International Journal of Engineering Trends and Technology
ISSN: 2231 – 5381 /doi:10.14445/22315381/IJETT-V70I3P201

Volume 70 Issue 3, 1-12, March, 2022

Original Article

Face Image Super-Resolution Using Combination of Max-Feature-Map and CMU-Net to Enhance Low-Resolution Face Recognition

Yulianto[1], Nurhasanah[2], Risma Yulistiani[3], Gede Putra Kusuma[4]

[1,2,3] *Scholar, Computer Science Department, BINUS Graduate Program - Master of Computer Science, Bina Nusantara University, Jakarta, Indonesia.*
[4]*Assistant Professor, Computer Science Department, BINUS Graduate Program - Master of Computer Science, Bina Nusantara University, Jakarta, Indonesia.*

[1]yulianto003@binus.ac.id, [2]nurhasanah001@binus.ac.id, [3]risma.yulistiani@binus.ac.id

Abstract *- The far distance of the camera while taking a picture makes the visual image look blurred. In deep learning, the blurred image eliminates the classification accuracy. The decreasing classification accuracy is caused by the loss of detailed information on High-Resolution (HR) images. The Generative Adversarial Networks (GAN) for Super-Resolution (SR) can overcome the problem and deal with the blurred images. In the usual term, GAN is used for reconstructing the quality of images visually. While not all GANs may be utilized to increase classification accuracy, SR image findings are highly realistic. This study offers a CMU-Net MFM SR technique based on this challenge, which combines a modified U-Net plus a Max-Feature-Map (MFM) module and a mix of BCE loss, Cosine matrix loss, and magnitude loss to restore the identity result of the SR image. With a classification accuracy of 78.45%, the experimental results of this method employing the LFW (Labelled Face in the Wild) dataset may be utilized to boost the image resolution from 8 x 8 pixels to 64 x 64 pixels.*

Keywords – *Generative Adversarial Networks, U-Net Model, Super-Resolution, Convolution Neural Networks, Low-Resolution Face Recognition.*

I. INTRODUCTION

Developing deep learning algorithms like Convolutional Neural Network (CNN) is robust for image recognition and classification [1]. Some areas are needed for image recognition, such as forensic and biometric [1][2]. The focus of facial image recognition research is separated into two fields: High-Resolution (HR) facial image recognition, also known as High-Resolution Face Recognition (HRFR) [3][4], and Low Resolution (LR) facial image recognition, also known as Low-Resolution Face Recognition (LRFR) [5]. If the dimensions of the original image produced from the camera sensor are between 112 × 112 pixels and 224 x 224 pixels, and the image seems crisp and sharp, the image is categorized as HR. When the resolution of a face image is less than 50 pixels (for example, 48 x 48 pixels, 40 x 40 pixels, 32 x 32 pixels, 24 x 24 pixels, 16 x 16 pixels, and 8 x 8 pixels), the image as a low resolution [6]. Visual observation shows that the LR face image is a blur. The CNN algorithm has now attained a very high accuracy in recognizing HR facial photographs. However, if the CNN model has been previously trained with an HR image dataset, then the model is used to classify LR image input, the accuracy of CNN classification will drop dramatically by up to 20% [3].

In General, the image resolution degraded since the distance while taking the picture, object moving condition, and the hardware specification designed to optimize electronic storage space and conditions of limited hardware specifications. All of this can cause the acquired picture to lose much detail and high-frequency information characteristics, making the image appear blurry, and can result in a drop in face image recognition accuracy if the image is classed using a deep-learning algorithm [7]. According to [8], low-resolution face photos include fewer structural elements of identification detail information than HR images, resulting in a 20% loss in classification accuracy in LR images with a size of 32 × 32 pixels. Even after employing bicubic interpolation to increase the facial picture, it still loses a lot of detailed identifying information [8]. The interpolation method is divided into two categories. First, non-learning-based interpolation uses data from neighbouring pixels in techniques like bilinear and bicubic interpolation [9]. Second, interpolation reconstructs image based-learning process that proposed [10] sampling and up-sampling using SR Generative Adversarial Networks (GAN) simultaneously for LR cases.

The study of LRFR consists of two techniques: the SR, which focuses on minimizing distances feature identity [11] by calculating the distance in the pixel domain using the L1 norm because distance calculations are sensitive to poses, lighting, and expressions. Another method of the SR technique ponders the difference in distance of feature space

between the original HR image and LR image [8]. The angle and norm-diff discrepancies in the feature space will be more significant when the picture resolution is reduced compared to the initial resolution. The difference between an angle and norm-diff distance is vast, resulting in losing identification information in face images and reduced classification accuracy. In order to reduce the angle difference, SR GAN uses a network whose generator uses a residual block to combine loss function with training the GAN network. The loss function is the cosine matrix loss associated with features that can enhance classification accuracy and reduce norm-diff space when used with the loss magnitude function. Visually, the magnitude loss increases image quality. The maximum accuracy was 98.98% when using the LFW dataset with a 16 by 16-pixel LR picture input.

This work presents LR face image identification using the SR approach with an 8 × 8 pixels LR picture. The GAN U-Net model was used to project LR face images onto an HR image dimension of 64 × 64 pixels SR [12]. The model may be trained on a small but accurate dataset condition that reconstructs face pictures with low FID scores, meaning they are as like the original image as feasible. The MFM (Max-Feature-Map) activation function from the LightCNN-29v2 architecture [12] was added to the U-Net architecture. MFM has been added to U-Net to improve feature extraction. Multiple loss functions are added to the discriminator section to recreate face identification in the feature space. The inclusion of cosine matrix loss [13] helps reduce the disparity in angle distance and increases classification accuracy in SR pictures. Additionally, the magnitude loss reduces the norm-diff distance, making the visually SR picture outputs more real.

II. RELATED WORKS

The earlier study of an LR face image identification approach without an SR methodology defined like The Multi-Resolution Convolutional Neural Network is a technology used called (MRCNN). The accuracy of the experimental results which use the LFW dataset was 70% higher than the bicubic method. One proposed LR facial image recognition technique utilizes a training technique involving two networks with the same VGG architecture [14]. One of the networks enables as a network teacher, extracting HR face images, while the other behaves like a student network, extracting LR image attributes. The most incredible accuracy in testing using the LFW dataset is 97.15%.

GAN is a profound image classification method. The Low-Resolution Facial Recognition Technique (LRFR) was developed using the CMU PIE dataset and a GAN combination [15], with a resolution of 640 × 486 pixels and classification accuracy of 0.9157. Siamese GAN (SiGAN), which also includes Generator identity embedding GAN (GieGAN), where the network generator is a version of DCGAN, and Discriminator identity embedding GAN (DieGAN) [11], is used to address LR (8 × 8 pixels) face recognition. GAN and fidelity losses are combined in the loss function of DieGAN. The LFW dataset was labelled to be 81.2% accurate. The study of Cascaded SR and Identity Prior GAN (C-SRIP GAN) focused on FRLR for a 24 × 24-pixel image input [16]. A tiered residual design serves as a generator network, whereas SSIM serves as a loss of function. The pre-cart model from SqueezeNet is also used. PSNR 27.995 and SSIM 0.8769 were acquired utilizing the LFW dataset in the assessment procedure.

Increasing the image accuracy in the biomedical field using the U-Net core network won the ISBI 2015 cell tracking contest [17]. Image in the biomedical field using the U-Net core network and won the ISBI 2015 cell tracking contest. When there is a limited quantity of training data, the U-Net network shines. The grayscale image received an average IOU (Intersection Over Union) value of 92% when tested with the PhC-U373 cell dataset and input image size of 572 × 572 pixels. Executing U-Net-based feature extraction on the generative network and picture translation with Conditional GAN (cGAN) [18]. All training procedures may simultaneously balance the network generator and discriminator by introducing skip connections at each step.

When evaluating GAN performance in general, aim to make cautious modifications to the GAN [19]. The impact of monitoring values as assessment techniques for evaluating generator results during the training process on SR image results is significant. Two types of evaluations may be employed. Frechet Inception Distance (FID) and Inception-Score (IS) based on evaluation can help to reduce noise. The lower FID, the more realistic the image produced by the generator network.

By adopting SR on the PSNR and SSIM values, namely Multi-Scale Gradient GAN, the picture outputs would be altered [20] by using SR on the PSNR and SSIM values Multi-Scale Gradient Generative Adversarial Network (MSG-GAN). CapsNet, a discriminator with sigmoid activation functions 1 and 0, has the same down-sampling architecture as the generative section, which does down sampling progressively ranging from 128 x 128 pixels, 64 x 64 pixels, and 32 x 32 pixels and has just been categorized by CapsNet. On the other hand, the network generator component implements up-sampling in steps, commencing with 32 x 32 pixels, 64 x 64 pixels, and 128 x 128 pixels. In the discriminative step, each layer output from generative up-sampling is merged in pairs with the output of the down-sampling block. PSNR values of 23.35 and SSIM of 0.673 were produced using the CelebA dataset, perception of loss using VGG, and optimization of Adam.

One of the variants of CNN to image recognition is the Light CNN architecture utilized for deep face representation. Light CNN is a CNN architecture investigating compact embedding on large-scale facial identification with noisy labels and other applications. The Max Feature Map is one

of the types of max-out activation used by CNN (MFM). MFM is used to conduct feature filter selection in order to distinguish between noise and meaningful data. MFM is responsible for determining the best characteristics for each place investigated by each filter. During backpropagation, MFM creates a binary gradient to suppress a neuron. For facial identification, LightCNN-29 is a LightCNN architecture with a 29-layer convolutional network. On LightCNN-29, the residual block has two 33 convolution layers, FM operations without batch normalization, 12,637K parameters, and 3.9G FLOPS [12].

The resolution of the image will affect facial identity recognition. Because recognizing identity becomes more complex as resolution decreases, an SR approach is required to overcome the challenge of boosting identity-aware accuracy. Identity awareness focuses on the angle and magnitude of the features, notably the cosine and magnitude loss features, to employ information from identity efficiently[8]. The angle size, and both the feature size, is affected by the difference between LR and HR. The considerable angle difference causes the loss in the cosine matrix. In hypersphere identity metric space, features with lower magnitude values will be susceptible to feature disturbances, but features with larger magnitude values will be susceptible to feature disturbances in magnitude loss. This research presents a CNN-29v2 to collect HR information from LR pictures adequately. Method for extracting light features.

III. PROPOSED METHOD

The proposed method is SR called MFM CMU-Net (Max-Feature-Map Cosine Magnitude U-Net), which can be utilized to SR image from 8 × 8-pixel LR resolution to 64 x 64-pixel SR resolution. The SR approach can increase the accuracy of face image categorization and generate realistic SR images. The architecture in Fig. 1 is a combination of bicubic interpolation [21] to interpolate input images from LR images 8 x 8 pixels to 64 x 64 pixels. Meanwhile, a network generator architecture based on U-Net, previously employed as an SR image, is used to rebuild the 64 x 64-pixel LR face picture [13] Improving the extraction of facial image features in the U-Net design, the Max Feature Map (MFM) activation function was included, as shown in the LightCNN architecture for HR face image classification [12]. The discriminator network used to train the U-Net Generator network was based on satellite picture segmentation design [22]. Furthermore, loss combinations such as Binary Cross-Entropy (BCE), Cosine, and Magnitude Loss [8] were used to save high-frequency identity features, increasing SR face picture classification accuracy.

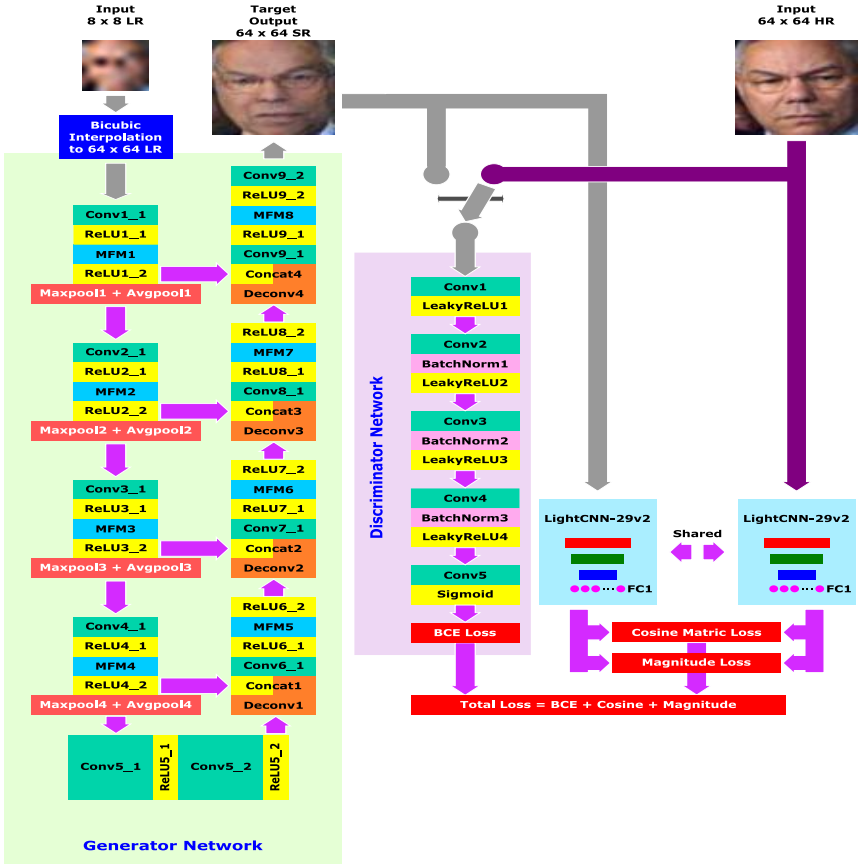


Fig. 1 Proposed method MFM CMU-Net (Max-Feature-Map Cosine Magnitude U-Net)

To train the CMU-Net MFM network proposed in Fig. 1, there are three images inputs involved: 8 x 8 LR images, 64 x 64 SR images, and 64 x 64 HR images. The 8 x 8 LR picture was created by bicubic down sampling the 64 × 64 HR image. Meanwhile, the picture formed by the network generator output creates the 64 x 64 SR image. The training dataset provided the 64 × 64 HR picture. In one epoch, there are three training steps. The first training discriminator network uses a 64 × 64 HR picture input. The difference between the loss and the BCE loss with a real label or 1 is then determined using the anticipated outcomes of the discriminator network output.

Updating weight on the discriminator network uses loss value. The second step, train the discriminator network using input image SR 64 x 64 pixels. SR image is also known as Fake image. The network generator generates the SR 64 x 64 pixels using image LR 8 x 8 pixel as input. The location of the SR picture is separated before it reaches the discriminator network to avoid weight changes to the network generator. It is necessary to detach the SR image since it solely focuses on updating the discriminator network. The difference between the loss and the BCE loss against the fake label or integer number 0 is then determined using the prediction output of the discriminator network that receives the input SR picture. After then, the loss value is solely utilized to update the discriminator network section. A network generator, a discriminator network, and a LightCNN-29v2 pre-train network are used in the third phase of the training process. A 64 × 64 HR picture dataset was used to train the LightCNN-29v2 pre-train network. In this study, 256 feature vector outputs from FC1 were used to calculate the cosine and magnitude loss. The difference in loss with BCE versus the true label or 1. is used to measure the prediction outcome of the discriminator network that gets the input SR picture in this third training stage.

A. Generator Network

A network generator aims to reconstruct or generate new images with higher resolution than the original image. A fake picture or SR image is the image formed by the network generator output. The U-Net design is utilized in the network generator [21]. The U-Net architecture has been tweaked to accommodate images input with 64 x 64 pixels resolution, and an MFM (Max-Feature-Map) module [22] has been added to enhance feature extraction. Table 1 describes the complete design of the network generator component in Fig. 1, especially to generator network block.

MFM1 to MFM8 modules that existed in generator networks Fig. 1 are listed in Table 1. The MFM module is based on the findings of [22]. Fig 2 shows the contents of the MFM module in detail. It utilizes the MFM module to replace Leaky ReLU (LReLU) activation function. The MFM module works through convolution, with the number of filter outputs being twice as many as the number of input filters. The convolution layer is used for generating multiple raw features from images input [23]. The output of the filters is then divided into two groups: group A and group B. The filter group with the highest total number is the one that can be output. As a result, the MFM output of the module can come from either Group A or B. A skip connection is introduced to the output of the MFM module Fig. 2 to prevent va gradients due to deep architectural strands. Table 1 shows the Conv2D parameters in the MFM module, such as input filters, output filters, kernel size, stride, and padding, as depicted in Fig 2.

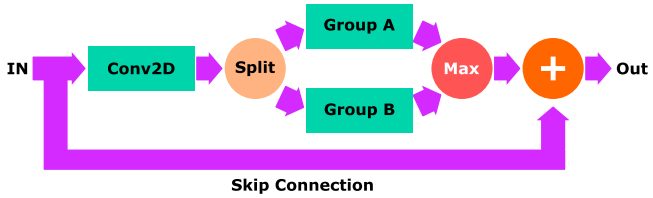


Fig. 2 Inside module MFM (Max-Feature-Map) with skip connection that used in U-Net architecture

Table 1. Generator network

Layer	Input Resolution (Pixel)	Filter Input	Filter Output	Kernel Size	Stride	Padding	Output Resolution (Pixel)
Conv1_1	64 x 64	3 (RGB)	64	3 x 3	1	1	64 x 64
ReLU1_1							
MFM1	64 x 64	64	64	3 x 3	1	1	64 x 64
ReLU1_2							
Maxpool1 + Avgpool1	64 x 64	64	64	2 x 2	2	0	32 x 32
Conv2_1	32 x 32	64	128	3 x 3	1	1	32 x 32
ReLU2_1							
MFM2	32 x 32	128	128	3 x 3	1	1	32 x 32
ReLU2_2							
Maxpool2 + Avgpool2	32 x 32	128	128	2 x 2	2	0	16 x 16
Conv3_1	16 x 16	128	256	3 x 3	1	1	16 x 16

Layer	Input Resolution (Pixel)	Filter Input	Filter Output	Kernel Size	Stride	Padding	Output Resolution (Pixel)
ReLU3_1							
MFM3	16 x 16	256	256	3 x 3	1	1	16 x 16
ReLU3_2							
Maxpool3 + Avgpool3	16 x 16	256	256	2 x 2	2	0	8 x 8
Conv4_1	8 x 8	256	512	3 x 3	1	1	8 x 8
ReLU4_1							
MFM4	8 x 8	512	512	3 x 3	1	1	8 x 8
ReLU4_2							
Maxpool4 + Avgpool4	8 x 8	512	512	2 x 2	2	0	4 x 4
Conv5_1	4 x 4	512	1024	3 x 3	1	1	4 x 4
ReLU5_1							
Conv5_2	4 x 4	1024	1024	3 x 3	1	1	4 x 4
ReLU5_2							
Deconv1	4 x 4	1024	512	2 x 2	2	0	8 x 8
Concat1	8 x 8	Output Deconv1 concat with output ReLU4_2 (512 + 512 = 1024)	512				8 x 8
Conv6_1	8 x 8	1024	512	3 x 3	1	1	8 x 8
ReLU6_1							
MFM5	8 x 8	512	512	3 x 3	1	1	8 x 8
ReLU6_2							
Deconv2	8 x 8	512	256	2 x 2	2	0	16 x 16
Concat2	16 x 16	Output Deconv2 concat with output ReLU3_2 (256 + 256 = 512)					16 x 16
Conv7_1	16 x 16	512	256	3 x 3	1	1	16 x 16
ReLU7_1							
MFM6	16 x 16	256	256	3 x 3	1	1	16 x 16
ReLU7_2							
Deconv3	16 x 16	256	128	2 x 2	2	0	32 x 32
Concat3	32 x 32	Output Deconv3 concat with output ReLU2_2 (128 + 128 = 256)					32 x 32
Conv8_1	32 x 32	256	128	3 x 3	1	1	32 x 32
ReLU8_1							
MFM7	32 x 32	128	128	3 x 3	1	1	32 x 32
ReLU8_2							
Deconv4	32 x 32	128	64	2 x 2	2	0	64 x 64
Concat4	64 x 64	Output Deconv4 concat with output ReLU1_2 (64 + 64 = 128)					64 x 64
Conv9_1	64 x 64	128	64	3 x 3	1	1	64 x 64
ReLU9_1							
MFM8	64 x 64	64	64	3 x 3	1	1	64 x 64
ReLU9_2							
Conv9_2	64 x 64	64	3 (RGB)	1 x 1	1	0	64 64

B. Discriminator Network

During the training phase, the discriminator network is exclusively employed. The discriminator network employed is based on the design of [24], which has previously been used for satellite imaging problems. Table 2 shows the architecture in more detail. On the discriminator network, the entire LReLU activation function has a negative slope of 0.2.

Table 2. Discriminator network

Layer	Input Resolution (Pixel)	Filter Input	Filter Output	Kernel Size	Stride	Padding	Output Resolution (Pixel)
Conv 64	64 x 64	3 (RGB)	64	4 x 4	2	1	32 x 32
LReLU							
Conv 128	32 x 32	64	128	4 x 4	2	1	16 x 16
BatchNorm							
LReLU							
Conv 256	16 x 16	128	256	4 x 4	2	1	8 x 8
BatchNorm							
LReLU							
Conv 512	8 x 8	256	512	4 x 4	2	1	4 x 4
BatchNorm							
LReLU							
Conv 1	4 x 4	256	512	4 x 4	2	0	1 x 1
Sigmoid							

C. Network Light CNN-29v2

The LightCNN-29v2 network from HR facial image recognition research [12] is shown in Fig. 3. The discriminator network is helped by LightCNN's design. The network is initially trained as a classifier using HR image data. Furthermore, the FC 2 is not used. The FC1 is the only one that is utilized. The Cosine matrix loss and magnitude are calculated using the FC 1 output of 256 neurons to get the feature vectors [25]. The foundation for this technique is identity-aware research [8].

Details of the Light CNN-29v2 network shown in Fig. 3 can be seen in Table 3. The components that make up the LightCNN-29v2 architecture, as shown in Fig. 3, are composed of several modules such as MFM in Fig. 4, a combination of Max pooling with Average pooling, ResBlock in Fig. 5, and Group in Fig. 6. The ResBlock module in Fig. 5 consists of two MFM modules arranged in series with a skip connection. The ResBlock Fig. 5 module parameters are MFM for kernel size 3, stride 1, and padding 1, respectively.

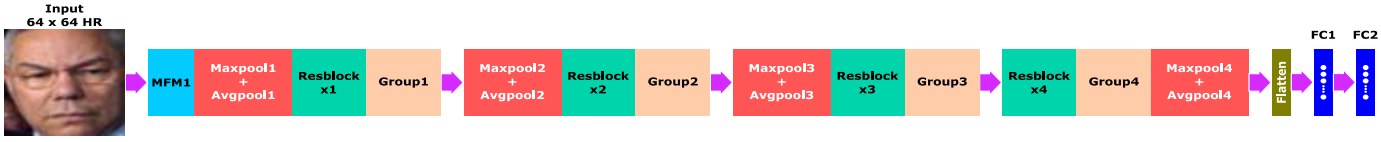


Fig. 3 Network LightCNN-29v2

Table 3. Detail architecture LightCNN-29v2

Layer	Filter Input	Filter Output	Kernel Size	Stride	Padding
MFM1	3	48	5 x 5	1	2
Maxpool1 + Avgpool1			2 x 2	2	0
Resblock x 1	48	48			
Group1	48	96	3 x 3	1	1
Maxpool2 + Avgpool2			2 x 2	2	0
Resblock x 2	96	96			
Group2	96	192	3 x 3	1	1
Maxpool3 + Avgpool3			2 x 2	2	0
Resblock x 3	192	192			
Group3	192	128	3 x 3	1	1
Resblock x 4	128	128			
Group4	128	128	3 x 3	1	1
Maxpool4 + Avgpool4			2 x 2	2	0
Flatten					
FC1	2048	256			
FC2	256	Number of classes			

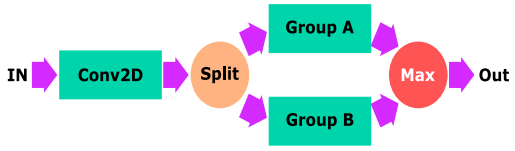


Fig. 4 Inside of Max-Feature-Map (MFM)

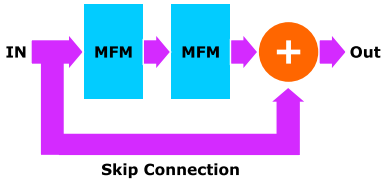


Fig. 5 Inside of resblock

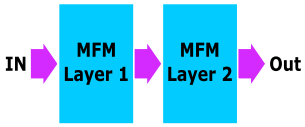


Fig. 6 Inside of module group

D. Network Light CNN-29v2

The difference between the discriminator network predictions that take input from the HR picture and the SR image is calculated using the loss scalar value. This study combines three loss functions to update the U-Net network using identity-aware research [8]. So, the classification accuracy of the SR picture can be improved. The angle distance in the feature space is minimized using the cosine loss function Eq. 1. To minimize norm-diff in feature space, use the magnitude loss function Eq. 2

The norm-diff distance in the feature space is minimized using the magnitude loss function Eq. 2. In the feature space, Eq.2 is employed to minimize the norm-diff distance. The loss of Binary Cross-Entropy (BCE) [22] in Eq. 3 is used to determine the loss difference between the expected outcomes of SR and HR pictures. In Eq. 1, F_i^{SR} and F_i^{HR} are 256 feature vector output from FC 1. Fig. 3. F_i^{SR} is the output feature from the results of SR images input. F_i^{HR} Is output feature vector from the HR image input. In Eq. 3 and Eq. 4, the symbol m is the batch size, y_i is the actual label value. $p(\hat{y}_i)$ Is the probability value of the prediction result. The scalar value of $L_{i,Cosine}$, $L_{i,Magnitude}$, and $L_{i,BCE}$ are then summed together with each loss given a weight value α, β_1, β_2 as shown in Eq. 4.

$$L_{i,Cosine} = 1 - cos\theta, where$$
$$cos\theta = \frac{(F_i^{SR})^T(F_i^{HR})}{\|F_i^{SR}\|_2\|F_i^{HR}\|_2} \quad (1)$$

$$L_{i,Magnitude} = \|norm(F_i^{SR}) - norm(F_i^{HR})\|_2 \quad (2)$$

$$L_{i,BCE} = -\frac{1}{m}\sum_{i}^{m}[y_i \log(p(\hat{y}_i)) + (1 - y_i)\log(1 - p(\hat{y}_i))] \quad (3)$$

$$L_{total} = \frac{1}{m}\sum_{i=1}^{m}\alpha L_{i,BCE} + \beta_1 L_{i,Cosine} + \beta_2 L_{i,magnitude} \quad (4)$$

III. EXPERIMENT

We utilize a laptop with an AMD Ryzen 7 CPU which runs on 2.9 GHz, and NVIDIA RTX 2060 GPU with 6 GB of VRAM and 16 GB of DDR4 RAM in this experiment. PyTorch is the deep learning framework used.

A. Dataset

The dataset used was Labeled Face in the Wild (LFW) [26]. To simulate face picture identification and SR techniques, the well-known LFW dataset is used as a reference for evaluation and benchmarking. In LFW labels or courses, there are 5,749 different people. Each class has an uneven distribution of face photos. HR has a resolution of 255 × 255 pixels and a size of HR. Images of the artist taken with various camera equipment and retrieved from the internet make up the dataset. The state of images includes changes in postures, emotions, lighting, and occlusion. Only 22 different label classes were employed in this experiment, which each have at least 10 face photos. The Multi-Task Cascaded Convolutional Networks (MTCNN) approach pre-process facial pictures by cutting directly on the face area [27]. We manually choose the image because the MTCNN detection and cropping findings still contain false situations. According to the method used by [28]. The dataset is then separated into 80 per cent training data, 10% validation data, and 10% test data. After that, using horizontal split augmentation, all of the training data was reproduced.

B. Pre-Train Preparation Network LightCNN-29v2

The initial training procedure was conducted in this work using the LightCNN-29v2 architecture [12]. LFW 64 x 64 pixels HR picture data is used in the training procedure. Stochastic Gradient Descent (SGD) was used to optimize the training process, with a learning rate of 0.003, the momentum of 0.9, and weight decay of 0.001. Cross-Entropy Loss is a loss function. The batch size is 32, and 100 epochs are the number of iterations. Because feature extraction will be employed later, the pre-train preparation step for the LightCNN-29v2 model is completed by putting the feature vector output on the FC 1 layer. In the LightCNN-29v2 training procedure, Fig. 7 shows a sample of 25 epochs. The

sample details of validation and loss values of Fig. 7 during training processes can be seen in Table 4. The validation accuracy of LightCNN-29v2 in the training phase is shown in Fig. 8, with the detailed validation accuracy can be seen in Table 5. The validation accuracy is known to be the maximum constant at 0.9432 at the 25th epoch, as shown in Fig. 8.

Fig. 7 Sample of train and validation losses value during training lightcnn-29v2

Table 4. Train and validation losses detail from fig.7 during training lightcnn-29v2

Epoch	Train Loss	Validation Loss
1	2,6009	2,5922
2	1,9483	2,0854
3	1,2806	1,4664
4	0,6898	0,9459
5	0,4614	0,7734
6	0,3239	0,6338
7	0,1242	0,5565
8	0,0931	0,4962
9	0,1190	0,7247
10	0,1218	0,5560
11	0,0355	0,5647
12	0,0348	0,6027
13	0,0242	0,4917
14	0,0793	0,5686
15	0,0234	0,5801
16	0,0179	0,4815
17	0,0283	0,6857
18	0,0157	0,4090
19	0,0145	0,4230
20	0,0018	0,3538
21	0,0002	0,3569
22	0,0001	0,3646
23	0,0001	0,3696
24	0,0001	0,3736

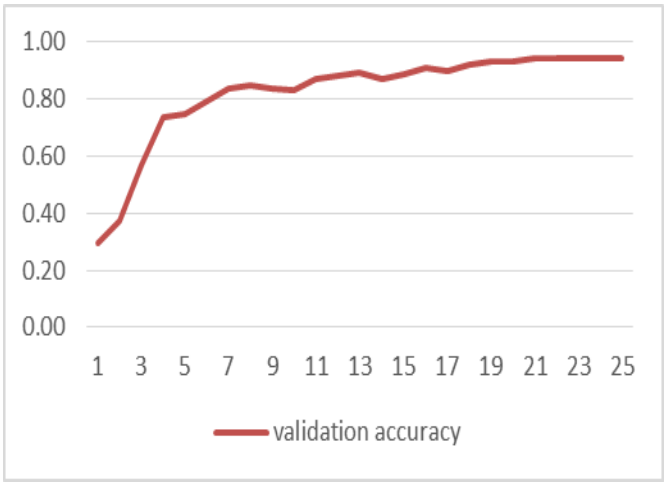


Fig. 8 The validation accuracy during training lightcnn-29v2

Table 5. The details of validation accuracy from fig. 8 during training lightcnn-29v2.

Epoch	Validation Accuracy
1	0,2938
2	0,3763
3	0,5670
4	0,7371
5	0,7474
6	0,7938
7	0,8402
8	0,8505
9	0,8351
10	0,8299
11	0,8711
12	0,8814
13	0,8918
14	0,8711
15	0,8866
16	0,9124
17	0,8969
18	0,9227
19	0,9330
20	0,9330
21	0,9433
22	0,9433
23	0,9433
24	0,9433
25	0,9433

C. Training MFM CMU-Net

The parameters represented to train the CMU-Net MFM network were Adam optimizer learning rates of 0.0002 and beta values for the training network generator in Table 1. and discriminator in Table 2. Respectively 0.5 and 0.999. The discriminator network utilizes BCE loss as its loss function (Eq. 3). The Light CNN-29v2 model (architecture in Fig. 3) employs SGD for the pre-train optimizer, with a learning rate of 0.003, a momentum of 0.9, and a weight decay of 0.001. In this section, Fully Connected (FC) layer 2nd of the pre-train model LightCNN-29v2 is not employed. The Light CNN-29v2 pre-train uses the Cosine matrix loss (Eq. 1) and magnitude loss (Eq. 2) functions. The total number of epochs is 100 epochs for the training process. The CMU-Net MFM training procedure consists of three steps: first, train the discriminator network using HR (High Resolution) picture input, then calculate the loss value against the real label or 1 with the BCE loss function, and finally utilize the loss value to update the weight of the discriminator network.

The discriminator network is then trained by feeding SR or false images generated by the SR network generator (details of the network generator can be seen in Table 1). The loss value is then computed against the false label or 0 and used to update the weight of the discriminator network. The discriminator network differentiated between the input HR and SR pictures until the second phase of the discriminator network training method. The training data loss value from sample epochs 91 to and the value of BCE loss are given in Fig. 9. The orange line in Fig.9 represents the real label's loss value from the discriminator network's input image HR. The loss value when the discriminator network is trained using the input of the SR picture from the network generator output with the fake label is shown by the grey line in Fig. 9. The detail of discriminator loss value for input images HR and SR during the training process can be seen in Table 6.

The third stage uses the LightCNN-29v2 pre-train model to train the network generator up to the discriminator, especially by supplying input to the SR image discriminator network without the detach process and loss computations. BCE loss is utilized, as well as label false or decimal 0 as a label. Fig. 10 shows the result from the output network discriminator when receiving SR image, and the final output BCE performed to calculate loss when the real label is given. The detail value of Fig. 10 is shown in Table 7. The pre-train LightCNN-29v2 model is alternately fed SR and HR pictures. The difference in loss between SR and HR pictures is then determined using Cosine Matric loss and Magnitude loss as the output feature vector. The output of BCE loss, Cosine matrix loss, and loss magnitude calculations are summed together, and the entire network is updated. The total loss Eq. 4 is calculated in Fig. 10. α=1 β_1= 10 and β_2=0.1, the constant value on Eq. 4 is that the overall loss value for the MFM-CMU-Net network training procedure in Step three looks to be higher, reaching between 4.8 and 5.2. The BCE Loss was used to evaluate the input SR picture with the label position of real or 1 that was previously trained with the label fake or 0. Each epoch has its training method for the first, second, and third steps. Pre-trained network generator models with the most outstanding validation accuracy are the only ones we keep. The graph with the maximum validation accuracy in the 98th epoch, with a score of 0.8247, is shown in Fig. 11 with the detail in Table 8.

Fig. 9 The output loss value from discriminator network during training of MFM CMU-Net on the first and second steps

Table 6. The output loss discriminator network from Fig. 9 with input image HR and SR

epoch	loss HR with label real	loss SR with label fake
91	0,1933	0,1947
92	0,2158	0,2176
93	0,2089	0,2079
94	0,1975	0,1955
95	0,2394	0,2352
96	0,3345	0,3163
97	0,2010	0,1936
98	0,2023	0,2015
99	0,2166	0,2130
100	0,2345	0,2295

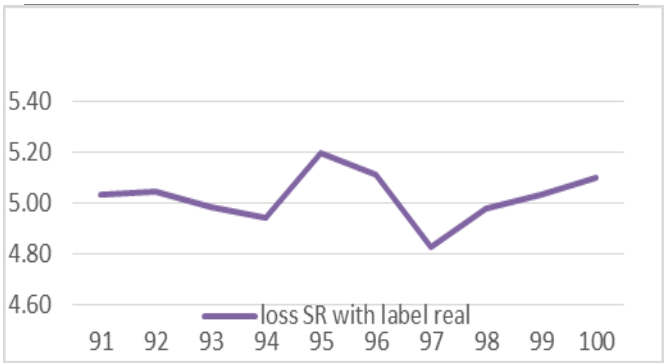


Fig. 10 The Total output loss combination of BCE, cosine, and magnitude loss during training process MFM CMU-Net on third step.

Table 7. The output loss of network discriminator with input SR and Real Label from Fig. 10

epoch	loss SR with label real
91	5,0321
92	5,0459
93	4,9828
94	4,9413
95	5,1998
96	5,1118
97	4,8270
98	4,9792
99	5,0311
100	5,1019

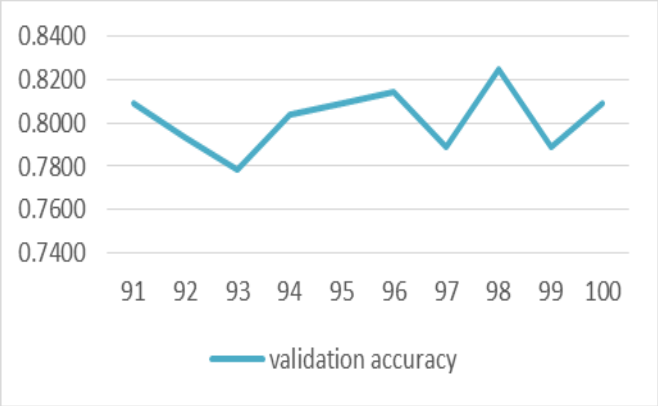
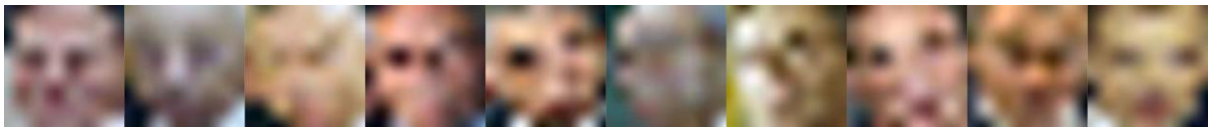


Fig. 11 Validation accuracy using image SR output from MFM CMU-Net that feeds into Lightcnn-29v2 as a classifier

Table 8. The detail of validation accuracy from Fig. 11

epoch	validation accuracy
91	0,8093
92	0,7938
93	0,7784
94	0,8041
95	0,8093
96	0,8144
97	0,7887
98	0,8247
99	0,7887
100	0,8093

The findings of the pre-train network generator model are then applied to the LFW test dataset to create SR pictures. When compared to bicubic interpolation, the SR results are superior. The outcome of bicubic interpolation from 8 x 8 pixels to 64 x 64 pixels with still blurry image circumstances is shown in Fig. 12.a. The MFM CMU-Net approach developed in this work was used to create an SR image in Fig. 12. b. The original picture, or HR, is shown in Fig. 12. c. Table 9 shows that the proposed method outperforms bicubic in PSNR and SSIM. The Light CNN-29v2 pre-train model, which had previously been trained using HR pictures, was used to evaluate the SR image for facial identification. The classification findings in Table 10 reveal that the CMU-Net MFM has a test accuracy score of 78.45% higher when employing merely Bicubic.



(a)

(b)

(c)

Fig. 12 (a) Bicubic interpolation (b) MFM CMU-Net and (c) Images reference

Table 9. PSNR and SSIM scores

Model	PSNR	SSIM
Bicubic Interpolation 8 x 8 pixel to 64 x 64 pixel	19.245 db	0.4548 db
MFM CMU-Net (proposed)	**20.4126 db**	**0.5895 db**

Table 10. Test accuracy

Model	Accuration
Bicubic Interpolation 8 x 8 pixel to 64 x 64 pixel	40.33 %
MFM CMU-Net (Proposed)	**78.45 %**

IV. CONCLUSION

This study proposes an SR facial image approach that may enhance classification accuracy from an 8 x 8-pixel LR picture input to a 64 x 64 pixels SR image input, using a U-Net-based network generator. The U-Net that was utilized has been adjusted to perform SR images of 64 x 64 pixels only. An MFM module is also included in the U-Net design. A skip connection has been added to the MFM module. It is also paralleled in the discriminator network with the LightCNN-29v2 pre-train model, which serves as feature extraction. The cosine matrix loss and loss magnitude are calculated using the outcome of feature extraction. The three losses are then combined to update the complete network, notably BCE, Cosine, and Magnitude. Compared to utilizing merely bicubic interpolation, the suggested CMU-Net MFM architecture can generate realistic pictures and boost PSNR scores, SSIM, and classification accuracy.

REFERENCES

[1] T. Gwyn, K. Roy, and M. Atay., Face Recognition Using Popular Deep Net Architectures: A Brief Comparative Study, Future Internet, 13(7) (2021) doi: 10.3390/fi13070164.

[2] Asma Shaikh, Aditi Mhadgut, Apurva Prasad, Bhagyashree Shinde, and Rohan Pandita, Two-way Credit Card Authentication With Face Recognition Using Webcam, International Journal of Engineering Trends and Technology (IJETT), 67(5) (2019) 160–162.

[3] O. A. Aghdam, B. Bozorgtabar, H. K. Ekenel, and J.-P. Thiran, Exploring factors for improving low-resolution face recognition, in 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), (2019) 2363–2370. doi: 10.1109/CVPRW.2019.00290.

[4] S.-C. Lai, M. Kong, K.-M. Lam and D. Li, High-Resolution Face Recognition Via Deep Pore-Feature Matching, in 2019 IEEE International Conference on Image Processing (ICIP), (2019) 3477–3481. doi: 10.1109/ICIP.2019.8803686.

[5] M. Rouhsedaghat, Y. Wang, S. Hu, S. You, and C.-C. J. Kuo, Low-resolution face recognition in resource-constrained environments, Pattern Recognition Letters, 49 (2021) 193–199. doi: 10.1016/j.patrec.2021.05.009.

[6] Z. Zhang, X. Pan, S. Jiang, and P. Zhao, High-quality face image generation based on generative adversarial networks, Journal of Visual Communication and Image Representation, 71 (2020) 1178–1182. doi: 10.1016/j.jvcir.2019.102719.

[7] S. D. Indradi, A. Arifianto, and K. N. Ramadhani, "Face image super-resolution using inception residual network and GAN framework, 2019 7th International Conference on Information and Communication Technology, ICoICT , 1 (2019) 1-6. doi: 10.1109/ICoICT.2019.8835253.

[8] J. Chen, J. Chen, Z. Wang, C. Liang, and C. Lin, Identity-Aware Face Super-Resolution for Low-Resolution Face Recognition, IEEE Signal Processing Letters, 9908(c) (2020) 1–5. doi: 10.1109/LSP.2020.2986942.

[9] D. Khaledyan, A. Amirany, K. Jafari, M. H. Moaiyeri, A. Z. Khuzani, and N. Mashhadi, Low-Cost Implementation of Bilinear and Bicubic Image Interpolation for Real-Time Image Super-Resolution, in 2020 IEEE Global Humanitarian Technology Conference (GHTC), (2020) 1–5. doi: 10.1109/GHTC46280.2020.9342625.

[10] X. Li, N. Dong, J. Huang, L. Zhuo, and J. Li, A discriminative self-attention cycle GAN for face super-resolution and recognition, IET Image Processing, 15(11) (2021) 2614–2628. doi: 10.1049/ipr2.12250.

[11] C.-C. Hsu, C.-W. Lin, W.-T. Su, and G. Cheung, Sigan: Siamese generative adversarial network for identity-preserving face hallucination, IEEE Transactions on Image Processing, 28(12) (2019) 6225–6236. doi: 10.1109/TIP.2019.2924554.

[12] X. Wu, R. He, Z. Sun, and T. Tan, "A light CNN for deep face representation with noisy labels," IEEE Transactions on Information Forensics and Security, 13(11) (2018) 2884–2896, 2018, doi: 10.1109/TIP.2018.2883743.

[13] J. He, J. Zheng, Y. Shen, Y. Guo, and H. Zhou, Facial Image Synthesis and Super-Resolution With Stacked Generative Adversarial Network, Neurocomputing, 402 (2020) 359–365. doi: 10.1016/j.neucom.2020.03.107.

[14] S. Ge, S. Zhao, C. Li, and J. Li, Low-resolution face recognition in the wild via selective knowledge distillation, arXiv, 28(4) (2018) 2051–2062. doi: 10.1109/TIP.2018.2883743.

[15] T. Lu, X. Chen, Y. Zhang, C. Chen, and Z. Xiong, SLR: Semi-Coupled Locality Constrained Representation for Very Low-Resolution Face Recognition and Super-Resolution, IEEE Access, 6 (2018) 56269–56281. doi: 10.1109/ACCESS.2018.2872761.

[16] K. Grm, W. J. Scheirer, and V. Štruc, Face hallucination using cascaded super-resolution and identity priors., IEEE Transactions on Image Processing, 29 (2019) 2150–2165. doi: 10.1109/TIP.2019.2945835.

[17] O. Ronneberger, P. Fischer, and T. Brox, U-net: Convolutional networks for biomedical image segmentation, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 9351 (2015) 234–241. doi: 10.1007/978-3-319-24574-4_28.

[18] P. Isola, J. Y. Zhu, T. Zhou, and A. A. Efros, Image-to-image translation with conditional adversarial networks, Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, (2017) (2017) 5967–5976. doi: 10.1109/CVPR.2017.632.

[19] M. Lucic, K. Kurach, M. Michalski, S. Gelly, and O. Bousquet, Are gans created equal? a large-scale study, arXiv preprint arXiv:1711.10337, (2017).

[20] M. M. Majdabadi and S. B. Ko, MSG-CapsGAN: Multi-Scale Gradient Capsule GAN for Face Super-Resolution, 2020 International Conference on Electronics, Information, and Communication, ICEIC (2020) 10–12. doi: 10.1109/ICEIC49074.2020.9051244.

[21] Nuno-Maganda and M. O. Arias-Estrada, Real-time FPGA-based architecture for bicubic interpolation: an application for digital image scaling, in 2005 International Conference on Reconfigurable Computing and FPGAs (ReConFig'05), (2005) 1. doi: 10.1109/RECONFIG.2005.34.

[22] X. Wei et al., Building Outline Extraction Directly Using the U2-Net Semantic Segmentation Model from High-Resolution Aerial Images and a Comparison Study, Remote Sensing, 13(16) (2021). doi: 10.3390/rs13163187.

[23] Sunil Pandey, Naresh Kumar Nagwani, and Shrish Verma, Analysis and Design of High-Performance Deep Learning Algorithm: Convolutional Neural Networks, International Journal of Engineering Trends and Technology (IJETT), 69(6) (2021) 216–224.

[24] Y. Shi, Q. Li, and X. X. Zhu, Building Footprint Generation Using Improved Generative Adversarial Networks, IEEE Geoscience and Remote Sensing Letters, 16(4) (2019) 603–607. doi: 10.1109/LGRS.2018.2878486.

[25] Yasser Mohammad Al-Sharo, Amer Tahseen Abu-Jassar, Svitlana Sotnik, and Vyacheslav Lyashenko, Neural Networks As A Tool For Pattern Recognition of Fasteners, International Journal of Engineering Trends and Technology (IJETT), 69(10) (2021) 151–160.

[26] 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, Oct. (2008).

[27] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, Joint Face Detection and Alignment Using Multi-task Cascaded Convolutional Networks, IEEE Signal Processing Letters, 23(10) (2016) 1499–1503. doi: 10.1109/LSP.2016.2603342.

[28] A. Rai, V. Chudasama, K. Upla, K. Raja, R. Ramachandra, and C. Busch, ComSupResNet: A compact super-resolution network for low-resolution face images, in 2020 8th International Workshop on Biometrics and Forensics (IWBF), (2020) 1–6. doi:10.1109/IWBF49977.2020.9107946.