Neural Computing and Applications (2021) 33:9125–9136
https://doi.org/10.1007/s00521-020-05676-y

ORIGINAL ARTICLE

FER-net: facial expression recognition using deep neural net

Karnati Mohan[1] · Ayan Seal[1,2] · Ondrej Krejcar[2,3] · Anis Yazidi[4]

Received: 13 April 2020 / Accepted: 28 December 2020 / Published online: 9 January 2021

Abstract
Automatic facial expression recognition (FER) is one of the most challenging tasks in computer vision. FER admits a wide range of applications in human–computer interaction, behavioral psychology, and human expression synthesis. Extensive works have been reported in this field, mainly, based on handcrafted features. However, it is a challenging task to accurately extract all the correlated handcrafted features due to the effect of variations caused by emotional state. Therefore, there is a quest for further research on accurately extracting relevant features that can capture changes in facial expressions (FEs) with high fidelity. In this study, we propose FER-net: a convolution neural network to distinguish FEs efficiently with the help of the softmax classifier. We implement our method FER-net along with twenty-one state-of-the-art methods and test them on five benchmarking datasets, namely FER2013, Japanese Female Facial Expressions, Extended CohnKanade, Karolinska Directed Emotional Faces, and Real-world Affective Faces. Seven FEs, namely neutral, anger, disgust, fear, happiness, sadness, and surprise, are considered in this work. The average accuracies on these datasets are 78.9%, 96.7%, 97.8%, 82.5% and 81.68%, respectively. The obtained results demonstrate that FER-net is preeminent in comparison with twenty-one state-of-the-art methods.

Keywords Facial expression recognition · Convolution neural network · Softmax classifier

1 Introduction

Automatic facial expression recognition (FER) system is a technology capable of identifying facial expressions (FEs) by analyzing visual cues or features that are extracted from a digital image or a video frame. Recently, FER has elicited much research attention because of its potential applications in human–computer interaction (HCI) [1]. FE also plays an imperative role in human behavior understanding [2], mental disorder detection [3], cognition of human emotions [4], safe driving [5], photo-realistic human expression synthesis [6], computer graphics animation [7] and other similar tasks. A typical FER consists of three steps, namely face detection, feature extraction, and classification of FEs. In simple terms, the face is detected and cropped from a scene or a video frame in the first step. Viola-Jones object detection framework is frequently used to detect and crop face from an image [8]. In case the cropped face region is tilted, a rotation rectification is required during the preprocessing step [9, 10]. Generally, facial landmarks such as eyes, nose, and mouth corners help to rotate the titled face. In the second step, features are extracted from the cropped face region and concatenated to form a feature vector. Then the feature vector is fed into a machine learning algorithm to identify FE. Some of the well-known texture-based feature extraction methods include Gabor texture [11], local binary pattern (LBP) [12], and histogram of oriented gradients [13]. When it comes to appearance-based feature extraction methods, some of the well-known methods are based on pixel intensity [14], landmark points from the local regions [15], extracting the motion features optical flow [16], motion history images [17], and volume LBP [18].

✉ Ayan Seal
ayan@iiitdmj.ac.in

[1] PDPM Indian Institute of Information Technology, Design and Manufacturing, Jabalpur 482005, India

[2] Faculty of Informatics and Management, Center for Basic and Applied Research, University of Hradec Kralove, Rokitanskeho 62, 50003 Hradec Kralove, Czech Republic

[3] Malaysia-Japan International Institute of Technology (MJIIT), Universiti Teknologi Malaysia, Jalan Sultan Yahya Petra, 54100 Kuala Lumpur, Malaysia

[4] Research Group in Applied Artificial Intelligence, Oslo Metropolitan University, 460167 Oslo, Norway

However, all the above-mentioned feature extraction methods are also called handcrafted features, which were exploited in the second step of FER and in the last step a classifier such as AdaBoost [19], support vector machines [20], k-nearest neighbor [21], and others [22–25] was applied. Even though many works have been reported in the recent past, FER is still a challenging task due to occlusions in face regions, illumination changes, and head deflection. In [26], Chen et al. proposed a feature descriptor called HOG from Three Orthogonal Planes (HOG-TOP) to extract dynamic textures from video sequences that are able to characterize facial appearance changes. Moreover, the challenging task is to extract accurately all the correlated handcrafted features due to the effect of variations caused by emotional state. The latter problem may negatively affect not only the performance of the face detection step but also the recognition of FEs. Thus, handcrafted features have clear shortcomings that limit their performance in identifying FEs.

These shortcomings can be overcome by resorting to deep learning (DL) techniques [27, 28]. In recent years, DL techniques gave superior performances in extracting distinctive features from face images leading as a consequence to increased classification performance of FEs. However, DL techniques entail a large number of methods and architectures and require appropriate parameter tuning and optimization. Some of most known adopted DL techniques for FER include deep belief networks [29], deep convolutional neural networks (DCNN) [28, 30], AlexNet, VGG19, and ResNet150-based transfer learning methods [31], shallow CNN (SCNN) and major CNN (MCNN) [32], multi-channel CNN [33, 34], action units inspired deep networks [35], extended deep FER [36], fusing-multi-stream deep networks [37, 38], three CNN networks [39], local direction-based robust features and deep belief network [40], ensembles of DCNNs [41], DCNN for binary classification (DCNN-BC) of two FEs, namely happy and sad [42], identity-aware CNN (IACNN) FER models [43, 44], deep metric learning (2B (N+M)Softmax) for jointly optimizing a deep metric and softmax loss [43], attentional DCNN named a Deep-Emotion in [45], weighted fusion of three CNN sub-networks (WFTS) [46], and weighted fusion of appearance feature-based CNN and geometric features (GF) [47]. For solving the FER problem, a spatiotemporal feature (STF) representation learning is presented by encoding the characteristics of FEs using DCNN and long short-term memory (LSTM) [48]. Benitez-Quiroz et al. [49], considered a FER system based on discriminant color features and a Gabor transform (CF + GT) based algorithm to make the algorithm resilient to variance in the timing of facial action units (AUs). In [50], Zhang et al. developed a broad learning system for FER. In [51], Zhao et al. presented a weighted mixture of double channel (WMDC) method for FER where a shallow CNN is considered as one channel on LBP (SCNN-LBP) images as well as grayscale (SCNN-gray) images, while another channel is a partial VGG16 (P-VGG16).

However, no perfect DL model which identifies FEs accurately is available as of now. Moreover, it is unclear from the literature whether legacy works performing well on datasets trained in lab-controlled environments would provide satisfactory performance on real-time datasets such as Facial Expression Recognition 2013 (FER2013) and Real-world Affective Faces (RAF). In this study, a DL network for recognizing FEs called FER-net is proposed, which focuses on extracting useful features from gray-scale face images. Softmax classifier is used for classifying FEs. Five benchmarking datasets, namely FER2013 [52], Japanese Female Facial Expressions (JAFFE) [53], Extended CohnKanade (CK+) [54], Karolinska Directed Emotional Faces (KDEF) [55], and RAF [56, 57], are considered to evaluate the effectiveness of the proposed model over twenty-one state-of-the-art DL models. Moreover, each dataset contains seven basic expressions, namely neutral (NE), anger (AN), disgust (DI), fear (FE), happiness (HA), sadness (SA), and surprise (SU).

The remainder of this paper is organized as follows. Section 2 describes our proposed model, FER-net. Experimental results and discussion are illustrated in Sect. 3. Section 3 also presents a comparative evaluation of the proposed FER-net with twenty-one state-of-the-art models. Finally, Sect. 4 concludes the work.

2 Proposed model: FER-net

Even though several DL networks exist for FER, most of them do not pan well when they are challenged with data that require a thorough understanding of the inherent features for FER. The proposed FER-net is specifically designed in order to learn the detailed local features like eyes and mouth corners that are exhibited by different FEs in face images. Micro-FEs play an important role in FER. These micro-expressions occur in everyone, often unconsciously and in an unnoticeable manner to an interlocutor. These expressions are essential for identifying the emotion of individual subjects. Traditional CNN-based methods suffer from the overfitting problem on small datasets. However, datasets with reliable expressions are relatively difficult to collect and tend to be small. Traditional CNN models are known to achieve supreme performance in imagenet classification, and examples of such models include LeNet [58], VGG16 [59], VGG19 [60], ResNet50 [61], GoogLeNet [62], Dense121 [63], and XceptionNet [64]. However, these DL models are specific to the datasets they trained on. In other words, although these models can learn common features from the datasets they were trained on which allow them to differentiate among a diversity of categories, they may fail to perform well on different datasets. Furthermore, they do not train well when the contrast between images isn't pronounced. More complex networks are usually able to learn deep features, but they often result in overfitting the model as the number of involved parameters is high. We propose a simple CNN network to classify static expressions that perform well even on small datasets. The detailed architecture of FER-net is explained in the following subsections:

2.1 Model architecture

The schematic block diagram of the FER-net is shown in Fig. 1. It consists of four convolution layers (C1, C2, C3, and C4), four max-pooling layers (P1, P2, P3, and P4), and two fully connected layers (F1 and F2). Batch-normalization is applied to the outputs of four convolutional layers and the two fully connected layers. The first and second convolution layers consist of 64 and 128 neurons, respectively. On the other hand, the last two convolution layers possess 512 neurons. Moreover, the first fully connected layer consists of 512 neurons, whereas only 256 neurons are present in the second fully connected layer. Each convolution layer has 3×3 kernel except the second convolutional layer. The second convolution layer and max-pooling layer have 5×5 and 2×2 kernels, respectively. Moreover, these filters are used to capture the enriched contextual information and allow the model to learn true edge variations. The feature map, F, is obtained by the first convolutional layer, which is known as a low-level feature. The remaining convolutional layers produce feature maps, which denote high-level features such as edges, corner points, color from the face region automatically. Here, both low-level and high-level features are considered for classifying FEs. The task of a convolution layer is to perform convolution operation on face image, I, with the help of a kernel, W_i. Further, convolved features are fed into the activation function, which is in this case a rectified linear unit (ReLU). Mathematically, a convolution operation can be represented by Eq. 1.

$$F(i) = R(I * W_i), \tag{1}$$

where i is the layer in consideration, the asterisk represents convolution operation, and $R(.)$ is the activation function. Here, ReLU is used as an activation function. ReLU is a piece-wise linear function. If the input, y, is positive, then ReLU produces y as an output; otherwise, it generates 0. ReLU is normally used in the hidden layer because it is better than all the available activation functions such as sigmoid, tanh, etc. [65]. ReLU is also known as a ramp function. Equation 2 is the mathematical representation of ReLU.

$$\mathrm{Relu}(y) = \begin{cases} 0 & \text{if}(y<0) \\ y & \text{otherwise} \end{cases} \tag{2}$$

Batch normalization is applied to normalize the output of the input layer and hidden layers by adjusting the mean and the scale of the activation functions because a high learning rate can be achieved without causing a vanishing gradient

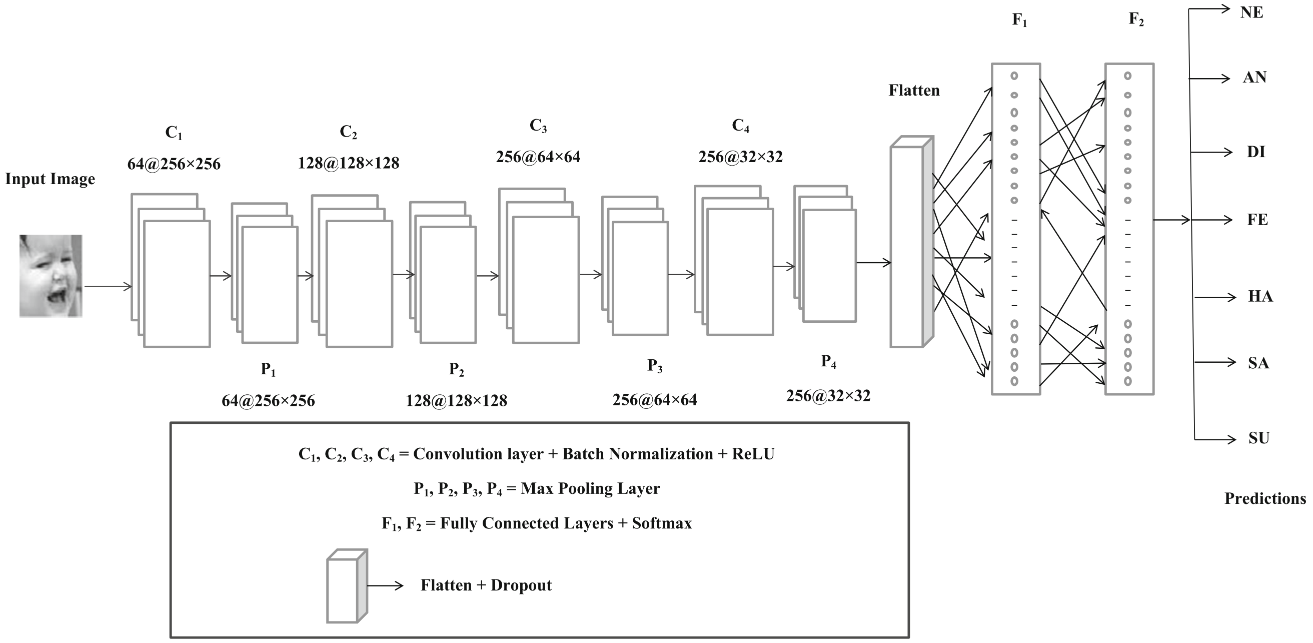


Fig. 1 Schematic block diagram of the proposed FER-net

problem by virtue of the batch-normalization. It gives better performance after the activation function. However, normalization is performed on the output of the input layer so that it always feeds immediately to the next layer. Pooling operation is applied to convolved feature maps obtained by Eq. 1 to reduce the overfitting problem. The pooling is also known as subsampling. It is able to reduce the spatial representation of an image by reducing the number of parameters associated with CNN. Three different types of pooling operations are mainly available such as max-pooling, min-pooling, and average-pooling. The max-pooling is used in FER-net. The example workflow of max-pooling is shown in Fig. 2. The size of the max-pooling kernel is 2×2.

Finally, the output of the second fully connected layer is fed into the softmax layer. The softmax activation function is applied in the dense layers of FER-net architecture. Softmax is used to calculate the probabilities of the predicted classes. The class with the highest probability is considered as an output. The mathematical representation of the softmax function is shown in Eq. 3.

$$S_x = \frac{e^{z_x}}{\sum_{y=1}^{m} e^{z_y}}, \tag{3}$$

where e^{z_x} and e^{z_y} represent the probability of belonging to the categories of x and y, respectively, whereas m denotes the number of classes. In this work, the value of m is equal to 7 because seven facial expressions are considered.

2.2 Model training

Five experiments are conducted based on the number of datasets. All the datasets are divided into three parts mainly for training sets, validation sets, and testing sets separately. The ratio of dividing datasets for training, validation, and testing is 80%, 10%, and 10%, respectively. Then FER-net is trained using the train set; moreover, while training the model too many epochs may lead to overfitting, too few epochs may lead to underfitting the model. In this work, the early stopping technique is used to address the aforementioned problem. However, early stopping is a technique that stops training once the model performance is not improving on the validation dataset. While training the FER-net, adjusting the learning rate along the way makes it faster to fit the model. Therefore, the training accuracy will increase, but the testing accuracy will decrease. This is also known as over-fitting. Dropout is added to overcome the over-fitting problem by shutting down some of the neurons in FER-net while training. Dropout is applied to each convolution layer of 0.25 and 0.5 to fully connected layers. Moreover, the loss function measures the difference between the predicted and the actual outputs. The loss function gives us feedback on how well the model works. Categorical-cross entropy is used to measure the loss in this work. The mathematical representation of the loss function is shown in Eq. 2.2.

$$L(\varphi) = \frac{1}{M}\sum_{x=1}^{M} cost(z^x, \hat{z}_x)$$
$$= -\frac{1}{M}\sum_{x=1}^{M}[z_x \log(\hat{z}_x) + (1 - z_x)\log(1 - \hat{z}_x)],$$

where z^x and $\hat{z}_x$ are the actual and predicted classes, respectively, and M denotes the number of samples, i.e., the number of images in a dataset. One of the aims of a CNN is to find the weight parameters, which minimize the loss, L. There are optimization techniques that update the weight parameters by optimizing the loss, viz., gradient descent, stochastic gradient descent (SGD), adaptive gradient descent (Adagrad), root means square propagation (RMSprop), adaptive moment estimation (Adam). In this study, adam is used for optimization and weight update. The various hyperparameters are reported in Table 1.

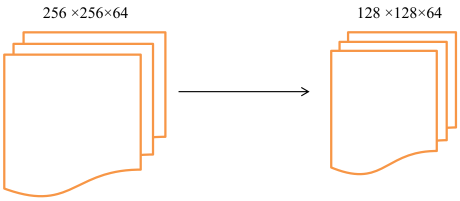


Fig. 2 Illustration of working principle of max-pooling

3 Experimental results and discussion

3.1 Environment setting

Keras framework, anaconda development platform, and Python language are considered for the implementation of the proposed method. The specifications of the system are 16GB GPU RAM, 2560 Cuda cores, 256-bit memory interface, GDDR5X as memory type, 288.5 GB/s bandwidth, NVIDIA Quadro P5000 as the graphic processor, and we used python 3.6.5.

Table 1 Various hyperparameters used for training

Name	Parameter
Optimization	Adam
Batch size	64
Learning rate	0.001
Weight decay	1e−6

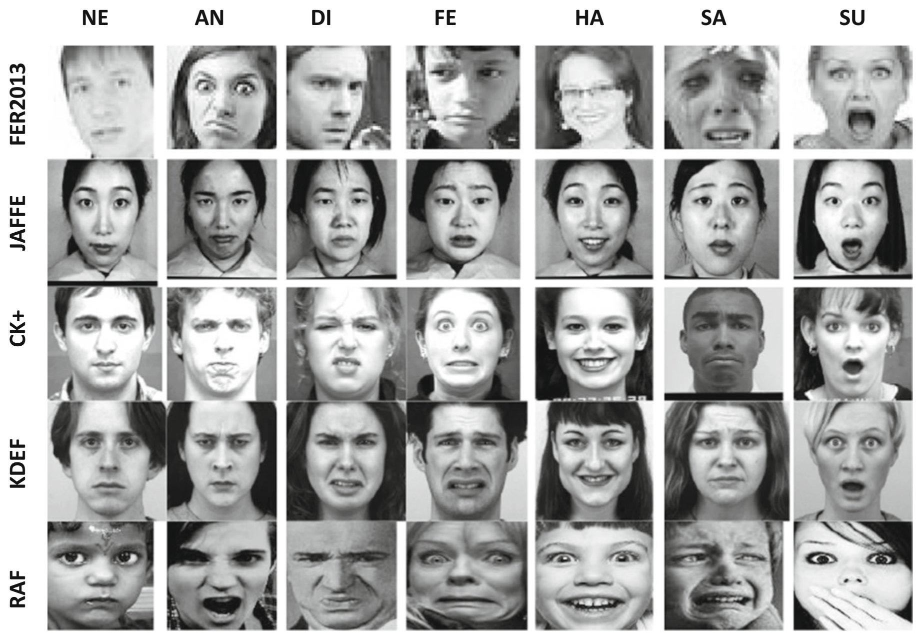


Fig. 3 Some of the sample images of each dataset [52–56] are shown row-wise, whereas identical expressions are portrayed in column

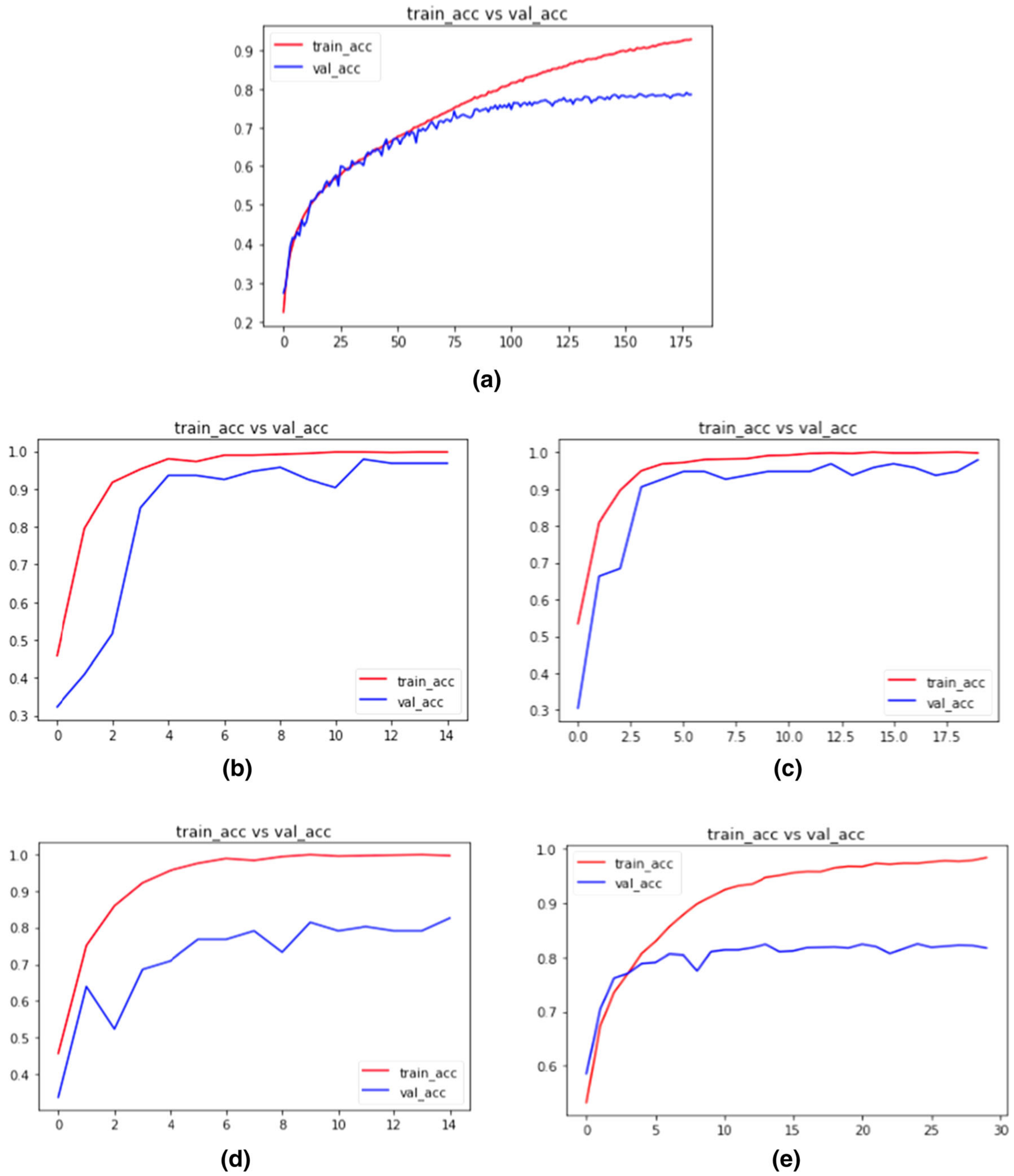
Table 2 Statistical information of five datasets

Dataset	Number of images							Total
	NE	AN	DI	FE	HA	SA	SU	
Before augmentation								
FER2013	3095	440	4097	7215	4830	3180	4965	27,822
JAFFE	30	30	29	31	31	31	30	212
CK+	50	47	61	24	59	28	62	331
KDEF	70	70	70	70	70	70	70	490
RAF	3204	867	877	355	5957	2460	1463	15,183
After augmentation								
FER2013	6995	740	7097	10,215	7830	6180	7965	47,022
JAFFE	130	130	129	131	131	131	130	912
CK+	150	147	161	124	159	128	162	1031
KDEF	120	120	120	120	120	120	120	840
RAF	3204	867	877	355	5957	2460	1463	15,183

3.2 Datasets description

In this study, five publicly available benchmarking datasets, namely FER2013, JAFFE, CK+, and KDEF, and RAF, are considered to validate the FER-net. All the datasets are considered, viz. seven facial expressions, namely NE, AN, SA, HA, FE, SU, and DI. The first, second, third, fourth, and fifth rows of Fig. 3 show some of the sample images of FER2013, JAFFE, CK+, KDEF, and RAF, respectively. All the images are of 256 × 256 pixels except images present in FER2013 and RAF dataset.

So, all the images of FER2013 are resized from 48 × 48 to 256 × 256 pixels using bilinear interpolation in the preprocessing step. Similarly, all the images of the RAF dataset are resized from 100 × 100 to 256 × 256 using the same algorithm. Datasets statistics are reported in the upper half of Table 2. However, a large number of data are required for training CNN. Image processing techniques such as translation, scaling, rotation, flipping the images vertically and horizontally, and adding noise to the images are applied to increase the size of the datasets. Statistical information of each of the datasets after augmentation is reported in the lower half of Table 2.



(a)

(b)

(c)

(d)

(e)

Fig. 4 Training and validation accuracies of **a** FER2013, **b** JAFFE, **c** CK+, **d** KDEF, and **e** RAF datasets

	NE	AN	DI	FE	HA	SA	SU
NE	529	14	40	38	41	9	44
AN	6	317	0	4	3	0	1
DI	54	9	482	25	55	37	50
FE	22	3	14	900	29	18	34
HA	56	6	67	39	561	6	73
SA	9	1	28	12	5	544	12
SU	45	2	13	45	73	6	592

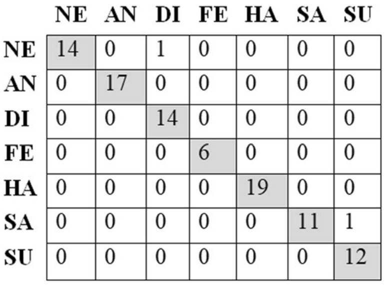
Fig. 5 Confusion matrix for FER2013 dataset

3.3 Results and discussion

In this work, four popularly used evaluation metrics, namely accuracy (A), precision (P), recall (R), and f1-score (F) [66–72], are considered in this study to compare the performance of the proposed FER-net over twenty-one state-of-the-art models. However, the values of all the metrics are computed from the confusion matrix. Figure 4 shows the training and validation accuracies of the above-mentioned datasets. Red and blue colors are used to

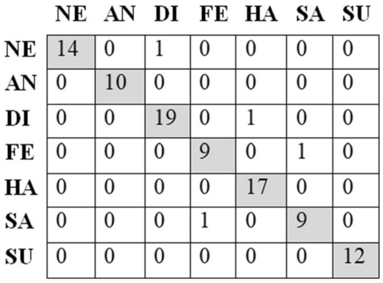
Table 3 Classification report for FER2013 dataset

Classes	A	P	R	F	Support
NE	0.73	0.73	0.74	0.74	715
AN	0.95	0.90	0.96	0.93	331
DI	0.67	0.75	0.68	0.71	712
FE	0.88	0.85	0.88	0.86	1020
HA	0.69	0.73	0.69	0.71	808
SA	0.89	0.88	0.89	0.88	611
SU	0.76	0.73	0.76	0.75	776



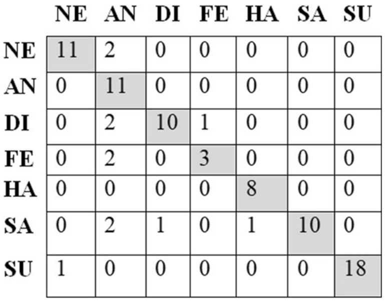
	NE	AN	DI	FE	HA	SA	SU
NE	14	0	1	0	0	0	0
AN	0	17	0	0	0	0	0
DI	0	0	14	0	0	0	0
FE	0	0	0	6	0	0	0
HA	0	0	0	0	19	0	0
SA	0	0	0	0	0	11	1
SU	0	0	0	0	0	0	12

Fig. 6 Confusion matrix for JAFFE dataset



	NE	AN	DI	FE	HA	SA	SU
NE	14	0	1	0	0	0	0
AN	0	10	0	0	0	0	0
DI	0	0	19	0	1	0	0
FE	0	0	0	9	0	1	0
HA	0	0	0	0	17	0	0
SA	0	0	0	1	0	9	0
SU	0	0	0	0	0	0	12

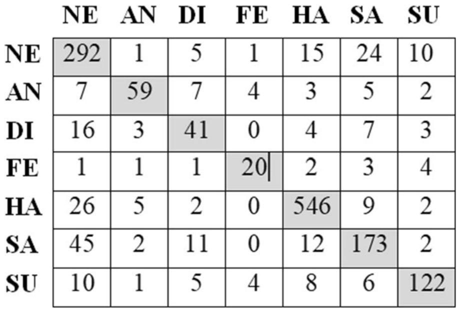
Fig. 7 Confusion matrix for CK+ dataset



	NE	AN	DI	FE	HA	SA	SU
NE	11	2	0	0	0	0	0
AN	0	11	0	0	0	0	0
DI	0	2	10	1	0	0	0
FE	0	2	0	3	0	0	0
HA	0	0	0	0	8	0	0
SA	0	2	1	0	1	10	0
SU	1	0	0	0	0	0	18

Fig. 8 Confusion matrix for KDEF dataset

highlight the training and validation accuracies, respectively, of each dataset. Finally, testing sets are fed into trained FER-net one after another in order to obtain confusion matrices.



	NE	AN	DI	FE	HA	SA	SU
NE	292	1	5	1	15	24	10
AN	7	59	7	4	3	5	2
DI	16	3	41	0	4	7	3
FE	1	1	1	20	2	3	4
HA	26	5	2	0	546	9	2
SA	45	2	11	0	12	173	2
SU	10	1	5	4	8	6	122

Fig. 9 Confusion matrix for RAF dataset

Table 4 Classification report for JAFFE dataset

Classes	A	P	R	F	Support
NE	0.93	1.0	0.93	0.97	15
AN	1.0	1.0	1.0	1.0	17
DI	1.0	0.93	1.0	0.97	14
FE	1.0	1.0	1.0	1.0	6
HA	1.0	1.0	1.0	1.0	19
SA	0.91	1.0	0.92	0.96	12
SU	0.92	0.92	1.0	0.96	13

Table 5 Classification report for CK+ dataset

Classes	A	P	R	F	Support
NE	1.0	1.0	1.0	1.0	14
AN	1.0	1.0	1.0	1.0	10
DI	0.95	1.0	0.96	0.97	20
FE	0.90	0.90	0.90	0.90	10
HA	1.0	0.94	1.0	0.97	17
SA	0.90	0.90	0.90	0.90	10
SU	1.0	1.0	1.0	1.0	12

Table 6 Classification report for KDEF dataset

Classes	A	P	R	F	Support
NE	0.84	0.92	0.85	0.88	13
AN	1.0	0.58	1.0	0.73	11
DI	0.76	0.91	0.77	0.83	13
FE	0.37	0.75	0.38	0.50	8
HA	1.0	0.89	1.0	0.94	8
SA	0.71	1.0	0.71	0.83	14
SU	0.94	0.86	0.85	0.90	19

Table 7 Classification report for RAF dataset

Classes	A	P	R	F	Support
NE	0.84	0.74	0.83	0.78	350
AN	0.68	0.80	0.68	0.73	87
DI	0.56	0.57	0.55	0.56	74
FE	0.62	0.69	0.62	0.66	32
HA	0.93	0.93	0.93	0.93	590
SA	0.71	0.76	0.71	0.73	245
SU	0.78	0.84	0.78	0.81	156

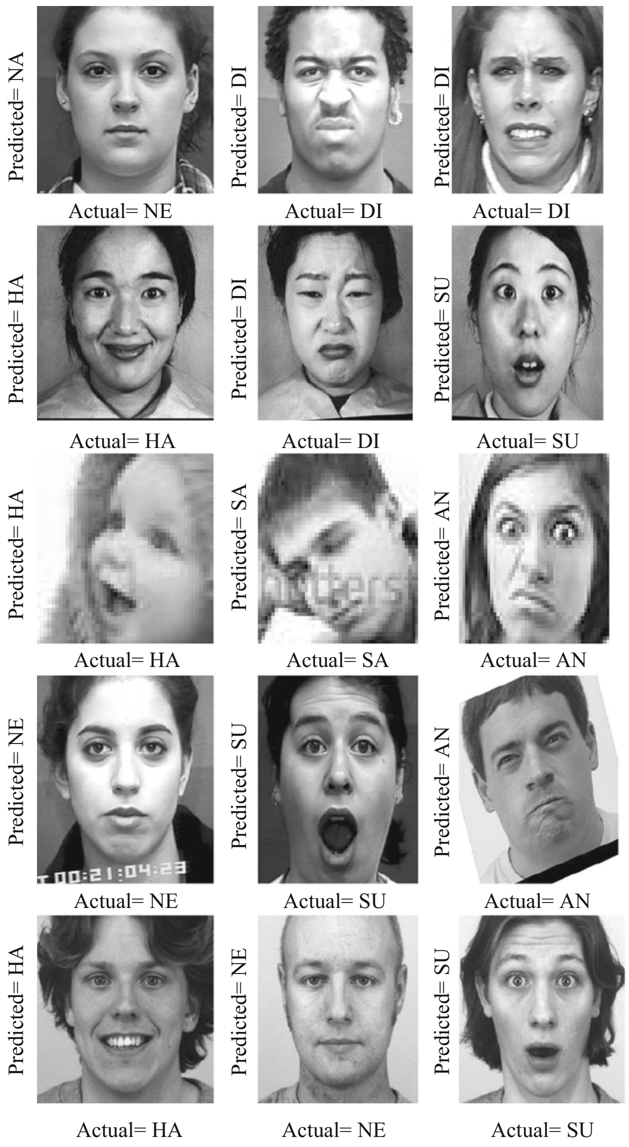


Fig. 10 Some of the successfully predicted FEs, images of each dataset [52–56] are shown row-wise

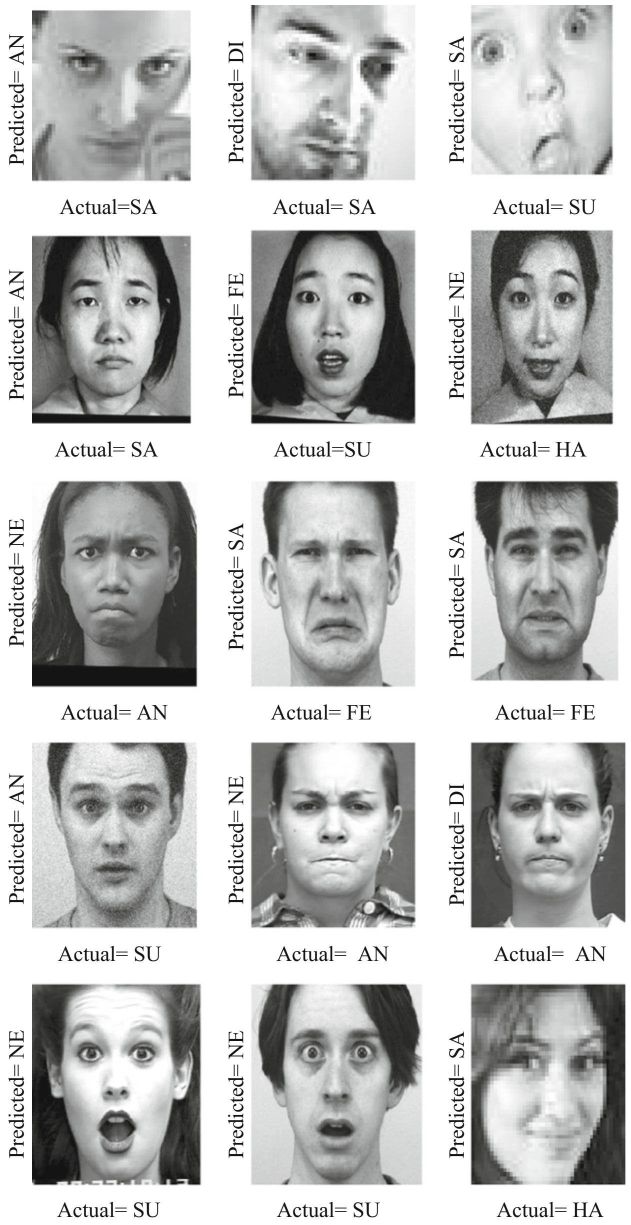


Fig. 11 Some of the wrongly predicted FEs, images of each dataset [52–56] are shown row-wise

Then confusion matrices help to calculate the values of metrics, which are used further to analyze the performance of FER-net. Figure 5 shows the confusion matrix of FER2013 dataset. The size of the confusion matrix is 7 × 7 as 7 facial expressions are considered for this study. The generated metrics from Fig. 5 are reported in Table 3. Similarly, the confusion matrices for JAFFE, CK+, KDEF, and RAF datasets are shown in Figs. 6, 7, 8 and 9, respectively, and their generated metrics are reported in Tables 4, 5, 6, and 7, respectively.

Table 8 Performance comparison of FER-net with classification accuracy (%) on five datasets viz. FER2013, JAFFE, CK+, KDEF, and RAF

S. no.	Method	FER2013	JAFFE	CK+	KDEF	RAF
Classification accuracy (%) on five datasets						
1.	AlexNet	77	95	97	76	78
2.	HOG-TOP [26]	52	60	65	55	53
3.	SCNN [32]	55	50	61	55	70
4.	MCNN [32]	64	68	85	67	78
5.	SCNN-LBP [51]	54	88	83	70	71
6.	SCNN-gray [51]	72	92	94	78	78
7.	P-VGG16 [51]	72	96	91	78	57
8.	WMDC [51]	75	92	97	81	75
9.	WFTS [46]	63	90	91	74	70
10.	ACNN-LBP [47]	50	90	95	66	77
11.	Fusion (ACNN-LBP + GF) [47]	53	90	94	69	78
12.	STF + LSTM [48]	71	90	82	81	73
13.	Ensemble DCNNs [41]	53	55	67	58	70
14.	DCNN-BC [42]	50	56	73	70	68
15.	IACNN [44]	68	75	95	67	74
16.	2B(N + M)Softmax [43]	67	78	87	81	69
17.	CF + GT [49]	66	77	86	80	70
18.	Broad learning [50]	44	93	81	**89**	64
19.	Deep-emotion [45]	70	93	94	81	72
20.	VGG19 [31]	74	95	96	81	60
21.	ResNet150 [31]	75	91	89	72	70
22.	**Proposed method**	**79**	**97**	**98**	83	**82**

Bold values indicate that the best results

It is clear from Tables 3, 4, 5, 6, and 7 that the proposed CNN model, i.e., FER-net, yields good classification accuracies along with other metrics for JAFFE, CK+, and KDEF datasets in all most all the cases. However, the performance is satisfactory for the other two datasets, namely FER2013 and RAF dataset. Some of the successfully and wrongly predicted FEs by the FER-net are shown in Figs. 10 and 11, respectively. In this study, the proposed FER-net is compared with twenty-one state-of-the-art FER methods. A detailed review of these models is beyond the scope of this work, which can be referred to [26, 31, 32, 41–51]. It is clear from Table 8 that the proposed model outperforms legacy works in almost all the cases except broad learning. Broad learning gives better results for the KDEF dataset only. We can conclude that the proposed model is simple, but still it gives good prediction accuracy in almost all the cases compared to complex state-of-the-art models. Moreover, we investigate the time taken for the training and testing of our model. Table 9 shows the execution time for training and testing the proposed model along with various comparative methods. However, our proposed method obtains satisfactory performance compared to other methods in terms of execution time. The testing time per image (TTPI) is also shown in Table 9. However, TTPI is the same for all the datasets since the size of the image is considered the same for all the datasets. The testing time for the test set varies with different datasets due to the size of the testing set.

4 Conclusion

In this study, we propose a simple CNN reckoned as FER-net for FER. Five publicly available benchmarking datasets, namely FER2013, JAFFE, CK+, KDEF, and RAF datasets, are considered here. These datasets consist of seven basic FEs, namely NE, AN, DI, FE, HA, SA, and SU, which are classified by the FER-net along with twenty-one state-of-the-art models. The FER-net extracts feature from face regions automatically. Then these features are fed to a softmax classifier for identifying FEs. It is clear from the obtained results that the proposed model is superior to the state of the art in almost all cases except broad learning. Broad learning yields good results for the KDEF dataset only. Moreover, the proposed model is simple as compared

Table 9 Performance comparison of FER-net in terms of execution time on five datasets viz. FER2013, JAFFE, CK+, KDEF, and RAF

S. no.	Method	TTPI	Training time					Testing time for all the images				
			FER2013	JAFFE	CK+	KDEF	RAF	FER2013	JAFFE	CK+	KDEF	RAF
Execution time in minutes												
1.	AlexNet	0.2412	120.0	43.0	47.0	51.0	81.0	1.9834	0.8028	0.8236	0.8429	0.9906
2.	HOG-TOP [26]	0.4500	220.0	86.0	92.33	115.0	165.0	2.1623	1.2124	1.3264	1.1576	2.1529
3.	SCNN [32]	0.1812	5.5	1.0	1.5	2.5	4.0	0.9992	0.8845	0.8985	0.9001	0.9845
4.	MCNN [32]	0.1872	9.0	1.5	2.0	2.5	6.0	1.1060	0.8883	0.9127	0.9168	1.0921
5.	SCNN-LBP [51]	0.1801	75.0	8.33	16.33	25.0	41.66	1.0721	0.8821	0.8991	0.9008	0.9698
6.	SCNN-gray [51]	0.1801	75.0	8.33	16.33	25.0	41.66	1.0721	0.8821	0.8991	0.9008	0.9698
7.	P-VGG16 [51]	0.4309	441.66	150.0	158.33	175.0	316.66	1.9348	1.4508	1.5238	1.5523	1.8900
8.	WMDC [51]	0.4699	516.66	158.33	174.99	200.0	358.32	2.2602	1.9665	1.9789	1.9965	2.1056
9.	WFTS [46]	0.8956	150.0	9.63	10.0	16.6	70.0	2.2012	1.9804	1.9912	1.9989	2.1020
10.	ACNN-LBP [47]	0.3945	105.0	10.0	20.0	30.0	70.0	1.3563	1.1230	1.1321	1.1394	1.2953
11.	Fusion(ACNN-LBP + GF) [47]	0.4612	280.0	70.0	110.0	135.0	210.0	2.3906	1.9023	1.9984	2.0102	2.2134
12.	STF + LSTM. [48]	0.4329	460.0	205.0	215.33	240.0	340.0	3.1652	2.6743	2.8628	2.9845	3.1107
13.	Ensemble DCNNs [41]	0.4917	420.0	200.0	220.0	300.0	360.0	3.8138	2.5827	2.6296	2.7013	3.4238
14.	DCNN-BC [42]	0.1725	103.33	24.66	28.0	37.33	75.0	1.6239	1.3469	1.4430	1.4523	1.5426
15.	IACNN [44]	0.3946	1033.0	600.0	633.0	700.0	866.66	2.1623	1.8920	1.9165	1.9730	2.0935
16.	2B(N+M)Softmax [43]	0.2814	265.0	80.0	85.33	93.33	165.0	1.1721	0.9643	0.9892	0.9946	1.0021
17.	CF + GT [49]	0.3218	240.0	98.66	100.0	115.33	180.66	2.8794	2.2167	2.4510	2.5503	2.7165
18.	Broad learning [50]	0.1023	7.12	1.0	1.5	2.0	4.5	0.4812	0.3101	0.3323	0.3812	0.4102
19.	Deep-emotion [45]	0.2908	316.0	41.6	50.0	66.66	241.66	1.0982	0.8943	0.9165	0.9346	0.9978
20.	VGG19 [31]-1	0.6128	45.5	22.0	23.0	26.5	34.5	2.2123	1.9821	1.9901	2.0981	2.1823
21.	ResNet150 [31]-2	0.7123	130.0	54.0	67.0	76.5	105.5	3.1190	2.5981	2.6180	2.7833	2.9009
22.	Proposed method	0.2381	81.0	2.0	3.160	3.5	12.0	1.1612	0.4026	0.5697	0.6101	0.6807

to state-of-the-art models and is preeminent in terms of accuracy, as well as execution time. In the future, more number of datasets would be considered to further validate the proposed model. The proposed model provides satisfactory result for KDEF dataset; however, there is provision to increase the performance, which deserves further study.

Acknowledgements This work is partially supported by the project "Prediction of diseases through computer-assisted diagnosis system using images captured by minimally invasive and non-invasive modalities", Computer Science and Engineering, PDPM Indian Institute of Information Technology, Design and Manufacturing, Jabalpur, India (under ID: SPARC-MHRD-231). This work is also partially supported by the project at Universiti Teknologi Malaysia (UTM) under Research University Grant Vot-20H04, Malaysia Research University Network (MRUN) Vot 4L876 and the Fundamental Research Grant Scheme (FRGS) Vot5F073 supported under Ministry of Education Malaysia for the completion of the research.

References

1. Yeasin M, Bullot B, Sharma R (2006) Recognition of facial expressions and measurement of levels of interest from video. IEEE Trans Multimed 8(3):500–508
2. Lola C, Philippe G (2005) Emotion understanding: robots as tools and models. University Press England, Oxford
3. Dornaika F, Raducanu B (2009) Facial expression recognition for hci applications. In: Encyclopedia of artificial intelligence. IGI Global, pp 625–631
4. Shakya S, Sharma S, Basnet A (2016) Human behavior prediction using facial expression analysis. In: 2016 international conference on computing, communication and automation (ICCCA). IEEE, pp 399–404
5. Dickey CC, Panych LP, Voglmaier MM, Niznikiewicz MA, Terry DP, Murphy C, Zacks R, Shenton ME, McCarley RW (2011) Facial emotion recognition and facial affect display in schizotypal personality disorder. Schizophr Res 131(1–3):242–249
6. Jeong M, Ko BC (2018) Driver's facial expression recognition in real-time for safe driving. Sensors 18(12):4270
7. Zhou Y, Shi BE (2017) Photorealistic facial expression synthesis by the conditional difference adversarial autoencoder. In: 2017 seventh international conference on affective computing and intelligent interaction (ACII). IEEE, pp 370–376

8. Viola P, Jones P et al (2001) Rapid object detection using a boosted cascade of simple features. In: CVPR 2001, vol 1, no 511–518, p 3
9. Siswanto ARS, Nugroho AS, Galinium M (2014) Implementation of face recognition algorithm for biometrics based time attendance system. In: 2014 international conference on ICT For Smart Society (ICISS). IEEE, pp 149–154
10. Turk MA, Pentland AP (1991) Face recognition using eigenfaces. In: Proceedings. 1991 IEEE computer society conference on computer vision and pattern recognition. IEEE, pp 586–591
11. Lundqvist D, Flykt A, Öhman A (1998) The karolinska directed emotional faces (kdef), vol 91. CD ROM from Department of Clinical Neuroscience, Psychology section, Karolinska Institutet, Solna, p 30
12. Mohammadi MR, Fatemizadeh E, Mahoor MH (2014) Pca-based dictionary building for accurate facial expression recognition via sparse representation. J Vis Commun Image Represent 25(5):1082–1092
13. Liu C, Wechsler H (2002) Gabor feature based classification using the enhanced fisher linear discriminant model for face recognition. IEEE Trans Image Process 11(4):467–476
14. Lucey P, Cohn JF, Kanade T, Saragih J, Ambadar Z, Matthews I (2010) The extended cohn-kanade dataset (ck+): a complete dataset for action unit and emotion-specified expression. In: 2010 IEEE computer society conference on computer vision and pattern recognition-workshops. IEEE, pp 94–101
15. 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
16. Kobayashi H, Hara F (1997) Facial interaction between animated 3d face robot and human beings. In: 1997 IEEE international conference on systems, man, and cybernetics. Computational cybernetics and simulation, vol 4. IEEE, pp 3732–3737
17. Zhong L, Liu Q, Yang P, Huang J, Metaxas DN (2014) Learning multiscale active facial patches for expression analysis. IEEE Trans Cybern 45(8):1499–1510
18. Mase K (1991) Recognition of facial expression from optical flow. IEICE Trans Inf Syst 74(10):3474–3483
19. Chen C-R, Wong W-S, Chiu C-T (2010) A 0.64 mm ^{2} real-time cascade face detection design based on reduced two-field extraction. IEEE Trans Very Large Scale Integr (VLSI) Syst 19(11):1937–1948
20. Kotsia I, Pitas I (2006) Facial expression recognition in image sequences using geometric deformation features and support vector machines. IEEE Trans Image Process 16(1):172–187
21. Sohail ASM, Bhattacharya P (2007) Classification of facial expressions using k-nearest neighbor classifier. In: International conference on computer vision/computer graphics collaboration techniques and applications. Springer, pp 555–566
22. Li X, Ji Q (2004) Active affective state detection and user assistance with dynamic Bayesian networks. IEEE Trans Syst Man Cybern Part A Syst Huma 35(1):93–105
23. Fanelli G, Gall J, Van Gool L (2011) Real time head pose estimation with random regression forests. In: CVPR 2011. IEEE, pp 617–624
24. 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
25. Sebe N, Lew MS, Cohen I, Garg A, Huang TS (2002) Emotion recognition using a Cauchy naive Bayes classifier. In: Makihara Y, Takizawa M, Shirai Y, Miura J, Shimada N (eds) Object recognition supported by user interaction for service robots, vol 1. IEEE, New York, pp 17–20
26. Chen J, Chen Z, Chi Z, Hong F (2016) Facial expression recognition in video with multiple feature fusion. IEEE Trans Affect Comput 9(1):38–50
27. Karlekar A, Seal A (2020) Soynet: Soybean leaf diseases classification. Comput Electron Agric 172:105342
28. Mohan K, Seal A, Krejcar O, Yazidi A (2020) Facial expression recognition using local gravitational force descriptor based deep convolution neural networks. IEEE Trans Instrum Meas 70:1–12
29. Zhao K, Chu W-S, Zhang H (2016) Deep region and multi-label learning for facial action unit detection. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp 3391–3399
30. Sun N, Li Q, Huan R, Liu J, Han G (2017) Deep spatial-temporal feature fusion for facial expression recognition in static images. Pattern Recogn Lett 119:49–61
31. Orozco D, Lee C, Arabadzhi Y, Gupta D. Transfer learning for facial expression recognition
32. Alizadeh S, Fazel A (2017) Convolutional neural networks for facial expression recognition arXiv:1704:06756
33. Zhao G, Pietikainen M (2007) Dynamic texture recognition using local binary patterns with an application to facial expressions. IEEE Trans Pattern Anal Mach Intell 6:915–928
34. Dhall A, Goecke R, Joshi J, Wagner M, Gedeon T (2013) Emotion recognition in the wild challenge 2013. In: Proceedings of the 15th ACM on international conference on multimodal interaction. ACM, pp 509–516
35. Yu Z, Zhang C (2015) Image based static facial expression recognition with multiple deep network learning. In Proceedings of the 2015 ACM on international conference on multimodal interaction. ACM, pp 435–442
36. Jain DK, Shamsolmoali P, Sehdev P (2019) Extended deep neural network for facial emotion recognition. Pattern Recogn Lett 120:69–74
37. Salmam FZ, Madani A, Kissi M (2019) Fusing multi-stream deep neural networks for facial expression recognition. SIViP 13(3):609–616
38. Li K, Jin Y, Akram MW, Han R, Chen J (2019) Facial expression recognition with convolutional neural networks via a new face cropping and rotation strategy. Vis Comput 36:391–404
39. Shao J, Qian Y (2019) Three convolutional neural network models for facial expression recognition in the wild. Neurocomputing 355:82–92
40. Uddin MZ, Hassan MM, Almogren A, Alamri A, Alrubaian M, Fortino G (2017) Facial expression recognition utilizing local direction-based robust features and deep belief network. IEEE Access 5:4525–4536
41. Pons G, Masip D (2017) Supervised committee of convolutional neural networks in automated facial expression analysis. IEEE Trans Affect Comput 9(3):343–350
42. Villanueva MG, Zavala SR, Zavala (2020) Deep neural network architecture: application for facial expression recognition. IEEE Lat Am Trans 18(07):1311–1319
43. Liu X, Vijaya Kumar BV, You J, Jia P (2017) Adaptive deep metric learning for identity-aware facial expression recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops, pp 20–29
44. Meng Z, Liu P, Cai J, Han S, Tong Y (2017) Identity-aware convolutional neural network for facial expression recognition. In: 2017 12th IEEE international conference on automatic face & gesture recognition (FG 2017). IEEE, pp 558–565
45. Minaee S, Abdolrashidi A (2019) Deep-emotion: Facial expression recognition using attentional convolutional network. arXiv:1902.01019
46. Hua W, Dai F, Huang L, Xiong J, Gui G (2019) Hero: human emotions recognition for realizing intelligent internet of things. IEEE Access 7:24321–24332

47. Kim J-H, Kim B-G, Roy PP, Jeong D-M (2019) Efficient facial expression recognition algorithm based on hierarchical deep neural network structure. IEEE Access 7:41273–41285
48. Kim DH, Baddar WJ, Jang J, Ro YM (2017) Multi-objective based spatio-temporal feature representation learning robust to expression intensity variations for facial expression recognition. IEEE Trans Affect Comput 10(2):223–236
49. Benitez-Quiroz CF, Srinivasan R, Martinez AM (2018) Discriminant functional learning of color features for the recognition of facial action units and their intensities. IEEE Trans Pattern Anal Mach Intell 41(12):2835–2845
50. Zhang T, Liu Z, Wang X-H, Xing X-F, Chen CP, Chen E (2018) Facial expression recognition via broad learning system. In: 2018 IEEE international conference on systems, man, and cybernetics (SMC). IEEE, pp 1898–1902
51. Zhao X, Shi X, Zhang S (2015) Facial expression recognition via deep learning. IETE Tech Rev 32(5):347–355
52. Chellappa R, Wilson CL, Sirohey S et al (1995) Human and machine recognition of faces: a survey. Proc IEEE 83(5):705–740
53. Samal A, Iyengar PA (1992) Automatic recognition and analysis of human faces and facial expressions: a survey. Pattern Recogn 25(1):65–77
54. Li H, Lin Z, Shen X, Brandt J, Hua G (2015) A convolutional neural network cascade for face detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 5325–5334
55. Lyons M, Akamatsu S, Kamachi M, Gyoba J (1998) Coding facial expressions with gabor wavelets. In: Proceedings third IEEE international conference on automatic face and gesture recognition. IEEE, pp 200–205
56. Li S, Deng W (2019) Reliable crowdsourcing and deep locality-preserving learning for unconstrained facial expression recognition. IEEE Trans Image Process 28(1):356–370
57. Li S, Deng W, Du JP (2017) Reliable crowdsourcing and deep locality-preserving learning for expression recognition in the wild. In: 2017 IEEE conference on computer vision and pattern recognition (CVPR). IEEE, pp 2584–2593
58. LeCun Y, Haffner P, Bottou L, Bengio Y (1999) Object recognition with gradient-based learning. In: Forsyth DA, Mundy JL, di Gesu V, Cipolla R (eds) Shape, contour and grouping in computer vision. Springer, Berlin, pp 319–345
59. Ding H, Zhou SK, Chellappa R (2017) Facenet2expnet: regularizing a deep face recognition net for expression recognition. In: 2017 12th IEEE international conference on automatic face and gesture recognition (FG 2017). IEEE, pp 118–126
60. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. arXiv:1409.1556
61. 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
62. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A (2015) Going deeper with convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1–9
63. Huang G, Liu Z, Van Der Maaten L, Weinberger KQ (2017) Densely connected convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4700–4708
64. Chollet F (2017) Xception: deep learning with depthwise separable convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1251–1258
65. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems, pp 1097–1105
66. Karlekar A, Seal A, Krejcar O, Gonzalo-Martin C (2019) Fuzzy k-means using non-linear s-distance. IEEE Access 7:55121–55131
67. Sharma KK, Seal A (2019) Modeling uncertain data using Monte Carlo integration method for clustering. Expert Syst Appl 137:100–116
68. Sharma KK, Seal A (2020) Spectral embedded generalized mean based k-nearest neighbors clustering with s-distance. Expert Syst Appl. https://doi.org/10.1016/j.eswa.2020.114326
69. Sharma KK, Seal A (2020) Outlier-robust multi-view clustering for uncertain data. Knowl Based Syst 211:106567
70. Jain S, Seal A, Ojha A, Krejcar O, Bureš J, Tachecí I, Yazidi A (2020) Detection of abnormality in wireless capsule endoscopy images using fractal features. Comput Biol Med 127:104094
71. Sharma KK, Seal A (2020) Clustering analysis using an adaptive fused distance. Eng Appl Artif Intell 96:103928
72. Sharma KK, Seal A (2020) Multi-view spectral clustering for uncertain objects. Inf Sci 547:723–745

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.