Generation of Traditional Japanese Patterns Using Deep Convolutional GAN

1st Yoga Afrizal Riandika
School of Computing
Telkom University
Bandung, Indonesia
yogaafrizal@student.telkomuniversity.ac.id

2nd Wikky Fawwaz Al Maki
School of Computing
Telkom University
Bandung, Indonesia
wikkyfawwaz@telkomuniversity.ac.id

***Abstract* — Japan has a variety of unique cultures, one of the unique cultures in Japan is the pattern of traditional clothes. The typical Japanese pattern has high popularity in the eyes of the world. Any variant of pattern, Starting from animals, plants and vectors typical of Japan. Examples of typical Japanese patterns are Uroko (鱗) which has a triangle pattern, Mame Shibori (豆 絞り) which has a dot pattern, then Tsubaki (椿) which has a camellia flower pattern and so forth. The Generator system for patterns was built to make patterns more varied and also have an unusual shape from other traditional Japanese patterns. By using Image Generation which was built with the Generative Adversarial Network method which will be built to make this system. The results of the generation of patterns have a fairly good output image quality, this is evidenced by the FID value of 0.48953 and the KID value of 3.557623863220215, both evaluation functions produce low values which means that the results of the generation have good quality. In addition, the generation has a combined new pattern output from several patterns from the test data.**

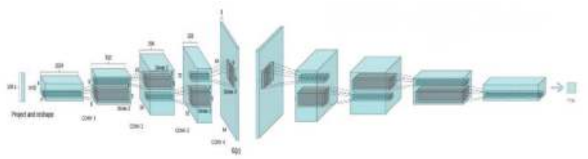
***Keywords*— Generative Adversarial Network (GAN), Japanese, Image synthesis**

I. INTRODUCTION

Traditional clothing is clothing that must be owned by every region in the world. Japan has traditional clothing which is very famous in foreign countries in the world. Introduced by Japan to the world using film media and also animation, traditional Japanese patterns have become famous in the world.

Traditional Japanese patterns are generally forms of animals, plants, or repeating patterns such as patterns of flowers and cherry trees, birds, and or other living things, namely patterns with traditional Japanese fan shapes. Traditional patterns typical of Japan have quite a number of types of these patterns, for example there is a pattern called Uroko (鱗) which has a triangular pattern, Mame Shibori (豆 絞り) which has a dot pattern, then there is Tsubaki (椿) which has camellia flower image patterns, and many more types of traditional patterns typical of this country of Japan. This typical Japanese pattern is usually used on traditional Japanese clothes such as kimono.

Artificial intelligence such as Generative Adversarial Network (GAN) can generate a traditional Japanese pattern. With this Japanese traditional pattern culture can develop with the production of a traditional Japanese pattern. With reference to the available datasets, GAN can produce Japanese and non-Japanese patterns.



From this research, the generation of images of traditional Japanese patterns has a background for evoking traditional Japanese patterns to increase cultural knowledge and also preserve culture, where the preservation of Japanese culture, especially in the aspect of traditional Japanese pattern images, still needs to be better introduced.

The limitation of this study is to generate images of traditional Japanese patterns using the Deep Convolutional Generative Adversarial Network architecture. The generation of this new Japanese traditional pattern image is the merging of 2 or more images of traditional Japanese patterns contained in 50 training data. By using the image synthesis technique on DCGAN, the image of the traditional Japanese pattern will be formed by utilizing the deep learning system on DCGAN.

By using machine learning, traditional patterns typical of Japan can be made using image synthesis techniques and using the Generative Adversarial Network architecture, traditional patterns typical of Japan are generated with several images of traditional patterns typical of the original Japanese country and a new traditional pattern is made of Japan from a reference Therefore, the preservation of traditional Japanese patterns can be enhanced by reviving these traditional Japanese patterns because patterns can be generated by combining 2 images of traditional Japanese patterns and becoming a new pattern. Generative Adversarial Network (GAN) can study the representation of various applications including image synthesis, semantic image editing, style transfer, image super resolution, and classification [1]. With this, it is hoped that other research related to image synthesis can develop rapidly and can create new research or methods related to image or photo synthesis.

II. METHODOLOGY

This section describe what was used in the research. Such as architecuture, method, evaluation function and so forth.

A. Generative Adversarial Network

The generative adversarial network architecture or GANs architecture can be abbreviated as a combination of supervised learning and unsupervised learning techniques. This Generative Adversarial Network uses supervised loss as part of the training component [1].

B. Deep Convolutional Generative Adversarial Network

Deep Convolutional GANs or commonly abbreviated as DCGAN is a GAN architecture that has differences in its image processing. In DCGAN, the processing combines the GAN architecture with the CNN (Convolutional Neural Network) architecture process. The following is a picture of how DCGAN works [5].

FIGURE 1.
DCGAN Architecture [3]

The picture above shows how the DCGAN architecture works. DCGAN uses the Deep Convolution technique, where there is a convolution transpose which makes the image bigger [2][3]. The Deep Convolutional GAN architecture is suitable for generating traditional Japanese patterns, because with this architecture the generation can produce better patterns and the training process will be more stable using DCGAN.

C. Pattern

In the image, there are many variations of the image available. Like images that have shapes and colors that represent shapes, images do not only have 1 variation but have many variations that are commonly known, such as abstract images, images of living things, animated images, photographic images, pattern images and so on.

The pattern image itself is a set of motifs put together. Therefore the pattern can be called an image that has regularity. Order here is a collection of motifs arranged in such a way as to form a regular pattern. The motif itself is a basic form that forms something that is known or not.

D. Image Generation

Image Generation or Synthesis is a task or system that can generate and generate new images from existing data sets. By referring to the existing test data, image generation will revive existing patterns or images and make it possible to produce new images from the generation results which combine several images into one image [4].

E. Kernel Inception Distance (KID)

Kernel Inception Distance or abbreviated KID is a machine learning evaluation technique that calculates the difference in the maximum squared average between feature representations taken from generated images and real images [6]. The results of the extraction of the two images will produce a KID score, the lower the KID score, the generated image has more similarities with the real image. The following is the formula for KID:

$$KID = MMD(f_{real\,,}\, f_{fake})^{degree}$$

To evaluate whether an image has good quality or not, the KID function will calculate feature extraction from freal and fake where f is feature extraction from fake and original images. Additionally MMD is the square of the Maximum Mean Discrepancy between inception representations. And KID has the advantage that KID is an unbiased estimator. By calculating this formula, KID can produce a score that can measure the quality of fake images, namely traditional Japanese pattern images generated from generators. In KID, it has a positive score from 0 to n, the closer n is to 0, the resulting image has good quality in the GAN architecture.

F. Fréchet inception distance (FID)

The Fréchet inception distance (FID) aims to evaluate the image quality of the generated Generator and evaluate the performance of the GAN architecture. This evaluation function has a data distribution that is modeled using a multivariate Gaussian distribution with mean μ and covariance ∑. FID can calculate the distance difference from the feature vectors obtained from the calculation of the original image and the image generated by the Generator. The smaller the FID score, the better the quality of the image [7], [8], [9]. The following is the formula for FID:

$$FID(r,g) = \left|\left|\mu_r - \mu_g\right|\right| + Tr(\textstyle\sum_r + \sum_g - 2(\sum_r \sum_g)^{\frac{1}{2}}$$

Information :
μ = average image extraction feature
r = original image (real)
∑ = image covariance matrix
G = fake image (generated)

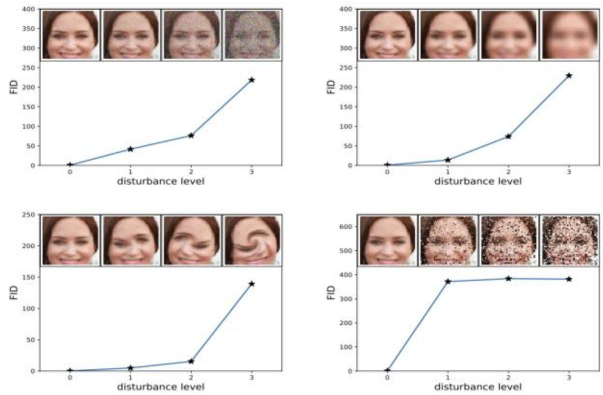


FIGURE 2.
FID score to image [12]

FID has a positive score from 0 to n depending on the results of the image generated by the generator. The greater the FID score, it can be interpreted that the results of image generation are still not of good quality, Figure 2 is an illustration of the FID score results for fake images generated by the generator.

III. METODE

This section explain about how system build to generate new model of japanese pattern and how system running the input program.

A. System Model Design

In this Final Project, analysis and trials are carried out to generate traditional Japanese patterns using the DCGAN architecture. The DCGAN architecture includes Generators and Discriminators to generate and improve the quality of the images produced by Generators. The generator creates a fake image from the latent space and the discriminator is in charge of checking the fake image, the Discriminator will distinguish between the original image and the fake image, the Generator will continue to improve the quality of the image until the Discriminator thinks the fake image is the original image, the visualization of the system design can be seen in Figure 3.

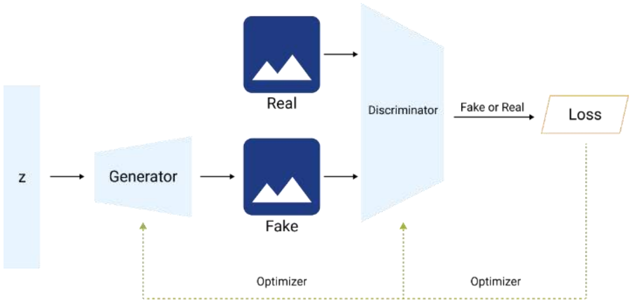


FIGURE 3.
System Design GAN process

The process of generating traditional Japanese patterns begins with generating images on a Generator which composes the latent space in such a way as to form a false image. The fake image will be juxtaposed with the original image derived from predetermined test data. The discriminator will compare the fake image, how similar it is to the test data. The loss function will improve the quality of the Generator and Discriminator so that it will create images that have better quality than before.

B. Test Scenario

This final project has a test scenario that describes how the test runs on the DCGAN architecture. By using 15,000 epochs. At every 100 epochs, the image generated by the Generator will be stored and can be compared every 100 epochs as shown in Figure 4.

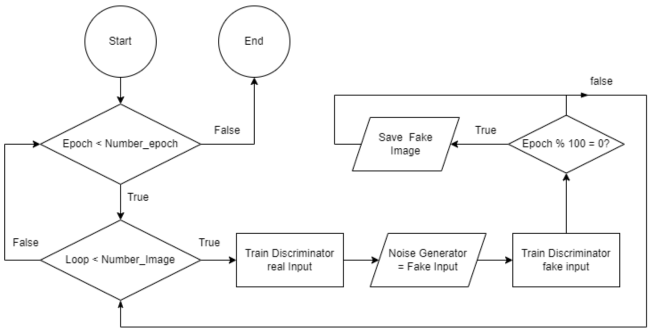


FIGURE 4.
Test Scenario

Figure 4 shows how the system flow works. Starting from checking whether the epoch that is being carried out is less than the epoch that was previously declared, namely 15,000 epoch. Epoch is an iteration for testing how many repetitions the system will perform to generate generators and recognize discriminators on fake images [10]. In this iteration, the Discriminator will first introduce the fake image, the Generator will generate the fake image which will later be studied by the Discriminator. Each iteration that is Modulus with 100 has a value of 0, then the fake image in that iteration is saved to the device. After the iteration exceeds the previously declared epoch, the system will stop.

C. Datasets

This training uses a dataset in the form of traditional Japanese pattern images with a png image format that has a size of 128x128. The total dataset is 50 traditional Japanese pattern images. This dataset of traditional Japanese patterns takes various existing patterns, such as traditional Japanese patterns Ume, Tsubaki, Nadeshiko (なでしこ), Karakusa (唐草), Yagasuri (やぐり), Seigaiha (行う), and so on. All traditional Japanese pattern images are downloaded from Google Images, Shutter Stock, Istock, etc. Figure 5 shows an example of a traditional Japanese dataset that is used as a real image or sample image to run the system to be created.

ginkgo	Shociku bai	kawa	Uroko	Mame Shibori

FIGURE 5.
Datasets

D. Generator

In the training process, Generator uses a latent space with a value of 100 which will be formed using a full convolutional layer. The generator performs upsampling using Convolutional 2D Transpose to produce a larger image size as output. The output from the Generator is an image that has an initial size of 128x128x3 with a fake image label. Visualization of the Generator architecture can be seen in Figure 6.

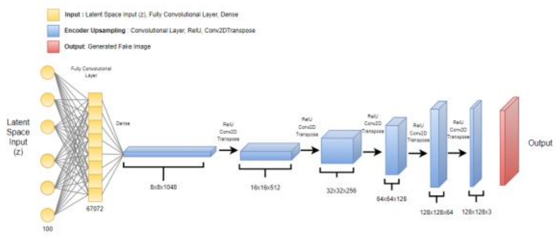


FIGURE 6.
Generator Function

E. Discriminator

During the training process, the Discriminator will compare the original and fake images by downsampling. Downsampling is done using LeakyRelu and Conv2D which ends up being an 8x8x128 image, the downsampled image will enter the Classifier stage which uses a Fully Convolutional Layer of several flatten layers. The result of training on the discriminator is to determine whether the original or fake image has been previously generated by the generator. Visualization of the Discriminator architecture can be seen in Figure 7.

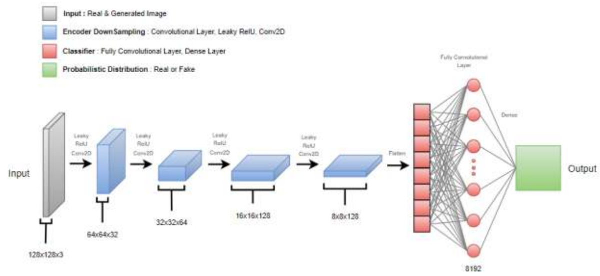


FIGURE 7.
Discriminator Function

F. Loss Function

To distinguish between fake images and original images, the loss function is used to distinguish between fake images and real images. In this study, the loss function used is Binary Cross Entropy (BCE) loss. Where this BCE loss is intended to carry out binary classification which has a target value between 0 to 1 [11]. BCE loss will evaluate and provide optimization to the Generator as well as the Discriminator, the loss in the Generator will be optimized for generating images that are more similar to the test data, while the Discriminator is optimized for the ability to distinguish which is the original image and which is a fake image generated by the Generator.

$$H_p(q) = -\frac{1}{N} y_i \,.\log\big(p(y_i)\big) + (1 - y_i)\,.\log\big(1 - p(y_i)\big)$$

IV. RESULT AND ANALYSIS

This section contains the results and analysis of the test. In this chapter, a test is carried out on a traditional Japanese pattern to be generated using the DCGAN architecture which uses the BCE loss function.deskripsi naratif. [10 pts]. Berikan kemungkinan pengembangan atau penelitian ke depan terkait penelitian ini

A. Evaluation

In image evaluation, an evaluation is carried out on traditional Japanese patterns that have been generated using the previously built DCGAN architecture. By using the image test data of traditional Japanese patterns as many as 50 images.

One of the important factors for evaluating the DCGAN architecture is based on visual analysis which states how similar the fake image generated by the Generator is to the original image from the test data. Another factor is the Generator which has the ability to generate various traditional Japanese pattern images, not only producing a large number of images but with the same image as well. The generator can also generate new traditional Japanese patterns by generating from 2 pattern sources. Pattern evaluation can be seen from the small loss function which indicates the generator can produce an image that is similar to the original image from the test data. But the loss function is not enough to evaluate GAN. Therefore it is made using another function.

In this study, a function is used to measure how well the DCGAN architecture can produce traditional Japanese pattern images, the FID and KID methods are used as a quantitative evaluation using test data of 50 images. The image used is a new image that has never been used in other studies.

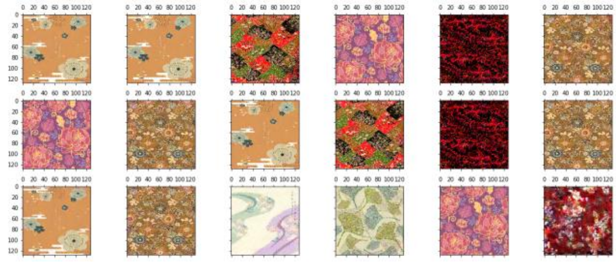


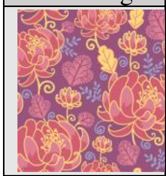
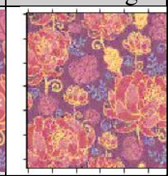
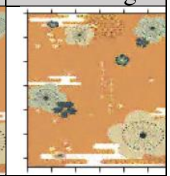
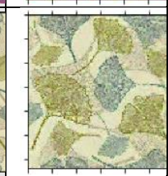
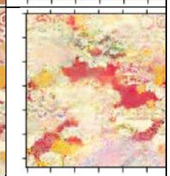
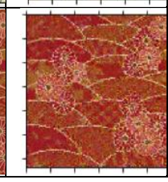
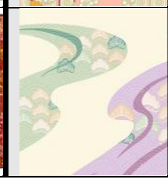
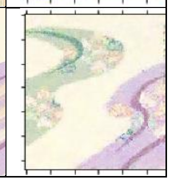
FIGURE 8.
Image Sample of DCGAN Geration Image Result

In Figure 8 is an example of a sample of traditional Japanese patterns. From the overall results obtained, the visualization of the traditional pattern from DCGAN has a good resemblance to the original traditional Japanese pattern. In addition, the traditional patterns generated by DCGAN have new patterns formed, these new patterns have several combined patterns from other patterns, this is because the test data varies so that the architecture tries to produce random patterns that are as similar as possible to each kernel.

B. Result and Image Analysis

To evaluate the results of the Generative Adversarial Network architectural model test, it can be done with periodic subjective human visual evaluations to determine the quality of the architecture, due to the weakness of the Generative Adversarial Network architecture, there is no objective way to evaluate the image quality of the test results. Images have various conditions depending on the model and also the shape of the pattern itself.

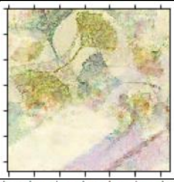
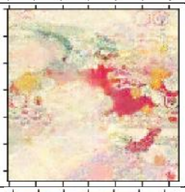
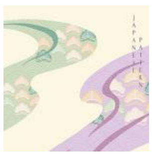
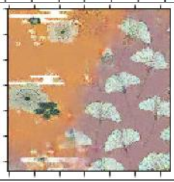
TABLE 1.
Generating Results of Traditional Japanese Patterns Using DCGAN

Real Image	Fake Image	Real Image	Fake Image
			
			
			

B. Result of Generating New Pattern

In the DCGAN training process using traditional Japanese pattern image test data, it not only produces images that are similar to the test data, but produces new images that are generated. The new image that is generated contains more than 1 image pattern. That way a new pattern is formed based on the tested test data. The following is a sample of the new pattern that is formed:

TABLE 2.
Generating New Pattern

New image form 2 pattern	Pattern Combination	
	Fake Image 1	Fake Image 2
		
		
		
		
		

It can be seen that there are several examples of new patterns that are formed based on the other 2 patterns, the patterns are formed randomly because the Generator generates patterns randomly as well. Pattern images can be formed with a generator architecture that regenerates patterns. Patterns are generated by utilizing existing neural networks in DCGAN. The neural network extracts each image that is used as test data by the discriminator, this feature extraction will be used as a benchmark for the generator to create fake images because each experiment will be carried out by the optimizer on the architecture of the generator and discriminator, so the generator can generate better images than before, and the dsicriminator it would be smarter to distinguish fake and real image.

This study uses a fake image generated by a generator. The generator synthesizes each image on the training data by using a latent space which contains random noise which is formed in such a way that the generator can generate new pattern images because the generator generates images that refer to 50 training data. Therefore, this study uses fake images to be used as a reference for generating traditional Japanese patterns. It can be seen in table 2, the Japanese pattern image can be formed from 2 fake image patterns that are generated by the generator, the generator selects or generates 2 random pattern images in 50 training data images.

The images generated by DCGAN are the images contained in the training data, each image contained in the training data is regenerated. However, a new image that often appears is the gingko pattern image (銀杏) which can be seen in table 2, but other patterns such as the kawa pattern image (かわ), the uroko pattern image (うろこ), the Shocikubai pattern image (松竹梅) and so on are also generated. by the generator as a new pattern.

C. KID and FID Evaluation

To evaluate the generated image results, it is proven by conducting a quantitative evaluation on DCGAN. Quantitative evaluation applied in this study is by using the FID and KID functions.

TABLE 3
Quantitative Result

Model	FID	KID
	15.000 *Epoch*	
DCGAN	0.48953	3.557623863220215

Based on the experimental results on the DCGAN architecture, the quality of the generated images is proven using quantitative evaluation. The FID function has a value of 0.48953 and the KID has a value of 3.557623863220215, the smaller the value of the two functions indicates the results of the image have good quality or it can be interpreted that the DCGAN image results have results that are similar to the actual test data. It can be concluded that the value of FID and KID against DCGAN produces good quality.

The use of FID and KID scores in this generation aims to see how well the DCGAN architecture revives traditional Japanese pattern images. With this, the generation of new traditional Japanese pattern images can be measured, by looking at the synthesis of images that have good quality, the generation of new images will produce images that are similar to the source image or the training data image.

V. CONCLUTION

In this research, experiments and analysis were carried out regarding the generation of traditional Japanese patterns using the Deep Convolutional Generative Adversarial Network architecture and the BCE Loss function as the loss function. It was carried out using test data of 50 traditional Japanese pattern images that had never been used for research. For evaluation using the FID and KID functions as well as human visual objective assessment to produce values as a measure of image quality that has been generated by DCGAN.

The generation that is often encountered is the image of the gingko pattern (銀杏) which can be seen in table 2, but other patterns such as kawa (かわ), uroko (うろこ), Shocikubai (松竹梅) and so on are also generated by the generator as new patterns. Generation produces images that are of fairly good quality, this is evidenced by the FID score of 0.48953, a small FID value means that the resulting image is of good quality, or in this case it can mean that it has a fairly good resemblance to the test data. The loss value has a small value for each and the KID function has a value of

3.557623863220215 where this value is a small value and means that the image quality produced by DCGAN is also of good quality. For images that can be assessed based on the visualization that is juxtaposed with the original test data images, it can be seen that the resulting images have a fairly high similarity.

In addition to generating a fairly good image, the results of this image generation produce new patterns based on existing test data. This new pattern is formed from 2 original traditional Japanese patterns. There are several new patterns formed from the 15,000 epochs that have been carried out.

REFERENCE

[1] Creswell, T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta, and A. A. Bharath, "Generative Adversarial Networks: An Overview," *IEEE Signal Processing Magazine*, vol. 35, no. 1, pp. 53–65, 2018, doi: 10.1109/MSP.2017.2765202.

[2] Prabhat, Nishant, and D. K. Vishwakarma, "Comparative Analysis of Deep Convolutional Generative Adversarial Network and Conditional Generative Adversarial Network using Hand Written Digits," *Proceedings of the International Conference on Intelligent Computing and Control Systems, ICICCS 2020*, no. Iciccs, pp. 1072–1075, 2020, doi: 10.1109/ICICCS48265.2020.9121178.

[3] Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *4th International Conference on Learning Representations, ICLR 2016 - Conference Track Proceedings*, pp. 1–16, 2016.

[4] Gregor, K., Com, D., Rezende, D. J., & Wierstra, D. (n.d.). *DRAW: A Recurrent Neural Network For Image Generation Ivo Danihelka*.

[5] Liu, J. Lv, X. Fan, J. Luo, and T. Zou, "Application of an Improved DCGAN for Image Generation," 2019.

[6] Y. Pang, J. Lin, T. Qin, and Z. Chen, "Image-to-Image Translation: Methods and Applications," *IEEE Transactions on Multimedia*, pp. 1–24, 2021, doi: 10.1109/TMM.2021.3109419.

[7] Bynagari, N. B. 2019. GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium. Asian Journal of Applied Science and Engineering. https://doi.org/ISSN 2305-915X(p); 2307-9584(e)

[8] Ci, Y., Ma, X., Wang, Z., Li, H., & Luo, Z. 2018. Userguided deep anime line art colorization with conditional adversarial networks. Proceedings of the 26th ACM International Conference on Multimedia. https://doi.org/10.1145/3240508.3240661

[9] Frans, K. 2017. Outline Colorization through Tandem Adversarial Networks. https://doi.org/arXiv:1704.08834

[10] M. Abdurrahman, N. H. Shabrina, and D. K. Halim, "Generative adversarial network implementation for batik motif synthesis," *Proceedings of 2019 5th International Conference on New Media Studies, CONMEDIA 2019*, pp. 63–67, 2019, doi: 10.1109/CONMEDIA46929.2019.8981834.

[11] Z. Zhang and M. R. Sabuncu, "Generalized Cross Entropy *Loss* for Training Deep Neural Networks with Noisy Labels," May 2018, [Online]. Available: http://arxiv.org/abs/1805.07836

[12] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "GANs trained by a two time-scale update rule converge to a local Nash equilibrium," Adv. Neural Inf. Process. Syst., vol. 2017-Decem, no. Nips, pp. 6627–6638, 2017.