Facial Expression Recognition Using Local Gravitational Force Descriptor-Based Deep Convolution Neural Networks

Karnati Mohan, Ayan Seal, *Senior Member IEEE*, Ondrej Krejcar, and Anis Yazidi, *Senior Member IEEE*

***Abstract*— An image is worth a thousand words; hence, a face image illustrates extensive details about the specification, gender, age, and emotional states of mind. Facial expressions play an important role in community-based interactions and are often used in the behavioral analysis of emotions. Recognition of automatic facial expressions from a facial image is a challenging task in the computer vision community and admits a large set of applications, such as driver safety, human–computer interactions, health care, behavioral science, video conferencing, cognitive science, and others. In this work, a deep-learning-based scheme is proposed for identifying the facial expression of a person. The proposed method consists of two parts. The former one finds out local features from face images using a local gravitational force descriptor, while, in the latter part, the descriptor is fed into a novel deep convolution neural network (DCNN) model. The proposed DCNN has two branches. The first branch explores geometric features, such as edges, curves, and lines, whereas holistic features are extracted by the second branch. Finally, the score-level fusion technique is adopted to compute the final classification score. The proposed method along with 25 state-of-the-art methods is implemented on five benchmark available databases, namely, Facial Expression Recognition 2013, Japanese Female Facial Expressions, Extended CohnKanade, Karolinska Directed Emotional Faces, and Real-world Affective Faces. The databases consist of seven basic emotions: neutral, happiness, anger, sadness, fear, disgust, and surprise. The proposed method is compared with existing approaches using four evaluation metrics, namely, accuracy, precision, recall, and f1-score. The obtained results demonstrate that the proposed method outperforms all state-of-the-art methods on all the databases.**

***Index Terms*— Deep convolution neural networks (DCNNs), facial expression recognition (FER), local gravitational force (GF) descriptor, score-level fusion, softmax classification.**

I. Introduction

AFFECTIVE computing is a field of study that attempts to develop instruments/devices and systems that can identify, interpret, process, and simulate human effects. Nowadays, it has got a considerable amount of attention toward the research communities in the fields of artificial intelligence and computer vision due to its noticeable academic and commercial applications, such as human–computer interaction (HCI), virtual reality, health care, deceptive detection, multimedia, augmented reality, driver safety, and surveillance. Generally, computational models of human expression process affective state, and they are of two types: decision-making models and predictive models. The former one accounts for the effect of expression, whereas the latter one can identify the state of the emotion. Models of nonverbal expression of different forms of facial expressions deduced from speech, body gesture, and physiological signals provide valuable sources for affective computing. Interested readers are referred to [1] to know more about various methods, models, and applications of affective computing. In this study, computer-based facial expression recognition (FER) is considered due to its ability to mimic human coding skills. FER is indispensable in affective computing. Facial expression is an essence of nonverbal communication to express the internal behaviors in interpersonal relations. Moreover, it is a sentiment analysis technology that uses biometric to automatically recognize seven basic emotions: neutral (NE), anger (AN), disgust (DI), fear (FE), happiness (HA), sadness (SA), and surprise (SU) from still images or videos. Although a considerable amount of works was conducted for developing instruments to access emotions, recognizing human expressions is still a challenging task that is affected by definite circumstances especially when performed

Manuscript received August 3, 2020; revised September 24, 2020; accepted October 8, 2020. Date of publication October 16, 2020; date of current version December 30, 2020. This work was supported in part by the project "Prediction of Diseases Through Computer Assisted Diagnosis System Using Images Captured by Minimally Invasive and Noninvasive Modalities," Computer Science and Engineering, PDPM Indian Institute of Information Technology, Design and Manufacturing Jabalpur, Jabalpur, India, under Grant ID: SPARC-MHRD-231, in part by the project IT4Neuro(degeneration) under Grant CZ.02.1.01/0.0/0.0/18 069/0010054, in part by the project "Smart Solutions in Ubiquitous Computing Environments," Grant Agency of Excellence, University of Hradec Kralove, Faculty of Informatics and Management, Czech Republic under Grant ID: UHK-FIM-GE-2020, in part by the project at Universiti Teknologi Malaysia (UTM) under Research University Grant Vot-20H04, in part by the Malaysia Research University Network (MRUN) under Grant Vot 4L876, and in part by the Fundamental Research Grant Scheme (FRGS) under the Ministry of Education Malaysia under Grant Vot5F073. The Associate Editor coordinating the review process was Dr. Hongrui Wang. *(Corresponding author: Ayan Seal.)*

Karnati Mohan is with the PDPM Indian Institute of Information Technology, Design and Manufacturing Jabalpur, Jabalpur 482005, India (e-mail: 1811011@iiitdmj.ac.in).

Ayan Seal is with the PDPM Indian Institute of Information Technology, Design and Manufacturing Jabalpur, Jabalpur 482005, India, and also with the Center for Basic and Applied Science, Faculty of Informatics and Management, University of Hradec Kralove, 500 03 Hradec Kralove, Czech Republic (e-mail: ayanseal30@ieee.org).

Ondrej Krejcar is with the Center for Basic and Applied Science, Faculty of Informatics and Management, University of Hradec Kralove, 500 03 Hradec Kralove, Czech Republic, and also with the Malaysia–Japan International Institute of Technology (MJIIT), Universiti Teknologi Malaysia, Kuala Lumpur 54100, Malaysia (e-mail: ondrej.krejcar@uhk.cz).

Anis Yazidi is with the Research Group in Applied Artificial Intelligence, Oslo Metropolitan University, 460167 Oslo, Norway (e-mail: anisy@oslomet.no).

Digital Object Identifier 10.1109/TIM.2020.3031835

in the wild. Some of the notable difficulties associated with FER are as follows: 1) when the difference between two facial expressions is small, it is difficult to distinguish them with high precision [2] and 2) generally, the expression of a particular facial emotion by different people is not the same due to the interperson variability and their face biometric shapes [3]. All the recent studies focus on FER methods that can be categorized into two groups: handcrafted feature-based methods and deep-learning feature-based methods. The former one is also divided into appearance-based features and geometric features. Appearance-based methods rely on various statistics of the pixels' values within the face image. Examples include Gobar wavelets [4], Haar wavelet [5], local binary pattern (LBP) [6], [7], histogram of oriented gradients (HOG) [7], [8], histogram of bunched intensity values (HBIV) [9], dynamic Bayesian network (DBN) [10], and so on. On the other hand, geometric features are obtained by transforming the image into geometric primitives, such as corner or minutiae points [11], edges, and curves [12]. This is accomplished, for example, by locating unique features, such as eyes, mouth, nose, and chin, and measuring their relative position [13], [14], width, and, perhaps, other parameters. However, extracting distinctive features based on traditional methods is limited to the human experience, so it is difficult to acquire and arduous to achieve better performance on large data. Traditional approaches are not up to the real FER application requirements, and they also require high computational cost and space [15].

Over the past few years, feature extraction from image data using deep convolution neural networks (DCNNs) has gained popularity in various computer vision tasks. By virtue of using DCNN, many breakthroughs were achieved for image classification problems, especially face related recognition tasks [16]–[18]. It is observed that DCNN has outperformed the traditional methods with handcrafted features in recent years [19]–[21]. DCNN is able to extract hypothetical features from a low level to a high level of facial images with the help of several nonlinear connections [16]. Furthermore, DCNN can extract useful unique features by solving several issues caused by traditional methods. In [22], a DCNN for FER was designed to provide better discrimination ability by combining the central loss function and the verification recognition model. In another work [23], a conditional generative adversarial network was presented to increase the size of the data and DCNN used for the facial expression classification [24]. Fathallah *et al.* [25] discussed a recognition algorithm based on the geometry group model.

A. Motivation and Contribution

It is clear from the literature that most of the existing works perform reasonably well on databases having images that were captured in controlled lab environments. However, these works do not yield satisfactory results on more challenging and real-time databases consisting of images with greater variations. Thus, there is a need to improve the performance of an FER system. The performance of an FER system relies on feature engineering. Engineering new features from existing ones can improve the performance of a system. This motivates us to work further in this direction. It is also clear from state-of-the-art methods that most of the works related to FER tasks are based on edge information [12], [26]–[30] because it varies in individual expression. Important features, such as corners, lines, and curves, can be extracted from the edges of an image. Edges are significant local changes of intensity in an image. In the recent past, various DCNN models were exploited to extract hypothetical deep features for developing FER systems. However, the number of features is quite large. Sometimes, deep features may lead to overfitting. Moreover, the extraction of deep features is time-consuming, and it requires powerful resources. Furthermore, only a small fraction of these overwhelming numbers of features are used. On the other hand, edge detection using gradient captures the small change in the x- and y-directions, which are known as gradients. The gradient is a vector that has a certain magnitude (M) and direction (D). M would be higher when there is a sharp change in intensity, such as around the edges. M provides information about edge strength. On the other hand, D is always perpendicular to D of the edge. D represents the geometric structure of the image. Thus, in the first step of the proposed method, edge descriptor based on gravitational force (GF) [31] is adopted because it uses surrounding pixel information instead of considering the adjacent pixels difference in the x- and y-directions while computing M and D images. However, the proposed system does not depend only on local edge information, but it also depends on holistic features. Thus, in the second step, M and D images are fed into a novel DCNN to extract useful information. The proposed DCNN consists of two branches: the first one consists of shallow DCNN and extracts the local features, whereas the second one fetches the holistic features from M and D images as it consists of major DCNN. Finally, a score-level fusion technique is adopted on classification results obtained from M and D images to get final results. The overview of the proposed method is shown in Fig. 1. The performance of the proposed method is compared with 25 state-of-the-art methods. All the methods are implemented on five benchmark databases, namely, FER 2013 (FER2013) [32], Japanese Female Facial Expressions (JAFFE) [33], Extended CohnKanade (CK+) [34], Karolinska Directed Emotional Faces (KDEFs) [35], and Real-world Affective Faces (RAFs) [36], [37]. To measure the efficiencies of all the methods, including the proposed one, four classification metrics, namely, accuracy, precision, recall, and f1-score, are considered for the quantitative evaluation. Empirical outcomes illustrate that the proposed method defeats all the 25 state-of-the-art methods.

The rest of the work is organized as follows. In Section II, a review of earlier works related to FER is conducted. The proposed method is described in Section III. Experimental results and discussion are presented in Section IV. Finally, Section V concludes the work.

II. Related Work

All the methods in the FER task can be categorized into two groups based on feature extraction techniques, namely, handcrafted features and deep-learning features. This section

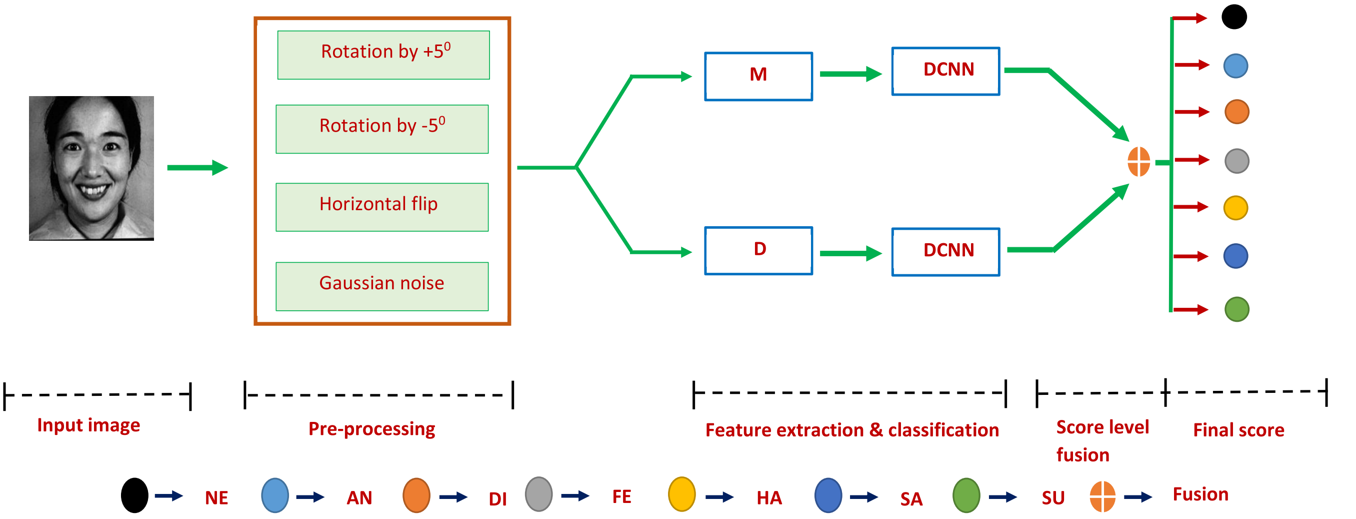


Fig. 1. Overview of the proposed FER scheme.

presents them briefly. Mainly two steps, namely, feature extraction and classification, are associated with the FER task. Conventional features, such as Gobar wavelets [4], curves [12], scale-invariant feature transform [21], HOG [8], LBP [6], minutiae points [11], Haar wavelet [5], HBIV [9], DBN [10], and edges [38], were exploited with advanced domain comprehension in the first step. In the second step, support vector machine (SVM) [39], feedforward neural network [40], and extreme learning machine [41] were adopted for classification. Chen *et al.* [42] offered a feature descriptor called HOG from three orthogonal planes (HOG-TOPs) to extract dynamic textures from video sequences to characterize facial appearance changes. However, handcrafted features-based methods have limited performance in real-life applications, especially for FER tasks. In recent years, it is observed from the literature that deep-learning-based methods are superior to handcrafted features-based methods for FER tasks [17], [18], [43], [44]. Shallow and Deep CNN considered for extraction on gray-scale images and classified using softmax classifiers on FER2013 [45]. In [46], the DCNN framework and Softmax were considered for feature extraction and classification respectively on the FER2013 database [32]. Orozco *et al.* [47] presented Alexnet-, VGG19-, and ResNet-based transfer learning methods for the FER task. Sun *et al.* [16] presented a DCNN model and DeepID features for face recognition. In another work, Sun *et al.* [48] considered the Siamese network to increase the efficiency of the FER task. Barsoum *et al.* [49] discussed the VGG13 network for the FER task on the FER+ database. A weighted mixture deep neural network is considered for the FER task, and it consists of two channels: one of them was used to extract the facial expression features on gray-scale images with partial VGG16 framework. On the other hand, features are extracted on LBP images with shallow DCNN on JAFFE [33] and CK+ [34] databases further, and softmax classifiers were used to classify the extracted features and then combined obtained outputs from both the channels using weighted fusion in [50]. In [51], features were extracted from pretrained VGG19 architecture on the ImageNet database for the FER task and SVM used for expression classification on JAFFE and CK+ databases. In [52], three DCNN subnetworks were considered and trained independently on FER2013 and AffectNet database [53], further ensembled three networks using weighted fusion. Furthermore, larger weights were assigned to the network, which obtained higher recognition accuracy. Moreover, appearance-based features were extracted using DCNN, and obtained features were fused with geometric feature-based DCNN in the hierarchical constitution according to [54]. However, appearance features were extracted on LBP images; likewise, gray-scale images were considered for geometric feature-based networks on JAFFE and CK+ databases. In [55], ensembled ResNet50 and VGG16 frameworks were utilized to extract facial features and classify individual expressions on the KDEF database [35]. Hasani and Mahoor [56] presented a DCNN framework that consists of 3-D Inception-ResNet layers followed by a long short-term memory (LSTM) unit that together extract the spatial and temporal relations from facial images (3-D Inception-ResNet + landmarks). Geometric and regional LBP features were merged by autoencoders followed by Kohonen self-organizing map (SOM)-based classifier (Autoencoders + SOM) to recognize facial expressions [57]. Kim *et al.* [58] considered a spatiotemporal feature representation learning for solving the FER problem by encoding the characteristics of facial expressions using DCNN and LSTM (spatiotemporal feature + LSTM). Pons and Masip [59] considered ensembles of DCNNs for solving the FER problem. Villanueva and Zavala [60] presented a DCNN for classifying two facial expressions: happy and sad only. Meng *et al.* [61] and Liu *et al.* [62] worked on identity-aware FER models. Meng *et al.* [61] used two identical DCNN streams to jointly estimate various expressions and identity features (IACNN) to find relief intersubject variations initiated by personal attributes for the FER task. On the other hand, Liu *et al.* [62] employed deep metric learning (2B (N + M)Softmax) to jointly optimize a deep metric and softmax loss. Alam *et al.* [63] resorted to a sparse-deep simultaneous recurrent network (S-DSRN) for the FER problem and incorporated a dropout rate to the model. Benitez-Quiroz *et al.* [64] presented an FER system based on discriminant color features and a Gabor transform-based algorithm (color features + Gabor transform) to gain invariance to the timing of facial action unit (AU) changes. In [65], a model called deep comprehensive multipatches aggregation convolutional neural networks (DCMA-CNNs) was presented. It had two branches. One branch extracted holistic features, whereas the other branch obtained local features from segmented expressional image patches. Then, both feature vectors were combined to classify expressions using DCNN with ETI-pooling. Zhang *et al.* [66] developed a broad learning system for FER. A multilevel DCNN was developed to extract midlevel and high-level features within facial images to solve the FER problem (ensemble of MLCNNs) [67]. In [68], an attentional DCNN named a deep emotion to tackle the FER problem was devised. In [69], a deep AU graph network was presented based on a psychological mechanism. In the first step, the face image is divided into small key areas using segmentation techniques. Furthermore, these key areas are then converted into corresponding AU-related facial expression regions. Second, from these regions, local appearance features were extracted for further AUs analysis. Then, considering AU-related regions as vertices and distance between every two landmarks as edges, AUs facial graph is constructed to represent expressions. Finally, learning hybrid features for FER adjacency matrices of the facial graph is put into a graph-based convolutional neural network to combine the local-appearance and global-geometry information. Kopaczka *et al.* [7] presented a high-resolution thermal facial image database for the FER task. Besides, they extend existing approaches for infrared landmark detection with a head pose estimation for improved robustness and analyze the performance of a deep-learning method on this task.

III. Proposed Method

This section presents a brief overview of an edge descriptor of an image using GF followed by a detailed description of our proposed DCNN for the FER task.

A. Edge Descriptor

Roy and Bhattacharjee [70] stated that each pixel value of an image is parallel to a universal body, and therefore, it is considered as a mass of the body. The GF of an image is employed by the central pixel on its adjacent pixels. The Law of Universal Gravitation states that everybody mass (m_1) attracts every other body mass (m_2) in the universe by a force pointing in a straight line (d) between the centers of mass of both bodies, and this force, GF, is proportional to the masses of the bodies and inversely proportional to the square of their separation. Mathematically, GF is computed using the following equation:

$$\mathrm{GF} = G\frac{(m_1 \times m_2)}{d^2} \tag{1}$$

where G is gravitational constant, and its value is 6.67259×10^{-11}. Let A be a gray-scale image. Let us consider a center pixel A_c of a local 3 mask. It means that A_c is surrounded by eight neighboring pixels A_i. It is clear from the law of universal gravitation that all the eight neighboring pixels A_i exert forces on A_c. Thus, the force exerted on A_c by the ith neighboring pixel can be represented by GF_{ic}. Then, the x and y components of GF_{ic} are $\mathrm{GF}_{ic_x} = \mathrm{GF}_{ic} \times \sin\phi$ and $\mathrm{GF}_{ic_y} = \mathrm{GF}_{ic} \times \cos\phi$, respectively, when GF_{ic} is at an angle of ϕ with respect to the x-axis. GF_{ic_x} and GF_{ic_y} can be computed using (2) and (3), respectively. Fig. 2 shows an input image and edge strengths, i.e., Ms in the x- and y-directions

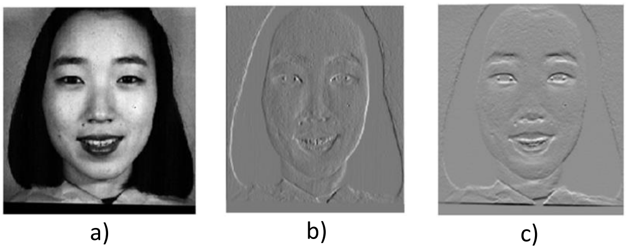


Fig. 2. (a) Sample gray-scale image from JAFFE database. (b) M in the x-direction. (c) M in the y-direction.

$$\mathrm{GF}_{ic_x} = \sum_{i=1}^{N}\left(G\frac{A_c * A_i}{d_{ic}^2} \times \sin\phi_{ic}\right) \tag{2}$$

$$\mathrm{GF}_{ic_y} = \sum_{i=1}^{N}\left(G\frac{A_c * A_i}{d_{ic}^2} \times \cos\phi_{ic}\right) \tag{3}$$

where N is the total number of neighboring pixels of a mask and d_{ic}^2 is the squared Euclidean distance between the ith pixel and the center pixel. GF_{ic_M} and GF_{ic_D} of GF_{ic} are calculated by the following equations:

$$\mathrm{GF}_{ic_M} = \sqrt{(\mathrm{GF}_{ic_x})^2 + (\mathrm{GF}_{ic_y})^2} \tag{4}$$

$$\mathrm{GF}_{ic_D} = \tan^{-1}\left(\frac{\mathrm{GF}_{ic_y}}{\mathrm{GF}_{ic_x}}\right). \tag{5}$$

Equations (4) and (5) can be used repeatedly by considering every pixel as a center pixel to find out M and D of gradients of a gray-scale image A.

B. Architecture of the Proposed DCNN

DCNNs learn features automatically and tend to describe the aimed task more accurately due to the parameters learning by backpropagation from the loss function of the aimed task. Existing DCNN-based models, such as VGG-16 and VGG-19, are built on a single branch sequentially connected with convolutional layers and usually focus on homogeneously scaled receptive fields and ignore detailed edge information. Thus, they lack gathering adequate features of spatial structure for facial appearance. Addressing this problem multiconvolutional networks was introduced. In this section, we introduce a novel DCNN for the FER task. The architecture of the proposed DCNN is shown in Fig. 3. It consists of two branches. The first branch is able to extract significant local features, such as edges, lines, curves, and corners from the M and D of an image, as shallow DCNN is designed. On the other hand,

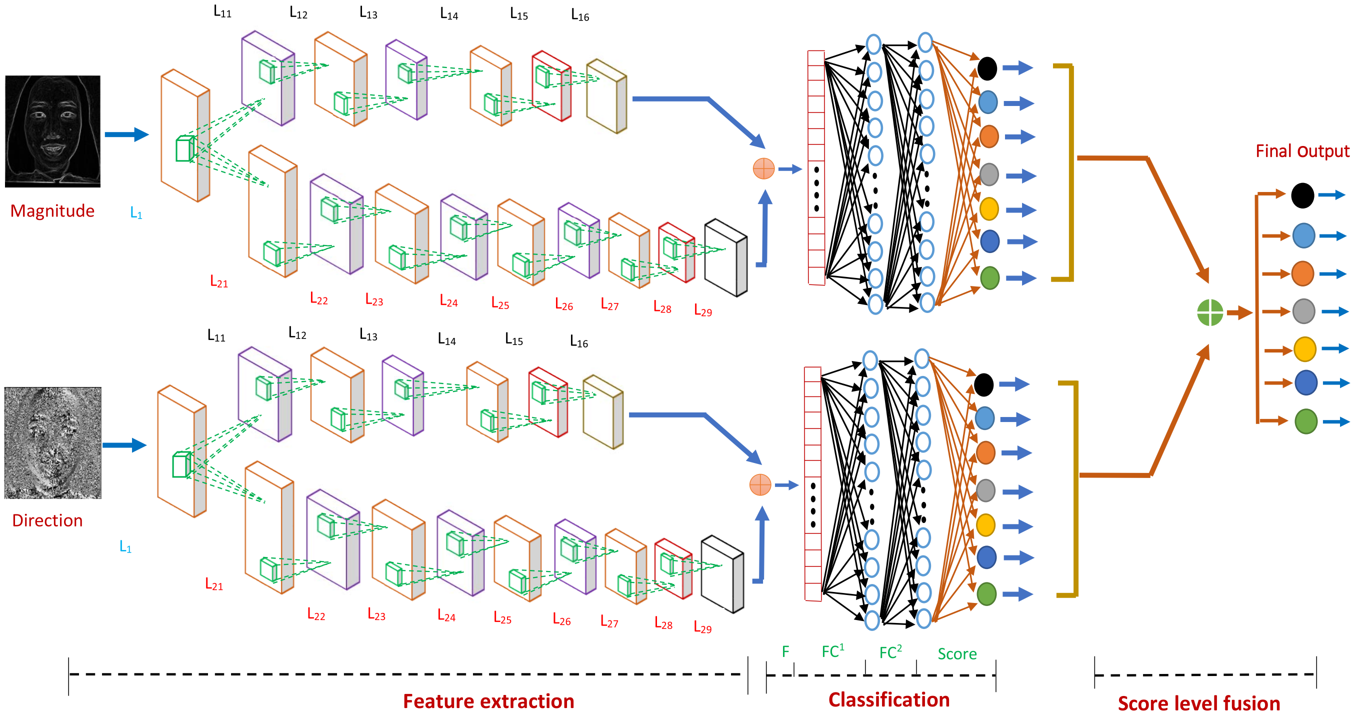


Fig. 3. Detailed DCNN architecture.

Layer	Type	Network Parameters				
L_1	Convolution	Filter size: 5 x 5	Filter number: 32	Stride: 1 x 1	Padding: Same	Activation Function = ReLU
L_{11}	Pooling	Pool size: 3 x 3		Stride: 3 x 3	Padding: Valid	Pooling Type: Max-Pooling
L_{12}	Convolution	Filter size: 4 x 4	Filter number: 32	Stride: 1 x 1	Padding: Same	Activation Function = ReLU
L_{13}	Pooling	Pool size: 3 x 3		Stride: 2 x 2	Padding: Valid	Pooling Type: Max-Pooling
L_{14}	Convolution	Filter size: 5 x 5	Filter number: 64	Stride: 1 x 1	Padding: Same	Activation Function = ReLU
L_{15}	Pooling	Pool size: 3 x 3		Stride: 2 x 2	Padding: Valid	Pooling Type: Average-Pooling
L_{16}	padding	Pad size: 4 x 4		-	-	Padding Type: Zero-padding
L_{21}	Convolution	Filter size: 5 x 5	Filter number: 32	Stride: 1 x 1	Padding: Same	Activation Function = ReLU
L_{22}	Pooling	Pool size: 5 x 5		Stride: 2 x 2	Padding: Valid	Pooling Type: Max-Pooling
L_{23}	Convolution	Filter size: 4 x 4	Filter number: 32	Stride: 1 x 1	Padding: Same	Activation Function = ReLU
L_{24}	Pooling	Pool size: 3 x 3		Stride: 2 x 2	Padding: Valid	Pooling Type: Max-Pooling
L_{25}	Convolution	Filter size: 5 x 5	Filter number: 64	Stride: 1 x 1	Padding: Same	Activation Function = ReLU
L_{26}	Pooling	Pool size: 3 x 3		Stride: 2 x 2	Padding: Valid	Pooling Type: Max-Pooling
L_{27}	Convolution	Filter size: 5 x 5	Filter number: 64	Stride: 1 x 1	Padding: Same	Activation Function = ReLU
L_{28}	Pooling	Pool size: 3 x 3		Stride: 2 x 2	Padding: Valid	Pooling Type: Average-Pooling
L_{29}	Up-sampling	Sample size: 4 x 4		-	-	Sample Type: Up-sampling
L_{16}, L_{29}	Early Fusion	Type: ⊕ ---- Concatenation				
F	Flatten	-----------------------				
FC_1	Full Connection	Number of Neurons = 1024				Dropout = 0.3
FC_2	Full Connection	Number of Neurons = 1024				Dropout = 0.3
Score	Output	Number of Neurons = 7				
Final Output	Score level Fusion	Type: ⊕ ----- Late Score Level Fusion				

Fig. 4. Various parameters used in the proposed DCNN architecture shown in Fig. 3 and their values.

the second branch is responsible for extracting the holistic features, which can differentiate one expression from others, since major DCNN is considered. Since M and D of an image are considered, the proposed DCNN is able to extract the features that are relevant to the individual expressions. The first branch of DCNN consists of three convolutional layers that are connected sequentially, namely, two max-pooling, one average pooling, and zero-padding; these are connected sequentially. On the other hand, the second branch of the DCNN network contains five convolutional, three max-pooling, one average-pooling, and an upsampling layer. Moreover, these two branches are concatenated and forwarded to the two dense layers for the classification of facial expressions. The detailed description of each layer and its parameters are shown in Fig. 4. Moreover, capturing the enriched contextual information filters 5 × 5 and 4 × 4 is employed. These filters allow the network to learn true edge variations. The biggest advantage of convolutional layers is that it is able to extract the features automatically, kth convolutional layer consists of n^k feature maps, denoted as F_p^k, where $p = 1, 2, 3, \ldots, n^k$, and k represents a particular convolutional layer. Each feature map, i.e., F_q^{k-1}, where $q = 1, 2, 3, \ldots, n^{k-1}$ from the $(k-1)$th convolutional layer is convolved with the filter W_{pq}^k and bias b_p^k is added. Furthermore, convolved feature maps are fed into the nonlinear activation function rectified linear unit (ReLu). Equation (6) shows how we can obtain a convolved feature map F_p^k

$$F_p^k = \Delta\left(\sum_{q=1}^{n^{k-1}} F_q^{k-1} * W_{pq}^k + b_p^k\right), \quad p = 1, 2, 3, \ldots n^k \tag{6}$$

where $*$ indicates the convolution operation. The responsibility of an activation function is to rework the weighted sum of input from one node to different activated nodes. Here, ReLU is adopted because it can reduce the value of cost/loss function by mitigating the vanishing gradient problem to some extent. It can compute faster and better performance on complex databases [71]. The mathematical representation of the ReLU activation function is shown in (7). Max-pooling is applied to convolved feature maps obtained by (6) to defeat the overfitting problem by providing an abstracted style of the representation of the convolved feature maps. Max-pooling calculates the utmost value of every patch from each feature map to spotlight the foremost presented feature within the patch. It also reduces the number of parameters in order to make the model simple. Moreover, it provides translation, rotation, and scale-invariant feature maps

$$\Delta\left(\sum_{q=1}^{n^{k-1}} F_q^{k-1} * W_{pq}^k + b_p^k\right) = \begin{cases} 0, & \text{if } \sum_{q=1}^{n^{k-1}} F_q^{k-1} * W_{pq}^k + b_p^k < 0 \\ \sum_{q=1}^{n^{k-1}} F_q^{k-1} * W_{pq}^k + b_p^k. & \text{otherwise.} \end{cases} \tag{7}$$

The feature maps, FM^{16} and FM^{29}, obtained from the layers L_{16} and L_{29} are concatenated. Then, the concatenated feature maps FM^{16} and FM^{29} are flattened by flattening layer F before feeding them as input FC^0 into the first fully connected layer. The output FC^1 of the first fully connected layer is fed into the second fully connected layer to generate FC^2 as output. The mathematical operation involved in two fully connected layers is denoted by the following equation:

$$\mathrm{FC}^i = W^i * \mathrm{FC}^{i-1} + b^i, \quad i = 1, 2 \tag{8}$$

where W_i and b^i are the weight and bias of the ith fully connected layer. It is observed from the experiments that the overfitting problem arises in both fully connected layers as a number of learnable parameters are associated with them. In this work, the dropout technique is adopted to resolve the overfitting problem that occurred in both the fully connected layers. The output of the second fully connected layer FC^2 is further fed into the softmax layer. The softmax layer consists of seven neurons and produces a probability vector, $\hat{y} = [\hat{y_1}, \hat{y_2}, \hat{y_3}, \hat{y_4}, \hat{y_5}, \hat{y_6}, \hat{y_7}]$. The probability vector consists of seven probability values as seven classes of facial expressions are considered in this study. The kth probability value is obtained by the following equation:

$$\hat{y_k} = \frac{e^{\mathrm{FC}_k^2}}{\sum_{k=1}^{7} e^{\mathrm{FC}_k^2}}, \quad k = 1, 2, \ldots, 7. \tag{9}$$

1) Network Training: The proposed network trains independently on M and D of gradients of facial images and estimates the probability of each class. The proposed network weights are initialized using the Glorot uniform method, and Adam optimization is employed to prevent local optimum [72], along with learning rate decay introduced to advance the training effect. In Adam optimization, initial learning rate and learning rate decay are assigned as 0.00001 and 1e-4, respectively. Categorical cross entropy is exploited to find the loss for multiclass classification, and it is a measure to quantify the error of our model. The categorical cross entropy is computed using the following equation:

$$\psi(y, \hat{y}) = -\frac{1}{7}\sum_{j=1}^{7} y_j \log \hat{y}_j \tag{10}$$

where $y(= [y_1, y_2, y_3, y_4, y_5, y_6, y_7])$ is one-hot encoding vector of the actual labels. Batch size 16 is considered while training the proposed DCNN since the network can occupy less memory in our system and 60 epochs are considered for JAFFE, CK+, KDEF, and RAF databases, while 200 epochs are used for FER2013.

2) Score-Level Fusion: To estimate the final prediction of seven basic expressions, the score-level fusion technique [73] is performed on M and D of gradients. Mathematically, score-level fusion is done by the following equation:

$$\mathrm{SF}_i = \arg\max_c \sum_{j=1}^{N} \alpha_j P_{ijc} \tag{11}$$

where c indicates the various expressions and that i represents the input sample and N indicates that the two different modalities, in this study M and D of the input image, are considered as two different modalities. P_{ijc} could be a prediction probability to belong to a class c for input sample i of modality j. The worth of α_j is chosen by searching values from 0 to 5 with a step size of 0.2. Score-level fusion is simpler to weigh individual scores of modalities, and it gave a better performance on the FER task.

IV. Experimental Results and Discussion

A. Environment Settings

In this study, the Keras framework and Anaconda development platform are considered for training and testing the proposed model. Python language is used since many of the deep-learning libraries are developed using it. The specifications of the system are reported in Table I.

TABLE I

Specification of the System

Name	Parameter
GPU RAM	16GB
Graphics processor	NVIDIA Quadro P5000
Cuda cores	2560
Memory interface	256-bit
Memory type	GDDR5X
Bandwidth	288.5 GB/s
Language	Python

TABLE II

Statistical Information of the Databases

Database	Image Size	Number of images							Total
		NE	AN	DI	FE	HA	SA	SU	
Before Augmentation									
FER2013	48 × 48	6183	4916	545	5089	8977	6069	3993	35772
JAFFE	256 × 256	30	30	29	32	31	31	30	213
CK+	256 × 256	50	47	61	24	59	28	62	331
KDEF	256 × 256	70	70	70	70	70	70	70	490
RAF	100 × 100	3204	867	877	355	5957	2460	1463	15183
After Augmentation									
FER2013	256 × 256	6183	4916	2725	5089	8977	6069	3993	37952
JAFFE	256 × 256	150	150	145	160	155	155	150	1065
CK+	256 × 256	250	235	305	120	295	140	310	1655
KDEF	256 × 256	350	350	350	350	350	350	350	2450
RAF	256 × 256	3204	4335	4385	1755	5957	2460	1463	23559

B. Database Description

In this work, five well-known benchmark databases, namely, FER2013 [32], JAFFE [33], CK+ [34], KDEF [35], and RAF [36], [37], are considered for the evaluation of the proposed network because these databases contain seven basic universal facial expressions, namely, NE, AN, DI, FE, HA, SA, and SU. The top of Table II describes the statistical information of the aforementioned databases.

C. Data Augmentation

Having a large database is important for measuring the performance of a DCNN model. Moreover, we can prevent a DCNN model from learning irrelevant features because irrelevant or partially relevant features can negatively impact model performance. However, the performance can be improved to some extent by augmenting the data that we already have. A DCNN can be invariant to translation, viewpoint, size, or illumination. Thus, some of the image processing techniques, such as the rotation of images by 5° clockwise and anticlockwise directions, horizontal flip, and adding Gaussian noise, are considered to extend the total number of images of seven facial expressions of JAFFE, CK+, and KDEF databases to increase the diversity of these databases. On the other hand, augmentation is done on DI facial expression of FER2013 and AN, DI, and FE facial expressions of RAF only due to their imbalance classes. The bottom of Table II describes the statistical information of five databases after data augmentation. Furthermore, each database is divided into three parts: training, validation, and testing. A tenfold cross-validation technique is adopted for all the experiments to evaluate the performance of the proposed method. In other words, out of ten subsets, eight subsets are used for training, one subset is considered for validation, and the rest of the one subset is adopted for testing. The average classification results are reported in this study. Table III shows the number of facial images used in the training, validation, and testing processes for each fold.

TABLE III

Train, Validation, and Test Split for the Five Databases

Database	Train	Validation	Test
FER2013	30362	3795	3795
JAFFE	851	107	107
CK+	1323	166	166
KDEF	1960	245	245
RAF	18847	2356	2356

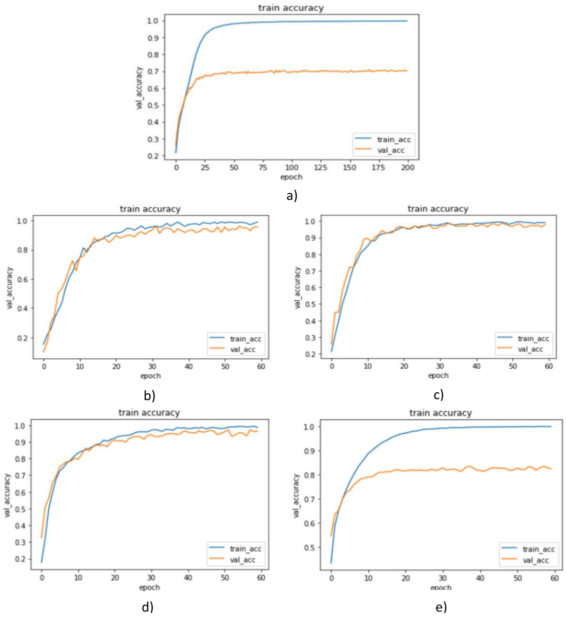


Fig. 5. Training and validation performances using M of GF on five databases. (a) FER2013. (b) JAFFE. (c) CK+. (d) KDEF. (e) RAF.

TABLE IV

Testing Accuracy and Loss When M of GF Is Used

Database	Testing Accuracy	Loss
FER2013	77.10%	0.4204
JAFFE	95.63%	0.1602
CK+	97.99%	0.0627
KDEF	96.36%	0.1485
RAF	82.33%	0.3820

D. Experimental Results Using M

In the first experiment, only M followed by the proposed DCNN is considered. In other words, the upper half of the proposed model is only used for training and validation. Thus, the upper half of the model is implemented in the abovementioned five databases. Fig. 5 shows training and validation performances with respect to the epoch on each database. The average testing accuracy and average loss on each database are reported in Table IV. It is clear from Table IV that results are very good on three databases: JAFFE, CK+, and KDEF. However, the results on FER2013 and RAF databases are relatively poor, but these could be accepted.

E. Experimental Results Using D

In the second experiment, only D followed by the proposed DCNN is adopted. In other words, the lower half of the

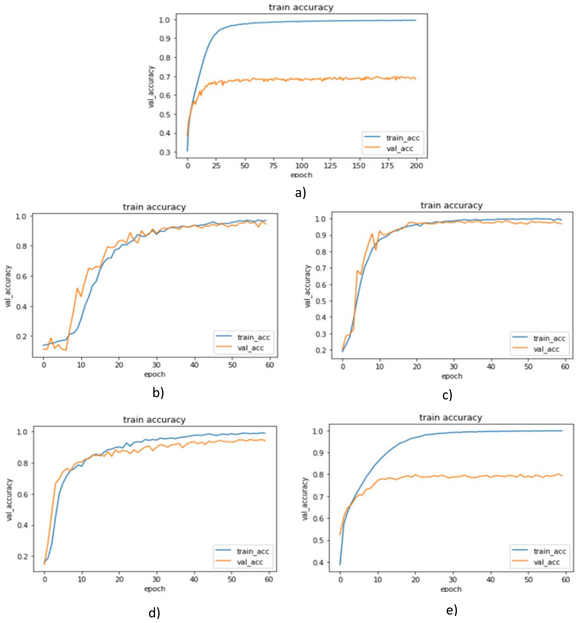


Fig. 6. Training and validation performances using D of GF on five databases. (a) FER2013. (b) JAFFE. (c) CK+. (d) KDEF. (e) RAF.

TABLE V

TESTING ACCURACY AND LOSS WHEN D OF GF IS CONSIDERED

Database	Testing Accuracy	Loss
FER2013	76.27%	0.4589
JAFFE	94.38%	0.1988
CK+	96.79%	0.0685
KDEF	94.12%	0.2040
RAF	79.27%	0.3910

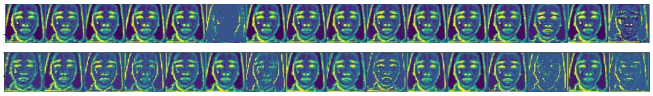


Fig. 7. Feature maps of the proposed DCNN. The primary row represents first branch followed by the second branch before the secondary pooling layer.

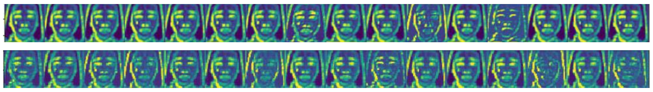
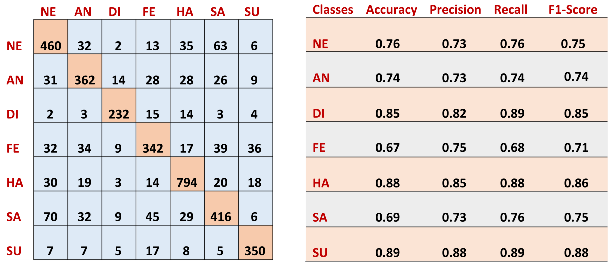


Fig. 8. Feature maps of proposed DCNN. The primary row represents first branch and followed by second branch before the secondary pooling layer.

proposed model is only considered for training and validation. Thus, the lower half of the model is executed on the above mentioned five databases. Fig. 6 shows training and validation accuracies with respect to iteration on each database. The average testing accuracy and average loss on each database are noted in Table V. It is observed from Table V that the results follow the same trend as Table IV. However, the performance is deteriorated when D is used followed by the proposed DCNN.

F. Experimental Results Using Both M and D

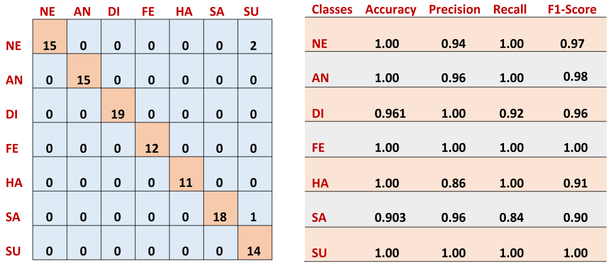
In the third experiment, the complete model depicted in Fig. 3 is used. The proposed model is run on five databases. Figs. 7 and 8 display the intermediate features maps obtained by the proposed DCNN. Figs. 9–13 show the one out of ten confusion matrices and the classification report obtained from the given confusion matrix on FER2013, JAFFE, CK+, KDEF, and RAF databases, respectively. The reported classification report of each database is obtained by averaging the classification reports of tenfold separately. The average accuracy obtained by the proposed model on five databases is reported in the last row of Table VIII.



	NE	AN	DI	FE	HA	SA	SU
NE	460	32	2	13	35	63	6
AN	31	362	14	28	28	26	9
DI	2	3	232	15	14	3	4
FE	32	34	9	342	17	39	36
HA	30	19	3	14	794	20	18
SA	70	32	9	45	29	416	6
SU	7	7	5	17	8	5	350

Classes	Accuracy	Precision	Recall	F1-Score
NE	0.76	0.73	0.76	0.75
AN	0.74	0.73	0.74	0.74
DI	0.85	0.82	0.89	0.85
FE	0.67	0.75	0.68	0.71
HA	0.88	0.85	0.88	0.86
SA	0.69	0.73	0.76	0.75
SU	0.89	0.88	0.89	0.88

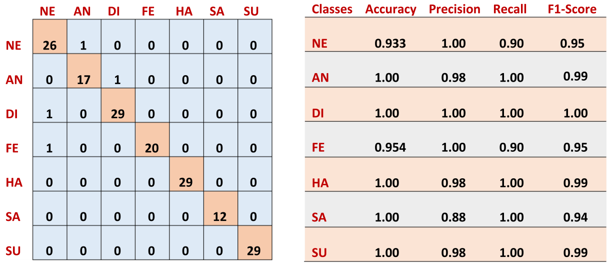
Fig. 9. Performance in terms of confusion matrix and classification report on the FER2013 database.



	NE	AN	DI	FE	HA	SA	SU
NE	15	0	0	0	0	0	2
AN	0	15	0	0	0	0	0
DI	0	0	19	0	0	0	0
FE	0	0	0	12	0	0	0
HA	0	0	0	0	11	0	0
SA	0	0	0	0	0	18	1
SU	0	0	0	0	0	0	14

Classes	Accuracy	Precision	Recall	F1-Score
NE	1.00	0.94	1.00	0.97
AN	1.00	0.96	1.00	0.98
DI	0.961	1.00	0.92	0.96
FE	1.00	1.00	1.00	1.00
HA	1.00	0.86	1.00	0.91
SA	0.903	0.96	0.84	0.90
SU	1.00	1.00	1.00	1.00

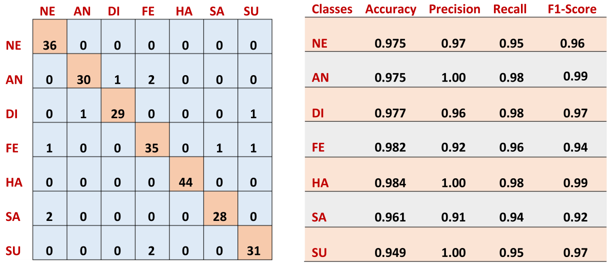
Fig. 10. Performance in terms of confusion matrix and classification report on the JAFFE database.



	NE	AN	DI	FE	HA	SA	SU
NE	26	1	0	0	0	0	0
AN	0	17	1	0	0	0	0
DI	1	0	29	0	0	0	0
FE	1	0	0	20	0	0	0
HA	0	0	0	0	29	0	0
SA	0	0	0	0	0	12	0
SU	0	0	0	0	0	0	29

Classes	Accuracy	Precision	Recall	F1-Score
NE	0.933	1.00	0.90	0.95
AN	1.00	0.98	1.00	0.99
DI	1.00	1.00	1.00	1.00
FE	0.954	1.00	0.90	0.95
HA	1.00	0.98	1.00	0.99
SA	1.00	0.88	1.00	0.94
SU	1.00	0.98	1.00	0.99

Fig. 11. Performance in terms of confusion matrix and classification report on the CK+ database.



	NE	AN	DI	FE	HA	SA	SU
NE	36	0	0	0	0	0	0
AN	0	30	1	2	0	0	0
DI	0	1	29	0	0	0	1
FE	1	0	0	35	0	1	1
HA	0	0	0	0	44	0	0
SA	2	0	0	0	0	28	0
SU	0	0	0	2	0	0	31

Classes	Accuracy	Precision	Recall	F1-Score
NE	0.975	0.97	0.95	0.96
AN	0.975	1.00	0.98	0.99
DI	0.977	0.96	0.98	0.97
FE	0.982	0.92	0.96	0.94
HA	0.984	1.00	0.98	0.99
SA	0.961	0.91	0.94	0.92
SU	0.949	1.00	0.95	0.97

Fig. 12. Performance in terms of confusion matrix and classification report on the KDEF database.

TABLE VI

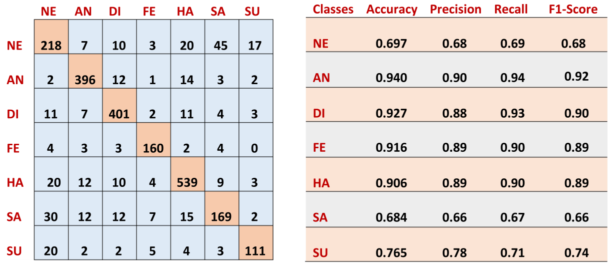
COMPARISON OF GF DESCRIPTOR WITH OTHER DESCRIPTORS

Database	LBP		Gobar Filter		Gravitational Force	
	compactness	separation	compactness	separation	compactness	separation
FER2013	2.1007	11.1001	1.6723	13.3321	0.3792	17.2005
JAFFE	2.5133	16.9478	0.3947	18.3839	0.2783	32.0319
CK+	0.9196	24.2100	0.4641	22.5207	0.2941	33.2091
KDEF	1.7509	18.0879	0.1164	11.9973	0.0976	28.9084
RAF	1.4702	7.8901	0.9493	9.4510	0.2301	13.0923

TABLE VII

Feature Extraction Analysis of All the Compared Methods and Proposed One for All the Five Databases

Method	FER2013		JAFFE		CK+		KDEF		RAF	
	compactness	separation	compactness	separation	compactness	separation	compactness	separation	compactness	separation
HOG-TOP [42]	6.3401	11.9010	11.4307	18.1801	9.1060	24.1107	12.4987	18.2312	5.9019	9.2809
Shallow CNN [45]	5.2903	13.6850	13.1721	16.8956	10.1040	21.1652	12.9472	17.9843	2.4212	13.5219
Major CNN [45]	3.2760	16.0395	7.2167	19.9009	4.3400	36.6180	9.8661	23.2107	1.0192	15.9305
Shallow CNN on LBP images [50]	5.0081	12.5627	3.3133	23.6127	4.9092	34.4410	8.3283	25.9169	2.8358	13.9102
Shallow CNN on gray-scale images [50]	2.0843	18.8401	2.1050	27.1873	2.0969	41.9901	6.3166	28.0498	1.1023	15.8001
Partial VGG16 [50]	2.1037	18.9823	1.6430	31.0262	3.0974	38.5981	6.0116	29.0963	4.6783	10.3281
Weighted mixture of double channel [50]	0.1702	20.9437	2.0051	27.9642	1.1268	46.0533	4.2403	31.1106	1.7281	14.1421
Weighted fusion of three subnetworks [52]	3.0103	15.6149	2.7613	26.2167	3.1943	40.9165	7.2389	26.9901	2.3929	13.6780
Appearance based CNN on LBP images [54]	7.9823	10.6843	2.9124	26.8913	1.8412	43.0935	8.9982	22.9307	1.5816	14.4389
Fusion of appearance and geometric features [54]	5.8124	12.0844	2.0962	27.6893	2.0082	42.2190	8.6826	24.2411	1.0741	16.0381
3D Inception-ResNet + landmarks [56]	2.1862	19.0125	1.9863	28.3013	2.5137	40.9810	3.9198	32.0250	2.4927	13.4617
Autoencoders + SOM [57]	1.9810	19.5213	1.8047	29.1677	2.4176	41.0869	3.9826	32.6810	1.9074	14.4837
Spatio-temporal feature + LSTM [58]	2.3814	18.9046	2.5913	26.9033	5.1905	32.1890	4.3001	31.0241	2.6481	14.0102
Ensemble DCNNs [59]	5.6321	12.8610	12.1622	17.4612	8.9567	26.6010	11.1725	19.4607	2.4217	13.6550
DCNN for binary classification [60]	7.1027	11.0913	12.8907	17.9917	6.0230	30.3851	7.9604	26.0021	3.0429	12.3489
IACNN [61]	2.9467	19.8437	6.3401	21.2157	1.8102	43.1880	9.6007	24.0420	1.8489	14.1013
2B(N+M)Softmax [62]	2.7890	19.5246	5.8804	21.9672	4.0801	36.7013	4.2509	30.9103	3.0134	12.6581
S-DSRN [63]	2.0133	19.0081	2.2081	27.8534	3.0162	39.4693	3.8964	32.8158	1.7618	14.1023
Color features + Gabor transform [64]	2.6448	18.6162	6.0182	20.6420	3.9180	36.9401	4.8321	30.9903	2.5001	13.2978
DCMA-CNNs [65]	1.8844	19.8916	1.8441	30.8757	2.4503	41.2251	3.7987	32.6380	1.0127	16.4190
Broad Learning [66]	8.9162	10.2761	1.9962	29.0123	5.2629	31.0829	2.0312	38.1023	3.7320	11.8902
Ensemble MLCNNs [67]	1.2491	20.0617	1.8001	29.8962	2.4984	41.0012	2.6512	33.8138	0.9326	16.8721
Deep-Emotion [68]	2.5193	19.2013	1.9125	28.9102	2.2630	41.8230	4.3612	31.0883	2.5216	14.1852
VGG19 [47]	1.1347	20.1933	1.3237	30.9817	1.4866	44.1805	4.3791	31.1990	2.9987	12.9029
ResNet150 [47]	0.1672	20.0347	2.4340	27.1583	2.8682	38.0113	7.5981	26.0894	2.4901	13.4823
Proposed method	**0.0916**	**23.4823**	**0.1525**	**36.3913**	**0.2760**	**51.9503**	**0.0844**	**46.5627**	**0.0727**	**17.1048**



	NE	AN	DI	FE	HA	SA	SU
NE	218	7	10	3	20	45	17
AN	2	396	12	1	14	3	2
DI	11	7	401	2	11	4	3
FE	4	3	3	160	2	4	0
HA	20	12	10	4	539	9	3
SA	30	12	12	7	15	169	2
SU	20	2	2	5	4	3	111

Classes	Accuracy	Precision	Recall	F1-Score
NE	0.697	0.68	0.69	0.68
AN	0.940	0.90	0.94	0.92
DI	0.927	0.88	0.93	0.90
FE	0.916	0.89	0.90	0.89
HA	0.906	0.89	0.90	0.89
SA	0.684	0.66	0.67	0.66
SU	0.765	0.78	0.71	0.74

Fig. 13. Performance in terms of confusion matrix and classification report on the RAF database.

G. Model Analysis

The proposed method consists of two steps. Extraction of edge information using the GF feature descriptor is done in the first step, whereas the proposed DCNN tunes the edge information to extract local and holistic features in the second step. Bhattacharjee and Roy [31] already showed that the GF feature descriptor performs better than other edge information extraction techniques. Thus, we have not conducted the same experiment again. However, the performance of the GF feature extractor is compared with two well-known texture measures named LBP and Gabor face descriptors in this study. A face image of size 256 × 256 pixels is fed as an input to the abovementioned three descriptors separately, and they produce a feature vector of size 1 × 65 536 as an output. Then, the obtained feature vector is mapped into a d-dimensional polyhedron to get a point, where the value of d is $256 \times 256 = 65\,536$. This process would be repeated for all the face images of a database. We will get a d-dimensional polyhedron at the end of this process, where a face image would be represented as a point. A feature descriptor is not good enough if the points of two classes are highly overlapping. When the points of the two classes are highly overlapping, the accuracy would decrease. Here, the overlap is computed based on compactness and separation. Compactness and separation define the quality of clustering results. A cluster has good compactness when points are close to each other and good separation when clusters do not overlap. In other words, the ideal values of compactness and separation are zero and infinity, respectively. Initially, k-means is applied to the points to divide them into $k = 7$ clusters as seven basic expressions are considered in this study. The values of compactness and separation are computed for the three abovementioned feature descriptors on five databases that are reported in Table VI.

It is noticed from Table VI that the value of compactness of the GF is less compared with LBP and Gabor face descriptor. On the other hand, the value separation of the GF is the highest among the three feature descriptors. Thus, we can conclude that the GF is better compared with LBP and Gabor face descriptor. In other words, the feature vector is more informative and is able to distinguish a facial expression from others when GF is used. The same experiment is conducted after extracting features by the proposed DCNN and other state-of-the-art models, and the values of compactness and separation are noted in Table VII.

Two conclusions can be drawn from Table VII: GF and the proposed DCNN jointly generate features that have more distinctive capabilities than the features produced by the GF alone as the value of compactness is less and the value of separation is high. We can, thus, state that GF followed by the proposed DCNN is better than the state-of-the-art methods for the same reason.

H. Comparative Results

In the last experiment, we provide comparative results against 25 state-of-the-art algorithms, for example, HOG-TOP [42], shallow CNN [45], major CNN [45], shallow CNN on LBP images [50], shallow CNN on gray-scale images [50], partial VGG16 [50], weighted mixture of double channel [50], weighted fusion of three subnetworks [52], appearance-based CNN on LBP images [54], fusion of appearance and geometric features [54], 3-D Inception-ResNet + landmarks [56], autoencoders + SOM [57], spatiotemporal feature + LSTM [58], ensemble DCNNs [59], DCNN for binary classification [60], IACNN [61], 2B(N + M)Softmax [62], S-DSRN [63], color features + Gabor transform [64], DCMA-CNNs [65], broad learning [66], ensemble MLCNNs [67], deep-emotion [68], VGG19 [47], and ResNet150 [47], on five publicly available databases. However, the comparison is done based on average recognition accuracy only. Some of the abovementioned

TABLE VIII

Classification Accuracy (%) and Execution Time in Seconds on Five Databases: FER2013, JAFFE, CK+, KDEF, and RAF by Various Methods

Classification Accuracy (%) on Five Databases							Execution Time in Minutes										
Sl. No.	Method	FER2013	JAFFE	CK+	KDEF	RAF	TTPI	Training Time					Testing Time for all the Images				
								FER2013	JAFFE	CK+	KDEF	RAF	FER2013	JAFFE	CK+	KDEF	RAF
1.	HOG-TOP [42]	52	60	65	55	53	0.4500	220.0	86.0	92.33	115.0	165.0	2.1623	1.2124	1.3264	1.1576	2.1529
2.	Shallow CNN [45]	55	50	61	55	70	0.1812	5.5	1.0	1.5	2.5	4.0	0.9992	0.8845	0.8985	0.9001	0.9845
3.	Major CNN [45]	64	68	85	67	78	0.1872	9.0	1.5	2.0	2.5	6.0	1.1060	0.8883	0.9127	0.9168	1.0921
4.	Shallow CNN on LBP images [50]	54	88	83	70	71	0.1801	75.0	8.33	16.33	25.0	41.66	1.0721	0.8821	0.8991	0.9008	0.9698
5.	Shallow CNN on gray-scale images [50]	72	92	94	78	78	0.1801	75.0	8.33	16.33	25.0	41.66	1.0721	0.8821	0.8991	0.9008	0.9698
6.	Partial VGG16 [50]	72	96	91	78	57	0.4309	441.66	150.0	158.33	175.0	316.66	1.9348	1.4508	1.5238	1.5523	1.8900
7.	Weighted mixture of double channel [50]	75	92	97	81	75	0.4699	516.66	158.33	174.99	200.0	358.32	2.2602	1.9665	1.9789	1.9965	2.1056
8.	Weighted fusion of three subnetworks [52]	63	90	91	74	70	0.8956	150.0	9.63	10.0	16.6	70.0	2.2012	1.9804	1.9912	1.9989	2.1020
9.	Appearance based CNN on LBP images [54]	50	90	95	66	77	0.3945	105.0	10.0	20.0	30.0	70.0	1.3563	1.1230	1.1321	1.1394	1.2953
10.	Fusion of appearance and geometric features [54]	53	90	94	69	78	0.4612	280.0	70.0	110.0	135.0	210.0	2.3906	1.9023	1.9984	2.0102	2.2134
11.	3D Inception-ResNet + landmarks [56]	72	93	93	83	70	0.8926	340.0	110.0	140.0	186.0	213.0	2.3472	1.9245	1.8612	1.9946	2.1980
12.	Autoencoders + SOM [57]	73	93	94	84	76	0.3298	240.0	85.0	96.66	110.0	153.33	2.1089	1.1426	1.1623	1.1982	1.9036
13.	Spatio-temporal feature + LSTM [58]	71	90	82	81	73	0.4329	460.0	205.0	215.33	240.0	340.0	3.1652	2.6743	2.8628	2.9845	3.1107
14.	Ensemble DCNNs [59]	53	55	67	58	70	0.4917	420.0	200.0	220.0	300.0	360.0	3.8138	2.5827	2.6296	2.7013	3.4238
15.	DCNN for binary classification [60]	50	56	73	70	68	0.1725	103.33	24.66	28.0	37.33	75.0	1.6239	1.3469	1.4430	1.4523	1.5426
16.	IACNN [61]	68	75	95	67	74	0.3946	1033.0	600.0	633.0	700.0	866.66	2.1623	1.8920	1.9165	1.9730	2.0935
17.	2B(N+M)Softmax [62]	67	78	87	81	69	0.2814	265.0	80.0	85.33	93.33	165.0	1.1721	0.9643	0.9892	0.9946	1.0021
18.	S-DSRN [63]	72	92	92	83	75	0.1927	980.0	320.0	340.0	380.33	460.0	2.3042	1.9756	1.9878	2.0262	2.1040
19.	Color features + Gabor transform [64]	66	77	86	80	70	0.3218	240.0	98.66	100.0	115.33	180.66	2.8794	2.2167	2.4510	2.5503	2.7165
20.	DCMA-CNNs [65]	73	95	93	83	78	0.3129	340.0	160.0	180.0	190.0	230.0	1.2439	1.1092	1.1123	1.1706	1.2034
21.	Broad Learning [66]	44	93	81	89	64	0.1023	7.12	1.0	1.5	2.0	4.5	0.4812	0.3101	0.3323	0.3812	0.4102
22.	Ensemble MLCNNs [67]	74	94	93	85	79	0.5612	88.33	23.33	25.0	28.33	63.33	2.3024	1.9823	1.9986	2.0145	2.1876
23.	Deep-Emotion [68]	70	93	94	81	72	0.2908	316.0	41.6	50.0	66.66	241.66	1.0982	0.8943	0.9165	0.9346	0.9978
24.	VGG19 [47]	74	95	96	81	60	0.6128	45.5	22.0	23.0	26.5	34.5	2.2123	1.9821	1.9901	2.0981	2.1823
25.	ResNet150 [47]	75	91	89	72	70	0.7123	130.0	54.0	67.0	76.5	105.5	3.1190	2.5981	2.6180	2.7833	2.9009
26.	**Proposed method**	**78**	**98**	**98**	**96**	**83**	0.3039	960.0	200.0	240.0	320.0	400.0	2.0974	1.6919	1.7346	1.7983	1.9029

methods were implemented on videos. Few works considered less number of classes. Thus, we change a few of these algorithms for this study by keeping the overall architecture the same. Table VIII shows the average classification accuracies achieved by the abovementioned state-of-the-art methods. It is clear from Table VIII that the proposed model defeats all the 25 existing methods on five databases, and it happens due to the use of GF-based local edge features along with holistic features extracted by the proposed DCNN, which is our main focus. However, all the methods are also compared based on training and testing times. Generally, the training time of a method depends on the size of the network, size of the input, number of epochs, number of folds, and others. In this study, all the models are implemented according to their respective specifications. However, we consider tenfold cross validation and 60 epochs while training the proposed method on all the databases except FER2013. The proposed method is trained for 200 epochs for the FER2013 database only. The training and testing times required by all the state-of-the-art methods, including the proposed method on all the five databases, are reported in Table VIII. However, testing time for onefold cross validation is noted only in Table VIII. Testing time per image (TTPI) is the same for all the images of a database as their size is equal. However, it varies from one method to another. It is clear from Table VIII that the proposed DCNN takes an average training and testing time across all the databases except the FER2013 database. The proposed method takes about 960 min to train the proposed method for FER2013, which is quite large.

V. Conclusion

It is clear from the empirical results that the proposed method can efficiently handle the problem of FER using static/still images. Facial expressions under lab-controlled environments are different from those in the wild, which are more natural and spontaneous. Thus, three databases, namely, JAFFE, CK+, and KDEF, developed in a lab-controlled environment are considered in this work. This study also adopted two databases, namely, FER2013 and RAF, built in the wild to demonstrate the efficacy of the proposed method over state-of-the-art methods. A novel DCNN framework is introduced to extract holistic features for identifying facial expression. However, before the use of the proposed DCNN model, a GF-based edge descriptor is adopted to fetch the low-level local features. The GF-based edge descriptor produces two intermediate local features, namely, M and D. At the end of the proposed DCNN model, a softmax classifier is used to compute the probability values in favor of either seven facial expressions. Finally, a score-level fusion technique is employed to combine the outputs obtained by the proposed model using M and D. The proposed method achieves an average recognition accuracy of 78%, 98%, 98%, 96%, and 83% for FER2013, JAFFE, CK+, KDEF, and RAF, respectively. Empirical results demonstrate that local and holistic features can together enhance the FER task. Experimental results also illustrate that the proposed method outperforms 25 baseline methods by considering the average time. However, the performance is generally not as good as that in FER under a lab-controlled environment, which deserves further study. Moreover, it is worth investigating to deploy the proposed model in some real-life applications.

References

[1] R. A. Calvo and S. D'Mello, "Affect detection: An interdisciplinary review of models, methods, and their applications," *IEEE Trans. Affect. Comput.*, vol. 1, no. 1, pp. 18–37, Jan. 2010.

[2] C. A. Corneanu, M. O. Simon, J. F. Cohn, and S. E. Guerrero, "Survey on RGB, 3D, thermal, and multimodal approaches for facial expression recognition: History, trends, and affect-related applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 8, pp. 1548–1568, Aug. 2016.

[3] P. Werner, A. Al-Hamadi, K. Limbrecht-Ecklundt, S. Walter, S. Gruss, and H. C. Traue, "Automatic pain assessment with facial activity descriptors," *IEEE Trans. Affect. Comput.*, vol. 8, no. 3, pp. 286–299, Jul. 2017.

[4] C. Liu and H. Wechsler, "Gabor feature based classification using the enhanced Fisher linear discriminant model for face recognition," *IEEE Trans. Image Process.*, vol. 11, no. 4, pp. 467–476, Apr. 2002.

[5] A. Seal, S. Ganguly, D. Bhattacharjee, M. Nasipuri, and D. K. Basu, "Thermal human face recognition based on Haar wavelet transform and series matching technique," in *Multimedia Processing, Communication and Computing Applications*. Bengaluru, India: PES Institute of Technology, 2013, pp. 155–167.

[6] D. Bhattacharjee, A. Seal, S. Ganguly, M. Nasipuri, and D. K. Basu, "A comparative study of human thermal face recognition based on Haar wavelet transform and local binary pattern," *Comput. Intell. Neurosci.*, vol. 2012, pp. 1–12, Jan. 2012.

[7] M. Kopaczka, R. Kolk, J. Schock, F. Burkhard, and D. Merhof, “A thermal infrared face database with facial landmarks and emotion labels,” *IEEE Trans. Instrum. Meas.*, vol. 68, no. 5, pp. 1389–1401, May 2019.

[8] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2005, pp. 886–893.

[9] A. Seal, D. Bhattacharjee, M. Nasipuri, C. Gonzalo-Martin, and E. Menasalvas, “Histogram of bunched intensity values based thermal face recognition,” in *Rough Sets and Intelligent Systems Paradigms*. Madrid, Spain: Springer, 2014, pp. 367–374.

[10] S. Ontañón, J. L. Montaña, and A. J. Gonzalez, “A dynamic-Bayesian network framework for modeling and evaluating learning from observation,” *Expert Syst. Appl.*, vol. 41, no. 11, pp. 5212–5226, Sep. 2014.

[11] A. Seal, S. Ganguly, D. Bhattacharjee, M. Nasipuri, and D. Kr. Basu, “Automated thermal face recognition based on minutiae extraction,” 2013, *arXiv:1309.1000*. [Online]. Available: http://arxiv.org/abs/1309.1000

[12] Y. Gao, M. K. H. Leung, S. Cheung Hui, and M. W. Tananda, “Facial expression recognition from line-based caricatures,” *IEEE Trans. Syst., Man, Cybern. A, Syst. Humans*, vol. 33, no. 3, pp. 407–412, May 2003.

[13] M. D. Cordea, E. M. Petriu, and D. C. Petriu, “Three-dimensional head tracking and facial expression recovery using an anthropometric muscle-based active appearance model,” *IEEE Trans. Instrum. Meas.*, vol. 57, no. 8, pp. 1578–1588, Aug. 2008.

[14] Z. Xu, H. R. Wu, X. Yu, K. Horadam, and B. Qiu, “Robust shape-feature-vector-based face recognition system,” *IEEE Trans. Instrum. Meas.*, vol. 60, no. 12, pp. 3781–3791, Dec. 2011.

[15] J. Li *et al.*, “Facial expression recognition with faster R-CNN,” *Procedia Comput. Sci.*, vol. 107, pp. 135–140, Jan. 2017.

[16] Y. Sun, X. Wang, and X. Tang, “Deep learning face representation from predicting 10,000 classes,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1891–1898.

[17] P. Liu, S. Han, Z. Meng, and Y. Tong, “Facial expression recognition via a boosted deep belief network,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1805–1812.

[18] H. Jung, S. Lee, J. Yim, S. Park, and J. Kim, “Joint fine-tuning in deep neural networks for facial expression recognition,” in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 2983–2991.

[19] C. Liu and H. Wechsler, “Enhanced Fisher linear discriminant models for face recognition,” in *Proc. 14th Int. Conf. Pattern Recognit.*, Aug. 1998, pp. 1368–1372.

[20] T. Ahonen, A. Hadid, and M. Pietikainen, “Face description with local binary patterns: Application to face recognition,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 12, pp. 2037–2041, Dec. 2006.

[21] D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, Nov. 2004.

[22] Z. Li, “A discriminative learning convolutional neural network for facial expression recognition,” in *Proc. 3rd IEEE Int. Conf. Comput. Commun. (ICCC)*, Dec. 2017, pp. 1641–1646.

[23] M. Mirza and S. Osindero, “Conditional generative adversarial nets,” 2014, *arXiv:1411.1784*. [Online]. Available: http://arxiv.org/abs/1411.1784

[24] H. Yang, Z. Zhang, and L. Yin, “Identity-adaptive facial expression recognition through expression regeneration using conditional generative adversarial networks,” in *Proc. 13th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG)*, May 2018, pp. 294–301.

[25] Y. Lv, Z. Feng, and C. Xu, “Facial expression recognition via deep learning,” in *Proc. Int. Conf. Smart Comput.*, Nov. 2014, pp. 745–750.

[26] M. T. B. Iqbal, M. Abdullah-Al-Wadud, B. Ryu, F. Makhmudkhujaev, and O. Chae, “Facial expression recognition with neighborhood-aware edge directional pattern (NEDP),” *IEEE Trans. Affect. Comput.*, vol. 11, no. 1, pp. 125–137, Jan. 2020.

[27] A. Bhavsar and H. M. Patel, “Facial expression recognition using neural classifier and fuzzy mapping,” in *Proc. Annu. IEEE India Conf. Indicon*, Dec. 2005, pp. 383–387.

[28] T. Jabid, “Robust facial expression recognition based on local directional pattern,” *ETRI J.*, vol. 32, no. 5, pp. 784–794, Oct. 2010.

[29] P. Zhao-yi, Z. Yan-hui, and Z. Yu, “Real-time facial expression recognition based on adaptive canny operator edge detection,” in *Proc. 2nd Int. Conf. Multimedia Inf. Technol.*, Apr. 2010, pp. 154–157.

[30] R. Samad and H. Sawada, “Edge-based facial feature extraction using Gabor wavelet and convolution filters,” in *Proc. MVA*, vol. 2011, pp. 430–433.

[31] D. Bhattacharjee and H. Roy, “Pattern of local gravitational Force(PLGF): A novel local image descriptor,” *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Aug. 1, 2019, doi: 10.1109/TPAMI.2019.2930192.

[32] I. J. Goodfellow *et al.*, “Challenges in representation learning: A report on three machine learning contests,” in *Proc. Int. Conf. Neural Inf. Process.* Daegu, South Korea: Springer, 2013, pp. 117–124.

[33] M. J. Lyons, S. Akamatsu, M. Kamachi, J. Gyoba, and J. Budynek, “The Japanese female facial expression (JAFFE) database,” in *Proc. 3rd Int. Conf. Autom. Face Gesture Recognit.*, 1998, pp. 14–16.

[34] P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews, “The extended Cohn-Kanade dataset (CK+): A complete dataset for action unit and emotion-specified expression,” in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. - Workshops*, Jun. 2010, pp. 94–101.

[35] D. Lundqvist, A. Flykt, and A. Öhman, “The Karolinska Directed Emotional Faces (KDEF),” *CD ROM Dept. Clin. Neurosci., Psychol. Sect., Karolinska Institutet*, vol. 91, no. 630, p. 2, 1998.

[36] S. Li and W. Deng, “Reliable crowdsourcing and deep locality-preserving learning for unconstrained facial expression recognition,” *IEEE Trans. Image Process.*, vol. 28, no. 1, pp. 356–370, Jan. 2019.

[37] S. Li, W. Deng, and J. Du, “Reliable crowdsourcing and deep locality-preserving learning for expression recognition in the wild,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2584–2593.

[38] X. Chen and W. Cheng, “Facial expression recognition based on edge detection,” *Int. J. Comput. Sci. Eng. Surv.*, vol. 6, no. 2, pp. 1–9, Apr. 2015.

[39] M. Abdulrahman and A. Eleyan, “Facial expression recognition using support vector machines,” in *Proc. 23nd Signal Process. Commun. Appl. Conf. (SIU)*, May 2015, pp. 276–279.

[40] C. Laurent, G. Pereyra, P. Brakel, Y. Zhang, and Y. Bengio, “Batch normalized recurrent neural networks,” in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2016, pp. 2657–2661.

[41] D. Ghimire and J. Lee, “Extreme learning machine ensemble using bagging for facial expression recognition,” *J. Inf. Process. Syst.*, vol. 10, no. 3, pp. 443–458, Sep. 2014.

[42] J. Chen, Z. Chen, Z. Chi, and H. Fu, “Facial expression recognition in video with multiple feature fusion,” *IEEE Trans. Affect. Comput.*, vol. 9, no. 1, pp. 38–50, Jan. 2018.

[43] Z. Yu and C. Zhang, “Image based static facial expression recognition with multiple deep network learning,” in *Proc. ACM Int. Conf. Multimodal Interact. - ICMI*, 2015, pp. 435–442.

[44] X. Zhao *et al.*, “Peak-piloted deep network for facial expression recognition,” in *Proc. Eur. Conf. Comput. Vis.* Amsterdam, The Netherlands: Springer, 2016, pp. 425–442.

[45] S. Alizadeh and A. Fazel, “Convolutional neural networks for facial expression recognition, 1704.06756,” Stanford Univ., Stanford, CA, USA, Tech. Rep., 2017.

[46] Y. Tang, “Deep learning using linear support vector machines,” 2013, *arXiv:1306.0239*. [Online]. Available: http://arxiv.org/abs/1306.0239

[47] D. Orozco, C. Lee, Y. Arabadzhi, and D. Gupta, “Transfer learning for facial expression recognition,” Florida State Univ., Tallahassee, FL, USA, Tech. Rep. 7, 2018.

[48] Y. Sun, Y. Chen, X. Wang, and X. Tang, “Deep learning face representation by joint identification-verification,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 1988–1996.

[49] E. Barsoum, C. Zhang, C. C. Ferrer, and Z. Zhang, “Training deep networks for facial expression recognition with crowd-sourced label distribution,” in *Proc. 18th ACM Int. Conf. Multimodal Interact. ICMI*, 2016, pp. 279–283.

[50] B. Yang, J. Cao, R. Ni, and Y. Zhang, “Facial expression recognition using weighted mixture deep neural network based on double-channel facial images,” *IEEE Access*, vol. 6, pp. 4630–4640, 2018.

[51] A. Ravi, “Pre-trained convolutional neural network features for facial expression recognition,” 2018, *arXiv:1812.06387*. [Online]. Available: http://arxiv.org/abs/1812.06387

[52] W. Hua, F. Dai, L. Huang, J. Xiong, and G. Gui, “HERO: Human emotions recognition for realizing intelligent Internet of Things,” *IEEE Access*, vol. 7, pp. 24321–24332, 2019.

[53] A. Mollahosseini, B. Hasani, and M. H. Mahoor, “AffectNet: A database for facial expression, valence, and arousal computing in the wild,” *IEEE Trans. Affect. Comput.*, vol. 10, no. 1, pp. 18–31, Jan. 2019.

[54] J.-H. Kim, B.-G. Kim, P. P. Roy, and D.-M. Jeong, “Efficient facial expression recognition algorithm based on hierarchical deep neural network structure,” *IEEE Access*, vol. 7, pp. 41273–41285, 2019.

[55] P. Dhankhar, “Resnet-50 and vgg-16 for recognizing facial emotions,” *Int. J. Innov. Eng. Technol. (IJIET)*, vol. 13, no. 4, pp. 126–130, 2019.

[56] B. Hasani and M. H. Mahoor, "Facial expression recognition using enhanced deep 3D convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jul. 2017, pp. 30–40.

[57] A. Majumder, L. Behera, and V. K. Subramanian, "Automatic facial expression recognition system using deep network-based data fusion," *IEEE Trans. Cybern.*, vol. 48, no. 1, pp. 103–114, Jan. 2018.

[58] D. H. Kim, W. J. Baddar, J. Jang, and Y. M. Ro, "Multi-objective based spatio-temporal feature representation learning robust to expression intensity variations for facial expression recognition," *IEEE Trans. Affect. Comput.*, vol. 10, no. 2, pp. 223–236, Apr. 2019.

[59] G. Pons and D. Masip, "Supervised committee of convolutional neural networks in automated facial expression analysis," *IEEE Trans. Affect. Comput.*, vol. 9, no. 3, pp. 343–350, Jul. 2018.

[60] M. G. Villanueva and S. Ramirez Zavala, "Deep neural network architecture: Application for facial expression recognition," *IEEE Latin Amer. Trans.*, vol. 18, no. 07, pp. 1311–1319, Jul. 2020.

[61] Z. Meng, P. Liu, J. Cai, S. Han, and Y. Tong, "Identity-aware convolutional neural network for facial expression recognition," in *Proc. 12th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG)*, May 2017, pp. 558–565.

[62] X. Liu, B. V. K. V. Kumar, J. You, and P. Jia, "Adaptive deep metric learning for identity-aware facial expression recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jul. 2017, pp. 20–29.

[63] M. Alam, L. S. Vidyaratne, and K. M. Iftekharuddin, "Sparse simultaneous recurrent deep learning for robust facial expression recognition," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 10, pp. 4905–4916, Oct. 2018.

[64] C. F. Benitez-Quiroz, R. Srinivasan, and A. M. Martinez, "Discriminant functional learning of color features for the recognition of facial action units and their intensities," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 12, pp. 2835–2845, Dec. 2019.

[65] S. Xie and H. Hu, "Facial expression recognition using hierarchical features with deep comprehensive multipatches aggregation convolutional neural networks," *IEEE Trans. Multimedia*, vol. 21, no. 1, pp. 211–220, Jan. 2019.

[66] T. Zhang, Z.-L. Liu, X.-H. Wang, X.-F. Xing, C. L. P. Chen, and E. Chen, "Facial expression recognition via broad learning system," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. (SMC)*, Oct. 2018, pp. 1898–1902.

[67] D. H. Nguyen, S. Kim, G.-S. Lee, H.-J. Yang, I.-S. Na, and S. H. Kim, "Facial expression recognition using a temporal ensemble of multi-level convolutional neural networks," *IEEE Trans. Affect. Comput.*, early access, Oct. 10, 2019, doi: 10.1109/TAFFC.2019.2946540.

[68] S. Minaee and A. Abdolrashidi, "Deep-emotion: Facial expression recognition using attentional convolutional network," 2019, *arXiv:1902.01019*. [Online]. Available: http://arxiv.org/abs/1902.01019

[69] Y. Liu, X. Zhang, Y. Lin, and H. Wang, "Facial expression recognition via deep action units graph network based on psychological mechanism," *IEEE Trans. Cognit. Develop. Syst.*, vol. 12, no. 2, pp. 311–322, Jun. 2020.

[70] H. Roy and D. Bhattacharjee, "Local-Gravity-Face (LG-face) for illumination-invariant and heterogeneous face recognition," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 7, pp. 1412–1424, Jul. 2016.

[71] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proc. 14th Int. Conf. Artif. Intell. Statist.*, 2011, pp. 315–323.

[72] Y. Bengio, "Practical recommendations for gradient-based training of deep architectures," in *Neural Networks: Tricks Trade*. Springer, 2012, pp. 437–478.

[73] A. Ross and A. Jain, "Information fusion in biometrics," *Pattern Recognit. Lett.*, vol. 24, no. 13, pp. 2115–2125, 2003.

Karnati Mohan received the B.Tech. degree in computer science and engineering and the M.Tech. degree from Jawaharlal Technological University, Hyderabad, India, in 2016 and 2020, respectively. He is currently pursuing the Ph.D. degree with the Computer Science Department, PDPM Indian Institute of Information Technology, Design and Manufacturing Jabalpur, Jabalpur, India.

His research interests include machine learning, affect recognition, deep learning, and image processing.

Mr. Mohan has served as a Program Committee Member for the ISAC Conference. He has also served as a Reviewer for the IEEE ACCESS journal.

Ayan Seal (Senior Member, IEEE) received the Ph.D. degree in engineering from Jadavpur University, West Bengal, India, in 2014.

He is currently an Assistant Professor with the Computer Science and Engineering Department, PDPM Indian Institute of Information Technology, Design and Manufacturing Jabalpur, Jabalpur, Madhya Pradesh, India. He is also associated with the Center for Basic and Applied Science, Faculty of Informatics and Management, University of Hradec Kralove, Hradec Kralove, Czech Republic. He has visited the Universidad Politecnica de Madrid, Madrid, Spain, as a Visiting Research Scholar. He has authored or coauthored several journals, conferences, and book chapters in the area of biometric and medical image processing. His current research interests include image processing and pattern recognition.

Dr. Seal was a recipient of several awards. He has received Sir Visvesvaraya Young Faculty Research Fellowship from Media Lab Asia, Ministry of Electronics and Information Technology, Government of India.

Ondrej Krejcar received the Ph.D. degree in technical cybernetics from the Technical University of Ostrava, Ostrava, Czech Republic.

From 2016 to 2020, he was the Vice-Dean for Science and Research at the Faculty of Informatics and Management, University of Hradec Kralove (UHK), Czech Republic. He is a Full Professor in systems engineering and informatics at the Faculty of Informatics and Management, Center for Basic and Applied Research, UHK; and a Research Fellow at the Malaysia-Japan International Institute of Technology, University Technology Malaysia, Kuala Lumpur, Malaysia. He is currently the Vice-Rector for science and creative activities of UHK since June 2020, where he is currently the Director of the Center for Basic and Applied Research. His h-index is 18, with more than 1150 citations received in the Web of Science. In 2018, he was the 14th top peer-reviewer in Multidisciplinary in the World according to Publons and a Top Reviewer in the Global Peer Review Awards 2019 by Publons. His research interests include control systems, smart sensors, ubiquitous computing, manufacturing, wireless technology, portable devices, biomedicine, image segmentation and recognition, biometrics, technical cybernetics, and ubiquitous computing. His second area of interest is in biomedicine (image analysis), as well as biotelemetric system architecture (portable device architecture, wireless biosensors), and the development of applications for mobile devices with the use of remote or embedded biomedical sensors.

Dr. Krejcar is the Vice-leader and Management Committee Member of project COST CA17136 at WG4, since 2018. He has also been a Management Committee Member Substitute at project COST CA16226 since 2017. Since 2019, he has been the Chairman of the Program Committee of the KAPPA Program, Technological Agency of the Czech Republic, as a regulator of the EEA/Norwegian Financial Mechanism in the Czech Republic (2019–2024). Since 2020, he has been the Chairman of the Panel 1 (Computer, Physical, and Chemical Sciences) of the ZETA Program, Technological Agency of the Czech Republic. From 2014 to 2019, he has been the Deputy Chairman of the Panel 7 (Processing Industry, Robotics, and Electrical Engineering) of the Epsilon Program, Technological Agency of the Czech Republic. At UHK, he is a guarantee of the doctoral study program in applied informatics, where he is focusing on lecturing on Smart Approaches to the Development of Information Systems and Applications in Ubiquitous Computing Environments. He is currently on the editorial board of the *MDPI Sensors IF* journal (Q1/Q2 at JCR), and several other ESCI indexed journals.

Anis Yazidi (Senior Member, IEEE) received the M.Sc. and Ph.D. degrees from the University of Agder, Grimstad, Norway, in 2008 and 2012, respectively.

He was a Researcher with Teknova AS, Grimstad. From 2014 to 2019, he was an Associate Professor with the Department of Computer Science, Oslo Metropolitan University, Oslo, Norway, where he is currently a Full Professor, leading the research group in applied artificial intelligence. He is also a Professor II with the Norwegian University of Science and Technology (NTNU), Trondheim, Norway. His current research interests include machine learning, learning automata, stochastic optimization, and autonomous computing.