Neural Computing and Applications (2023) 35:23311–23328
https://doi.org/10.1007/s00521-021-06012-8

SPECIAL ISSUE ON HUMAN-IN-THE-LOOP MACHINE LEARNING AND ITS APPLICATIONS

Deep learning-based facial emotion recognition for human–computer interaction applications

M. Kalpana Chowdary[1] · Tu N. Nguyen[2] · D. Jude Hemanth[1]

Received: 7 January 2021 / Accepted: 2 April 2021 / Published online: 22 April 2021

Abstract

One of the most significant fields in the man–machine interface is emotion recognition using facial expressions. Some of the challenges in the emotion recognition area are facial accessories, non-uniform illuminations, pose variations, etc. Emotion detection using conventional approaches having the drawback of mutual optimization of feature extraction and classification. To overcome this problem, researchers are showing more attention toward deep learning techniques. Nowadays, deep-learning approaches are playing a major role in classification tasks. This paper deals with emotion recognition by using transfer learning approaches. In this work pre-trained networks of Resnet50, vgg19, Inception V3, and Mobile Net are used. The fully connected layers of the pre-trained ConvNets are eliminated, and we add our fully connected layers that are suitable for the number of instructions in our task. Finally, the newly added layers are only trainable to update the weights. The experiment was conducted by using the CK + database and achieved an average accuracy of 96% for emotion detection problems.

Keywords Human–computer interaction · Transfer learning · Resnet50 · VGG 19 · Inception V3 · MobileNet

1 Introduction

Emotions play a major role during communication. Recognition of facial emotions is useful in so many tasks such as customer satisfaction identification, criminal justice systems, e-learning, security monitoring, social robots, and smart card applications, etc. [1, 2]. The main blocks in the traditional emotion recognition system are detection of faces, extracting the features, and classifying the emotions [3]. Based on the literature the most used feature extraction methods are Bezier curves [4], clustering methods [5], Independent Component analysis [6], two-directional two-dimensional Fisher principal component analysis ($(2D)^{2-}$FPCA) [7, 8], two-directional two-dimensional Modified Fisher principal component analysis ($(2D)^2$ MFPCA) [9], Principle component analysis [10], Local binary patterns [11] and feature level fusion techniques [12], etc. After that, the features are given to the classifiers like Support vector machines [13], Hidden Markov models [14], k-nearest neighbors [15], Naïve Bayes, and Decision trees [16], etc. for classification. The drawback in conventional systems is that the feature extraction and classification phases are independent [17]. So, it is challenging to increase the performance of the system.

Deep learning networks uses end to end learning process to overcome the problems in conventional approaches [18–20]. The data size is very important in deep learning, the greater the dataset the performance is good. To improve the performance of deep learning, researchers are using data augmentation [21], translations, normalizations, cropping, adding noise and, scaling techniques [22] to increase the data size. Convolution Neural Networks is the best-proven algorithms in segmentation and classification tasks. The automatic feature extraction is one of the main advantages of this convolution neural network. Transfer

✉ D. Jude Hemanth
judehemanth@karunya.edu

[1] Department of ECE, Karunya Institute of Technology and Sciences, Coimbatore, India

[2] Department of Computer Science, Purdue University Fort Wayne, Fort Wayne, USA

learning is one of the methods in deep learning; in this, a model trained for a particular task is reused for another task by the transfer of knowledge [23]. The main advantages of transfer learning are time-saving and more accurate [24].

Some of the recent works in the area of Expression recognition using convolutional neural networks are discussed. Yingruo Fan et al. [25] proposed the multi-region ensemble convolutional neural network method for facial expression identification. In this, the features are extracted from three regions of eyes, nose, and mouth are given to three sub-networks. After that, the weights from three sub-networks are ensemble to predict the emotions. The databases used in this work are AFEW 7.0 and RAF-DB. Yingying Wang et al. [26] proposed recognition of emotions based on the auxiliary model. In this work, the information from three major sub-regions of eyes, nose, and mouth are combined with the complete face image through the weighting process to seize the maximum information. The model is evaluated by using four databases of CK + , FER2013, SFEW, and JAFFE. Frans Norden et al. [27] presented facial expression recognition using VGG16 and Resnet50. The databases used in this work are JAFFE and FER2013. The experimental outcome shows the finest classification accuracy is attained by Resnet50 when evaluated with other state of art methods.

Jyostna Devi Bodapati et al. [28] proposed recognition of emotions by using deep Convolution neural networks based-features. In this work, VGG16 is used to extract the features and a multi-class Support vector machine (SVM) is used for classification. The proposed algorithm achieved an accuracy of 86.04% with the face detection algorithm and 81.36% without the face detection algorithm on the CK + database. Nithya Roopa et al. [29] proposed emotion recognition using the Inception V3 model. The work is evaluated on the KDEF database and achieved a test accuracy of 39%. To handle the occlusions and pose variations Sreelakshmi et al. [30] presented an emotion recognition system by using MobileNet V2 architecture. The model is tested on real-time occluded images and achieves an accuracy of 92.5%. Aravind Ravi [31] proposed pre-trained CNN features based on facial emotion recognition. In this work, a pre-trained VGG19 network is used to extract the features and the support vector machine is used to predict the expressions. The experiment was conducted on two databases JAFFE and CK + and achieved the accuracies of 92.86% and 92.26%, respectively.

Shamoil shaees et al. [32] proposed a transfer learning approach with a support vector machine classifier. In this work, features are extracted by using CNNs, AlexNet, and feeding those features to SVM for classification. The work has been done using two databases of CK + and NVIE and achieved good accuracy. The authors of [33] presented facial emotion recognition with convolution neural networks. The experiment was conducted using different models such as VGG 19, VGG 16, and ResNet50 using the fer2013 dataset. Compared to all three models VGG 16 achieved the highest accuracy of 63.07%.Mehmet Akif OZDEMIR et al. [34] presented LeNet architecture-based emotion recognition system. In this work, a merged dataset (JAFFE, KDEF, and own custom data) is used. Haar cascade library is used in this work to remove the unwanted pixels that are not used for expression recognition. The accuracy achieved in this work is 96.43%. Poonam Dhankhar et al. [35] presented Resnet50 and VGG16 architectures for facial emotion recognition. The Ensemble model is suggested in this work by combining the models of Resnet50 and VGG16. The ensemble model proposed in this work is achieved the highest accuracy when compared with baseline SVM, and individual Resnet50 and VGG16 models. SVM achieves an accuracy of 37.9%, Resnet50 and VGG16 achieve the accuracies of 73.8% and 71.4%, respectively, and finally, the ensemble model achieves the highest accuracy of 75.8%. The authors of explored the transfer learning approach for facial expression recognition. In this work, the pre-trained networks of Alexnet, VGG, and Resnet architectures are used and attained an average accuracy of 90% on the combined dataset of JAFFE and CK + .

In this paper, transfer learning approach is used for facial emotion recognition. This paper is further subdivided into the subsequent sections. Section 2 discusses theories of emotions and emotion models, Sect. 3 explains the materials and methods, Sect. 4 describes the training procedure of proposed models, Sect. 5 discusses implementation parameters, Sect. 6 discusses the experimental results, Sect. 7 is comparisons, and Sect. 8 is the conclusion.

2 Related background

One of the most active research in the recent scenario is affective computing. The process of improvement of systems to recognize and simulate human affects is called affective computing [36]. The purpose of affective computing is to increase the intelligence of computers for human–computer interaction. Some of the applications of affective computing are Distance education, Internet banking, Virtual sales assistant, Neurology, Medical and Security fields, etc. [37]. In affective computing, the main step is to recognize human emotions by speech signals, body postures, or by facial expressions [38].

2.1 Theories of emotions

The emotions theories are grouped into three categories: Physiological (James–Lange and Cannon–Bard theories), Cognitive (Lazarus theory), and Neurological (Facial feedback theory) as shown in Fig. 1.

The James–Lange model proposes the happening of emotion is due to the interpretation of the physiological response. After that, Walter Cannon disagreed with James–Lange theory and proposed that the emotions and physiological reactions are occurring simultaneously in Cannon-Bard theory [39]. Lazarus theory is also called Cognitive appraisal theory, in this physiological response occurs first, and then the person thinks the reason for the physiological response to experience the emotion [40]. Finally, the facial feedback theory explains the emotional experience through facial expressions.

2.2 Emotion models

Emotion models are mainly classified into two types: categorical models and dimensional models. The basic emotions of anger, fear, sadness, happiness, surprise, and disgust proposed by Ekman and Friesen are presented in the categorical model [41]. Dimensional model describes the emotions in two dimensional (Arousal and Valence) or three dimensional (Power, Arousal, and Valence). The Emotion models as shown Fig. 2.

Valence determines the emotion's positivity or negativity and Arousal measures the intensity of excitement of the expression. Circumplex, vector, and PANA (Positive Action- Negative Action) are two-dimensional models Plutchik's and PAD (Pleasure, Arousal and Dominance) are three- dimensional models. The detailed explanation of all the models is explained in [42].

3 Materials and methods

Nowadays, extracting human emotions are playing a major role in affective computing. The process of emotion detection using pre-trained Convnets is shown in Fig. 3.

In this work, 918 images are taken from the CK + dataset. Sample pictures are displayed in Fig. 4.

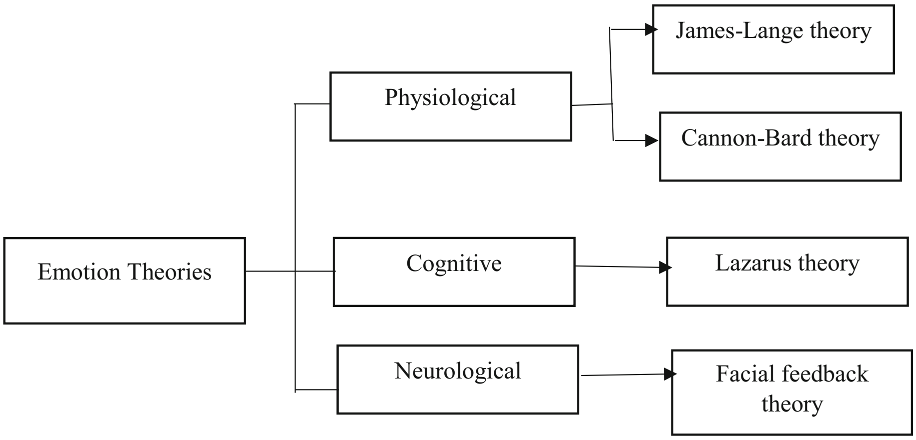
All the images are in.png format. Among 918 images, 770 images are used for training purposes and 148 are used for testing purposes. It contains seven emotions such as anger, surprise, contempt, sadness, happiness, disgust, and fear. The official web link of the CK + database is http://www.jeffcohn.net/Resources/.

The initial step in the process is image resizing. We have to resize the inputs according to the input sizes of the pre-trained models. The CK + dataset images are mostly gray with a resolution of 640*490. The actual input sizes of Resnet50, vgg19, and MobileNet are 224*224 and Inception V3 is 299*299. So all the images are resized according to the input size of pre-trained Convnets. After that, all the layers of the pre-trained Convnets are frozen except the fully connected layers. Finally, the fully connected layers are only trainable to update the weights. Based on the number of classes in a fully connected layer, the emotions are classified. In this work, we are using the networks of Resnet50, VGG19, Inception V3 and MobileNet that are trained on the ImageNet. These pre-trained networks are used in our classification task by the process of transfer learning.

4 Training procedure of proposed models

Transfer learning is a strategy of reusing the model developed for a particular task is used for another task. The fundamental concept of transfer learning is taking a model

Fig. 1 Theories of Emotions



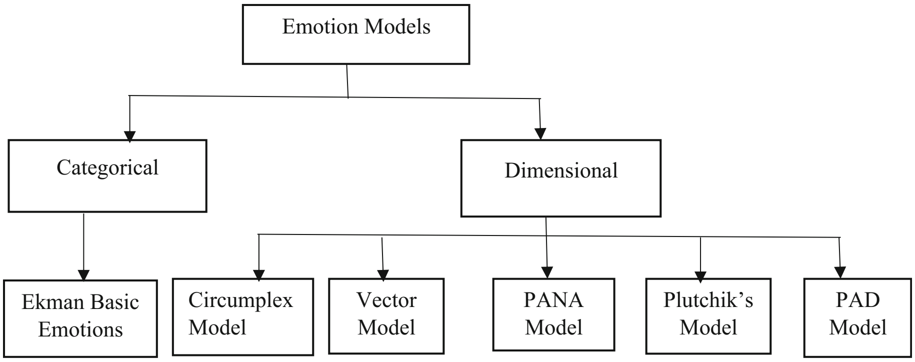


Fig. 2 Emotion Models

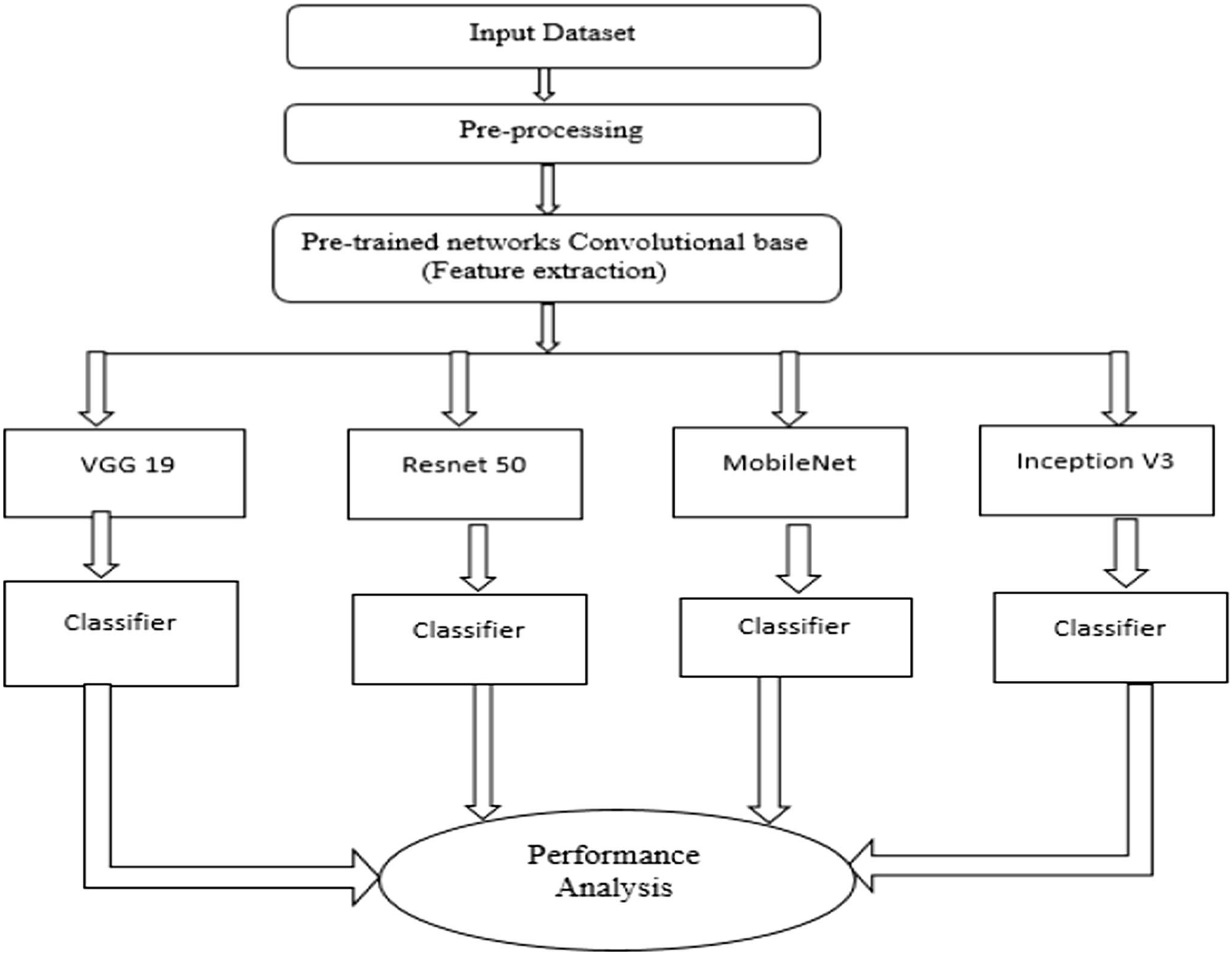


Fig. 3 Emotion detection process

trained on a big dataset and transferring its knowledge to a small dataset. Training a convolutional neural network from scratch requires more data and computationally expensive; on the other hand, transfer learning is computationally efficient, and a lot of data are also not needed. In this work, the training procedure for all the models is same, in the first step the weights are initialized from the ImageNet database before the training on the emotion dataset. By considering the advantage of transfer learning the last three layers (fully connected layer, a softmax layer, and classification output layer) of pre-trained models are replaced. And then, add the newly connected layers that are suitable to the classification task. Let us see the architectures of various networks.

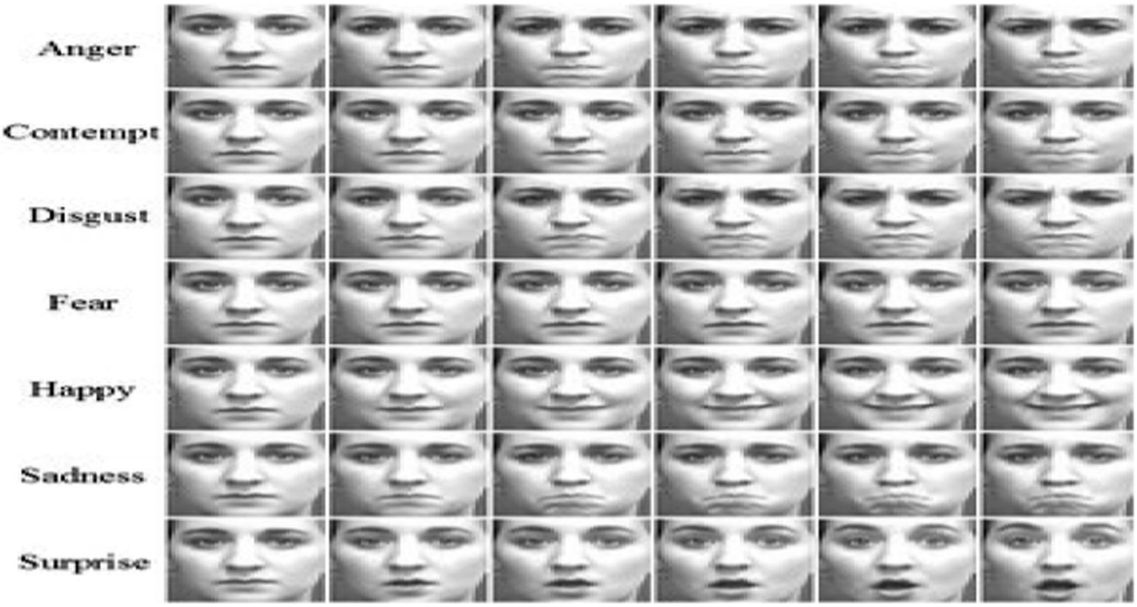


Fig. 4 Sample pictures from CK + dataset for seven expressions

4.1 VGG 19

The total number of layers in VGG 19 architecture is 19 layers. This VGG 19 is trained on the ImageNet database [43]. The ImageNet contains more than 14 million images and also capable to classify the images into 1000 different class labels. Figure 5 explains the architecture of VGG19.

The input size of this model is 224*224*3(RGB image). The architecture of VGG19 consists of sixteen convolutional layers and three fully connected layers. The size of the convolution kernels is 3*3 with a one-pixel stride. The network contains five max-pooling layers with a kernel size of 2*2 with a two-pixel stride. It consists of three fully connected layers, in that the first two fully connected layers having 4096 channels each, and the last fully connected layer comprises 1000 channels. The last layer of the architecture is the Softmax layer [44].

In this effort, we used the pre-trained model to extract the features and changed the fully connected layers as per our classification task. In this work, we are aiming to classify a total of seven emotions. The VGG19 network consists of 4096*1000 fully connected layers, as per our classification task we are replacing the last layer with 1024*7 fully connected layer. Below Table 1 shows the

Fig. 5 VGG19 model

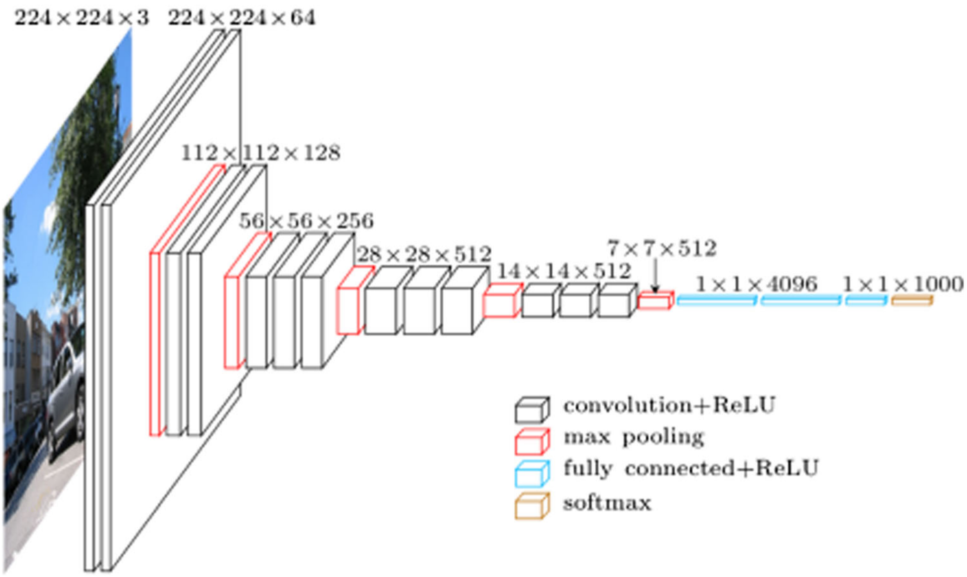


Table 1 Keras summary of the model using VGG19 as a feature extractor

Layer (type)	Output shape	Param #
vgg19 (Model)	(None, 7, 7, 512)	20,024,384
dense_9 (Dense)	(None, 7, 7, 1024)	525,312
dense_10 (Dense)	(None, 7, 7, 7)	7175

Total params: 20,556,871

Trainable params: 532,487

Non-trainable params: 20,024,384

summary of the proposed CNN using VGG19 as the base model and added our own fully connected layers on the top of the base model.

4.2 Resnet50

One of the classes of deep neural networks is Resnet50. Resnet stands for residual networks. The architecture of the Resnet50 contains 50 layers. In this also the convolution and pooling layers are similar to standard convolution neural networks. The main block in the resnet architecture is the residual block. The purpose of the residual block is to make connections between actual inputs and predictions. The residual block functioning is displayed in Fig. 6.

From the above diagram, x is the prediction, and F(x) is the residual. When x is equal to the actual input the value of F(x) is zero. Then, the identity connection copies the same x value [45].

The Resnet50 architecture mainly contains five stages with convolution and identity blocks. The input size of the resnet50 is 224*224 and is three channeled. Initially, it consists of a convolution layer with kernel size 7*7 and a max-pooling layer with 3*3 kernel size. In this architecture, each convolution block has three convolution layers and each identity block also contains three convolution layers. After the five stages, the next is the average pooling layer and the final layer is fully a connected layer with 1000 neurons. The architecture of the resnet50 is shown in Fig. 7. As per our work, we are considering resnet50 as the base model and we added our fully connected layers on the uppermost of it. For that, we are replacing the last layer into 1024*7 fully connected layers. Table 2 displays the summary of the proposed CNN using Resnet50 as the base model and added our own fully connected layers on the topmost of the base model.

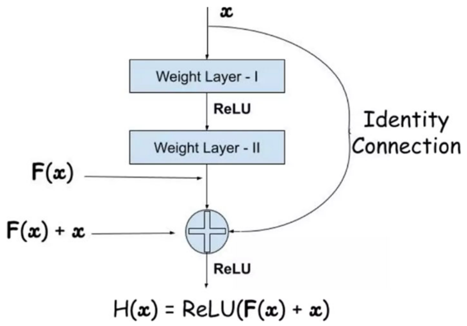


Fig. 6 Residual block

4.3 MobileNet

MobileNet is also called a lightweight convolution neural network. This is the most efficient architecture for mobile applications. The advantage of the MobileNet is it required less computational power to run. Instead of standard convolutions, the MobileNet used depth-wise separable convolutions. The number of multiplications required for depth-wise separable convolutions is less than the standard convolution so that the computational power is also reduced. Figure 8 shows the MobileNet Architecture.

Depth-wise separable convolution involves depth-wise convolutions and point-wise convolutions. In standard CNN's the convolution is applied to all the M channels at the same time but in depth-wise convolution the convolution is applied to a single channel at a time. In point-wise convolution, the 1*1 convolution is applied to merge the outputs of depth-wise convolutions [46, 47]. Figure 9 shows the depth-wise and point-wise convolutions.

The computational cost of standard convolution is

$$D_k.D_k.M.N.D_F.D_F \quad (1)$$

And the computational cost of depth-wise separable convolution is

$$D_k.D_k.M.D_F + M.N.D_F.D_F \quad (2)$$

The overall MobileNet construction consists of convolution layers with stride 2, depth-wise layers, and also point-wise layers to double the channel size. The structure of the MobileNet is presented in Table 3.

The final layer of the MobileNet architecture contains 1024*1000 fully connected layers, as per our emotion classification task we are replacing the final layer of the MobileNet with 1024*7 fully connected layers as displayed in Table 4.

4.4 Inception V3

Inception V3 is also one type of convolution neural network model. The input size of Inception V3 is 299*299 and it is a 48 layer deep network. The below Fig. 10 shows the base Inception V3 module. The 1 × 1 convolutions are added before the bigger convolutions to reduce the dimensionality and the same is done after the pooling layer

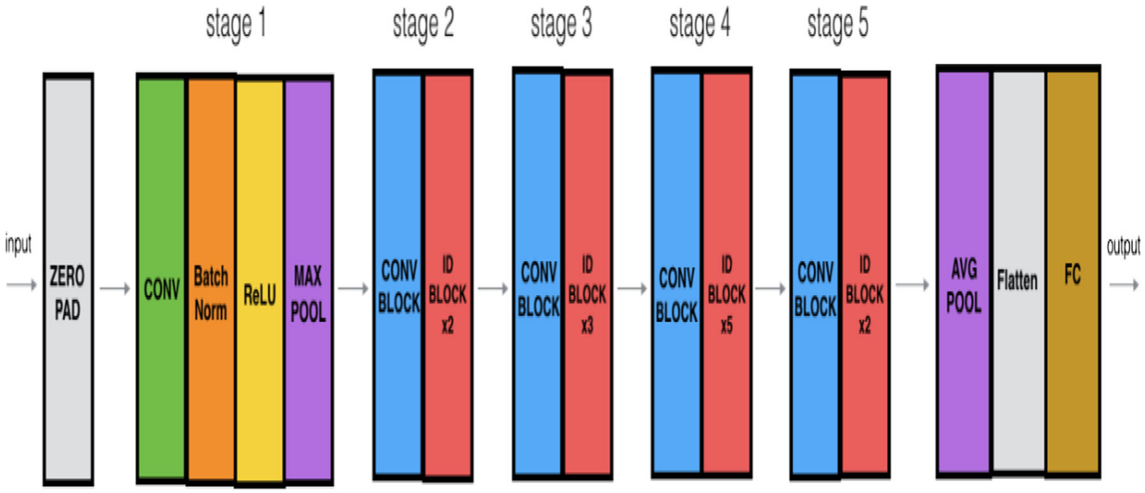


Fig. 7 Resnet50 Architecture

Table 2 Keras summary of the model using Resnet50 as a feature extractor

Layer (type)	Output shape	Param #
resnet50 (Model)	(None, 7, 7, 2048)	23,587,712
dense_13 (Dense)	(None, 7, 7, 1024)	2,098,176
dense_14 (Dense)	(None, 7, 7, 7)	7175

Total params: 25,693,063

Trainable params: 2,105,351

Non-trainable params: 23,587,712

also. To increase the performance of the architecture the 5 × 5 convolutions are into two 3 × 3 layers. It is also possible to factorize N × N convolutions into 1 × N and N × 1 convolutions.

The detailed structure of Inception V3 with input sizes, layer types (convolutional, pooling and softmax) and kernel sizes are presented in Table 5.

The final layer of the Inception V3 architecture [48] contains 2048*1000 fully connected layers as shown in Table 5, according to our emotion classification task we are replacing the last layer of the Inception V3 with 1024*7 fully connected layers as presented in Table 6.

5 Implementation

The experiment was done using Google Colaboratory with GPU backend using RAM of 12 GB. Using the Tensorflow and Keras API we can design VGG 19, Resnet50, MobileNet and Inception V3 architectures from scratch. For this implementation, we used the CK + dataset. The number of

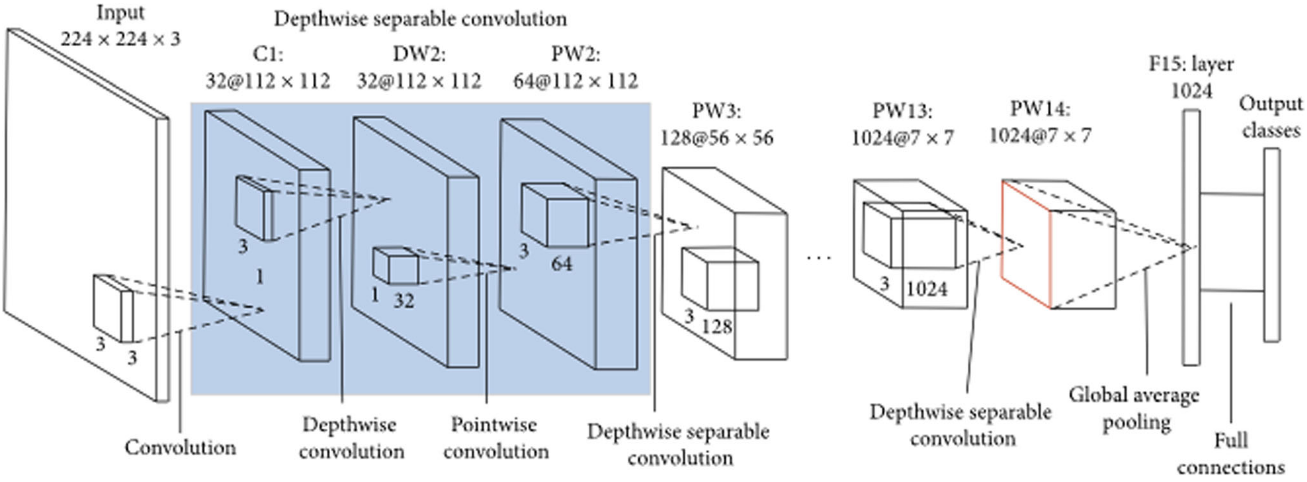
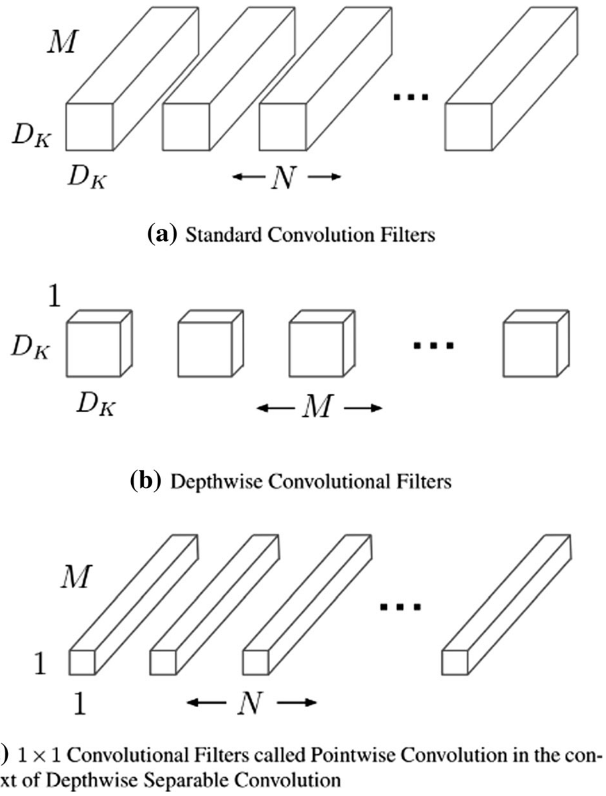


Fig. 8 MobileNet Architecture


M
D_K
D_K
$\leftarrow N \rightarrow$
1
D_K
D_K
$\leftarrow M \rightarrow$
M
1
1
$\leftarrow N \rightarrow$

(a) Standard Convolution Filters

(b) Depthwise Convolutional Filters

(c) 1×1 Convolutional Filters called Pointwise Convolution in the context of Depthwise Separable Convolution

Fig. 9 Depth-wise and point-wise convolutions

Table 4 Keras summary of the model using MobileNet as a feature extractor

Layer (type)	Output Shape	Parameters
mobilenet_1.00_224 (Model)	(None, 7, 7, 1024)	3,228,864
dense_25 (Dense)	(None, 7, 7, 1024)	1,049,600
dense_26 (Dense)	(None, 7, 7, 7)	7175

Total params: 4,285,639

Trainable params: 1,056,775

Non-trainable params: 3,228,864

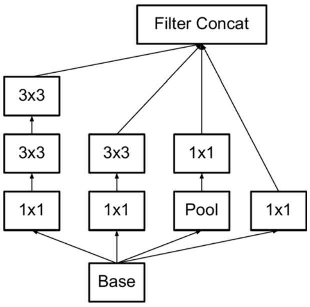


Fig. 10 Base Inception V3 module

Table 3 Structure of the MobileNet

Type/stride	Filter shape	Input size
Conv / s2	3 × 3 × 3 × 32	224 × 224 × 3
Conv dw /s1	3 × 3 × 32 dw	112 × 112 × 32
Conv /s1	1 × 1 × 32 × 64	112 × 112 × 32
Conv dw / s2	3 × 3 × 64 dw	112 × 112 × 64
Conv / s 1	1 × 1 × 64 × 128	56 × 56 × 64
Conv dw /s1	3 × 3 × 128 dw	56 × 56 × 128
Conv /s1	1 × 1 × 128 × 128	56 × 56 × 128
Conv dw / s2	3 × 3 × 128 dw	56 × 56 × 128
Conv / s1	1 × 1 × 128 × 256	28 × 28 × 128
Conv dw / s1	3 × 3 × 256 dw	28 × 28 × 256
Conv / s1	1 × 1 × 256 × 256	28 × 28 × 256
Conv dw / s2	3 × 3 × 256 dw	28 × 28 × 256
Conv / s1	1 × 1 × 256 × 512	14 × 14 × 256
5 × Conv dw /s1 Conv / s1	3 × 3 × 512 dw 1 × 1 × 512 × 512	14 × 14 × 512 14 × 14 × 512
Conv dw / s2	3 × 3 × 512 dw	14 × 14 × 512
Conv / s1	1 × 1 × 512 × 1024	7 × 7 × 512
Conv dw / s2	3 × 3 × 1024 dw	7 × 7 × 1024
Conv / s 1	1 × 1 × 1024 × 1024	7 × 7 × 1024
Avg Pool /s1	Pool 7 × 7	7 × 7 × 1024
FC / s1	1024 × 1000	1 × 1 × 1024
Sofrmax / s1	Classifier	1 × 1 × 1000

Table 5 Implementation of Inception V3

Type	Kernel size/stride	Input size
Convolution	3 × 3/2	299 × 299 × 3
Convolution	3 × 3/1	149 × 149 × 32
Convolution	3 × 3/1	147 × 147 × 32
Pooling	3 × 3/2	147 × 147 × 64
Convolution	3 × 3/1	73 × 73 × 64
Convolution	3 × 3/2	71 × 71 × 80
Convolution	3 × 3/1	35 × 35 × 192
Inception module	Three modules	35 × 35 × 288
Inception module	Five modules	17 × 17x 768
Inception module	Two modules	8 × 8 × 1,280
Pooling	8 × 8	8 × 8 × 2,048
Linear	Logits	1 × 1 × 2,048
Softmax	Output	1 × 1 × 1,000

Table 6 Keras summary of the model using Inception V3 as a feature extractor

Layer (type)	Output Shape	Parameters
inception_v3 (Model)	(None, 5, 5, 2048)	21,802,784
dense_31 (Dense)	(None, 5, 5, 1024)	2,098,176
dense_32 (Dense)	(None, 5, 5, 7)	7175

Total parameters: 23,908,135

Trainable parameters: 2,105,351

Non-trainable parameters: 21,802,784

Convolutional layers, Max pooling layers (with filter and stride sizes), and fully connected layers used in each model are explained clearly in Sect. 4.

5.1 Implementation parameters

The below Table 7 shows some of the implementation parameters for all the four models used in this work. The input shape of the three networks VGG 19, Resnet 50, and MobileNet are the same but Inception V3 is different. For all the networks, weights are initialized from ImageNet. The classifier used for the models is the Softmax classifier and the optimizer is the Adam optimizer and the loss function is categorical_crossentropy. The regularization used for all the models is Batch normalization. And some of the parameters like Dropout, Epoch size, and Batch size are the same for all four models.

6 Experimental results and discussions

Below are the test results of various models used in this work. The performance metrics used in this work are Accuracy, Sensitivity (Recall), Specificity, Precision, and F1 score. These metrics are defined in terms of true-positive (TP), false-positive (FP), false-negative (FN), and true-negative (TN). The sample confusion matrix for the calculation of TP, TN, FP and FN values are clearly mentioned in Table 8.

7 Accuracy

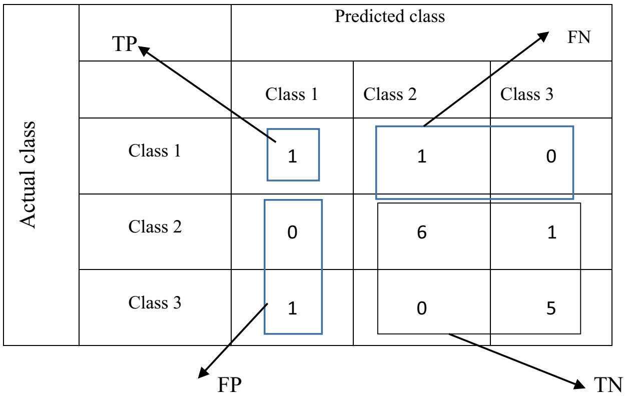
Accuracy is defined as the proportion of the number of correct samples to the number of all samples.

$$\text{Accuracy} = \frac{\text{TN} + \text{TP}}{\text{TN} + \text{TP} + \text{FN} + \text{FP}} \tag{3}$$

Table 7 Implementation Parameters

Parameter	VGG 19	Resnet 50	MobileNet	Inception V3
Input shape	(224, 224,3)	(224, 224,3)	(224, 224,3)	(299,299,3)
Weights	Initialized to ImageNet	Initialized to ImageNet	Initialized to ImageNet	Initialized to ImageNet
Epochs	50	50	50	50
Batch size	50	50	50	50
Classifier	Softmax	Softmax	Softmax	Softmax
Optimizer	Adam	Adam	Adam	Adam
Loss function	Categorical_crossentropy	Categorical_crossentropy	Categorical_crossentropy	Categorical_crossentropy
Dropout	0.5	0.5	0.5	0.5
Regularization	Batch Normalization	Batch Normalization	Batch Normalization	Batch Normalization

Table 8 Sample Confusion Matrix showing TP, TN, FP, and FN values for class 1



		Predicted class		
		Class 1	Class 2	Class 3
Actual class	Class 1	1	1	0
	Class 2	0	6	1
	Class 3	1	0	5

Where TP = model predicts the positive class accurately
TN = model predicts the negative class accurately
FP = model predicts the positive class inaccurately
FN = model predicts the negative class inaccurately

8 Sensitivity

Sensitivity is defined as the proportion between the number of true-positive cases to the total number of true-positive and false-negative cases.

$$\text{Sensitivity}(\text{Recall}) = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{4}$$

9 Specificity

The ratio of the number of true-negative cases to the total number of true-negative and false-positive cases is known as specificity.

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}} \tag{5}$$

10 Precision

The ratio of correctly predicted positive cases to the total predicted positive cases is known as precision.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{6}$$

11 F1 Score

The weighted average of precision and recall is the F1 score. The higher F1 score means the model is more accurate in doing the predictions.

$$\text{F1 Score} = 2.\frac{\text{precision} * \text{recall}}{\text{precision} + \text{recall}} \tag{7}$$

11.1 Results of VGG19 on test data

Figure 11 shows the data fitting results by using pertained VGG19 as a feature extractor.

Figures 12 and 13 show the accuracy and loss of the model. The number of epochs changes the loss and accuracy values are also changed.

Table 9 displays the confusion matrix for the test data of 148 samples. According to Table 10, the model is highly accurate in predicting the emotions of contempt and less accurate at the prediction of happy emotion.

The metrics of the model accuracy, specificity and sensitivity are calculated by using true-positive (TP), false-positive (FP), and false-negative, (FN) and true-negative (TN) values. The below table shows the performance measures of the proposed model by using VGG19 as a feature extractor.

From the above calculations, the F1 score is 0.83 and the accuracy of the model by using a pre–trained VGG19 model is 96%.

11.2 Results of Resnet50 on test data

Figure 14 shows the data fitting results by using pertained Resnet50 as a feature extractor.

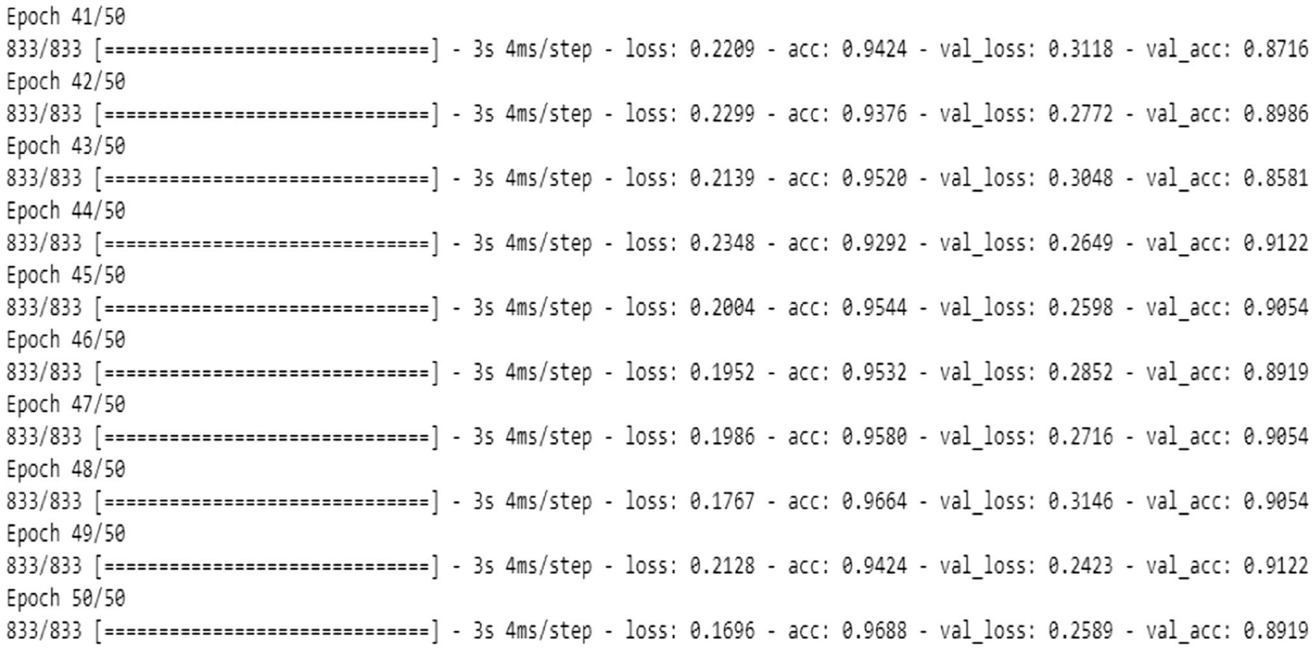
Below Table 11 displays the confusion matrix using the Resnet50 model for the test data of 148 samples. According to Table 12, the model is highly accurate in predicting the emotions of sadness and less accurate at the prediction of happy emotion.

The below table displays the performance measures of the proposed model by using Resnet50 as a feature extractor.

From the above calculations, the F1 score is 0.91 and the accuracy of the model by using a pre-trained Resnet50 model is 97.7%. Figures 15 and 16 show the accuracy and loss of the model by using Resnet50.

11.3 Results of MobileNet on test data

The below Fig. 17 shows the data fitting results by using pertained MobileNet as a feature extractor.



```
Epoch 41/50
833/833 [==============================] - 3s 4ms/step - loss: 0.2209 - acc: 0.9424 - val_loss: 0.3118 - val_acc: 0.8716
Epoch 42/50
833/833 [==============================] - 3s 4ms/step - loss: 0.2299 - acc: 0.9376 - val_loss: 0.2772 - val_acc: 0.8986
Epoch 43/50
833/833 [==============================] - 3s 4ms/step - loss: 0.2139 - acc: 0.9520 - val_loss: 0.3048 - val_acc: 0.8581
Epoch 44/50
833/833 [==============================] - 3s 4ms/step - loss: 0.2348 - acc: 0.9292 - val_loss: 0.2649 - val_acc: 0.9122
Epoch 45/50
833/833 [==============================] - 3s 4ms/step - loss: 0.2004 - acc: 0.9544 - val_loss: 0.2598 - val_acc: 0.9054
Epoch 46/50
833/833 [==============================] - 3s 4ms/step - loss: 0.1952 - acc: 0.9532 - val_loss: 0.2852 - val_acc: 0.8919
Epoch 47/50
833/833 [==============================] - 3s 4ms/step - loss: 0.1986 - acc: 0.9580 - val_loss: 0.2716 - val_acc: 0.9054
Epoch 48/50
833/833 [==============================] - 3s 4ms/step - loss: 0.1767 - acc: 0.9664 - val_loss: 0.3146 - val_acc: 0.9054
Epoch 49/50
833/833 [==============================] - 3s 4ms/step - loss: 0.2128 - acc: 0.9424 - val_loss: 0.2423 - val_acc: 0.9122
Epoch 50/50
833/833 [==============================] - 3s 4ms/step - loss: 0.1696 - acc: 0.9688 - val_loss: 0.2589 - val_acc: 0.8919
```

Fig. 11 Fitting results by using VGG19 model

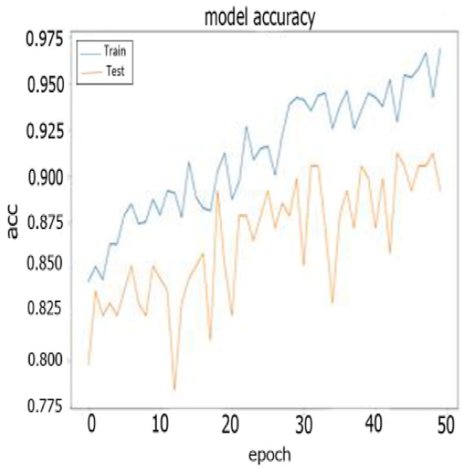


Fig. 12 accuracy by using VGG19

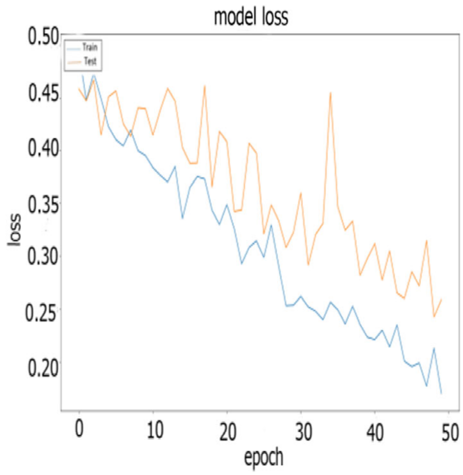


Fig. 13 loss by using VGG19

By using MobileNet as a feature extractor, Figs. 18 and 19 display the accuracy and loss of the design.

The underneath Table 13 displays the confusion matrix for the test data of 148 samples. According to Table 14, the model is highly accurate in predicting the emotions of surprise and fear and less accurate at the prediction of disgust emotion.

The below table displays the performance measures of the proposed model by using MobileNet as a feature extractor.

From the above calculations, the F1 score is 0.93 and the accuracy of the model by a using pre-trained MobileNet model is 98.5% (Fig. 20).

11.4 Results of inception V3 on test data

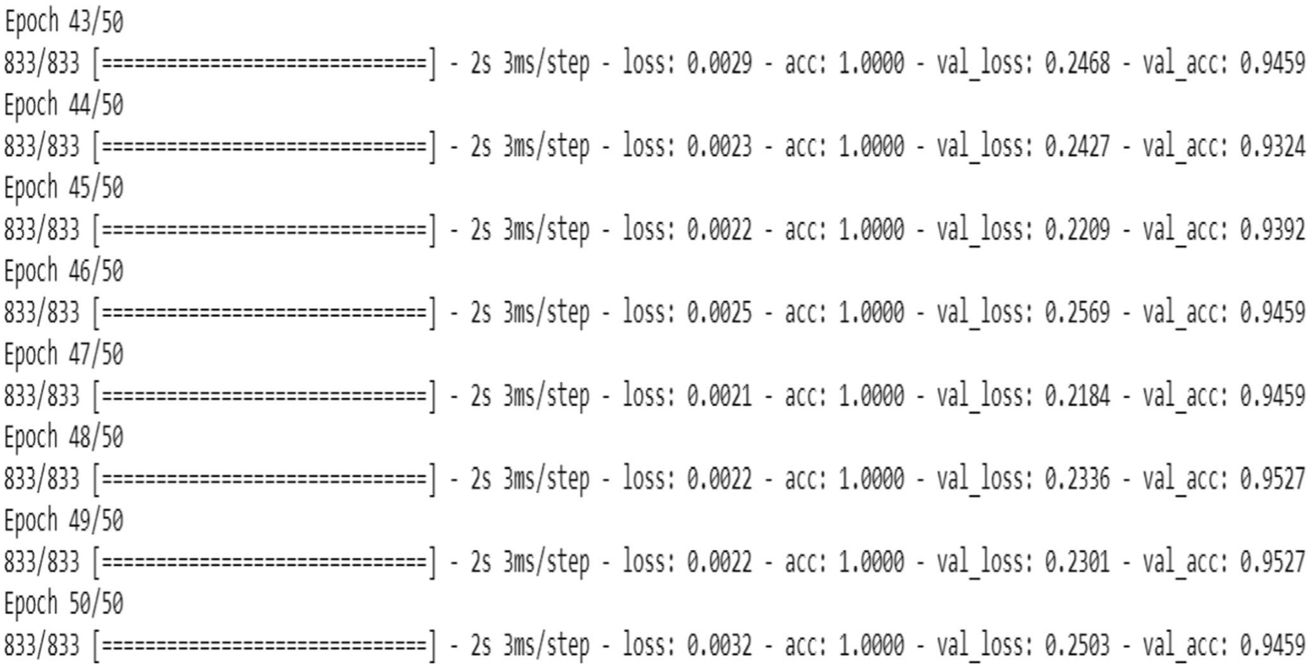
The below Figure shows the data fitting results by using pertained Inception V3 as a feature extractor.

Table 9 Confusion matrix by using VGG19 model

	Anger	Contempt	Disgust	Fear	Happy	Sad	Surprise
Anger	16	1	0	0	3	0	0
Contempt	0	6	1	0	0	0	0
Disgust	1	0	28	1	2	0	0
Fear	0	0	0	8	2	0	0
Happy	0	0	0	0	23	0	0
Sad	0	0	0	0	5	4	0
Surprise	0	0	0	0	0	0	47

Table 10 Performance measures by using the VGG19 model

	TP	TN	FP	FN	Sensitivity/Recall	Specificity	Precision	F1 score	Accuracy
Anger	16	127	1	4	0.8	0.99	0.94	0.86	0.96
Contempt	6	140	1	1	0.85	0.99	0.85	0.84	0.98
Disgust	28	115	1	4	0.87	0.99	0.96	0.90	0.96
Fear	8	137	1	2	0.8	0.99	0.88	0.833	0.97
Happy	23	113	12	0	1	0.90	0.65	0.78	0.91
Sadness	4	139	0	5	0.44	1	1	0.611	0.96
Surprise	47	101	0	0	1	1	1	1	1
Average results					0.82	0.98	0.84	0.83	0.96



```
Epoch 43/50
833/833 [==============================] - 2s 3ms/step - loss: 0.0029 - acc: 1.0000 - val_loss: 0.2468 - val_acc: 0.9459
Epoch 44/50
833/833 [==============================] - 2s 3ms/step - loss: 0.0023 - acc: 1.0000 - val_loss: 0.2427 - val_acc: 0.9324
Epoch 45/50
833/833 [==============================] - 2s 3ms/step - loss: 0.0022 - acc: 1.0000 - val_loss: 0.2209 - val_acc: 0.9392
Epoch 46/50
833/833 [==============================] - 2s 3ms/step - loss: 0.0025 - acc: 1.0000 - val_loss: 0.2569 - val_acc: 0.9459
Epoch 47/50
833/833 [==============================] - 2s 3ms/step - loss: 0.0021 - acc: 1.0000 - val_loss: 0.2184 - val_acc: 0.9459
Epoch 48/50
833/833 [==============================] - 2s 3ms/step - loss: 0.0022 - acc: 1.0000 - val_loss: 0.2336 - val_acc: 0.9527
Epoch 49/50
833/833 [==============================] - 2s 3ms/step - loss: 0.0022 - acc: 1.0000 - val_loss: 0.2301 - val_acc: 0.9527
Epoch 50/50
833/833 [==============================] - 2s 3ms/step - loss: 0.0032 - acc: 1.0000 - val_loss: 0.2503 - val_acc: 0.9459
```

Fig. 14 Fitting results by using Resnet50

Table 11 Confusion matrix by using Resnet50 model

	Anger	Contempt	Disgust	Fear	Happy	Sad	Surprise
Anger	19	1	0	0	0	0	0
Contempt	0	6	1	0	0	0	0
Disgust	1	0	30	0	1	0	0
Fear	0	0	0	10	0	0	0
Happy	0	0	1	1	21	0	0
Sad	0	0	0	0	1	8	0
Surprise	0	0	0	0	2	0	45

Table 12 Performance measures by using Resnet50 model

	TP	TN	FP	FN	Sensitivity/ Recall	Specificity	Precision	F1 score	Accuracy
Anger	19	127	1	1	0.95	0.99	0.95	0.94	0.98
Contempt	6	140	1	1	0.85	0.99	0.85	0.84	0.98
Disgust	30	114	2	2	0.93	0.98	0.93	0.92	0.97
Fear	10	137	1	0	1	0.99	0.90	0.94	0.99
Happy	21	121	4	2	0.91	0.96	0.84	0.86	0.95
Sadness	8	139	0	1	0.88	1	1	0.93	0.99
Surprise	45	101	0	2	0.95	1	1	0.97	0.98
Average results					0.92	0.98	0.92	0.91	0.977

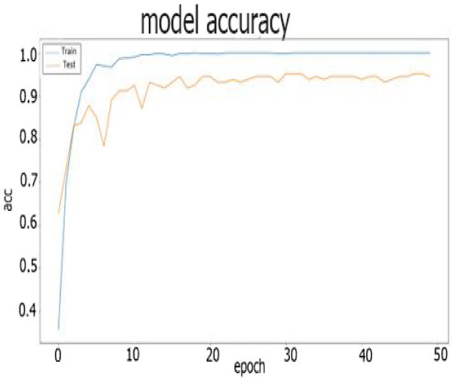


Fig. 15 accuracy by using Resnet50

Table 15 displays the confusion matrix using the Inception V3 model for the test data of 148 samples. According to Table 16, the model is highly accurate in predicting the emotions of surprise and less accurate at the prediction of happy emotion.

The below table displays the performance measures of the proposed model by using Inception V3 as a feature extractor.

From the above calculations, the F1 score is 0.75 and the accuracy of the model by using a pre-trained Inception V3 is 94.2%. The below Figs. 21 and 22 exhibits the accuracy and loss of the proposed model by using Inception V3.

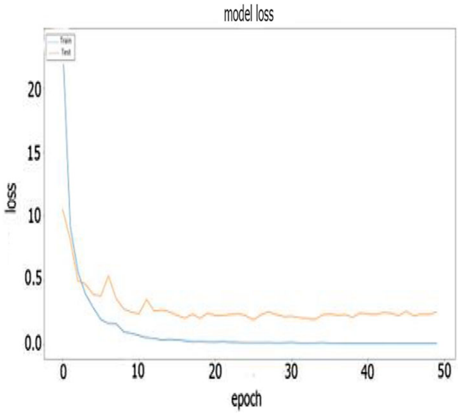
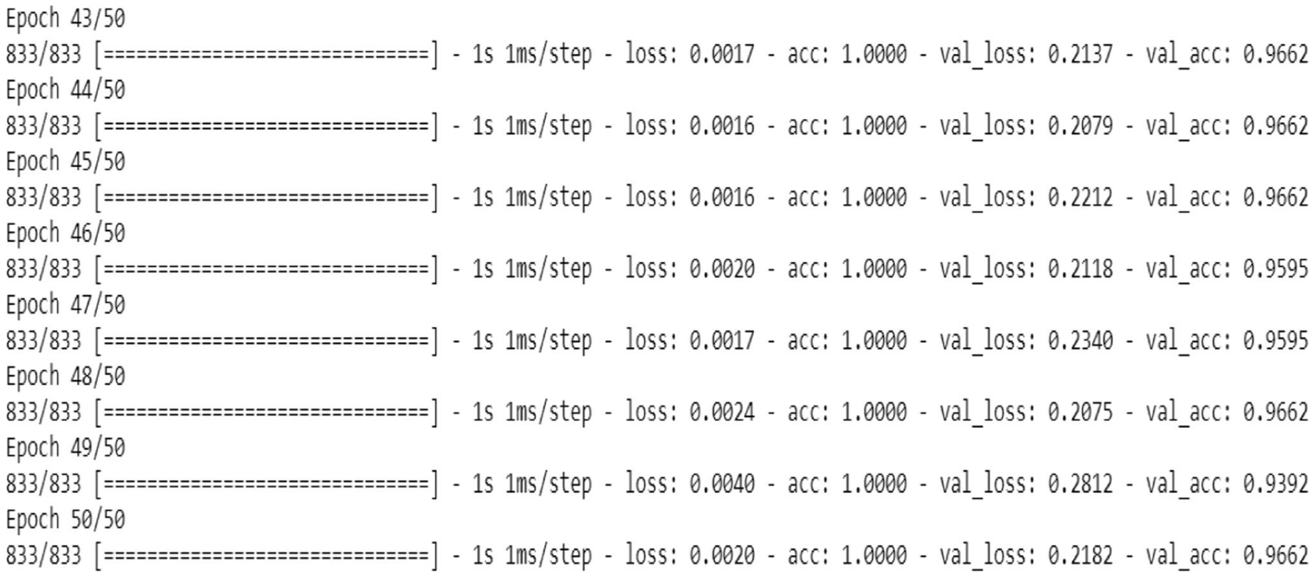


Fig. 16 loss by using Resnet50

12 Comparative analysis

12.1 Comparisons within proposed methods

In this work, four pre-trained networks of VGG19, Resnet50, MobileNet, and Inception V3 are used for recognizing emotions. The sensitivity, specificity, precision, F1 score, and accuracy values are calculated for every network. Table 17 shows the values obtained for all the networks.



```
Epoch 43/50
833/833 [==============================] - 1s 1ms/step - loss: 0.0017 - acc: 1.0000 - val_loss: 0.2137 - val_acc: 0.9662
Epoch 44/50
833/833 [==============================] - 1s 1ms/step - loss: 0.0016 - acc: 1.0000 - val_loss: 0.2079 - val_acc: 0.9662
Epoch 45/50
833/833 [==============================] - 1s 1ms/step - loss: 0.0016 - acc: 1.0000 - val_loss: 0.2212 - val_acc: 0.9662
Epoch 46/50
833/833 [==============================] - 1s 1ms/step - loss: 0.0020 - acc: 1.0000 - val_loss: 0.2118 - val_acc: 0.9595
Epoch 47/50
833/833 [==============================] - 1s 1ms/step - loss: 0.0017 - acc: 1.0000 - val_loss: 0.2340 - val_acc: 0.9595
Epoch 48/50
833/833 [==============================] - 1s 1ms/step - loss: 0.0024 - acc: 1.0000 - val_loss: 0.2075 - val_acc: 0.9662
Epoch 49/50
833/833 [==============================] - 1s 1ms/step - loss: 0.0040 - acc: 1.0000 - val_loss: 0.2812 - val_acc: 0.9392
Epoch 50/50
833/833 [==============================] - 1s 1ms/step - loss: 0.0020 - acc: 1.0000 - val_loss: 0.2182 - val_acc: 0.9662
```

Fig. 17 Fitting results by using MobileNet

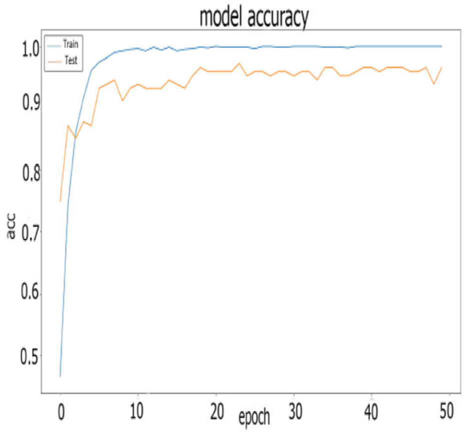


Fig. 18 accuracy by using MobileNet

12.1.1 Inference from the results

From the above results, among all the four convolutional neural networks, MobileNet achieved the highest F1 score of 0.93 and accuracy of 98% and the second Resnet50 achieved the highest F1 score of 0.91 and the accuracy of 97%. MobileNet has the advantages of reduced size, reduced parameters and faster performance so it achieved high accuracy compared to the other state-of-the-art models. Because of tackling, the vanishing gradient problem Resnet also achieved high accuracy. The drawback in VGG Net is slow in training process.

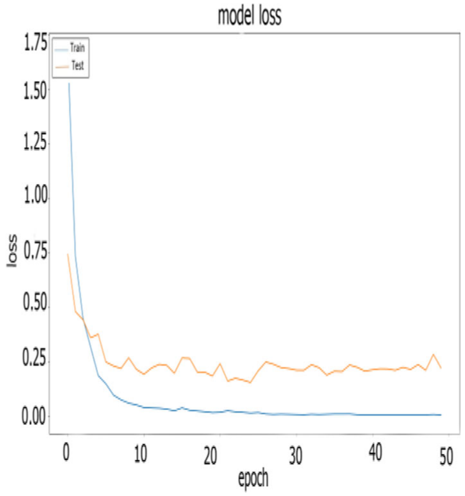


Fig. 19 loss by using MobileNet

12.2 Comparisons with other approaches

The below Table 18 displays the comparisons of various deep learning approaches by some of the researchers for facial emotion recognition problem in terms of accuracy.

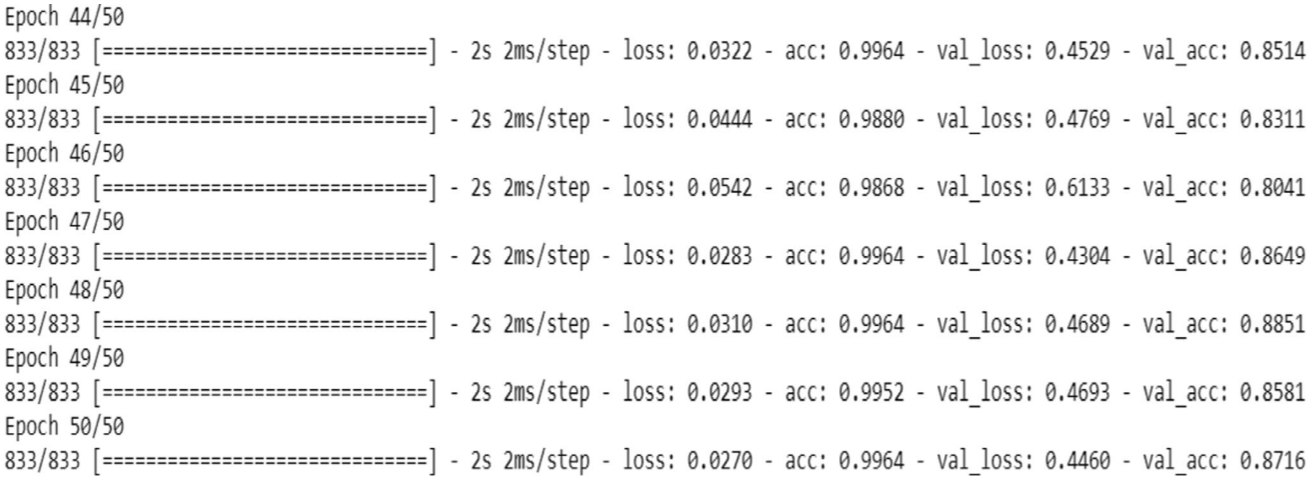
Compared to all the existing works our proposed method achieved the highest accuracy of 98% for facial emotion recognition.

Table 13 Confusion matrix by using the MobileNet

	Anger	Contempt	Disgust	Fear	Happy	Sad	Surprise
Anger	19	1	0	0	0	0	0
Contempt	0	6	1	0	0	0	0
Disgust	0	1	30	0	1	0	0
Fear	0	0	0	10	0	0	0
Happy	0	0	0	0	23	0	0
Sad	0	0	0	0	1	8	0
Surprise	0	0	0	0	0	0	47

Table 14 Performance measures by using MobileNet model

	TP	TN	FP	FN	Sensitivity/ Recall	Specificity	Precision	F1 score	Accuracy
Anger	19	128	0	1	0.95	1	1	0.97	0.99
Contempt	6	139	2	1	0.85	0.98	0.75	0.78	0.97
Disgust	30	115	1	2	0.93	0.99	0.96	0.94	0.97
Fear	10	138	0	0	1	1	1	1	1
Happy	23	123	2	0	1	0.98	0.92	0.95	0.98
Sadness	8	139	0	1	0.88	1	1	0.93	0.99
Surprise	47	101	0	0	1	1	1	1	1
Average results					0.94	0.99	0.94	0.93	0.985



```
Epoch 44/50
833/833 [==============================] - 2s 2ms/step - loss: 0.0322 - acc: 0.9964 - val_loss: 0.4529 - val_acc: 0.8514
Epoch 45/50
833/833 [==============================] - 2s 2ms/step - loss: 0.0444 - acc: 0.9880 - val_loss: 0.4769 - val_acc: 0.8311
Epoch 46/50
833/833 [==============================] - 2s 2ms/step - loss: 0.0542 - acc: 0.9868 - val_loss: 0.6133 - val_acc: 0.8041
Epoch 47/50
833/833 [==============================] - 2s 2ms/step - loss: 0.0283 - acc: 0.9964 - val_loss: 0.4304 - val_acc: 0.8649
Epoch 48/50
833/833 [==============================] - 2s 2ms/step - loss: 0.0310 - acc: 0.9964 - val_loss: 0.4689 - val_acc: 0.8851
Epoch 49/50
833/833 [==============================] - 2s 2ms/step - loss: 0.0293 - acc: 0.9952 - val_loss: 0.4693 - val_acc: 0.8581
Epoch 50/50
833/833 [==============================] - 2s 2ms/step - loss: 0.0270 - acc: 0.9964 - val_loss: 0.4460 - val_acc: 0.8716
```

Fig. 20 Fitting results by using Inception V3

Table 15 Confusion matrix by using Inception V3 model

	Anger	Contempt	Disgust	Fear	Happy	Sad	Surprise
Anger	14	1	0	0	4	0	1
Contempt	1	3	2	0	1	0	0
Disgust	1	1	26	1	2	1	0
Fear	0	0	1	8	1	0	0
Happy	1	0	0	0	21	1	0
Sad	0	0	0	0	3	6	0
Surprise	0	0	0	0	2	0	45

Table 16 Performance measures by using Inception v3model

	TP	TN	FP	FN	Sensitivity/ Recall	Specificity	Precision	F1 score	Accuracy
Anger	14	125	3	6	0.7	0.97	0.82	0.75	0.93
Contempt	3	139	2	4	0.42	0.98	0.6	0.49	0.95
Disgust	26	113	3	6	0.81	0.97	0.89	0.84	0.93
Fear	8	137	1	2	0.8	0.99	0.88	0.83	0.97
Happy	21	112	13	2	0.91	0.89	0.61	0.72	0.89
Sadness	6	137	2	3	0.66	0.98	0.75	0.69	0.96
Surprise	45	100	1	2	0.95	0.99	0.97	0.95	0.97
Average results					0.75	0.96	0.78	0.75	0.942

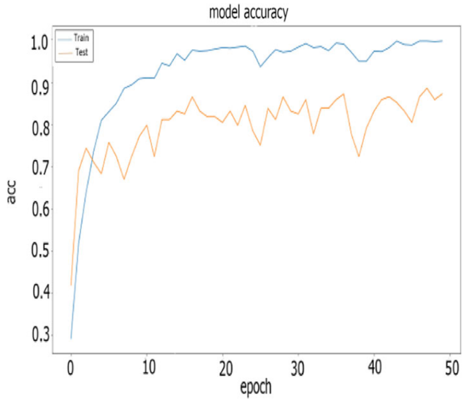


Fig. 21 accuracy by using Inception V3

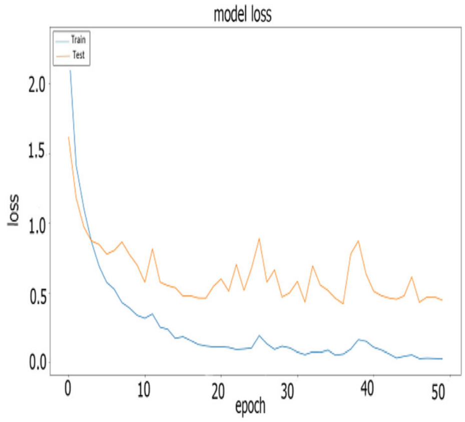


Fig. 22 loss by using Inception V3

Table 18 Comparison of Existing works

S. No	Work	Accuracy (%)
1	Milad Mohammad Taghi Zadeh et al. [49]	97.1%
2	DY Liliana [50]	92.81%
3	David Orozco [33]	90%
4	Yijun Gan et al. [51]	64.2%
5	Akash Saravanan et al. [52]	60%

13 Conclusions

This paper presented facial emotion recognition system using transfer learning approaches. In this work, pre-trained convolutional neural networks of VGG19, Resnet50, Inception V3 and MobileNet that are trained on ImageNet database, are used for facial emotion recognition. The experiments were tested using the CK + database. The accuracy achieved using the VGG19 model is 96%, Resnet50 is 97.7%, Inception V3 is 98.5%, and MobileNet is 94.2%. Among all four pre-trained networks, MobileNet achieved the highest accuracy. In future, these networks will be implemented for speech and EEG signals to recognize the emotions.

Table 17 Comparisons within proposed networks

S. No	Network	Sensitivity	Specificity	Precision	F1 score	Accuracy
1	VGG19	0.82	0.98	0.84	0.83	0.96
2	Resnet50	0.92	0.98	0.92	0.91	0.97
3	MobileNet	0.94	0.99	0.94	0.93	0.98
4	Inception V3	0.79	0.96	0.78	0.75	0.94

References

1. Kołakowska A, Landowska A, Szwoch M, Szwoch W, Wrobel MR (2014) Emotion recognition and its applications. In: Human-computer systems interaction: backgrounds and applications, pp 51–62
2. Dubey M, Singh L (2016) Automatic emotion recognition using facial expression: a review. Int Res J Eng Technol (IRJET) 3:488
3. Tian Y, Kanade T, Cohn JF (2011) Facial expression recognition. In Handbook of face recognition. Springer, London, pp 487–519
4. Bansal S, Nagar P (2015) Emotion recognition from facial expression based on bezier curve. Int J Adv Inf Technol 5(4):5
5. Senthilkumar TK, Rajalingam S, Manimegalai S, Srinivasan VG (2016) Human facial emotion recognition through automatic clustering based morphological segmentation and shape/orientation feature analysis. In: 2016 IEEE international conference on computational intelligence and computing research (ICCIC) pp. 1–5. IEEE
6. Guo X, Zhang X, Deng C, Wei J (2013) Facial expression recognition based on independent component analysis. J Multimed 8(4):402–409
7. Wang N, Li Q, Abd El-Latif AA, Peng J, Niu X (2013) Multibiometrics fusion for identity authentication: dual iris, visible and thermal face imagery. Int J Secur Appl 7(3):33–44
8. Wang N, Li Q, Abd El-Latif AA, Peng J, Niu X (2013) Two-directional two-dimensional modified Fisher principal component analysis: an efficient approach for thermal face verification. J Electron Imaging 22(2):023013
9. Abd El-Latif AA, Hossain MS, Wang N (2019) Score level multibiometrics fusion approach for healthcare. Clust Comput 22(1):2425–2436
10. Mansour AH, Salh GZA, Alhalemi AS (2014) Facial expressions recognition based on principal component analysis (PCA). arXiv preprint arXiv:1506.01939.
11. Shan C, Gong S, McOwan PW (2009) Facial expression recognition based on local binary patterns: a comprehensive study. Image Vis Comput 27(6):803–816
12. Wang N, Li Q, El-Latif AA, Peng J, Niu X (2013) A novel multibiometric template security scheme for the fusion of dual iris, visible and thermal face images. J Comput Inf Syst 9(19):1–9
13. Michel P, El Kaliouby R (2005) Facial expression recognition using support vector machines. In: Paper presented at 10th international conference on human-computer interaction, Crete, Greece
14. Wang J, Wang S, Ji Q (2014) Early facial expression recognition using hidden Markov models. In: Paper presented at 22nd International conference on pattern recognition pp. 4594–4599. IEEE
15. Thakare PP, Patil PS (2016) Facial expression recognition algorithm based on KNN classifier. Int J Comput Sci and Netw 5(6):941
16. Salmam FZ, Madani A, Kissi M (2016) Facial expression recognition using decision trees. In: 2016 13th international conference on computer graphics, imaging and visualization (CGiV). IEEE. pp. 125–130
17. Nonis F, Dagnes N, Marcolin F, Vezzetti E (2019) 3D Approaches and challenges in facial expression recognition algorithms—A literature review. Appl Sci 9(18):3904
18. Jain N, Nguyen TN, Gupta V, Hemanth DJ. (2021) Dental X-ray image classification using deep neural network models. Ann Telecommun
19. Dash R, Nguyen TuN, Cengiz K, Sharma A (2021) FTSVR: fine-tuned support vector regression model for stock predictions. Neural Comput Appl. https://doi.org/10.1007/s00521-021-05842-w
20. Vu D, Nguyen T, Nguyen TV, Nguyen TN, Massacci F, Phung PH (2019) A convolutional transformation network for malware classification. In 2019 6th NAFOSTED conference on information and computer science (NICS), pp. 234–239
21. Li S, Deng W (2018) Deep facial expression recognition: a survey. arXiv preprint arXiv:1804.08348
22. Pitaloka DA, Wulandari A, Basaruddin T, Liliana DY (2017) Enhancing CNN with preprocessing stage in automatic emotion recognition. Proc Comput Sci 116:523–529
23. Ng HW, Nguyen VD, Vonikakis V, Winkler S (2015) Deep learning for emotion recognition on small datasets using transfer learning. In: Proceedings of the 2015 ACM on international conference on multimodal interaction. pp. 443–449
24. Xu M, Cheng W, Zhao Q, Ma L, Xu F (2015) Facial expression recognition based on transfer learning from deep convolutional networks. In: Proceedings of 11th international conference on natural computation, Zhangjiajie, China. pp 702–708
25. Fan Y, Lam JC, Li VO (2018) Multi-region ensemble convolutional neural network for facial expression recognition. In: Proceedings of International conference on artificial neural networks, Rhodes, Greece. pp 84–94
26. Wang Y, Li Y, Song Y, Rong X (2019) Facial expression recognition based on auxiliary models. Algorithms 12(11):227
27. Nordén F, von Reis Marlevi F (2019) A comparative analysis of machine learning algorithms in binary facial expression recognition (Dissertation). http://www.diva-portal.org/smash/record.jsf?pid=diva2%3A1329976&dswid=3676
28. Jyostna Devi B, Veeranjaneyulu N (2019) Facial emotion recognition using deep cnn based features. Int J Innov Technol Explor Eng (IJITEE), Vol. 8, No. 7
29. Nithya Roopa S (2019) Emotion recognition from facial expressions using deep learning. Int J Eng Adv Technol (IJEAT) 8(6S):91–65
30. Sreelakshmi P, Sumithra (2019) Facial expression recognition to robust to partial occlusion using MobileNet. Int J Eng Res Technol (IJERT) Vol. 8. No. 06
31. Ravi A (2018) Pre-trained convolutional neural network features for facial expression recognition. arXiv preprint arXiv:1812.06387
32. Shaees, S, Naeem H, Arslan M, Naeem MR, Ali SH, Aldabbas H (2020) Facial emotion recognition using transfer learning. In: 2020 International conference on computing and information technology (ICCIT-1441). IEEE. pp. 1–5
33. Gulati N, Arun Kumar D (2020) Facial expression recognition with convolutional neural networks. Int J Future Gener Commun Netw 13(3):1923–1931
34. Ozdemir MA, Elagoz B, Alaybeyoglu A, Sadighzadeh R, Akan A (2019) Real time emotion recognition from facial expressions using CNN architecture. In: Proceedings of International Conference on medical technologies national congress, Kusadasi, Turkey. pp 1–4
35. Dhankhar P (2019) ResNet-50 and VGG-16 for recognizing Facial Emotions. Int J Innov Eng Technol 13(4):126–130
36. Picard RW (1999) Affective computing for HCI. In: HCI (1): 829–833
37. Daily SB, James MT, Cherry D, Porter III JJ, Darnell SS, Isaac J, Roy T (2017) Affective computing: historical foundations, current applications, and future trends. In: Emotions and affect in human factors and human-computer interaction, vol. 1, pp 213–231
38. Tao J, Tan T (2005) Affective computing: a review. In: International conference on affective computing and intelligent interaction. Springer, Berlin, Heidelberg, pp. 981–995
39. Cannon WB (1927) The James-Lange theory of emotions: A critical examination and an alternative theory. Am J Psychol 39(1/4):106–124

40. Lazarus RS, Averill JR, Opton Jr, EM (1970) Towards a cognitive theory of emotion. In: Feelings and emotions. Academic Press. pp. 207–232
41. Ekman P (1992) An argument for basic emotions. Cogn Emot 6(3–4):169–200
42. https://en.wikipedia.org/wiki/Emotion_classification
43. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556
44. Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, Huang Z, Karpathy A, Khosla A, Bernstein M, Berg AC (2015) Imagenet large scale visual recognition challenge. Int J Comput Vision 115(3):211–252
45. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778
46. Howard AG, Zhu M, Chen B, Kalenichenko D, Wang W, Weyand T, Andreetto M, Adam H (2017) Mobilenets: efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861.
47. Sifre L, Mallat S (2014) Rigid-motion scattering for image classification. Ph. D. thesis
48. Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z (2016) Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2818–2826
49. Zadeh MMT, Imani M, Majidi B (2019) Fast facial emotion recognition using convolutional neural networks and Gabor filters. In: 5th conference on knowledge based engineering and innovation (KBEI) IEEE. pp. 577–581
50. Liliana DY (2019) Emotion recognition from facial expression using deep convolutional neural network. J Phys Conf Ser 1193(1):012004
51. Gan Y (2018) Facial expression recognition using convolutional neural network. In: Proceedings of the 2nd international conference on vision, image and signal processing. pp. 1–5
52. Saravanan A, Perichetla G, Gayathri DK (2019) Facial emotion recognition using convolutional neural networks. arXiv preprint arXiv:1910.05602.