


Article

Conditional Generative Adversarial Networks with Total Variation and Color Correction for Generating Indonesian Face Photo from Sketch

Mia Rizkinia ^{1,*}, Nathaniel Faustine ¹  and Masahiro Okuda ²¹ Department of Electrical Engineering, Faculty of Engineering, Universitas Indonesia, Depok 16424, Indonesia² Faculty of Science and Engineering, Doshisha University, Kyoto 610-0394, Japan

* Correspondence: mia@ui.ac.id

Featured Application: Indonesia's police force generates hand-drawn face sketches to reconstruct the facial visualization of a fugitive from eyewitness testimony. By the proposed method, more realistic photographs based on forensic drawings can be provided to improve the quality of visualizations given to the public regarding the suspected criminals in the all-points bulletin or wanted poster.

Abstract: Historically, hand-drawn face sketches have been commonly used by Indonesia's police force, especially to quickly describe a person's facial features in searching for fugitives based on eyewitness testimony. Several studies have been performed, aiming to increase the effectiveness of the method, such as comparing the facial sketch with the all-points bulletin (DPO in Indonesian terminology) or generating a facial composite. However, making facial composites using an application takes quite a long time. Moreover, when these composites are directly compared to the DPO, the accuracy is insufficient, and thus, the technique requires further development. This study applies a conditional generative adversarial network (cGAN) to convert a face sketch image into a color face photo with an additional Total Variation (TV) term in the loss function to improve the visual quality of the resulting image. Furthermore, we apply a color correction to adjust the resulting skin tone similar to that of the ground truth. The face image dataset was collected from various sources matching Indonesian skin tone and facial features. We aim to provide a method for Indonesian face sketch-to-photo generation to visualize the facial features more accurately than the conventional method. This approach produces visually realistic photos from face sketches, as well as true skin tones.

Keywords: GAN; sketch to photo; total variation; image generator



Citation: Rizkinia, M.; Faustine, N.; Okuda, M. Conditional Generative Adversarial Networks with Total Variation and Color Correction for Generating Indonesian Face Photo from Sketch. *Appl. Sci.* **2022**, *12*, 10006. <https://doi.org/10.3390/app121910006>

Academic Editors: Yang-Lang Chang, Mohammad Alkhaleefah and Tan-Hsu Tan

Received: 6 August 2022

Accepted: 25 September 2022

Published: 5 October 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

One of the main tasks of the police is to find missing persons or suspected persons (fugitives). Several methods can be used for this, including fingerprints, dactyloscopy, ballistic methods, lie detection methods, and one of the most widely used, face sketches [1]. Face sketches can provide more comprehensive information regarding a person's physical characteristics. Two of the benefits of face sketches, which Indonesian police have commonly used, are that they empower the public to help the police identify suspect, and they increase alertness in the general population [1]. In addition, face sketches can also be a reference that can be used in searching for people through the all-points bulletin or wanted person database (DPO in Indonesian terminology). Hence, face sketches are still the most effective way to catch suspects. However, making a high-quality face sketch with takes a long time. One example is sketching the faces of the Novel Baswedan case's suspects, which took up to 109 days after the victim was doused [2]. Therefore, various studies

were performed with the goal of making the process of searching with face sketches more effective, fast, and accurate.

Several research works have proposed methods for translating face sketches into photo images using deep learning. An earlier study by Zhang et al. [3] introduced a model developed from a fully convolutional representation learning, which experienced losses in pixel level and in generating texture details. The subsequent work employed generative models; the most widely-used of which is Generative Adversarial Networks (GAN) [4–8]. Isola et al. [8] developed a GAN-based framework, namely, conditional GAN (cGAN), which is also known as pix2pix. It deployed U-Net for the encoder–decoder function, PatchGAN for the discriminator, and L_1 term for the loss function. Pix2pix became a promising framework for extensive image-to-image translation tasks, e.g., sketches to photos, labels to scenes, day to night, etc. However, it failed to capture fine-grained details and photo-realistic textures [7].

Other GAN-based approaches have been extensively proposed. Lei et al. [9] combined a multiscale convolutional neural network (CNN) and a self-attention mechanism to transform face sketches to photos. Despite the fact that the quantitative performance outperformed the state-of-the-art methods, it suffered from a brightness problem affected by the weights of its feature extractor. Li et al. [7] introduced sketch-to-photo GAN (SPGAN) with a two-stage approach. They applied semantic loss in the first stage, then implemented color refinement loss, texture loss, a multiscale discriminator, and a patch level discriminator in the second stage. It achieved relatively low FID in the datasets. To generate detailed texture while preserving global structural information, Zhu et al. [10] proposed a sketch–transformer network consisting of three modules, i.e., a multiscale feature and position encoder, a self-attention module, and a multiscale spatially adaptive denormalization decoder. Compared to SPGAN, it outperformed with a lower FID for the same dataset. However, they have not been trained for Southeast Asian faces, particularly Indonesian faces.

The translation of face sketches into photos using GANs has been widely studied and applied by law enforcement agencies abroad [11]. By using realistic photos, the visualizations of fugitives becomes more like real faces, allowing the public to become more vigilant. However, GAN-based translation has a high bias towards the dataset used, especially in terms of generating the correct racial face features, including skin tone. Without proper adjustments to the framework, this method is unable to accurately represent Indonesian people. Furthermore, to be usable for Indonesian face sketches, a dataset in the form of pairs of face photos and sketches that can represent Indonesian people is needed, as well as a GAN model consisting of a generator and discriminator. The dataset can be collected from various sources with the same characteristics, namely having a Southeast Asian race, frontal face position, and bright lighting.

In this study, a GAN-based face sketch-to-photo generator is proposed, which combines U-Net and PatchGAN architectures on generators and discriminators inspired by pix2pix [8], thereby avoiding bias toward Indonesian faces. We found a potential area of improvement for the pix2pix framework, primarily since U-Net and PatchGAN have performed well in image translation tasks, such as creating images from image segmentation or sketches [8]. However, the translation of human faces is not easy because the resulting images look rough and retain some artifacts. To avoid this drawback, in this work, another term was also applied in addition to the loss from Conditional GAN and L_1 , i.e., the Total Variation (TV) term. TV loss was imposed to reduce artifacts contained in the images generated by the generator. Several scenarios with different L_1 and TV weight values were carried out and evaluated using two evaluation parameters, namely, Structural Similarity Index Measure (SSIM) and Fréchet Inception Distance (FID). The scenario that produced the best accuracy was analyzed and recommended as a system that could assist in Indonesian face photo generation using face sketches, namely cGAN-TV. Inspired by the sketch transformer [10], a color correction was added as postprocessing to improve the skin tone results.

The rest of this paper is organized as follows. Section 2 reviews the basics of GAN and the proposed method, including the dataset and the experiment scenario. Section 3 describes the results of our proposed method. Section 4 discusses the proposed method's performance compared to the baseline methods and the potential future works. Finally, we conclude our work in Section 5.

2. Materials and Methods

2.1. Generative Adversarial Networks (GAN)

Generative Adversarial Networks, commonly abbreviated as GAN, is a machine learning framework in which there are two working neural networks, namely generators and discriminators [12,13]. These two components are composed of several layers, one of which is CNN as a layer for image segmentation. However, the function of these two neural networks is different. The generator will create a fake sample, and the discriminator will evaluate whether the sample is genuine or fake in the form of a loss value. Both models will be trained until they reach stability [13].

The stability between the discriminator and generator may occur because the loss value generated by the discriminator in each epoch changes the weights in both models. In each epoch, the generator will create a sample of the input, which will be the input along with the ground truth of the generator input. The discriminator will assess the results generated by the generator and classify whether the sample is genuine or fake. This process will be repeated until the generator and discriminator obtain the most optimal weight. After the discriminator cannot distinguish between fake and real samples, it can be said that the GAN has reached a stable point or Nash equilibrium where the generator and discriminator have been optimized.

In recent years, GAN has become the best architecture for synthesizing a natural image based on the dataset used to train the GAN model. Some of the applications of GAN include image imputation, where the GAN model will try to increase the resolution of the image or complete the missing parts of an image, and image-to-image translation, where the GAN will perform an image translation, such as converting a sketch into an image. Image-to-image translation aims to obtain a mapping relationship between the input image and the output image through a neural network model and generate a new image associated with that relationship. Since this study aims to generate realistic photo images of Indonesian faces from sketches, this research is included in the category of image-to-image translation, or generative model.

2.2. Conditional GAN (cGAN)

The GAN architecture used as our baseline consists of two different neural networks trained simultaneously for 40,000 iterations. The generator that has been trained is one that operates by synthesizing an image from an input sketch.

The main task of a model generator in this research is to build a realistic image based on the input sketch and the given noise. To achieve an acceptable result, the generator must be able to find out what features are in the input, convert it into a feature map, and reconstruct the feature map into an image according to the desired target. Feature maps can be easily generated by CNN. However, the generator's task does not stop until it gets a feature map. The generator must make a new image reconstruction from the feature map. To carry out this task, a decoder layer is needed to change the encoder results' features. This architecture is commonly referred to as an autoencoder.

Since the features learned in the downsample can be lost in the encoding process, the autoencoder is not suitable for use as a generator of a GAN. Therefore, Ronneberger et al. introduced a new architecture called U-Net [14]. U-Net implements what are known as skip-connections. As the name suggests, skip-connections pass through multiple layers and take the output of one convolution layer as the input of another layer of the same dimension.

Using this architecture, the features found from each downsample can be decoded by the upsample. This provides much better image segmentation than a traditional auto-encoder which only upsamples the smallest feature vector in the code layer.

The generator architecture that will be used in this research is composed of upsample blocks composed of Conv2DTranspose, Dropout, with LeakyReLU activation and a down-sample composed of Conv2D, BatchNormalization, with LeakyReLU activation. The first downsample layer will not use BatchNormalization, and the first three upsample layers will drop out by 50%, this is done to prevent overfitting.

As seen from Figure 1, every downsample, other than the first downsample, has a skip connection to its upsample pair. Note that the upsample will accept two inputs. Input from the previous upsample output and from the downsample output. These two inputs have the exact dimensions and will be concatenated. For example, the first upsample will accept two inputs with dimensions of $1 \times 1 \times 512$, and after concatenation, the dimensions become $1 \times 1 \times 1025$.

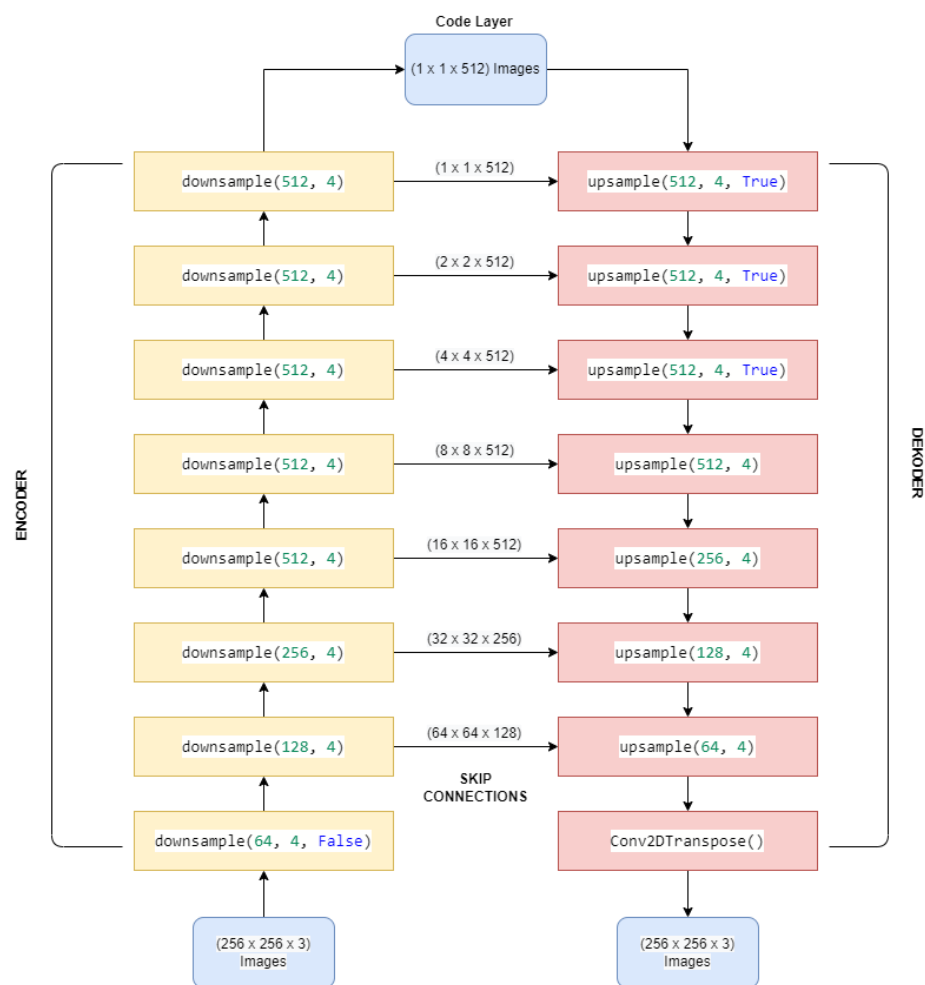


Figure 1. U-NET architecture used as the generator.

The discriminator has a vital role in the GAN model. Its functions are to evaluate the results produced by the generator and provide loss to change the weights in the generator and discriminator models. The discriminator is a component that distinguishes GAN from an ordinary autoencoder.

This research used the PatchGAN architecture, as shown in Figure 2, for the discriminator, due to its ability to enforce local-region consistency in the output image by using its patch. Suppose the ordinary discriminator uses the entire image to determine whether the image generated by the generator is genuine or fake. In that case, PatchGAN will try

to classify if every $N \times N$ patch in the image is real or fake. The discriminator will run convoluntarily on the entire image and find the average of all the responses given from that classification. The value of N can still give good results even though the value is smaller than the original image. It is advantageous because the smaller PatchGAN has fewer parameters, runs faster, and can be applied to large images. The implementation of PatchGAN has been widely used, especially for systems related to image or video [15,16].

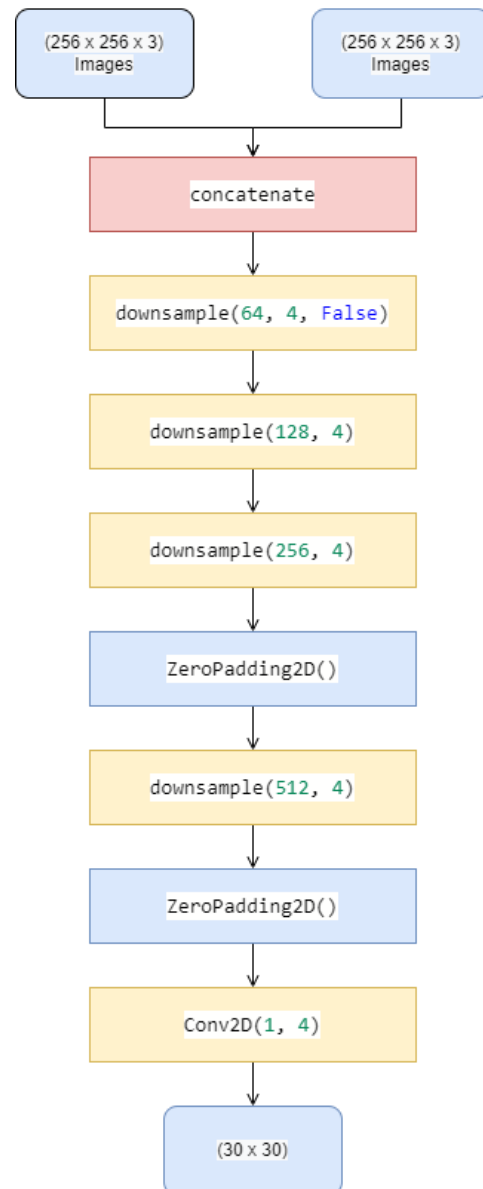


Figure 2. PatchGAN architecture used as the discriminator.

2.3. Proposed Method

Our model development began with collecting Southeast Asian face images from several datasets and dividing them into two different folders for training and validation. Before entering the training stage, the training data was augmented and preprocessed. After the model was completed from the training phase, the performance was evaluated using SSIM and FID. Based on these two metrics, the model performance was further assessed, and the training process was repeated with different values of L_1 and TV, until a model with optimal results had been obtained. Finally, the results from the optimal model were post-processed with a color correction technique to improve the visual quality,

bringing it closer to the real face photo. The evaluation was carried out again to observe the contribution of the color correction stage.

2.3.1. Dataset

The creation of a GAN model requires a dataset that consists of many samples, which are very influential for the generator results because the GAN model has a very high bias towards the dataset. Therefore, collecting image samples for a dataset is an essential stage. In this work, we used several datasets for compilation, namely, face images from generated photos [17], Fair Face Dataset [18], IMED [19], personal collection (volunteers), and CUHK Face Sketch database (CUFS) [20]. The dataset composition was 1243 from generated photos, 88 from CUFS, 20 from Fair Face Dataset, and 14 from IMED and private collections (from volunteers). The total of the entire dataset was 1365 faces. The whole dataset had no sketches other than CUFS. We generated sketches of these faces to complete the dataset representing Indonesian faces.

Datasets compiled from multiple sources were not the same size. Therefore, we changed the dimensions of each image in the dataset to the desired size of 256×256 pixels. The images and sketches were combined into the same image to simplify the project structure. The image was placed on the left, and the sketch was placed on the right. This image had dimensions of 256×512 . By combining images and sketches, we did not need separate folders for the two. The dataset was stored in one folder in JPG format.

After unifying all the datasets in the same folder, we separated the dataset used for testing purposes from the dataset used for validation purposes. Chen et al. [21], in their paper, used a ratio of 20:1 which resulted in 16,860 samples for training and 842 for validation. However, because our dataset was only 1365, while GAN requires a lot of data for training, we only used 10 samples for validation. The selection of data for validation was done randomly. The final dataset was placed in two different folders, namely, training and validation, where each image in the folder consisted of two images, namely, the original photo image and the sketch.

After preparing the dataset, we created a pipeline that contained several preprocessing steps before the data could be input into the GAN model, namely:

1. Splitting. In the splitting step, the image was separated into two different images, i.e., the original image and the sketch.
2. Converting to tensors. Because the GAN model can only accept tensor input, both images were converted into a tensor with the float data type.
3. Resize and augmentation. The resize step was carried out to ensure that the input tensor was in the appropriate size. In addition, we added noise to the training data, such as mirroring and cropping them arbitrarily in this step. This was to increase the variety of the training data.
4. Normalization. At this stage, each pixel in the image was normalized and converted to a corresponding value in a range of 0–1. This was to ensure that each pixel had the same distribution, and to speed up the GAN's convergence during training.

2.3.2. cGAN-TV Loss

In this study, the discriminator produced a final result with dimensions of 30×30 . Each grid had a value between 0 and 1, which indicated the fake and the real image, respectively, as seen in the final output of Figure 2. Each grid represented the original image patch with a size of 70×70 . The classification results were then be compiled and used to calculate the discriminator loss.

The generator obtains its input in the form of a sketch, and therefore we used the loss equation from the cGAN (cGAN loss). This equation was modified by adding the term $L_1(G)$ to the equation, as follows:

$$\min_G \max_D L_{\text{cGAN}}(G, D) + \lambda L_1(G) \quad (1)$$

where $L_{\text{cGAN}}(G, D)$ is the loss obtained from the sigmoid cross-entropy and $L_1(G)$ is the loss obtained from the Mean Absolute Error (MAE) between the image generated by the generator and the ground truth of the image, and λ is the weight that controls the contribution of L_1 loss. The use of L_1 loss aims to preserve the low-frequency features so as to improve the quality of the generated images [22].

Isola et al. [8] compared several generator loss experiments, i.e., $L_{\text{cGAN}}(G, D)$ only, $L_1(G)$, and both. The best results were obtained by using both, with $\lambda = 100$. The term $L_1(G)$ was imposed so that the image generated by the generator would be more similar to the ground truth. In the case of translation from sketch to image, this similarity is required.

However, the image results contained artifacts in the form of unsmooth borders that clearly separated colors among local regions of the face. To smooth the image and minimize noise that the generator may produce, this study used an additional term, i.e., Total Variation Loss (TVL). Images that have a high Total Variation (TV) also have high noise, and by regularizing the TV, the image can have minimized noise while maintaining the edges and promoting the connectivity of pixels in the images [23–26]. By using TVL, the generator can produce a smoother image while maintaining the edges of the image. The final formula for total loss with the addition of TVL is as follows:

$$\min_G \max_D L_{\text{cGAN}}(G, D) + \lambda_1 L_1(G) + \lambda_2 \text{TV}(G) \quad (2)$$

TVL is achieved by calculating TV from the image generated by the generator and weighted by the value of λ_2 . TV is defined as the sum of the absolute differences between adjacent pixels in the horizontal and vertical directions, as follows:

$$\text{TV}(G) = \sum_{i,j} |g_{i+1,j} - g_{i,j}| + |g_{i,j+1} - g_{i,j}| \quad (3)$$



2.3.3. Color Correction

The resulting image produced by cGAN might have incorrect skin tones, especially for the unseen data. Our preliminary research found that the cGAN generated face images with the same skin, hair, and eye colors. This is because, in the sketch to photo task, the cGAN only accepts input in the form of a grayscale face sketch and learns the translation pattern from the given paired training data. Hence, another piece of information should be added to assist the cGAN in understanding what skin tone should be shown in the resulting image. In this study, the skin tone is always light brown, and hair and eyes are always black. This is not realistic because Indonesian people have different skin tones. Therefore, color correction of the image is needed to make it more realistic and to better represent Indonesian faces.

In this study, there were three skin tone groups most commonly found in Indonesian society, namely pale white, fair, and tan. The system received additional input in the form of a skin tone group and corrected the skin tone in the image generated by the cGAN.

This study used the color transfer method developed by Reinhard et al. [27] and combined it with gamma alteration. By this method, a guide image's color characteristics were analyzed by a simple statistic to impose it on a target image. In the application of searching criminals, for example, the information about the skin tone can be inputted into the color correction phase. The correction was achieved by selecting an appropriate guide image so that its characteristic was applied to the target image. The configuration used can be seen in Table 1. Because cGAN successfully generated photo images without color skin bias, instead of applying color transfer, we only applied gamma alteration.

Table 1. Color Transfer Configuration. Photos by Generated Photos [17].

Scenario	Photo for Color Transfer	Gamma
Pale White	Not applied	0.8
Fair	 [17]	0.85
Tan	 [17]	1.2

2.3.4. Test Scenario

We conducted several scenarios for parameter tuning, i.e., finding the optimal values of λ_1 and λ_2 , which are the weight for the L_1 (G) and TV (G) terms, respectively. The values of λ_1 were varied, i.e., 0.01, 0.1, 1, 10, 100, 150. These values were chosen to determine the effect of the L_1 loss on the results generated by the generator. The baseline value was referred to the cGAN from Isola's work [8], in which the optimal value of λ_1 was 100. As for λ_2 , this value varied among 0.00001, 0.0001, 0.1, and 1. However, the values of 1 and 0.1 resulted in an over-smoothed image. Therefore, the value of λ_2 to be tested was very small. Table 2 shows the scenarios.

Table 2. Tuning-parameter scenarios.

Scenario	a	b	c	d	e	f	g*	h	i	j	k
λ_1	0.01	0.1	1	10	100	150	100	150		100	
λ_2			0.00001				0	0.0001	0.001	0.0001	0.00005

* This is the baseline method, i.e., cGAN (pix2pix) by Isola et al. [8].

All scenarios were trained with 40,000 iterations and the same dataset. The completed model was evaluated using the same dataset and the same evaluation parameters.

2.3.5. Evaluation Parameters

The evaluation was carried out using two different parameters. The first evaluation parameter was the Fréchet Inception Distance or abbreviated as FID. This parameter is one of the most frequently used metrics in evaluating generative models. FID uses the InceptionV3 model to generate a multivariate embedding distribution of two images. If the two images are very similar, then the resulting FID will be very low and close to 0. Conversely, a high FID means that the two images have very different distributions. In this study, the expected generator was a generator that could create an image that was very similar to the ground truth. Therefore, the use of FID was very relevant for this purpose, and the expected value of FID was close to 0.

The second parameter used was Structural Similarity Index Measurement (SSIM). SSIM works by comparing the structure of the image generated by the generator with ground truth. This study used SSIM because the resulting image was to be made as close as possible to the ground truth. Therefore, using SSIM, one can see how similar the resulting image is to the ground truth. The SSIM value was calculated and would produce a value from 0 to 1, where 1 meant that the two images compared were identical.

3. Results

3.1. Testing Results

The experiment was carried out by varying the λ_1 and λ_2 , which are the weight for L_1 and TV terms. Other parameters, such as the number of epochs and the dataset, will be equalized. The number of paired data used was 1365, divided into training and validation datasets. The selection of the validation dataset was done randomly and the number of datasets to be used for validation was 10. This number is relatively very small because the GAN generative model requires a lot of data for the training process. In some references, the paired data used in the training process number from tens to tens of thousands [7,10–12,28]. Therefore, in the experiment, we maximized the number of paired data for the training process; the cGAN-TV results can be seen clearly and analyzed correctly. In addition, we also prepared 68 testing datasets containing images of real Indonesian people obtained from photographs of students from the Department of Electrical Engineering, University of Indonesia, and several images from compiled datasets. The 68-testing dataset was used because it refers to the GAN study conducted by Chen et al. [21] where they used 5% testing of the total training data.

The training stage took approximately two hours and was carried out for 40,000 epochs, where, for every 1000 epochs, we displayed the temporary results of the cGAN-TV using one of the validation sketch images. This aimed to monitor the progress of the cGAN-TV more efficiently so that the training process could be manually stopped if it produced unexpected results, such as over smoothing.

The training process for each scenario was carried out one by one, and after each model was saved, the SSIM and FID values were calculated. The test results for each scenario can be seen in Table 3. The eleven scenarios were carried out in three phases. The first phase (a–f) aimed to find the optimal value of λ_1 , in which the λ_2 was set to a small value to avoid the over-smoothing effect. After we achieved the optimal λ_1 , then we stepped into the second phase (g–j) to find the optimal λ_2 with the constant λ_1 optimal value. The last stage was to test any higher λ_1 and λ_2 values. The visual results of each scenario can be seen in Figure 3.

Table 3. Results of tuning-parameter scenarios.

Scenario	a	b	c	d	e	f	g*	h	i	j	k
λ_1	0.01	0.1	1	10	100	150	100	150		100	
λ_2			0.00001				0	1×10^{-4}	1×10^{-3}	1×10^{-4}	5×10^{-5}
SSIM	0.695	0.756	0.725	0.763	0.834	0.809	0.815	0.823	0.832	0.837	0.813
FID	220.976	235.060	241.096	162.398	94.705	97.285	97.078	101.960	101.114	97.050	115.742

* This is the baseline method, i.e., cGAN (pix2pix) by Isola et al. [8].

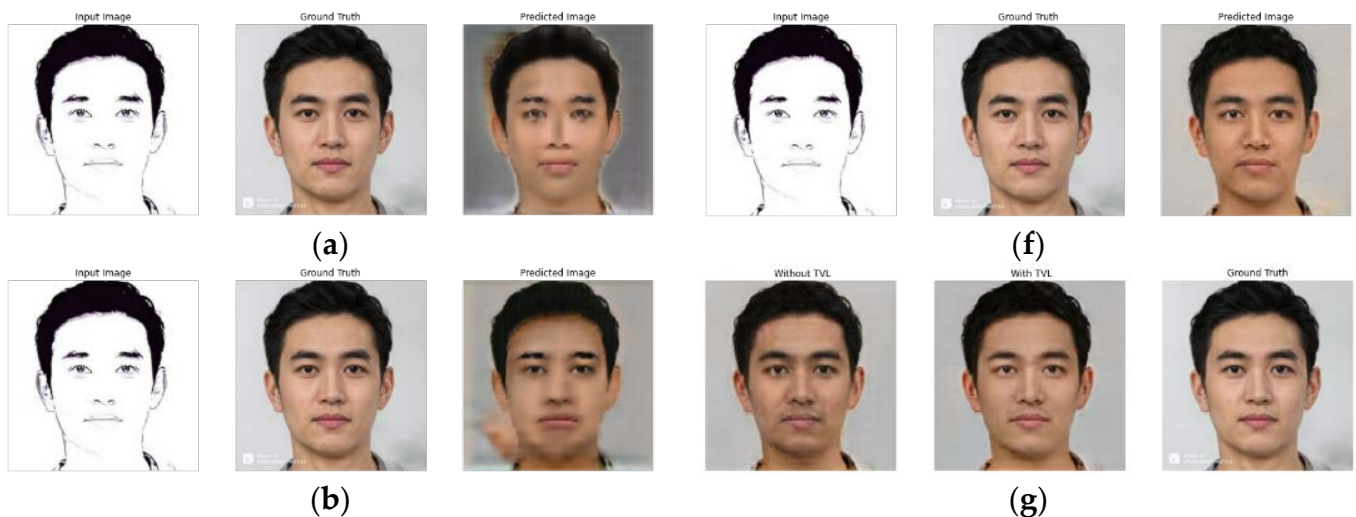


Figure 3. Cont.

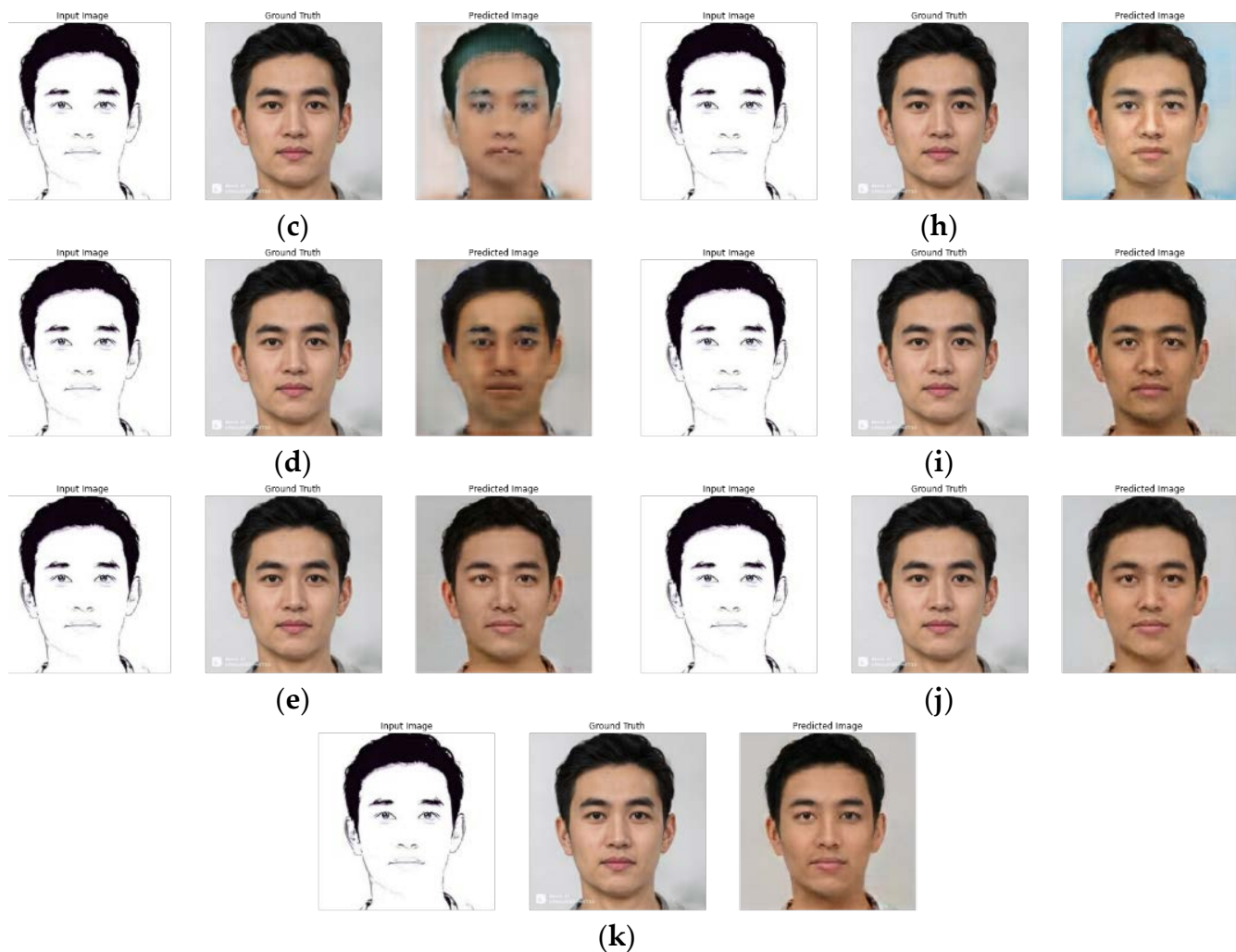


Figure 3. Visual result of each scenario. (a–k) Visual result of scenario a–k, respectively; (g) is the result of pix2pix (baseline method) [8]. The ground truth images are from generated photos [17].

It should be noted that the SSIM value was calculated by comparing the grayscale structure of the image generated by the generator and the original image of the sketch. In comparison, the FID value was achieved by comparing the two distributions of the entire validation image generator with the distribution of all the original images. The color difference in the image would greatly affect the FID value while having no impact on the SSIM value. Therefore, every time an evaluation was carried out, the SSIM would not change, while the FID would always change because the generator results would generate random background and skin tones.

Based on the test results, it can be seen that the effect of λ_1 was very significant on the SSIM and FID values. The lowest SSIM value was 0.695, found in the scenario, with a very small λ_1 value of 0.01, and the largest FID value was achieved in scenario c, where the λ_1 value was 1. However, when the λ_1 value was too large, it appears that the SSIM value was actually decreased, and the FID value increased. This is due to overfitting, and the model could not generalize properly. In scenario e, the SSIM value was 0.834, and the FID value was 94.705. When the λ_1 value increased to 150 with the same λ_2 value in scenario f, the SSIM dropped to 0.809, while the FID increased to 97.285. The λ_2 value also had a significant impact, as seen in scenarios e and g, where the SSIM value increased to 0.834 from 0.815, and the FID value decreased to 94.705 from 97.078. In addition, from scenario g, which used the cGAN (pix2pix) method [8], we conclude that the TV term contributed to better visual results than that of the L_1 term only (cGAN). The cGAN-generated image

contained artifacts around the mouth and uneven skin tone. This shows that our proposed method successfully improved the baseline method.

From the eleven scenarios that were carried out, scenario e produced the best evaluation value, with the $\lambda_1 = 100$, and $\lambda_2 = 0.00001$. Figure 4 shows the resulting photo images of the best scenario of cGAN-TV.

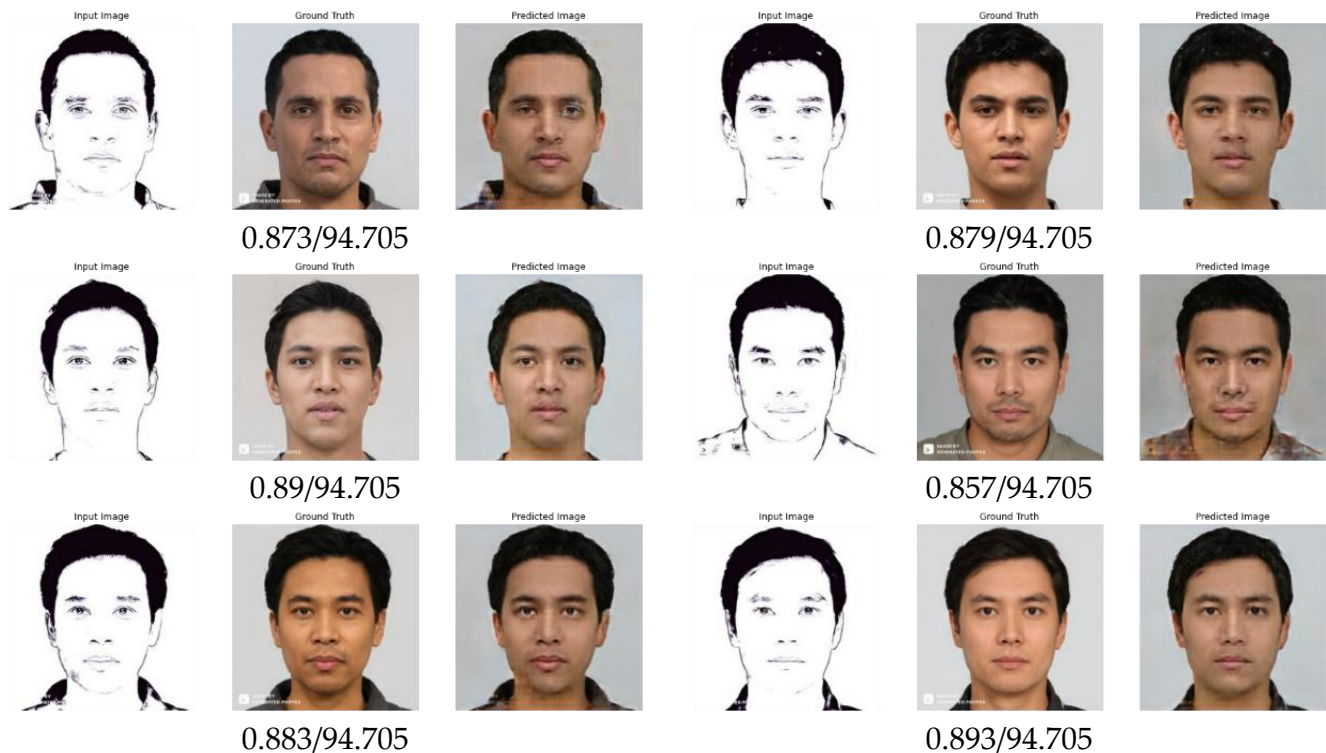


Figure 4. Result of cGAN-TV on validation dataset (ground truth images by generated photos [17]) with the best scenario. The values below the predicted images represent SSIM/FID from the cGAN-TV results.

Table 4 shows the SSIM and FID evaluation values from the validation and testing datasets. It can be observed that the SSIM value generated from the testing dataset was 0.06 lower than that of the validation dataset. The obtained FID value was also higher than that of the validation dataset. This is because the photo images in the testing dataset came from real data and had never been seen before by the model. Meanwhile, the validation data was collected from the same source as the training dataset. The validation and training datasets had the same face size, head position, and lighting. While the test dataset had a slightly different configuration.

Table 4. Result of cGAN-TV on validation and testing data.

Dataset	SSIM	FID
Validation	0.83	94.705
Testing	0.73	93.019

Figure 5 shows three examples of the results of the testing phase. Visually, the generator could create a similar image, but the result was not as good as the validation dataset. This is because the character of the images in the training dataset was quite different from those in the testing dataset, which were collected from real persons. In addition, this was caused by the sketch dataset used, which was not perfectly generated. However, our GAN was able to create images that were quite similar to the corresponding original images, with SSIM values of 0.83, 0.74, and 0.71, with an FID of 93.02.

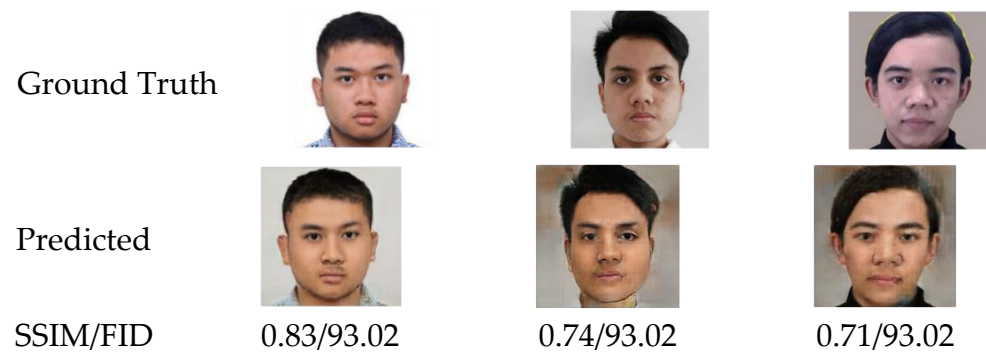


Figure 5. Results of cGAN-TV on the testing data.

3.2. Color Correction Evaluation

As shown in Figure 4, the images produced by cGAN-TV tended to have a similar skin tone. Hence, a color correction was required to make the photo images more realistic and more in accordance with the ground truth. By applying the configuration contained in Table 2, the resulting image could be more visually similar to the original image.

The first skin tone configuration applied for color correction was for pale white, where the color transfer was not required due to the satisfying results of this color category. We only needed to set the gamma at 0.8. As shown in Figure 6, this configuration succeeded in producing the skin tone of the resulting image that was more similar to the skin tone of the original image.

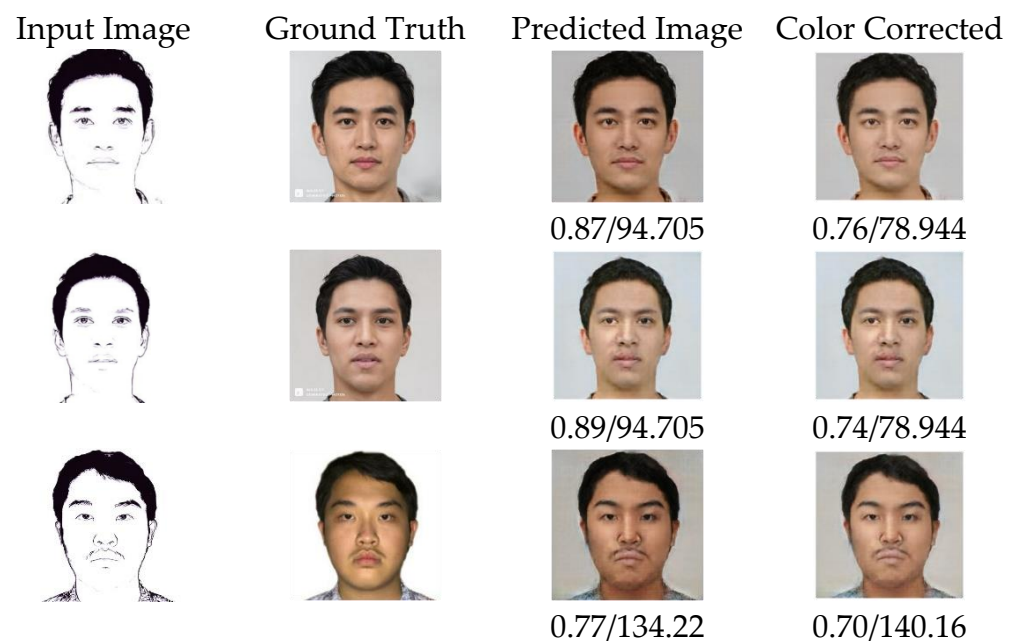


Figure 6. Results of pale white configuration in correcting the color of predicted images. The values below the images represent SSIM/FID. The ground truth images are from generated photos [17] and volunteers (private collection).

The second configuration was fair, in which the color transfer was carried out, and the gamma was set to 0.85. In this case, the color transfer must be performed, because the gamma change only adjusts the exposure of the image and cannot change the skin tone. However, when we only performed the color transfer, the resulting images were too dark and unnatural. Therefore, by using gamma 0.8, the resulting images looked more natural, with the skin tone turning yellowish, as shown in Figure 7.

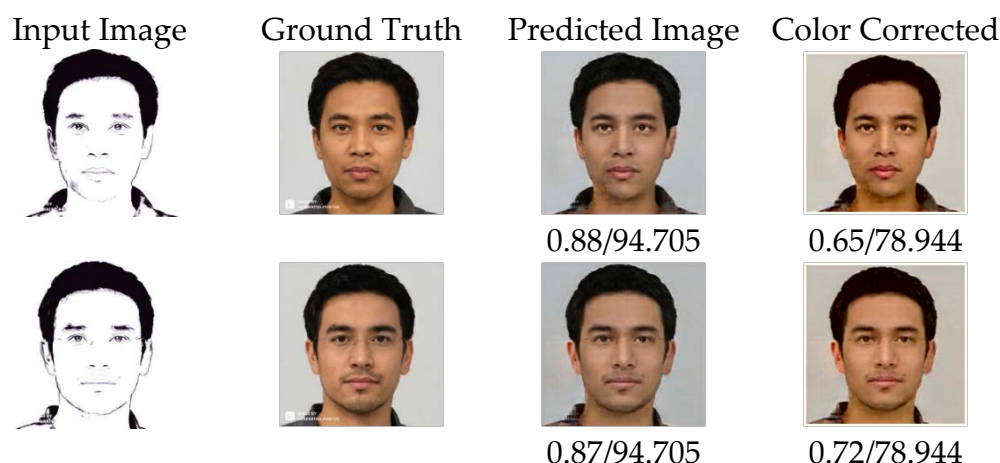


Figure 7. Results of fair configuration in correcting the color of predicted images. The values below the images represent SSIM/FID. The ground truth images are from generated photos [17].

As shown in Figure 7, the third or tan configuration used color transfer and changed the gamma to 1.2. As with the fair configuration, this configuration also used two stages, namely color transfer, and gamma changes. The gamma of the image was increased to 1.2. The effect of the tan configuration can be seen in Figure 8.

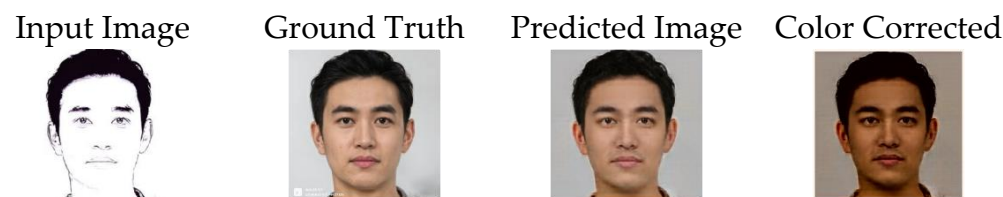


Figure 8. Results of tan configuration in correcting the color of predicted images. The ground truth images are from generated photos [17].

Table 5 shows the result evaluation of cGAN-TV with the implementation of color correction. Although the color correction visually gives a more realistic result, the SSIM value of the color correction implementation decreased from 0.83 to 0.76. This was caused by the SSIM calculation process, which begins with converting the image to grayscale. Since the image has undergone a gamma change, the grayscale result is different for the one without the gamma change. This inappropriate grayscale image causes the image structure to not be in accordance with the original image. However, the decrease in SSIM visually did not have the effect of reducing the visual quality of the results, therefore, it was still tolerable. On the other hand, the FID value increased by 16% from 94,705 to 78,944. This is because the FID evaluation takes color similarity into account, and with color correction, the image between the GAN result and the original image has a more similar distribution when compared to that without color correction. Overall, the implementation of color correction provided a more realistic image result and a better FID value. This is prioritized over a high SSIM value because the improvement in visual quality produced is more helpful for ordinary users (in this case the Indonesian people) in assessing the similarity of faces between two images.

Table 5. Result of cGAN-TV with and without color correction.

Scenario	SSIM	FID
Without Color Correction	0.83	94.705
With Color Correction	0.76	78.944

4. Discussion

4.1. Comparison with Other Methods

We include the comparison between the proposed cGAN-TV and other generative models to see how well it performed in comparison with others. In this case, the compared models are autoencoder and pix2pix (cGAN) [8] because autoencoder is a part of our architecture and cGAN is the baseline of our method. The comparison of the results is shown in Figure 9.

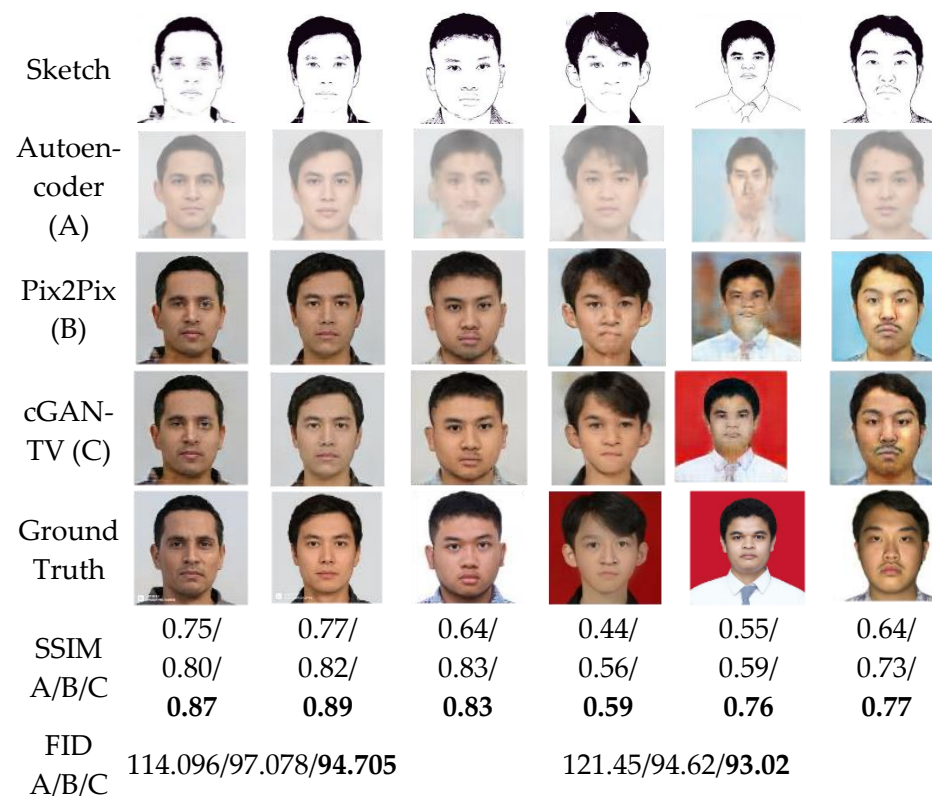


Figure 9. Our method with best scenario compared with cGAN (pix2pix) and autoencoder. The ground truth images are from generated photos [17] and volunteers (private collection).

Visually, it is clear that the resulting image from our cGAN-TV (best scenario) is far superior and more similar to the original image. The autoencoder-generated image tends to be more blurry and has not been able to describe facial characteristics well, especially in the last dataset, where the autoencoder failed to generate an image. A more detailed comparison of SSIM and FID can be seen in Table 6. Quantitatively, the best GAN method scenario is also superior in both SSIM and FID metrics. The autoencoder failed to produce the face, as shown in the last dataset in Figure 9. This was due to the less varied and large dataset. Almost the entire training dataset was taken from the generated photos dataset [17], and because the testing dataset was not from the same source, the autoencoder failed to create images. On the other hand, GANs could still produce images, even though the resulting images are not as good as the validation dataset when viewed visually.

Table 6. SSIM and FID comparison between cGAN-TV and other methods.

Method	Validation		Testing	
	SSIM	FID	SSIM	FID
Autoencoder	0.69	114.096	0.64	105.638
Pix2pix	0.815	97.078	0.71	101.376
cGAN-TV	0.83	94.705	0.73	93.019

4.2. Results of Hand-Drawn Sketch Input

Because the number of labelled data with hand-drawn sketches has not been adequate for training the GAN-based model, this research used synthetic sketches. Meanwhile, generating images from arbitrary hand-written sketches is a very challenging task because the accuracy depends on the person drawing the sketch and the roughness of the sketch. To examine the generalization capability of our model when it comes to the hand-drawn sketch input that resembles the original to some extent, we tested it using hand-drawn sketches from the CUHK dataset [20]. The results are depicted in Figure 10. These show that the system has a certain degree of versatility.

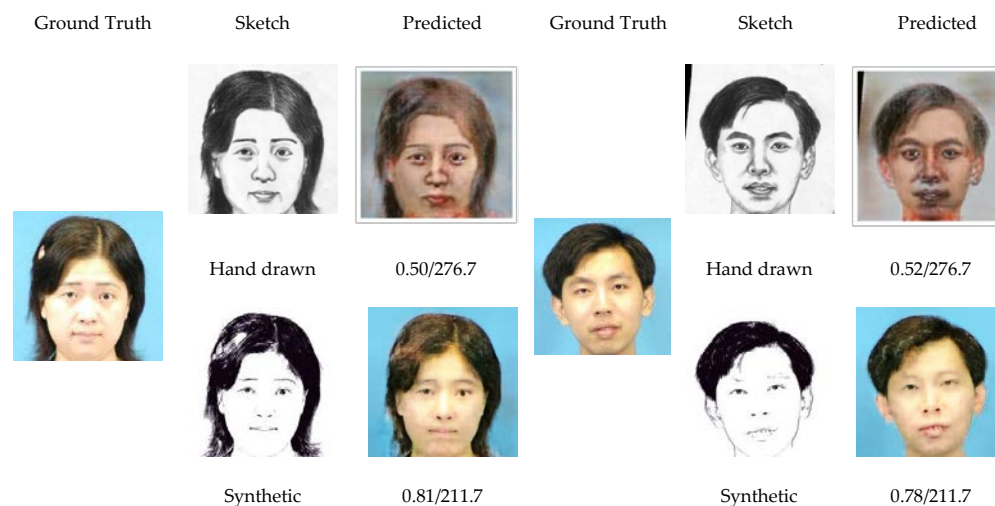


Figure 10. Results of our method using hand-drawn sketch input [20]. The values below the images represent SSIM/FID.

Visually, the predicted results with hand-drawn inputs are fairly similar, but less realistic, than those with synthetic sketch inputs. The reason is that the character of the hand-drawn samples used in this testing differs from our synthetic sketches for the training phase. They have firmer lines on the edges and shading lines that were not shown in our synthetic sketches. Our training process did not learn these features, therefore, our model failed to translate them correctly and generated artifacts, especially in the shading area. This problem results in lower SSIM and higher FID values. However, we cannot depend on these quantitative values, since they are sensitive to pixel misalignment, which is prone to occur in the drawing process. The misalignment contributes to worsening SSIM and FID. Hence, the visual quality evaluation is sufficient.

4.3. Loss Behavior

To evaluate the influence of TVL in the learning process, we observed the loss behavior along the epochs produced by cGAN (with L₁ loss) and cGAN-TV. As depicted in Figure 11, cGAN-TV successfully decreased the loss of cGAN. Every point of the curves represents the loss of every 1000 epochs that reach values between 6–45. The values are relatively large due to the value of λ_1 , which is 100. Despite the fluctuation, cGAN-TV obtained loss values that are mostly lower than those of cGAN, including at the last epoch. In the beginning of the training, cGAN-TV reached the lowest loss, however, the generated images were still far from realistic features, where the overfitting occurred at this epoch, as shown in Figure 12. This shows that a cGAN-based model needs a huge number of epochs to generate an optimal result.

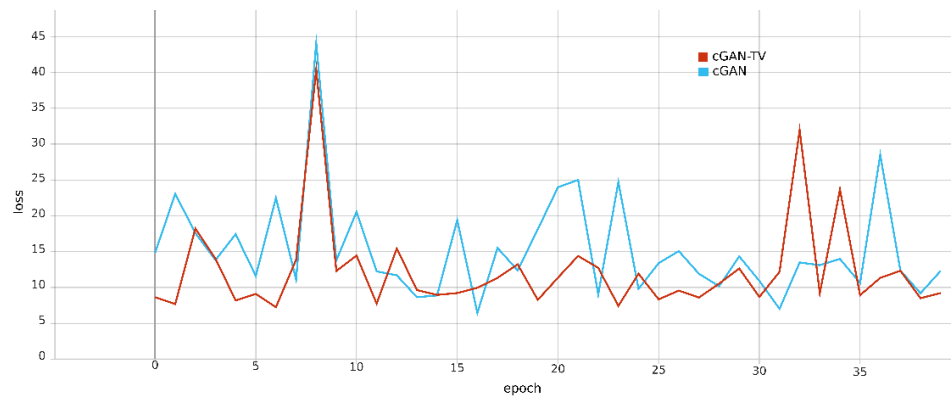


Figure 11. Loss values of cGAN and cGAN-TV for every 1000 epochs.

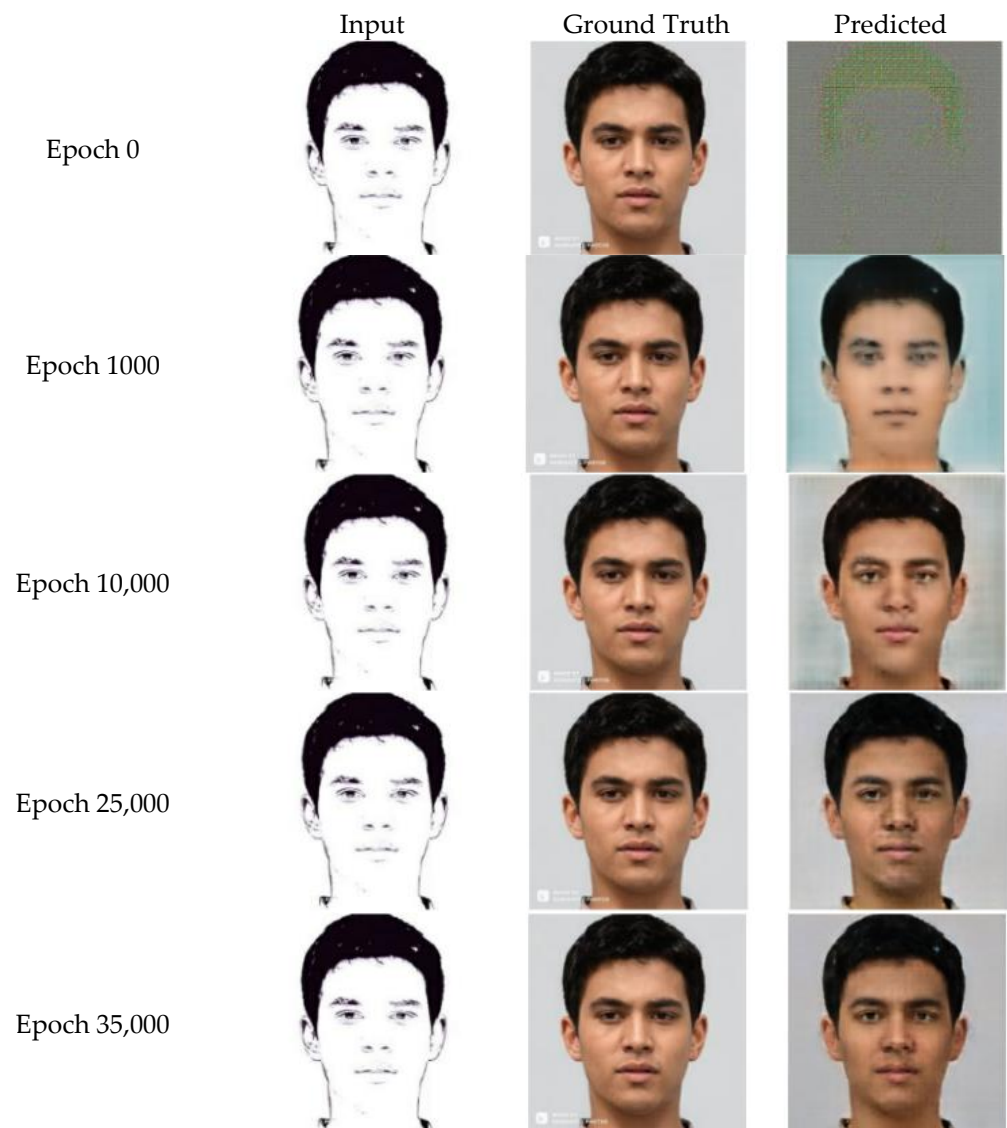


Figure 12. Generated images of increasing epoch show the improvement of the visual quality. (ground truth images by generated photos [17]).

4.4. Future Works

Based on our experiment, we found some recommendations for improving future works that could be performed to enhance the results. Since GAN requires a lot of data for

training, and since the number of pair images containing the three skin tones of Indonesian faces is still limited, an effort to collect more such images is required. It is also recommended to use a more varied dataset so that the GAN model can be more generalized to the Indonesian people. As for sketch images, using hand-drawn sketches instead of synthetic images will give closer images to the ground truth and be more appropriate for the targeted application. Furthermore, for the architecture itself, the color correction could be added inherently to the loss function as another regularizer, thus the color correction process would need no separate postprocessing step, and the result would be more precise.

5. Conclusions

In this paper, we present a new conditional GAN sketch-to-photo translation model for the task of generating images of Indonesian face photos. We developed our model with pix2pix as a baseline, added with total variation (TV) loss and a color correction stage. The TV loss gives a smooth effect in a way that makes the resulting images more similar to the ground truth. The color correction using guide images and gamma alteration aims to reconstruct true skin tones. The experiment shows that our proposed method successfully outperformed the conventional methods. Our contribution lies in improving the baseline's performance, which includes the avoidance of bias towards the Indonesian face dataset, the reduction of artifacts, and the production of uneven and similar skin tones, thereby resulting in more plausible images.

Author Contributions: Conceptualization, M.R. and N.F.; methodology, M.R. and N.F.; software, N.F.; validation, N.F., M.R. and M.O.; formal analysis, M.R.; investigation, M.R. and M.O.; resources, M.R.; data curation, N.F.; writing—original draft preparation, M.R. and N.F.; writing—review and editing, M.R. and M.O.; visualization, N.F.; supervision, M.O.; project administration, M.R.; funding acquisition, M.R. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Universitas Indonesia through Hibah Publikasi Artikel di Jurnal Internasional Kuartil Q1 dan Q2 (Q1Q2) research grant, under contract number NKB-0309/UN2.R3.1/HKP.05.00/2019.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Restrictions apply to the availability of these data. Data was obtained from generated.photos [17], FairFaceDataset [18], IMED [19], personal collections (volunteers), and CUHK Face Sketch database (CUFS) [20] with the permission.

Acknowledgments: The authors acknowledge Universitas Indonesia for the funding support (Q1Q2 research grant under contract number NKB-0309/UN2.R3.1/HKP.05.00/2019).

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

References

1. Kurniawan, M.H. *Penggunaan Metode Sketsa Wajah Dalam Menemukan Pelaku Tindak Pidana—Ums Etd-Db*; Universitas Muhammadiyah Surakarta: Surakarta, Indonesia, 2009.
2. "Begini Rumitnya Membuat Sketsa Pelaku Penyiraman Novel Baswedan." n.d. Available online: <https://news.detik.com/berita/d-3583949/begini-rumitnya-membuat-sketsa-pelaku-penyiraman-novel-baswedan> (accessed on 10 September 2021).
3. Zhang, L.; Lin, L.; Wu, X.; Ding, S.; Zhang, L. End-to-End Photo-Sketch Generation via Fully Convolutional Representation Learning. In Proceedings of the 5th ACM on International Conference on Multimedia Retrieval, Shanghai, China, 23–26 June 2015; pp. 627–634. [CrossRef]
4. Chen, W.; Hays, J. SketchyGAN: Towards Diverse and Realistic Sketch to Image Synthesis. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 9416–9425. [CrossRef]
5. Li, Y.; Wu, F.; Chen, X.; Zha, Z.J. Linestofacephoto: Face Photo Generation from Lines with Conditional Self-Attention Generative Adversarial Network. In Proceedings of the 27th ACM International Conference on Multimedia, Nice, France, 21–25 October 2019; pp. 2323–2331. [CrossRef]

6. Sangkloy, P.; Lu, J.; Fang, C.; Yu, F.; Hays, J. Scribbler: Controlling Deep Image Synthesis with Sketch and Color. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 6836–6845. [[CrossRef](#)]
7. Li, L.; Tang, J.; Shao, Z.; Tan, X.; Ma, L. Sketch-to-Photo Face Generation Based on Semantic Consistency Preserving and Similar Connected Component Refinement. *Vis. Comput.* **2021**, 1–18. [[CrossRef](#)]
8. Isola, P.; Zhu, J.-Y.; Zhou, T.; Efros, A.A. Image-to-Image Translation with Conditional Adversarial Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017.
9. Lei, Y.; Du, W.; Hu, Q. Face Sketch-to-Photo Transformation with Multi-Scale Self-Attention GAN. *Neurocomputing* **2020**, *396*, 13–23. [[CrossRef](#)]
10. Zhu, M.; Liang, C.; Wang, N.; Wang, X.; Li, Z.; Gao, X. A Sketch-Transformer Network for Face Photo-Sketch Synthesis. *IJCAI Int. Jt. Conf. Artif. Intell.* **2021**, *2*, 1352–1358. [[CrossRef](#)]
11. Hu, M.; Guo, J. Facial Attribute-Controlled Sketch-to-Image Translation with Generative Adversarial Networks. *EURASIP J. Image Video Process.* **2020**, *2020*, 1–13. [[CrossRef](#)]
12. Zhang, F.; Zhao, H.; Ying, W.; Liu, Q.; Raj, A.N.J.; Fu, B. Human Face Sketch to Rgb Image with Edge Optimization and Generative Adversarial Networks. *Intell. Autom. Soft Comput.* **2020**, *26*, 1391–1401. [[CrossRef](#)]
13. Goodfellow, I.J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative Adversarial Nets. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QU, Canada, 8–13 December 2014; Volume 3, pp. 2672–2680.
14. Ronneberger, O.; Fischer, P.; Brox, T. U-Net: Convolutional Networks for Biomedical Image Segmentation. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Munich, Germany, 5–9 October 2015.
15. Demir, U.; Unal, G. Patch-Based Image Inpainting with Generative Adversarial Networks. *arXiv* **2018**, arXiv:1803.07422.
16. Chang, Y.-L.; Liu, Z.-Y.; Lee, K.-Y.; Hsu, W. Free-Form Video Inpainting with 3D Gated Convolution and Temporal PatchGAN. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27–28 October 2019; pp. 9066–9075.
17. Image Datasets for Machine Learning | Generated Photos. Available online: <https://generated.photos/datasets> (accessed on 10 September 2021).
18. Karkkainen, K.; Joo, J. FairFace: Face Attribute Dataset for Balanced Race, Gender, and Age for Bias Measurement and Mitigation. In Proceedings of the 2021 IEEE Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 3–8 January 2021; pp. 1547–1557.
19. Liliana, D.Y.; Basaruddin, T.; Oriza, I.I.D. The Indonesian Mixed Emotion Dataset (IMED): A Facial Expression Dataset for Mixed Emotion Recognition. In Proceedings of the ACM/International Conference on Artificial Intelligence and Virtual Reality, Nagoya, Japan, 23–25 November 2018; pp. 56–60.
20. Wang, X.; Tang, X. Face Photo-Sketch Synthesis and Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2009**, *31*, 1955–1967. [[CrossRef](#)] [[PubMed](#)]
21. Chen, S.Y.; Su, W.; Gao, L.; Xia, S.; Fu, H. DeepFaceDrawing: Deep Generation of Face Images from Sketches. *ACM Trans. Graph.* **2020**, *39*, 72. [[CrossRef](#)]
22. Abu-Srhan, A.; Abushariah, M.A.M.; Al-Kadi, O.S. The Effect of Loss Function on Conditional Generative Adversarial Networks. *J. King Saud Univ. Comput. Inf. Sci.* **2022**, *34*, 6977–6988. [[CrossRef](#)]
23. Strong, D.; Chan, T. Edge-Preserving and Scale-Dependent Properties of Total Variation Regularization. *Inverse Probl.* **2003**, *19*, S165–S187. [[CrossRef](#)]
24. Minaee, S.; Minaei, M.; Abdolrashidi, A. Palm-GAN: Generating Realistic Palmprint Images Using Total-Variation Regularized GAN. *arXiv* **2020**, arXiv:2003.10834.
25. Chadha, A.; Britto, J.; Roja, M.M. ISeeBetter: Spatio-Temporal Video Super-Resolution Using Recurrent Generative Back-Projection Networks. *Comput. Vis. Media* **2020**, *6*, 307–317. [[CrossRef](#)]
26. Wang, Z.; Chen, J.; Hoi, S.C.H. Deep Learning for Image Super-Resolution: A Survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *43*, 3365–3387. [[CrossRef](#)] [[PubMed](#)]
27. Reinhard, E.; Adhikhmin, M.; Gooch, B.; Shirley, P. Color Transfer between Images. *IEEE Comput. Graph. Appl.* **2001**, *21*, 34–41. [[CrossRef](#)]
28. Yi, R.; Liu, Y.-J.; Lai, Y.-K.; Rosin, P.L. APDrawingGAN: Generating Artistic Portrait Drawings from Face Photos with Hierarchical GANs. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 10743–10752.