Combination of CNN and GAN in Deepfake Images Detection

Juan Christopher Santoso - 13521116
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung
E-mail: juan.csantoso@gmail.com

***Abstract*— Images are used by people to communicate certain information. However, similar to other types of information media, images can also be manipulated. Therefore, the authenticity and validity of the information within an image is often questionable. Moreover, with the rapid development of current technology, manipulating images is no longer a hard task to do. Even these so-called fake images can be so trustful as they seem to look like real images. These fake images are referred to as deepfake images. Nowadays, detecting deepfake images can be a challenge in evaluating the images' validity. With the help of deep learning, the detection process can be done by using a classification model. Common models used to classify deepfake images are CNN and GAN. Both models offer their own strengths and weaknesses when detecting images. Therefore, both models are hypothesized to be able to be combined to support one another in its classification process.**

Keywords—information; deepfake images; detection; deep learning; classification; CNN; GAN;

I. Introduction

Images are one of the media people use to communicate with one another. Not only to communicate, it is not rare for people to use images to express their feelings or spread information. Nowadays, there are uncountable images that are provided on the internet. Moreover, with the unlimited potential of the internet, it can be inferred that "*almost everyone can see almost everything*" on the internet. Although there are indeed access limitations on the internet, it cannot be denied that people can encounter incalculable images on the internet, furthermore, if we include all the images outside the internet.

Images often carry a message. Whether it is crucial or not, people tend to try to understand the information inside the image. There is a proverb that says "*Seeing is believing.*" Therefore, people tend to believe whatever information they see with their own eyes, without doing further investigation. This what makes images can be dangerous sometimes because the information within the image is not always true. Images can be manipulated, one way or another, to make the information within the image is either incomplete or incorrect. With the advancement of current technology, one of the biggest challenges in gaining information from images is detecting deepfake images.

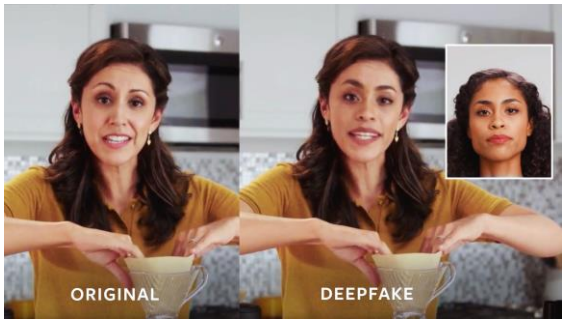

Image I.1 Changes in deepfake images
(Source: World Economic Forum)

Deepfake images are fake images that are generated using machines and algorithms. Deepfake images are usually based on a specific real image but manipulated so they become not-so-original anymore. They are called 'deep' because deepfake images are generated using deep-learning algorithms [1]. Therefore, deepfake images can achieve a high level of similarities to real images.

With the high similarity of the fake and real images, detecting the fake from one becomes a new challenge. Since deepfake images have become a threat to misinformation across the globe [2], people have to provide a way to classify which ones are the fake ones and which are the real ones.

II. Theoretical Basis

The experimentation done in this paper is constructed based on theoretical knowledge. This theoretical knowledge includes the concepts, methods, and understandings of all the relevant topics in this paper.

A. Image

An image is a visual representation of something [3]. Images can take the form of photographs, graphics, or even individual video frames. Images have come into many forms and one of its forms is digital images. A digital image is an image that was created and stored in electronic form.

An image is represented in the 2D plane. Therefore, an image should have a certain size. The size of an image is dependent on its height and width. The longer its height and width, the bigger the size of the image. The smallest unit within

an image is called a pixel. Each pixel represents a value within the image which is the color.

B. Deepfake Image

Deepfake images are images that are generated using deep-learning algorithms. Deepfakes are generated by manipulation of certain real images so the actual information within the actual images is altered. One of the most common applications of deepfakes is swapping people's faces.

Since deepfakes are generated from real images, deepfakes can over a high level of similarity to actual images. Although synthetic images, deepfakes can be deceiving, especially to human eyes. With the rapid progression of technology and knowledge, the differences between generated deepfakes and real images become less and less noticeable. Therefore, generating a deepfake might be an easy task to do. On the other hand, classifying deepfakes and real images will become more complex.

C. Deep Learning

Deep learning is a form of neural network process where the data is processed from layer to layer [4]. The term *'deep'* itself is used to describe the amount of hidden layer used within its algorithm, which is usually described as depth. Similar to neural networks, deep learning consists of three types of layers: input layer, hidden layers, and output layer.

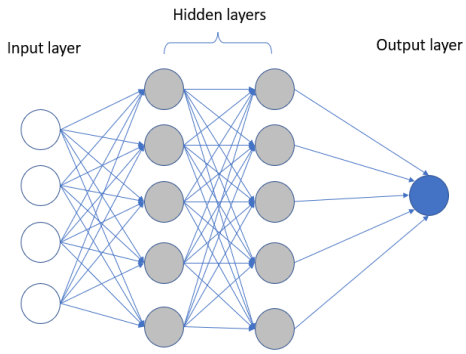

Image II.1 Visualization of deep learning
(Source: Medium)

Deep learning must have a certain depth. Therefore, the hidden layers in deep learning consist of more than one layer. The process of the data from the input layer to the output layer is a sequence of functions, which can be formulated as:

$$y = f^{n}(f^{n-1}(\ldots(f^{2}(f^{1}(x)))\ldots))$$

with n represents the layer-n within the deep learning model.

D. Convolutional Neural Network (CNN)

Convolutional neural network, as how it is named, is a deep learning process that adopts the concept of convolution in processing the data. CNN is usually used to process image, speech, and audio signal inputs [5]. The requirement to use CNN is to have a grid topology input, which is suitable for the topology of images [6]. Based on its components, CNN consists of 3 types of layers:

1. Convolutional Layer

This layer processes the data by doing a convolution operation. The input of this layer is the source image and the output of this layer is called a feature map. Each feature map represents stored information within an image.

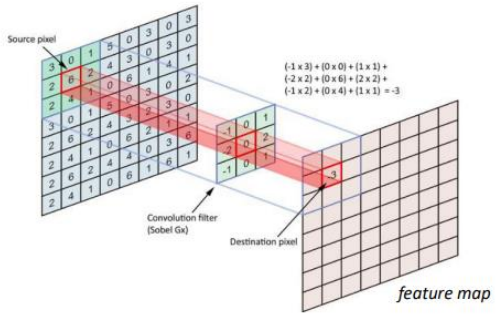

Image II.2 Visualization of the convolution process
(Source: IF4073 Class Materials)

2. Pooling Layer

This layer is responsible for reducing the dimensionality of the image. This process is necessary to minimize the needed computational power when processing the data. The most common pooling types to be used are max pooling and average pooling. Max pooling is an operation to select the maximum value of a local area. On the other hand, average pooling is an operation to get the mean value of a local area.

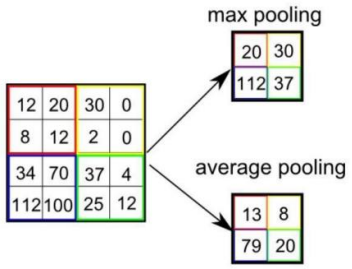

Image II.3 Visualization of the pooling process
(Source: IF4073 Class Materials)

Convolutional and pooling layers are processed iteratively. This is done to extract information from the image and transform it into a desired format. Therefore, the iteration process of these layers is usually called feature extraction. The output of feature extraction is a sequence of features.

3. Fully-Connected Layer

This layer has a similar process as the usual neural networks. After the information is extracted to a

sequence of features, each feature will be connected to a specific node. Each node will be connected to all forward nodes, making it fully connected to one another. The output of this layer is the classification result which is the output of the whole CNN model.

E. Generative Adversarial Network (GAN)

A generative adversarial network (GAN) is a deep learning architecture, consisting of two neural networks that try to compete against each other [7]. It is built with the foundation to make a machine learn by trying to beat the other machine. GAN has been used to generate various synthetic images, including deepfakes.

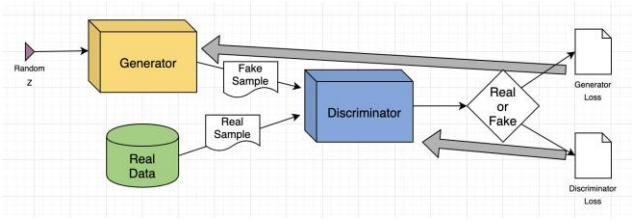

Image II.4 GAN architecture
(Source: Amazon Web Services)

The two competing machines inside the GAN architecture are two neural network models, the generator and the discriminator [8]. These components are named based on their functionalities, to generate images and to classify images.

1. Generator Component

The generator component has the responsibility to generate new images based on real images. The generator takes random noise as input and converts it into complex data samples. The generated images will be called fake images or counterfeit images. These images will be sent to the discriminator component.

In every iteration, the generator component will encounter more and more images, making it more creative and intelligent. The generated image will also be more similar to the real images, even to the point that the difference might not be noticeable by human eyes.

2. Discriminator Component

The discriminator component has the responsibility to differentiate fake images from real images. The discriminator takes turns to learn the real images and the fake images. The real images are received from the dataset while the fake images are received from the generator component.

In every iteration, the discriminator component will encounter more and more images, making it more careful and reliable. The discriminator would detect even the slightest difference between both types of images, so it could decide whether or not the image is a fake.

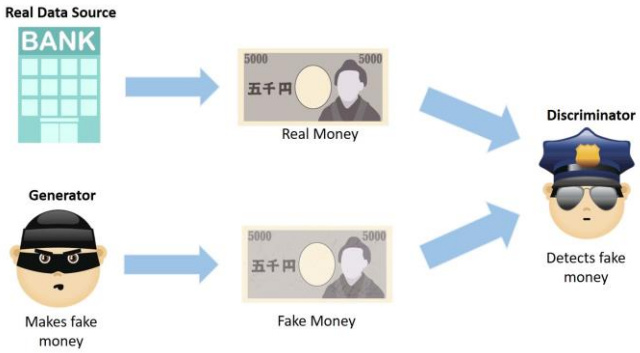

Image II.5 Representation of GAN as a roleplay
(Source: Macnica)

To conclude, in the system roleplay, the generator component has the role of the counterfeiter. The goal of this component is to create counterfeit images so real that it would deceive the discriminator component. On the other hand, the discriminator component has the role of the cops. The goal of this component is to intelligently detect and classify the counterfeit images from the real ones.

III. Application

The experimentation is done using Python programming language. Throughout the entire process of experimentation, the system has been thoroughly planned starting from the planning of system architecture, determining learning parameters, choosing libraries, and constructing models.

A. System Architecture

The core components of the system are the deep learning models, which are the CNN and GAN models. The goal of the system is to combine the prediction capabilities of both models to have a better prediction result. Aside from the models, the system itself consists of several supporting components (see Image III.1):

i. Preprocessing component

This component has the responsibility to preprocess the images in the dataset and transform it into a desirable format. The input of this component is the image dataset. On the other hand, the output of this component is splitted and preprocessed data in the form of: train data, valid data, and test data.

ii. Prediction component

This component has the responsibility of making predictions about the test data using a specific model. The inputs of this component are the test data and a trained model. On the other hand, the output of this component is the prediction result.

iii. Weighting and combining component

This component has the responsibility of combining the prediction result of the CNN model and the GAN model. This component traverses each prediction value within the prediction sequence and

does a weighting operation. The final prediction of the system can be formulated as:

$$Final\_pred = CNN\_pred * CNN\_weight + GAN\_pred * GAN\_weight$$

B. Learning Parameters

Here are the parameters used in the learning process to build both the CNN and GAN models:

- Image size: 128 pixels,
- Batch size: 32 batches,
- Number of epochs:
  - CNN: 20 epochs
  - GAN: 10 epochs
- Latent dimension: 100
- Image Color Channels: 3
- CNN weight: 0.6
- GAN weight: 0.4

C. Libraries

Here is the list of used libraries in the system development:

- pandas, is used to construct dataframes and process data
- numpy, is used to construct data image and represent images
- tensorflow, is used to construct deep learning models

D. Model Summaries

Each model within the system has its own implementation. Different types of models contain different types and amounts of layers (see Image III.2). Each model's summary can be explained as:

- The CNN model consists of 3 pairs of convolutional and pooling layers and 1 flattened layer.
- The GAN generator model consists of 1 dense layer and 4 convolutional transpose layers.
- The GAN discriminator model consists of 3 convolutional layers and 1 flattened layer.

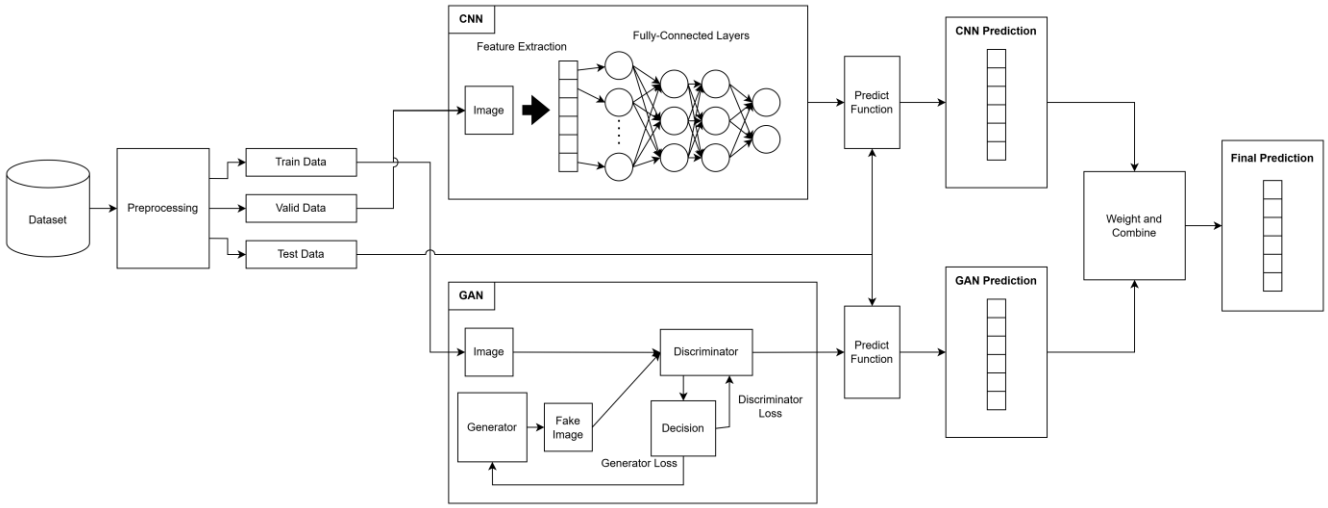

Image III.1 System's architecture diagram
(Source: Author's archive)

Layer (type)	Output Shape	Param #
conv2d_6 (Conv2D)	(None, 126, 126, 32)	896
batch_normalization_3 (BatchNormalization)	(None, 126, 126, 32)	128
max_pooling2d_3 (MaxPooling2D)	(None, 63, 63, 32)	0
dropout_7 (Dropout)	(None, 63, 63, 32)	0
conv2d_7 (Conv2D)	(None, 61, 61, 64)	18,496
batch_normalization_4 (BatchNormalization)	(None, 61, 61, 64)	256
max_pooling2d_4 (MaxPooling2D)	(None, 30, 30, 64)	0
dropout_8 (Dropout)	(None, 30, 30, 64)	0
conv2d_8 (Conv2D)	(None, 28, 28, 128)	73,856
batch_normalization_5 (BatchNormalization)	(None, 28, 28, 128)	512
max_pooling2d_5 (MaxPooling2D)	(None, 14, 14, 128)	0
dropout_9 (Dropout)	(None, 14, 14, 128)	0
flatten_2 (Flatten)	(None, 25088)	0
dense_4 (Dense)	(None, 128)	3,211,392
dropout_10 (Dropout)	(None, 128)	0
dense_5 (Dense)	(None, 1)	129

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 64, 64, 64)	3,136
leaky_re_lu_4 (LeakyReLU)	(None, 64, 64, 64)	0
dropout (Dropout)	(None, 64, 64, 64)	0
conv2d_1 (Conv2D)	(None, 32, 32, 128)	131,200
leaky_re_lu_5 (LeakyReLU)	(None, 32, 32, 128)	0
dropout_1 (Dropout)	(None, 32, 32, 128)	0
conv2d_2 (Conv2D)	(None, 16, 16, 256)	524,544
leaky_re_lu_6 (LeakyReLU)	(None, 16, 16, 256)	0
dropout_2 (Dropout)	(None, 16, 16, 256)	0
flatten (Flatten)	(None, 65536)	0
dense_1 (Dense)	(None, 1)	65,537

Layer (type)	Output Shape	Param #
dense (Dense)	(None, 16384)	1,654,784
leaky_re_lu (LeakyReLU)	(None, 16384)	0
reshape (Reshape)	(None, 8, 8, 256)	0
conv2d_transpose (Conv2DTranspose)	(None, 16, 16, 128)	524,416
leaky_re_lu_1 (LeakyReLU)	(None, 16, 16, 128)	0
conv2d_transpose_1 (Conv2DTranspose)	(None, 32, 32, 64)	131,136
leaky_re_lu_2 (LeakyReLU)	(None, 32, 32, 64)	0
conv2d_transpose_2 (Conv2DTranspose)	(None, 64, 64, 32)	32,800
leaky_re_lu_3 (LeakyReLU)	(None, 64, 64, 32)	0
conv2d_transpose_3 (Conv2DTranspose)	(None, 128, 128, 3)	1,539

Image III.2 Summaries of (a) CNN model (b) GAN generator model (c) GAN discriminator model
(Source: Author's archive)

IV. Implementation

The implementation will be explained in two parts, the workflow of the system and the result of the experimentation. The system workflow part explains the process within the system. On the other hand, the experimental result displays the performance of the developed system.

A. System Workflow

The workflow of the system can be described as a sequence of processes. The sequence starts from the initial state of the system until resulting the classification or detection to deepfake images. Each process within the workflow can be explained below:

1. Firstly, the dataset was preprocessed and transformed to be in a desirable format.
2. Next, the dataset was separated into three types: train data, valid data, and test data.
3. Then, the CNN model was built by determining the type of layers and the learning parameters.
4. The learning process for the CNN model using train data and valid data was conducted.
5. Parallel to steps 3 and 4, the GAN model was built and followed by conducting the learning process for the model using train data.
6. Using the test data, the prediction was calculated using both trained CNN and GAN models.
7. The prediction result of both CNN and GAN was combined and weighted to produce the final prediction of the system.

B. Experimental Result

The used data is 2000 data for train data and 400 data for test. When training the model, the needed duration to train the model is 7 minutes for CNN and 23 minutes for GAN. Three types of assessment were conducted in evaluating the performance of the system. These assessments are:

1. Doing prediction using the CNN model only
2. Doing prediction using the GAN model only
3. Doing prediction using the combination of CNN and GAN

The metrics used to represent the performance of the model are precision, recall, f1 score, and accuracy. The result of all the assessments is displayed in the table below:

Table IV.1 Model performance metrics and its result

Metrics	**CNN**	**GAN**	**CNN+GAN**
Precision – Real Img	0.7	0.53	0.71
Precision – Fake Img	0.59	0.39	0.59
Recall – Real Img	0.74	0.15	0.74
Recall – Fake Img	0.53	0.80	0.55
F1 Score – Real Img	0.72	0.24	0.73
F1 Score – Fake Img	0.56	0.52	0.57
Accuracy	0.66	0.41	0.67

In addition, the performance of the model can also be displayed using a confusion matrix. The result of the confusion matrix for each model is displayed below:

Table IV.2 Confusion matrix result of each model

Metrics	**CNN**	**GAN**	**CNN+GAN**
True Positive	178	37	178
False Positive	61	202	61
False Negative	75	33	73
True Negative	86	128	88

V. Analysis and Evaluation

Based on the experimental result in Table IV.1, it is concluded that the performance of GAN to detect deepfake images is not so good compared to CNN. CNN offers higher accuracy and higher score in almost every metric. Therefore, CNN is still superior to GAN when it comes to detecting deepfake images. This result might happen because of several factors:

1. The difference number of epochs in building CNN and GAN. CNN used 20 epochs while GAN only used half of it. This is the main suspect for the low performance of GAN since the amount of training was not as much as the training in CNN.
2. The fake images generated by GAN have huge differences from the actual test images. This makes the model cannot detect the test images well, the model might look at the test image as a brand new image, not similar to any images it has ever encountered.
3. The GAN model encountered more fake images than real images during the learning process. Therefore, the prediction result of the GAN model tends to return a positive prediction. This can be seen by the high intensity of False Positive for GAN prediction result.

On the other hand, the combination of CNN and GAN can be used to improve the performance of the model. However, based on the result, the improvement is not significant. This might happen because of several factors:

1. The low performance of GAN compared to CNN making adding GAN to CNN does not lead to significant improvement on the detection process.
2. The combining method might be not optimal with just weighting the prediction results from both of them. There might be another ideal algorithm to maximizing the combination result of both models.

VI. Conclusion

The combination of the CNN and GAN models can be used to detect deepfake images from the real ones. However, this method has not yet been proven to lead to significant differences compared to doing detection only using CNN. Based on the current evaluation, the improvement of only 1 % to the final detection might not be worth the effort and resources to combine CNN and GAN. Therefore, using only

CNN models can be considered to have more value. However, seeing that there is indeed an improvement, this indicates that combining CNN and GAN has the potential to be implemented in the future. With better planning, better process steps, and better data, the combination of both models might produce a significant improvement in compared to current evaluation.

VII. Future Works

Here are some factors that need to be put attention on for future development of this system:

1. In the learning process, searching the parameter can be done by implementing hyperparameter tuning.
2. For every image, in both the learning and testing process, can be preprocessed first.
3. Putting more attention to the GAN model so the trained GAN model can have a significantly improved performance compared to the current model.

Source Code

The source code of this deepfake image detection program is accessible in:

https://github.com/Gulilil/IF4073_Deepfake_Detection

Acknowledgment

First, I would sincerely express my gratefulness to the Lord that this paper can be completed well and within the given deadline. Then, I would like to express my deepest gratitude to:

1. Mr. Dr. Ir. Rinaldi Munir, M.T. for all the Image Processing lectures that have been given to me,
2. My closest friends, that has been supporting me for the time being, and
3. The rightful owners of all the references that I used in writing this paper, for every knowledge that has been passed upon me.

References

[1] J. S. Rickson and M. Urwin, "What Is a Deepfake?" Accessed: Jan. 15, 2025. [Online]. Available: https://builtin.com/machine-learning/deepfake

[2] N. Hamiel, "Deepfakes proved a different threat than expected. Here's how to defend against them." Accessed: Jan. 15, 2025. [Online]. Available: https://www.weforum.org/stories/2025/01/deepfakes-different-threat-than-expected/

[3] A. Zola, "What is an image?" Accessed: Jan. 15, 2025. [Online]. Available: https://www.techtarget.com/whatis/definition/image

[4] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. 2016. Accessed: Nov. 04, 2024. [Online]. Available: https://www.deeplearningbook.org/

[5] IBM, "What are convolutional neural networks?" Accessed: Jan. 15, 2025. [Online]. Available: https://www.ibm.com/think/topics/convolutional-neural-networks

[6] R. Munir, "Convolutional Neural Network," 2024. Accessed: Jan. 15, 2025. [Online]. Available: https://informatika.stei.itb.ac.id/~rinaldi.munir/Citra/2024-2025/21-CNN-2024.pdf

[7] AWS, "What is a GAN?" Accessed: Jan. 15, 2025. [Online]. Available: https://aws.amazon.com/what-is/gan/

[8] GeeksForGeeks, "Generative Adversarial Network (GAN)." Accessed: Jan. 15, 2024. [Online]. Available: https://www.geeksforgeeks.org/generative-adversarial-network-gan/

Statement

Hereby, I state that this paper is written on my own, not an adaptation, or translation from other people's paper, and not plagiarism.

Bandung, 15th January 2024

Juan Christopher Santoso - 13521116