Ning Ding[*], Knut Möller

# Generating adversarial images to monitor the training state of a CNN model

**Abstract:** Deep neural networks have shown effectiveness in many applications, however, in regulated applications like automotive or medicine, quality guarantees are required. Thus, it is important to understand the robustness of the solutions to perturbations in the input space. In order to identify the vulnerability of a trained classification model and evaluate the effect of different perturbations in the input on the output class, two different methods to generate adversarial examples were implemented. The adversarial images created were developed into a robustness index to monitor the training state and safety of a convolutional neural network model. In the future work, some generated adversarial images will be included into the training phase to improve the model robustness.

**Keywords:** deep learning, convolutional neural network, adversarial attack.

https://doi.org/10.1515/cdbme-2021-2077

## 1 Introduction

Deep neural networks have broad application in image classification tasks and natural language processing, However, the deep learning algorithms have been discovered vulnerable to the adversaries, even though some perturbing on benign samples are imperceptible to human vision. This security threat turns to be more significant when the application launched for medical tasks, automotive products or other highly regulated domains [1,2]. Therefore, for the purpose of safety requirement, a technique to evaluate the training state and robustness becomes crucial before deployment of deep learning models. It is necessary to prove that all functional and safety requirements are met by the product. Adversarial attack is a machine learning technique that attempts to fool the model to wrong classification, and is effective to identify the vulnerability of the trained deep neural network model and thus, assesses the robustness of the learned solution.

Many approaches for generating adversarial examples have been proposed recently, such as the fast gradient sign method (FGSM) [3], the iterative fast gradient sign method (I-FGSM) [4], the momentum iterative fast gradient sign method (MI-FGSM) [5], the saliency map approach [6], DeepFool [7], and generative adversarial networks (GANs) [8]. Among these approaches, a few can be used to generate target-class adversarial examples. For instance, the target-FGSM method can decrease the adversarial loss between predicted class and target class [5], or use forward derivative to construct adversarial saliency maps to map input perturbations to output variations [6].

Surgical tool recognition is one of the medical application of convolutional neural network model (CNN) in recent years, as the neural network can automatically learn visual features and aid in developing the context aware system in modern operating room [9]. In this paper, a trained CNN model performed on surgical tool classification task was evaluated by adversarial attack technique. In order to measure the robustness and training state of the model, we focus on how much perturbations on the input space can lead the model to wrong classification, hence linear interpolation and target fast gradient sign method were used for generating adversarial samples with a specific target class.

## 2 Method

The convolutional neural network model is a fine-tuned AlexNet [10] model that was trained by laparoscopic video images to perform a surgical tool classification task. The training and testing images are both extracted from the Cholec80 database [11]. The Cholec80 dataset is a large dataset contains 80 cholecystectomy surgery videos, we extract 80,190 1-class images from these videos as a derived dataset, 25,000 images were used to train the model, and the rest for testing. The model has 7 classes in total, 10 correctly classified images of each class were selected from the testing images and processed to adversarial attack to gain a

**[*]Corresponding author: Ning Ding:** ITeM, Hochschule Furtwangen University, Villingen-schwenningen, Germany, e-mail: n.ding@hs-furtwangen.de.
**Knut Möller:** ITeM, Villingen-schwenningen, Germany.

preliminary evaluation. Initially, the training state of the CNN model can be defined with training accuracy, the training progress was stopped when the training accuracy reached to 99% to avoid overfitting. In order to generate target adversarial images and monitor the training state of this model, as well as to map the input perturbation onto output variation, we chose the linear interpolation to morph images of different classes, and the fast gradient sign method to generate adversarial images with minimal distortions and not noticeable for human observers.

## 2.1 Linear interpolation

Two images are selected randomly as inputs, one from class A and another from class B. The adversarial images were generated using linear interpolation between these two inputs. Since we have 10 legitimate images for 7 classes, there will be $10(class\ A) \times 60(class\ B) \times 7 = 4200$ cases for further evaluation.

- The linear interpolation of two inputs is defined as:

$$x^* = (1-T)x_1 + Tx_2\ ,\ T \in [0{,}1] \quad (1)$$

Where $x^*$ is the generated adversarial image, $x_1$ and $x_2$ are the two selected images.

- T was assigned with 100 values that were evenly spaced between 0 and 1. The generated image was used to evaluate the CNN model classification performance.
- As T gradually increased from 0 to 1 with the step 0.01, the classification of $x^*$ will shift from class A to class B.

## 2.2 Fast gradient sign method

We select one image $x$ correctly classified as class A. A gradient based search in the input space was implemented minimizing the cross entropy loss between the input (belonging to class A) and an incorrect class (e.g. class B) by iteratively modifying the input in the opposite direction of the gradient. These iterations are performed till the generated image change into incorrect class (B). We have $70 \times 6(class\ B) = 420$ cases using this method.

- Select one image $x$ correctly classified as class A. We calculate the cross entropy loss between the score and the target adversarial class, the sign of the gradients from last iteration to update the image for faster calculation:

$$x_0^* = x,$$
$$x_n^* = Clip_{x,\epsilon}\{x_{n-1}^* - \alpha\ \mathrm{sign}(\nabla_x J(x_{n-1}^*, y_{target}))\} \quad (2)$$

Where $x_n^*$ is the generated adversarial image, $x_{n-1}^*$ is the generated adversarial image from the last iteration. $y_{target}$ is the target class (or class B). The values of pixels in image $x_n^*$ are clipped to the range [0 255]. α is valued by 1. The iterations performed till the generated image finally classified as the target class.

- The distance between the original image and the generated adversarial image is summed as the mean-absolute error:

$$MAE(x, x^*) = \frac{1}{N}\|x^* - x\|_1 = \frac{1}{N}\sum|x^* - x|$$
$$N = 227 \times 227 \times 3\ \ (number\ of\ pixels\ in\ x, x^*) \quad (3)$$

# 3 Result

In the linear interpolation method, we calculate the mean threshold of the classification change from one class to a target adversarial class, these thresholds represent classification

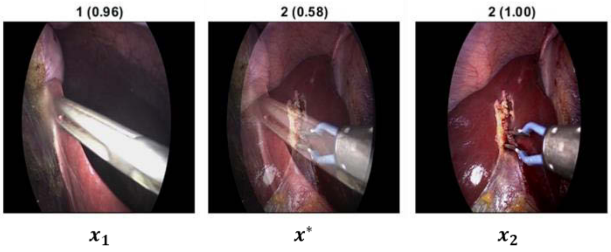

**Figure 1**: The two inputs $x_1$ and $x_2$ , and the generated adversarial image $x^*$, the numbers above the image are the class it belongs to and probability score.

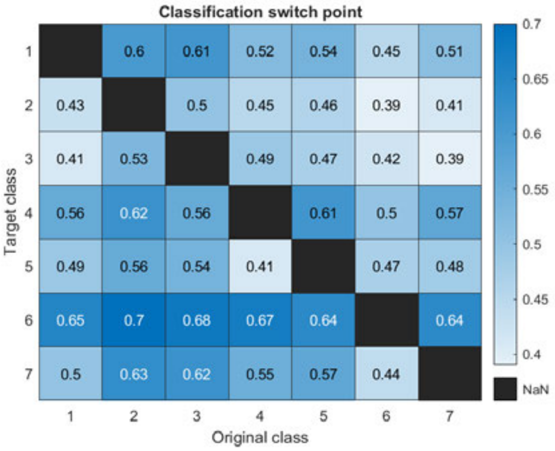

**Figure 2:** The classification switch threshold (t) using linear interpolation method.

priority when the features from different classes both exist in one image.
The lowest class switch point exists during interpolating images from class 7 to class 3, and the highest threshold exists from interpolating the images from class 2 to class 6. Theoretically, the highest and lowest threshold should exist in interpolating the same classes in the inverse direction (exp. the highest threshold should show up during interpolate class 3 to class 7) and the values should be summed up to 1. We will mention the reason to this phenomenon in the discussion part.

In the fast gradient sign method, we try to modify an image to another incorrect class, recording the average absolute pixel difference between the original image and the generated adversarial image. These values and also the iterations indicate the hardness to modify an image from the original class to a target class. We calculate the average pixel distance between the original image and the generated adversarial image, the mean distance of 10 images required was recorded in figure 4. The maximum changes happened in class 3 when the target class is 6. Minimum distance was found in class 6 when the target class is 4.

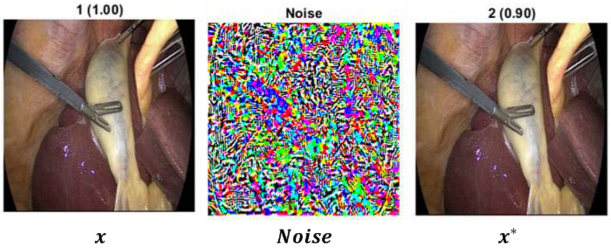

**Figure 3:** The original image $x$, the perturbation subtracted from $x$ using fast gradient sign method and the generated adversarial image $x^*$.

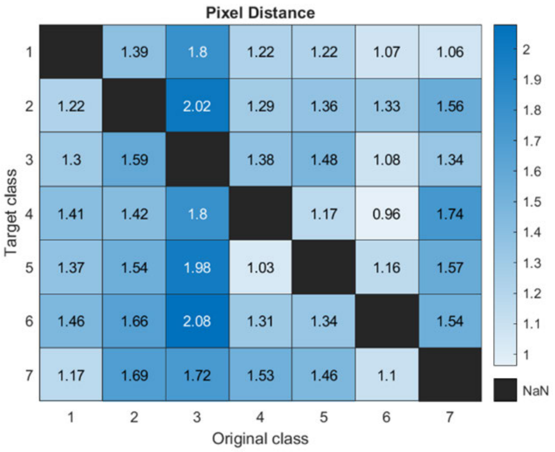

**Figure 4:** The mean absolute error between the original image and the final generated adversarial images of target class.

The hardness of adversarial attack using fast gradient method has discrepancy compared to the result of the interpolation method, which is not surprising, do they make use of different effects.

## 4 Discussion

Both methods can generate adversarial samples with the target incorrect class. However, there are some special cases when applying the interpolation method. While the features from two classes mixed in one image during interpolation, the model classified these adversarial images to a class which is neither the original class nor the target class. The popped classes are unpredictable, somehow indicating that the model is confused by these mixed features. These cases are occur in 1/3 of the investigated 4200 cases.
This phenomenon also influence on the threshold record when the same two images are processed in reverse order. The sum of thresholds should be 1. However, in most cases, the thresholds were influenced by the pop up classes. The reason is t was recorded as the threshold when the target class start appearing, but the target class was delayed by the pop-up class. For example, figure 5 shows the plot of classification probability when interpolating the same two images. There is one input image from class 1 and another from class 2. The recorded t was shifted to the right because of popped classes. The thresholds are summing up to 1.10 instead of 1.

## 5 Conclusion

In this research, we generate adversarial samples by using two different methods. The performance on the generated adversarial images showing the model has different classification priorities, some classes are much easier to be perturbed than others. In the same time, the space around discrete samples were explored and tested by the trained CNN model. The results showing the model classification boundary even though it is discrete and discontinuous. Nevertheless, the hardness of adversarial images generating process are consistent with the robustness of the model classification ability.
In the future work, we will use the same methods to evaluate the fine-tuned model with different training accuracies and compare their robustness or their resistance to different input perturbations. Besides, the fast gradient sign method is suitable for generating images with invisible perturbations,

additional training with these adversarial images can be applied to enhance the robustness of the model.

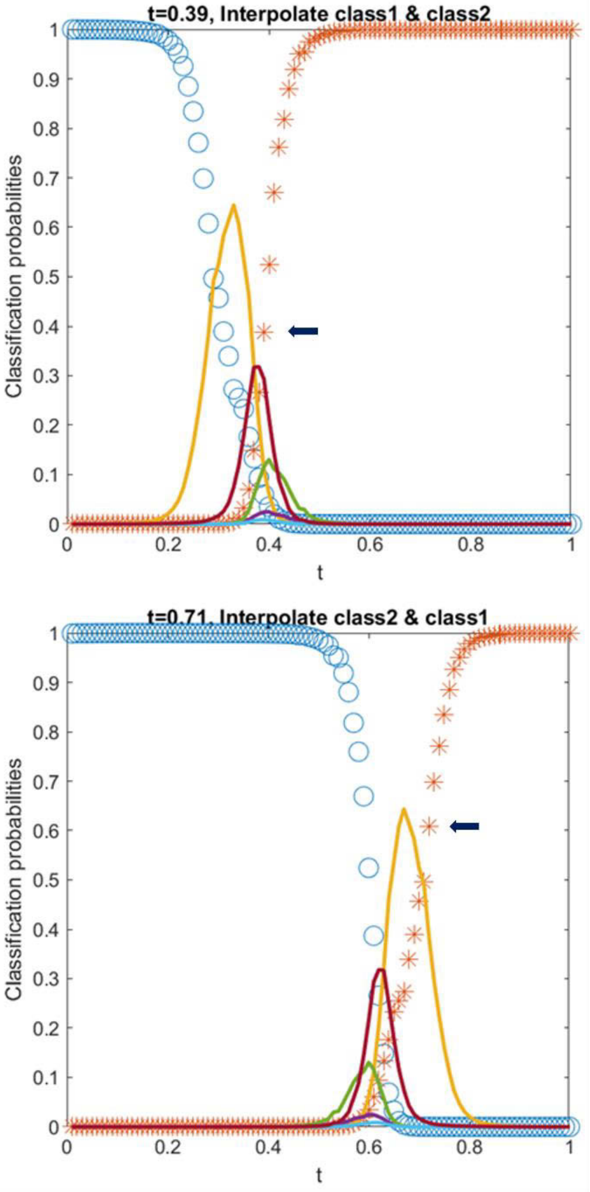

**Figure 5:** The interpolation of same two images from both sides. The circle symbol annotated the probability of original class and the star symbol annotated the probability of target class. The solid line annotated for other classes. The black arrows point to the sample which classified as target class at the threshold t.

### Author Statement

Research funding: This work was supported by the German Federal Ministry of Research and Education (BMBF under grant CoHMed/IntelliMed grant no. 03FH5I01IA). Conflict of interest: Authors state no conflict of interest. Informed consent: Informed consent is not applicable. Ethical approval: The research is not related to either human or animals use.

## References

[1] Ren, Kui, et al. "Adversarial attacks and defenses in deep learning." Engineering 6.3 (2020): 346-360.

[2] Ruan, Wenjie, et al. "Global Robustness Evaluation of Deep Neural Networks with Provable Guarantees for the $ L_0 $ Norm." arXiv preprint arXiv:1804.05805 (2018).

[3] Goodfellow, Ian J., Jonathon Shlens, and Christian Szegedy. "Explaining and harnessing adversarial examples." arXiv preprint arXiv:1412.6572 (2014).

[4] Kurakin, Alexey, Ian Goodfellow, and Samy Bengio. "Adversarial machine learning at scale." arXiv preprint arXiv:1611.01236 (2016).

[5] Dong, Yinpeng, et al. "Boosting adversarial attacks with momentum." Proceedings of the IEEE conference on computer vision and pattern recognition. 2018.

[6] Papernot, Nicolas, et al. "The limitations of deep learning in adversarial settings." 2016 IEEE European symposium on security and privacy (EuroS&P). IEEE, 2016.

[7] Moosavi-Dezfooli SM, Fawzi A, Frossard P. DeepFool: a simple and accurate method to fool deep neural networks. In: Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition; 2016 Jun 27–30; Las Vegas, NV, USA; 2016. p. 2574–82.

[8] Xiao C, Li B, Zhu JY, He W, Liu M, Song D. Generating adversarial examples with adversarial networks. 2018. arXiv:1801.02610.

[9] Alshirbaji, T. A., Ding, N., Jalal, N. A., & Möller, K. (2020). The effect of background pattern on training a deep convolutional neural network for surgical tool detection. Proceedings on Automation in Medical Engineering, 1(1), 024-024.

[10] Krizhevsky, A., Sutskever, I. and Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems (pp. 1097-1105).

[11] Twinanda, A.P., Shehata, S., Mutter, D., Marescaux, J., De Mathelin, M. and Padoy, N., 2016. Endonet: a deep architecture for recognition tasks on laparoscopic videos. IEEE transactions on medical imaging, 36(1), pp.86-97.