Received: July 20, 2020. Revised: August 17, 2020.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Facial Inpainting Using Generative Adversarial Network with Feature Reconstruction and Landmark Loss to Preserve Spatial Consistency in Unaligned Face Images

Avin Maulana[1] **Chastine Fatichah**[1]* **Nanik Suciati**[1]

[1]*Department of Informatics, Institut Teknologi Sepuluh Nopember, Indonesia*
* Corresponding author's Email: chastine@if.its.ac.id

Abstract: Facial inpainting is a process to reconstruct some missing or damaged pixels in the facial image. The reconstructed pixels should still be realistic, so the observer could not differentiate between the reconstructed pixels and the original one. However, there are a few problems that may arise when the inpainting algorithm has been done. There was an inconsistency between adjacent pixels when done on an unaligned face image, which caused a failure to reconstruct. We propose an improvement method in facial inpainting using Generative Adversarial Network (GAN) with additional loss using pre-trained network VGG-Net and face landmark. The feature reconstruction loss will help to preserve deep-feature on an image, while the landmark will increase the result's perceptual quality. The training process has been done using a curriculum learning scenario. Qualitative results show that our inpainting method can reconstruct the missing area on unaligned face images. From the quantitative results, our proposed method achieves the average score of 21.528 and 0.665, while the maximum score of 29.922 and 0.908 on PSNR (Peak Signal to Noise Ratio) and SSIM (Structure Similarity Index Measure) metrics, respectively.

Keywords: Facial inpainting, Feature reconstruction loss, Generative adversarial network, Unaligned face, Spatial correlation.

1. Introduction

Image inpainting is one of the problems in the image domain. Inpainting is a process to reconstruct missing areas on an image such that the reconstructed area remains visually consistent with other areas. The overall image should still look realistic [1]. Image inpainting can also be used to restore damaged areas or to remove some unwanted objects in pictures, like logos [2]. With the development of technology and data availability, inpainting can be done using neural network.

Tanaka [3] proposed an inpainting method combining patch-based inpainting algorithms with CNN. Tanaka used CNN to classify damaged regions automatically; then, the inpainting process is carried out using a patch-based method. This algorithm will generate good results if the inpainting process is carried out to reconstruct a region with many similarities with the surrounding region, such as the background of the sea, or the sky. Problems may arise when the inpainting process is carried out in some areas with particular features, such as eyes, mouth, or nose in human face images. Therefore, another approach is needed to do inpainting, so the reconstructed images remains consistent and realistic. The problem of inpainting on the human face is a particular type of inpainting, namely facial inpainting or face completion.

Variational Auto-Encoder (VAE) [4] and Generative Adversarial Network (GAN) [5] are some networks that can be used to do inpainting. GAN was first proposed by Goodfellow [5], using generative (G) and discriminative (D) models with two-player minimax games principle. Yijun [6] proposed a method using GAN to perform inpainting, namely Generative Face Completion (GFC). Yijun [6] utilizes the semantic face segmentation to improve the quality of facial inpainting results. The results of

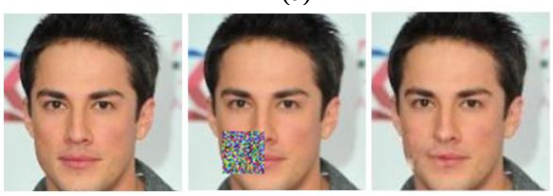
facial segmentation are used as a guide to reconstruct areas with specific characteristics, such as eyes, nose, or mouth. Yijun [6] also uses two types of discriminators; local discriminator and global discriminator. The local discriminator is used only for the missing region, while the global discriminator is used for the whole image. It is expected to get a detailed inpainting image, but still realistic.

However, problems may arise when GFC method is applied to an unaligned face image, where the face orientation is not perpendicular to the horizontal axis, as shown in Figure. 1 (a). The inpainting result would be unrealistic because of its difference between the reconstructed and original regions. Inpainting results may also show different colours between the generated region and its adjacent original regions. For instance, when the missing regions were half of the lips area, the inpainting results may show spatial inconsistency between its adjacent pixels, as shown in Figure. 1 (b).

Deep-feature information from images can be used to maintain spatial consistency, shown by better and consistent perceptual quality, proven in Hou's research [7, 8]. One strategy undertaken to sustain deep-features in the image is done using a pre-trained network, such as VGG-Net. But in the previous researches [7, 8], the use of VGG-Net pre-trained networks to maintain spatial consistency is used in the process to reconstruct entire image from random vector input, not a special case for inpainting problems. Still, the use of adversarial concept along with variational auto encoder can further improve the quality of the generator result, as shown in [8].

Another GAN-based method was proposed by Yu [9] using refinement mechanism to reconstruct the missing regions. However, even though it relied to information around the missing region, the spatial information has been lost due to feature warping, which is needed to preserve the spatial consistency and obtain semantic representation of the input image.

(a)



(b)

Figure. 1 Inpainting problem in GFC [6] method

Haofu [1] also proposed a method for inpainting using GAN with modification in the discriminator and additional landmark loss. The proposed method successfully carried out the inpainting process with better results than the GFC method. It shows that the new concept of discriminator proposed in [1] allows the inpainting process to be carried out even though the missing region in the image is random and scattered like salt and pepper noise. GAN-based inpainting method is also done in [10]. In this study, the concept of Recurrent GAN (RGAN) is used with two discriminators, as proposed in the GFC method [6]. The results of inpainting with RGAN show an image with good perceptual quality and detail thanks to the use of the recurrent GAN concept, which allows the division of parameters between networks and the use of two discriminators. But both research [1] and [10] did not discuss specific conditions when the input image is an unaligned face image, nor the spatial inconsistency problem when inpainting has been done.

Li [11] uses a different approach to do facial inpainting. As stated in [11], reflectional symmetry in face is a prominent property of face image and benefits face recognition and consistency modelling. By assuming a face should be symmetrical reflected over half-face, facial inpainting can be done using illumination-reweighted warping or generative reconstruction subnet based on CNN as Li proposed [11]. However, the proposed method may fail to reconstruct missing region when the face is not in symmetrical pose, such as shown in Figure. 1 (a). The face input image is looking slightly to the right. In this condition, the reflection symmetrical properties can’t be used to reconstruct the missing region.

Based on research that has been done before, two problems arise when facial inpainting is done using GAN. The first problem, the inpainting results are not realistic when the input images are unaligned face images. The second problem, the inpainting result shows inconsistency with its surrounding pixels colours. These problems may be defined as a spatial correlation/consistency problem.

To overcome this problem, we propose an improvement method in facial inpainting using GAN with additional loss that are the feature reconstruction loss based on pre-trained VGG-Net network, and landmark loss based on face landmark network. Our landmark network is based on Object Contour Detection [12]. VGG-Net will be used to obtain feature reconstruction loss, it is because VGG-Net can preserve deep-features within an image, as proved by Hou's research [8-9]. Thus, VGG-Net will also be used in the inpainting process to preserve its deep-features, which relate to its perceptual quality.

Therefore, facial inpainting can still be performed on unaligned face images, and reconstruction of missing facial areas can be done with preserving spatial consistency. Face landmark information can help to improve the perceptual quality of reconstructed facial images, as Liao [1] proposed. The closest work to ours is [13]. It advances the state of the art [8] by using a pre-trained VGG-Net to improve the inpainting's perceptual quality in unaligned face. However, [13] didn't use additional landmark loss to share with generator. As a result, the quality of the result is limited only by spatial feature from VGG-Net which further limits the final performance of the generator.

Our proposed method uses GAN with two types of discriminator: local and global discriminator, curriculum learning [14] strategies by gradually extending the defined losses. In the first stage of training, we train the network by using two defined loss, which comes from the generator itself and features reconstruction loss. In the second stage, the generator is trained with additional losses from the local and global discriminator. As the last stage, face landmark loss is incorporated into the objective function, and the training process continues until the network achieves the optimal visual result. The landmark loss aims to synthesize the better inpainting result with spatial consistency. The dataset used in training process is CelebA [15]. Then, we compare our proposed method results using Peak Signal to Noise Ratio (PSNR) and Structure Similarity Index Measure (SSIM) [16] metric with the previous method. We also show our visual results for qualitative measurement.

This organization of this paper is arranged as follows. Section 2 discusses the basic of generative adversarial network and its application in facial inpainting. Section 3 presents the materials and method for the proposed method. Section 4 describes the experiments and compares some existing methods with our proposed method. Finally, the conclusion and future work of this research are presented in Section 5

2. Literature review

2.1 Generative adversarial network (GAN)

Ian Goodfellow [5] first proposed the Generative Adversarial Network (GAN). GAN architecture consists of two models, namely the generative (G) and discriminative (D) models. Both models will mutually conduct training processes with adversarial concept. The generative model, G, is responsible for synthesizing x data from random input z. Then, the discriminator model, D, becomes adversarial of G, in charge of determining the data generated by the G model is original data or synthesis data. The D model used in GAN is a network to carry out the classification process, so the result of $D\ (x)$ is class probabilities.

This adversarial principle resembles the concept of a two-player mini-max game. The training process in D is carried out with the aim of model D being able to classify original data or synthesis data, which means that the success rate of model D is high. While the G model aims to produce synthesis data $x = G(z)$ such that the D model has the minimum success rate and fail to differentiate between original and synthesis data. Implicitly, G will define a probability distribution p_g, because the distribution of synthesis data $G(z)$ is obtained by $z \sim p\ (z)$. Therefore, the objective function of GAN can be stated as in the Eq. (1)

$$\min_G \max_D V(D,G)\ \ A + B,$$
$$\text{where } A = \mathbb{E}_{x\sim p_{data}(x)}[\log D(x)]$$
$$B = \mathbb{E}_{z\sim p_z(z)}\left[\log\left(1 - D\left(G(z)\right)\right)\right] \quad (1)$$

where p_{data} represents the original data distribution, $x \sim p_{data}(x)$ states that the sample x follows the $p_{data}(x)$ distribution. The objective of GAN training process is that the p_g distribution is similar to p_{data} so that D has difficulty in differentiate a sample following the p_{data} or p_g distribution.

2.2 Generative face completion

Generative Face Completion (GFC) was first proposed by Yijun [6]. GFC architecture consist of 3 main networks: 1) generator, 2) two types of discriminator, 3) semantic parsing network. Overall loss function of Generative Face Completion formulated on Eq. (2)

$$L\ = L_r + \lambda_1 L_{\alpha 1} + \lambda_2 L_{\alpha 2} + \lambda_3 L_P, \quad (2)$$

where $\lambda_1, \lambda_2, \lambda_3$ are the weights to balance the effects of different losses. L_r denote the reconstruction loss, which is only euclidean distance between original image and inpainting result on RGB channel. $L_{\alpha 1}$ and $L_{\alpha 2}$ are local and global discriminator loss, respectively. L_P is simple pixel-wise softmax loss from semantic segmentation result between original and inpainting result.

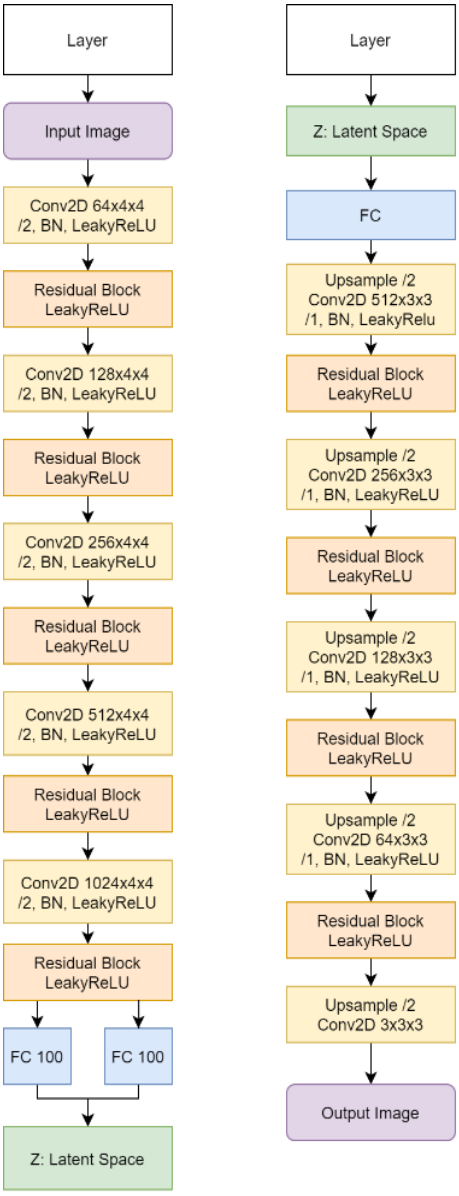


Figure. 2 Generator network. Consists of encoder, decoder, with residual block

3. Materials and method

3.1 Dataset

The data used in this study are secondary data from CelebFaces Attributes Dataset (CelebA), taken from the research of Ziwei [15]. The CelebA dataset consists of 10,177 identities, with each identity having around 20 images. Thus, the total image from CelebA is 202,599 images.

The pre-processing stage is carried out by performing cropping and resizing operations on CelebA data. The cropping is done without any alignment, and the cropping area should still contain eyes, nose, and mouth. Then, the images are resized to 128×128 pixel. The masking size for training is set to 64×64, following GFC method. The choice of this size is to guarantee the mask will cover at least one specific feature of human faces, like an eye. The masking is a random noise pixel and placed randomly in the image.

3.2 Generator

The Generator, denoted by G, is based on VAE [4] network. The generator network consisted of two main models, encoder and decoder, following the architecture of DFC-VAE models [7-11]. The encoder consists of 5 convolution layers and two fully connected. Each convolution layer is 2-dimension convolution with 4×4 kernel size, with stride 2. The selection of stride 2 aims to do down-sampling without using deterministic spatial functions such as maxpool. Then, through the Batch-Normalization (BN) process and the LeakyReLU activation function. Each convolution layer is followed by a residual block [17]. Each residual block consisted of a convolution operation with 3×3 filter size, BN, and LeakyReLU activation function, followed by convolution with 3×3 filter size, as shown in Figure. 3. No downsampling operation is needed in the residual block, so the stride size used is 1. LeakyReLU activation function is applied to the residual block's output.

The fully-connected layer will map the inputs image array into z_μ and z_σ values before becoming latent z variable, where z_μ denote mean vector and z_σ denote standard deviation vector. The z latent variable simply $z = z_\mu + z_\sigma \epsilon$, where ϵ is an auxiliary noise, and $\epsilon \sim \mathcal{N}(0,1)$ [4]. The decoder section is symmetrical with the encoder. Consists of 5 convolution layers with kernel size 3×3, and stride size 1. Before each convolution on decoder, upsampling is done using nearest neighbor method with a scale 2. The input image is in range [-1,1], $tanh$ activation function activations is used in

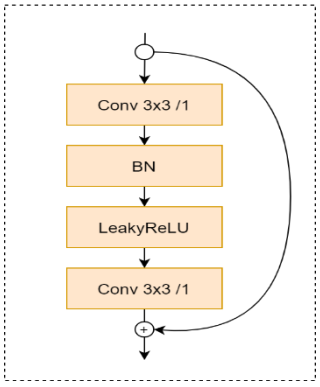


Figure. 3 Residual block used in network

network G to keep the output result in range [-1,1]. The illustration of generator model is shown in Figure. 2.

3.3 Discriminator

The discriminator architecture is identical to the encoder. The difference lies in the last layer as the output. Note that on this proposed method, discriminator network does not use logarithmic activation functions such as $sigmoid$, or hyperbolic functions such as $tanh$. The output layer is only consisted with LeakyReLU and $1 \times 4 \times 4$ convolutions to achieve 1-d result.

The proposed method uses two types of discriminator: local and global discriminator. The illustration of discriminator models shown in Figure. 4. The difference between local and global discriminator lies in the number of layers used, and the input size of the discriminator. There are five convolution layers to do downsampling and one last convolution layer to produce scalar output on global discriminator. In comparison, local discriminator only has 4 convolution layers to do downsampling and 1 final convolution layer to produce scalar output. It is because global discriminator input is $128 \times 128 \times 3$, while local discriminator input is masking size, which is $64 \times 64 \times 3$.

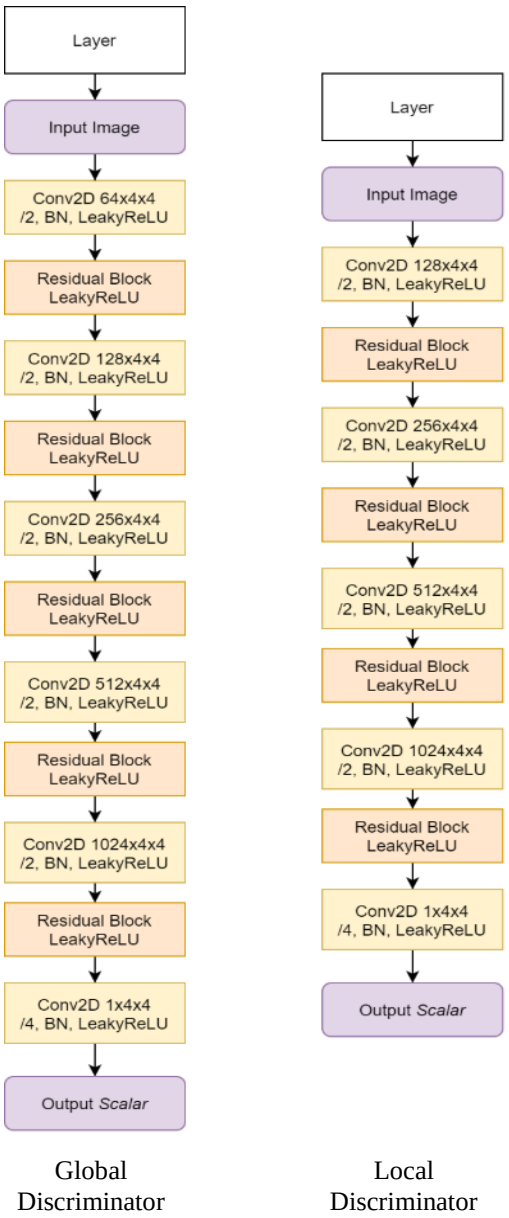


Figure. 4 Two types of discriminator network: global and local discriminator, with residual block

3.4 Training strategies

In the first stage of training, the generator network is trained using only two defined loss, K-L Divergence Loss ($\mathcal{L}_{KL}$), and Feature Reconstruction Loss $(\mathcal{L}_f)$. K-L Divergence Loss is formulated in Eq. (3), where n is latent vector (z) length.

$$\mathcal{L}_{KL} = \frac{1}{2}\left[\sum_{i=1}^{n} z_{\mu,i}^2 + \sum_{i=1}^{n} z_{\sigma,i}^2 - \sum_{i=1}^{n} \log\left(z_{\sigma,i}^2 + 1\right)\right] \quad (3)$$

In the proposed method, the feature reconstruction loss is a loss obtained from euclidean distance between the original image and inpainting result on the VGG-Net feature domain, not in RGB channel like usual reconstruction loss.

VGG-Net pre-trained network defined as a mapping of the original image I into the VGG-Net feature domain, denoted by $\psi_{(i,j)}(I)$. Thus, feature reconstruction loss $(\mathcal{L}_f)$ has the equation presented in Eq. (4) which is the sum of loss for each layer in VGG-Net feature domain. In this study, we limited the layers used in VGG-Net feature domain to three first layers. Eq. (5) states the loss value for feature map at layer l, denoted by $\mathcal{L}_l$, where C_l, W_l, H_l, denote the number of channels, width, and height in the VGG-Net feature domain on the l^{th} layer, respectively.

$$\mathcal{L}_f = \sum_{l=1}^{L} \frac{100}{C_l^2} \mathcal{L}_l \quad (4)$$

$$\mathcal{L}_l = \frac{1}{2C_l W_l H_l} \sum_{c=1}^{C_l} \sum_{i=1}^{W_l} \sum_{j=1}^{H_l} \left(\psi_{c,i,j}(I) - \psi_{c,i,j}(\hat{I})\right)^2 \quad (5)$$

In the second stage, the generator is trained with additional losses from the local and global discriminator. In this study, the discriminator used was not a standard GAN discriminator as in the GFC [1], but a discriminator critic as in Wasserstein GAN (WGAN) [18]. The choice of WGAN's type is because standard GAN difficult to achieve stability, and the original GAN loss has poor generalization ability [19], so there are several improvements to overcome this problem, such as WGAN.

The losses used to update local and global discriminator weight stated in Eq. (6) and Eq. (7) respectively. $D\mathcal{L}_{ld}$ is used to update local discriminator, while $D\mathcal{L}_{gd}$ is used to update global discriminator. Where $D_{type}(x)$ denote the output of discriminator with respect to input image array x, where $type \in \{ld, gd\}$. D_{ld} denote local discriminator, while D_{gd} denote global discriminator. The m value is the batch size.

$$D\mathcal{L}_{ld} = \frac{1}{m} \sum_{i=1}^{m} \left[D_{ld}\left(x^{(i)}\right) - D_{ld}(G(z^{(i)})) \right] \quad (6)$$

$$D\mathcal{L}_{gd} = \frac{1}{m} \sum_{i=1}^{m} \left[D_{gd}\left(x^{(i)}\right) - D_{gd}(G(z^{(i)})) \right] \quad (7)$$

Then, the discriminator losses to update the generator by using the adversarial concept, which is loss from local discriminator ($G\mathcal{L}_{ld}$) and loss from global discriminator $\left(G\mathcal{L}_{gd}\right)$, stated in Eq. (8) and Eq. (9), respectively.

$$G\mathcal{L}_{ld} = \frac{1}{m} \sum_{i=1}^{m} \left[D_{ld}(G(z^{(i)})) \right] \quad (8)$$

$$G\mathcal{L}_{gd} = \frac{1}{m} \sum_{i=1}^{m} \left[D_{gd}(G(z^{(i)})) \right] \quad (9)$$

In this second stage, the training process of discriminator and generator is done simultaneously with ratio 5:1. As it is stated in [20], there are some tricks that may be done to stabilize GAN, such as balancing between the generator and the discriminator update. Based on our experiments and as recommended in [18], we choose ratio 5:1 to update the discriminator. Following the WGAN algorithm, before the gradient from the loss is passed to the entire network for weight update, the clipping gradient operation is carried out with a clipping limit of $[-c, c]$, where a value of $c = 0.01$ is given, as it is used in [8] to clip the gradient value between -0.01 and 0.01 to stabilize WGAN. When batch normalization is off, the discriminator's gradients might explode when c increases from 0.01, such as 0.1.

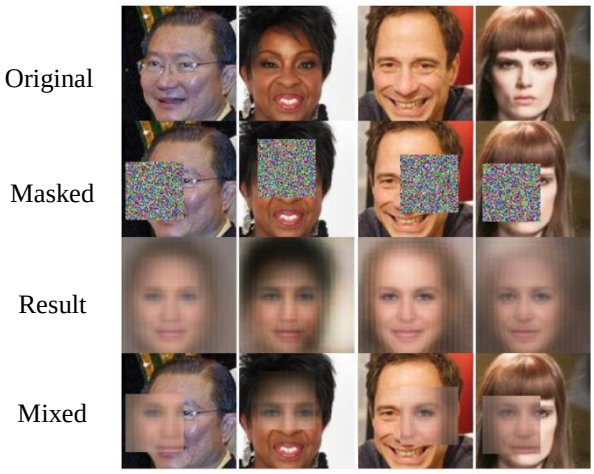


Figure. 5 Inpainting result at the beginning of training process

In the last stage, the landmark loss, namely $(L_h)$is incorporated to the total loss. The addition of this loss landmark aims to improve the inpainting result with better perceptual quality so the image produced by the network looks more realistic. The landmark loss $(\mathcal{L}_{lm})$ is simply pixel-wise logistic loss between the landmark result from original image (as label) and the inpainting result (as predicted class).

Thus, the total loss function of the network can be formulated as Eq. (10), where λ_i, $i \in [1, 5]$ denote the weight for each loss to regularize the effect of the loss.

$$\mathcal{L} = \lambda_1 \mathcal{L}_{KL} + \lambda_2 \mathcal{L}_f + \left(\lambda_3 \mathcal{L}_{ld} + \lambda_4 \mathcal{L}_{gd}\right) + \lambda_5 \mathcal{L}_{lm} \quad (10)$$

The optimization methods used to train the generator and discriminator are ADAM [21] and RMSProp method, respectively. The masking given to the image is random normal noise with a constant size, 64×64 pixels. The limit of the first stage and second stage are set to 15000 and 25000 steps, and the batch size used in training is 16.

The batch size may very between each research. As in our experiment, we found that the choosing of batch size = 16 is already enough to get better result for the trade-off its training time. Choosing smaller batch size will affect to longer training time, faster to converge, fits to limited GPU, but tend to noisy result. The limit for each stage is chosen based on the convergency of each stage.

The generator learning rate is set to 0.0001, while the discriminator learning rate is 0.0005. Based on our experiment, we started by using higher learning rate, such as 0.01. But the training process is converged to zero or infinity on beginning step. Then

by gradually decrease the learning rate, we found that 0.0001 and 0.0005 is enough to achieve better result in our proposed method without trapped in zero or infinity.

4. Results and analysis

4.1 Training results

The first stage of network training uses two types of loss: K-L divergence loss ($\mathcal{L}_{KL}$) and Feature Reconstruction Loss ($\mathcal{L}_f$). This stage is carried out at 15000 steps. The result of the network at the beginning of training process is presented in Fig. 5. The first row on Fig. 5 is the original image (ground truth) before masking, the second row shows the original image after being given a masking with a size of 64 × 64 pixels. The choice of masking size is intended to ensure that the masking covered at least one specific feature of face, such as the eyes, mouth, or nose. The third row shows the output from the network, which is the inpainting result. The fourth row is a combination of ground truth image with the output of the network.

The result when the first stage training has been passed is shown on Fig. 6. Previous research, GFC [6] did not use KL-Divergence loss nor VGG-Net and only applied the euclidean distance between the input image and the output image in the RGB channel. The network generator results in the GFC method with the first stage of loss are shown in Fig. 7. At this stage, it can be seen qualitatively from the result that the addition of the KL-Divergence loss along with Feature Reconstruction Loss using VGG-Net can help to produce better perceptual quality inpainting results.

The feature reconstruction loss manages to capture more detailed and specific face shape patterns. However, although the results by the generator are good enough in terms of perceptual, the colour difference is still visible between the part of the synthesized image with its surroundings, and the resulting detail is still not good enough. This will be corrected with the use of subsequent losses on next training stage.

The first loss added to the network is adversarial loss: local discriminator ($\mathcal{L}_{ld}$) and global discriminator ($\mathcal{L}_{gd}$). Optimization performed on the type of GAN network is a mini-max concept between discriminator and generator. To achieve the desired equilibrium, the discriminator and generator must have balanced power. When one sub-network part of GAN is too dominating/superior, the desired equilibrium condition cannot be achieved. The

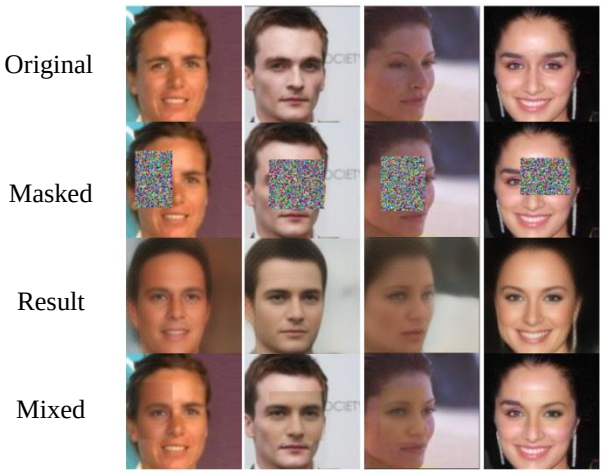
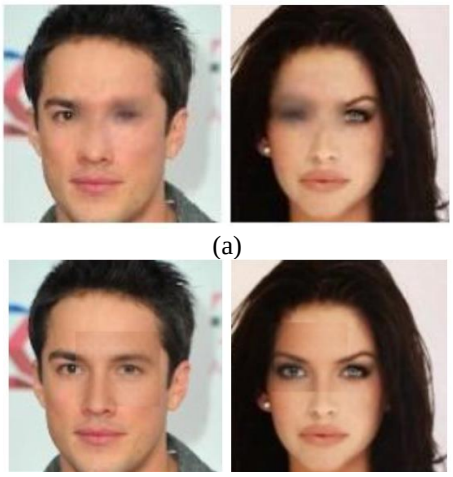


Figure. 6 Inpainting result at step 15000, using only $\mathcal{L}_{KL}$ and $\mathcal{L}_f$



(a)

(b)

Figure. 7 : (a) Network generator results from the first stage training in GFC [6] method. Without the feature reconstruction loss from VGG-Net or KL-Divergence Loss and (b) first stage training results from our proposed method

equilibrium condition between the generator and discriminator can be achieved more easily by using WGAN. The strategy taken to help stabilize WGAN in this proposed method is enough by adding noise to the input of the discriminator and using -10 and 10 as the label for fake and original images. The generator results after 25.000 steps with additional two losses from local and global discriminator are shown in Figure. 8.

Then, as the last stage of training, landmark loss is incorporated to the network ($\mathcal{L}_{lm}$). The qualitative results from the generator after the training step has reached 40,000 are shown in Figure. 9. It can be seen from the visual result that the network able to produce more realistic and better perceptual quality results than the previous stage.

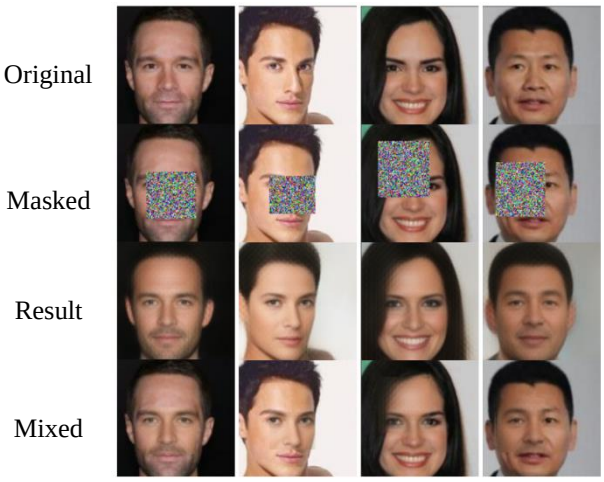


Figure. 8 Inpainting result at step 25000, using $\mathcal{L}_{KL}$, $\mathcal{L}_{f}$, $\mathcal{L}_{ld}$, and $\mathcal{L}_{gd}$

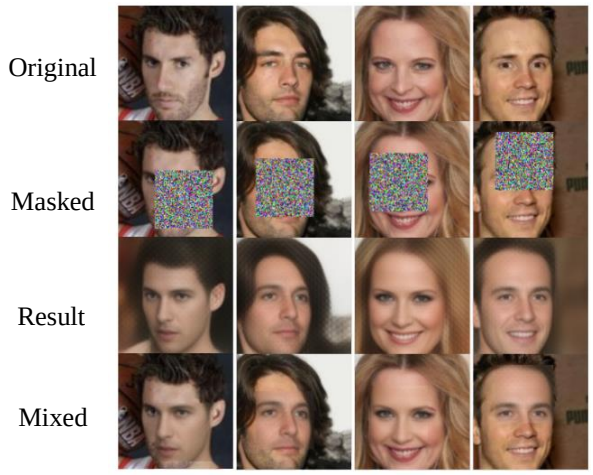
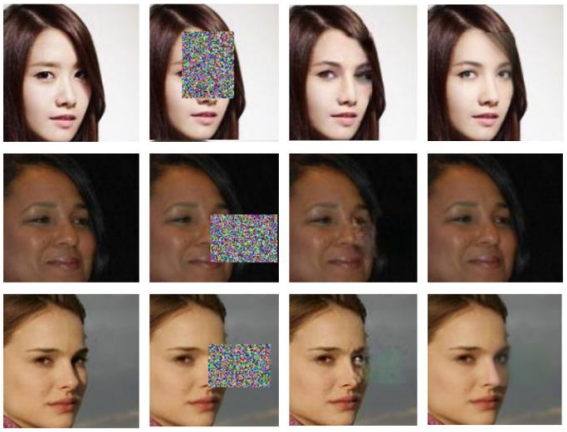


Figure. 9 Inpainting result at step 40000, using $\mathcal{L}_{KL}$, $\mathcal{L}_{f}$. $\mathcal{L}_{ld}$, $\mathcal{L}_{gd}$, and $\mathcal{L}_{lm}$

4.2 Testing results

We provide inpainting results on specific scenario where the problem arises in previous research: unaligned faces input, and some areas which needs to preserve spatial consistency between its adjacent pixels, as shown in Figure. 1. We provide our inpainting results compare to previous method (GFC) [6] in unaligned face inputs in Figure. 10. The first column (a) shows the original unaligned face image before masking. The second column (b)shows the image that has been masked. The third column (c) shows GFC results, and the fourth column (d) shows our inpainting results. It can be seen from the figure provided that our method can perform the reconstruction process even the input images are unaligned face images, better than previous method.

The previous method (GFC) [6] also tend to reconstruct a slightly different colours with its adjacent pixel, and might fail to reconstruct realistic and synchronous pixels to its adjacent, as shown in



(a) (b) (c) (d)

Figure. 10 Inpainting results on unaligned face: (a) original input images, (b) masked images, (c) GFC [6] results, and (d) our proposed method results

(a) (b) (c) (d)

Figure. 11 Inpainting results in preserving spatial consistency: (a) original input images, (b) masked images, (c) GFC [6] results, and (d) our proposed inpainting method results

Figure. 11 (c). We perform our method in similar condition where the missing area was half of the lips, or part of eyeglasses. Qualitatively by observation, it can be seen from Figure. 11 (d) that our inpainting method results looks more synchronous with its surrounding colours and realistic with better perceptual quality. It shows that the addition of feature reconstruction loss and landmark loss can improve the perceptual quality of the resulting image and help to preserve spatial consistency.

4.3 Performance metrics evaluation

After the training process has been done, the generator is tested using test data from CelebA. Evaluation metrics used for evaluations are Structure Similarity Index Measure (SSIM) [16] and Peak Signal to Noise Ratio (PSNR) compared with previous methods such as CE [22], and GFC [6]. PSNR gives the similarity score at pixel-level while SSIM evaluates the similarity at perceptual or structure level. Both of PSNR and SSIM, higher

Table 1. Quantitative result comparison

Metric	Quantitative	Proposed Method	[13]	[9]	[22]	[6]	[1]	[10]
SSIM	Minimum	0.068	0.070	-	0.818	0.84	0.857	0.899
	Maximum	0.908	0.890					
	Average	0.665	0.651					
	Standard deviation	0.083	0.089					
PSNR	Minimum	11.657	11.668	18.91	19.3	20.2	20.71	23.96
	Maximum	29.922	27.996					
	Average	21.528	21.067					
	Standard deviation	1.907	1.909					

value of PSNR or SSIM means better result. The testing process is done using random mask size in range $[32, 64]$, randomly placed on input image. The PSNR and SSIM result obtained from the test are shown in Table 1.

It can be seen from Table 1 that our proposed method are able to achieve better average PSNR result, compared to previous research [1, 6, 9, 12, 21]. However, our proposed method still got smaller average PSNR than [10], but greater in PSNR maximum which is 29.922. Nevertheless, our proposed method got smaller average on SSIM, which is only 66.5%, compared to 81.8%, 84.1%, 85.7% and 89.9%, while [9] didn't use SSIM as its quantitative metrics. Based on our testing experiments, we get maximum SSIM value on 90.8% which means highest structure similarity we get by using our proposed method is 90.8% similar to its original image on structure level. But quantitative result could not guarantee the visual result have better perceptual quality or more realistic. It means that along with higher PSNR or SSIM, higher similarity to its original image in pixel level or structure level [6].

5. Conclusion

Inpainting method on face images can be done using Generative Adversarial Network (GAN) with additional loss from feature reconstruction loss using VGG-Net pre-trained network and landmark loss from facial landmark network to improve the perceptual quality result. Asides from overcoming the problem when inpainting inputs are unaligned face images, these two additional losses can help maintain the spatial consistency of the output image, as shown with better perceptual quality and more realistic inpainting results.

Based on random masking scenarios conducted while testing using data test, our proposed inpainting method with additional losses makes it possible to get better results. Our proposed method achieves the average score of 21.528 and 0.665, while the maximum score of 29.922 and 0.908 on PSNR and SSIM metrics, respectively. From the qualitative results, our method still able to reconstruct the missing areas even though the input face images are unaligned face images, and preserving its spatial consistency, showed with more synchronous colours and realistic inpainting results.

Conflicts of Interest

The authors declare no conflict of interest

Author Contributions

As the first author, Avin Maulana contributed to the formation of the paper, including the formulation of methods, the implementation of methods and the conduct of experiments. Chastine Fatichah supervised in formulation of the proposed method and article preparation. Nanik Suciati supervised in article preparation and assured the process while the research is conducted.

Acknowledgments

The authors would like to sincerely thank Institut Teknologi Sepuluh Nopember for supporting the research.

References

[1] H. Liao, G. Funka-Lea, Z. Yefeng, L. Jiebo, and S. K. Zhou, "Face Completion with Semantic Knowledge and Collaborative Adversarial Learning", In: *Proc. of Asian Conf. on Computer Vision*, pp. 382–397, 2018, doi: 10.1007/978-3-030-20887-5_24.

[2] M. A. Qureshi, M. Deriche, A. Beghdadi, and A. Amin, "A critical survey of state-of-the-art

image inpainting quality assessment metrics", *J. Vis. Commun. Image Represent.*, Vol. 49, pp. 177–191, 2017, doi: 10.1016/j.jvcir.2017.09.006.

[3] T. Tanaka, N. Kawai, Y. Nakashima, T. Sato, and N. Yokoya, "Iterative applications of image completion with CNN-based failure detection", *J. Vis. Commun. Image Represent.*, Vol. 55, No. May, pp. 56–66, 2018, doi: 10.1016/j.jvcir.2018.05.015.

[4] D. P. Kingma and M. Welling, "Auto-encoding variational bayes", In: *Proc. of 2nd International Conf. on Learning Representations, ICLR 2014 - Conf. Track Proc.*, No. Ml, pp. 1–14, 2014.

[5] I. J. Goodfellow *et al.*, "Generative Adversarial Nets," *Adv. Neural Inf. Process. Syst.*, Vol. 27, pp. 4089–4099, 2014.

[6] L. Yijun, L. Sifei, Y. Jimei, and Y. Ming-Hsuan, "Generative Face Completion", In: *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 5892–5900, 2017, doi: 10.1109/CVPR.2017.624.

[7] X. Hou, L. Shen, K. Sun, and G. Qiu, "Deep Feature Consistent Variational Autoencoder", In: *Proc. of IEEE Winter Conf. on Applications of Computer Vision*, pp. 1133–1141, 2017, doi: 10.1109/WACV.2017.131.

[8] X. Hou, K. Sun, L. Shen, and G. Qiu, "Improving variational autoencoder with deep feature consistent and generative adversarial training", *Neurocomputing*, Vol. 341, pp. 183–194, 2019, doi: 10.1016/j.neucom.2019.03.013.

[9] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, "Generative Image Inpainting with Contextual Attention", In: *Proc. of the IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, pp. 5505–5514, 2018, doi: 10.1109/CVPR.2018.00577.

[10] Q. Wang, H. Fan, G. Sun, W. Ren, and Y. Tang, "Recurrent Generative Adversarial Network for Face Completion", *IEEE Trans. Multimed.*, pp. 1–1, 2020, doi: 10.1109/tmm.2020.2978633.

[11] X. Li *et al.*, "Learning Symmetry Consistent Deep CNNs for Face Completion", *IEEE Trans. Image Process.*, Vol. 29, 2020, doi: 10.1109/TIP.2020.3005241.

[12] Y. Jimei, B. Price, S. Cohen, H. Lee, and Y. Ming-Hsuan, "Object contour detection with a fully convolutional encoder-decoder network", In: *Proc. of IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, Vol. 2016-Decem, pp. 193–202, 2016, doi: 10.1109/CVPR.2016.28.

[13] A. Maulana, C. Fatichah, and N. Suciati, "Facial inpainting pada citra wajah unaligned menggunakan generative adversarial network dengan feature reconstruction loss", *JUTI J. Ilm. Teknol. Informasi.*, pp. 1–9, 2020.

[14] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, "Curriculum Learning", In: *Proc. of International Conf. on Machine Learning*, pp. 41–48, 2009, doi: 10.1017/S1047951100000925.

[15] L. Ziwei, L. Ping, W. Xiaogang, and T. Xiaoou, "Deep learning face attributes in the wild", In: *Proc. of IEEE International Conf. on Computer Vision*, pp. 3730–3738, 2015, doi: 10.1109/ICCV.2015.425.

[16] W. Zhou, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image Quality Assessment: From Error Visibility to Structural Similarity", *IEEE Trans. Image Process.*, Vol. 13, No. 4, p. 600, 2004.

[17] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition", In: *Proc. of IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Vol. 2016-Decem, pp. 770–778, 2016, doi: 10.1109/CVPR.2016.90.

[18] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein GAN", 2017. [Online]. Available: http://arxiv.org/abs/1701.07875.

[19] H. Thanh-Tung, S. Venkatesh, and T. Tran, "Improving generalization and stability of generative adversarial networks", In: *Proc. of 7th International Conf. on Learning Representations, ICLR 2019*, pp. 1–18, 2019.

[20] I. Goodfellow, "NIPS 2016 Tutorial: Generative Adversarial Networks", 2016, [Online]. Available: http://www.iangoodfellow.com/slides/2016-12-04-NIPS.pdf.

[21] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization", In: *Proc. of 3rd International Conf. on Learning Representations, ICLR 2015 - Conf. Track Proc.*, pp. 1–15, 2015.

[22] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros, "Context Encoders: Feature Learning by Inpainting", In: *Proc. of IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 2536–2544, 2016, doi: 10.1109/CVPR.2016.278.