Deep Experiments on Deepfake Detection

2022 Spring Semester, Artificial Neural Networks and Deep Learning

Yebin Lee, Boyoung Han and Juhyeon Jung

Department of Data Science, Seoultech, Republic of Korea, {yebin, byhan2253, jjh990307}@ds.seoultech.ac.kr

Keywords: Deepfake, deepfake detection, VGGFace2, ResNet50, DenseNet121, image classification, deep learning, CNN, final report

1. Introduction

1.1. Background and Motivation

Deepfake is a portmanteau of 'deep learning' and 'fake'. Deepfake is a technology that creates a manipulated image by superimposing another image on a video through deep learning (Chawal, 2019). March 2022, a video clip of the Ukrainian president declaring his surrender to Russia was distributed, drawing much attention (Wakefield, 2022). However, the video is fake video utilized Deepfake technology. This technology can cause national, political, and social problems.

1.2. Purpose

Hany Farid, Professor of California State University at Berkeley, pointed out that Ukrainan presi-dent surrender video was fake when he saw that the resolution was lowered to hide the distortion caused by the manipulation process (Metz, 2022).

This project is aiming to investigate the effect of image change on Deepfake detection problem. Deepfake detection performance is compared using original (color) images, images with color changes, and images with changes in saturation.

2. Literature Review

Shad et al. (2021) implemented eight CNN models to detect deepfake images and make a compara-tive analysis. In this research, models are trained by dataset from Kaggle, which had 70,000 images from the Flickr dataset and 70,000 images produced by styleGAN. The eight CNN models utilized are as follows: DenseNet121, DenseNet169, DenseNet201, VGG16, VGG19, VGGFace, Res-Net50, and custom CNN which was proposed by authors. The models are evaluated by five met-rics: accuracy, precision, recall, F1-score, and area under the ROC (receiver operating characteristic) curve. Amongst all the models, VGGFace performed the best, with 99% accuracy. The second highest performance models were ResNet50 and DenseNet121 with 97% accuracy.

3. Data

This project intends to use the Deepfake Detection Challenge(DFDC) dataset from Kaggle (Dolhansky et al. 2019). The DFDC dataset consists of 400 videos. To utilize the CNN model, we cut the video into frames and used it as images. The project was conducted with only 20% of the videos because if we use images of all 400 videos, there are many overlapping images, and the size of the data is too large for the model to train.

3.1. Split video to images

One video was cut into up to 300 frames. In train dataset, the number of fake images is 14,880 and the number of real images is 4,320. In test dataset, the number of fake images is 3,720 and the number of real images is 1,080.

3.2. Make different color type of video

In this project, we define the corresponding image as an Original Image (OI), an image with RGB values adjusted from the original image as an RGB Transformed Image (RTI), and finally an image with HSV values adjusted as an HSV Transformed Image (HTI). The examples of each image are shown in Fig.1 and Fig.2.

Figure 1. Real Image Example

Figure 2. Fake Image Example

Input data combination is like below:

① Original Image (OI): Full color Image, Original Data from Kaggle

② RGB Transformed Image (RTI): Modify RGB ratio to make black & white image

③ HSV Transformed Image (HTI): Modify HSV ratio to make high saturation image

④ OI + RTI

⑤ OI + HTI

⑥ RTI + HTI

⑦ OI + RTI + HTI

4. Methodology

The framework of this project is shown in Fig.3.

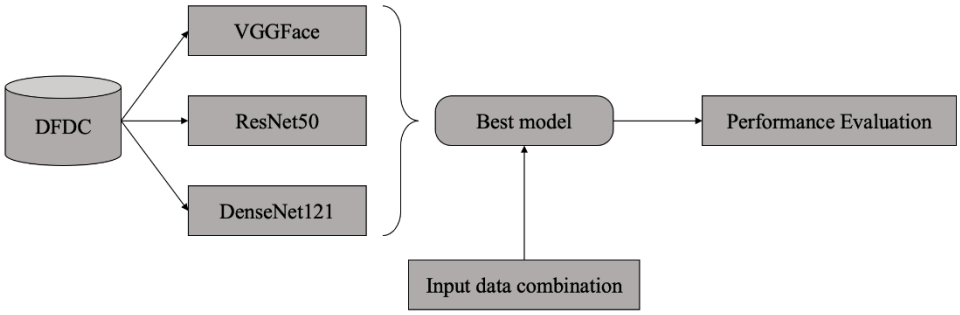

Figure 3. Project Framework

First, we use three pre-trained model: VGGFace2, ResNet50, and DenseNet121, which had good performance in previous study (Shad et al., 2021). The models are applied to the original data (Input data combination 1), and then find the best-performing model.

The purpose of the project is to see how variations in data affect classification performance. Therefore, the classification performance is compared by putting input data combinations 2 to 7 in the best model.

To compare the result, we will use train accuracy, test accuracy, precision, recall, and F1-score from confusion matrix. Through this evaluation, we are going to find out how image change affects Deepfake detection performance.

5. Experimental Results

5.1. Find the best model with original image data

The train accuracy of each model is shown in table Table.1. Train accuracy showed high performance in all three models. The test accuracy was relatively low in ResNet50 and showed the best performance in DenseNet121. Therefore, we conducted additional experiments using DenseNet121 with the converted data.

Table 1. Accuracy of the three models

	VGGFace2	ResNet50	DenseNet121
Train Accuracy	0.9658	0.9863	**0.9995**
Test Accuracy	0.9714	0.7229	**0.9983**

5.2. Find the best data combination

Table 2 Results of one data group

Data	Label	Accuracy	Precision	Recall	F1-score
OI	Fake (0)	0.9983	0.9984	0.9995	0.9989
	Real (1)		0.9981	0.9944	0.9963
RTI	Fake (0)	0.9842	0.9805	0.9995	0.9899
	Real (1)		0.9980	0.9315	0.9636
HTI	Fake (0)	**0.9998**	1.0000	0.9997	0.9999
	Real (1)		0.9991	1.0000	0.9995

Table 3. Results of data combination

Data	Label	Accuracy	Precision	Recall	F1-score
OI + RTI	Fake (0)	0.9995	0.9996	0.9998	0.9997
	Real (1)		0.9994	0.9985	0.9990
OI + HTI	Fake (0)	0.9997	1.0000	0.9996	0.9998
	Real (1)		0.9987	1.0000	0.9994
RTI + HTI	Fake (0)	**0.9998**	1.0000	0.9997	0.9998
	Real (1)		0.9989	1.0000	0.9994
OI + RTI +HTI	Fake (0)	0.9997	1.0000	0.9996	0.9998
	Real (1)		0.9987	1.0000	0.9993

The results of models trained with one data group are shown in Table 2. Among the models trained with one data group, the model with the highest performance is the model trained with HTI data. In model trained with RTI data, performance has been slightly reduced. Therefore, we suggest that training by increasing saturation of image can help Deepfake detection.

For the results of data combinations which is shown in Table 3, overall performance was improved due to the data augmentation effect. Especially, data combinations which contain HTI data have better performance.

Therefore, it was confirmed that increasing the image saturation for deepfake image detection helps to improve model performance. In addition, it was confirmed that data augmentation using color change in model learning improved the performance.

6. Conclusion

There are serious impersonation damage using deepfake technology. In the case of the video of the Ukrainian president's declaration of surrender, it can be a threat to national security. Therefore, it is essential to re-implement deepfake detection technology and study various experiments.

First, in this study, three models, VGGFace2, ResNet50, and DenNet121, with the best performance in previous studies were used. Among the three models, the model with the best performance in the dataset used in this study was DensNet121. Therefore, DesNet121 was selected as the final model for our project.

Next, we tested the effect on the model performance by giving color changes to the data as well as the original data to the model with the best performance. As a result, it was confirmed that the performance was slightly reduced when the black-and-white image was trained compared to the original image, but the performance improved greatly when the training was performed with the saturation transformed image. In addition, as a result of training by combining the original image, black-and-white image, and high saturation image, it was confirmed that the model performance was improved due to the effect of data augmentation. Among them, in the case of including the high saturation image, the detection effect was close to 100%.

Based on this study, it can be seen that increasing the image saturation will be helpful in deepfake detection. As a future study, larger data can be used for experiments, and model performance can be checked using data focused on faces.

Github

Yebin Lee : https://github.com/biiinnn/deepfake_detection

Boyoung Han : https://github.com/bobo-0/deepfake_detection

Juhyeon Jung : https://github.com/JuhyeonJung/deepfake_detection

References

Wakefield, B. J. (2022, 18 maart). Deepfake presidents used in Russia-Ukraine war. BBC News. Geraadpleegd op 1 april 2022,

R. Chawla, "Deepfakes : How a pervert shook the world." international journal of advance research and development, vol 4, issue 6, p.4-8, 2019.

Metz, R. (2022, March 16). Facebook and YouTube say they removed Zelensky deepfake. CNN Business. Retrieved April 1, 2022, from https://edition.cnn.com/2022/03/16/tech/deepfake-zelensky-facebook-meta/index.html.

Shad, H. S., Rizvee, M., Roza, N. T., Hoq, S. M., Monirujjaman Khan, M., Singh, A., ... & Bourouis, S. (2021). Comparative Analysis of Deepfake Image Detection Method Using Convolutional Neural Network. Computational Intelligence and Neuroscience, 2021.

Salpekar, O. DeepFake Image Detection.

Deepfake Detection Challenge. Kaggle. (n.d.). Retrieved April 1, 2022, from https://www.kaggle.com/competitions/deepfake-detection-challenge

Do, N. T., Na, I. S., & Kim, S. H. (2018). Forensics face detection from GANs using convolutional neural network. ISITC, 2018, 376-379.

Serengil, S. (2022, 20 januari). Deep Face Recognition with VGG-Face in Keras | sefiks.com. Sefik Ilkin Serengil. Geraadpleegd op 7 april 2022, van https://sefiks.com/2018/08/06/deep-face-recognition-with-keras

Dolhansky, B., Howes, R., Pflaum, B., Baram, N., & Ferrer, C. C. (2019). The deepfake detection challenge (dfdc) preview dataset. arXiv preprint arXiv:1910.08854.

APPENDIX

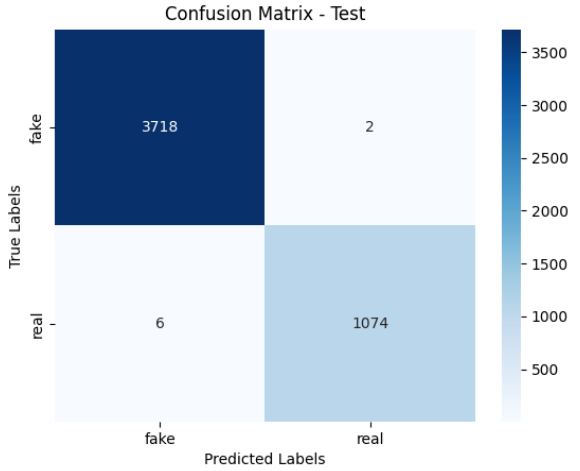

Figure 4. Confusion Matrix - OI

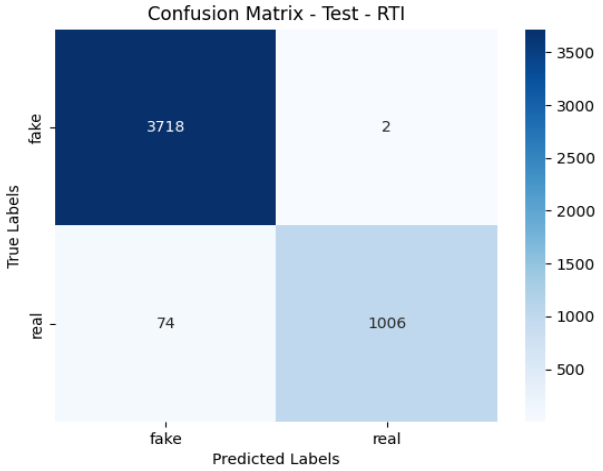

Figure 5. Confusion Matrix - RTI

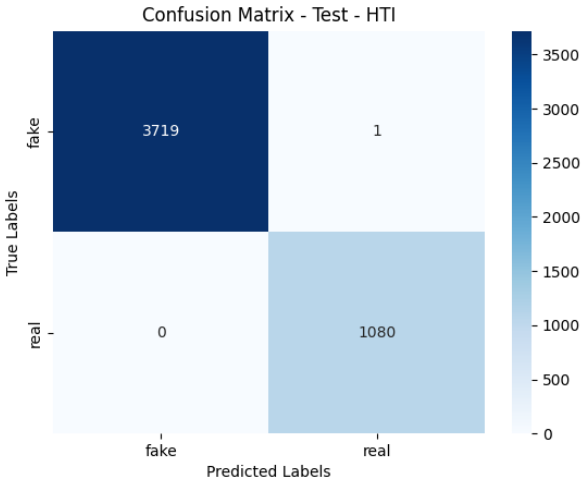

Figure 6. Confusion Matrix - HTI

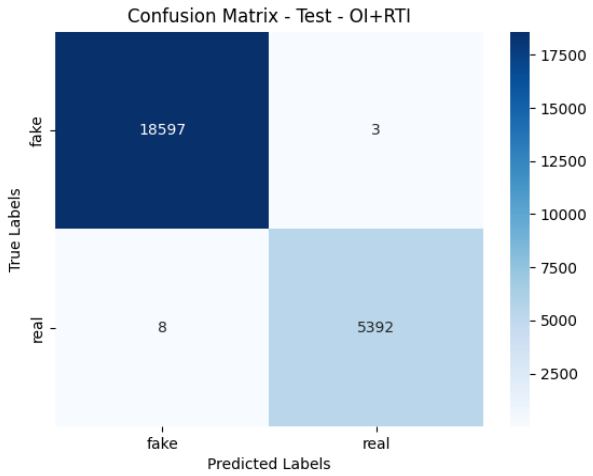

Figure 7. Confusion Matrix – OI + RTI

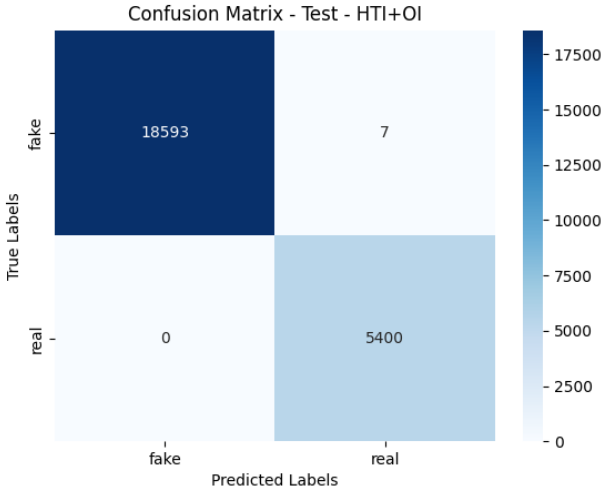

Figure 8. Confusion Matrix – OI + HTI

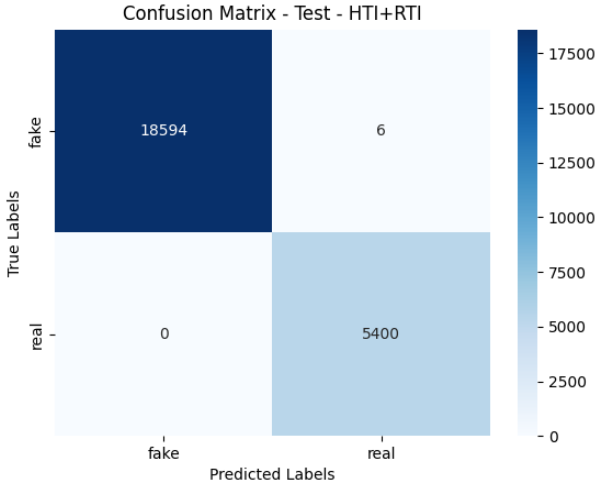

Figure 9. Confusion Matrix – RTI + HTI

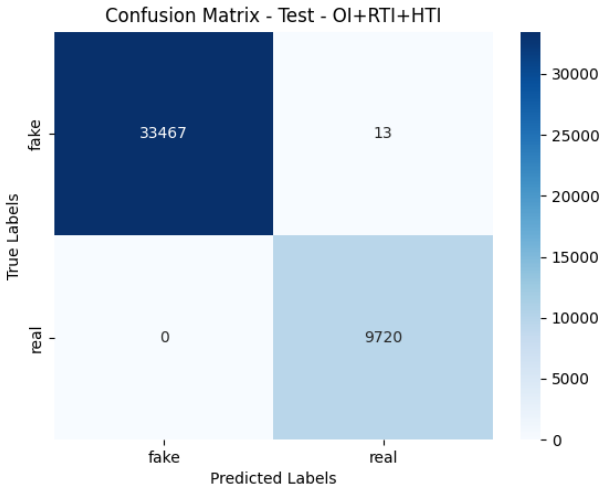

Figure 10. Confusion Matrix – OI + RTI + HTI