Low Resolution Face Recognition Using Generative Adversarial Network (GAN)

Howard Martin (✉ howard.martin@mail.ugm.ac.id)

Binus University - Anggrek Campus <https://orcid.org/0000-0002-1171-2850>

Suharjito -

Binus University

Research

Keywords: Super resolution, Low Resolution, Face Recognition, Generative Adversarial Network

Posted Date: December 15th, 2020

DOI: <https://doi.org/10.21203/rs.3.rs-36971/v2>

License: This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Low Resolution Face Recognition Using Generative Adversarial Network (GAN)

Howard Martin¹ and Suharjito^{1,2*}

*Correspondence: suharjito@binus.edu

¹ Department Computer Science,
Binus Online Learning,
Bina Nusantara University,
Jakarta, 11480,
Indonesia

Abstract

Face recognition has a lot of use on smartphone authentication, finding people, etc. Nowadays, face recognition with a constrained environment has achieved very good performance on accuracy. However, the accuracy of existing face recognition methods will gradually decrease when using a dataset with an unconstrained environment. Face image with an unconstrained environment is usually taken from a surveillance camera. In general, surveillance cameras will be placed on the corner of a room or even on the street. So, the image resolution will be low. Low-resolution image will cause the face very hard to be recognized and the accuracy will eventually decrease. That is the main reason why increasing the accuracy of the Low-Resolution Face Recognition (LRFR) problem is still challenging. This research aimed to solve the Low-Resolution Face Recognition (LRFR) problem. The datasets are YouTube Faces Database (YTF) and Labelled Faces in The Wild (LFW). In this research, face image resolution would be decreased using bicubic linear and became the low-resolution image data. Then super resolution methods as the preprocessing step would increase the image resolution. Super resolution methods used in this research are Super resolution GAN (SRGAN) [1] and Enhanced Super resolution GAN (ESRGAN) [2]. These methods would be compared to reach a better accuracy on solving LRFR problem. After increased the image resolution, the image would be recognized using FaceNet. This research concluded that using super resolution as the preprocessing step for LRFR problem has achieved a higher accuracy compared to [3]. The highest accuracy achieved by using ESRGAN as the preprocessing and FaceNet for face recognition with accuracy of 98.96 % and Validation rate 96.757 %.

Keywords: Super resolution; Low Resolution; Face Recognition; Generative Adversarial Network;

Introduction

Face recognition is a system that has been designed to recognize faces based on the database of known faces [4]. Face recognition system has a good improvement in recent years, especially when using a deep learning architecture. One of the state-of-the-art methods based on deep learning has been achieved by [5] with over 97% accuracy on public face datasets such as LFW [6]. Although this algorithm has a good performance on a good condition of the face and pose, these faces generally are high resolution. These designed that has a good performance on High Resolution (HR) image cannot be directly used on Low-Resolution (LR) image. The surveillance camera has been widely used on public places such as on the streets, office, even in a room (usually placed at the corner). It creates a challenge on face recognition system, where the detected faces would be low in resolution. This research is motivated to solve this challenge.

This work is based on prior work published in [3], but has added pre-processing step using super resolution and compared to reach the highest accuracy for Low-Resolution Face Recognition (LRFR). [3] had researched LRFR problems without super resolution method and directly used FaceNet to recognize LR images. This work used YTF [7] and LFW [6] dataset. The image resolution would be decreased using bicubic interpolation as the low-resolution image dataset. Then super resolution

used as preprocessing step to increase the resolution. Super resolution methods SRGAN [1] and ESRGAN [2] used to compared for reaching the best accuracy of LRFR. Those methods were used because those methods had reached a good result in Peak Signal-to-Noise Ratio (PSNR) and won the PIRM-SR Challenge. There are several super resolution methods like Bicubic Interpolation [8], Convolutional Neural Network (CNN), Generative Adversarial Network (GAN) [9], etc. Now GAN is the state-of-the-art of super resolution compared with other methods [1]. After increased the image resolution, FaceNet will be used for face recognition and compared the accuracy with [3].

This research will be divided into several section as in introduction, related works, proposed method, experiment result and discussion section. Introduction section will discuss about the background problem of LRFR. Related works section will discuss about literature study about recent works on LRFR and the differences with previous works. Proposed method section will discuss all method that had been proposed to increase the accuracy of LRFR. Experimental result and discussion section will present the results of this research.

Related Works

There are three ways to solve LRFR problem [8]. The first method is down sampling HR image to LR image. The second method is increasing the resolution of LR image using an interpolation method or super resolution method [10]. After increasing the LR image resolution, then face recognition methods for HR image can be used to increase the accuracy. The third method is by combining several methods. This research use FaceNet [11] as the face recognition method. It is the same method used in [3]. The accuracy of FaceNet will be compared between FaceNet using SUPER RESOLUTION methods and the accuracy reached by [3].

In 2015, [5] used deep CNN to build a face recognition system. [5] combined several parameter configurations based on [12]. [5] also used triplet loss for learning the face embedding. The datasets used in this research are Labelled Faces In The Wild (LFW) [6] and YouTube Faces Database (YTF) [13] with size 224 x 224 pixel. The accuracy result for LFW dataset is 98.95% and YTF dataset is 97.3%. [5] concluded that the error value would keep getting lower if CNN used more layers. CNN has reached very good results in the face recognition system, especially a face recognition system that has HR images [5], [14], [15]. From there, CNN has been widely used to solved face recognition problems. CNN has been modified like on [8], which used Multi-Resolution CNN. For the pre-processing stage, the bicubic interpolation method is used to decrease the value of cosine similarity.

[16] did some research on low-resolution face recognition (LRFR). They did face recognition from a video using CNN. This research modified the CNN method by adding manifold based track comparison strategy for low-resolution video recognition. The datasets used for training are from FaceScrub [17] and MSRA-CFW (dataset of celebrity faces on the web) [18]. YTF dataset is used for testing. This research has reached 80.3% accuracy on YTF dataset. [7] researched on LRFR using Deep Coupled ResNet. The proposed method consists of one network trunk and two networks branch. Network trunk used for training using three different resolutions on each face image. The network branch is also used for training on the HR and LR images. This training aims to reduce the distance of HR and LR image as small as possible. Coupled Mapping was used to optimize the parameter on the network branch. This research used two datasets from LFW and S.C. Face Database [19]. LFW dataset used for testing with two different resolutions. The 8 x 8 pixels reached 93.6% accuracy and 112 x 96 pixels reached 98.7% accuracy.

[3] had research on face recognition using FaceNet with several image sizes. The dataset that had been used on [3] was LFW. Every image on the dataset had been resized using bicubic linear to 10

x 10, 20 x 20, 40 x 40, 60 x 60, 80 x 80, 100 x 100, dan 160 x 160. Facenet accuracy on 160 x 160-pixel image was 99.2 % and the validation rate 97.1 %. But the accuracy gradually decreased with a smaller image. Face accuracy on 10 x 10 pixel was 58.2 % and validation rate 1.6 %. The size that would be compared on this research is 40 x 40 pixel. Facenet accuracy on 40 x 40 was 98.5 % and validation rate 88.5 %. On FaceNet, validation rate is used to measure the True Accept rate comparing to all data pair of same identities.

Super resolution method is used on LRFR to decrease the distance value between LR and HR image. Peak Signal-to Noise Ratio (PSNR) and Structural Similarity (SSIM) were used in [1], [2] to calculate the distance. There are several methods of super resolution like bicubic interpolation [20], CNN [21], GAN [22][23], etc. GAN architecture uses deep CNN and needed more layers to minimize the error. GAN architecture used Deep Convolutional and was proposed for the first time by Radford in 2015 [24]. Two modified GANs used in this work are SRGAN [1] and ESRGAN [2]. ResNet or skip connection is being used on GAN, so CNN could be used more than 1,000 layers [25], resulting in a very minimum error. These GAN methods had been proposed by [2] and [1]. But those methods had never been applied directly to LRFR problems. Super resolution methods are used to improve the accuracy reached by [3]. The face recognition method that will be used is FaceNet [11]. FaceNet learns to map face images to an efficient Euclidean space to measure the face similarity based on the distance. FaceNet embeddings as feature vectors can be used by standard methods to solve tasks like face recognition, verification, and clustering [26][27]. All previous works are summarized in Table 1.

Table 1 Recall Result Using Confusion Matrix

Author	Case	Pre-Processing	Method	Evaluation Result
[8]	Low-Resolution	super resolution with Bicubic Interpolation	Multi-Resolution CNN (MRCNN)	Pie DB: 96.02% Yale DB: 94.63%
[16]	Low-Resolution		CNN with Manifold-based Track Comparison Strategy.	YTF: 80.3%
[7]	Low-Resolution	Extract discriminative features with ResNet Trunk Network	CNN with Deep Coupled ResNet	8x8 pixel: 93.6% 112x96 pixel: 98.7%
[3]	Low-Resolution	Resized image with Bicubic Linear	Facenet	10 x 10: 58.2 % 40 x 40: 98.5 % 160 x 160: 99.2 %
[2]	Super resolution		ESRGAN	
[1]	Super resolution		SRGAN	

Proposed Method

The proposed method for this research is shown at Figure 1. The first step is collecting the dataset from LFW and YTF. The second step is preparing dataset by cropping the face image using Multi-task Cascaded Convolutional Networks (MTCNN) [28]. The third step is preprocessing using super resolution. The super resolution methods are SRGAN [1] and ESRGAN [2]. The fourth step is face recognition using FaceNet [11]. And the last step is evaluating and comparing the result to get the highest accuracy.

Preparing Datasets

This research will be using two face datasets for the training and evaluation phase. YouTube Faces Database (YTF) [13] as a training phase dataset on super resolution neural network. Labelled Faces In The Wild (LFW) [6] will be used as training on FaceNet neural network and used on an evaluation phase. The first dataset is YTF. It is a dataset that designed especially for solving the face recognition problems in video. This dataset contains 3,425 videos of 1,595 subjects. The average video for every subject is 2.15. For each subject, the neural network will use 10 sample images to train.

The second dataset is LFW. It is a dataset that becomes a standard benchmark for automatic face verification. This dataset contains 13,233 images with 5,749 subjects. It also provides 6,000 pairs of sample data for the evaluation phase. Images in this database also have various challenges like variations of pose, expression and illumination. YTF and LFW dataset will be prepared by cropping and aligning. Every image on the dataset will be checked if any face is detected on the image. If a face is detected, the image will be cropped precisely on the face only so that the background image and other attributes will not affect when the face is recognized. The image face also being aligned to a straight position. MTCNN method would be used for this task. MTCNN would ensure all the images is still detected as a face. The goal of this process is to get the cropped face and align the face. The sample output of MTCNN is shown in Figure 2.

Preprocessing Using Super resolution

Super resolution is a method used to increase the resolution of LR image to HR image. Super resolution is widely used to keep the quality of images when the images are stretched to a bigger size. Super resolution is not the same as upsizing the image. Super resolution ensures that the upsize images don't lose their details and visual appearances. The main goal of super resolution is to find

the missing pixel value in the HR image [29]. Super resolution can be divided into two classes: Dynamic resolution (multiple image super resolution) and static resolution (single image super resolution). The dynamic resolution method assumes that the multiple images are geometric transform and misaligned versions of each other. Combining and evaluate them together will result in more details and fill the missing pixel in HR image. But this method is difficult because the multiple is not always available. So, the static resolution is more applicable and will be used for this research. Many methods can be used for super resolution like bicubic linear, nearest neighbor, generative adversarial network (GAN), etc. But recently, GAN has reached a very good performance on super resolution and become state of the art in super resolution problem.

Super resolution gets the input of cropped and aligned images. Fine tuning super resolution needs Low-Resolution (LR) and Ground Truth (GT) from image D . D can be defined as $D = \{(I_1^{LR}, I_1^{GT}), (I_2^{LR}, I_2^{GT}), (\dots, \dots), (I_n^{LR}, I_n^{GT})\}$, where I is a face image, I^{LR} is the low-resolution, I^{GT} is the image with target resolution and n is the total image from the dataset. I^{GT} is an image get from cropping the face image with 128 x 128 pixel. While I^{LR} is resized from I^{GT} to 32 x 32 pixel. The preparation of LR and GT image is shown on Figure 3. Images size with 32 x 32 pixel will be up sampled using both super resolution methods, SRGAN and ESRGAN. Those methods will enlarge the images 4 times. So, the output size of these super resolution would be 128 x 128. Then the image will be feed and recognize using FaceNet.

Fig. 3 Preparation for GT and LR dataset

Generative Adversarial Network (GAN)

GAN is a better method for super resolution because with GAN the generated image looks more real. GAN firstly proposed by Ian Goodfellow through the concept of game theory in 2014 [23]. GAN consists of two neural networks: Generator and Discriminator. Generator (G) used for generating a fake image and Discriminator (D) used for learning what features that make image real. The main principle of GAN is to make G and D to compete with each other. G tries to generated an image so that D can not recognize if that image is not a real image. In the other hand, D keep learning on decide whether the input image is real or not. The GAN equation can be seen below.

$$\min_G \max_D V(D, G) = \mathbb{E}_{I^{HR} \sim p_{train}(I^{HR})} [\log D(I^{HR})] + \mathbb{E}_{I^{LR} \sim p_G(I^{LR})} [\log (1 - D(G(I^{LR})))] \quad (1)$$

where $\mathbb{E}$ denotes as the expected value, I^{HR} is the real image, and $G(I^{LR})$ denotes the generated image from generator. Equation 1 goals to optimize mini-max game with value $V(D, G)$. This equation has to maximize D value to recognize between real and fake images better. But on the other hand, it also has to minimize G so the generated image can fool the discriminator. The result of super resolution method will be evaluated using Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity (SSIM). SSIM value is between 0 and 1. Higher PSNR and SSIM value mean that the quality of super resolution image is better and more similar to the real image.

Super resolution GAN (SRGAN)

SRGAN method had been proposed by [1]. SRGAN used ResNet GAN Architecture, but there are several modifications to calculate the perceptual loss function. In this single image super resolution (SISR) task, SRGAN is aiming to predict HR image and resulting in the super-resolved image I^{SR} from the LR image I^{LR} . On the dataset, I^{LR} and I^{GT} are a pair from the dataset, I^{SR} is the predicted image from SRGAN. Based on the basic GAN, the ultimate goal of this method is to train the G network and the D network as the generator and discriminator network. The G network is trained using feed-forward CNN G_{θ_G} where θ_G denotes as the parameter for G . The value of $\theta_G = \{W_{1:L}, b_{1:L}\}$, where W and b denote as the weight and the bias of L -Layer deep network. The parameter is obtained by optimizing the loss function l^{SR} . From the pre-processing step, we get I_i^{LR} where $i = 1, 2, \dots, N$ and I_i^{HR} where $i = 1, 2, \dots, N$. So, the function is

$$\hat{\theta}_G = \arg \min_{\theta_G} \frac{1}{N} \sum_{i=1}^N l^{SR}(G_{\theta_G}(I_i^{LR}), I_i^{HR}). \quad (2)$$

According to the proposed formula of GAN [23] the objective of GAN is to solve equation 1. This equation's goal is to get G to generate a fake image that can fool D . D is also trained to classify whether the input image is a fake or real. The objection of generator and discriminator is the generator can generate an image that can hardly classify by discriminator.

The architecture of generator G consists of 16 residual blocks with the same layout. The layout consists of two convolutional layers with 3x3 kernels, 64 feature maps, batch normalization layers, and Parametric ReLU as the activation function. In the end, two sub-pixel convolutional layers are used for increasing the resolution of the input image. The architecture of G can be seen in Table 2. Discriminator D needs to be trained to differentiate the fake image and the real image. The architecture of D used LeakyReLU with $\alpha = 0.2$ as the activation function. The architecture of D starts with convolutional layers with 3x3 kernels and an activation function. Then there are 7 blocks that contains of convolutional layers with 3x3 kernels, batch normalization, and activation function. Each of blocks will increase the image resolution with factor of 2 from 64 to 512 kernel. The result of 512 feature maps is followed by two dense layers and sigmoid activation function as a probability to classify the image. The architecture of D can be seen in Table 3. SRGAN pre-trained using image dataset from ImageNet. ImageNet contains 350,000 images with various categories like animal, landscape, etc. Then the pre-trained weight will be used to finetune using YTF image dataset. The training process used 1000 iterations with 100 epochs.

Table 2 Architecture of SRGAN Generator Network

Layer	Output Shape	Param #
Input Layer	(None, 32, 32, 3)	0
Conv2D_54	(None, 32, 32, 64)	15616
Activation_20	(None, 32, 32, 64)	0
Residual Blocks - 16 Times		
Conv2D_55	(None, 32, 32, 64)	36928
Batch Normalization_48	(None, 32, 32, 64)	256
Activation_21	(None, 32, 32, 64)	0
Conv2D_56	(None, 32, 32, 64)	36928
Batch Normalization_49	(None, 32, 32, 64)	256
Add_18	(None, 32, 32, 64)	0
Conv2D_87	(None, 32, 32, 64)	36928
Batch Normalization_80	(None, 32, 32, 64)	256
Add_34	(None, 32, 32, 64)	0
UpSampling2D_3	(None, 64, 64, 64)	0
Conv2D_88	(None, 64, 64, 256)	147712
Activation_37	(None, 64, 64, 256)	0
UpSampling2D_4	(None, 128, 128, 256)	0
Conv2D_89	(None, 128, 128, 256)	590080
Activation_38	(None, 128, 128, 256)	0
Conv2D_90	(None, 128, 128, 3)	62211
Total params:	2,042,691	

Table 3 Architecture of SRGAN Discriminator Network

Layer	Output Shape	Param #
Input Layer	(None, 128, 128, 3)	0
Conv2D_46	(None, 128, 128, 64)	1792
Activation_23	(None, 128, 128, 64)	0
Conv2D_47	(None, 64, 64, 64)	36928
Batch Normalization_41	(None, 64, 64, 64)	256
Activation_24	(None, 64, 64, 64)	0
Conv2D_48	(None, 64, 64, 128)	73856
Batch Normalization_42	(None, 64, 64, 128)	512
Activation_25	(None, 64, 64, 128)	0
Conv2D_49	(None, 32, 32, 128)	147584
Batch Normalization_43	(None, 32, 32, 128)	512
Activation_26	(None, 32, 32, 128)	0
Conv2D_50	(None, 32, 32, 256)	295168
Batch Normalization_44	(None, 32, 32, 256)	1024
Activation_27	(None, 32, 32, 256)	0
Conv2D_51	(None, 16, 16, 256)	590080
Batch Normalization_45	(None, 16, 16, 256)	1024
Activation_28	(None, 16, 16, 256)	0
Conv2D_52	(None, 16, 16, 512)	1180160
Batch Normalization_46	(None, 16, 16, 512)	2048
Activation_29	(None, 16, 16, 512)	0
Conv2D_53	(None, 8, 8, 512)	2359808
Batch Normalization_47	(None, 8, 8, 512)	2048
Activation_30	(None, 8, 8, 512)	0
Dense_3	(None, 8, 8, 1024)	525312
Activation_31	(None, 8, 8, 1024)	0
Dense_4	(None, 8, 8, 1)	1025
Total params:	10,434,562	

Enhanced Super resolution GAN (ESRGAN)

ESRGAN method had been proposed by [2]. ESRGAN network architecture is Residual in Residual Dense Block (RRDB). On the basic ResNet GAN, the generator network architecture contains a convolutional layer, batch normalization (BN) layer, and activation layer. In RRDB, it removed all BN layers and added Dense Block (DB) on every block. BN layer is removed from the basic block on ResNet GAN because it has proven to increase the performance and reduce the complexity of SUPER RESOLUTION [30] and deblurring [31] tasks. BN layer is used to do the normalization of the feature using a current batch's mean and variance during the training phase. Meanwhile, in the testing phase, the BN layer will normalize the feature using the whole training dataset estimated mean and variance. In case when the training and testing dataset differs a lot, then BN layer will reduce the performance and limit the generalization ability. So [2] removed BN layer to produce a stable and consistent performance during training.

Table 4 Architecture of ESRGAN Generator Network

Layer	Output Shape	Param #
Input Layer	(None, 32, 32, 3)	0
Conv2D_20	(None, 32, 32, 64)	15616
Activation_15	(None, 32, 32, 64)	0
	Dense Block 3 times	
Conv2D_21	(None, 32, 32, 64)	36928
Activation_16	(None, 32, 32, 64)	0
Concatenate	(None, 32, 32, 64)	0
Conv2D_22	(None, 32, 32, 128)	73856
Activation_17	(None, 32, 32, 128)	0
Concatenate	(None, 32, 32, 128)	0
Conv2D_23	(None, 16, 16, 128)	147712
Activation_18	(None, 16, 16, 128)	0
Concatenate	(None, 16, 16, 128)	0
Conv2D_24	(None, 16, 16, 256)	295424
Activation_19	(None, 16, 16, 256)	0
Concatenate	(None, 16, 16, 256)	0
Conv2D_25	(None, 16, 16, 512)	590848
Activation_20	(None, 16, 16, 512)	0
Concatenate	(None, 16, 16, 512)	0

Conv2D_26	(None, 64, 64, 256)	36928
UpSampling2D_3	(None, 128, 128, 256)	147712
Conv2D_27	(None, 128, 128, 256)	36928
Conv2D_28	(None, 128, 128, 3)	1731
Total params:	3,802,243	

In CNN, more layers and connections will always result in better performance [32][33]. RRDB network uses dense block on the main path to make the network capacity better with the dense connection. Residual scaling and smaller initialization were also used to improve the architecture. Scaling down the residual with β where $0 < \beta < 1$ before adding to the main path will stable the process. Residual architecture also trains easier when the initial parameter variance is smaller. There is also improvement on the Discriminator based on Relativistic GAN. The standard Discriminator (D) evaluates if images x is naturally real or fake. But relativistic discriminator will evaluate if image x_r is relatively more realistic than the fake x_f . Relativistic Discriminator D_{Ra} can be calculated as

$$D_{Ra}(x_r, x_f) = \sigma(C(x_{rf}) - \mathbb{E}_{x_f}[C(x_f)]) \quad (3)$$

where σ is a sigmoid function, C is the discriminator output, and $\mathbb{E}_{x_{fake}}$ is the average of all the fake images in a single batch.

The architecture of generator G used convolutional layer with 3x3 kernels, 64 features map, 1 stride, and Leaky ReLU as the activation function. Then there are 3 dense block that substitute residual blocks in SRGAN. Each dense block consists of 4 pairs of convolutional layers with 3x3 kernels, 64 features map, 1 stride, and Leaky ReLU as the activation function with $\alpha = 0.2$. At the end of the dense block, there is a convolutional layer with 3x3 kernels, 64 features map, 1 stride. Then there is a skip connection between the first dense block until the last convolutional layer. The architecture of generator network in ESRGAN can be seen in Table 4.

The loss function in ESRGAN used equation 3. The loss function calculates the images and compares it with the average of real and fake images. ESRGAN pre-trained using image dataset from ImageNet. ImageNet contains of 350,000 images with various categories like animal, landscape, etc. Then the pre-trained weight will be used to finetune using YTF image dataset. The training process also used 1000 iterations with 100 epochs.

FaceNet

FaceNet is a method proposed by [11]. It uses a deep CNN architecture with triplet loss to get an embedding that can be used for face clustering, verification, and recognition. FaceNet provides a unified embedding $f(I)$ that maps face I into a d – dimensional Euclidian space, where $f(I) \in \mathbb{R}^d$. This triplet loss method is inspired by nearest-neighbor classification. Triplet loss is a loss function that will ensure that an image of a person face I_i^a (anchor) is closer to all other image I_i^p (positive) of the same person than to all other image I_i^n (negative) of other person. The triplet loss function L can be defined as

$$L = \sum_{i=1}^N \left[\|f(I_i^a) - f(I_i^p)\|_2^2 - \|f(I_i^a) - f(I_i^n)\|_2^2 + \alpha \right]_+, \quad (4)$$

where N is the amount of image in a set of all possible triplet pairs on the training set and α is a difference margin between the anchor-positive and anchor-negative. FaceNet trains the CNN using Stochastic Gradient Descent (SGD) with back propagation and adaptive gradient. The initial learning rate is 0.05, the α is 0.2 and ReLU as the activation function. The LFW dataset from super resolution is used on the training and evaluation phase. Every identity will be split into training set, validation set and testing set. The portion of the dataset is 90% for training, 5% for validation, and 5% for testing. The minimum image for each identity is 4 images. 2 for training, 1 for validation, and 1 for testing. Validation is used to prevent the training process from getting underfitting or overfitting.

During training and testing $L2$ – distance threshold will be set on 1.242 based on [11] experiment. The squared $L2$ of $D(I_i, I_j)$ is used to differentiate if those pair image is same or different. All pair faces (i, j) of the same person is denoted as P_{same} and all pair faces (i, j) of different person is denoted as P_{diff} . A set of true accept identities are calculate with

$$TA(d) = \{(i, j) \in P_{same}, D(i, j) \leq d\}. \quad (5)$$

It will create a set of face pair (i, j) that correctly classified as the same person with threshold d . Then

$$FA(d) = \{(i, j) \in P_{diff}, D(i, j) \leq d\} \quad (6)$$

will create a set of face pair (i, j) that correctly classified as a different person with threshold d .

The Validation rate ($Val(d)$) will be calculated with $Val(d) = \frac{|TA(d)|}{|P_{same}|}$, and the False accept rate ($Far(d)$) will be calculated with $Far(d) = \frac{|FA(d)|}{|P_{diff}|}$. validation rate is used to measure the True Accept rate comparing to all data pair of same identities.

Experimental Result and Discussion

The comparison result between SRGAN [1] and ESRGAN [2] on LFW images can be seen in Table 5. ESRGAN has achieved a better result in PSNR and SSIM comparing to SRGAN. This result achieved by using weight from training phase with YTF dataset.

Table 5 Super resolution comparison result using LFW dataset

Super Resolution Method	Average PSNR	Average SSIM
SRGAN	30.53	0.89
ESRGAN	39.19	0.96

Figure 4 shows the comparison result of several source images. The first is the Ground Truth (GR) image with 128 x 128 pixel. Then the second image is LR image with 32 x 32 pixel. As can be seen in figure 4, LR image is very blurry and will decrease the accuracy of FaceNet. The third image is super resolution image using SRGAN with 128 x 128 pixel. The fourth image is super resolution image using ESRGAN with 128 x 128 pixel. For these 2 sample images, ESRGAN has a better result for PSNR and SSIM. The difference result between SRGAN and ESRGAN for increasing the accuracy will be shown at the conclusion of this research.

Fig. 4 Super resolution result

Figure 5 shows the training result for FaceNet using the dataset that had been increased by SRGAN. After using the pre-trained weight, the accuracy of epoch 0 reached by SRGAN was 75.5%. The accuracy increased significantly in the first 20 epoch. The accuracy reached on epoch 20 was over 90 %. But the accuracy increased slowly on the next epoch. The accuracy started to converge approximately on epoch 70. On epoch 70 the accuracy had reached 97 %. Until epoch 100, the accuracy that had been reached by FaceNet using SRGAN was 97.02 %. On the other hand, the validation accuracy on epoch 0 reached 92.3%. The validation accuracy increased up to 96% on

epoch 50. After that, the accuracy still increased slowly and converged approximately on epoch 60. Until epoch 100, the accuracy using the validation set by FaceNet using SRGAN was 98.03 %.

The training loss that had been reached on epoch 0 was 9.546. The loss value decreased significantly on epoch 20. The loss value on that epoch was approximately 2.5. The training loss kept decreasing until 1.5 at epoch 50. Training loss started to converge at epoch 60. Until epoch 100, the training loss that had been reached by FaceNet using SRGAN was 0.935. The validation loss on epoch 0 was 1.68. Validation loss kept decreasing until 0.5 on epoch 20. Validation loss started to converge approximately at epoch 80. The validation loss reached on epoch 100 was 0.237.

Figure 6 shows the training result for FaceNet using the dataset that had been increased by ESRGAN. After using the pre-trained weight, the accuracy on epoch 0 reached by SRGAN was 76.4%. The accuracy increased significantly on the first 20 epoch. The accuracy reached on epoch 20 was approximately on 93 %. On the next epoch, the accuracy kept increased slowly until epoch 70 was approximately on 97%. The accuracy started to converge approximately on epoch 85. Until epoch 100, the accuracy that had been reached by FaceNet using ESRGAN was 98.4 %. The validation accuracy on epoch 0 reached 92.5%. The validation accuracy increased until 95.5% on epoch 20. Validation accuracy kept increased slowly until 97% on epoch 70. It started to converge on epoch 80. Until epoch 100, the accuracy using the validation set by FaceNet using ESRGAN was 98.4 %.

The training loss that had been reached on epoch 0 was 9.46. The loss value decreased significantly until epoch 30. The loss value on that epoch was approximately 2.0. The training loss kept decreasing until epoch 60. Training loss on epoch 60 was approximately at 1.0. Until epoch 100, the training loss that had been reached by FaceNet using SRGAN was 0.92. The validation loss on epoch 0 was 1.46. Validation loss kept decreasing until 0.3 on epoch 30. Validation loss started to converge approximately at epoch 80. The validation loss on that epoch was 0.27. The validation loss reached on epoch 100 was 0.244.

The testing comparison result can be seen on Table 6. After training the FaceNet model using the training and validation set, the model was tested using the testing set. The first result is using low-resolution image and recognized by FaceNet without preprocessing. The accuracy is 75.28 % and the validation rate is 55.22 %. The second result is using super resolution image with SRGAN. Then the data recognize by FaceNet and reached the accuracy of 98.53 % and validation rate of 94.67 %. The accuracy on Facenet using super resolution image with ESRGAN reached 98.96 % and the validation rate was 96.757 %. The result is better because ESRGAN has a higher value of PSNR and SSIM than SRGAN. ESRGAN has a higher value of PSNR and SSIM by 8.66 and 0.07. The fourth result is testing the FaceNet with high resolution image. The image resolution is 160 x 160 pixel. The accuracy reached was 99.25 % and validation rate was 97.34 %. This result is also compared with [3]. [3] recognized low-resolution image using FaceNet without preprocessing and reached the accuracy of 98.5 % and validation rate of 88.5 %. The best method reached in this research is by using ESRGAN [2] as the preprocessing. This accuracy is 0.46 % and the validation rate is 8.26 % higher than [3].

Table 6 Comparison results on FaceNet

Author	Pre-Processing	Face Recognition	Size		Super resolution Evaluation		Face Recognition Evaluation	
			LR	HR	PSNR	SSIM	Accuracy	Validation rate
<i>proposed</i>		FaceNet	32 x 32	-	-	-	75.28 %	55.22%
	SRGAN		32 x 32	128 x 128	30.53	0.89	98.53 %	94.67 %
	ESRGAN		32 x 32	128 x 128	39.19	0.96	98.96 %	96.76 %
			-	160 x 160	-	-	99.25 %	97.34 %
[3]	-		40 x 40				98.50 %	88.50 %

Conclusion

This research proposed a model to solve the LRFR problem. This research proposed super resolution methods using SRGAN and ESRGAN before recognition method using FaceNet. This model was evaluated using LFW dataset. The result showed that using super resolution before face recognition can increase accuracy. Table 6 shown that using FaceNet directly on an image with 32 x 32 pixel had an accuracy of 75.28%. After using super resolution image, the accuracy increased to over 98%. The highest accuracy achieved in this research is by using ESRGAN as the super resolution method. FaceNet with ESRGAN achieved an accuracy of 98.96% and VAL rate of 96.76%. This result is getting closer to FaceNet with HR images. For further research, some features on face recognition model like gender and age category could be added. It would narrow down the possibility for FaceNet to false predict.

Acknowledgements

This research was supported by Research Grant No. 039/VR.RTT/IV/2019 from Ministry of Research Technology and Higher Education of Republic of Indonesia.

Funding

Not applicable

Abbreviations

GAN: Generative Adversarial Network

Availability of data and materials

Not applicable

Ethics approval and consent to participate

Not applicable

Competing interests

This research has no competing interests

Consent for publication

Not applicable.

Authors' contributions

All authors read and approved the final manuscript

Author details

¹Computer Science Department, BINUS Graduate Program - Master of Computer Science, Bina Nusantara University, Jakarta 11480, Indonesia.

²Department Computer Science, BINUS Online Learning, Bina Nusantara University, Jakarta, 11480, Indonesia

References

- [1] C. Ledig *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network," *Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017*, vol. 2017-Janua, pp. 105–114, 2017, doi: 10.1109/CVPR.2017.19.
- [2] X. Wang *et al.*, "ESRGAN: Enhanced super-resolution generative adversarial networks," 2019, doi: 10.1007/978-3-030-11021-5_5.
- [3] M. R. Golla and P. Sharma, *Performance evaluation of facenet on low resolution face images*, vol. 839. Springer Singapore, 2019.
- [4] S. Singh and S. V. A. V. Prasad, "Techniques and challenges of face recognition: A critical review," *Procedia Comput. Sci.*, vol. 143, pp. 536–543, 2018, doi: 10.1016/j.procs.2018.10.427.
- [5] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep Face Recognition," *Proceedings Br. Mach. Vis. Conf. 2015*, no. Section 3, pp. 41.1-41.12, 2015, doi: 10.5244/C.29.41.
- [6] G. B. Huang, M. Mattar, T. Berg, and E. Learned-miller, "Labeled Faces in the Wild : A Database for Studying Face Recognition in Unconstrained Environments," *Tech. rep.*, 2008.
- [7] Z. Lu, X. Jiang, and A. Kot, "Deep Coupled ResNet for Low-Resolution Face Recognition," *IEEE Signal Process. Lett.*, vol. 25, no. 4, pp. 526–530, 2018, doi: 10.1109/LSP.2018.2810121.
- [8] T. Bao, C. Ding, S. Karmoshi, and M. Zhu, "Video-based face recognition via convolutional neural networks," *Second Int. Work. Pattern Recognit.*, vol. 10443, p. 1044301, 2017, doi: 10.1117/12.2280286.
- [9] P. Li, L. Prieto, D. Mery, and P. Flynn, "Low Resolution Face Recognition in the Wild," pp. 1–11, 2018, [Online]. Available: <http://arxiv.org/abs/1805.11529>.
- [10] J. W. Hwang and H. S. Lee, "Adaptive Image Interpolation Based on Local Gradient Features," *IEEE Signal Process. Lett.*, vol. 11, no. 3, pp. 359–362, 2004, doi: 10.1109/LSP.2003.821718.
- [11] F. Schroff and J. Philbin, "FaceNet : A Unified Embedding for Face Recognition and Clustering," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 815–823.
- [12] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc.*, pp. 1–14, 2015.
- [13] L. Wolf, T. Hassner, and I. Maoz, "Face Recognition in Unconstrained Videos with Matched Background Similarity."
- [14] H. M. Chang, H. Seo, and S. Bum, "A face recognition system based on convolution neural," *Soft Comput.*, 2016, doi: 10.1007/s00500-016-2095-0.
- [15] X. Tang, "DeepID3: Face Recognition with Very Deep Neural Networks," pp. 2–6, 2015.
- [16] C. Herrmann, D. Willersinn, and J. Beyerer, "Low-resolution Convolutional Neural Networks for video face recognition," *2016 13th IEEE Int. Conf. Adv. Video Signal Based Surveillance, AVSS 2016*, no. August, pp. 221–227, 2016, doi: 10.1109/AVSS.2016.7738017.
- [17] S. Ng, Hong-Wei, Winkler, "A DATA-DRIVEN APPROACH TO CLEANING LARGE FACE DATASETS Hong-Wei Ng and Stefan Winkler Advanced Digital Sciences Center (ADSC), University of Illinois at Urbana-Champaign , Singapore," *2014 IEEE Int. Conf. Image Process.*, pp. 343–347, 2014.
- [18] X. Zhang, L. Zhang, S. Member, and X. Wang, "Finding Celebrities in Billions of Web Images," vol. 14, no. 4, pp. 995–1007, 2012.
- [19] M. Grgic, K. Delac, and S. Grgic, "SCface – surveillance cameras face database," no. October 2009, pp. 863–879, 2011, doi: 10.1007/s11042-009-0417-2.
- [20] V. K. Tanwar, "Privacy preserving image scaling using 2D bicubic interpolation over the cloud," *2018 IEEE Int. Conf. Syst. Man, Cybern.*, pp. 2073–2078, 2018, doi: 10.1109/SMC.2018.00357.
- [21] W. Yang, X. Zhang, Y. Tian, W. Wang, J. Xue, and Q. Liao, "Deep Learning for Single Image Super-Resolution :," no. 1, pp. 1–16.
- [22] W. S. Lai, J. Bin Huang, N. Ahuja, and M. H. Yang, "A Fully Progressive Approach to Single-Image Super-Resolution Supplemental Material," *Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017*, vol. 2017-Janua, pp. 5835–5843, 2017, doi: 10.1109/CVPR.2017.618.
- [23] I. J. Goodfellow *et al.*, "Generative Adversarial Networks," pp. 1–9, 2014, [Online]. Available: <http://arxiv.org/abs/1406.2661>.
- [24] A. Radford, L. Metz, and S. Chintala, "Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks," pp. 1–16, 2015.
- [25] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," 2015, doi: 10.1109/CVPR.2016.90.
- [26] S. Karamizadeh and S. M. Abdullah, "Race classification using gaussian-based weight K-nn algorithm for face recognition," vol. 6, no. M, 2018.
- [27] H. S. Dadi, G. Krishna, and M. Pillutla, "Improved Face Recognition Rate Using HOG Features and SVM Classifier," vol. 11, no. 4, pp. 34–44, 2016, doi: 10.9790/2834-1104013444.
- [28] K. Zhang, Z. Zhang, Z. Li, S. Member, Y. Qiao, and S. Member, "Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks," *IEEE Signal Process. Lett.*, vol. 23, no. 10, pp. 1499–1503, 2016.
- [29] R. A. A. Al-falujji, A. A. Youssif, and S. K. Guirguis, "Single Image Super Resolution Algorithms : A Survey and Evaluation," *Int J Adv Res Comput Eng Technol*, vol. 6, no. 9, pp. 1445–1451, 2017.
- [30] B. Lim and K. M. Lee, "Enhanced Deep Residual Networks for Single Image Super-Resolution."
- [31] S. Nah, T. H. Kim, and K. M. Lee, "Deep Multi-scale Convolutional Neural Network for Dynamic Scene Deblurring."

[32] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu, "Residual Dense Network for Image Super-Resolution."

[33] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu, "Image Super-Resolution Using Very Deep Residual Channel Attention Networks," pp. 15–17.

Figures

Figure 1

Proposed method's flow diagram

Figure 2

Align and crop face image

Figure 3

Preparation for GT and LR dataset

Figure 4

Super resolution result

Figure 5

Accuracy and loss on FaceNet using SRGAN

Figure 6

Accuracy and loss on FaceNet using ESRGAN