Mathematical Biosciences and Engineering

MBE, 18(4): 3699–3717.
DOI: 10.3934/mbe.2021186
Received: 17 February 2021
Accepted: 15 April 2021
Published: 28 April 2021

http://www.aimspress.com/journal/MBE

Research article

Identity preserving multi-pose facial expression recognition using fine tuned VGG on the latent space vector of generative adversarial network

R Nandhini Abirami and P M Durai Raj Vincent*

School of Information Technology, Vellore Institute of Technology, Vellore, Tamilnadu, India

* **Correspondence:** Email: pmvincent@vit.ac.in.

Abstract: Facial expression is the crucial component for human beings to express their mental state and it has become one of the prominent areas of research in computer vision. However, the task becomes challenging when the given facial image is non-frontal. The influence of poses on facial images is alleviated using an encoder of a generative adversarial network capable of learning pose invariant representations. State-of-art results for image generation are achieved using styleGAN architecture. An efficient model is proposed to embed the given image into the latent vector space of styleGAN. The encoder extracts high-level features of the facial image and encodes them into the latent space. Rigorous analysis of semantics hidden in the latent space of styleGAN is performed. Based on the analysis, the facial image is synthesized, and facial expressions are recognized using an expression recognition neural network. The original image is recovered from the features encoded in the latent space. Semantic editing operations like face rotation, style transfer, face aging, image morphing and expression transfer can be performed on the image obtained from the image generated using the features encoded latent space of styleGAN. L_2 feature-wise loss is applied to warrant the quality of the rebuilt image. The facial image is then fed into the attribute classifier to extract high-level features, and the features are concatenated to perform facial expression classification. Evaluations are performed on the generated results to demonstrate that state-of-art results are achieved using the proposed method.

Keywords: computer vision; deep learning; facial expression recognition; convolutional neural network; human-robot interaction; generative adversarial network

Abbreviations: CNN: Convolutional neural network; GAN: Generative adversarial network; StyleGAN: Style based generative adversarial network; DNN: Deep neural network; AI: Artificial intelligence; SIFT: Scale invariant feature transform; LBP: Local binary pattern; HOG: Histogram of oriented gradients; FFHQ: Flickr-Faces-HQ dataset; ResNet: Residual neural network; CK: Cohn-Kanade dataset; RNN: Residual neural network; JAFFE: The Japanese female facial expression; VGG: Visual geometry group; AdaIn: Adaptive instance normalization.

1. Introduction

Cognitive science and affective computing are the two critical areas demanding significant research for expression analysis of facial images[1–3]. Human beings convey their emotional states using facial expressions. Hence, determining these facial expressions has become crucial in emotion robots, non-verbal human behavior [4], human-robot interaction [5–8], and sentiment analysis [9]. However, facial expression recognition remains challenging in the wake of pose variations under uncontrolled circumstances. This work's ultimate goal is to perform facial expression recognition by considering seven different facial expressions like angry, surprise, happy, disgust, sad, neutral and fear. Several works have already been established in facial expression recognition. Still, most of the works administer hand-engineered features like HOG [10,11] SIFT [12], and LBP [13] for facial feature extraction and considered only frontal views of the faces for emotion recognition. It is computationally challenging and complex to use these hand designed feature extraction techniques for facial emotion recognition.

This work proposes a DNN model for facial emotion recognition. DNN is a multilayer perceptron with several hidden layers built between the input and output layers. With the advancements in the regularization and optimization techniques, DNN can learn large and complex data representations. With additional research and fine-tuning of DNN, supervised learning models, namely CNN and RNN, and other unsupervised learning models, namely Autoencoders and Boltzmann machine, were developed [14,15]. Generative models are considered to be one of the important classes of DNN [16].

Generative adversarial network are generative models that estimate the density function of the data distribution. GAN is built with two adversarial networks, namely generator and discriminator. The generator and the discriminator are designed to play a minimax game. The generator generates realistic samples to deceive the discriminator, which classifies the real and the fake samples. The performance of the generator and the discriminator constantly improves with training and each of them trying to win the minimax game [17]. When the discriminator cannot identify the real and the fake samples, the generator is said to have learned the data distribution.

Evolution of GAN in 2014 have opened new opportunities for many state-of-art applications including image to image translation [18–20], text to image translation [21,22], image to text translation, and facial expression recognition [23,24]. In all these use cases generator generates realistic images and the discriminator identifies the fake images. The idea behind GANs is to perform adversarial training to learn the representations from a latent space and map them to real data distribution. Many new architectures of GAN models were proposed based on the basic GAN architecture. Since GAN's evolution, human faces generated by various GAN models have seen progressive improvement in the quality and resolution. StyleGAN is proposed based on GAN architectures and it is a state-of-art GAN model for generating high-resolution photorealistic images [25,26].

Latent space vectors are poorly explored in existing researches. In contrast, this work explores the latent space vector of styleGAN to synthesize the non-frontal facial image retaining its expression and identity. The latent space concept has enormous uses in deep learning, from learning the features of the data to simplifying the representation of data for finding the data patterns. All the important information necessary to represent the data is hidden in the latent representation. In other words, the model learns features of data to simplify the representation for easier analysis. It makes it easier to understand the data points' structural similarities or patterns by analyzing data in the latent space. As the latent representations have the data in its compressed form and carry only the important information, the processing is faster when compared to other classical approaches. Enough research was not performed on connecting the latent representations of images to semantic attributes for editing the image. The proposed model interprets the latent space of styleGAN, which is trained for face synthesis. This work is split up into two phases wherein the synthesis of facial image is performed using a well-trained GAN latent space vector. The second phase involves modeling a neural network for facial expression recognition. This work

(i) Explores the latent space of styleGAN and identifies the relationship between the latent representations and semantic attributes of the output.

(ii) Rigorously analyses the capability of GANs to map the latent vectors to high-resolution images.

(iii) Presents an efficient model for generating non-frontal facial images preserving the identity and expression of the face.

(iv) It provides insight to the researchers about how a random distribution is mapped to a high-quality semantic image and how to interpret the semantics of latent space and use the latent space vectors for various applications.

The remainder of the paper is organized as follows. In Section II, existing works relevant to emotion recognition and GAN are presented. In Section III, the architecture of the proposed method is described. In Section IV, experimental results are discussed with performance analysis. Finally, in Section V, the paper is concluded with a discussion about future work.

2. Related works

Existing works analyses on generating high-resolution images from ground truth [27–29], however, very few works exist on analyzing the capability of GANs concerning latent space. Radford et al. [30] were the first to propose that GANs learn various semantic attributes in the hidden latent space. Mirza M et al. [31] proposed a model to generate images using disentangled latent vectors and labeled attributes. This model is extended with a customized loss function and semantic attributes to improve the synthesis quality [32–34]. Arvanitidis et al. [35] proposed a model to vary the output smoothly through latent space interpolation. Some works were also performed in the reverse direction by generating the latent space from the image space [36,37]. Wang et al. [38] performed facial expression recognition using an unsupervised domain adaptation method with four datasets, namely FER2013, CK+, MMI, and JAFFE. Seven different emotions, anger, happy, fear, sad, disgust, surprise and neutral, were recognized using the developed model. This model for expression recognition was built using Alexnet and VGG11. The facial images from CK+, MMI, and JAFFE datasets are cropped, while the images from FER2013 were resized to 224 × 224 as the original images were too small, measuring 48 × 48. Stochastic gradient descent was used during training. Zhang et al. [39] proposed a feature learning model based on DNN for facial expression recognition. The proposed method extracted features from facial images using SIF method. A feature matrix is arrived using the extracted SIFT features and passed as input to DNN model for expression classification. The DNN model explores the relationship between the SIFT and their semantic features. The proposed model learns the features corresponding to facial expressions.

Yang et al. [40] extracted features suitable for classifying facial expressions using a weighted mixture DNN model. Rotation rectification, data augmentation, and face detection are implemented on the input data. The proposed model processed the grayscale images and LBP facial images. Features of the images are extracted using the VGG16 model. Features of the LBP image are extracted using CNN. The models' outputs are combined in a weighted manner, and the classification is performed using softmax. Kim et al. [41] developed a facial expression recognition system using deep hierarchical learning. The proposed model utilizes two networks: appearance feature-based and geometric feature-based networks, to extract holistic features and coordinate action units. An autoencoder is designed to generate a neutral expression facial image. Dynamic facial features are extracted between the emotional and neutral expression facial image. Zhang et al. [42] used a deep identity network for identifying faces. A deep learning framework based on local facial patches and multi-scale global images was proposed for facial expression recognition. The proposed model localized the foreground image from the background image. Face part patches are generated with local and global identity information. The generated face patches are fed into CNN to perform facial expression classification.

Ferreira et al. [43] proposed a DNN architecture with loss functions based on the fact that expressions are associated with facial muscles' movement. The loss function regularizes the learning process to make the proposed DNN learn features that are specific to an expression. The model identifies the face components, namely nose, eyes, eyebrows and mouth and expression wrinkles to recognize the facial expression. Also, the model is also capable of learning expression-specific features and facial relevance maps. González-Lozoya et al. [44] improved generalization in facial expression recognition by fusing the instances extracted from different facial databases. The proposed method is capable of recognizing micro-expressions. Facial expression recognition is performed using face detection from the facial image, facture extraction using CNN and modeling. In a nutshell, the proposed model is a prototype system for facial expression recognition and micro-expression recognition for analyzing videos.

Deng et al. [6] proposed a conditional GAN-based approach. The proposed approach individually controls the facial expression. It simultaneously learns the generative and discriminative representations. Similarly, Cai et al. [23] proposed a Condition GAN-based approach to reduce the inter-subject variations for expression recognition from facial images. Yang et al. [45] proposed a feature separation model for facial expression recognition tasks. The feature separation is achieved through partial feature exchange and various constraints. Liong et al. [46] proposed four steps: facial landmarks annotations, optical flow guided image computation, feature extraction, and emotion class categorization. Here, GAN is used to perform data augmentation to generate more image samples. Wu et al. [47] proposed a Cascade Expression Focal GAN to perform progressive facial expression editing with local expression focuses. This approach preserved identity-related features and details around the nose, eyes and mouth.

The current work extracts the pose component, identity component and expressive component from the facial image. The extracted expressive component is used to perform facial expression recognition. This work exploits the latent representation of the facial image to analyze the semantic contents of the image. The model identifies the relationship that exists between the latent vector and the semantics of the image. GAN-based face synthesis is performed by controlling the facial attributes and preserving the identity. A new approach is proposed for performing emotion classification based on GAN and fine-tuned VGG19 model. The emotions are classified into seven classes namely anger, fear, sad, happy, disgust, surprise and neutral. Given any multi-pose facial image, the latent vector is obtained and passed through a generator to generate the facial image. Facial features are extracted from the generated image and the non-frontal facial image. The difference is calculated as loss, and gradient descent is applied to reduce the loss. The gradients are backpropagated and images are fine-tuned until the image very close to the input image is obtained. The facial image is then passed through facial expression recognition neural network to perform emotion classification. The expression recognition neural network is a deep CNN model which extracts high-level facial features and predicts the output as a probability of seven classes. The features extracted from the latent representation are concatenated with features extracted by the deep CNN model and facial expression classification is performed.

3. Materials and methods

When a generative model is trained on a dataset, the model discovers the data's underlying structure. Given that the model has discovered the underlying structure, it can be utilized to perform a variety of applications. This work explores the extent to which the latent space interpolation can navigate the visual world, like manipulating an image of a female to look like a male, making an image with neutral expression to smile, face aging and more. The basic science behind encoder-decoder is that an encoder encodes the pixel space into the latent vector space. The decoder decodes the available information from the latent space vector to rebuilt the actual input. The latent space contains the actual input in a compressed version of the actual data at a lower dimension when compared to the pixel space. It has only the information that is required to reconstruct the actual input from the latent space vector. A generator in the GAN architecture exploits the latent space and maps the latent vector to the output [48]. Mapping performed by the generator varies for every epoch. By using the random points in the latent vector space, the generator generates a new image. Figure 1 shows the mapping network that maps the latent space vector to another intermediate vector fed to the image synthesis network.

This work involves exploiting the latent space of styleGAN trained on Flickr Faces High-Quality Dataset (FFHQ). Vector arithmetic operations are performed on the points in the latent space to generate images. The random vector from the latent space is passed as input to the generator model. The size of the latent space and the points are the input samples for the generator. The generator model returns the generated images as output. The number of epochs required for training is arbitrary and it can be increased if the quality of the images generated is to be improved.

Traditionally the generator gets a random noise vector as input. The random noise is fed into a bunch of up-sampling networks until the desired image is generated. In contrast to the traditional approach, the styleGAN generator has a mapping network f as shown in Figure 1, which takes a random sample $z \in Z$ as the input and transforms into an intermediate vector called $w \in W$. The disentanglement observed at the W space is much stronger than the disentanglement observed at Z space. Unlike Z, W is not restricted to a specific distribution and it can better understand the underlying features of the real data. Since the disentanglement in W space is much superior to Z space, attribute editing is far better with W space. The distribution of vector w is not required to be Gaussian, and rather it can be any other distribution.

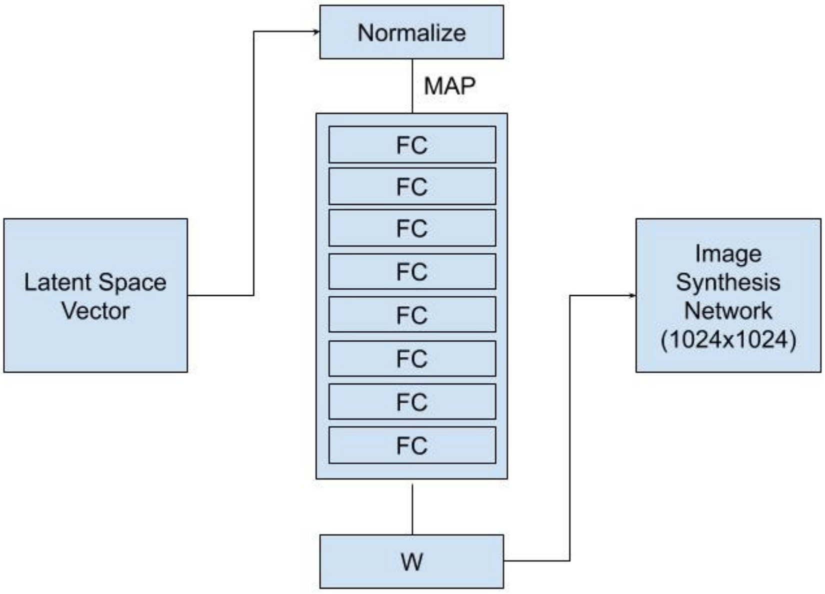


Figure 1. Mapping network.

Hence, the actual generator architecture of styleGAN does not start with a random noise vector, rather starts from a constant vector. This constant vector is optimized during the training. The vector w is plugged into multiple layers of the generative architecture using a blending layer called AdaIN. During training, in addition, the vector w noise is also added to these parameters. The general principle of a generative model is that its latent space learns the underlying structure. The structure learnt by the generative model is unsupervised during the adversarial training process. In order to leverage this structure, the image in the latent space is manipulated instead of manipulating in pixel space. Manipulating the image in pixel space is complicated. To simplify this, the image in latent space is manipulated. The latent vector is determined in the latent space to perform this manipulation for a given query image. Two different methodologies can be adapted to determine the latent vector.

1) Given that the generator model is a fully differentiable neural network, a random latent code is passed through the differentiable generator and generated images. The generated image is compared with the query image by calculating the loss $L2$, which is the pixel difference of the two images. The gradients are backpropagated through the generator and update the latent vector at the generator model's beginning. By applying gradient descent on the pixel lose $L2$, the optimal latent vector is generated. But, considering the $L2$ pixel loss alone will generate an image that is not very close to the query image. The optimization may get stuck in the local minima. To overcome this issue, a trained classifier is used as a lens to look at the image. Both the generated image and the query image are sent through a trained VGG network that was trained to classify ImageNet images. Instead of traveling through the entire VGG network until the classification, the feature vectors are distilled from the fully connected layers. These feature vectors give a high-level semantic representation of the facial image content.

2) Sampled random vectors are passed through generators to generate faces. With the dataset generated, a ResNet (Residual Network) is trained to obtain the image's latent code. Given a query

image, it is passed through a ResNet model, which gives an initial estimate of the latent vector in the styleGAN network. This latent vector is taken and passed through a generator to generate an image. On the generated image, a pre-trained VGG network is used to extract features from the image. Similarly, the VGG network is applied to the query image, and high-level features are extracted from the query image. Loss $L2$ is calculated in the VGG feature space. L2 distance is minimized in the feature space using gradient descent. The gradients are then back-propagated through the generator network until the latent code. During this optimization process, the generator weights are fixed. Only the latent code at the input end is updated. Finally, an optimized image is generated, which is very close to the query image.

This work adopted the second approach to obtain the latent code. The flowchart for the overall approach is represented in Figure 2.

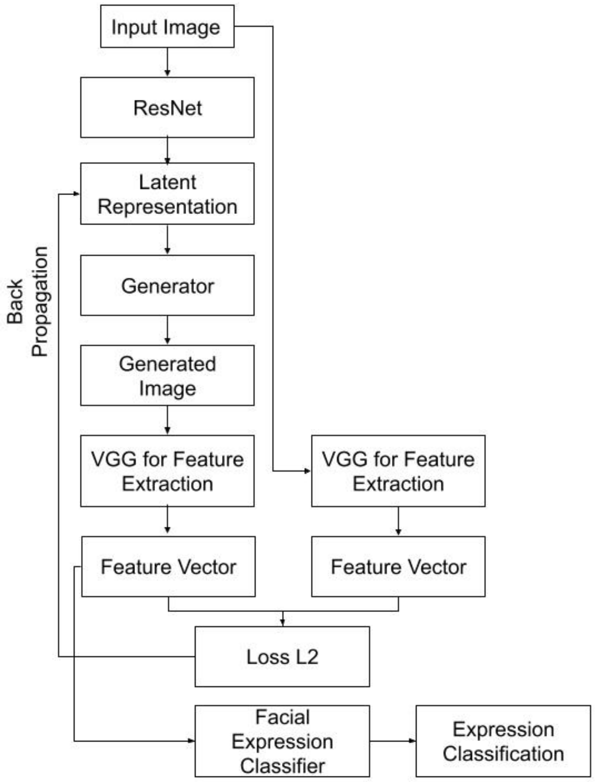


Figure 2. Flowchart for facial expression recognition.

The latent vector is sampled and pass it through the generator to obtain the image. A classifier is applied on the generated on the facial image generated to extract the attributes. A syleGAN latent space has 512 dimensions. Figure 3 shows the schematic illustration of facial expression recognition of a non-frontal facial image. Given a facial image with expression, the corresponding identity, pose and expressive components are extracted through an encoder. The extracted components are concatenated and sent to the decoder. This is performed to distill the expressive component from the facial image to classify the expressions. Facial expression recognition is performed in two phases. The two phases are separated by the dotted rectangular box in the schematic illustration. The first phase involves determining the latent vector of the given query image. The second phase involves extracting high-level facial attributes and facial expression recognition using an expression recognition neural network. Six different basic emotions, namely happy, sad, angry, fear, disgust, and surprise, are recognized. Facial images with no expression are classified as neutral. The drawback with the current work is the proposed model does not handle the images in a noisy environment. The future work may explore facial expression recognition on an unconstrained expression dataset with a noisy environment and explore the real-time applications of facial expression recognition.

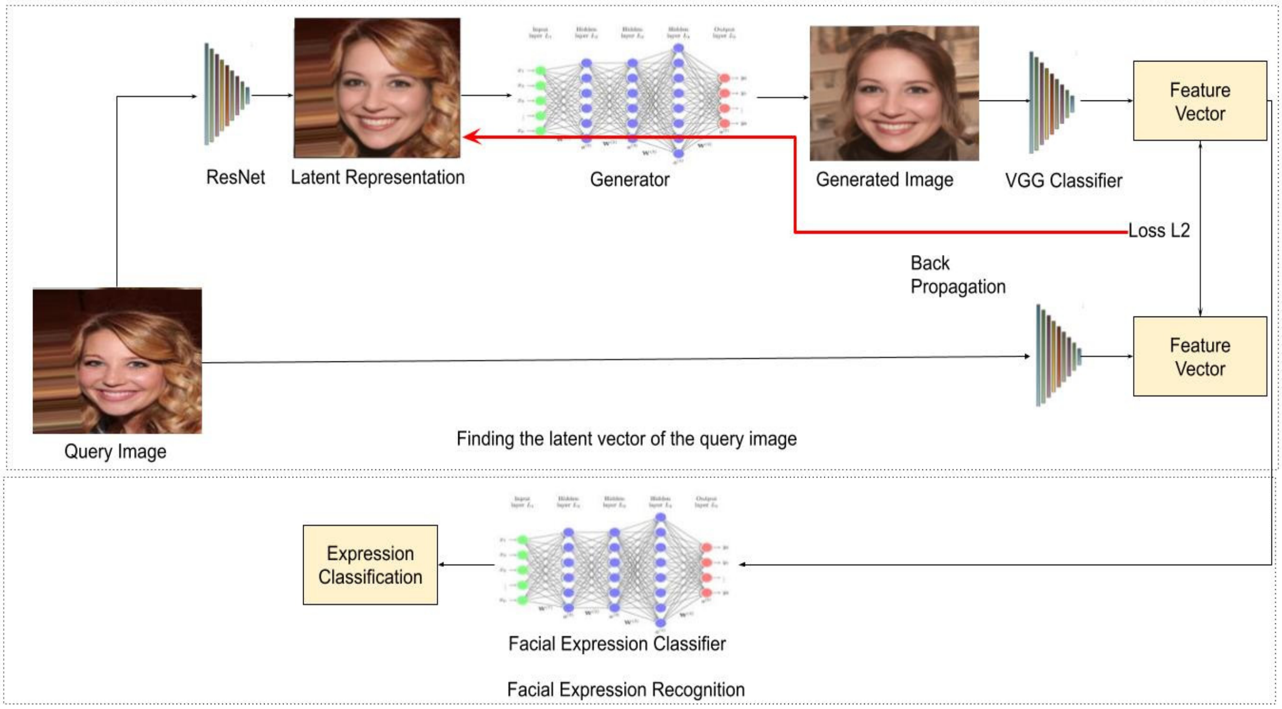


Figure 3. Schematic illustration of facial expression recognition from a non-frontal face. (i) The latent code is obtained using the ResNet. With the initial latent codes, gradient descent optimization is performed. (ii) Using a neural network, facial features are obtained and facial expression is classified.

The overall framework involves extracting the attributes such as pose, expression and identity. Let $X_i$be the input sample, $E_{encoder}$ be the encoder and $D_{decoder}$ be the decoder. The encoder and decoder are built with multiple convolutional layers to map the attributes into the latent vector and to recover the image back from the latent vector, which is represented as,

$$\hat{x} = D_{decoder}\left(E_{encoder}(X_i)\right) = D_{decoder}(Z) \tag{1}$$

Where, X_i is the input sample, $\hat{x}$ is the reconstructed image, $E_{encoder}$ is the encoder that encodes the given input to the latent space, $D_{decoder}$ is the decoder that decodes the latent vector back to the original input and Z is the latent space vector.

In terms of latent space, the ultimate goal is to maneuver latent space to achieve a given image's transformations. The model generator is formulated to map the given latent space to the image space $\boldsymbol{G}: \boldsymbol{Z} \rightarrow \boldsymbol{I}$. $\boldsymbol{Z} \subseteq \mathbb{R}^{\boldsymbol{n}}$, where $\mathbb{R}^n$ denotes n-dimensional latent space. Here, $\boldsymbol{G}$ is the generator, $\boldsymbol{Z}$ is the latent space and $\boldsymbol{I}$ is the image space. $\boldsymbol{z} \in \boldsymbol{Z}$, where z is a latent space vector and $\boldsymbol{i} \in \boldsymbol{I}$, where $\boldsymbol{i}$ is a sample in the image space. Figure 4 shows the generative network, where a random sample from a given distribution is passed to the generator G. The generator generates image i, and the loss L_2 is calculated as the feature-wise difference between the generated distribution and the real distribution.

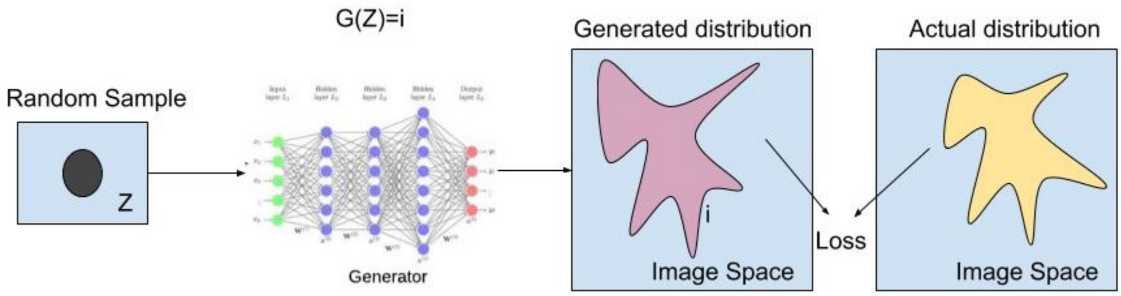


Figure 4. Generative network.

```
Algorithm 1: Training GAN using gradient descent.
Input:
z: Random vector from the given distribution
Z:Input latent space
W: Intermediate latent code
Functions:
Generator $G:Z \rightarrow I$
Loss← $L_2$: Calculate cross-entropy loss.
Output:
I: Image space
i: Sample in the image space
$Y_i$: Synthesized facial image

for number of images in the image space do:
  for each $i \in I$:
    Sample random vector z, from the given distribution
    Generate the facial image using the random vector
                    $G:Z \rightarrow I$
    Calculate feature-wise loss $L_2$ and update the latent vector
    Synthesize the frontal facial image, $Y_i$ retraining the identity and expression of the image
  end
end.
```

3.1. Facial expression recognition

Deep CNN is used for extracting facial features and emotion recognition. Fine-tuned VGG19 architecture is used in the model. The architecture of VGG19 is fine-tuned to optimize the classification performance of deep CNN. The dropout technique is used between the fully connected layer and the final convolutional layer to avoid over-fitting. The final fully connected layer uses softmax for classifying the expressions into one of the seven categories. The softmax activation function output is represented as probabilities corresponding to seven different classes, which sum up to 1. The cross-entropy loss function is used to handle the noise labels and for faster training. Another advantage of using cross-entropy as a loss function is improved generalization capabilities. Figure 5 shows the fine-tuned VGG19 network. The output of the classifier corresponding to maximum probability is determined as the expression of the facial image, represented by the equation,

$$C_{exp} = \left(\max(P(y_i)\right) \tag{2}$$

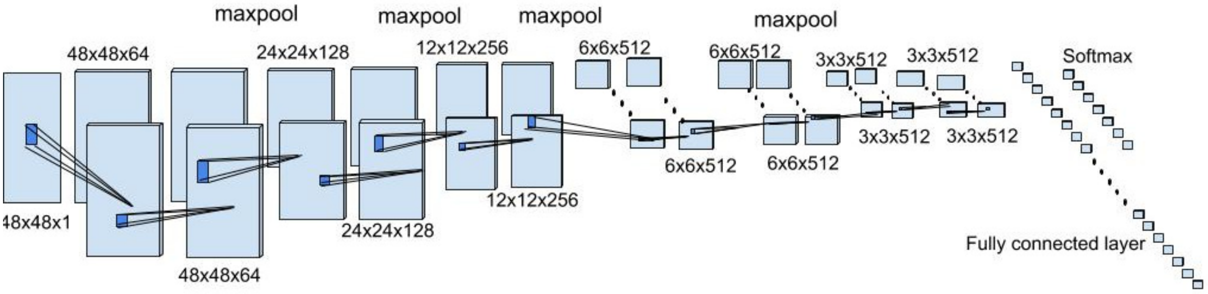
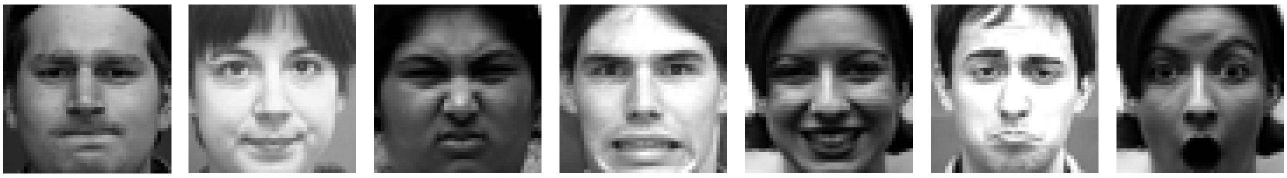


Figure 5. Fine-tuned VGG19 network.

To perform facial expression recognition, CK+ dataset is used, which is released as an extension of the Cohn-Kanade (CK) dataset [49]. The CK+ dataset has 593 image sequences of 123 subjects. Among the 593 sequences of images, 327 sequences have labels containing the emotion. The last three frames of each of the 327 are extracted from the dataset, making 981 facial expressions. The dataset is more robust and reliable as the dataset was obtained under a laboratory environment. Data augmentation is done to expand the database volume. 10-fold cross-validation is performed to improve the accuracy. Seven different facial expressions, namely happy, sad, fear, surprise, disgust, neutral and angry, are classified. Figure 6 shows sample images from the CK+ dataset displaying seven different emotions.



(a) Anger (b) Neutral (c) Disgust (d) Fear (e) Happy (f) Sad (g) Surprise

Figure 6. Sample images from the CK+ dataset displaying seven different emotions.

3.2. Loss function

Cross-entropy is used as a loss function to calculate the loss. The formula to calculate cross-entropy is,

$$CE = -\frac{1}{N}\sum_{i=0}^{N} y_i.\log(\widehat{y_i}) + (1-y_i).\log(1-\widehat{y_i}) \quad (3)$$

The output probabilities of seven classes of a fully connected model are normalized to 1, $y_i \in \{1, \ldots.\mathrm{N}\}$ using softmax activation function. Softmax activation function handles $y_i \in \{1, \ldots\ldots, N\}$ where N is the number of classes. The formula to calculate softmax activation function is,

$$\sigma(z_i) = \frac{e^{x_i}}{\sum_{i=1}^{N} e^{x_i}} \quad (4)$$

$where\ \sigma(z_i)$ = Softmax activation function,

N is the number of classes of a multiclass classifier.

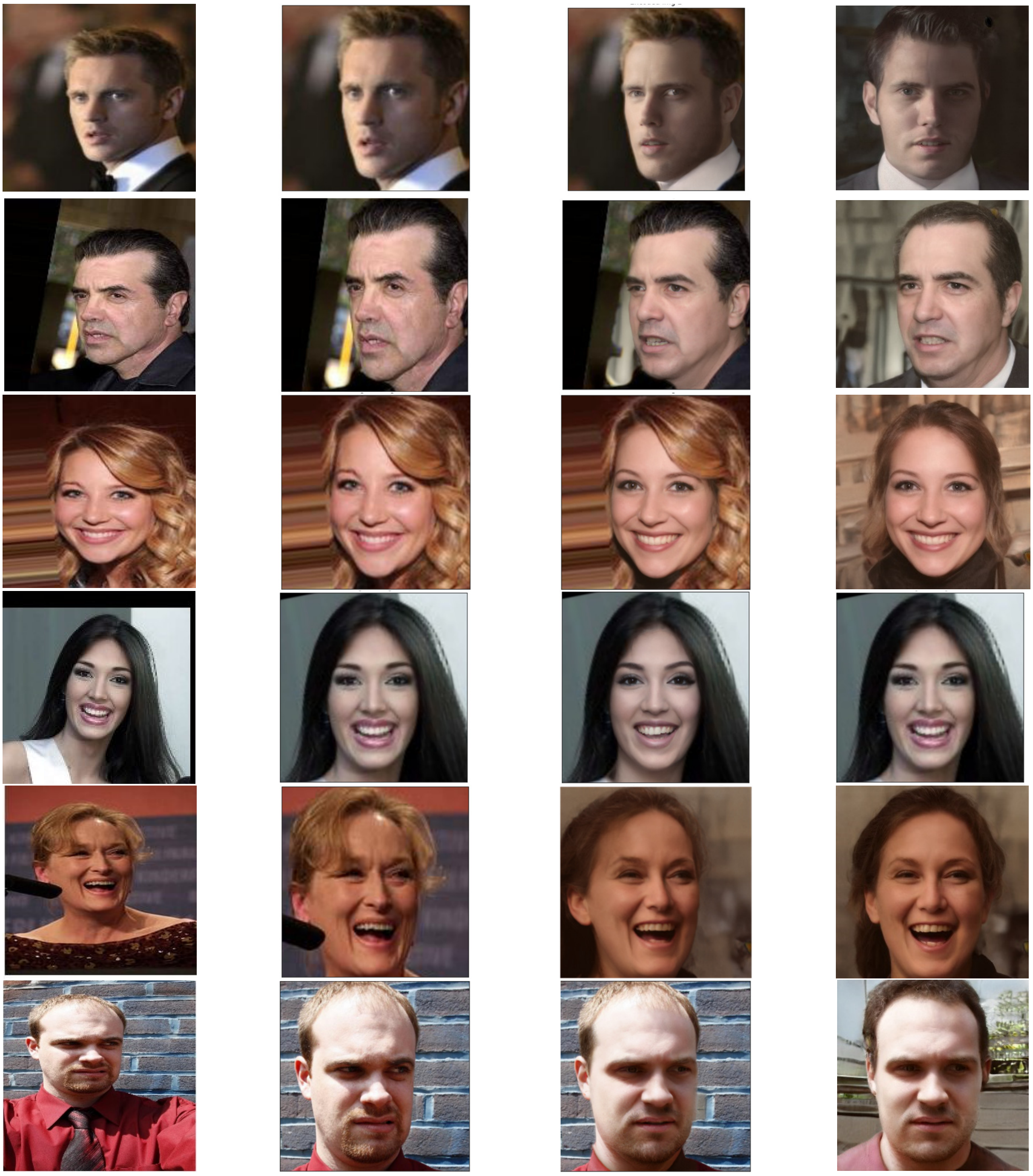
***Algorithm 2*:** Training CNN model for classification.

```
Input:
$M_{CNN}$: Deep CNN model
$x_i$: Input image
$X$: No of input images
$i \in [1,7]$
Functions:
$y_i \leftarrow M_{CNN}(x)$: Output of $M_{CNN}$ in probability, given $x_i \in X$
$loss \leftarrow l_{CE}$: Calculate cross-entropy loss.
Training:
for number of training iterations do:
  for each $x_i \in X$:
  $y_i \leftarrow M_{CNN}(x_i)$
  $loss \leftarrow l_{CE} = j(\theta) = -\frac{1}{m}\sum_{i=1}^{m}[y_i \log(h_\theta(x_i)) + (1-y_i)\log(1-h_\theta(x_i))]$
  Update $M_{CNN}$ with loss
  $C_{exp} = (\max(P(y_i))$
  end
end
```

4. Results and discussion

The proposed model is evaluated with a benchmark dataset and real images. The ground truth images are passed into the model to predict the latent code. The expression and identity components are distilled from the ground truth image. The results displayed below show that the generated images are very close to the ground truth image.

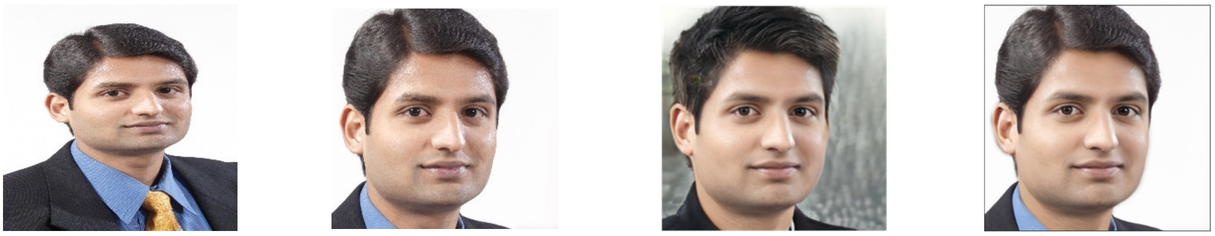
Figure 7 represents the results obtained from the first stage of the proposed model. Figure 7(a) represents the ground truth image passed as input to the encoder to obtain the feature vector. Figure 7(b) represents the aligned image. Figure 7(c) represents the latent representation of the facial image. Figure 7(d) represents the rebuilt image generated by concatenating the feature vectors. It is evident from the results that our model preserved the identity and expression of the ground truth image. The model achieves high-quality image synthesis for a multi-posed facial image with expression.



(a) Ground truth (b) aligned image (c) latent representation (d) generated image.

Figure 7. Image synthesis using latent space vector of generative adversarial network.

In this section, the real faces are manipulated to analyze the performance of the proposed model for real faces. Figure 8 shows the results of generating facial images from the latent code of the image. Results show that the image can successfully predict the facial expression for real faces.



(a) Ground truth (b) aligned image (c) latent representation (d) generated image

Figure 8. Manipulation on real faces.

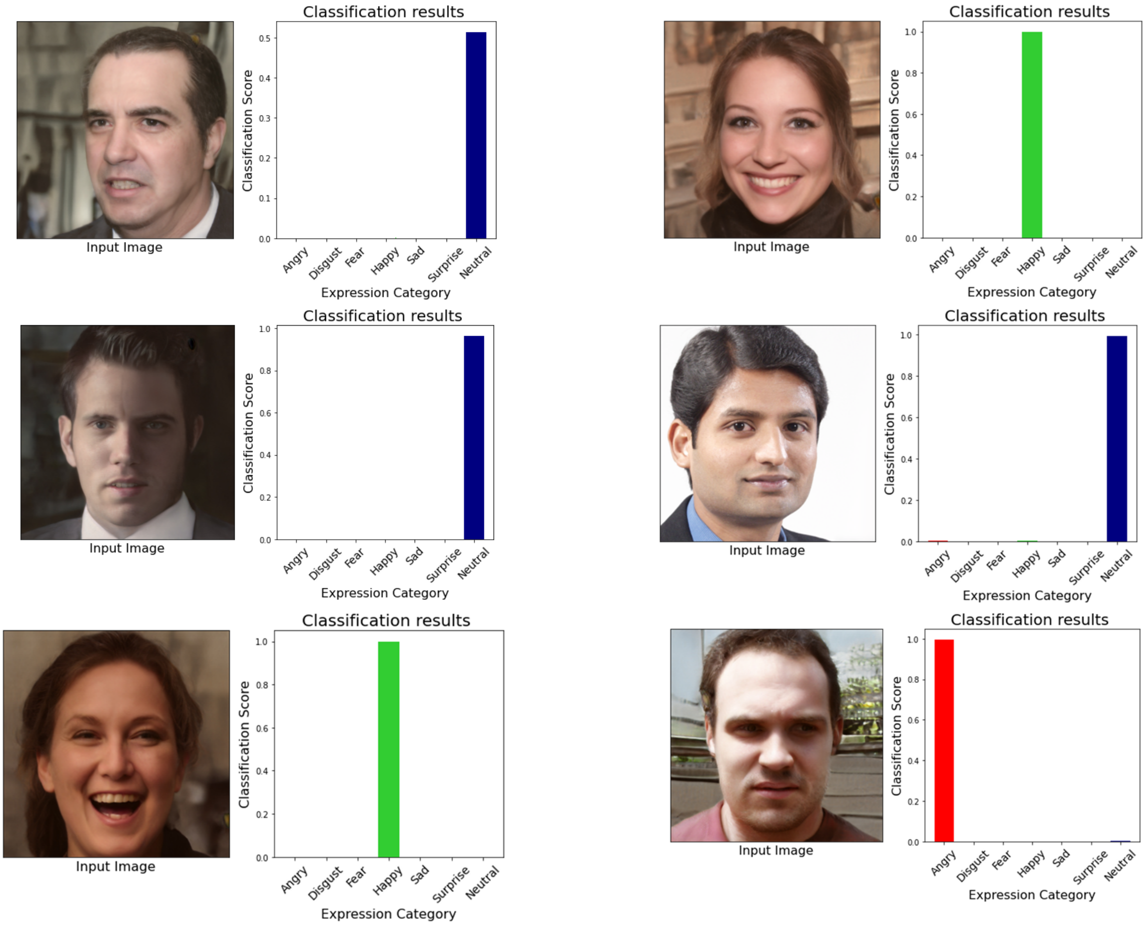
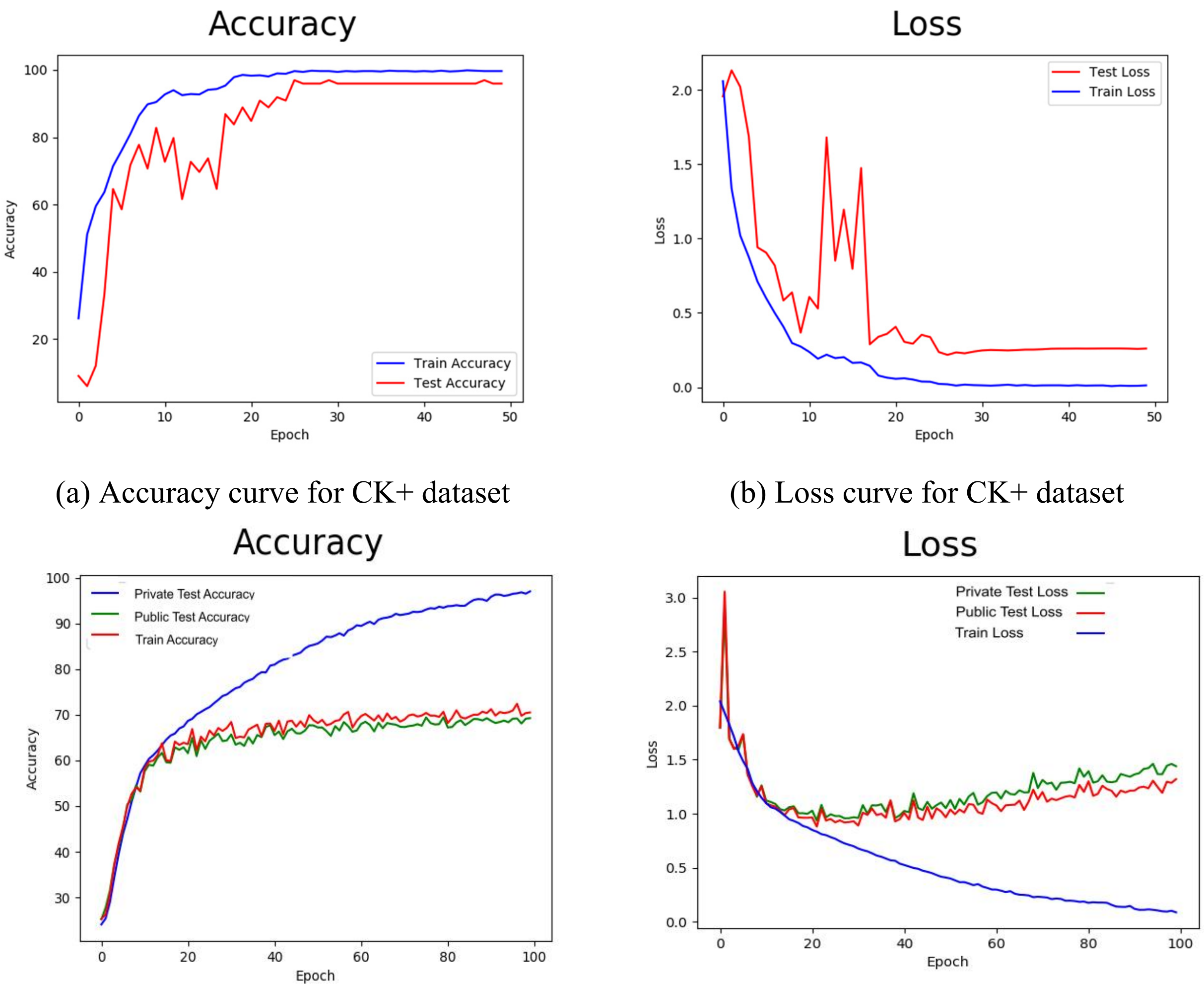


Figure 9. Experimental results–facial expression recognition.

Figure 9 shows the results achieved using the proposed method. The left side of the results shows the input image and the right side of the results shows the predictions of the proposed model. The features of the facial image passed to the neural network are extracted and concatenated with features extracted in the first phase. With concatenated features, the facial expressions are recognized and categorized into seven different classes. Experimental results show that the proposed model accurately classifies the emotions into seven different classes. The recognized expressions are plotted graphically, representing the emotion category against the classification score. Different colored bars are used to represent different emotions. The results classified the facial images for happy, neutral and angry emotions.



(a) Accuracy curve for CK+ dataset

(b) Loss curve for CK+ dataset

(c) Accuracy curve for FER2013 dataset

(d) Loss curve for FER2013 dataset

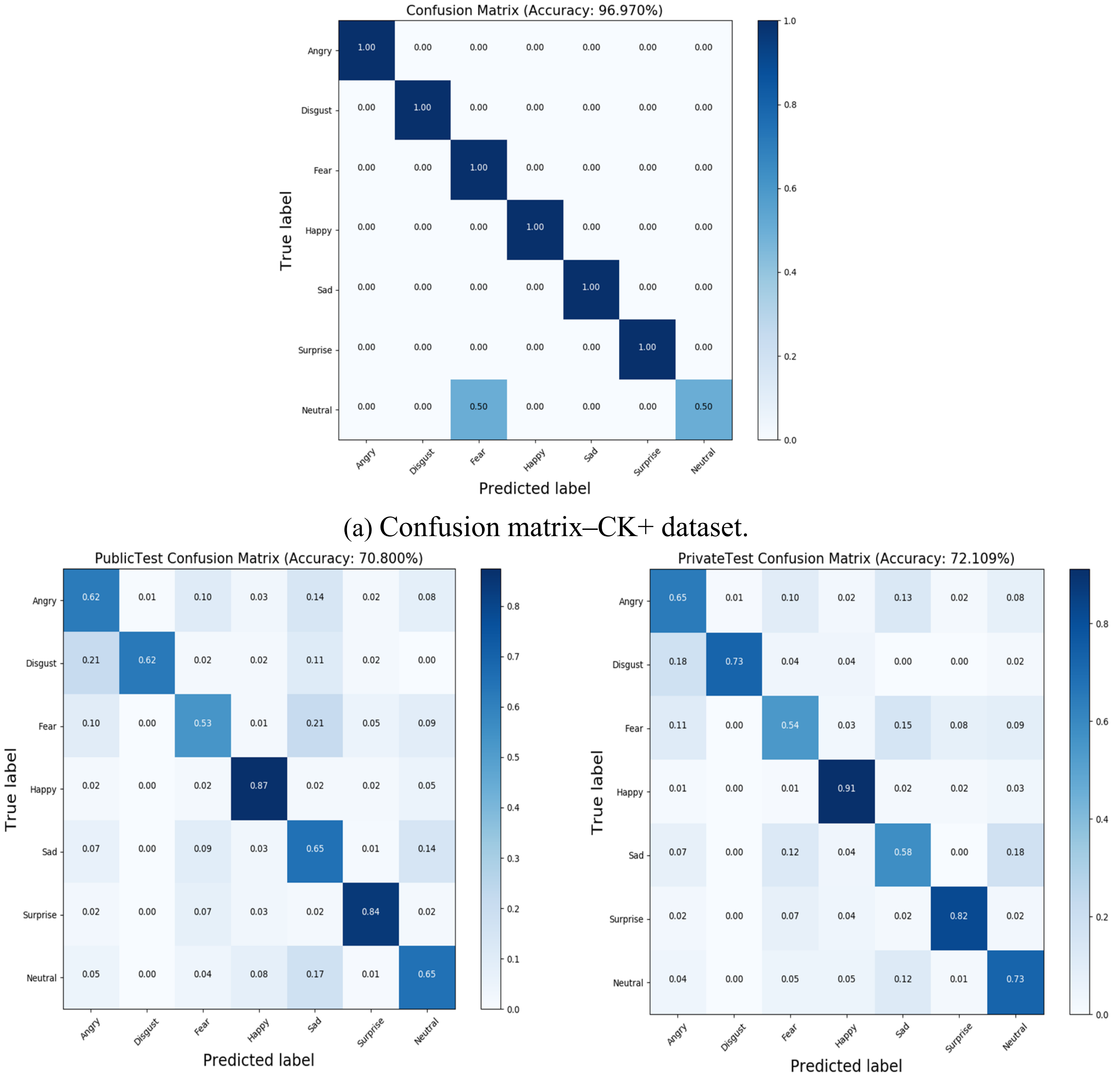
Figure 10. Accuracy and Loss curves for the proposed model.

The effectiveness of the model is evaluated by calculating the facial expression recognition accuracy. The accuracy of the model is calculated using the formula,

$$Accuracy = \frac{Number\ of\ correct\ predictions}{Total\ number\ of\ predictions}$$

Figure 10(a),(b) show the accuracy and loss curves of the proposed model for the CK+ dataset in predicting facial expression. The performance of the model is improved using 10-fold cross-validation. The recognition accuracy of 96.97% is achieved using the proposed model. The model outperforms other models that adopted hand-crafted features. The accuracy achieved showcases the superiority of the GAN-based deep learning model in extracting the facial expression features and recognizing the facial emotions. Recognition accuracy of 95.94% is achieved using ResNet18. Recognition accuracy of 95.39% is achieved using ResNet50. The accuracy achieved using VGG19 is higher than the accuracy achieved using ResNet18 and ResNet50. The other CNN architectures, namely LeNet and AlexNet, have drawbacks when compared to VGG and ResNet architectures. In the case of LeNet architecture, it struggles with overfitting, and average pooling is used, whereas in other architectures, max pooling is used. Average pooling does not select prominent features as in the case of max pooling.

The use of Tanh in LeNet architecture is another drawback because of the vanishing gradient problem. Drawbacks of LeNet are overcome in AlexNet with the use of Max Pooling and Relu activation function. But, the main drawback of AlexNet architecture is rapid down-sampling of the intermediate representations through strided convolution and max-pooling layers. Figure 10(c),(d) show the accuracy and loss curves of the proposed model for the FER2013 dataset in predicting facial expression. The performance of the model is improved using 10-fold cross-validation. The recognition accuracy of 72.38% is achieved, which is higher than other models on the FER2013 dataset. The images present in the dataset are noisy with low illumination, blurred and occluded. The recognition accuracy can further be improved by applying denoising techniques on the images and data augmentation can be performed to increase the number of the images. The model achieves high performance when compared to the models that handled only frontal view of the facial images [38].



	Angry	Disgust	Fear	Happy	Sad	Surprise	Neutral
Angry	1.00	0.00	0.00	0.00	0.00	0.00	0.00
Disgust	0.00	1.00	0.00	0.00	0.00	0.00	0.00
Fear	0.00	0.00	1.00	0.00	0.00	0.00	0.00
Happy	0.00	0.00	0.00	1.00	0.00	0.00	0.00
Sad	0.00	0.00	0.00	0.00	1.00	0.00	0.00
Surprise	0.00	0.00	0.00	0.00	0.00	1.00	0.00
Neutral	0.00	0.00	0.50	0.00	0.00	0.00	0.50

(a) Confusion matrix–CK+ dataset.

	Angry	Disgust	Fear	Happy	Sad	Surprise	Neutral
Angry	0.62	0.01	0.10	0.03	0.14	0.02	0.08
Disgust	0.21	0.62	0.02	0.02	0.11	0.02	0.00
Fear	0.10	0.00	0.53	0.01	0.21	0.05	0.09
Happy	0.02	0.00	0.02	0.87	0.02	0.02	0.05
Sad	0.07	0.00	0.09	0.03	0.65	0.01	0.14
Surprise	0.02	0.00	0.07	0.03	0.02	0.84	0.02
Neutral	0.05	0.00	0.04	0.08	0.17	0.01	0.65

(b) Confusion matrix–FER2013 public test set

	Angry	Disgust	Fear	Happy	Sad	Surprise	Neutral
Angry	0.65	0.01	0.10	0.02	0.13	0.02	0.08
Disgust	0.18	0.73	0.04	0.04	0.00	0.00	0.02
Fear	0.11	0.00	0.54	0.03	0.15	0.08	0.09
Happy	0.01	0.00	0.01	0.91	0.02	0.02	0.03
Sad	0.07	0.00	0.12	0.04	0.58	0.00	0.18
Surprise	0.02	0.00	0.07	0.04	0.02	0.82	0.02
Neutral	0.04	0.00	0.05	0.05	0.12	0.01	0.73

(c) Confusion matrix–FER 2013 private test set

Figure 11. Confusion matrix – CK+ and FER2013 dataset.

Confusion matrix was analyzed to evaluate the performance in determining the facial expression. Figure 11(a) shows the confusion matrix for the performance evaluation of our model on the CK+ dataset for different expressions and the accuracy of overall expression recognition. The average recognition rate is 96.97%. The recognition accuracy represents that the model can recognize the facial expression regardless of the angle of the head. The confusion matrix depicts that the model performs exceptionally well for happy, sad, surprise, angry and disgust expressions with very high accuracies. One hundred percent accuracy is achieved for the expressions with a good number of samples for each expression. It should be noted that the classification error occurs in recognizing fear and neutral emotions. The accuracy of recognizing fear and neutral is low because of a small number of training images for the two expressions. The results suggest that automated models can perform equally well as a human observer does. Figure 11(b) shows the confusion matrix for the performance evaluation of our model on the FER2013 Public Test set for different expressions and the accuracy of overall expression recognition. Figure 11(c) shows the confusion matrix for the performance evaluation of our model on the FER2013 Private Test set for different expressions and the accuracy of overall expression recognition.

Table 1. Comparison of facial expression recognition performance with existing methods.

Reference	Method	Dataset	Class	Accuracy (%)
[50]	Coordinates of facial key point tracking	CK+	6	94.31
[51]	Three stage support vector machine	CK+	7	93.29
[52]	CNN based expression recognition	CK+	7	80.30
[53]	general purpose graphic processing unit	CK+	7	96.02
[41]	Weighted mixture deep neural network	CK+	6	96
[46]	CNN for facial expression recognition	FER2013	6	65%
	Multi-Pose Facial Expression Recognition using Latent Space Vector	CK+	7	96.97
	Multi-Pose Facial Expression Recognition using Latent Space Vector	FER2013	7	72.38%

Table 1 shows the comparison of facial expression recognition performance with existing methods. The methods listed in the table perform facial expression recognition on the frontal view of the facial images. The proposed work takes multi-pose facial images and performs facial expression recognition. From the results, it can be observed that the model outperforms the existing state-of-art methods for multi-pose facial expression recognition.

5. Conclusions

The work proposed model to extract features from the latent representation of the facial image. The given facial image is encoded into feature vectors from which the input ground truth image is recovered back. The model recovers identity and expression discriminative representation of the facial image. Experiments were conducted using real images and the CK+ dataset. When compared with the existing works, the current work generalizes well and synthesizes visually appealing images preserving the semantics of the facial image. From the results, the proposed model is capable of extracting the facial expression regardless of the facial image view. The proposed model achieved state-of-art results with an accuracy of 96.97% for the CK+ dataset. The future work may explore facial expression recognition on an unconstrained expression dataset with a noisy environment and also explore the real-time applications of facial expression recognition.

Conflict of interest

All authors declare no conflicts of interest in this paper.

References

1. T. Fádel, S. Carvalho, B. Santos, Facial expression recognition in Alzheimer's disease: a systematic review, *J. Clin. Exp. Neuropsychol.,* **41** (2019), 192–203.
2. S. Li, W. Deng, Blended emotion in-the-wild: Multi-label facial expression recognition using crowdsourced annotations and deep locality feature learning, *Int. J. Comput. Vis.,* **127** (2019), 884–906.
3. W. Su, M. Liu, Y. Yang, J. Wang, S. Li, H. Lv, et al., PPD: a manually curated database for experimentally verified prokaryotic promoters, *J. Mol. Biol.,* (2019), forthcoming.
4. D. Jain, P. Shamsolmoali, P. Sehdev, Extended deep neural network for facial emotion recognition, *Pattern Recognit. Lett.,* **120** (2019), 69–74.
5. C. Gong, F. Lin, X. An, A novel emotion control system for embedded human–computer interaction in green Iot, *IEEE Access,* **7** (2019), 185148–185156.
6. J. Deng, G. Pang, Z. Zhang, Z. Pang, H. Yang, G. Yang, cGAN based facial expression recognition for human-robot interaction, *IEEE Access,* **7** (2019), 9848–9859.
7. F. Y. Dao, H. Lv, D. Zhang, Z. Zhang, L. Liu, H. Lin, DeepYY1: a deep learning approach to identify YY1-mediated chromatin loops, *Brief. Bioinf.,* **356** (2020).
8. M. Wu, W. Su, L. Chen, Z. Liu, W. Cao, K. Hirota, Weight-adapted convolution neural network for facial expression recognition in human-robot interaction, *IEEE Trans. Syst. Man Cybern.,* **51** (2021), 1473–1484.
9. M. Sajjad, A. Shah, Z. Jan, S. I. Shah, S. W. Baik, I. Mehmood, Facial appearance and texture feature-based robust facial expression recognition framework for sentiment knowledge discovery, *Cluster Comput.,* **21** (2018), 549–567.
10. A. Nandi, P. Dutta, M. Nasir, Automatic facial expression recognition using Histogram oriented Gradients (HoG) of shape information matrix, in *International Conference on Intelligent Computing and Communication*, Springer, (2019), 343–351.
11. F. Y. Dao, H. Lv, H. Zulfiqar, H. Yang, W. Su, H. Gao, et al., A computational platform to identify origins of replication sites in eukaryotes, *Brief. Bioinf.,* **22** (2020), 1940–1950.
12. S. Berretti, B. B. Amor, M. Daoudi, A. Bimbo, 3D facial expression recognition using SIFT descriptors of automatically detected keypoints, *Visual Comput.,* **27** (2011), 1021.
13. O. Starostenko, C. Cruz-Perez, V. Alarcon-Aquino, R. Rosas-Romero, Real-time facial expression recognition using local appearance-based descriptors, *J. Intell. Fuzzy Syst.,* **36** (2019), 5037–5049.
14. H. Yang, W. Yang, F. Dao, H. Lv, H. Ding, W. Chen, et al., A comparison and assessment of computational method for identifying recombination hotspots in Saccharomyces cerevisiae, *Brief. Bioinf.,* **21** (2020), 1568–1580.
15. J. Chorowski, R. J. Weiss, S. Bengio, A. Oord, Unsupervised speech representation learning using wavenet autoencoders, *IEEE/ACM Trans. Audio Speech Lang. Process.,* **27** (2019), 2041–2053.

16. R. R. N. Abirami, P. M. D. R. Vincent, K. Srinivasan, U. Tariq, C. Y. Chang, Deep CNN and deep GAN in computational visual perception-driven image analysis, *Complexity*, **2021** (2021), 5541134.
17. I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, et al., Generative adversarial nets, preprint, arXiv:1406.2661.
18. C. Han, L. Rundo, R. Araki, Y. Nagano, Y. Furukawa, G. Mauri, et al., Combining noise-to-image and image-to-image GANs: brain MR Image augmentation for tumor detection, *IEEE Access*, 7 (2019), 156966–156977.
19. C. Xu, P. M. Feng, H. Yang, W. Qiu, W. Chen, H. Lin, iRNAD: a computational tool for identifying D modification sites in RNA sequence, *Bioinformatics,* **35** (2019), 4922–4929.
20. H. Y. Lai, Z. Zhang, Z. Su, W. Su, H. Ding, W. Chen, et al., iProEP: a computational predictor for predicting promoter, *Mol. Ther. Nucleic Acids,* **17** (2019), 337–346.
21. R. Yanagi, R. Togo, T. Ogawa, M. Haseyama, Query is GAN: scene retrieval with attentional text-to-image generative adversarial network, *IEEE Access,* **7** (2019), 153183–153193.
22. Z. Y. Liang, H. Y. Lai, H. Yang, C. J. Zhang, H. Yang, H. Wei, et al., Pro54DB: a database for experimentally verified sigma-54 promoters, *Bioinformatics,* **33** (2017), 467–469.
23. J. Cai, Z. Meng, A. S. Khan, Z. Li, J. O'Reily, Y. Tong, Identity-free facial expression recognition using conditional generative adversarial network, preprint, arXiv:1903.08051.
24. K. Ali, C. Hughes, Facial expression representation learning by synthesizing expression images, preprint, arXiv:1912.01456.
25. T. Karras, S. Laine, T. Aila, A style-based generator architecture for generative adversarial networks, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR),* (2019), 4401–4410.
26. W. Li, Z. J. Zhong, P. P. Zhu, E. Deng, H. Ding, W. Chen, et al., Sequence analysis of origins of replication in the Saccharomyces cerevisiae genomes, *Front. Microbiol.,* **5** (2014), 574.
27. H. Zhang, I. Goodfellow, D. Metaxas, A. Odena, Self-attention generative adversarial networks, in *International Conference on Machine Learning*, (2019), 7354–7363.
28. H. Ding, S. H. Guo, E. Z. Deng, L. Yuan, F. Guo, J. Huang, et al., Prediction of Golgi-resident protein types by using feature selection technique, *Chemom. Intell. Lab. Syst.,* **124** (2013), 9–13.
29. T. Karras, T. Aila, S. Laine, J. Lehtinen, Progressive growing of GANs for improved quality, Stability, and Variation, preprint, arXiv:1710.10196.
30. A. Radford, L. Metz, S. Chintala, Unsupervised representation learning with deep convolutional generative adversarial networks, preprint, arXiv:1511.06434.
31. M. Mirza, S. Osindero, Conditional generative adversarial nets, preprint, arXiv:1411.1784.
32. X. Yin, X. Yu, K. Sohn, X. Liu, M. Chandraker, Towards large-pose face frontalization in the wild, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, (2017), 3990–3999.
33. Y. Shen, B. Zhou, P. Luo, X. Tang, FaceFeat-GAN: a two-stage approach for identity-preserving face synthesis, preprint, arXiv:1812.01288.
34. H. Lin, H. Ding, F. B. Guo, J. Huang, Prediction of subcellular location of mycobacterial protein using feature selection techniques, *Mol. Diversity,* **14** (2010), 667–671.
35. G. Arvanitidis, L. Hansen, S. Hauberg, Latent space oddity: on the curvature of deep generative models, preprint, arXiv:1710.11379.
36. F. Ma, U. Ayaz, S. Karaman, Invertibility of convolutional generative networks from partial measurements, *Adv. Neural Inf. Process. Syst.*, **31** (2018) , 9628–9637.

37. H. Lin, Q. Z. Li, Using pseudo amino acid composition to predict protein structural class: approached by incorporating 400 dipeptide components, *J. Comput. Chem.,* **28** (2007), 1463–1466.
38. X. Wang, X. Wang, Y. Ni, Unsupervised domain adaptation for facial expression recognition using generative adversarial networks, *Comput. Intell. Neurosci.,* **2018** (2018), 7208794.
39. T. Zhang, W. Zheng, Z. Cui, Y. Zong, J. Yan, K. Yan, A deep neural network-driven feature learning method for multi-view facial expression recognition, *IEEE Trans. Multimedia,* **18** (2016) ,2528–2536.
40. B. Yang, J. Cao, R. Ni, Y. Zhang, Facial expression recognition using weighted mixture deep neural network based on double-channel facial images, *IEEE Access,* **6** (2017), 4630–4640.
41. J. H. Kim, B. G. Kim, P. P. Roy, D. Jeong, Efficient facial expression recognition algorithm based on hierarchical deep neural network structure, *IEEE Access,* **7** (2019), 41273–41285.
42. C. Zhang, P. Wang, K. Chen, J. K. Kämäräinen, Identity-aware convolutional neural networks for facial expression recognition, *J. Syst. Eng. Electron.,* **28** (2017), 784–792.
43. P. M. Ferreira, F. Marques, J. S. Cardoso, A. Rebelo, Physiological inspired deep neural networks for emotion recognition, *IEEE Access,* **6** (2018), 53930–53943.
44. S. M. González-Lozoya, J. de la Calleja, L. Pellegrin, H. J. Escalante, M. A. Medina, A. Benitez-Ruiz, Recognition of facial expressions based on CNN features, *Multimedia Tools Appl.,* **79** (2020), 13987–14007.
45. L. Yang, Y. Tian, Y. Song, N. Yang, K. Ma, L. Xie, A novel feature separation model exchange-GAN for facial expression recognition, *Knowl. Based Syst.*, **204** (2020), 106217.
46. S. Liong, Y. Gan, D. Zheng, S. Li, H. Xu, H. Zhang, et al., Evaluation of the spatio-temporal features and gan for micro-expression recognition system, *J. Signal Process. Syst.*, **92** (2020), 705–725.
47. R. Wu, G. Zhang, S. Lu, T. Chen, Cascade ef-gan: Progressive facial expression editing with local focuses, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, (2020), 5021–5030.
48. A. Jahanian, L. Chai, P. Isola, On the "steerability" of generative adversarial networks, preprint, arXiv:1907.07171.
49. T. Kanade, J. F. Cohn, Y. Tian, Comprehensive database for facial expression analysis, in *Proceedings Fourth IEEE International Conference on Automatic Face and Gesture Recognition (Cat. No. PR00580)*, IEEE, (2000), 46–53.
50. N. Aifanti, A. Delopoulos, Linear subspaces for facial expression recognition, *Signal Process. Image Commun.,* **29** (2014), 177–188.
51. I. Dagher, E. Dahdah, M. Al Shakik, Facial expression recognition using three-stage support vector machines, *Visual Comput. Ind. Biomed. Art,* **2** (2019), 24.
52. K. Shan, J. Guo, W. You, D. Liu, R. Bie, Automatic facial expression recognition based on a deep convolutional-neural-network structure, in *2017 IEEE 15th International Conference on Software Engineering Research, Management and Applications (SERA)*, IEEE, (2017), 123–128.
53. V. Mayya, R. M. Pai, M. M. Pai, Automatic facial expression recognition using DCNN, *Procedia Comput. Sci.,* **93** (2016), 453–461.