Journal of Ambient Intelligence and Humanized Computing (2021) 12:10581–10599
https://doi.org/10.1007/s12652-020-02866-3

ORIGINAL RESEARCH

Virtual facial expression recognition using deep CNN with ensemble learning

Venkata Rami Reddy Chirra[1,3] · Srinivasulu Reddy Uyyala[1,2] · Venkata Krishna Kishore Kolli[3]

Received: 22 May 2020 / Accepted: 16 December 2020 / Published online: 16 March 2021

Abstract
In the current era, virtual environments and virtual characters have become popular. In the near future, recognition of virtual facial expressions plays an important role in virtual assistants, online video games, security systems, entertainment, psychological study, video conferencing, virtual reality, and online classes. The objective of this work is to recognize the facial emotions of virtual characters. Facial expression recognition (FER) from virtual characters is a difficult task due to its intra-class variation and inter-class similarity. The performances of existing FER systems are limited in this aspect. To address these challenges, we designed and developed a multi-block deep convolutional neural networks (DCNN) model to recognize the facial emotions from virtual, stylized and human characters. In multi-block DCNN, we defined four blocks with various computational elements to extract the discriminative features from facial images. To increase stability and to make better predictions two more models were proposed using ensemble learning which are bagging ensemble with SVM (DCNN-SVM), and the ensemble of three different classifiers with a voting technique (DCNN-VC). Image data augmentation was applied to expand the dataset to improve model performance and generalization. The accuracy of the proposed DCNN model was studied by tuning hyperparameters. Performances of the three proposed models were examined in contrast with pre-trained models such as VGGNet-19, ResNet50 with a voting technique for emotion recognition. The proposed models are evaluated and achieved the best accuracy when compared with other models on five publicly available facial emotion datasets that include UIBVFED, FERG, CK+, JAFFE, and TFEID.

Keywords Virtual facial expression recognition · DCNN · Intra-class variation · Inter-class similarity · Majority voting

1 Introduction

Emotions are mainly expressed through hand, voice, body gestures, and facial expressions. Facial expressions are being used to convey emotions during interactions. Mehrabian (2007) stated that 55% of emotions are conveyed via facial expressions only. Ekman et al. (1972) identified six expressions, which are basic universal emotional expressions. A few decades ago Ekman et al. (1978) had done a systematic study on facial emotion analysis and identified six basic expressions that include anger, joy, sad, disgust, surprise, and fear. The human face exhibits relevant information cues to express emotional state or behavior. Humans can identify a person's emotions accurately by observing the human face in a few seconds. Facial emotion recognition is used for human–computer interaction (Bartlett et al. 2003), patient care, and student awareness estimation (Whitehill et al. 2014),

multimedia, emotion aware devices (Soleymani and Pantic 2013), surveillance (Wang et al. 2015), autism disorder patients (Cockburnet al. 2008), and driver safety (Reddy et al. 2019; Mahesh Babu et al. 2019).

The dataset used in this work for recognizing facial expressions from virtual characters is UIBVFED which is a challenging dataset due to its intra-class variation (Fig. 1) and inter-class similarity (Fig. 2). In inter-class similarity,

✉ Venkata Rami Reddy Chirra
chvrr58@gmail.com

Srinivasulu Reddy Uyyala
usreddy@nitt.edu

[1] Machine Learning and Data Analytics Lab, Department of Computer Applications, National Institute of Technology, Tiruchirappalli 620015, India

[2] Centre of Excellence in Artificial Intelligence, National Institute of Technology, Tiruchirappalli 620015, India

[3] Department of Computer Science and Engineering, VFSTR, Guntur 522213, India

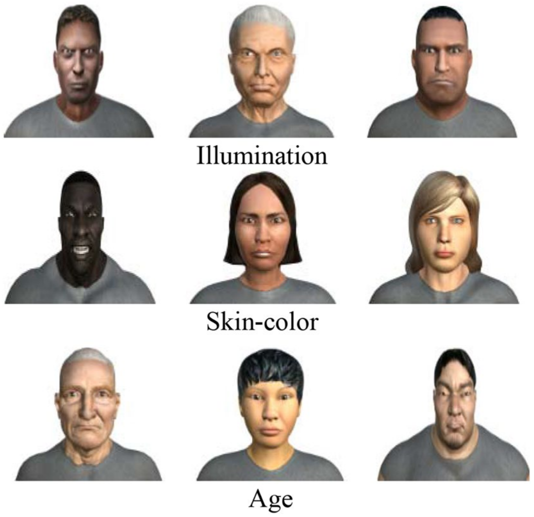

Fig. 1 Intra-class variations

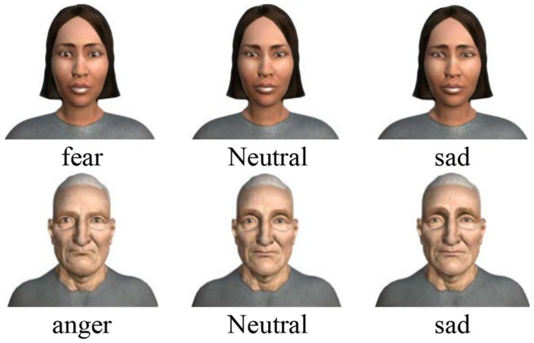

Fig. 2 Inter-class similarities

some of the images of different expression classes have the same similar appearances which make their discrimination difficult. In the intra-class variation, some of the images in the same expression class have different variations like illumination, age, and skin-color which make the model difficult to recognize the expression. Intra-class variations are intractable for facial expression recognition. The performance of FER degrades in virtual environments due to high intra-class variations and high inter-class similarities introduced by subtle facial appearance changes, illumination variations, skin-color changes, and identity-related attributes, e.g., age gender, and race.

In the literature, many experiments were conducted on the datasets with small intra-class variations only. However, the requirement is hard to satisfy when we recognize the virtual facial expressions from virtual environments. Researchers have proposed various methodologies to solve the above-mentioned problems. However, the intra-class variation was not explicitly considered in many existing approaches on FER but they used the datasets having intra-class variations. Most of the existing methods (Mayya et al. 2016; Venkata Rami Reddy et al. 2019; Gogić et al. 2020) depend on engineered features that lack generalization ability to perform virtual characters expression recognition.

The recent approaches in computer vision, especially deep learning models have improved the performance of the facial emotion classification tasks. Convolution Neural Network (CNN) based models are very robust and performing well in facial expression classification tasks. In CNN, convolutional filter parameters are fine-tuned at each layer to attain high-level features to generalize and represent the desired features for recognizing the unseen images.

Lee et al. (2014) address the intra-class variation problem by generating the intra-class variation image for each expression by training images and differences between these images are the features for sparse representation. This method addresses the illumination variation. An intra-class variation reduced features were used in (Xie et al. 2018) to reduce the intra-class variation influence. This method didn't consider the effects of skin-color and age variations.

The performance of the existing FER systems was limited by inter-class similarity and intra-class variation. To address these issues, we propose a CNN based model for recognizing facial emotions from virtual characters. A multi-block deep CNN model was designed to extract the discriminative features from the virtual characters. The discriminative power of features can reduce the impact caused by intra-class variations and inter-class similarities to make the model robust in spite of variations. CNN models were used to obtain discriminative features of facial expression images, and these features are given as input to three classifiers (Support Vector Machine (SVM), Random Forest (RF), and Logistic Regression (LR)) for recognition. Based on the classifier used three models namely DCNN (softmax), DCNN-SVM (SVM with Bagging), and DCNN-VC (Voting technique) were being proposed.

The major contributions of this paper are as follows:

- Proposed a multi-block DCNN model to extract discriminative features to recognize the seven facial emotions of virtual characters.
- To the best of author's knowledge first model which recognizes the facial expressions from three kinds of characters that include virtual, stylized, and human was proposed.
- Image data augmentation was performed to expand the datasets for improving the performance and model generalization.
- Bagging ensemble with SVM (DCNN-SVM) and the DCNN ensemble of SVM, RF, and LR classifiers with majority voting technique (DCNN-VC) was proposed to make better predictions.

2 Related works

In the emotional analysis, facial emotion recognition and classification has been considered as a challenging task. In recent years, many authors have proposed and developed various deep learning and machine learning (ML) models for emotion recognition tasks. In most of the existing works, the intra-class variation was not explicitly considered but they have done the experimentation using the datasets having intra-class variations.

Ramireddy et al. (2013) proposed a fusion-based method for recognizing emotions using Gabor wavelets and DCT (Discrete Cosine Transform). In this work, a different type of feature was extracted using Gabor filters and DCT. The kernel principal component analysis was applied to extract features, reduce dimensions. The RBFNN (Radial basis function neural network) was applied to classify the expression images into six basic emotions. Experimentation was performed on the CK dataset and accuracy of 99% was obtained with limited training and testing samples. Pons et al. (2018) developed a framework for recognizing emotions using the Supervised Committee of CNNs. 72 CNNs with the same baseline architecture was used for feature extraction. The proposed work was evaluated on FER2013, MMI, and LFW datasets. Li et al. (2019) designed a CNN model for recognizing emotions using Attention Mechanism (ACNN). pACNN was applied on local facial patches whereas gACNN combined both patch-level and image-level features. Experimentation was performed on Affect Net and RAF-DB datasets and attained 85 and 58.75% accuracy respectively.

Xie et al. (2018) developed a model based on Deep Comprehensive Multi patches Aggregation CNNs. In this work, two branches of CNNs were used. One branch of CNN was used for extracting the local features from patches and the other branch of CNNs were used for obtaining the holistic features from the entire face sample and these features were combined to create a feature vector and given to the classifier for expression classification. Experimentation was performed on CK+, JAFFE datasets and attained 93.46, 94.75% accuracy respectively. Mayya et al. (2016) developed a new method for recognizing emotions using DCNNs. In their work, the first face was detected from dataset images and given those frontal face images to CNN for extracting features. SVM with a grid search was used for classification. Proposed models were evaluated on CK+, JAFEE and achieved 97, 98.12% accuracy respectively. Rami Reddy et al. (2019) proposed different methods of FER. In this work, local and global features were extracted using Gabor wavelets and HWT respectively. Non-linear PCA (NLPCA) was used for reducing the feature dimension. Weighted and Concatenated fusion techniques were applied to combine those two types of features. SVM was used for classification. Experimentation was performed on the CK+ and achieved 98% accuracy.

An RGB–D Microsoft Kinect camera was adapted to record facial expressions of students in the classroom for recognizing the emotions in (Purnama and Sari 2019). The Adaptive-Network-Based Fuzzy Inference System machine learning algorithm was used to train and classify the expressions. A combination of EURECOM and the Cohn-Kanade dataset was used for training the algorithm. In biometric recognition, the accuracy of the system depends on the quality of input images. The impact of the image quality on accuracy was discussed in (Alsmirat et al.2019). In this study, the system provides good accuracy until the 30–40% compression ratio of raw images and higher ratio negatively impacted the accuracy of the system. Li et al. (2019) introduced deep overlap and weighted filter concepts in the macro pixel approach to extract the richer features from macro pixels. The experiment result shows that the proposed approach achieved better accuracy when compared with the original macro pixel approaches.

A CNN features are merged with the SIFT features to increase the FER accuracy by Connie et al. (2017). This work was tested on FER2013 and CK+ datasets and attained 73.4% and 99.1% accuracy respectively. A CNN feature-based FER was developed by Gonzalez-Lozoya et al. (2020) in which facial features were extracted using CNN. Model generalization was improved by mixing different dataset images. Ozcan et al. (2020) use transfer learning with hyperparameter optimization for FER on static images. They utilized hyperparameter optimization to increase the accuracy of the model. This work was experimented on JAFFE and ERUFER datasets. Gogić et al. (2020) developed a joint optimization framework for FER using local binary features and shallow networks with improved execution time. The hybrid deep learning model was developed by Garima and Hemraj (2020) for facial expression recognition. Here, the primary emotion being sad or joy was identified by one CNN and secondary CNN recognizes the secondary emotion of the image. This work was tested on FER2013, and JAFFE datasets.

All the mentioned works produced good results on human-based datasets but these models are sensitive to the illumination and specific poses present in that dataset because these models are evaluated on a single kind of dataset. These models have a lack of generalization ability to perform virtual and stylized character's expression recognition. The performance of the existing FER systems was limited by above said two problems. Most of the existing methods for facial expression recognition used a single classifier hence, models suffered from bias and variance which affects the performance of the model. Henceforth, there is a wide scope for a new model that recognizes the emotions of virtual and stylized characters with better accuracy. Therefore, we developed a new model that recognizes the emotions from three kinds of characters which include virtual, stylized, and humans. To make better predictions ensemble learning techniques were used during classification.

3 The proposed models

DCNN, DCNN-SVM, and DCNN-VC models were proposed for facial expression recognition from three kinds of characters namely virtual, stylized characters, and humans. Initially, the face was detected and cropped followed by data augmentation to increase the number of image samples that are given as input to DCNN for extracting features. Finally, these features are fed into classifiers (SVM, RF, and LR) for recognition and the process is described in detail below.

3.1 Face detection

Viola-Jones algorithm (Viola et al. 2004) was applied to recognize the faces and those detected faces are cropped as shown in Fig. 3. The same algorithm has been adopted for detecting the face because of its low false-positive rate. The working of the algorithm is as follows: The image is subdivided into a grid of rectangles. The Haar feature selection uses the rectangles to detect features using windows in the image. The AdaBoost algorithm creates a strong classifier by integrating a set of weak learners. A weak learner uses Haar-like features to find the face in the sub-region of an image. Each classifier looks at the sub-region and if it finds a face then that region is forwarded to the next classifier otherwise that sub-region is rejected and repeated until the last weak classifier is reached. If all classifiers detected face, then the strong classifier approves the sub-region as a human face.

3.2 Data augmentation

The UIBVFED, CK+, JAFFE, and TFEID datasets have limited samples and there is a possibility of under-fitting as deep learning models required more samples for training. Image data augmentation was applied to expand the dataset to improve model performance and generalization. The data augmentation techniques such as flipping, rotation, and shifting were applied to expand the number of samples in UIBVFED, JAFFE, CK+, and TFEID datasets. In the proposed model, horizontal flipping, rotation range with 20, and shifting with 0.2 was used. The sample image after data augmentation was shown in Fig. 4.

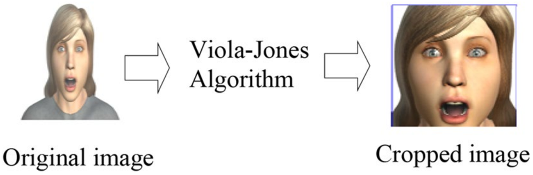

Fig. 3 Pre-processed face detected image

3.3 DCNN

The multi-block DCNN was proposed for FER and the architecture was depicted in Fig. 5. It consists of four blocks for extracting the features from facial images. Each block contains two convolution layers, ELU (exponential linear unit), batch normalization, a max-pooling layer, and dropout. Kernel and bias regularizer with L2 regularization was used in the first convolution layer of each block to minimize the over fitting by penalizing the weight and bias values. Kernel initializer was used to initialize the weights in the first convolution layer of the first block. The Batch normalization was applied to improve the performance, stability and speed up learning after each convolution layer. The dropout was applied to prevent the developed model from over fitting. Each block generates a feature map that is given as input to the next block. The first block extracts the low-level features like dots, lines, and curves. The second and third blocks extract middle-level features whereas the last block generates high-level features. The feature map of the last block is flattened and forwarded to the fully connected (fc) layer that is given as input to a softmax layer. The softmax classifies the facial expression images into corresponding emotion classes. The convolution, ELU, max-pooling, and softmax are the main computational elements of our proposed multi-block DCNN model. The following subsections describe the functionality of these elements.

3.3.1 Convolution layer

The convolution layer (Teow 2017) was applied to obtain the pixel-wise visual features from an input face image. In this layer, the weights of the kernels are automatically adjusted using the back propagation to learn the input expression features. These features are forwarded to the next layer to process using the corresponding operation. In the proposed DCNN model, two convolution layers were used in each block. The convolution is a dot (.) product between the face image and kernel.

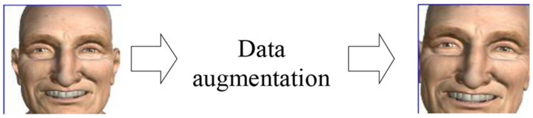

Fig. 4 Pre-processed data augmented image

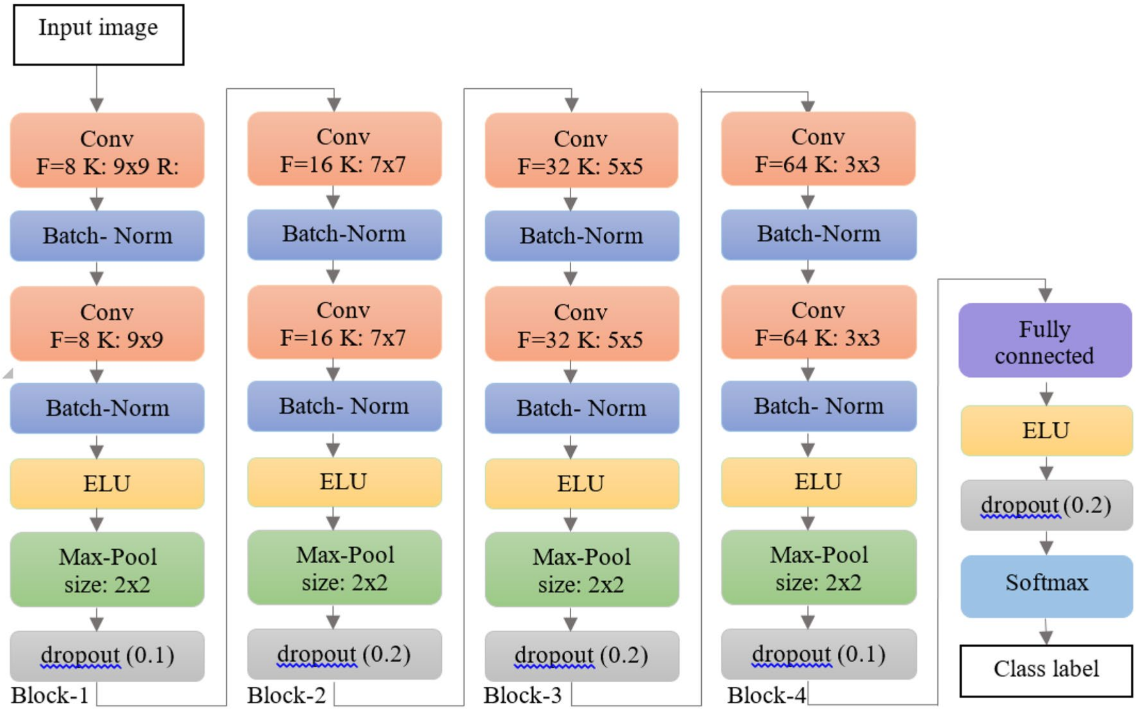

Fig. 5 Architecture of DCNN, R: Regularizer, F: Number of filters

$$f_c = \sum_m \sum_n I(m,n)W(i-m,j-n) \tag{1}$$

Here, *fc* is a convolution feature map, *I* represent a facial input image, *and W* represents a convolution kernel.

3.3.2 ELU layer

The ELU activation function was applied to speed up the learning and improve the model generalization. In the proposed DCNN, we use elu before max-pooling in each block. ReLU introduces a dead ReLU problem where network components are not updated frequently with new value so ELU activation function was used. The ELU activation function is given in Eq. 2. In Eq. 2, if *x* value is greater than zero then the result is *x* otherwise resultant value is slightly below zero and which depends on α. Here ELU produces the negative value which helps the network nudge biases and weights in the correct directions and produces activations instead of zeros during gradient calculation. The output of ELU is a feature map f_e.

$$f_e = \begin{cases} x, & if x > 0 \\ \alpha(e^x - 1), & if x < 0 \end{cases} \tag{2}$$

Here, f_e is an ELU feature map, *x* represents the input and α represents the nonlinearity parameter.

3.3.3 Pooling layer

The feature map f_e generated by the ELU is forwarded to the pooling layer which subsamples f_e for the dimensionality reduction. In this work, a 2×2 max-pooling without stride and zero padding is applied for downsampling. In the max-pooling layer, pooling operation outputs the maximum value of the input within the kernel area at a given position which is given by the Eq. (3).

$$f_p = max_{i,j=1}^{h,w} x_{i,j} \tag{3}$$

where, f_p is a pooled feature map that is generated by the max-pooling operation.

3.3.4 Softmax layer

In a multiclass classification, softmax returns a probability distribution over the target classes. The probability distribution contains the range of real values between 0 and 1. It assigns probabilities to each class in a multiclass problem. The sum of those decimal probabilities is equal to 1. Mathematically the softmax function is given by Eq. (4).

$$\text{Softmax}(x_i) = \frac{e^{x_i}}{\sum_{j=1}^{n} e^{x_j}} \tag{4}$$

where n is the number of target classes. Here, DCNN was trained to classify the facial expression images from 0 to 6 classes. From Eq. 4 the expression image with the highest probability is recognized as the correct output.

3.4 DCNN-SVM

To reduce the variance and increase the accuracy, a second model was proposed using the bagging ensemble technique. In this model bagging ensemble with SVM (Kim et al. 2002) was used as the base classifier for facial expression classification.

In DCNN-SVM, the DCNN model was applied to obtain the discriminate features from face images then these features were given as input to ensemble bagging with SVM as a base classifier for facial expression classification. In this model, three SVMs were trained independently on deep features using a bootstrap technique which are combined using a majority voting technique. In majority voting, the class which receives the highest number of votes can be predicted as a final class. Figure 6 shows the architecture of the DCNN-SVM model. The bagging algorithm (Lango and Stefanowski 2017) is given in Table 1.

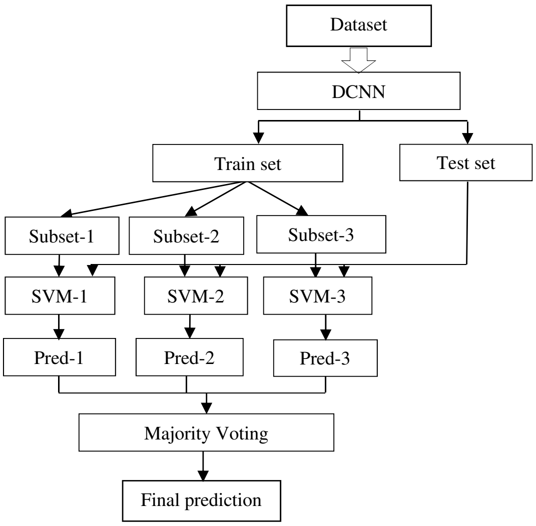

Fig. 6 Architecture of DCNN-SVM

Table 1 Bagging algorithm

Algorithm Bagging

```
Input: $T$: Training set of size $N$, $K$: No of bootstrap samples, L: Base learning algorithm
Output: $C^*$ bagging ensemble with K base classifiers
Training Phase:
1: $for(i=1; i<=K; i++)\{$
2: $S_i$=Bootstrap sample from $T$;
3: Generate classifier $C_i=L(S_i)$
4: $\}$
Predict a class label for input $x$:
5: $C^*(x) = \arg max_y \sum_{j=1}^{K}[C_j(x) = y]$
```

DCNN-SVM procedure

1. Dataset is preprocessed using face detection and data augmentation techniques.
2. The DCNN model was used to obtain the discriminative features from input images.
3. These discriminative features are separated into training and testing datasets.
4. The bootstrapping technique randomly generates K replicated subsets from the training set.
5. The SVM base classifier is trained on each subset.
6. Deep features of the test set given as input to each trained SVM classifier that predicts the class label.
7. The majority voting technique selects the class predicted by the most classifiers as a final class.

3.4.1 SVM

In computer vision, SVM is the most suitable algorithm for image classification. It exhibits good performance in facial emotion analysis. It is the best suitable algorithm for binary classification and with the help of different kernels, it is also used for multi-classification. During experimentation, we use a one-vs-one approach of SVM with a linear kernel for facial expression recognition. In a one-vs-one approach, SVM constructs C(C-1)/2 number of different binary classifiers to classify C-classes of input data.. The SVM uses the maximum margin principle to classify the data points.

3.5 DCNN-VC

In this paper, a model using ensemble learning was proposed to increase stability and to make better predictions. In DCNN-VC, the DCNN model was applied to obtain the discriminate features from face images. These extracted features were forwarded to the ensemble of classifiers with a voting technique for emotion recognition. Voting combines the predictions of various ML algorithms. The Voting technique is not a classifier but a wrapper for a set of machine learning algorithms that are trained and tested in parallel to exploit the different peculiarities of each algorithm. In this work, we have trained deep features using SVM, RF, and LR. Different combinations of machine learning algorithms were trained but finally chosen this combination as it provides better accuracy when compared with the other combinations. Majority Vote based ensemble learning method increases the accuracy by combining the advantages of each classifier. The majority voting technique selects the class predicted by most classifiers as a final class. The architecture of the DCNN-VC is depicted Fig. 7. The majority voting algorithm is given in Table 2.

DCNN-VC procedure

1. Dataset is preprocessed using face detection and data augmentation techniques.
2. The DCNN model was used to obtain the discriminative features from input images.
3. These discriminative features are separated into training and testing datasets.
4. SVM, RF, and LR classifiers are trained in parallel on the training set.
5. Deep features of the test set given as input to each trained classifier that predicts the class label.
6. The majority voting technique selects the class predicted by the most classifiers as a final class.

Table 2 Majority Voting algorithm

```
Algorithm Voting
Input: $T$: Train set, L: learning algorithm
Output: final classifier: $C^*$
1: Step-1: Train N different number of classifiers
2: for i = 1 to N do
3: Generate classifier $C_i = L_i(T)$
4: end for
Predict a class label for input $x$:
5:$C^*(x) = \arg max_y \sum_{j=1}^{N} \left(C_j(x) = y\right)$
```

3.6 Random Forest classifier

RF (Pu et al. 2015) is the fastest, robust algorithm and is mainly used for classification tasks. RF itself is an ensemble method with various decision trees. Prediction from each of the decision trees is combined by using voting. It overcomes the overfitting problem by combining different decision tree predictions. It works well for unbalanced data. RF classifier is stated in Eq. 5.

$$F(x) = argmax \sum_{i=1}^{N} I\left(f_i(x) = Y\right) \tag{5}$$

where $F(x)$ is the majority voting technique, N specify the number of decision trees, f_i is the decision function of the ith decision tree, Y represents the class label, $\boldsymbol{I}\left(\boldsymbol{f}_i(\boldsymbol{x}) = \boldsymbol{Y}\right)$ indicates x belongs to class Y.

3.7 Logistic regression classifier

Logistic regression is ML algorithm that is used to solve different classification problems. Its algorithm is based on probability. Mainly it is used for binary classification but also used to solve multi-classification problems using one-vs.-rest scheme. It uses the sigmoid function to map predicted values to probabilities.

3.8 FER using transfer learning

Transfer learning approaches are used for emotion classification in this work. In transfer learning, pre-trained models are used instead of layered architecture for learning the complex features. ResNet50, VGG19 pre-trained models were used to obtain required features from face samples. These features are forwarded to ensemble classifiers with voting for emotion classification. The two methods namely Resnet50-VC and VGG19-VC are used to train on UIBVFED, FERG, CK+, JAFFE, and TFEID datasets.

3.9 ResNet50-VC

ResNet (He et al. 2016) (Deep residual network) is a deeply layered architecture. The vanishing gradient problem could not occur in ResNet due to its skip connections feature. So, the main idea behind the ResNet is introducing skip connections that skip one or more layers as shown in Fig. 8. If any of the layers are not useful during training, then skip connections feature skip those layers. This helps faster training and tuning the parameters effectively. The output G(i) can be defined by Eq. 6

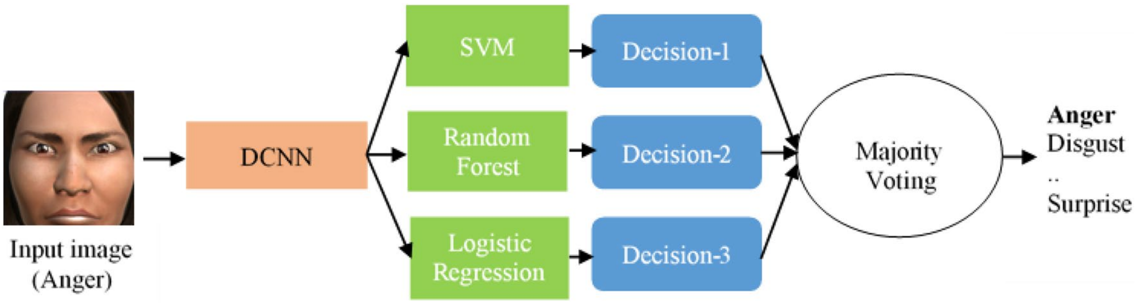

Fig. 7 Architecture of DCNN-VC

$$G(i) = F(i) + i \tag{6}$$

Here F(i) represents stacked layers and i represent identity.

ResNet50 model has 50 layers, which are used to obtain the complex features from the training dataset. These features are fed into the ensemble voting classifier for classification. Figure 9 shows the structure of ResNet50-VC. The ResNet50 has a convolution layer with 64 kernels of size 7×7, max-pooling of size 3×3 and stride 2, sixteen residual blocks with common sizes 1×1 and 3×3 and number of kernels are 64, 128, 256, 512, 1024 and 2048, average pooling of size 7×7 with stride 7. In this proposed architecture, 2048 features were extracted in the last layer i.e. average pooling layer. Finally, a feature vector of size Nx2048 is generated; where N represents the number of images. This feature vector is fed into an ensemble of classifiers with a majority voting technique for emotion recognition. The architecture of VGG19-VC is shown in Fig. 9.

3.10 VGG-19 with voting classifier

Visual Geometry Group (VGG) Net (Simonyan et al. 2014) have different variations that include VGG-11, VGG-13, VGG-16, and VGG-19. VGG-19 pre-trained model is used in this work for emotion classification. The input image with a size of 224×224 is given as input to this model. The number of parameters required for representation in VGG-19 is reduced by using very small 3×3 convolutions. The VGG-19 model with 19 layers was used for obtaining the features from facial emotion samples. The VGG-19 has 5 blocks with 3×3 filters in the convolution layer, 2×2 max pooling, two fully-connected layers, and a softmax layer. The first block consists of two convolutions each with 64 kernels, 2nd block also having two convolutions each with 128 kernels, and 3rd, 4th and 5th blocks contain 4 convolutions each with 256, 512, 512 kernels respectively. 4096 features are extracted in the last layer i.e. fully connected layer 2. Finally, this feature vector is fed into an ensemble of classifiers with a majority voting technique for facial emotion recognition. The architecture of VGG19-VC is shown in Fig. 10.

4 Experimental results

The proposed models were implemented on an Intel Core i5 system with 8 GB RAM and ASUS GeForce GTX 1060 Ti 3 GB graphics. This section covers the detailed discussion about the experiment analysis of our models on five benchmark FER datasets. Table 3 presents information about datasets and the sample images were shown in Fig. 11.

4.1 Proposed DCNN model hyperparameters

Various hyperparameters are tuned to improve the performance of the proposed DCNN model. L2 kernel regularizer and bias regularizer are used to reduce the overfitting. It also uses the he_uniform kernel initializer for initializing the weights. Batch normalization was applied for improving the performance, stability, and speed up the learning. The ELU activation function was also applied to speed up learning and improve the generalization.

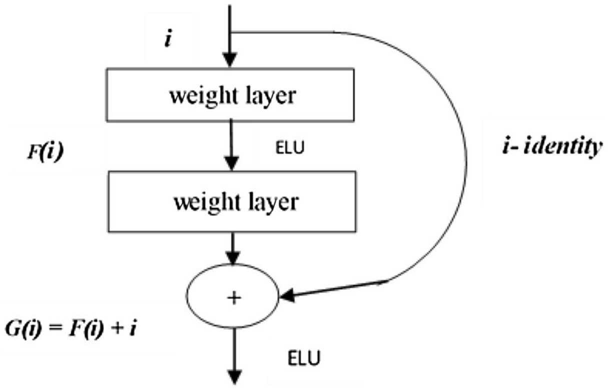

Fig. 8 Residual block

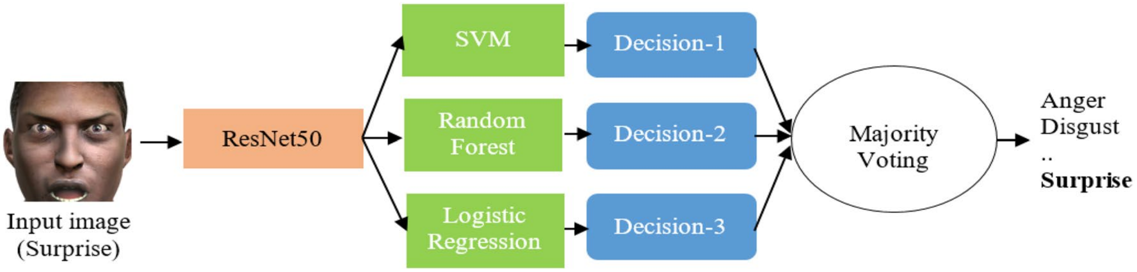

Fig. 9 Architecture of ResNet50-VC

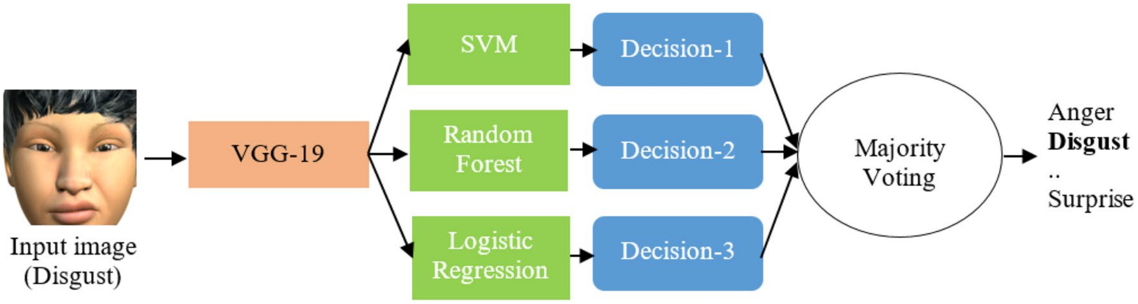

Fig. 10 Architecture of VGG19-VC

Table 3 Dataset description

S.No	Datasets	No of subjects	No of samples	After data augmentation
1	UIBVFED (Oliver MM et al. 2020)	20	640	7542
2	FERG (Aneja et al. 2016)	6	55,767	–
3	CK+(Cohn et al. 2010)	123	593	2100
4	JAFFE (Michael J et al. 1998)	10	213	2380
5	TFEID (Chen L-F and Yen Y-S 2007)	41	283	2450

4.1.1 Learning rate selection

The learning rate determines the speed at which the weights of the model changes and is used for minimizing the cost function of the network. If the learning rate is high, training may not be converging as a result cost function may increase. The training is reliable and the loss function may decrease when the learning rate is low but the model takes time for optimization. For minimizing cost function and improving the accuracy of the model an optimal value was set for the learning rate. The DCNN model was trained with different learning rates (0.01, 0.001, 0.0001, and 0. 00,001) for five datasets. The accuracy of the model on various datasets with different learning rates was evaluated. From Fig. 12 it was observed that DCNN achieved the best accuracy when the learning rate is 0.0001.

4.1.2 The mini-batch size selection

Mini-batch size determines the number of images processed before the model parameters are updated. If the mini-batch size is large it needs more memory and the model runs the longest period with constant weights which affect the performance of the model. So, the optimal value for the batch size needs to be selected for improving the performance of the system. The proposed DCNN was examined with the different mini-batch sizes of 4, 8, 16, and 32 for selecting the best suitable batch size. The performance of the model with various batch sizes on five datasets was compared as shown in Fig. 13. The proposed DCNN executed up to 15 epochs with a learning rate of 0.0001 The mini-batch size of 4 provides better accuracy for CK+, JAFFE, TFEID, and 16 for the UIBVFED dataset. We have used a batch size of 64 for the model on the FERG as it has a large number of samples. The proposed model achieved 99.97% accuracy on FERG with batch size 64.

4.1.3 Optimizer selection

The purpose of the optimizer in deep learning is to update bias and weights parameters to reduce the cost or loss function. The choice of the best optimizer for the model based on the problem produces better results at a faster rate by updating the weight and bias values of the model. The proposed model was evaluated with various optimizers like SGD (Stochastic Gradient Descent), Adam, RMSprop, and Adagrad. The performance of the model with various optimizers on five datasets after 15 epochs and learning rate 0.0001 was shown in Fig. 14. The accuracy of the DCNN model was improved with Adam optimizer as compared with other optimizers.

4.1.4 Number of epochs selection

In each epoch, updating weights of the network with all input images in the dataset are considered during each iteration of model learning. The optimum value for the number of epochs depends on the dataset size, depth of the model,

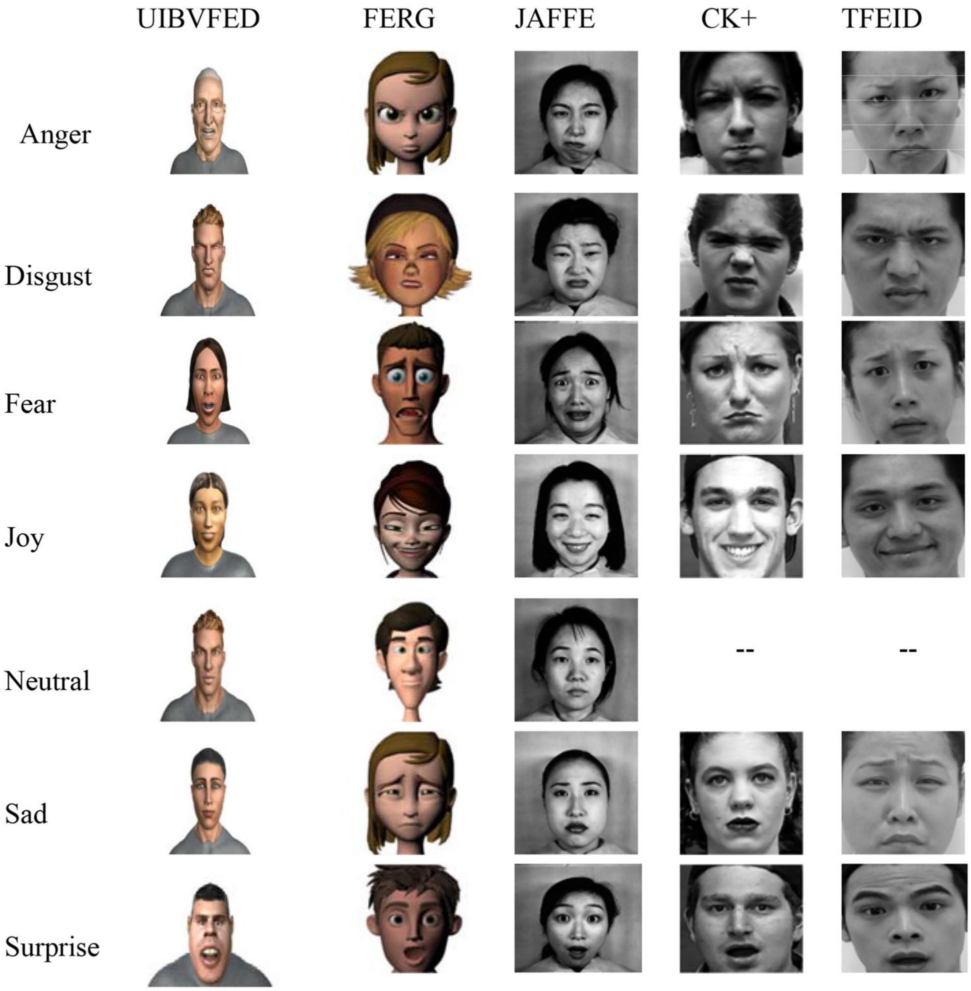

Fig. 11 Sample images of various datasets

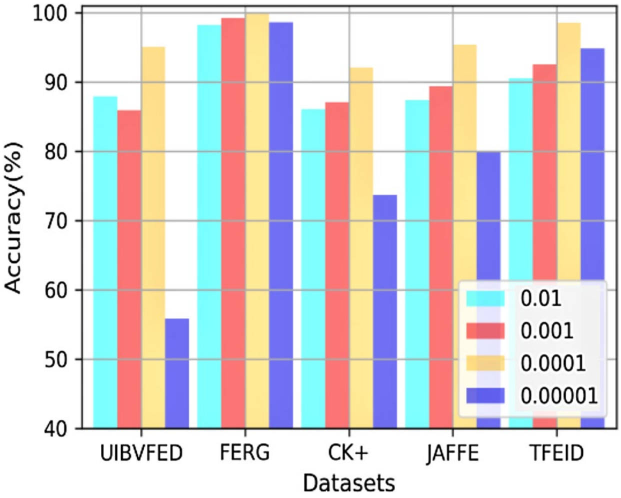

Fig. 12 Comparison of classification accuracy based on various learning rates

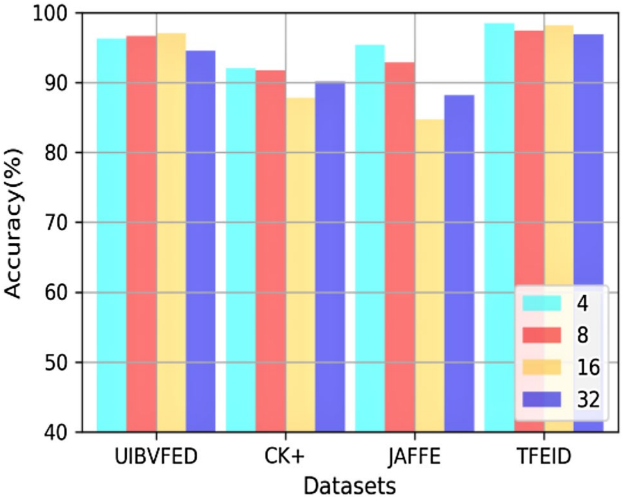

Fig. 13 Comparison of classification accuracy based on mini-batch size

learning rate, and optimizers. In this work, we have chosen the number of epochs based on the high facial expression recognition rate. The proposed DCNN was trained up to 100 epochs on five datasets. The model classification accuracy with various epochs was tested and depicted in Fig. 15. It clearly shows that classification accuracy increases up to 100 epochs Highest recognition rate was observed for all the datasets when the epoch was set to 100.

4.2 Overall recognition accuracy of proposed models

The proposed models were evaluated using accuracy (Eq. 7), recall (Eq. 8), precision (Eq. 9), F_1-score (Eq. 10), confusion matrix, precision-recall curve, and ROC curves.

$$Accuracy = \frac{1}{k}\sum_{i=1}^{k}\frac{TP_i + TN_i}{TP_i + TN_i + FP_i + FN_i} \tag{7}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{8}$$

$$\text{Precision} = \frac{TP}{TP + FP} \tag{9}$$

$$F_1 = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{10}$$

where k is the number of classes. TP represents true positives, TN represents true negatives, FN represents false negatives and FP represent false positive.

The UIBVFED and FERG datasets are more challenging due to its intra-class variation, inter-class similarity, and imbalance nature of emotion classes. From each of the datasets, we used 55% of the image samples for training the model, 15% of the image samples for validation, and 30% image samples for testing. The performance of the proposed models on five datasets was reported in Table 4. DCNN method provides the highest recognition rate for large datasets that include UIBVFED and FERG. The DCNN-VC model produces the highest recognition rate for small datasets that include CK+, JAFFE, and TFEID.

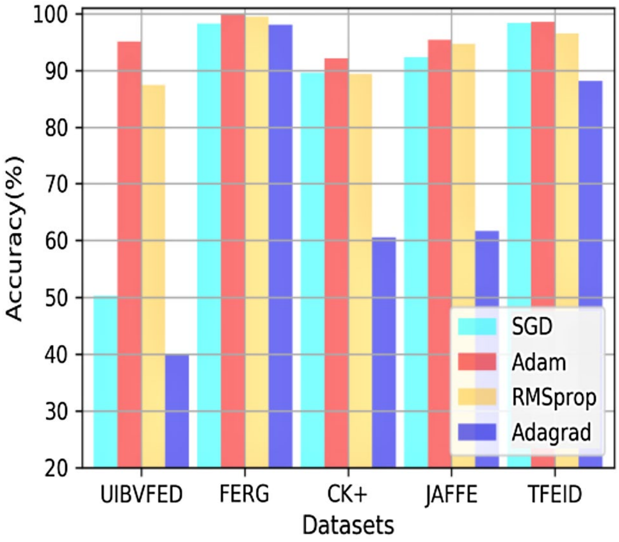

Fig. 14 Comparison of classification accuracy based on optimizers

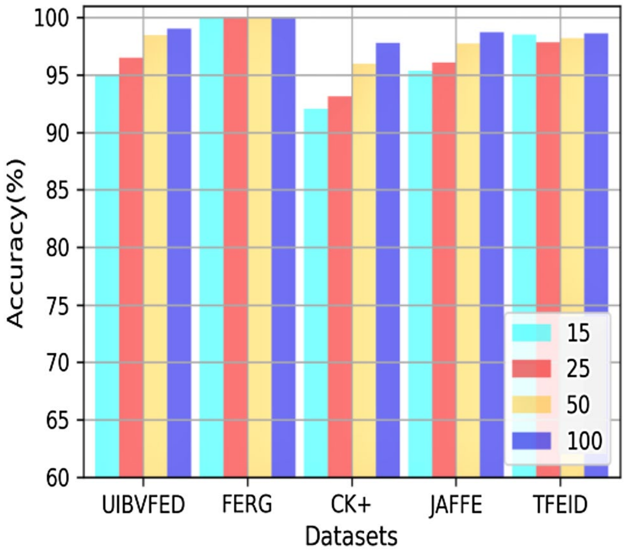

Fig. 15 Comparison of classification accuracy based on the number of epochs

The recognition rate of each expression of the DCNN model on the five datasets was presented in Table 5. It can be noted from Table 5 that the DCNN model performs best when recognizing all expressions except fear. The recognition rate of each facial expression of the DCNN-SVM model on the five datasets was presented in Table 6. It can be noted from Table 6 that the DCNN-SVM model performs best when recognizing anger, neutral, and surprise expressions. The recognition rate of each facial expression of the DCNN-VC model on the five datasets was presented in Table 7. From Table 7, it can be noted that the DCNN-VC model performs better when recognizing all expressions except fear and sad.

From Table 8, it can be seen that Precision, Recall, and F_1-score have high scores for each facial expression that

Table 4 Overall accuracy of proposed methods on five datasets (%)

Proposed Model	UIBVFED	FERG	CK+	JAFFE	TFEID
DCNN	99.02	99.97	97.77	98.73	98.63
DCNN- SVM	98.54	99.95	98.09	99.29	99.04
DCNN- VC	98.85	99.96	99.04	99.57	99.31

Table 5 Recognition rate of each expression of DCNN model (%)

Expression	UIBVFED	FERG	CK+	JAFFE	TFEID
Anger	99.97	100	100	99	97
Disgust	100	99.9	96	100	100
Fear	97.44	100	96	97	94
Joy	98.75	100	100	100	100
Neutral	98.10	99.80	–	99	–
Sad	99.47	100	97	98	100
Surprise	100	100	98	98	100

Table 6 Recognition rate of each expression of DCNN-SVM model(%)

Expression	UIBVFED	FERG	CK+	JAFFE	TFEID
Anger	98	100	99	100	100
Disgust	99	99.90	99	97	100
Fear	97	99.90	99	99	97
Joy	99	100	96.84	100	100
Neutral	98	99.80	–	100	–
Sad	99	100	95	100	100
Surprise	100	99.90	99	99	100

Table 7 Recognition rate of each expression of DCNN-VC model(%)

Expression	UIBVFED	FERG	CK+	JAFFE	TFEID
Anger	99	100	99.09	100	100
Disgust	99	99.90	99.07	100	99
Fear	97	99.90	99.09	100	98
Joy	99	100	100	100	99
Neutral	99	99.80	–	100	–
Sad	99	100	97	98.06	100
Surprise	100	99.90	100	98.94	100

indicates that the proposed models perform best when recognizing each facial expression as a result of returning more true positive values. The precision-recall curves of the DCNN and DCNN-VC on specific datasets are depicted in Figs. 16, 17, 18, 19 and 20. Our DCNN model extracted more discriminative features. To prove our claim, ROC curves were plotted and calculated the AUC score for the proposed models which achieved the highest accuracy on five datasets. From Figs. 21, 22, 23, 24 and 25 and Table 9, it clearly shows that the proposed models accurately recognized each facial expression. Table 10 presents the confusion matrix of DCNN on UIBVFED. Few samples of fear are misclassified as neutral and sad because visually these three expressions are very much similar. Table 11 presents the confusion matrix of DCNN on FERG. Very few samples of fear and neutral are misclassified as sad. Table 12 presented the confusion matrix of DCNN-VC on CK+. Some samples of sad are misclassified as fear. Table 13 presented the confusion matrix of DCNN-VC on JAFFE. Few samples of sad are misclassified as disgust and joy. Table 14 presented the confusion matrix of DCNN-VC on TFEID. Very few samples of contempt and joy are misclassified with each other. Table 15 presents the overall accuracy of pre-trained models on five datasets. The proposed models produced better results when compared with the pre-trained models ResNet50 and VGG-19 with a voting technique.

4.3 Performance of proposed models on closed expressions (inter-similarity)

In general, humans recognize anger, disgust, joy, and surprise expressions accurately because these expressions have unique features that distinguished from other facial expressions. Sometimes humans fail to recognize the fear, neutral, and sad facial expressions because these expressions

Table 8 Statistical analysis of proposed models on UIBVFED

	DCNN			DCNN-SVM			DCNN-VC		
	Precision	Recall	F_1-score	Precision	Recall	F_1-score	Precision	Recall	F_1-score
Anger	100	100	100	98	98	98	98	99	98
Disgust	99	100	99	98	99	99	100	99	99
Fear	100	97	99	99	97	98	99	97	98
Joy	99	99	99	98	99	98	98	99	99
Neutral	99	98	99	98	98	98	98	99	98
Sad	97	99	98	100	99	99	99	99	99
surprise	100	100	100	100	100	100	100	100	100

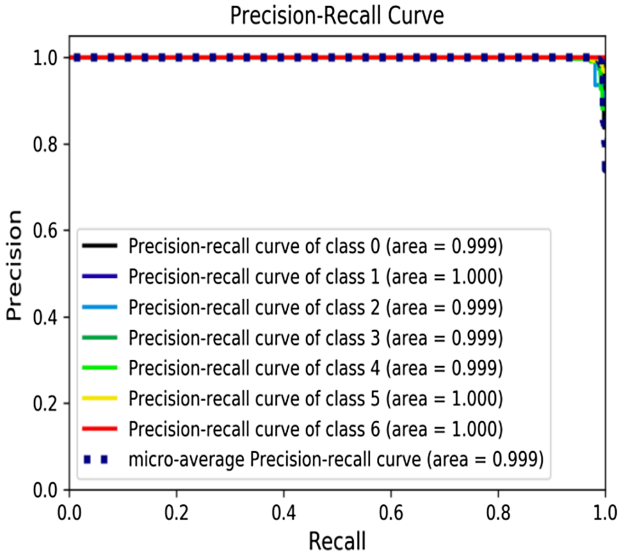

Fig. 16 Precision-Recall curve of DCNN on UIBVFED

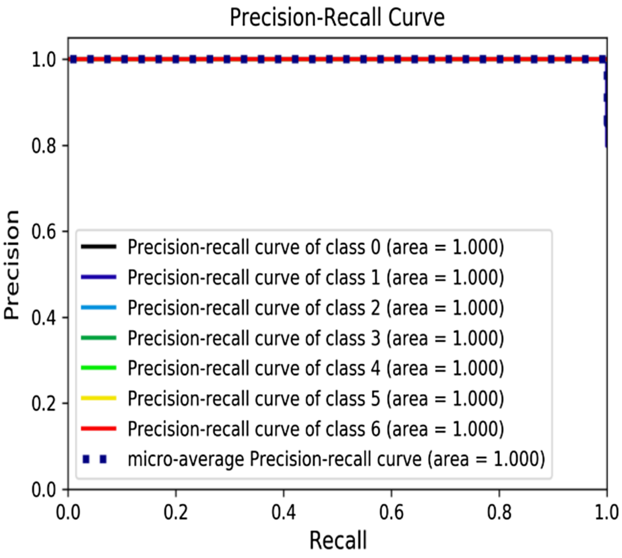

Fig. 17 Precision-Recall curve of DCNN on FERG

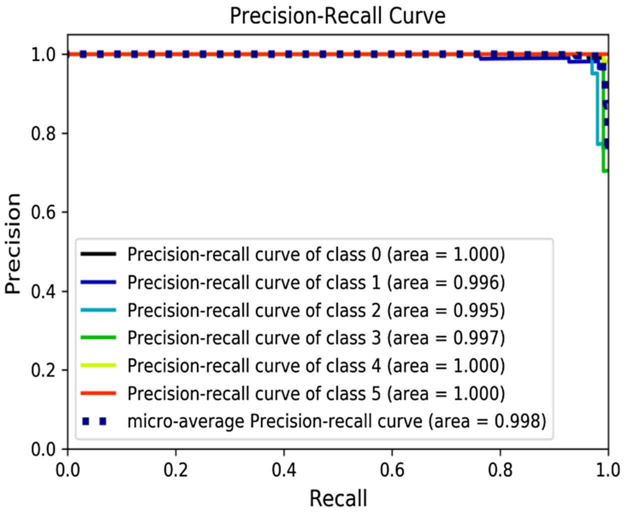

Fig. 18 Precision-Recall curve of DCNN-VC on CK+

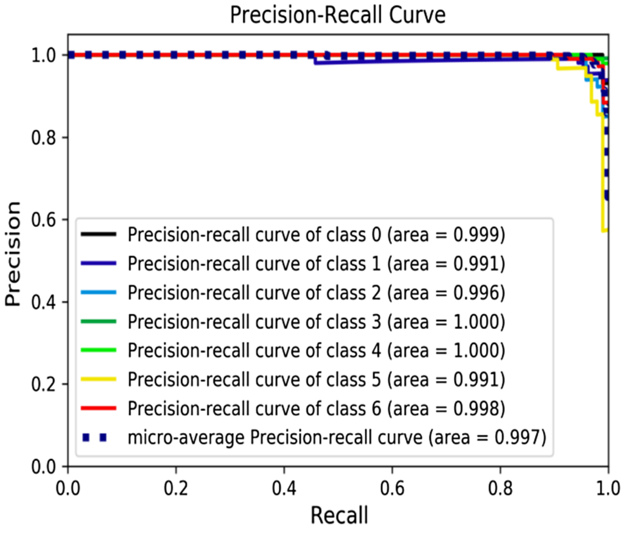

Fig. 19 Precision-Recall curve of DCNN-VC on JAFFE

have more similarities (inter-class similarity). The proposed models recognize fear, neutral, and sad facial expressions with more than 98% accuracy as our model can extract more prominent and discriminative features. The performance of the DCNN, DCNN-SVM and DCNN-VC for closed facial expressions is shown in Figs. 26, 27 and 28 respectively.

4.4 State-of-art models

The performance of the proposed models was compared with the state of art approaches for UIBVFED (Table 16), FERG (Table 17), CK+ (Table 18), JAFFE (Table 19), and TFEID (Table 20) dataset respectively. It can be noted from the aforementioned tables (Tables 15, 16, 17, 18, 19, 20) that our proposed models outperform the existing state-of-art models on all five datasets. The proposed models exhibit superiority over the challenging datasets UIBVFED and FERG.

Moreover, our proposed models exhibit better performance than existing deep learning models such as Deep Comprehensive Multi patches Aggregation CNNs (Xie et al. 2018), hierarchical CNNs (Kim et al. 2019), IB-CNN (Han et al. 2016), Attentional CNNs (Minaee et al. 2019), DeepExpr (Aneja et al. 2016), Ensemble Multi-feature (Zhao et al. 2018), TL-HO (Ozcan and Basturk 2020), CNNS

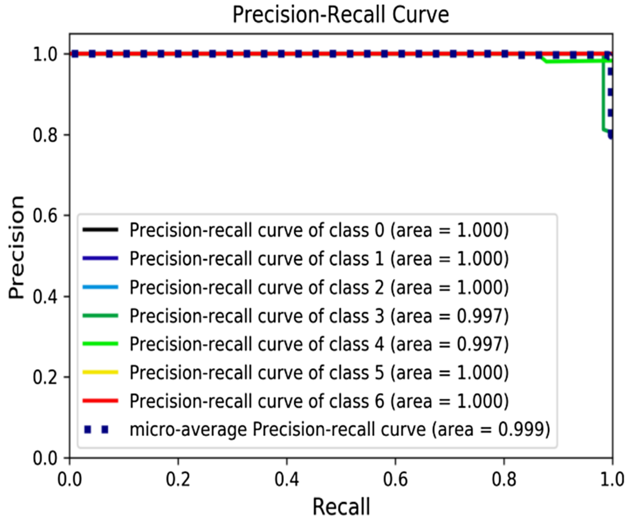

Fig. 20 Precision-Recall curve of DCNN-VC on TFEID

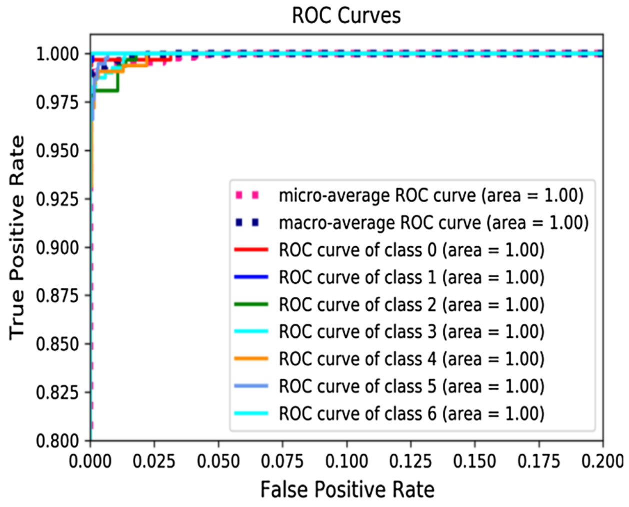

Fig. 21 ROC curves of DCNN on UIBVFED

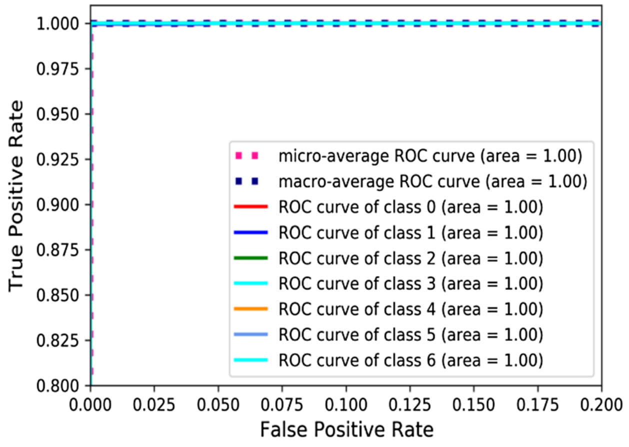

Fig. 22 ROC curves of DCNN on FERG

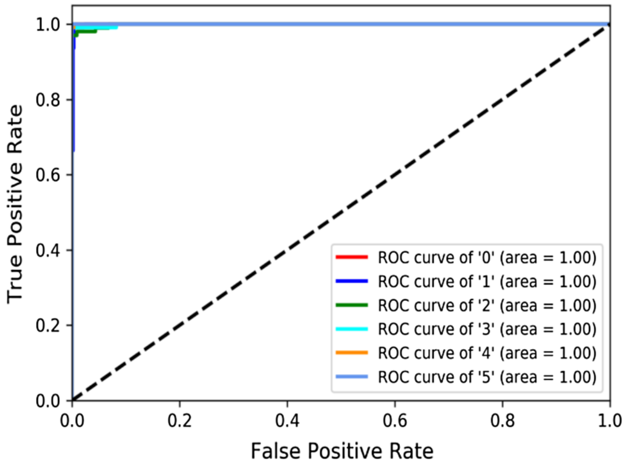

Fig. 23 ROC curves of DCNN-VC on CK+

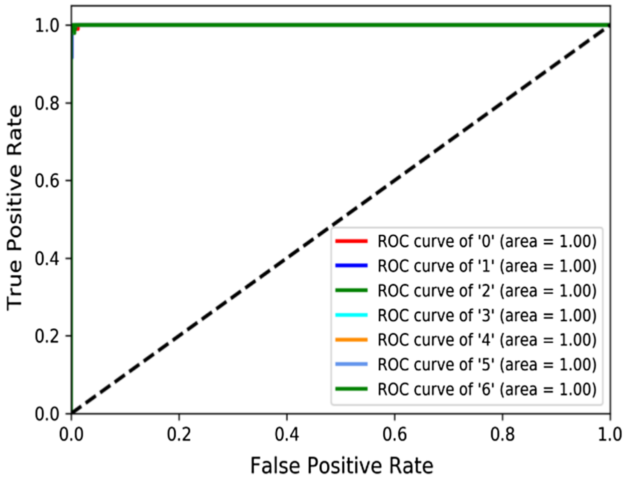

Fig. 24 ROC curves of DCNN-VC on JAFFE

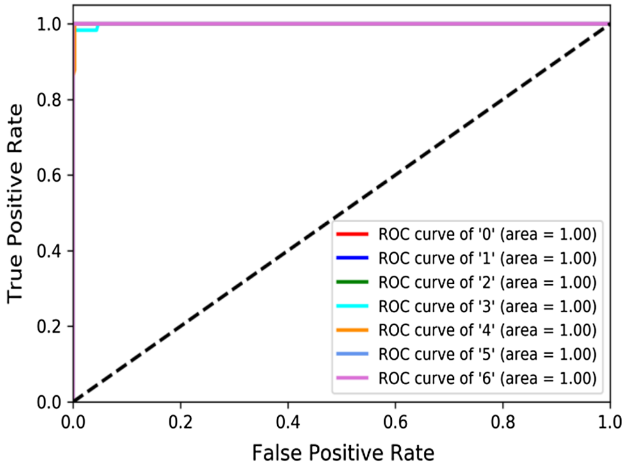

Fig. 25 ROC curves of DCNN-VC on TFEID

Table 9 AUC score of proposed models

Model	Dataset	AUC Score
DCNN	UIBVFED	99.45
DCNN	FERG	99.98
DCNN-VC	CK+	99.42
DCNN-VC	JAFFE	99.75
DCNN-VC	TFEID	99.58

(Gonzalez-Lozoya et al. 2020) and Hybrid DL (Garima and Hemraj 2020).

5 Conclusions

In this work, a new model for recognizing the facial expressions of virtual characters was proposed using multi-block DCNN and ensemble classifiers. In multi-block DCNN, we

Table 10 Confusion matrix of DCNN on UIBVFED (%)

	Anger	Disgust	Fear	Joy	Neutral	Sad	Surprise
Anger	99.97	–	–	0.03	–	–	–
Disgust	–	100	–	–	–	–	–
Fear	–	–	97.44	-	0.64	1.92	–
Joy	0.25	0.50	–	98.75	0.25	0.25	–
Neutral	–	0.30	–	0.63	98.10	0.94	–
Sad	–	–	–	0.53	–	99.47	–
Surprise	–	–	–	–	–	–	100

Table 11 Confusion matrix of DCNN on FERG (%)

	Anger	Disgust	Fear	Joy	Neutral	Sad	Surprise
Anger	100	-	–	–	–	–	–
Disgust	–	99.96	–	0.04	–	–	–
Fear	–	–	99.95	–	–	0.05	–
Joy	–	–	–	100	–	–	–
Neutral	–	–	–	–	99.85	0.15	–
Sad	–	–	–	–	–	100	–
Surprise	-	–	0.04	–	–	–	99.96

Table 12 Confusion matrix of DCNN-VC on CK+ (%)

	Surprise	Fear	Sad	Anger	Disgust	Joy
Surprise	100	–	–	–	–	–
Fear	0.91	99.09	–	–	–	–
Sad	–	3	97	–	–	–
Anger	–	–	0.91	99.09	–	–
Disgust	–	–	0.93	–	99.07	–
Joy	–	–	–	–	–	100

Table 13 Confusion matrix of DCNN-VC on JAFFE (%)

	Anger	Disgust	Fear	Joy	Neutral	Sad	Surprise
Anger	100	–	–	–	–	–	–
Disgust	–	100	–	–	–	–	–
Fear	–	–	100	–	–	–	–
Joy	–	–	–	100	–	–	–
Neutral	–	–	–	–	100	–	–
Sad	–	0.97	–	0.97	–	98.06	–
Surprise	–	–	–	1.06	–	–	98.94

Table 14 Confusion matrix of DCNN-VC on TFEID (%)

	Anger	Contempt	Disgust	Fear	Joy	Sad	Surprise
Anger	100	–	–	–	–	–	–
Contempt	–	99.05	–	–	0.95	–	–
Disgust	0.97	–	99.03	–	-	–	–
Fear	–	–	–	98.31	1.69	–	–
Joy	–	0.87	–	–	99.13	–	–
Sad	–	–	–	–	–	100	–
Surprise	–	–	–	–	–	–	100

Table 15 Overall accuracy of pre-trained models on various datasets (%)

Pre-trained models	UIBVFED	FERG	CK+	JAFFE	TFEID
ResNet-50 -VC	98.33	99	92.53	91.59	98.77
VGG19-VC	95.20	98.35	91.90	88.09	96.31

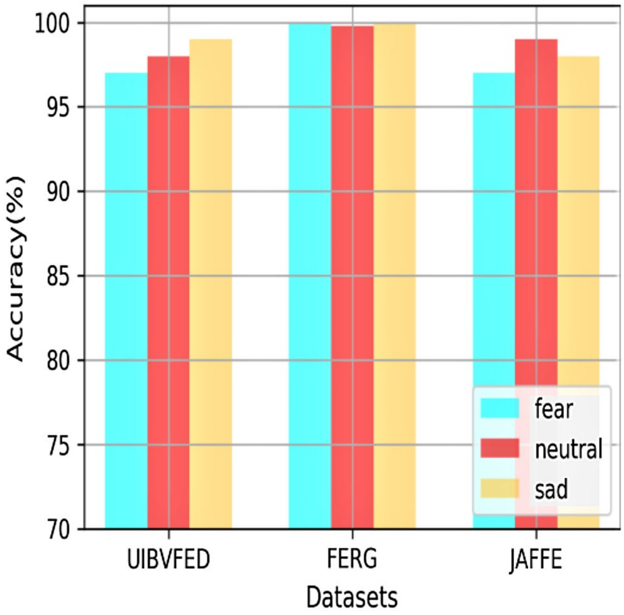

Fig. 26 Performance of DCNN on closed expressions

defined four blocks with various computational elements to extract the discriminative features from facial images and these features were fed into the softmax layer for classification. In DCNN-SVM, the DCNN model was applied to obtain the discriminate features from face images then these features were given as input to ensemble bagging with SVM as a base classifier for facial expression classification.

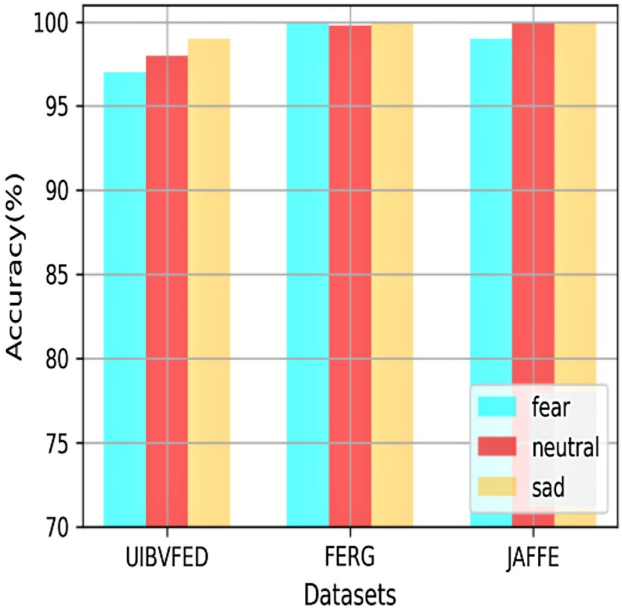

Fig. 27 Performance of DCNN-SVM on closed expressions

In DCNN-VC, the DCNN model was applied to obtain the discriminate features from face images. These extracted features were forwarded to the ensemble of classifiers with a voting technique for emotion recognition. Our proposed models have experimented on five publicly available datasets (UIBVFED, FERG, CK+, JAFFE, and TFEID). The UIBVFED and FERG are challenging datasets due to intra-class variation and inter-class similarities. The proposed models overcome these two issues and produced the best accuracy on these two datasets. The proposed models outperform the state of existing works on these five datasets. The limitation of the proposed models is the relatively low performance in case of face occlusion. In future work, this issue will be taken up for further enhancements.

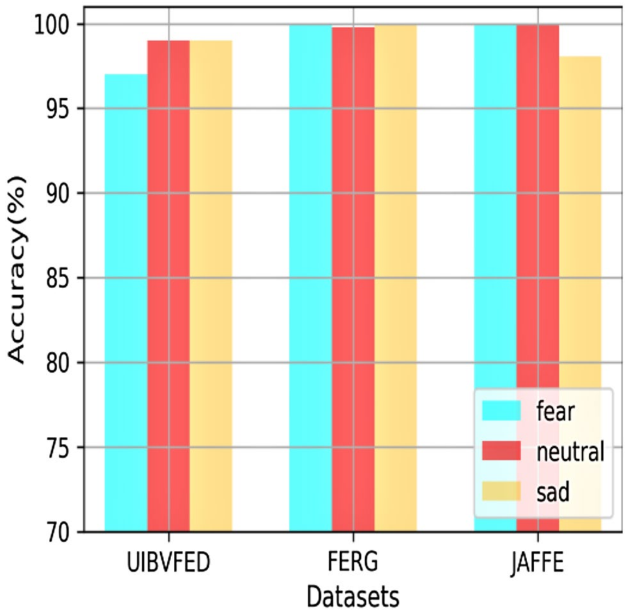

Fig. 28 Performance of DCNN-VC on closed expressions

Table 16 Performance comparisons of existing methods on UIB-VFED (%)

Methods	Accuracy
Perez-Gomez et al. 2020	93.92
DCNN	99.02
DCNN-SVM	98.54
DCNN-VC	98.85

Table 17 Performance comparisons of existing methods on FERG (%)

Methods	Accuracy
Aneja et al. 2016	89.02
Zhao et al. 2018	97.00
Feutry et al. 2018	98.20
Minaee et al. 2019	99.30
DCNN	99.97
DCNN-SVM	99.95
DCNN-VC	99.96

Table 18 Performance comparisons of existing methods on CK+ (%)

Methods	Accuracy
Mayya et al. 2016	97.00
Kim et al. 2019	96.46
Rami Reddy et al. 2019	98.00
Han et al. 2016	95.10
Mandal et al. 2019	90.63
Minaee et al. 2019	98.00
Gonzalez-Lozoya et al. 2020	89.40
Sadeghi H & Raie A-A 2019	95.11
Bendjillali, et al. 2019	96.46
Xie et al. 2019	95.88
Yang B et al. 2018	97.00
Sikkandar H & Thiyagarajan H 2020	98.12
Li, K et al. 2020	97.38
Fan et al. 2020	98.92
Zhang et al. 2020	98.50
DCNN	97.77
DCNN-SVM	98.09
DCNN-VC	99.04

Table 19 Performance comparisons of existing methods on JAFFE (%)

Methods	Accuracy
Mayya et al. 2016	98.12
Kim et al. 2019	91.27
Minaee et al. 2019	92.80
Bendjillali, et al. 2019	98.43
Xie et al. 2019	99.32
Gogić et al. 2020	98.10
Ozcan & Basturk, 2020	99.53
Garima and Hemraj 2020	94.12
Gonzalez-Lozoya et al. 2020	98.26
Yang B et al. 2018	92.20
Sikkandar et al. 2020	98.25
Li, K et al. 2020	97.18
Zhang et al. 2020	92.30
DCNN	98.73
DCNN-SVM	99.29
DCNN-VC	99.57

Table 20 Performance comparisons of existing methods on TFEID (%)

Methods	Accuracy
Goyani et al. 2017	89.58
Farajzadeh, et al. 2014	92.54
Benitez-Garcia et al. 2017	94.94
Ashir, A.M. et al. 2017	93.38
Xie et al. 2019	93.36
DCNN	98.63
DCNN-SVM	99.04
DCNN-VC	99.31

Funding Not Applicable.

Availability of data and materials The datasets used in the current study are available in the below links. UIBVFED: http://ugivia.uib.es/uibvfed/. FERG: http://grail.cs.washington.edu/projects/deepexpr/ferg-2d-db.html. CK+: https://www.pitt.edu/~emotion/ck-spread.htm. JAFFE: https://zenodo.org/record/3451524#.X3NMx2gzbIU. TFEID:http://bml.ym.edu.tw/tfeid/modules/wfdownloads/.

Compliance with ethical standards

Conflict of interest The authors doesn't have any conflicts of interest.

Code availability Not Applicable.

References

Kim J, Kim B, Roy PP, Jeong D (2019) Efficient facial expression recognition algorithm based on hierarchical deep neural network structure. In: IEEE Access 7:41273–41285.

Mandal M, Verma M, Mathur S, Vipparthi S, Murala S, Deveerasetty K (2019) Radap: regional adaptive affinitive patterns with logical operators for facial expression recognition. IET Image Process 13:850–861

Bartlett MS, Littlewort G, Fasel I, Movellan JR (2003) Real time face detection and facial expression recognition: Development and applications to human computer interaction. Proc IEEE Conf Comput Vis Pattern Recog Workshop 5:53–53.

Teow MYW (2017) Understanding convolutional neural networks using a minimal model for handwritten digit recognition(2017). In: 2017 IEEE 2nd international conference on automatic control and intelligent systems (I2CACIS), Kota Kinabalu, pp 167–172.

Lyons M, Akamatsu S, Kamachi M, Gyoba J (1998) Coding facial expressions with Gabor wavelets. In: Proceeding - 3rd IEEE Int Conf Autom Face Gesture Recognition, FG 1998, pp 200–205

Minaee S, Abdolrashidi A (2019) Deep-emotion: Facial expression recognition using attentional convolutional network. arXiv preprint http://arxiv.org/abs/1902.01019

Xie S, Hu H, Wu Y (2019) Deep multi-path convolutional neural network joint with salient region attention for facial expression recognition. Pattern Recognit 92:177–191

Connie T, Al-Shabi M, Cheah WP, Goh M (2017) Facial expression recognition using a hybrid CNN–SIFT aggregator. In: Proceedings of the MIWAI, Cham, Switzerland Springer, vol 10607. pp 139–149

Fan Y, Li V, Lam JCK (2020) Facial expression recognition with deeply-supervised attention network. In: IEEE transactions on affective computing, vol 3045, pp 1–1

Alsmirat MA, Al-Alem F, Al-Ayyoub M, Jararweh Y, Gupta B (2019) Impact of digital fingerprint image quality on the fingerprint recognition accuracy. Multimedia Tools and Applications 78(3):3649–3688

Aneja D, Colburn A, Faigin G, Shapiro L, Mones B (2016) Modeling stylized character expressions via deep learning. Asian conference on computer vision. Springer, Cham, pp 136–153

Ashir AM, Eleyan A (2017) Facial expression recognition based on image pyramid and single-branch decision tree. Signal, Image Video Process, 11:1017–1024

Bendjillali RI, Beladgham M, Merit K, Taleb-Ahmed A (2019) Improved facial expression recognition based on DWT feature for deep CNN. Electronics 8:324

Benitez-Garcia G, Nakamura T, Kaneko M (2017) Facial expression recognition based on local Fourier coefficients and facial Fourier descriptors. J Signal Inf Process 08:132–151

Chen L-F, Yen Y-S (2007) Taiwanese Facial Expression Image Database. Institute of Brain Science, National Yang-Ming University, Taipei, Taiwan, Brain Mapping Laboratory

Reddy Chirra VR, Uyyala SR, Kishore Kolli VK (2019) Deep CNN: A machine learning approach for driver drowsiness detection based on eye state. Rev d'Intelligence Artif 33:461–466

Cockburn J, Bartlett M, Tanaka J, Movellan J, Pierce M, Schultz R (2008) SmileMaze: a tutoring system in real-time facial expression perception and production in children with autism spectrum disorder. In: Proceedings of the workshop facial bodily expressions control adaptation games

Ekman P, Friesen WV, O'Sullivan M, Chan AYC, Diacoyanni-Tarlatzis I, Heider KG, Krause R, LeCompte WA, Pitcairn T, Bitti PER (1972) Universals and cultural differences in facial expressions of emotion. J Pers Soc Psychol 53(4):712–717

Ekman P, Friesen W (1978) The Facial Action Coding System: A Technique for the Measurement of Facial Movement. Consulting Psychologists Press, Santa Clara, CA, USA

Farajzadeh N, Pan G, Wu Z (2014) Facial Expression recognition based on meta probability codes. Pattern Anal Appl 17:763–781

Feutry C, Piantanida P, Bengio Y, Duhamel P (2018) Learning anonymized representations with adversarial neural networks. arXiv 1–20

Gogić I, Manhart M, Pandžić IS, Ahlberg J (2020) Fast facial expression recognition using local binary features and shallow neural networks. Vis Comput 36:97–112

González-Lozoya S, de la Calleja J, Pellegrin L, Escalante HJ, Medina M, Benitez-Ruiz A (2020) Recognition of facial expressions based on CNN features. Multimedia Tools Appl 79:13987–14007

Goyani M, Patel N (2017) Multi-level Haar wavelet based facial expression recognition using logistic regression. Indian J Sci Technol 10:1–9

Han S, Meng Z, Khan AS, Tong Y (2016) Incremental boosting convolutional neural network for facial action unit recognition. In: Advances in neural information processing systems, pp 109–117

He K, Zhang X, Ren S and Sun J, (2016) Deep Residual Learning for Image Recognition, In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, 2016, pp 770–778

Kim H-C, Pang S, Je H-M, Kim D, Bang S (2002) Support vector machine ensemble with bagging, vol. 2388, pp 397–407

Mayya V, Pai RM, Manohara Pai MM (2016) Automatic Facial Expression Recognition Using DCNN. Procedia Comput Sci 93:453–461

Lango M, Stefanowski J (2017) Multi-class and feature selection extensions of roughly balanced bagging for imbalanced data. J Intell Inf Syst 50(1):97–127

Lee SH, Plataniotis KN, Ro YM (2014) Intra-Class Variation Reduction Using Training Expression Images for Sparse Representation Based Facial Expression Recognition. In: IEEE Transactions on Affective Computing, vol. 5, pp 340–351

Li K, Jin Y, Akram MW, et al (2020) Facial expression recognition with convolutional neural networks via a new face cropping and rotation strategy. Vis Comput 36:391–404

Li Y, Zeng J, Shan S, Chen X (2019) Occlusion aware facial expression recognition using CNN with attention mechanism. IEEE Trans Image Process 28(5):2439–2450

Li Y, Shi H, Chen L, Jiang F (2019) Convolutional approach also benefits traditional face pattern recognition algorithm [208!] International Journal of Software Science and Computational Intelligence, vol. 11, pp 1–16

Lucey P, Cohn JF, Kanade T, Saragih J, Ambadar Z, Matthews I (2010), The extended cohn-kanade dataset (CK+): a complete expression dataset for action unit and emotion-specified expression. In: Proceedings of the third international workshop on CVPR for human communicative behavior analysis, San Francisco, USA, pp 94–101

Mahesh Babu D, VenkataRamiReddy Ch, Srinivasulu Reddy U (2019) An automatic driver drowsiness detection system using DWT and RBFNN. Int J Recent Technol Eng 7(5S4):41–44

Mehrabian G (2007) Nonverbal communication. Aldine, New Brunswick, NJ, USA

Oliver MM, Alcover EA (2020) UIBVFed: Virtual facial expression dataset. PLoS One 15:1–10

Ozcan T, Basturk A (2020) Static facial expression recognition using convolutional neural networks based on transfer learning and hyperparameter optimization. Multimedia Tools and Applications 79:26587–26604

Perez-Gomez V, Rios-Figueroa HV, Rechy-Ramirez EJ, Mezura-Montes E, Marin-Hernandez A (2020) Feature selection on 2D and 3D geometric features to improve facial expression recognition. Sensors 20:1–20

Pons G, Masi D (2018) Supervised committee of convolutional neural networks in automated facial expression analysis. IEEE Trans Affect Comput 9:343–350

Pu X, Fan, Ke& Chen, Xiong&Ji, Luping & Zhou, Zhihu. (2015) Facial expression recognition from image sequences using twofold random forest classifier. Neurocomputing 168:1173–1180

Purnama J, Sari R (2019) Unobtrusive academic emotion recognition based on facial expression using rgb-d camera using adaptive-network-based fuzzy inference system (ANFIS). Int J Softw Sci Comput Intell 11:1–15

Ramireddy C V., Kishore KVK (2013) Facial expression classification using Kernel based PCA with fused DCT and GWT features. 2013 IEEE Int Conf Comput Intell Comput Res IEEE ICCIC, vol. 2013, pp 2–7

VenkataRamiReddy Ch, Kishore KVK, Bhattacharyya D, Kim TH (2014) Multi-feature fusion based facial expression classification using DLBP and DCT. Int J Softw Eng Appl 8:55–68

Reddy CVR, Reddy US, Kishore KVK (2019) Facial emotion recognition using NLPCA and SVM. Trait du Signal 36:13–22

Sadeghi H, Raie AA (2019) Human vision inspired feature extraction for facial expression recognition. Multimed Tools Appl 78:30335–30353

Sikkandar H, Thiyagarajan R (2020) Deep learning based facial expression recognition using improved Cat Swarm Optimization. J Ambient Intell Human Comput.

Soleymani M, Pantic M (2013) Emotionally Aware TV. Proc TVUX-2013 Work Explor Enhancing User Exp TV ACM CHI 2013

Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. CoRR

Verma G, Verma H (2020) Hybrid-Deep Learning Model for Emotion Recognition Using Facial Expressions. Rev Socionetwork Strateg 14:171–180

Viola P, Jones M (2004) Robust real-time face detection. Int J Comput Vis 57:137–154

Wang Q, Jia K, Liu P (2016) Design and Implementation of Remote Facial Expression Recognition Surveillance System Based on PCA and KNN Algorithms. Proc - 2015 Int Conf Intell Inf Hiding Multimed Signal Process IIH-MSP 2015, pp 314–317

Whitehill J, Serpell Z, Lin YC, et al (2014) The faces of engagement: Automatic recognition of student engagement from facial expressions. IEEE Trans Affect Comput 5:86–98

Xie S, Hu H (2019) Facial expression recognition using hierarchical features with deep comprehensive multipatches aggregation convolutional neural networks. IEEE Trans Multimedia 21:211–220

Xie S, Hu H, Yin Z (2017) Facial expression recognition using intra-class variation reduced features and manifold regularisation dictionary pair learning. IET Comput Vis 12(4):458–465

Yang B, Cao J, Ni R, Zhang Y (2018) Facial expression recognition using weighted mixture deep neural network based on double-channel facial images. IEEE Access 6:4630–4640

Zhang H, Huang B, GuohuiTian, (2020) Facial expression recognition based on deep convolution long short-term memory networks of double-channel weighted mixture. Pattern Recogn Lett 131:128–134

Zhao H, Liu Q, Yang Y (2018) Transfer Learning with Ensemble of Multiple Feature Representations. Proc - 2018 IEEE/ACIS 16th Int Conf Softw Eng Res Manag Appl SERA 2018 54–61

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