X level; the allX model on the 100X level.

It seems interesting to use networks trained on a single level of magnification for the lowest levels of magnification up to 90X. Then, from 100X, it is advisable to use gratings trained on several levels of magnification.

Finally, it should be kept in mind that these results were obtained with the Inception-ResNet-v2 architecture on a breast cancer dataset. Depending on the method and the data used, the conclusions may change.

5 Conclusion

In recent years, the rapid evolution of the computing capabilities of graphics cards has largely contributed to the appearance of new approaches improving the overall performance of deep learning methods. These advances allow us to more seriously consider deep learning methods for embedded real-time microscope tools. Such applications can be used to assist pathologists in their daily routine.

We have proposed an approach that takes advantage of the intermediate magnification levels of WSIs to improve the performance of an ARM method that uses standard magnification levels of an optical microscope.

Future work should focus on the use of ensemble learning methods to take full advantage of the features of each level of magnification. Finally, it would be interesting to see real-time approaches being implemented on different tasks in various medical and other applications. This will raise awareness of the challenges related to the use of real-time deep learning methods in real conditions.

6 Acknowledgments

The authors would like to acknowledge the High Performance Computing center of the University of Strasbourg for supporting this work by providing scientific support and access to computing resources. Part of the computing resources were funded by the Equipex Equip@Meso project (Programme Investissements d'Avenir) and the CPER Alsacalcul/Big Data.

References

1. Al-Janabi, S., Huisman, A., Van Diest, P.J.: Digital pathology: current status and future perspectives. Histopathology **61**(1) (2012) 1–9
2. Voulodimos, A., Doulamis, N., Doulamis, A., Protopapadakis, E.: Deep learning for computer vision: A brief review. Computational intelligence and neuroscience **2018** (2018)
3. Madabhushi, A., Lee, G.: Image analysis and machine learning in digital pathology: Challenges and opportunities. Medical image analysis **33** (2016) 170–175
4. Niazi, M.K.K., Parwani, A.V., Gurcan, M.N.: Digital pathology and artificial intelligence. The lancet oncology **20**(5) (2019) e253–e261
5. Tizhoosh, H.R., Pantanowitz, L.: Artificial intelligence and digital pathology: challenges and opportunities. Journal of pathology informatics **9** (2018)
6. Chugh, G., Kumar, S., Singh, N.: Survey on machine learning and deep learning applications in breast cancer diagnosis. Cognitive Computation **13**(6) (2021) 1451–1470
7. Debelee, T.G., Schwenker, F., Ibenthal, A., Yohannes, D.: Survey of deep learning in breast cancer image analysis. Evolving Systems **11**(1) (2020) 143–163
8. Chen, H., Li, C., Li, X., Rahaman, M.M., Hu, W., Li, Y., Liu, W., Sun, C., Sun, H., Huang, X., et al.: Il-mcam: An interactive learning and multi-channel attention mechanism-based weakly supervised colorectal histopathology image classification approach. Computers in Biology and Medicine **143** (2022) 105265
9. Golatkar, A., Anand, D., Sethi, A.: Classification of breast cancer histology using deep learning. In: International conference image analysis and recognition, Springer (2018) 837–844
10. Han, Z., Wei, B., Zheng, Y., Yin, Y., Li, K., Li, S.: Breast cancer multi-classification from histopathological images with structured deep learning model. Scientific reports **7**(1) (2017) 1–10
11. Khan, S., Islam, N., Jan, Z., Din, I.U., Rodrigues, J.J.C.: A novel deep learning based framework for the detection and classification of breast cancer using transfer learning. Pattern Recognition Letters **125** (2019) 1–6
12. Wang, D., Khosla, A., Gargeya, R., Irshad, H., Beck, A.H.: Deep learning for identifying metastatic breast cancer. arXiv preprint arXiv:1606.05718 (2016)
13. Afework, A., Beynon, M.D., Bustamante, F., Cho, S., Demarzo, A., Ferreira, R., Miller, R., Silberman, M., Saltz, J., Sussman, A., et al.: Digital dynamic telepathology–the virtual microscope. In: Proceedings of the AMIA Symposium, American Medical Informatics Association (1998) 912
14. Ferreira, R., Moon, B., Humphries, J., Sussman, A., Saltz, J., Miller, R., Demarzo, A.: The virtual microscope. In: Proceedings of the AMIA Annual Fall Symposium, American Medical Informatics Association (1997) 449
15. Treanor, D., Jordan-Owers, N., Hodrien, J., Wood, J., Quirke, P., Ruddle, R.A.: Virtual reality powerwall versus conventional microscope for viewing pathology slides: an experimental comparison. Histopathology **55**(3) (2009) 294–300
16. Heckenauer, R., Weber, J., Wemmert, C., Feuerhake, F., Hassenforder, M., Muller, P.A., Forestier, G.: Real-time detection of glomeruli in renal pathology. In: 2020 IEEE 33rd International Symposium on Computer-Based Medical Systems (CBMS), IEEE (2020) 350–355
17. Szegedy, C., Ioffe, S., Vanhoucke, V., Alemi, A.A.: Inception-v4, inception-resnet and the impact of residual connections on learning. In: Thirty-first AAAI conference on artificial intelligence. (2017)
18. Spanhol, F.A., Oliveira, L.S., Petitjean, C., Heutte, L.: A dataset for breast cancer histopathological image classification. Ieee transactions on biomedical engineering **63**(7) (2015) 1455–1462
19. Chen, P.H.C., Gadepalli, K., MacDonald, R., Liu, Y., Nagpal, K., Kohlberger, T., Dean, J., Corrado, G.S., Hipp, J.D., Stumpe, M.C.: Microscope 2.0: An augmented reality microscope with real-time artificial intelligence integration. arXiv preprint arXiv:1812.00825 (2018)
20. Razavian, N.: Augmented reality microscopes for cancer histopathology. Nature Medicine **25**(9) (2019) 1334–1336

21. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2016) 2818–2826
22. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2016) 770–778
23. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2015) 1–9
24. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
25. Keys, R.: Cubic convolution interpolation for digital image processing. IEEE transactions on acoustics, speech, and signal processing **29**(6) (1981) 1153–1160
26. Parsania, P.S., Virparia, P.V.: A comparative analysis of image interpolation algorithms. International Journal of Advanced Research in Computer and Communication Engineering **5**(1) (2016) 29–34
27. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition, Ieee (2009) 248–255
28. Kora, P., Ooi, C.P., Faust, O., Raghavendra, U., Gudigar, A., Chan, W.Y., Meenakshi, K., Swaraja, K., Plawiak, P., Acharya, U.R.: Transfer learning techniques for medical image analysis: A review. Biocybernetics and Biomedical Engineering (2021)
29. Dodhiawala, R.T., Sridharan, N., Raulefs, P., Pickering, C.: Real-time ai systems: A definition and an architecture. In: IJCAI, Citeseer (1989) 256–264
30. Tufail, A.B., Ma, Y.K., Kaabar, M.K., Martínez, F., Junejo, A., Ullah, I., Khan, R.: Deep learning in cancer diagnosis and prognosis prediction: a minireview on challenges, recent trends, and future directions. Computational and Mathematical Methods in Medicine **2021** (2021)

Generative Adversarial Network for Facial Emotion Recognition: A Feasibility Study

Herag Arabian and Knut Moeller

Institute of Technical Medicine (ITeM) – Furtwangen University
H.Arabian@hs-furtwangen.de

Abstract. Integration of artificial intelligence into different domains has been a trending topic over the past few years. A closed-loop feedback system which immerses the subject in a virtual reality environment with a novel reward platform is being developed to help people suffering from autism spectrum disorder. In this work, the feasibility of using generative adversarial networks to generate synthetic images by restructuring unseen input data to match that of the training set for the recognition of human emotions is being studied. System performance was based on true positive predictions from the different classification models developed in previous work. Preliminary results showed that the proposed system was able to improve class predictions, but lacked in the ability to generate different class sets. The performance highlights the feasibility of this method and its practical applications in generating more data and improving model robustness.

Keywords: Facial Emotion Recognition; Generative Adversarial Networks; Therapeutic Application.

1 Introduction

Integration of artificial intelligence (AI) into different domains has been a trending topic over the past few years. One of the most popular forms of AI is the use of Deep learning techniques i.e. neural networks for classification tasks, as they show better performance over traditional machine learning methods [1]. Applications such as voice-command recognition and text transcription have become second nature in our daily lives. In contrast to the popularity and acceptance of this technology, it is important to note that they are still considered “Black Boxes” and vulnerable to even the smallest of disturbances [2], [3]. The robustness and reliability of deep learning algorithms is a key topic in research, as ever progressing studies highlight the need for AI to be incorporated in the medical domain.

The standard for image classification in deep learning has been the use of convolution neural networks (CNN) due to their ability to identify relevant features from data. However, several studies [3], [4] have shown that they are vulnerable to slight pixel changes and are strongly dependent on the training data. In order to improve the robustness of these models a different approach is proposed in this work, whereby generative adversarial networks (GAN) are used to generate synthetic images by restructuring the unseen input data to match that of the training set. GAN [5] is a system composed of a generator and a discriminator network, where synthetic data is generated then compared to the real data to see if they match.

In this work, the feasibility of the proposed approach is studied for the recognition of human emotions. Facial emotion recognition (FER) is currently being considered as a method to help treat patients with autism spectrum disorder (ASD) a developmental brain disorder that affects the social interactions and communications of individuals [6]. Facial expressions have shown to convey 55% of a person’s feelings and attitudes [7]. A closedloop feedback system with a novel reward system is being developed which immerses the subject in a virtual reality environment i.e., a game, in which the user is subjected to different activities i.e. social interactions as well as emotional stimuli [8]. In [9]–[12] the use of Conditional GANs were studied and the results showed the possibilities of utilizing such systems in reducing the impact of variations of new unseen data on trained FER models.

Three facial emotion databases were selected for this study the OULU-CASIA [13], FACES [14], and Japanese female facial expressions (JAFFE) [15]. Images are first pre-processed to highlight the face of the subject by removing background noise according to the technique described in [16]. The proposed model is then trained to generate synthetic images, from the FACES and JAFFE datasets, that resemble part of the data from the OULU-CASIA database. The performance of the model is based on the outcome of the generated images and its similarity to the original data.

The aim of this study is to determine the feasibility of such a proposed model in improving CNN model robustness for FER.

2 Methods

Image pre-processing, developed in previous work [16], was implemented on the datasets of OULU-CASIA, FACES, and JAFFE to focus on the face of the subject and remove background noise. Three subjects from the OULU-CASIA were chosen as the input for real images and the images of the FACES and JAFFE were set as input into to system to generate synthetic images that match the real inputs. **2.1 Model Development**

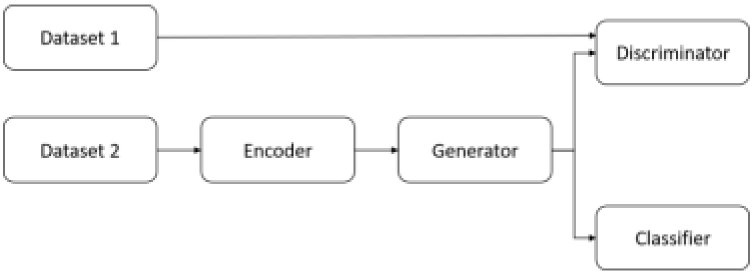


Fig. 1. Proposed GAN model

Figure 1 represents the flowchart of the proposed GAN system. The data first pass through an Encoder block (Enet) where the relevant features are extracted, the output is then taken as input to the Generator block (Gnet) which generates the synthetic images. After which the output is passed into the Discriminator block (Dnet) along with the labels which are embedded into the image, the Dnet distinguishes between the real and synthetic images. The outputs from the Gnet are also passed to the Classification block (Cnet) which classifies the inputs into the different emotion classes.

The Enet represents a shallow CNN with an architecture similar to that of Alexnet [17] coupled with normalization blocks after each convolution block. The Enet extracts different features from the input images of size 96x96x3 from Dataset 2 which refers to the FACES and JAFFE combined and outputs a 1x4096 feature vector. The Gnet also represents a shallow decoder network that takes in input from the Enet and outputs an image with the same size as the input images.

The Dnet takes in input the real images and labels of Dataset 1 which refers to the OULU-CASIA dataset, the generated images from Gnet and the labels from the Dataset 2. The labels are embedded into the image to make the input dimension 96x96x4. The Cnet represents a CNN that was trained on the OULU-CASIA data from previous work [16]. The system loss of Gnet is based on the logarithmic mean log likelihood function combined with the loss of the Cnet, which is computed by cross entropy loss. The Dnet loss is based on the logarithmic mean log likelihood function. The parameters of the Enet, Gnet and Dnet are updated after each iteration by means of the adaptive moment estimation (adam) optimization method. The Cnet parameters were not updated as the trained model showed high performance accuracies of 98% on its validation set [16].

2.2 Database Description

The Oulu-CASIA database is composed of 80 different subjects expressing six basic emotions of anger, disgust, fear, happiness, sadness and surprise. The database consists of image sequences beginning with Neutral expression and ending with strong emotion expression. Images of original RGB, visible light with strong illumination lighting were selected with a total of 10,379 images [13].

The Japanese female facial expressions (JAFFE) database is composed of 213 facial portrait images portrayed in grey scale from 10 different Japanese female students expressing seven emotions (six basic plus Neutral) [15]. The FACES dataset has a total of 2,052 images expressing six emotion classes of anger, disgust, fear, happiness, neutral and sadness of varying subject ages [14]. To analyze the performance of the model in generating synthetic images close to the real images, the classification performance was assessed by evaluating the true positive predictions from the different classification models developed in [16].

3 Results & Discussion

Table 1 shows the distribution of the images into each class for the FACES and JAFFE datasets. The image pre-processing algorithm excluded 8.77% and 1.41% of the images of FACES and JAFFE datasets respectively, from further processing due to the failure of the method to segment the prescribed regions [16]. The classes were distributed near equally with the Neutral class being removed from the analysis. Three subjects from the OULU-CASIA database were selected by random and set as the real images for training the system.

Table 1. Class Distribution before and after image pre-processing for the FACES and JAFFE databases.

	FACES			JAFFE		
Class	**Original**	**After Processing**	**% each Class***	**Original**	**After Processing**	**% each Class***
Anger	342	292	18.94	30	30	16.67
Disgust	342	292	18.94	29	29	16.11
Fear	342	303	19.65	32	32	17.78
Happiness	342	338	21.92	31	31	17.22
Sadness	342	317	20.55	31	31	17.22
Surprise	0	0	N/A	30	27	15.00
Total	**1710**	**1542**	**100.00**	**183**	**180**	**100.00**

*The percentage is calculated based on the Pre-processed image data

The mean performance of the classification models from [16] on the FACES and JAFFE datasets before and after synthetic image generation is represented in table 2. As seen from the results the GAN system was able to achieve a slightly better overall performance with a lower standard deviation. However, looking at the mean accuracies of each class it was seen that the GAN system was not able to generate the classes of Happiness, Sadness and Surprise correctly and lacked any predictive results. The performance of Anger, Disgust and Fear showed improvements of greater than 20% per class. This signifies that the GAN system was robust for these three classes especially taking into consideration the color variation between the datasets of FACES and JAFFE.

Table 2. Performance results of the true positive predictions on the testing set of FACES and JAFFE before and after synthetic image generation.

Mean ± SD %	**Before GAN**	**After GAN**
Anger	32.59 ± 16.38	72.41 ± 10.27
Disgust	73.83 ± 11.06	100.00 ± 0.00
Fear	73.87 ± 3.88	94.40 ± 1.97
Happiness	57.87 ± 13.05	0
Sadness	8.57 ± 1.86	0
Surprise	20.00 ± 17.21	0
Mean	**48.81 ± 4.31**	**49.39 ± 1.84**

4 Conclusion

In this study the feasibility of implementing a GAN system to improve FER robustness was analyzed. The preliminary results showed that the GAN system was able to improve the class predictions, however it lacked in the ability to generate the complete class set. The performance highlights the feasibility of this method and its practical applications in generating more data and improving the robustness of FER. More work is planned with better fine tuning of parameters and inclusion of more subjects for real images to improve on the existing results.

Author’s Statement

Research funding: Partial support by a grant from the German Federal Ministry of Research and Education (BMBF) under project No. 13FH5I06IA – PersonaMed is gratefully acknowledged. Conflict of interest: Authors state no conflict of interest.

References

1. Y. LeCun, Y. Bengio, and G. Hinton, ‘Deep learning’, Nature, vol. 521, no. 7553, pp. 436–444, May 2015.
2. W. Samek, T. Wiegand, and K.-R. Müller, ‘Explainable Artificial Intelligence: Understanding, Visualizing and Interpreting Deep Learning Models’, ArXiv170808296 Cs Stat, Aug. 2017
3. X. Yuan, P. He, Q. Zhu, and X. Li, ‘Adversarial Examples: Attacks and Defenses for Deep Learning’. arXiv, Jul. 06, 2018.
4. N. Akhtar and A. Mian, ‘Threat of Adversarial Attacks on Deep Learning in Computer Vision: A Survey’. arXiv, Feb. 26, 2018.
5. I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning. MIT Press, 2016.
6. C. on C. W. Disabilities, ‘The Pediatrician’s Role in the Diagnosis and Management of Autistic Spectrum Disorder in Children’, Pediatrics, vol. 107, no. 5, pp. 1221–1226, May 2001.
7. A. Mehrabian, ‘Communication without words’, in Communication Theory, C. D. Mortensen, Ed. Routledge, 2017.
8. H. Arabian, V. Wagner-Hartl, J. Geoffrey Chase, and K. Moeller, ‘Image Pre-processing Significance on Regions of Impact in a Trained Network for Facial Emotion Recognition’, IFAC BMS 21, IFACPapersOnLine, Volume 54, Issue 15, 2021, Pages 299-303, ISSN 2405-8963, https://doi.org/10.1016/j.ifacol.2021.10.272.
9. J. Cai et al., ‘Identity-Free Facial Expression Recognition Using Conditional Generative Adversarial Network’, in 2021 IEEE International Conference on Image Processing (ICIP), Sep. 2021, pp. 1344–1348.
10. J. Chen, J. Konrad, and P. Ishwar, ‘VGAN-Based Image Representation Learning for Privacy-Preserving Facial Expression Recognition’. arXiv, Sep. 07, 2018.
11. H. Yang, U. Ciftci, and L. Yin, ‘Facial Expression Recognition by De-expression Residue Learning’, in 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, Jun. 2018, pp. 2168–2177.
12. H. Yang, Z. Zhang, and L. Yin, ‘Identity-Adaptive Facial Expression Recognition through Expression Regeneration Using Conditional Generative Adversarial Networks’, in 2018 13th IEEE International Conference on Automatic Face Gesture Recognition (FG 2018), May 2018, pp. 294–301.
13. G. Zhao, X. Huang, M. Taini, S. Z. Li, and M. Pietikäinen, ‘Facial expression recognition from nearinfrared videos’, Image Vis. Comput., vol. 29, no. 9, pp. 607–619, Aug. 2011.
14. N. C. Ebner, M. Riediger, and U. Lindenberger, ‘FACES—A database of facial expressions in young, middle-aged, and older women and men: Development and validation’, Behav. Res. Methods, vol. 42, no. 1, pp. 351–362, Feb. 2010.
15. M. J. Lyons, M. Kamachi, and J. Gyoba, ‘Coding Facial Expressions with Gabor Wavelets (IVC Special Issue)’, 2020.
16. H. Arabian, V. Wagner-Hartl, and K. Moeller, ‘Image Pre-processing Effects on Attention Modules in Facial Emotion Recognition’, IUPESM World Congress on Medical Physics and Biomedical Engineering (IUPESM WC2022), In Press.
17. A. Krizhevsky, I. Sutskever, and G. E. Hinton, ‘ImageNet classification with deep convolutional neural networks’, Commun. ACM, vol. 60, no. 6, pp. 84–90, May 2017.
18. H. Arabian, V. Wagner-Hartl, and K. Moeller, ‘Traditional versus Neural Network Classification Methods for Facial Emotion Recognition’, presented at the VDE BMT, 2021.
19. H. Arabian, V. Wagner-Hartl, and K. Möller, ‘Facial emotion recognition based on localized region segmentation’, Jun. 17, 2021. doi: 10.5281/zenodo.4922791.
20. H. Arabian, V. Wagner-Hartl, J. Geoffrey Chase, and K. Möller, ‘Facial Emotion Recognition Focused on Descriptive Region Segmentation’, in 2021 43rd Annual International Conference of the IEEE Engineering in Medicine Biology Society (EMBC), Nov. 2021, pp. 3415–3418. doi: 10.1109/EMBC46164.2021.9629742.

21. H. Arabian, V. Wagner-Hartl, and K. Moeller, ‘Attention Modules for Facial Emotion Recognition Network Robustness Improvement’, In Press.
22. H. Arabian, V. Wagner-Hartl, J. Geoffrey Chase, and K. Möller, ‘Facial Emotion Recognition Focused on Descriptive Region Segmentation’, in 2021 43rd Annual International Conference of the IEEE Engineering in Medicine Biology Society (EMBC), Nov. 2021, pp. 3415–3418. doi: 10.1109/EMBC46164.2021.9629742.
23. H. Arabian, T. Abdulbaki Alshirbaji, N. A. Jalal, N. Ding, B. Laufer, and K. Moeller, ‘Identifying User Adherence to Digital Health Apps’, presented at the IUPESM WORLD CONGRESS ON MEDICAL PHYSICS AND BIOMEDICAL ENGINEERING (IUPESM WC2022), in press.
24. H. Arabian, V. Wagner-Hartl, and K. Moeller, ‘Traditional versus Neural Network Classification Methods for Facial Emotion Recognition’, Current Directions in Biomedical Engineering, vol. 7, no. 2, pp. 203–206, Oct. 2021, doi: 10.1515/cdbme-2021-2052.
25. H. Arabian, V. Wagner-Hartl, and K. Moeller, ‘Network Architecture Influence on Facial Emotion Recognition’, In Press.
26. H. Arabian, V. Wagner-Hartl, and K. Moeller, ‘Transfer Learning in Facial Emotion Recognition: Useful or Misleading?’, In Press.

German Medical Natural Language Processing – A Data-centric Survey

Torsten Zesch[1] and Jeanette Bewersdorff[2]

[1] Computational Linguistics, CATALPA – Center for Advanced Technology-Assisted Learning and Predictive Analytics,
FernUniversität in Hagen, Germany
torsten.zesch@fernuni-hagen.de
[2] RTG WisPerMed, University of Duisburg-Essen, Germany
jeanette.bewersdorff@uni-due.de

Abstract. Even though AI in general, and NLP in particular, has made a lot of progress in recent years, the impact on the processing of medical written data has so far been limited. We argue that this is mainly because publicly available data is scarce in the medical domain and thus provide an overview of available data sources as well as strategies to overcome data scarcity. We also discuss de-identification approaches and possible challenges when working with de-identified data. Finally, we give an overview of available German NLP models for the medical domain and discuss domain adaptation as a way to transfer models from a specific application area to another.

Keywords: language technology; medical NLP; German; datasets; domain adaptation

1 Introduction

Making sense of written medical data (e.g. from electronic patient records or laboratory analyses) is still a major challenge that gets even more difficult if we want to tackle languages other than English [35]. Medical NLP gained a lot of attention in recent years, e.g. in the form of re-occurring challenges like the National NLP Clinical Challenges (n2c2).[1] However, only very few medical datasets get released after being created, mainly because medical data contains sensitive personal information. When data is not available or cannot be shared, it has been proposed to instead share the resulting models [9, 19], however still have to care about model inversion attacks [9, 11], especially for overparametrized recent neural models. As a result, only very few datasets or models are available for public use. We thus give an comprehensive overview of the (very few) data sources in German medical NLP and discuss strategies to overcome data scarcity.[2]

2 Data Acquisition in Medical NLP

Even though AI in general, and NLP in particular, has made a lot of progress recently, the impact on the processing of medical written data has so far been limited. We argue that this is mainly because (as has been noted before for example by Lohr et al. [30]) publicly available data is scarce in the medical domain. To understand why, it is helpful to have a closer look at the data collection process.

Figure 1 presents an overview of the process of collecting or creating medical data. The origin of data can be divided into two sources: real data and synthetic data. **Real data** is collected as part of the patient treatment process in form of clinical notes and referral letters. In most cases, **de-identification** [24, 31, 37, 38] is required, as all personal information that may be contained in real data has to be removed. It is crucial to ensure that at no point a link between the data and the respective patient it originates from can be made. An alternative (that does not require de-identification) is using **synthetic data**. It can either be manually created (writing medical documents imagining patients and cases) or even automatically generated [29]. A natural source of manually created synthetic data are made-up reports and case studies written by medical professionals for educational purposes and published in medical textbooks [30], but also fictitious data written for the purpose of training a specific model [21]. The advantage of synthetic data, that it can be freely distributed without data protection issues, is countered by the looming question whether it closely enough resembles real data.

[1] https://n2c2.dbmi.hms.harvard.edu/

[2] Our focus on German is based on research within the WisPerMed DFG research training group ('Knowledge- and data-based personalization of medicine at the point of care', https://www.uni-due.de/grk_wispermed/grk_wispermed.php). The RTG combines the research expertise of Dortmund University of Applied Sciences and Arts, University of Duisburg-Essen, University Medical Center Essen and FernUniversität in Hagen. The overarching goal of the RTG is to make the knowledge contained in various data formats available and usable at the point of treatment for concrete individual therapy decisions. As a prototypical use case, the RTG is focusing on the treatment of malignant melanoma. All usable patient documentation in form of clinical notes is exclusively available in German.

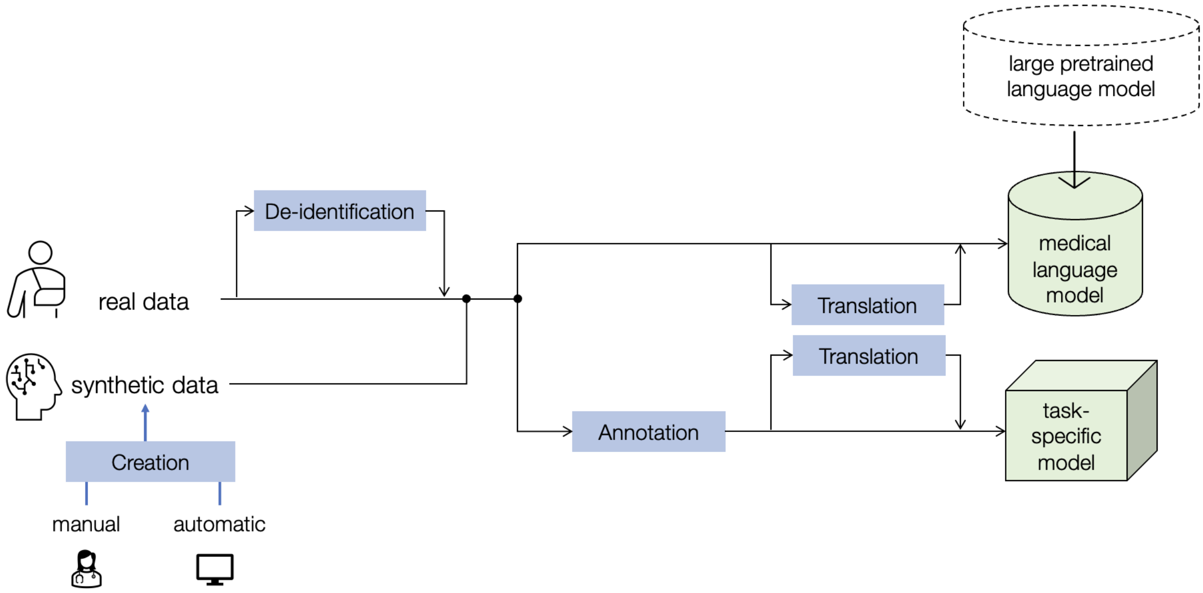


Fig. 1: Overview of collecting and creating data for medical NLP

As there is very little data in any given language, one might use **translation** to convert either real or synthetic data into the target language [13]. Finally, the raw data can be used to train a general purpose medical language model or the dataset needs additional **annotation** so that one can train a task-specific model.

We now give a more detailed overview of the individual steps involved in collecting and creating data for medical NLP.

2.1 De-identification

When working with *real data*, de-identification (also called anonymization) needs to be performed, before the data can be used. De-identification removes or replaces sensitive information such as the patient's name, their phone number, or address. Sensitive information items are collectively called *protected health information* (PHI) in the literature. For an overview of PHI types see Dernoncourt et al. [7]

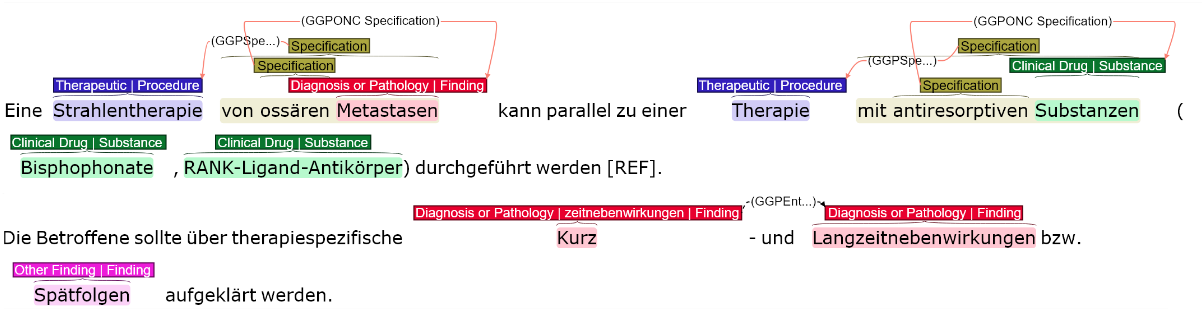
For English, several de-identification approaches have been evaluated, for example [34, 45], especially as part of the i2b2 de-identification challenges [43].[3] Similar approaches have also been adapted to German [24, 38].

Table 1: Examples of different de-identification approaches, adapted from Berg et al. [2]

ORIGINAL	Mr. John Berry was prescribed 400mg of Ibuprofen today.
PSEUDO	Mr. Harry Miller was prescribed 400mg of Ibuprofen yesterday.
. CLASS	Mr. {FirstName} {LastName} was prescribed 400mg of Ibuprofen {Date}.
MASK	Mr. XXX XXX was prescribed 400mg of Ibuprofen XXX.
REMOVE	-

Removing PHI from documents might however decrease the performance of NLP models. Berg et al. [2] found that the impact is significant and strongly depends on the de-identification strategy. Table 1 shows examples for the different methods. ORIGINAL shows the text as it was found in real medical data. The PSEUDO strategy replaces PHI with surrogates (i.e. another name/date/number etc.) that are as close to the intended meaning as possible depending on the context. For example, when replacing the age of a patient, granularity might be in decades for adults but that would not work well for young children where it makes quite a difference if they are 1 or 9 years old. CLASS replaces the PHI with a class marker, e.g. {FirstName} for given names like John or Mary. MASK hides the PHI by replacing it with some masking character, e.g. *X* or *#*. Finally, REMOVE just deletes any sentence

[3] The de-identification pipeline used for preparation of the i2b2 challenge is described in Stubbs and Özlem Uzuner [44].



Eine Strahlentherapie von ossären Metastasen kann parallel zu einer Therapie mit antiresorptiven Substanzen (Bisphophonate , RANK-Ligand-Antikörper) durchgeführt werden [REF].

Die Betroffene sollte über therapiespezifische Kurz - und Langzeitnebenwirkungen bzw. Spätfolgen aufgeklärt werden.

Fig. 2: Example of an annotated text from the GGPONC corpus created using the INCEpTION annotation platform (Source: `https://inception-project.github.io/use-cases/ggponc/`).

containing a single PHI from the corpus, which might be an unacceptable strategy if the the rate of PHIs in a text is high.

The study by Berg et al. [2] found that PSEUDO has the least impact on downstream task performance, while (unsurprisingly) REMOVE dramatically reduces performance (with the other two strategies in-between). However, at the same time it can be argued that PSEUDO has the weakest protection level and that surrogates have to be carefully designed to ensure de-identification.

So in de-identification, we not only have to find PHI, but also make an informed decision on how to retain as much information as possible, without compromising the de-identification itself.

2.2 Creating Synthetic Data

Synthetically created datasets have mainly been used in the domain of structured data [47]. The main goal is to transfer statistical properties (i.e. dependencies and distributions within the real data) to the synthetic data. While these methods cannot be used to create synthetic *texts*, other methods have been proposed for this purpose. Libbi et al. [29] compare an LSTM and a transformer-based generative model for creating synthetic medical care reports in Dutch. Guan et al. [15] propose to use generative adversarial networks (GANs) to generate Chinese electronic medical records.

As those methods have to be trained, at least some non-synthetic data is always required which could lead to a cold-start problem. It is also unclear, whether the automatically generated synthetic data is of high enough quality to be used as a replacement for manual synthetic data [15]. Another issue is that the generative model might leak PHI from the training data into the synthetic data [29].

2.3 Annotation

Real or synthetic data in its raw form is in most cases not enough to train a task-specific NLP model. What is also needed is *annotations* on the data, i.e. some kind of markup or codes that provide additional information that cannot be derived in some obvious fashion directly from the text. Usually this annotation process is thus performed manually. Figure 2 gives an example, where data from the GGPONC corpus [3] was annotated with SNOMED-CT classes classes using the INCEpTION platform [23].[4]

Some annotations are related to the linguistic properties, e.g. token, sentence, POS, or negation scope [48]). They do not directly correspond to a use case, but support downstream medical NLP tasks like concept extraction (e.g. findings, symptoms, drugs, diseases), relation extraction (e.g. adverse effects), or multi-label document classification (e.g. according to a scheme like ICD-10).[5] For a more comprehensive overview of different NLP tasks see for example Névéol et al. [35].

2.4 Translation

As so few datasets are available, it is an interesting strategy to translate data from other languages. If done manually, a high quality corpus can be created at very high costs, but it remains an open question whether the time-consuming effort to translate a dataset manually actually pays off or whether the synthetic creation or collection of new

[4] `https://inception-project.github.io/`

[5] `https://www.who.int/standards/classifications/classification-of-diseases`

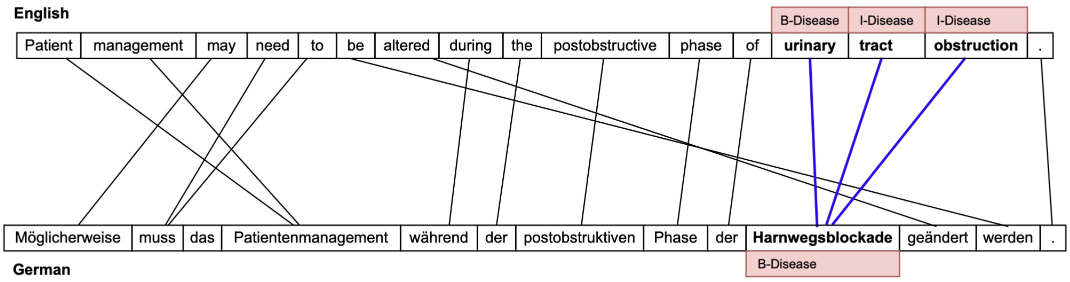


Fig. 3: Example of aligning original and translated sentence for annotation transfer (Figure by Schäfer et al. [41]).

data in the target language would be more efficient. Automatic machine translation can be used instead, which is much cheaper. If a remote translation service is used, only already de-identified data can be translated; otherwise, sensitive information could be disclosed. A work-around is to use a locally installed translation service, but the translation quality is usually lower. In any case, the automatic translation step is likely to introduce mistakes that will lower the quality of downstream task trained on that data [18].

When already annotated data is translated, a major challenge is to correctly transfer the annotations. Figure 3 shows an example provided by Schäfer et al. [41]. As the tokens do not align one-to-one between languages, it is challenging to correctly transfer the original annotation.

3 German Medical Corpora and Datasets

Table 2 gives an overview of the current situation for German medical corpora. A major distinction is whether a corpus is publicly available (upper part of the table) or not (lower part). The table is sorted by publication year, meaning that all more recently published corpora are available, while all older ones are not. Hopefully this represents a trend towards Open Science. As can be seen in the 'Origin' column, two of the available corpora are synthetic (GGPONC and JSYNCC), one is translated from English (GERNERMED) and only one contains real data (BRONCO150). The latter one is also the smallest, as it was manually de-identified. As medicine is a very broad field, the corpora cover a lot of domains (like radiology, surgery, or dermatology) and document types.

We now describe the four publicly available corpora in more detail:

GGPONC (German Guideline Program in Oncology NLP Corpus) is a corpus[6] consisting of 30 medical guidelines related to oncology. The corpus was created by Borchert et al. [3] and contained in its first release 25 guidelines. Version 2.0 added 5 more guidelines in 2022. The guidelines were manually annotated with SNOMED-CT classes (*finding, substance, procedure*). The corpus contains over 1.8 million tokens. More than 200,000 entities were annotated. As no sensitive data is contained, the corpus is publicly available for research purposes upon request.

BRONCO150 the Berlin-Tuebingen-Oncology Corpus[7] was released in 2021 by Kittner et al. [22]. It contains a selection of manually de-identified sentences from 150 discharge summaries, collected from melanoma or hepatocelluar carcinoma patients at Universitätsklinikum Tübingen and Charité Berlin. The full corpus consists of 200 documents, but only 150 were released (thus the name BRONCO150). The remaining 50 documents are disclosed for unbiased evaluation of future models.

To further increase the data protection, the order of all sentences from the 150 documents is randomized, so that a reconstruction of documents is nearly impossible. The resulting corpus contains 67,456 tokens. Annotations were made for diagnoses (following ICD10), treatments (following OPS) and medications (following ATC). To access the corpora, a user-agreement has to be signed.

GERNERMED is a translated corpus created by Frei and Kramer [13]. The English source data was taken from the 2018 ADE and medication extraction challenge (n2c2, Track 2) [20]. It contains 303 annotated discharge summaries with overall 172,695 tokens. The English source text was translated using the pretrained *transformer.wmt19.en-de* model from the Facebook fairseq model architecture [36]. Named entity annotations (drug, route, strength, frequency, duration, form, and dosage) were mapped to the respective positions in the translated documents using *FastAlign* [8], an unsupervised method for the word alignment between two

[6] https://www.leitlinienprogramm-onkologie.de/projekte/ggponc-deutsch/
[7] https://www2.informatik.hu-berlin.de/~leser/bronco/index.html

Table 2: Overview of German medical corpora and datasets.

Corpus	Year	Domain	Document Type	Origin	Available	# tokens $[10^3]$	# docs
GGPONC [3]	2022	oncology	guidelines	synthetic	yes	1800	-
BRONCO150 [22]	2021	oncology	discharge summaries	real	yes	70	150
GERNERMED [13]	2021	-	discharge summaries	real (translated)	yes	173	303
JSYNCC [30]	2018	surgery	mixed educational texts	synthetic	yes	313	867
3000PA [17]	2018	various	various	real	no	-	3000
Krebs et al. [25]	2017	radiology	reports	real	no	-	3000
Roller et al. [40]	2016	nephrology	clinical notes/ discharge summaries	real	no	158	1725
Cotik et al. [6]	2016	-	clinical notes/ discharge summaries	real	no	13	183
Lohr and Herms [32]	2016	surgery	intervention reports	real	no	266	450
Kreuzthaler et al. [26], Kreuzthaler and Schulz [27]	2016	dermatology	discharge summaries	real	no	-	1696
Toepfer et al. [46]	2015	cardiology	reports	real	no	-	140
Bretschneider et al. [4]	2013	radiology	reports	real	no	28	2713
Fette et al. [10]	2012	-	various	real	no	-	544
FraMed [49]	2004	-	various	real	no	100	-

languages. Additionally, as the original English data used masking for de-identification, pseudo names were introduced in the German translation to give it a more natural appearance. The corpus is available via Github.[8]

JSYNCC [30] is a synthetic German corpus.[9] It contains 867 documents with overall 312,784 tokens. It is based on ten medical e-books from different domains. The corpus itself consists of the various synthetic reports and discharge summaries contained in the e-books.[10] The conversion from the e-books to the final corpus is performed by a fully automated script, ensuring that everyone can recreate an exact copy of the corpus. The only prerequisite is that one needs to obtain the e-books.

4 Model and Tools for German

The list of publicly available German medical NLP tools and models is rather short:

German-MedBERT [42] is a finetuned version of the German BERT model and is publicly available on Huggingface.[11] For fine-tuning, medical reports and articles related to symptoms, diseases, and diagnoses were collected.

German NegEx by Cotik et al. [6] adapts the original NegEx [5], a regular expression algorithm developed to identify negations in English discharge summaries, to work on German clinical data by developing a publicly available German trigger set.[12] The trigger set is adapted to medical text by including also lexical items such as *gram-negativ* that are not found in standard trigger sets.

[8] `https://github.com/frankkramer-lab/GERNERMED`

[9] `https://github.com/JULIELab/jsyncc`

[10] We consider them as synthetic data, as they were written as examples for the text books.

[11] `https://huggingface.co/smanjil/German-MedBERT`

[12] `http://macss.dfki.de/german_trigger_set.html`

- **GERNERMED++** [12] is a named entity recognition model based on the dataset with the same name and the successor of the GERNERMED model [13]. It is available via Github and HuggingFace.[13]
- **JCoRe 2.0** the Julie Lab Component Repository by Hahn et al. [16] is a Java-based framework for creating NLP pipelines.[14] It also contains a German medical tokenizer, sentence splitter, and POS tagger [19].[15] The POS model was trained and evaluated on the non-publicly available FraMed corpus (see Table 2).
- **mEx** [39] provides models for German POS tagging, NE recognition, and relation extraction.[16] It is based on a non-publicly available dataset (see Table 2 Roller et al.), so these models are good examples of the 'share the model, if you cannot share the data' paradigm.

Beyond that, there are quite some scientific papers on processing German medical text, e.g. on sentence boundary and abbreviation detection [26, 27], negation detection [6] and negation scope detection [14], or grammars and parsing [4, 21, 28], but to the best of our knowledge none of those are currently available for public use.

To improve this situation, a possible short-term strategy could be to rely more on translation from other languages or on synthetic training data.

4.1 Domain Adaptation

The German-MedBERT [42] model is already an example of domain adaption, where a general language model is fine-tuned to medical language. Another example is by Kara et al. [21], who adapt a standard German dependency parser (Stanford Core NLP) with relatively little in-domain training data. They show that the transfer learning model outperforms both the standard model as well as a model directly trained on the in-domain data. Apart from this domain shift from the non-medical into the medical domain, there is also an intra-medical domain shift when a model that is developed in one domain (e.g. multiple myeloma [33], colorectal cancer [1], nephrology [40], or radiology [4]) is applied in another. Frei et al. [12] find that their GERNERMED++ model performs much worse on out-of-distribution samples. Recognition performance drops considerably from 0.95 to 0.87 F_1 when applied to another domain.

In summary, it remains unclear if annotation efforts in one domain can be leveraged in another, or if each domain needs to train there one models (with the corresponding effort and data availability problems). With the very few available datasets, it is also challenging to even test the domain transfer capabilities of existing models.

5 Summary

Availability of suitable data is currently the major bottleneck for German medical NLP.

Even if more datasets have been made publicly available in the last few years, the amount and diversity of public data is still not sufficient. Releasing more real data depends on reliable de-identification and bears legal and ethical risks. Thus, using synthetic data and translating data from other languages are increasingly used (but still under-explored) strategies.

Besides the data scarcity issue, there is also a lack of high-quality models being made public for research purposes. In theory, models can be more freely distributed, even if the underlying training data cannot. However, in a practical setting there is always the fear that sensitive data could be exposed through the model. High-quality medical NLP models also have potentially high commercial value which also impedes open distribution.

When no big and comprehensive medical corpus is available, resulting models are necessarily domain-specific. Thus, domain adaptation is another under-explored area with some promising results, but it still requires at least some in-domain data.

Availability of suitable data is going to remain the major bottleneck for German medical NLP.

Acknowledgements

This work was funded by a PhD grant from the DFG Research Training Group 2535 'Knowledge- and data-based personalization of medicine at the point of care (WisPerMed)'. This work was partially conducted in the framework of CATALPA - Center for Advanced Technology-Assisted Learning and Predictive Analytics of the FernUniversität in Hagen, Germany. We thank Andrea Horbach and Christin Seifert for their insightful comments and suggestions.

[13] https://github.com/frankkramer-lab/GERNERMED-pp

[14] https://julielab.de/Resources/JCoRe.html

[15] https://github.com/JULIELab/jcore-pipelines/tree/master/jcore-medical-pos-pipeline

[16] https://github.com/DFKI-NLP/mEx-Docker-Deployment

Bibliography

1. Becker, M., Kasper, S., Böckmann, B., Jöckel, K.H., Virchow, I.: Natural language processing of German clinical colorectal cancer notes for guideline-based treatment evaluation. International Journal of Medical Informatics **127**, 141–146 (2019), ISSN 1386-5056
2. Berg, H., Henriksson, A., Dalianis, H.: The Impact of De-identification on Downstream Named Entity Recognition in Clinical Text. In: Proceedings of the 11th International Workshop on Health Text Mining and Information Analysis, pp. 1–11, Association for Computational Linguistics, Online (Nov 2020)
3. Borchert, F., Lohr, C., Modersohn, L., Langer, T., Follmann, M., Sachs, J.P., Hahn, U., Schapranow, M.P.: GGPONC: A Corpus of German Medical Text with Rich Metadata Based on Clinical Practice Guidelines. In: Proceedings of the 11th International Workshop on Health Text Mining and Information Analysis, pp. 38–48, Association for Computational Linguistics, Online (Nov 2020)
4. Bretschneider, C., Zillner, S., Hammon, M.: Identifying Pathological Findings in German Radiology Reports Using a Syntacto-semantic Parsing Approach. In: Proceedings of the 2013 Workshop on Biomedical Natural Language Processing, pp. 27–35, Association for Computational Linguistics, Sofia, Bulgaria (Aug 2013)
5. Chapman, W.W., Bridewell, W., Hanbury, P., Cooper, G.F., Buchanan, B.G.: A simple algorithm for identifying negated findings and diseases in discharge summaries. Journal of Biomedical Informatics **34**(5), 301–310 (2001), ISSN 1532-0464
6. Cotik, V., Roller, R., Xu, F., Uszkoreit, H., Budde, K., Schmidt, D.: Negation Detection in Clinical Reports Written in German. In: Proceedings of the Fifth Workshop on Building and Evaluating Resources for Biomedical Text Mining (BioTxtM2016), pp. 115–124, The COLING 2016 Organizing Committee, Osaka, Japan (Dec 2016)
7. Dernoncourt, F., Lee, J.Y., Uzuner, O., Szolovits, P.: De-identification of patient notes with recurrent neural networks. Journal of the American Medical Informatics Association **24**(3), 596–606 (12 2016), ISSN 1067-5027, doi:10.1093/jamia/ocw156
8. Dyer, C., Chahuneau, V., Smith, N.A.: A simple, fast, and effective reparameterization of ibm model 2. In: NAACL (2013)
9. Faessler, E., Hellrich, J., Hahn, U.: Disclose Models, Hide the Data - How to Make Use of Confidential Corpora without Seeing Sensitive Raw Data. In: Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14), European Language Resources Association (ELRA), Reykjavik, Iceland (May 2014)
10. Fette, G., Ertl, M., Wörner, A., Klügl, P., Stoerk, S., Puppe, F.: Information Extraction from Unstructured Electronic Health Records and Integration into a Data Warehouse (01 2012)
11. Fredrikson, M., Lantz, E., Jha, S., Lin, S., Page, D., Ristenpart, T.: Privacy in Pharmacogenetics: An End-to-End Case Study of Personalized Warfarin Dosing. Proceedings of the ... USENIX Security Symposium. UNIX Security Symposium **2014**, 17–32 (aug 2014)
12. Frei, J., Frei-Stuber, L., Kramer, F.: GERNERMED++: Transfer Learning in German Medical NLP (2022)
13. Frei, J., Kramer, F.: GERNERMED – An Open German Medical NER Model (2021)
14. Gros, O., Stede, M.: Determining Negation Scope in German and English Medical Diagnoses, pp. 113–126. Brill, Leiden, The Netherlands (2014), ISBN 9789004258174
15. Guan, J., Li, R., Yu, S., Zhang, X.: A Method for Generating Synthetic Electronic Medical Record Text. IEEE/ACM Transactions on Computational Biology and Bioinformatics **18**, 173–182 (2021)
16. Hahn, U., Matthies, F., Faessler, E., Hellrich, J.: UIMA-Based JCoRe 2.0 Goes GitHub and Maven Central — State-of-the-Art Software Resource Engineering and Distribution of NLP Pipelines. In: Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16), pp. 2502–2509, European Language Resources Association (ELRA), Portorož, Slovenia (May 2016)
17. Hahn, U., Matthies, F., Lohr, C., Löffler, M.: 3000PA-Towards a National Reference Corpus of German Clinical Language. Studies in health technology and informatics **247**, 26–30 (01 2018)
18. Hayakawa, T., Arase, Y.: Fine-grained error analysis on English-to-Japanese machine translation in the medical domain. In: Proceedings of the 22nd Annual Conference of the European Association for Machine Translation, pp. 155–164, European Association for Machine Translation, Lisboa, Portugal (Nov 2020)
19. Hellrich, J., Matthies, F., Faessler, E., Hahn, U.: Sharing models and tools for processing German clinical texts. Studies in health technology and informatics **210**, 734–738 (2015), ISSN 1879-8365 (Electronic)
20. Henry, S., Buchan, K., Filannino, M., Stubbs, A., Uzuner, Ö.: 2018 N2c2 Shared Task on Adverse Drug Events and Medication Extraction in Electronic Health Records. Journal of the American Medical Informatics Association : JAMIA (2020)

21. Kara, E., Zeen, T., Gabryszak, A., Budde, K., Schmidt, D., Roller, R.: A Domain-adapted Dependency Parser for German Clinical Text. In: KONVENS (2018)
22. Kittner, M., Lamping, M., Rieke, D.T., Götze, J., Bajwa, B., Jelas, I., Rüter, G., Hautow, H., Sänger, M., Habibi, M., Zettwitz, M., Bortoli, T.d., Ostermann, L., Ševa, J., Starlinger, J., Kohlbacher, O., Malek, N.P., Keilholz, U., Leser, U.: Annotation and initial evaluation of a large annotated German oncological corpus. JAMIA Open **4**(2) (04 2021), ISSN 2574-2531, ooab025
23. Klie, J.C., Bugert, M., Boullosa, B., de Castilho, R.E., Gurevych, I.: The inception platform: Machine-assisted and knowledge-oriented interactive annotation. In: Proceedings of the 27th International Conference on Computational Linguistics: System Demonstrations, pp. 5–9, Association for Computational Linguistics (Juni 2018), veranstaltungstitel: The 27th International Conference on Computational Linguistics (COLING 2018)
24. Kolditz, T., Lohr, C., Hellrich, J., Modersohn, L., Betz, B., Kiehntopf, M., Hahn, U.: Annotating German Clinical Documents for De-Identification. Studies in health technology and informatics **264**, 203–207 (aug 2019), ISSN 1879-8365 (Electronic)
25. Krebs, J., Corovic, H., Dietrich, G., Ertl, M., Fette, G., Kaspar, M., Krug, M., Stoerk, S., Puppe, F.: Semi-Automatic Terminology Generation for Information Extraction from German Chest X-Ray Reports. Studies in health technology and informatics **243**, 80–84 (01 2017)
26. Kreuzthaler, M., Oleynik, M., Avian, A., Schulz, S.: Unsupervised Abbreviation Detection in Clinical Narratives. In: Proceedings of the Clinical Natural Language Processing Workshop (ClinicalNLP), pp. 91–98, The COLING 2016 Organizing Committee, Osaka, Japan (Dec 2016)
27. Kreuzthaler, M., Schulz, S.: Detection of sentence boundaries and abbreviations in clinical narratives. BMC Medical Informatics and Decision Making **15**(2), S4 (2015), ISSN 1472-6947
28. Krieger, H.U., Spurk, C., Uszkoreit, H., Xu, F., Zhang, Y., Müller, F., Tolxdorff, T.: Information Extraction from German Patient Records via Hybrid Parsing and Relation Extraction Strategies. In: Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14), pp. 2043–2048, European Language Resources Association (ELRA), Reykjavik, Iceland (May 2014)
29. Libbi, C.A., Trienes, J., Trieschnigg, D., Seifert, C.: Generating synthetic training data for supervised de-identification of electronic health records. Future Internet **13**(5) (2021), ISSN 1999-5903, doi:10.3390/fi13050136
30. Lohr, C., Buechel, S., Hahn, U.: Sharing Copies of Synthetic Clinical Corpora without Physical Distribution — A Case Study to Get Around IPRs and Privacy Constraints Featuring the German JSYNCC Corpus. In: Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018), European Language Resources Association (ELRA), Miyazaki, Japan (May 2018)
31. Lohr, C., Eder, E., Hahn, U.: Pseudonymization of PHI Items in German Clinical Reports. Studies in health technology and informatics **281**, 273–277 (may 2021), ISSN 1879-8365 (Electronic)
32. Lohr, C., Herms, R.: A Corpus of German Clinical Reports for ICD and OPS-based Language Modeling (05 2016)
33. Löpprich, M., Krauss, F., Ganzinger, M., Senghas, K., Riezler, S., Knaup, P.: Automated Classification of Selected Data Elements from Free-text Diagnostic Reports for Clinical Research. Methods of information in medicine **55**(4), 373–380 (aug 2016), ISSN 2511-705X (Electronic)
34. Neamatullah, I., Douglass, M., Lehman, L.: Automated de-identification of free-text medical records. BMC Med Inform Decis Mak 8, 32 (2008)
35. Névéol, A., Dalianis, H., Velupillai, S., Savova, G., Zweigenbaum, P.: Clinical Natural Language Processing in languages other than English: opportunities and challenges. Journal of Biomedical Semantics **9**(1) (2018)
36. Ott, M., Edunov, S., Baevski, A., Fan, A., Gross, S., Ng, N., Grangier, D., Auli, M.: fairseq: A fast, extensible toolkit for sequence modeling (2019)
37. Richter-Pechanski, P., Amr, A., Katus, H.A., Dieterich, C.: Deep Learning Approaches Outperform Conventional Strategies in De-Identification of German Medical Reports. Studies in health technology and informatics **267**, 101–109 (sep 2019), ISSN 1879-8365 (Electronic)
38. Richter-Pechanski, P., Riezler, S., Dieterich, C.: De-Identification of German Medical Admission Notes. Studies in health technology and informatics **253**, 165–169 (2018), ISSN 1879-8365 (Electronic)
39. Roller, R., Alt, C., Seiffe, L., Wang, H.: mEx - An Information Extraction Platform for German Medical Text. In: Proceedings of the 11th International Conference on Semantic Web Applications and Tools for Healthcare and Life Sciences (SWAT4HCLS'2018), Antwerp, Belgium (Dec 2018)
40. Roller, R., Uszkoreit, H., Xu, F., Seiffe, L., Mikhailov, M., Staeck, O., Budde, K., Halleck, F., Schmidt, D.: A fine-grained corpus annotation schema of German nephrology records. In: Proceedings of the Clinical Natural Language Processing Workshop (ClinicalNLP), pp. 69–77, The COLING 2016 Organizing Committee, Osaka, Japan (Dec 2016)
41. Schäfer, H., Idrissi-Yaghir, A., Horn, P., Friedrich, C.: Cross-Language Transfer of High-Quality Annotations: Combining Neural Machine Translation with Cross-Linguistic Span Alignment to Apply NER to Clinical

Texts in a Low-Resource Language. In: Proceedings of the 4th Clinical Natural Language Processing Workshop, pp. 53–62, Association for Computational Linguistics, Seattle, WA (Jul 2022)

42. Shrestha, M.: Development of a Language Model for Medical Domain. masterthesis, Hochschule Rhein-Waal (2021)
43. Stubbs, A., Kotfila, C., Uzuner, Ö.: Automated systems for the de-identification of longitudinal clinical narratives: Overview of 2014 i2b2/UTHealth shared task Track 1. J Biomed Inform. 2015;58 Suppl(Suppl):S11-S19 (2015)
44. Stubbs, A., Özlem Uzuner: Annotating longitudinal clinical narratives for de-identification: The 2014 i2b2/uthealth corpus. Journal of Biomedical Informatics **58**, S20–S29 (2015), ISSN 1532-0464, doi:https://doi.org/10.1016/j.jbi.2015.07.020, supplement: Proceedings of the 2014 i2b2/UTHealth Shared-Tasks and Workshop on Challenges in Natural Language Processing for Clinical Data
45. Szarvas, G., Farkas, R., Busa-Fekete, R.: State-of-the-art Anonymization of Medical Records Using an Iterative Machine Learning Framework. Journal of the American Medical Informatics Association **14**(5), 574–580 (09 2007), ISSN 1067-5027
46. Toepfer, M., Corovic, H., Fette, G., Klügl, P., Störk, S., Puppe, F.: Fine-grained information extraction from German transthoracic echocardiography reports. BMC Medical Informatics Decis. Mak. **15**, 91 (2015)
47. Tucker, A., Wang, Z., Rotalinti, Y., Myles, P.: Generating high-fidelity synthetic patient data for assessing machine learning healthcare software. Digit. Med. **3**(147) (2020)
48. Vincze, V., Szarvas, G., Farkas, R., Móra, G., Csirik, J.: The BioScope corpus: biomedical texts annotated for uncertainty, negation and their scopes. BMC Bioinformatics **9**(11), S9 (2008), ISSN 1471-2105, doi:10.1186/1471-2105-9-S11-S9
49. Wermter, J., Hahn, U.: An Annotated German-Language Medical Text Corpus as Language Resource (01 2004)

Risk-based Assessment of ML-based Medical Devices

Martin Haimerl

Innovation and Research Center Tuttlingen of Furtwangen University – Furtwangen University
Martin.Haimerl@hs-furtwangen.de

Abstract. The rating of risks is a crucial aspect for assessing the performance of medical devices. For machine learning (ML) based systems, this means that an integration of risks into the corresponding metrics should be addressed. The main goal of this paper is to demonstrate the effect when differences in the impact of certain errors is not adequately considered during the development of ML based systems, in particular when they refer to classification problems. An artificial model was utilized to demonstrate the different outcomes when considering different risk ratings. The differences were analyzed visually as well as quantitatively. As a result, a difference of up to 50% was obtained for the total outcome, when a ratio of 4.0 between the types of risks was assumed. This demonstrates that differences in risk impact should be systematically considered and integrated into the associated metric, when assessing the performance of ML based medical devices.

Keywords: Machine Learning; Evaluation; Performance Metrics; Medical Devices; Risk Management.

1 Introduction

The assessment of machine learning (ML) based systems strongly depends on the criteria which are used for the training, validation, and testing of the developed model. These criteria have to provide a score or metric how well the models perform. For supervised learning approaches, the agreement between the ground truth and the values predicted by the model is a core aspect of such metrics. For classification problems, this means the rates of successful assignment to the particular classes. Standard techniques for the assessment of binary classification are accuracy rates, false positive (FP) / false negative (FN) rates, or receiver operator characteristics (ROC curves) (cf. [1] for an overview of common assessment criteria). These techniques refer to the probabilities how often correct respectively incorrect assignments to the classes occur in the training, validation, and testing data sets.

These techniques are well established. However, they are often applied in a very generic way without considering the specific context of its application. In particular for medical applications, such a generic approach has considerable limitations. The impact of different types of errors needs to be considered in an application-specific way. An FN (e.g. missing the presence of a tumor illness) may be more critical than an FN (false alarm for the disease, which can be double-checked subsequently). From the perspective of the patient health, an FN often leads to substantially worse outcomes. Such risks should be considered in a dedicated way when assessing the overall performance of a classifier. This means that not only accuracies and probabilities have to be taken into account but also the impact of the different types of errors. This leads from a primarily probability-based to a risk-based approach for the assessment of ML-based classification systems. Similar approaches can also be found under the names cost curves [2], utility curves [3], or decision curves 4].

For demonstrating the impact of different types of errors, this paper develops a risk-based approach for the assessment of medical devices and analyzes its differences when comparing it to methods using probability-based measures. A parametric model is used for the distributions to systematically analyze the changes in outcome. Assessment scores are developed and analyzed which address individual risks for a single patient as well as aggregate risks representing the overall performance of the device.

2 Materials and Methods

In this paper, a generic setup is used with a classifier F predicting the binary outcome $Y \in \{0,1\}$ from a set of input features X, i.e. the prediction is performed according to $\hat{Y} = F(X)$. This prediction is performed on a set of test data (X_i, Y_i), where Y_i are considered as the ground truth, i.e. the correct classification values for the input values X_i. The classifier is considered to depend on a threshold s, i.e. it predicts a 1 if and only if a certain score value $S = S(X)$ is above the threshold s. Thus, a particular instance of the classifier can be represented by a binary-valued function $F(s, X)$ which includes the threshold s as a parameter. In this paper, we utilize an artificially constructed error distribution to demonstrate the behavior of performance metrics when certain parameters get changed. This means, that we assume that the false positive $FPR(s)$ and false negative rates $FNR(s)$ are given by a parametric function. We use modified Gaussian functions of the form $FPR(s) =$

$(1-s)\cdot\exp\left(\frac{s^2}{\sigma_{FP}}\right)$ and $FNR(s) = s\cdot\exp\left(\frac{(1-s)^2}{\sigma_{FN}}\right)$, for this purpose. The terms $(1-s)$ and s modify the Gaussians in a way that $FPR(1) = FNR(0) = 0$.

As a next step, a risk model is constructed which assigns certain “costs” to the different types of errors FP and FN. These costs reflect the risks or other associated costs which are caused by the particular type of error. Subsequently, they are named risk scores and denoted by R_s. In terms of conditional probabilities $P(\hat{Y}|Y)$, the individualized risk $IR(s)$ can be calculated as the expected risk for a particular individual, i.e. $IR(s) = E\big(R_s(\mathrm{FP}) + R_s(\mathrm{FN})\big) = E\left(R_s\left(P\big(\hat{Y}=1|Y=0\big)\right) + R_s\left(P\big(\hat{Y}=0|Y=1\big)\right)\right)$, where $E(\cdot)$ denotes the expectation value. In this paper, we do not include risks which depend on the threshold levels. Thus, the risk scores boil down to $IR(s) = E\big(R_s(\mathrm{FP}) + R_s(\mathrm{FN})\big) = c_{FP}\cdot FPR(s) + c_{FN}\cdot FNR(s)$, where c_{FN} and c_{FR} are constants reflecting the impact of the particular type of error. Further on, not the absolute values but only the relationship between the costs of FP and FN matters. Thus, the value of c_{FP} can be set to 1, without loss of generality. Subsequently, the equation reduces to $IR(s) = E\big(R_s(\mathrm{FP}) + R_s(\mathrm{FR})\big) = FPR(s) + c_{FN}\cdot FNR(s)$. Subsequently, c_{FN} is called risk ratio.

So far, all these curves reflect an individual risk, since they only take into account the general error rates for a particular individual as well as the impact such kind of an error has. The analysis does not encounter a situation where the number of positive $(Y=1)$ and negative $(Y=0)$ cases differ and where an aggregate risk score should be used as a reference. The aggregate risk score $AR(s)$ sums up all the individual risks for the given data set (X_i, Y_i). If we again assume an individual risk score of $c_{FP}=1$ for FP and c_{FN} for FN, this overall risk score is calculated as $AR(s) = FP(s) + c_{FN}\cdot FN(s)$ where $FP(s) = \left|\left\{i\middle|F(s,X_i) = \widehat{Y_i} = 1, Y_i = 0\right\}\right|$ is the number of false positives and $FN(s) = \left|\left\{i\middle|F(s,X_i) = \widehat{Y_i} = 0, Y_i = 1\right\}\right|$ the number of false negatives for a fixed threshold s. Again, only the ratio $q = \frac{FN(s)}{FP(s)}$ between $FP(s)$ and $FN(s)$ matters, since we do not focus on absolute levels of risk values but only on relationships between them. This can also be written as $q = \frac{FNR(s)}{FPR(s)}$. Based on this, the aggregate risk score can be calculated as $AR(s) = FPR(s) + q\cdot c_{FN}\cdot FNR(s)$. Contracting q and c_{FN} to a single factor $\tilde{c}_{FN}$, we see that $AR(s)$ has the same form as the individual risk, i.e. $FPR(s) + \tilde{c}_{FN}\cdot FNR(s)$.

3 Results

Based on the described approach, the model was first applied to a test scenario where σ_{FN} and σ_{FP} were both set to 0.3. For the risk ratio c_{FN}, the values were set to $0.25, 0.5, 1.0, 2.0$, and 4.0. The results are shown in Fig. 1. On the left side, the model is shown with the corresponding $FPR(s)$ and $FNR(s)$ values. The diagram on the right side demonstrates the impact of different risk ratios c_{FN} on the overall outcome for the individual risks. The risks associated with both types of errors are balanced out where the curves have their minimum. For $c_{FN} = 1.0$, this is exactly at $s = 0.5$ as indicated by a short line for the black curve. This symmetry appears since the error distributions are symmetric between FPR and FNR. The remaining curves show the situation when the risk ratio c_{FN} takes on the other values $0.25, 0.5, 2.0$, and 4.0. For each curve, the minimum risk is indicated by a short line. The position changes, since it depends on the relationship of the impact for the different types of errors.

Additionally, the dotted lines show the reference $s = 0.5$, where the curve with equal risks between FP and FN, i.e. $c_{FN} = 1.0$, has its minimum. The intersection between the dotted line and the particular curve reflects the risk value which would have been obtained when the optimization would have been performed solely according to the error rates, i.e. without considering the risk factors. It can be seen that in the cases $c_{FN} \neq 1.0$, the minimum lies in a region where the curves considerably decay or increase. This demonstrates a deviation between the $c_{FN} = 1.0$ assumption and the actual risk ratio. This effect is more apparent with increasing difference between the actual c_{FN} and the balanced case $c_{FN} = 1.0$. The same basic model applies to the cases of the aggregate risk. However, the outcomes have to be interpreted according to a replacement of c_{FN} by $\tilde{c}_{FN}$. Thus, the basic analysis can be transferred to the aggregate risk case.

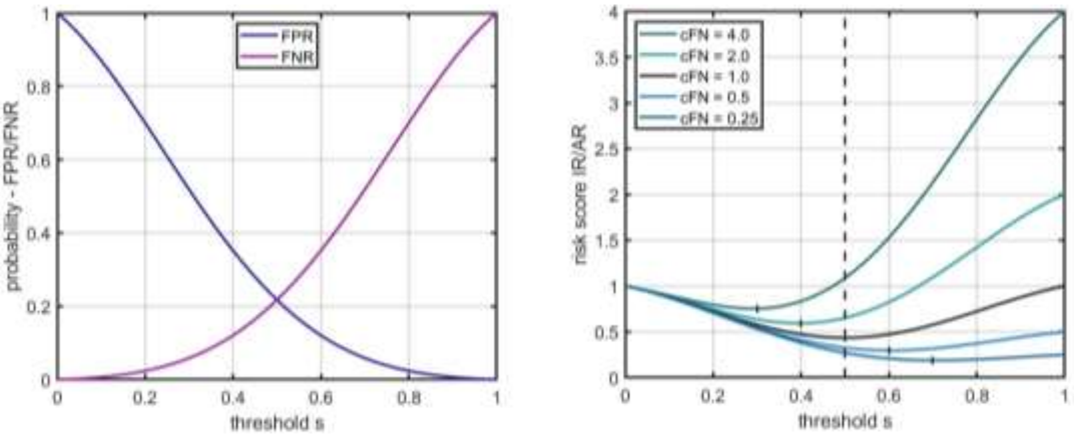


Fig. 1. Left side: Artificial model of error distributions, i.e. $FPR(s)$ and $FNR(s)$ in dependence of the threshold s. The model is based on modified Gaussian functions of the form $FPR(s) = (1 - s) \cdot \exp\left(\frac{s^2}{\sigma_{FP}}\right)$ and $FNR(s) = s \cdot \exp\left(\frac{(1-s)^2}{\sigma_{FN}}\right)$, where $\sigma_{FP} = \sigma_{FN} = 0.3$. Right side: Risk scores $IR(s)$ respectively, $AR(s)$ for the same casse, when the risk ratio c_{FN} is varied ($c_{FN} = 0.25, 0.5, 1.0, 2.0,$ and 4.0). The short line shows the minimum for the particular curve. The dotted line represents the reference $s = 0.5$, where the curve with equal risks between FP and FN, i.e. $c_{FN} = 1.0$, has its minimum.

Fig. 2 shows a comparison when different parameter setting for the artificial model are applied. This includes three scenarios with $\sigma_{FN} = \sigma_{FP} = 0.2, \sigma_{FN} = \sigma_{FP} = 0.3$, and $\sigma_{FN} = \sigma_{FP} = 0.4$. On the left side, the corresponding ROC curves are shown to provide an overview about the particular probability-based model performance. In the right diagram, the impact of the particular parameters on the risk-based approach is visualized. It is shown how much higher the resulting risk score would have been, when c_{FN}=1.0 would have been assumed instead of the suited risk ratio. The relative difference exceeds 50% for the case $\sigma_{FN} = \sigma_{FP} = 0.4$ and the risk ratios $c_{FN} = 4.0$ as well as $c_{FN} = 0.25$. On the c_{FN}-axis, a logarithmic scaling was applied since this better reflects that c_{FN} is a ratio parameter. In this logarithmic scaling, all the curves show a symmetric appearance with respect to the $c_{FN} = 1.0$ axis. The difference substantially decays when c_{FN} gets closer to 1.0. Additionally, it can be recognized that the differences decrease slightly when the parameters σ_{FP} / σ_{FN} decrease and subsequently the area under the ROC curve (i.e. the AUC value) increases. The AUC is a standard probability-based measure for assessing the performance of a classifier (see 1).

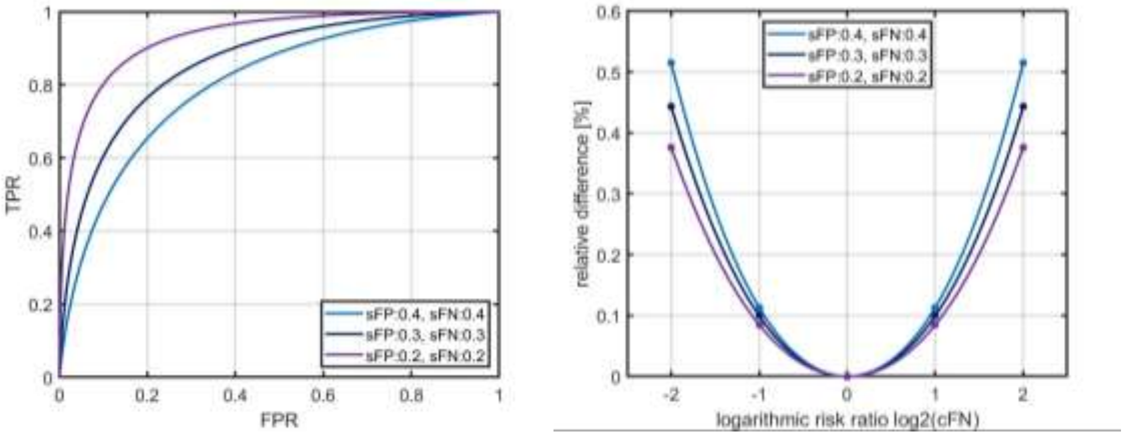


Fig. 2. Variation of the parameters of the artificial model including $\sigma_{FN} = \sigma_{FP} = 0.2, \sigma_{FN} = \sigma_{FP} = 0.3$ (i.e. same case as in Fig. 1), and $\sigma_{FN} = \sigma_{FP} = 0.4$. In the diagram, σ_{FP} is named sFP and σ_{FN} sFN. Left side: ROC curves for the particular cases. Right side: Increase in risk values, when a risk adaption would not have been performed, i.e. the standard probability based threshold $s = 0.5$ would have been used. On the vertical axis, this value is shown as a relative increase in comparison to the true optimum risk value. On the horizontal axis, the used risk ratios c_{FN} respectively $\tilde{c}_{FN}$ are shown. A logarithmic scaling is used for this axis.

4 Discussion

Using an artificial model for the error distribution, this paper demonstrates the relationships between a pure probability-based assessment of ML models on the one hand and risk-based approaches (individual as well as aggregate risks) on the other hand. It was demonstrated that substantial differences occur when risk factors are not addressed adequately. The difference in resulting risk scores goes up to 50% in this simple setting. According to the applicable regulations like the Medical Device Regulation (MDR) and associated standards like ISO 14971, the risks of medical devices should be adequately managed during device development. Following these requirements, appropriate scores should be applied to assess and optimize the outcome of ML-based devices or components. This is not included in standard metrics for ML-based classifiers, like accuracy, FP/FN rates, or also ROC curves. This could only be achieved using risk-based approaches. For this purpose, this paper provides insights regarding the potential approaches as well as the behavior when applying different risk ratios.

An additional question in this context is whether individual or aggregate risks should be used, i.e. whether the risk should be addressed for an individual patient or across the entire population / number of cases. In the latter case, the distribution of the error cases plays a central role. Such an approach is included in ISO 14971, representing the relevant risk management standard for medical devices. Not only the severity but also the likelihood of the hazards / harms for the patient have to be included according to 5. From this perspective, the application of an aggregate risk-based approach is applicable and should be pursued. One further challenge is the assessment of proper “costs” and likelihoods for the particular types of errors in a quantitative way. However, ISO 14971 allows to basically use semi-quantitative approaches for risk management. This means, that probabilities as well as the level of harm (or “costs”) may be categorized. Thus, the rating could be addressed and integrated into the models in this way. Adjustments towards true quantitative ratings could be approached during the lifetime of the ML based system, i.e. when enough data is gathered during the operation of the device in real settings. Of course, the rating of different types of errors is strongly application specific and often a balancing of different types of impacts is difficult to justify. It also includes challenging ethical questions. More research in this direction is required to provide proper approaches for achieving an overall optimal result.

This study has some limitations. First of all, the model does not reflect a real case scenario. Thus, future research should include an analysis of the behavior in concrete applications using the actual error distributions. This would also enable to better address the derivation of appropriate risk ratios for the particular applications. Further on, our model includes some simplifications. It assumes that the risk ratios are constant, i.e. do not depend on the threshold s. This may not be adequate in real case scenarios when the severity of the risks may change between clear diagnoses (high values for s) and ambiguous cases (with s in the mid range). For example, a lower rate of pathological substances in a diagnostic test may be associated with less severe courses of an illness. In the current paper, we do only apply the risk scores during threshold selection and not within the model training, i.e. directly in the optimization procedure. Thus, the analysis could be extended in this direction, in the future. This also addresses general steps to approach real case scenarios. For the future of ML-based medical devices, it will be important that a more consequent understanding is achieved how risk factors have to be incorporated in the development of ML-based systems.

5 Conclusion

In summary, this study provides a systematic analysis of the behavior of ML based systems with respect to differences in risk ratings, utilizing an artificial model. It demonstrates that standard metrics have substantial deficiencies when they are applied to ML systems without any adjustments towards the impact of different error types. The systematic integration of risks into the metrics is a crucial point to achieve an appropriate balancing of risk impact. Further steps are necessary to systematically integrate such approaches into the validation of ML based medical devices, in the future. In particular, this is related to the question how to obtain proper ratings for the risks and how to combine performance metrics with risk management requirements in a compliant way with respect to regulatory requirements.

References

1. Tharwat A. Classification assessment methods. Applied Computing and Informatics, 17(1) (2021) 168–192.
2. Drummond C, Holte R. Cost curves: An improved method for visualizing classifier performance. Machine Learning 65 (2006) 95–130.
3. Baker S, Cook N, Vickers A, Kramer B. Using relative utility curves to evaluate risk prediction. Journal of the Royal Statistical Society: Series A. 172(4) (2009) 729–748.
4. Rousson V, Zumbrunn T. Decision curve analysis revisited: overall net benefit, relationships to ROC curve analysis, and application to case-control studies. BMC Medical Informatics and Decision Making 11 (2011) 45.
5. International Organization of Standards. ISO 14971:2019: Medical devices – Application of risk management to medical devices. (2019).

Specification of neck muscle dysfunction through digital image analysis using machine learning

Filip Paskali[1], Angela Dieterich[2] and Matthias Kohl[3]

[1]Institute of Precision Medicine, Medical and Life Sciences, Hochschule Furtwangen, Villingen-Schwenningen
F.Paskali@hs-furtwangen.de
[2] Physiotherapie, Fakultät Gesundheit, Gesellschaft, Hochschule Furtwangen, Studienzentrum Freiburg
Angela.Dieterich@hs-furtwangen.de

Abstract. Everyone has or will have experience some degree of neck pain. Typically, neck pain is associated with the sensation of tense, tight or stiff neck muscles. However, it is unclear whether the neck muscles are objectively stiffer with neck pain. Some investigations documented higher stiffness of the neck muscles with neck pain, while others did not find differences compared to asymptomatic study participants. This is a cross-sectional, observational study that analyses shear wave elastography data obtained from 38 women. In this study, we trained machine learning models that can classify the shear wave elastography images at the level of an expert. Knowledge on a potentially increased objective stiffness of the neck muscles is important when related to diagnosis or therapeutic decisions. Moreover, such an automated approach enables a computed image analysis, which may provide new insights of the physiological properties of the neck muscles in individuals suffering from neck pain.

Keywords: Neck pain 1; Shear Wave Elastography 2; Ultrasound 3; Image Analysis 4; Machine Learning 5.

1 Introduction

Almost everyone has or will have experience some degree of neck pain [1]. The one-year prevalence of disabling neck pain has been estimated between 1.7% and 11.5% [2], and the risk is especially high in middle-aged woman who have already experience with neck pain [3]. Typically, neck pain is associated with the sensation of tense, tight or stiff neck muscles [4]. Therapeutic interventions often include measures to reduce neck muscle tone. However, it is unclear whether the neck muscles are objectively stiffer with neck pain. Some investigations documented higher stiffness of the neck muscles with neck pain [5], [6], while others did not find differences compared to asymptomatic study participants [7], [8]. Knowledge on a potentially increased objective stiffness of the neck muscles is important when related to diagnosis or therapeutic decisions.

Ultrasound shear elastography is the superior and reliable modality for measurement of tissue stiffness [9], [10]. However, the methods to measure muscle stiffness vary between studies. The active and individual positioning of the region of interest has biasing potential. Some studies reported inter-session stiffness variability [11]. An automated image analysis might provide new insights into differences of the mechanical properties of the neck muscles between individuals with and without chronic neck pain.

Recently, there has been a growing number of publications focusing on the usage of the artificial intelligence algorithms for medical diagnostics in ultrasonography imaging. The machine learning algorithms have been employed for classification of chronic liver diseases [12], breast tumors [13], thyroid nodules [14], and prostate cancer [15]. On other hand, deep learning convolutional neural network algorithms have been used for segmentation of muscles, as well as automatic measurement of muscle thickness and muscle fat infiltration [16]–[18]. To the best of our knowledge, there has been no attempts to employ the machine learning algorithms in ultrasound elastography images for classification of neck pain.

The experimental work presented here provides a fully automatic investigation using machine learning techniques to examine in detail group differences in two-dimensional ultrasound elastography images measured in adult woman suffering from neck pain and women without symptoms.

2 Materials and Methods

2.1 Study

This is a cross-sectional, observational study that analyses data obtained from 38 women. The participants belong to two groups, 20 women suffering from chronic neck pain and 18 women without symptoms. The inclusion criteria for the pain group were non-specific pain longer than six months with symptoms duration of at least one week, neck stiffness sensation and Neck Disability Index higher than 10/50. On other hand, for the control group, no history of recurrent neck or low back pain or neck pain that affected the neck function or required treatment. Exclusion criteria were major circulatory, neurological, or respiratory disorders, pregnancy, cervical spinal surgery, participation in neck muscle training in past 6 months, body mass index higher than 30 and intake of medication that can affect muscle stiffness. On the days of the imaging, participants were asked to not use pain medication. As shown in the demographic summary (Table 1), the sample size, age, and body mass index of the two groups is comparable. At the same time, it is noticeable that the range of motion in flexion and extension, and maximal voluntary isometric contraction is limited in the pain group.

Table 1. Demographic summary: mean ± SD. Abbreviations: MVIC, maximal voluntary isometric contraction; NRS, numerical rating scale.

	Neck pain, n = 20	**Control, n = 18**
Age (years)	52.5 (12.0)	48.5 (9.0)
Body Mass Index (kg/m^2)	23.8 ± 3.2	22.2 ± 2.4
Range of Motion neck flexion	50.6° ± 12.4°	69.4° ± 10.8°
Range of Motion neck rotation (left + right)	115.5° ± 14.3°	145.5° ± 26.6°
MVIC neck extension (N)	56.3 ± 18.5	64.0 ± 19.2
Pain today (NRS)	3.6 ± 2.2	0
Pain duration (years)	8.0 (16.3)	-
Neck Disability Index % (scored 0% - 100%)	32.5% ± 12.3%	-

Participants performed four diverse activities including head lift from prone, stressful office work, balancing a weight (1 kg) on the head, and graded isometric neck extension under force levels of 12 N, 24 N, 36 N and 48 N. All activities were repeated and measured in three sessions. More information about the study cohort can be found in Dieterich et al. 2020 [7].

2.2 Image acquisition

Shear wave elastography images were obtained on the neck extensor muscle group 1 cm lateral to the spinous processes, the transducer centered at the C4 level in longitudinal orientation (Acuson S3000; Siemens, Germany, 9L4 linear transducer with 4 cm footprint). Maximal shear wave speed of 10m/s was set. Gain (14-20 dB), dynamic range (45-65 dB), and image depth 3.5-4.5 cm were adjusted for good visualization. In total, 1099 images were recorded. More information about the imaging in Dieterich et al, 2020 [7]

Each measurement produced three images, an ultrasound image; a color coded elastography image, indicating the shear wave velocity in the region of interest with range 0.5 -10 m/s; and a color-coded quality map presenting the quality of each pixel ranging from low to high quality. The images have a resolution of 1024 x 768 and are stored in BMP file format.

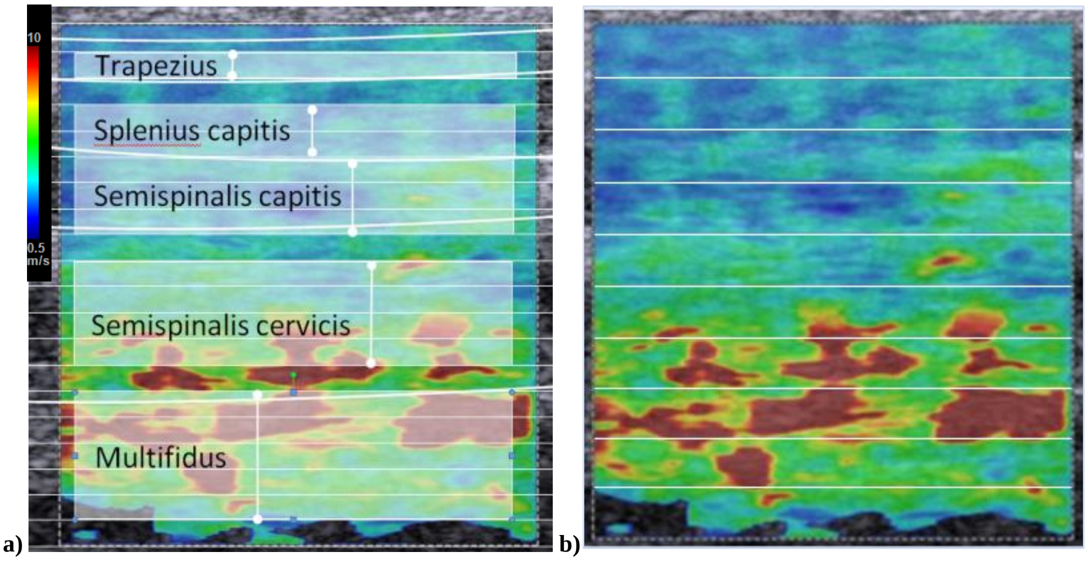


Fig 1. a) Color coded elastography shear wave image with muscle layer organization b) Color coded elastography shear wave image with representation of the horizontal segments.

2.3 Image Analysis

The image analysis and feature extraction was carried out using the free open-source programming language Python [19].

For the machine learning, 30 diverse features were handcrafted, extracting numeric information from the shear wave velocity color-coded images, as described below.

The first set of numerical features was computed by performing various statistical operations to the whole image, such as computing grand mean, median and standard deviation. Furthermore, the images were separated in three color channels, red, green, and blue. The statistical operations and grand sum were computed for each color channel of each image. The images were converted into grayscale color mode and same features were extracted.

For the second set of features, an inverse mapping was performed from RGB color mode to stiffness using the full range shear velocity scale (Figure 1a). The number of pixels in the images was quantified from three stiffness level categories: low level stiffness category with shear velocity between 0,5 – 3,67 m/s, medium level stiffness category with shear velocity between 3,67 – 6,84 m/s, and high level stiffness category with shear velocity between 6,84 – 10 m/s. Additionally, the mean area and the weighted average distance of high level stiffness areas from the top edge of the images was calculated. As weight, the size of the area was used.

The third set of features was obtained with Gabor filtering, which in several studies was used for texture classification [20]–[22]. In this study, we used mean and variance of the resulting convoluted image as features, using the following four different Gabor kernels (Fig. 2).

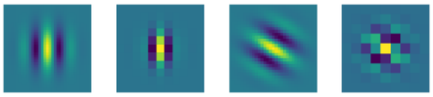


Fig 2. The Gabor kernels used for convolution

Finally, the images were divided in ten equal horizontal segments (Figure 1b) and the aforementioned operations were repeated for each of the segments leading to total number of 330 features. To the 330 features, the 14 activities performed by the participants, together with the imaging session number were added as dummy variables. Overall, this gives 345 features extracted from 1099 images.

2.4 Machine learning

A supervised binary classification was performed, using six classification machine learning algorithms from the package scikit-learn [23]. The main task was a prediction of one of the two target classes: participants with neck pain and participants without neck pain. The class label was assigned to the images before the machine learning. The training was repeated 50 times, with a new random train-test split, where the model was trained with 1049 images 10-fold cross validated and tested with 50 held-out images. Finally, in order to investigate the feature importance, the best performing model was trained with the whole dataset and feature importance were extracted.

3 Results

Table 2 displays the arithmetic mean and the confidence interval of performance metrics, as empirical quantiles (2.5%; 97.5%) computed from 500 cross-validation results.

Table 2. Summary of the cross-validation during training of the classification machine learning algorithms (mean, confidence interval 95%) Abbreviations: KNN – K-Nearest Neighbor; SVM – Support Vector Machine; AUC – Area Under Curve.

Metrics	KNN	Logistic Regression	Naïve Bayes	SVM	Decision Trees	Random Forest
Accuracy	0. 789 (0.785-0.792)	0.703 (0.700-0.707)	0.622 (0.618-0.625)	0.710 (0.706-0.714)	0.707 (0.703-0.711)	**0.810** (0.807-0.813)
Balanced Accuracy	0.789 (0.786-0.792)	0.701 (0.698-0.705)	0.620 (0.617-0.624)	0.709 (0.705-0.713)	0.706 (0.702-0.709)	**0.808** (0.804-0.811)
Precision	0.763 (0.759-0.768)	0.686 (0.682-0.691)	0.593 (0.588-0.597)	0.690 (0.686-0.695)	0.690 (0.685-0.695)	**0.815** (0.811-0.819)
Recall	**0.796** (0.791-0.801)	0.674 (0.668-0.680)	0.605 (0.599-0.612)	0.690 (0.684-0.696)	0.680 (0.674-0.686)	0.769 (0.763-0.775)
AUC Score	0.849 (0.846-0.852)	0.774 (0.771-0.778)	0.683 (0.678-0.687)	0.776 (0.773-0.780)	0.706 (0.702-0.709)	**0.895** (0.892-0.897)
Brier Score	0.157 (0.155-0.159)	0.201 (0.199-0.204)	0.364 (0.360-0.368)	0.196 (0.195-0.198)	0.293 (0.289-0.297)	**0.152** (0.151-0.153)

The trained models with 1049 images were used to predict classes of the 50 held-out images. The result of this evaluation is summarized in Table 3.

Table 3. The summary of the results obtained 50 evaluations with 50 held-out images of the classification machine learning models. Abbreviations: KNN – K-Nearest Neighbor; SVM – Support Vector Machine; AUC – Area Under Curve.

Metrics	KNN	Logistic Regression	Naïve Bayes	SVM	Decision Trees	Random Forest
Accuracy	0.792 (0.779-0.805)	0.695 (0.677-0.713)	0.619 (0.600-0.638)	0.697 (0.679-0.715)	0.724 (0.708-0.739)	**0.811** (0.795-0.827)
Balanced Accuracy	0.791 (0.777-0.805)	0.691 (0.673-0.710)	0.618 (0.599-0.638)	0.694 (0.675-0.713)	0.722 (0.706-0.739)	**0.807** (0.791-0.823)
Precision	0.763 (0.741-0.786)	0.673 (0.646-0.699)	0.590 (0.562-0.617)	0.668 (0.643-0.693)	0.699 (0.677-0.721)	**0.807** (0.786-0.828)
Recall	**0.802** (0.778-0.826)	0.669 (0.635-0.704)	0.612 (0.585-0.639)	0.689 (0.658-0.721)	0.714 (0.685-0.743)	0.780 (0.752-0.808)
AUC Score	0.854 (0.842-0.865)	0.757 (0.737-0.778)	0.677 (0.656-0.698)	0.760 (0.739-0.780)	0.722 (0.706-0.739)	**0.892** (0.879-0.905)
Brier Score	0.154 (0.146-0.162)	0.209 (0.199-0.220)	0.368 (0.349-0.387)	0.201 (0.194-0.208)	0.276 (0.261-0.292)	**0.150** (0.145-0.156)

An inspection of the data in Table 2 and 3 reveals that there is an overlap between confidence intervals of training and testing results, with slightly decrease in the performance during the testing step. Random Forest model outperforms other models in almost every performance metrics; hence it has the best performance during the training and testing step. The second-best performing algorithm is K-Nearest Neighbor.

The most important features extracted from the Random Forest model trained with the whole dataset are presented in Table 4. What is interesting in this table is that most of the represented features are coming from the deepest horizontal segment, that corresponds to the multifidus muscle closest to the spine.

Table 4. Top 20 most important features extracted from trained Random Forest model

	Features
1	blue_layer_value_sum_hsegment_1/10
2	blue_layer_value_sum_hsegment_2/10
3	red_layer_value_sum_hsegment_1/10
4	blue_pixels
5	gray_value_min_hsegment_10/10
6	gray_value_hsegment_9/10
7	red_layer_value_median_hsegment_10/10
8	gray_value_median_hsegment_10/10
9	red_layer_value_sum_hsegment_2/10
10	gabor_kernel4_mean_hsegment_10/10
11	gabor_kernel1_mean_hsegment_10/10
12	median_hsegment_10/10
13	green_layer_value_median_hsegment_10/10
14	gabor_kernel3_mean_hsegment_10/10
15	mean_hsegment_10/10
16	red_layer_value_mean_hsegment_10/10
17	gray_value_hsegment_3/10
18	gray_value_mean_hsegment_10/10
19	gray_value_hsegment_6/10
20	blue_layer_value_sum_hsegment_3/10

4 Discussion

Previous studies, when analyzing neck muscle pain, mainly focused on the superficial neck muscles [5], [6], [8], [24]. Dieterich et al, 2020 was the first study that included the deep muscles. Xie et al. 2019 reported a problem in assessment of the deep muscles, in terms of high variability [25]. Furthermore, Haldemann et al. 2009 also noted that the measurement in shear wave elastography is very sensitive to the orientation of muscle fibers [1]. Only few studies have found a correlation between neck muscle pain and stiffness levels in shear wave elastography images [5], [6], while other studies could not find significant difference [7], [8]. An initial objective of this research was to successfully train a machine learning model, that can predict with high accuracy the group of an elastography shear wave image. So far, the neck pain diagnostics is a complex task, based on the subjective sensation of the patient and the diagnostic experience of the trained physician [4]. A machine learning model that can classify with an accuracy of or better than a trained diagnostician will improve medical diagnosis and therapy.

In this study we trained and evaluated a machine learning model that can diagnose neck pain with high accuracy by analyzing selected features in shear wave elastography images. Closer inspection of the Table 2 and 3 shows an overlap of the confidence intervals of training and testing results, with tendency of slightly lower performance in the testing results. However, there are a few exceptions where the mean result is slightly higher in the results of testing step compared to the results from training step. This might be a result of heterogeneity of the medical data, and the variance introduced by the pseudo-random selection process, or overfitting due to the high number of features in the training step. This assumption is supported by the fact that the results from the algorithms that have an embedded feature selection are clearly better than the other algorithms. The possibility to train a classification machine learning algorithm with very good accuracy provides some tentative initial evidence that there is a correlation between subjective sensation of stiffness and tissue properties recorded with shear wave elastography modulus. Our research is still in progress, therefore further work has to be done with focus on feature selection and hyper-parameter optimization in order to attain more generalizable well performing model.

The second objective of this study was to find a correlation between extracted features and physiological tissue properties, with the intention to find reliable biomarkers that can be used to objectify and enhance the future diagnostics. What stands out in the Table 4 is that the most important features chosen from the Random Forest algorithm predominantly are extracted from the lowest horizontal segment which correspond to the deepest neck muscle. This might suggest that the main difference is located in the deep neck muscles, closest to the spine. Contrary to our expectations, the activities that participants performed do not have an important role in the classification. Due to complexity of the features extracted from the images, it is not easy to find a direct link between these features and physiological neck muscle tissue properties. Further investigations will be necessary to gain a better understanding of a possible correlation between information in shear wave elastography images and biological properties, and to better understand the mechanisms underlying neck muscle pain.

5 Conclusion

We successfully trained machine learning algorithms that can classify the shear wave elastography images at the level of an expert. Neck pain is a complex condition/experience and until now the diagnostics are highly subjective, dependent on the participants personal sensation of pain, and the diagnostic experience of the diagnostician. The algorithms used in this study could be regarded as first step to objectify the neck pain diagnosis. Moreover, such an automated approach enables a computed image analysis, which may provide new insights into the differences of the physiological properties of the neck muscles in individuals with neck pain compared to asymptomatic individuals. Further research is required to better understand the discrepancies between both groups and ascertain the possibility of adopting the shear wave elastography as a robust and reliable diagnostic tool for neck pain diagnosis.

Conflict of Interest

The authors declare that there are no conflicts of interest.

Authors' Contribution

F. P. wrote the manuscript and carried out the analysis. A.D. provided the sample datasets and supervised the project. M.K. contributed to the interpretation of the results and supervised the project.

Acknowledgements

The work was supported by an internal grant from Hochschule Furtwangen.

References

1. S. Haldeman, L. J. Carroll, and J. D. Cassidy, "The empowerment of people with neck pain: introduction. The Bone and Joint Decade 2000-2010 Task Force on Neck Pain and Its Associated Disorders," *J. Manipulative Physiol. Ther.*, vol. 32, no. 2 Suppl, Art. no. 2 Suppl, Feb. 2009, doi: 10.1016/j.jmpt.2008.11.006.
2. S. Hogg-Johnson *et al.*, "The burden and determinants of neck pain in the general population: results of the Bone and Joint Decade 2000-2010 Task Force on Neck Pain and Its Associated Disorders," *Spine*, vol. 33, no. 4 Suppl, Art. no. 4 Suppl, Feb. 2008, doi: 10.1097/BRS.0b013e31816454c8.
3. P. R. Blanpied *et al.*, "Neck Pain: Revision 2017," *J. Orthop. Sports Phys. Ther.*, vol. 47, no. 7, Art. no. 7, Jul. 2017, doi: 10.2519/jospt.2017.0302.

4. J. C. MacDermid, D. M. Walton, P. Bobos, M. Lomotan, and L. Carlesso, “A Qualitative Description of Chronic Neck Pain has Implications for Outcome Assessment and Classification,” *Open Orthop. J.*, vol. 10, pp. 746–756, 2016, doi: 10.2174/1874325001610010746.
5. S. Taş, F. Korkusuz, and Z. Erden, “Neck Muscle Stiffness in Participants With and Without Chronic Neck Pain: A Shear-Wave Elastography Study,” *J. Manipulative Physiol. Ther.*, vol. 41, no. 7, Art. no. 7, Sep. 2018, doi: 10.1016/j.jmpt.2018.01.007.
6. J. Hvedstrup, L. T. Kolding, M. Ashina, and H. W. Schytz, “Increased neck muscle stiffness in migraine patients with ictal neck pain: A shear wave elastography study,” *Cephalalgia*, vol. 40, no. 6, Art. no. 6, May 2020, doi: 10.1177/0333102420919998.
7. A. V. Dieterich, U. Ş. Yavuz, F. Petzke, A. Nordez, and D. Falla, “Neck Muscle Stiffness Measured With Shear Wave Elastography in Women With Chronic Nonspecific Neck Pain,” *J. Orthop. Sports Phys. Ther.*, vol. 50, no. 4, Art. no. 4, Apr. 2020, doi: 10.2519/jospt.2020.8821.
8. H. Ishikawa *et al.*, “Changes in stiffness of the dorsal scapular muscles before and after computer work: a comparison between individuals with and without neck and shoulder complaints,” *Eur. J. Appl. Physiol.*, vol. 117, no. 1, Art. no. 1, Jan. 2017, doi: 10.1007/s00421-016-3510-z.
9. R. Akagi and S. Kusama, “Comparison Between Neck and Shoulder Stiffness Determined by Shear Wave Ultrasound Elastography and a Muscle Hardness Meter,” *Ultrasound Med. Biol.*, vol. 41, no. 8, Art. no. 8, Aug. 2015, doi: 10.1016/j.ultrasmedbio.2015.04.001.
10. A. V. Dieterich *et al.*, “Shear wave elastography reveals different degrees of passive and active stiffness of the neck extensor muscles,” *Eur. J. Appl. Physiol.*, vol. 117, no. 1, Art. no. 1, Jan. 2017, doi: 10.1007/s00421-016-3509-5.
11. Ž. Kozinc and N. Šarabon, “Shear-wave elastography for assessment of trapezius muscle stiffness: Reliability and association with low-level muscle activity,” *PloS One*, vol. 15, no. 6, Art. no. 6, 2020, doi: 10.1371/journal.pone.0234359.
12. I. Gatos *et al.*, “A Machine-Learning Algorithm Toward Color Analysis for Chronic Liver Disease Classification, Employing Ultrasound Shear Wave Elastography,” *Ultrasound Med. Biol.*, vol. 43, no. 9, pp. 1797–1810, Sep. 2017, doi: 10.1016/j.ultrasmedbio.2017.05.002.
13. Y.-J. Mao, H.-J. Lim, M. Ni, W.-H. Yan, D. W.-C. Wong, and J. C.-W. Cheung, “Breast Tumour Classification Using Ultrasound Elastography with Machine Learning: A Systematic Scoping Review,” *Cancers*, vol. 14, no. 2, Art. no. 2, Jan. 2022, doi: 10.3390/cancers14020367.
14. B. Zhang *et al.*, “Machine Learning–Assisted System for Thyroid Nodule Diagnosis,” *Thyroid*, vol. 29, no. 6, pp. 858–867, Jun. 2019, doi: 10.1089/thy.2018.0380.
15. R. R. Wildeboer *et al.*, “Automated multiparametric localization of prostate cancer based on B-mode, shear-wave elastography, and contrast-enhanced ultrasound radiomics,” *Eur. Radiol.*, vol. 30, no. 2, pp. 806–815, Feb. 2020, doi: 10.1007/s00330-019-06436-w.
16. K. Orhan, G. Yazici, M. E. Kolsuz, N. Kafa, I. S. Bayrakdar, and Ö. Çelik, “An Artificial Intelligence Hypothetical Approach for Masseter Muscle Segmentation on Ultrasonography in Patients With Bruxism,” *J. Adv. Oral Res.*, vol. 12, no. 2, pp. 206–213, Nov. 2021, doi: 10.1177/23202068211005611.
17. K. A. Weber *et al.*, “Multi-muscle deep learning segmentation to automate the quantification of muscle fat infiltration in cervical spine conditions,” *Sci. Rep.*, vol. 11, no. 1, p. 16567, Aug. 2021, doi: 10.1038/s41598-021-95972-x.
18. I. Loram *et al.*, “Objective Analysis of Neck Muscle Boundaries for Cervical Dystonia Using Ultrasound Imaging and Deep Learning,” *IEEE J. Biomed. Health Inform.*, vol. 24, no. 4, pp. 1016–1027, Apr. 2020, doi: 10.1109/JBHI.2020.2964098.
19. G. Van Rossum and F. L. Drake, *Python 3 Reference Manual*. Scotts Valley, CA: CreateSpace, 2009.
20. C. Palm and T. M. Lehmann, “Classification of color textures by gabor filtering,” vol. 11, no. 2, p. 26, 2002.
21. F. Bianconi and A. Fernández, “Evaluation of the effects of Gabor filter parameters on texture classification,” *Pattern Recognit.*, vol. 40, no. 12, pp. 3325–3335, Dec. 2007, doi: 10.1016/j.patcog.2007.04.023.
22. M. Idrissa and M. Acheroy, “Texture classification using Gabor filters,” *Pattern Recognit. Lett.*, vol. 23, no. 9, pp. 1095–1102, Jul. 2002, doi: 10.1016/S0167-8655(02)00056-9.
23. F. Pedregosa *et al.*, “Scikit-learn: Machine Learning in Python,” *J. Mach. Learn. Res.*, vol. 12, no. 85, Art. no. 85, 2011.
24. J. Aljinović *et al.*, “Can measuring passive neck muscle stiffness in whiplash injury patients help detect false whiplash claims?,” *Wien. Klin. Wochenschr.*, vol. 132, no. 17–18, Art. no. 17–18, Sep. 2020, doi: 10.1007/s00508-020-01631-y.
25. Y. Xie, L. Thomas, F. Hug, V. Johnston, and B. K. Coombes, “Quantifying cervical and axioscapular muscle stiffness using shear wave elastography,” *J. Electromyogr. Kinesiol. Off. J. Int. Soc. Electrophysiol. Kinesiol.*, vol. 48, pp. 94–102, Oct. 2019, doi: 10.1016/j.jelekin.2019.06.009.

Spatial-temporal Modelling for Surgical Tool Classification in Cholecystectomy Videos

Tamer Abdulbaki Alshirbaji[1], Nour Aldeen Jalal[1], Thomas Neumuth[2] and Knut Moeller[3]

[1]Institute of Technical Medicine (ITeM), Furtwangen University, and Innovation Centre Computer Assisted Surgery (ICCAS), University of Leipzig
{Nour.A.Jalal,abd}@hs-furtwangen.de
[2] Innovation Centre Computer Assisted Surgery (ICCAS). University of Leipzig
thomas.neumuth@uni-leipzig.de
[3]Institute of Technical Medicine (ITeM), Furtwangen University
Knut.Moeller@hs-furtwangen.de

Abstract. Surgical tool classification is an essential component to analyse the surgical workflow of laparoscopic intervention. It has many potential applications, for instance, developing decision-support systems, automatic indexing of laparoscopic videos, and assessing surgical skills. In this work, a framework for surgical tool presence detection is presented. The proposed approach consists of a CNN model and two LSTM units to model spatial and temporal information encoded in the laparoscopic video. The proposed approach achieved a mean average precision of 94.57%. Experimental results show the value of temporal modelling in improving the classification performance of surgical tools.

Keywords: Surgical tool classification; Laparoscopic video; CNN; LSTM.

1 Introduction

Current operating rooms are equipped with advanced surgical devices and instrumentations. Those devices enable performing the surgical intervention and provide the surgical team with the necessary information. However, it is challenging for the surgeon to process all available data from different surgical devices and keep focusing on surgical actions. Hence, active research has been conducted to analysis surgical workflow, in particular for laparoscopic interventions.

Laparoscopic interventions provide a wealth of data as this type of procedures are performed with special instrumentations like laparoscopic camera which enable monitoring the procedure [1]. Thus, the main focus of the conducted research has been on analyzing laparoscopic videos for several purposes. Identifying surgical tools in laparoscopic images is essential for recognizing surgical actions and phases. That kind of knowledge can serve other intelligent systems with a variety of potential applications. For instance, notifying the surgeon with possible complications, provide assistive guidance, automatic indexing of videos for training purpose and predicting required time to optimize schedule of operating room [1–3].

The revolution of computing power and availability of surgical data have empower applying deep learning approaches. Twinanda et al. released Cholec80 dataset and proposed a convolutional neural network (CNN) architecture called EndoNet to perform surgical phase and tool recognition [4]. However, the prediction of EndoNet model was based only on a single image. Nevertheless, many disturbances can occur in laparoscopic images, for example, emergence of smoke [5, 6], bleeding or light reflections. Hence, anatomical structures and surgical tools might be covered due to such disturbances, and thus impeding capability of image-based approaches. To alleviate the challenging nature of laparoscopic images, modelling temporal information along the video were addressed using different techniques. Hidden Markov model (HMM) [4, 7], graph convolutional network [3], nonlinear autoregressive network with exogenous inputs (NARX) [2], and long short-term memory (LSTM) [8–12] were used to detect surgical tools and/or phases.

In this work, a deep learning approach for classification surgical tools in laparoscopic images was proposed. A CNN model was used to encode spatial features from laparoscopic images. Two LSTM units utilised CNN features of labelled and some unlabelled frames to model temporal dependencies along short and complete sequence of procedure video.

2 Methods

2.1 Model overview

A pipeline consisting of a CNN and two LSTM units was implemented. **Fig. 1** depicts an overview of the methodology pipeline. The CNN was employed to capture visual features of a laparoscopic frame. To obtain high-level discriminative features, the CNN model was initially trained on cholecystectomy images to perform surgical tool detection and surgical phase recognition. To this ends, the architecture of the CNN was modified to perform those tasks similar to the EndoNet architecture [4]. Every single labelled frame and a number of surrounding unlabelled frames formed a short sequential data. Using the trained CNN model, visual features were extracted for every frame in the short sequences. The first LSTM unit exploited the CNN-features of every short data sequence. Thus, temporal dependencies across the short sequence were utilised. To enhance the performance further, a second LSTM unit was employed to consider temporal information along the entire video.

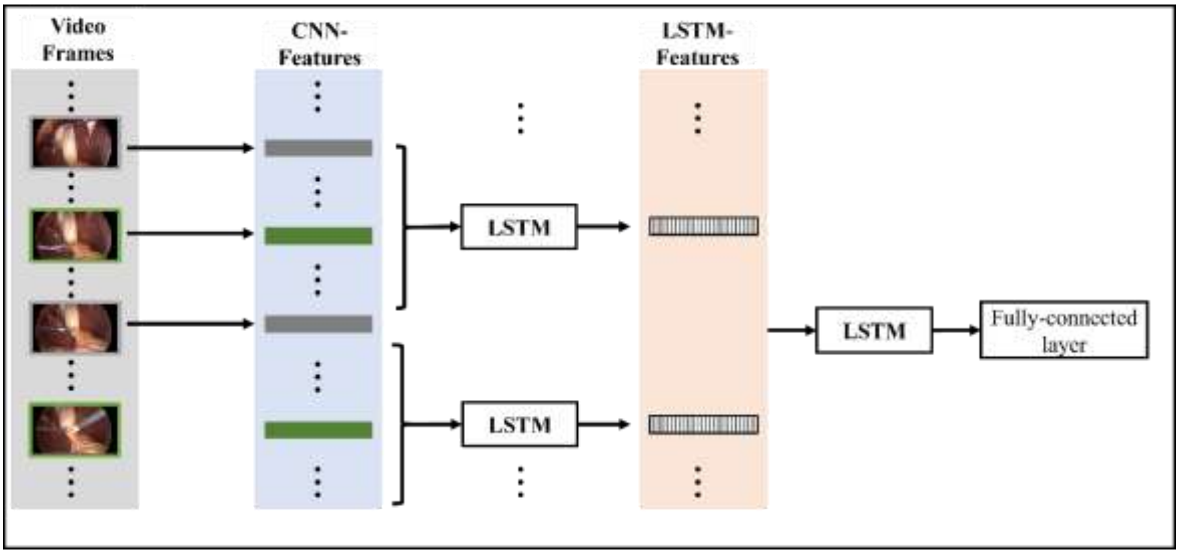


Fig. 1. Pipeline architecture of the proposed approach. Labelled and unlabelled frames are in green and grey rectangles, respectively.

2.2 Dataset description

Eighty cholecystectomy videos of Cholec80 dataset [4] were used in this work. The videos were recorded at University Hospital of Strasbourg at 25 Hz. The dataset contains labels for surgical phases at 25 Hz and labels for surgical tools at 1 Hz. The surgical tools defined in Cholec80 dataset are grasper, bipolar, hook, scissors, clipper, irrigator, specimen bag. Forty videos were used for training, and the remaining videos were used for performance evaluation. The distribution of training and testing data are shown in **Fig. 2**.

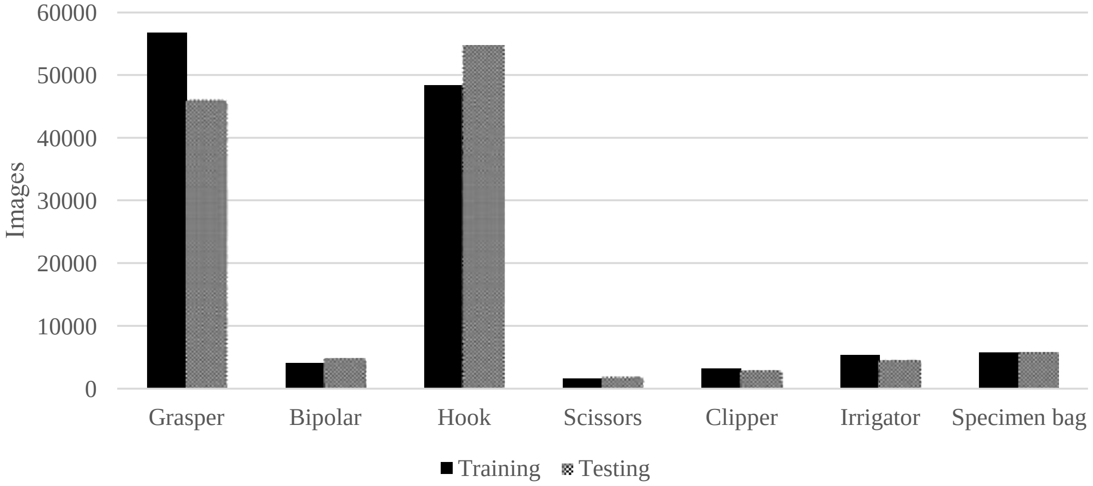


Fig. 2. Distribution of surgical tools in the training and testing data.

2.3 Experimental setup

The base model ResNet-50, pretrained on ImageNet dataset, was employed. The model was trained for ten epochs with an initial learning rate of 2.10^{-3}. The batch size was set to 50 images. The first LSTM contained 512 cells, whereas the second LSTM had 4096 cells. Each of the LSTMs was trained for 30 epochs with an initial learning rate of 1.10^{-4}. The first LSTM had a batch size of 50 short sequences. The length of every short sequence was set to 21 frames, where a labelled frame was in the middle of the sequence. The second LSTM was trained with a batch size of one video.

A fully-connected layer with seven nodes was used in each model to perform tool classification. This layer had a sigmoid activation function since the task was a binary classification. The cross-entropy function was used to compute loss of each tool. Tool losses were weighted according to the number of images belonging to each tool in the training data, as in [13]. The proposed approach was implemented using Keras framework. The implementation was conducted on a PC with NVIDIA GeForce RTX 2080Ti GPU.

3 Results

Each component of the proposed approach was evaluated on the same testing set. **Fig. 3** presents the average precision (AP) of each surgical tool and the mean AP over all tool for CNN and LSTMs.

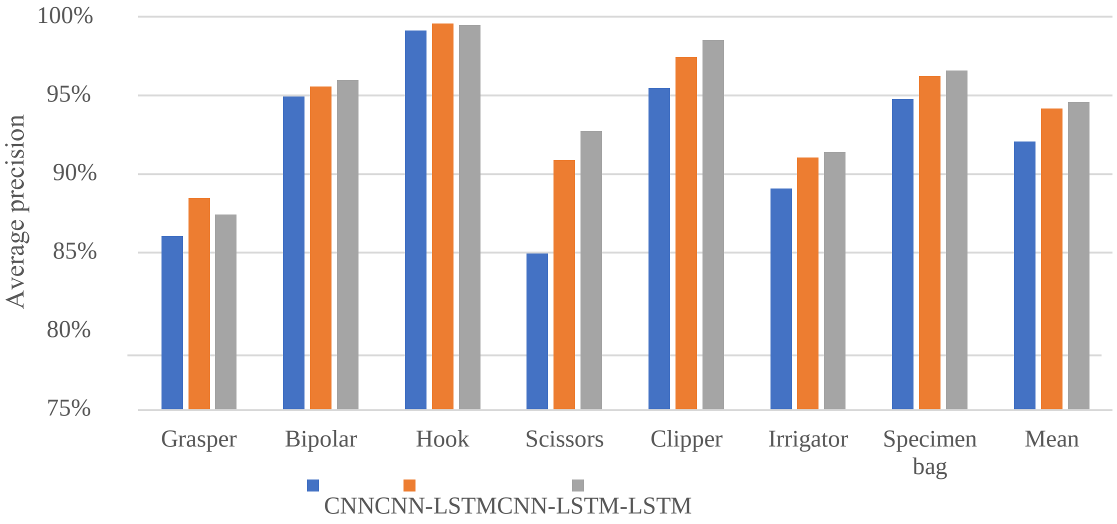


Fig. 3. Average precicion of tool classification yielded from each component of the proposed pipeline.

4 Discussion

This work presents a framework for classifying surgical tools in laparoscopic videos. The framework is based on utilisation of spatial and temporal information encoded in videos. To this end, ResNet-50 and two LSTM units were employed.

The CNN model had a high capability to identify surgical tools in a laparoscopic image. However, the CNN model failed to detect surgical tools when they were partially appeared in the scene or were covered by smoke, blood or a tissue. Therefore, using some unlabelled frames before and after the target frame (labelled frame) helps to recognise surgical tools. Hence, applying the first LSTM unit improved the classification results for all surgical tools (see **Fig. 3**).

The laparoscopic procedure can be segmented into some surgical phases which are executed in some specific order. Since particular surgical tools are used in each surgical phase, there is a correlation between surgical phases and tool usage. Thus, modelling sequential dependencies along entire video, conducted by the second LSTM unit, enhanced the average precision for all tools, except for grasper (see **Fig. 3**). Grasper is used frequently during the entire procedure and appear in all surgical phases, and hence, applying the second LSTM had a marginal effect on enhancing performance for this surgical tool.

The proposed approach achieved a mean average precision of 94.57% higher performance than state-of-the-art methods. Twinanda et al. reported a mean average precision of 81% with EndoNet model [4]. Jin et al. used a loss function which models correlation between surgical phases and tools, and an average precision of 89% for tool presence detection was reported [14]. Similar to our approach, Wang et al. proposed a CNN and graph convolutional networks (GCN) to model spatial and temporal information, respectively, across short video clips. The CNN-GCN approach achieved a mean average precision of 90.13% [3].

The CNN and LSTM models were trained separately. It would be interesting to train the complete framework and investigate the effectiveness of end-to-end training. Moreover, the robustness of the proposed approach to data from different surgical sources could be evaluated.

5 Conclusion

Experimental results demonstrate the value of exploiting temporal information for surgical tool classification. Moreover, this study highlights feasibility of using unlabelled data to improve the classification performance.

References

1. Anteby, R., Horesh, N., Soffer, S., Zager, Y., Barash, Y., Amiel, I., Rosin, D., Gutman, M., Klang, E.: Deep learning visual analysis in laparoscopic surgery: a systematic review and diagnostic test accuracy meta-analysis. Surgical Endoscopy. 35, 1521–1533 (2021). https://doi.org/10.1007/s00464-020-08168-1.
2. Jalal, N.A., Alshirbaji, T.A., Möller, K.: Predicting surgical phases using CNN-NARX neural network. Current Directions in Biomedical Engineering. 5, 405–407 (2019). https://doi.org/10.1515/cdbme-2019-0102.
3. Wang, S., Xu, Z., Yan, C., Huang, J.: Graph convolutional nets for tool presence detection in surgical videos. In: International Conference on Information Processing in Medical Imaging. pp. 467–478. Springer (2019). https://doi.org/10.1007/978-3-030-20351-1_36.
4. Twinanda, A.P., Shehata, S., Mutter, D., Marescaux, J., De Mathelin, M., Padoy, N.: Endonet: a deep architecture for recognition tasks on laparoscopic videos. IEEE transactions on medical imaging. 36, 86–97 (2016). https://doi.org/10.1109/TMI.2016.2593957.
5. Abdulbaki Alshirbaji, T., Jalal, N.A., Mündermann, L., Möller, K.: Classifying smoke in laparoscopic videos using SVM. https://doi.org/10.1515/cdbme-2017-0040.
6. Jalal, N.A., Alshirbaji, T.A., Mündermann, L., Möller, K.: Features for detecting smoke in laparoscopic videos. Current Directions in Biomedical Engineering. 3, 521–524 (2017). https://doi.org/10.1515/cdbme2017-0110.
7. Jalal, N.A., Alshirbaji, T.A., Möller, K.: Evaluating convolutional neural network and hidden markov model for recognising surgical phases in sigmoid resection. Current Directions in Biomedical Engineering. 4, 415–418 (2018). https://doi.org/10.1515/cdbme-2018-0099.
8. Yengera, G., Mutter, D., Marescaux, J., Padoy, N.: Less is more: Surgical phase recognition with less annotations through self-supervised pre-training of CNN-LSTM networks. arXiv preprint arXiv:1805.08569. (2018).
9. Jalal, N.A., Abdulbaki Alshirbaji, T., Docherty, P.D., Neumuth, T., Möller, K.: Surgical Tool Detection in Laparoscopic Videos by Modeling Temporal Dependencies Between Adjacent Frames. In: European Medical and Biological Engineering Conference. pp. 1045–1052. Springer (2020).
10. Alshirbaji, T.A., Jalal, N.A., Docherty, P.D., Neumuth, T., Möller, K.: A deep learning spatial-temporal framework for detecting surgical tools in laparoscopic videos. Biomedical Signal Processing and Control. 68, 102801 (2021).
11. Alshirbaji, T.A., Jalal, N.A., Möller, K.: A convolutional neural network with a two-stage LSTM model for tool presence detection in laparoscopic videos. Current Directions in Biomedical Engineering. 6, (2020).
12. Jalal, N.A., Alshirbaji, T.A., Docherty, P.D., Neumuth, T., Moeller, K.: A Deep Learning Framework for Recognising Surgical Phases in Laparoscopic Videos. IFAC-PapersOnLine. 54, 334–339 (2021). https://doi.org/10.1016/j.ifacol.2021.10.278.
13. Alshirbaji, T.A., Jalal, N.A., Möller, K.: Surgical tool classification in laparoscopic videos using convolutional neural network. Current Directions in Biomedical Engineering. 4, 407–410 (2018). https://doi.org/10.1515/cdbme-2018-0097.
14. Jin, Y., Li, H., Dou, Q., Chen, H., Qin, J., Fu, C.-W., Heng, P.-A.: Multi-task recurrent convolutional network with correlation loss for surgical video analysis. Medical image analysis. 59, 101572 (2020). https://doi.org/10.1016/j.media.2019.101572.