

Realistic Smile Expression Recognition Approach Using Ensemble Classifier with Enhanced Bagging

Oday A. Hassen^{1,*}, Nur Azman Abu¹, Zaheera Zainal Abidin¹ and Saad M. Darwish²

¹University Technical Malaysia Melaka, Department of Information Technology, Melaka, 76100, Malaysia

²Institute of Graduate Studies and Research, University of Alexandria, Alexandria, El Shatby, 21526, Egypt

*Corresponding Author: Oday A. Hassen. Email: odayali@uowasit.edu.iq

Received: 03 April 2021; Accepted: 08 June 2021

Abstract: A robust smile recognition system could be widely used for many real-world applications. Classification of a facial smile in an unconstrained setting is difficult due to the invertible and wide variety in face images. In this paper, an adaptive model for smile expression classification is suggested that integrates a fast features extraction algorithm and cascade classifiers. Our model takes advantage of the intrinsic association between face detection, smile, and other face features to alleviate the over-fitting issue on the limited training set and increase classification results. The features are extracted taking into account to exclude any unnecessary coefficients in the feature vector; thereby enhancing the discriminatory capacity of the extracted features and reducing the computational process. Still, the main causes of error in learning are due to noise, bias, and variance. Ensemble helps to minimize these factors. Combinations of multiple classifiers decrease variance, especially in the case of unstable classifiers, and may produce a more reliable classification than a single classifier. However, a shortcoming of bagging as the best ensemble classifier is its random selection, where the classification performance relies on the chance to pick an appropriate subset of training items. The suggested model employs a modified form of bagging while creating training sets to deal with this challenge (error-based bootstrapping). The experimental results for smile classification on the JAFFE, CK+, and CK+48 benchmark datasets show the feasibility of our proposed model.

Keywords: Ensemble classifier; smile expression detection; features extraction

1 Introduction

Automatic facial expression detection has been extensively researched during the past two decades and has developed into a very productive field of study in computer vision and pattern recognition. Classification of facial expressions is a critical aspect of human-computer interaction and associated areas. The smile, as a significant component of facial expression, conveys a multitude of emotional messages, including excitement, satisfaction, attractiveness, and kindness, and therefore plays a critical role in human speech. Automatic smile classification technology has



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

permeated people's lives and is growing in popularity as commercial multimedia devices such as digital cameras, digital images, and social robots become more prevalent. Faces' extensive visual alterations, such as occlusions, pose transitions, and drastic lightings, make certain functions very difficult in real-world implementations [1–5]. Fig. 1 shows the components (facial action units (AUs)) of facial muscle movements [6,7].

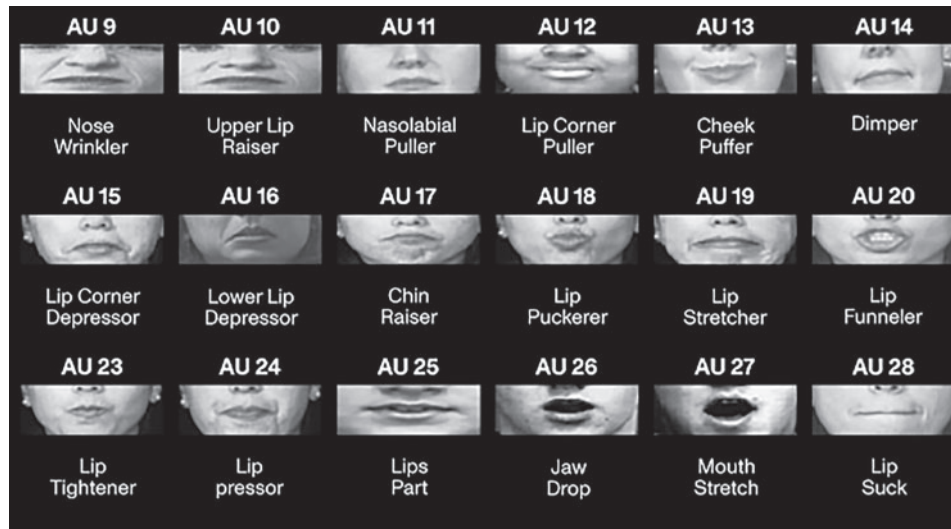


Figure 1: Facial action units (AUs)

In managed settings, current smile expression recognition has promising results, but performance on real-world datasets is still unsatisfactory [8]. This is because there are broad differences in facial appearances through the color of the skin, lighting, posture, expression, orientation, head location, lightening state, and so on. By incorporating deep learning [9], optimization [10], and ensemble classification, automatic methods of identification for smile expression are suggested to deal with existing system difficulties. Five key steps are given for the planned work: pre-processing, a deep evolutionary neural network used for feature extraction, features selection utilizing swarm optimization, and facial expression classification employing support vector machine, ensemble classifiers [11], and neural network [12]. Fig. 2 depicts the recognition of facial expression' action units.

Bagging and Boosting are close in that they are both ensemble approaches in which a group of poor learners is assembled to shape a powerful learner capable of outperforming a single learner [11]. Ensembles are a subset of a larger class of approaches known as multi-classifiers, which combine hundreds or thousands of learners with a similar purpose to solve a problem. Though Bagging's training stage is concurrent (each paradigm is constructed independently), Boosting constructs the new learner sequentially. Each classifier is trained on data using boosting algorithms, taking into account the performance of previous classifiers, as shown in Fig. 3. Weights are redistributed after each training phase. The weights of misclassified data are increased to highlight the more complicated scenarios. This way, subsequent learners' attention would be drawn to them during their training.

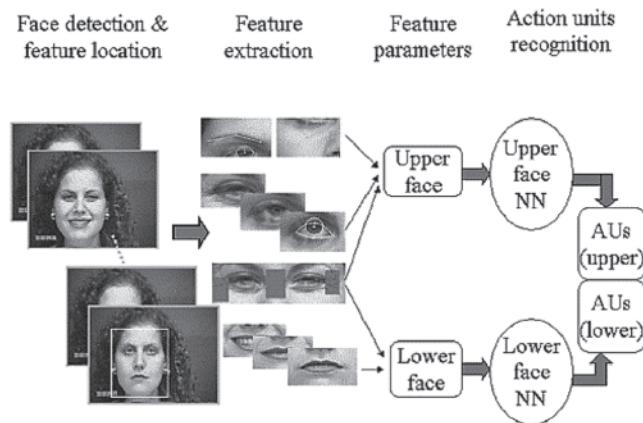


Figure 2: Recognizing action units for facial expression

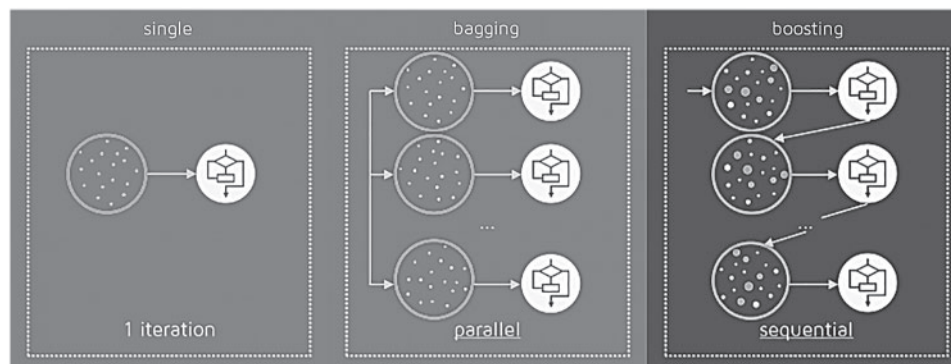


Figure 3: Ensemble classifiers

All that is needed to predict the class of new data is to apply the N learners to the new observations. The effect of Bagging is determined by combining the responses of the N learners (or majority vote). Boosting, on the other hand, applies a second collection of weights, this time to the N classifiers, in order to compute a weighted average of their predictions. The algorithm then assigns weights to each resulting model during the Boosting training stage. A learner with superior classification outcomes on training data would be given a higher weight than one with inferior classification results. Thus, when assessing a new learner, Boosting must often maintain stock of the learner's mistakes (see Fig. 4). Bagging and boosting also help to reduce the uncertainty of a single calculation by combining predictions from several models. As a consequence, a more stable model could emerge. If the issue is that the single model performs poorly, Bagging can seldom provide a better bias. Boosting, on the other hand, can result in a combined model with lower errors by optimizing the advantages and minimizing the drawbacks of the single model. By comparison, if the single model's complexity is over-fitting, Bagging is the optimal solution. Boosting, on the other hand, does not help prevent over-fitting; in effect, this technique exacerbates the issue. As a result, Bagging is more efficient than Boosting.



Figure 4: Training stage of ensemble classifiers

1.1 Problem Statement

Real-time and effective smile detection can significantly enhance the development of facial expression recognition. Classification of smiles in an unconstrained environment is difficult owing to the invertible and wide variety of facial pictures. The majority of the current works deals with smile detection not smile classification. However, within the current smile classification approaches, their models are not smile's attribute-specific hence their performance may be limited.

The combination of several classifiers, referred to as a classifier ensemble, has previously been shown to improve the accuracy of smile classification when opposed to single classifiers. Because of its demonstrated efficiency gain on classification tasks, bagging is one of the most frequently employed ensemble learning methods. The downside of the conventional bootstrap approach is that the training subsets produced by random selection with replacement do not include a high proportion of misclassified instances. In other words, since the difficult-to-classify instances will not be included in the training sets, the learning algorithm is unable to concentrate on such data points in order to reduce training errors.

1.2 Contribution

The main goal of this paper is to build an adaptive model for the classification of facial smiles that incorporates both the fast features extraction module and the ensemble classifier to increase the accuracy of facial expression classification. In contrast to the current methods of classifying smile, which rely on deep neural networks to extract features that, in turn, require a large number of samples and more computation, the suggested model relies on a histogram-based feature extraction module to reduce and improve the discriminatory capability of the extracted features. Furthermore, the suggested model utilizes the ensemble classification concept to build an accurate classifier depending on a small number of samples. To overcome the drawback of bagging ensemble classifiers, the suggested model employs eBootstrapping that ensures the presence of misclassified instances in training sets to encourage their correct classification. A chain of experiments proves that the suggested model technique is substantially reliable and quicker than other widespread prototypes. This paper is a revised version of our research paper [13]. This version of the paper provides a more comprehensive and systematic report regarding eBagging ensemble classifiers.

The remainder of the article is organized as follows. Section 2 discusses the current related work. Section 3 presents the proposed model steps in detail. Section 4 explains experimental designs. Section 5 includes the conclusion and future work.

2 Related Work

Several scientific studies have been performed in the field of identification of facial expressions that apply to a range of technologies such as computer vision, image recognition, bioindustry, forensics, authentication of records, etc. [14–16]. In a recent study, the pyramid histogram of orientation gradient features and an Adaboost and SVM classifier are used to build a high-performance smile classification system [14]. These algorithms work admirably on several publicly available standard databases. However, as applied to more difficult and practical problems such as classifying random expressions with varying levels of light, pose combinations, graphical transformations, occlusion, and clutter, the accuracy of the majority of these algorithms degrades. Among their flaws is their omission of high-level information, such as relationships between local orientations. As a result, there is plenty of space for designing more effective algorithms that solve real-world issues.

In many studies, Principal Component Analysis (PCA) was used to provide a coding framework for facial action that models and recognizes various forms of facial action [17,18]. However, PCA-based solutions are subject to a dilemma in which the projection maximizes variance in all images and negatively affects recognition performance. Independent Component Analysis (ICA) is adapted to perform expression recognition to elicit statistically independent local face characteristics that proceed better than PCA [19]. Recently, deep learning among the science community has attracted substantial interest in the field of smile detection [20]. Numerous previous approaches classify several attributes using a single deep network and solve them concurrently. However, since their templates are not attribute-specific, their success on a particular attribute (gender or smile, for example) can be constrained. Certain approaches define distinct attributes separately but fail to account for the intrinsic association between smile, and other face attribute prediction tasks [21,22].

As the feature extraction module represents the core module for facial classification, many algorithms are suggested to select the characteristics of the facial image [23–32]. Through using meta-heuristic evolutionary optimization algorithms such as Ant Colony Optimization (ACO), Bee Colony Optimization (BCO), Particle Swarm Optimization, chaotic gray-wolf algorithm, Whale Optimization Algorithm (WOA), and Multi-Verse Optimization (MVO) algorithm will minimize drawbacks of facial features selection such as redundancy. Such approaches are inefficient in evaluating the global optimum concerning the pace of convergence, capability for experimentation, and consistency of solution [28,29]. To overcome these problems, a chaotic MVO algorithm (CMVO) is applied that minimize the slow convergence problem and trap local optima [31,32]. A graphical model for extraction and description of functions using a hybrid approach to recognize a person's facial expressions was developed in [33]. However, large memory complexity is the main disadvantage. In [34], a Zernike model was developed based on a local moment to classify a person's expressions. However, it takes a long training time and has a large difficulty to understand and interpret the final model.

Recently, several methods for classifying face speech using a neural network approach have been suggested [35,36]. In [37], the detection technique was used to perform automatic recognition of facial expressions using the Elman neural network to recognize feelings. However, neural networks demand processors with parallel processing power, by their structure. Furthermore, experience and trial and error are used to achieve the appropriate network structure. Inspired by the good performance of the Conventional Neural Network (CNNs) in computer vision tasks, such as image classification and face recognition, several CNNs based smile classification approaches have been proposed in recent years. In [38] a deeper CNN that has a complex CNN network

consisting of two convolution layers, each accompanied by a max-pooling and four initiation layers was suggested for facial expression recognition. Another related work in [39] utilizes deep learning-based facial expression to minimize the dependency on face physics. In [40], a deep learning approach is introduced to track consumer behavior patterns by measuring customer behavior patterns. The authors in [41] presented a deep region and multi-label learner's scheme for estimation of head poses and study of facial expressions to report the interest of customers.

In contrast to the previous methods, which rely on a deep learning concept for smile classification and in order to solve the problem facing this type of learning in terms of its difficulty to gather vast amounts of training data for facial expression recognition under different circumstances; the suggested approach utilizes both fast feature extraction technique and ensemble classifier in a unified framework. The ensemble classifier can process a large number of features. Even so, the effectiveness of this method is fundamentally dependent on the extracted features, which may indeed not require much time to realize its purpose. In this case, the fast feature extraction technique is used to exclude any redundant coefficients from a vector of features, thus increasing the discriminatory capacity of the derived features and reducing computational complexity.

3 Methodology

3.1 Face Detection

The first step in the identification of a smile is to locate the face in the picture. For this function, the Viola-Jones method was used [42]. The face identified represents a Region of Interest (ROI) in the picture of a smile. The Viola-Jones method has also been used to locate the eyes and mouth. The area of the eyebrow was determined from the position of the eye region. After identification of facial regions, different techniques of image processing are used in each of the detected ROIs to remove the eyes and mouth. Then a search is carried out on each of the extracted components to identify facial expressions [43]. Fig. 5 shows the block diagram of the proposed model.

To soften the image, minor noises such as defects in the image and scarcely visible lines were discarded. In order to locate points of interest on the face, it was initially important to enhance and extract the relevant information from the image. For this reason, different techniques of image processing have been used in this work such as contrast correction, thresholding, context subtraction, contour detection, and Laplacian and Gaussian filters for extraction points of interest. To segment the image into regions (set of pixels) two methods were used in segmentation: thresholding and morphological operations. To re-move the edges of the eyes and mouth, the canny detector was used and a search was carried out on each of the resulting edges to detect facial landmarks. Fig. 6 illustrates the output facial landmarks that are detected from the image processing techniques in each of the ROIs; see [43] for in-depth details.

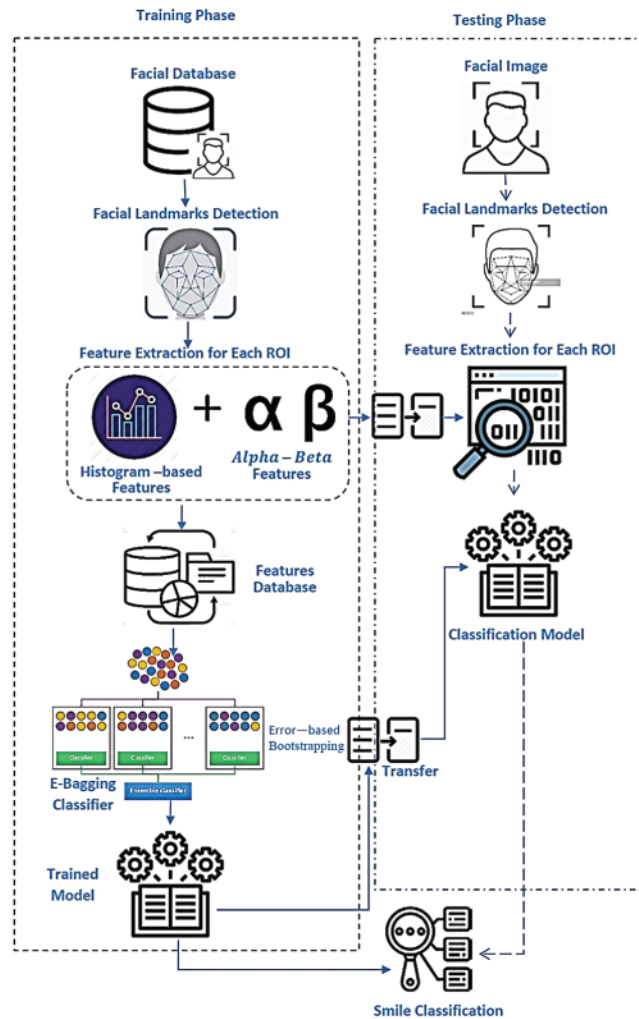


Figure 5: System flowchart

3.2 Feature Extraction

After the pre-processing stage, feature extraction is done in a facial expression recognition system. The most important knowledge present in the original ROI is a kind of dimensional decrease technique. This is the knowledge gathered in a small space from the photo. The main goal of the extraction function is to minimize the initial ROI size into a manageable processing vector that has histogram and alpha and beta features.

3.2.1 Histogram Feature Extraction

Herein, six parameters of histograms are calculated for each ROI. The histogram features are statically based features as a model of the probability distribution of the gray levels. We define the first-order histogram probability as [44]:

$$P(g) = \frac{N(g)}{M} \tag{1}$$

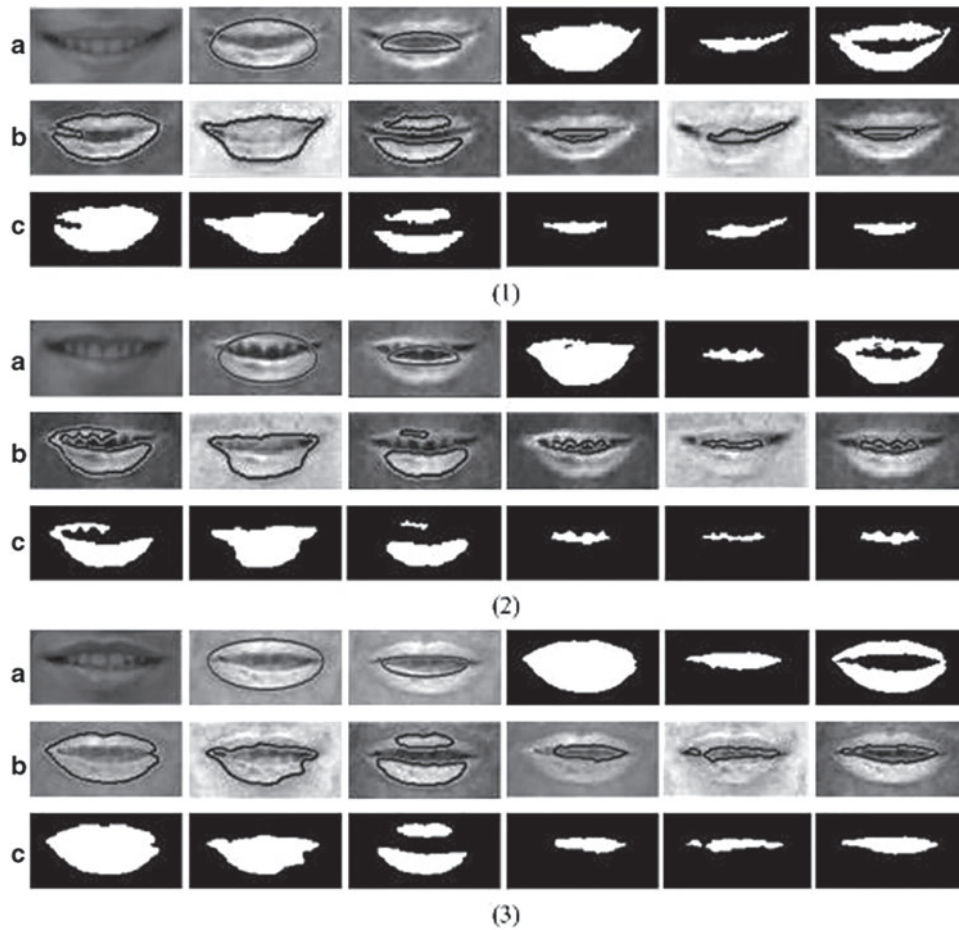


Figure 6: (a) From left-hand to right-hand: RGB images after brightness equalization, out-side contours, internal contours, outcomes of outside boundary, outcomes of internal boundary, finishing segmentation outcomes; (b) Convergence outcomes of outside and internal contour; (c) Segmentation outcomes of images in b

$$\mu = \sum_{g=0}^{L-1} g P(g) \quad (2)$$

$$\sigma = \sum_{g=0}^{L-1} (g - P_1)^2 P(g) \quad (3)$$

$$P_z = \sum_{g=0}^{L-1} \frac{(g - \mu)^z P(g)}{(\mu_1)^{z/2}} \quad z \in \{3, 4\} \quad (4)$$

$$\zeta = \sum_{g=0}^{L-1} [P(g)]^2 \quad (5)$$

$$\eta = - \sum_{g=0}^{L-1} P(g) \log_2 [P(g)] \quad (6)$$

$N(g)$ is the number of pixels at grey level value g , and M is the number of pixels in the ROI. $P(g)$ has all values less than or equal to 1. The total number of grey levels available will be L , so the grey levels range from 0 to $L-1$. Histogram probabilities include mean (μ), standard deviation (σ), skewness (P_3), kurtosis (P_4), energy (ζ), and entropy (η). See [44] for more details.

3.2.2 Alpha and Beta Features

Alpha and Beta are the comparisons between the area of teeth and lips. In order to reduce the amount of redundant information, the oral region needs to be extracted and the lip area, teeth area, and eye area are taken as a region of interest. A method based on a localized active contour model can segment the mouth area by general structure and face proportion, see [45] for all method details. Fig. 6 illustrates the steps to pick lips area.

$$Alpha(\alpha) = teeth_Area / Lips_Area \quad (7)$$

$$Beta(\beta) = Lip_Width / Eye_Length \quad (8)$$

3.3 eBagging Ensemble Classifier

The ensemble methodology's central concept is to weigh many different classifiers and merge them to create a classifier that outperforms them all. A standard ensemble system for classification tasks is constructed as follows [11]: (1) a training set is a classified dataset that is used to train an ensemble. The training collection can be expressed in a number of different languages. The instances are commonly represented as attribute-value vectors. We use the notation A to denote the collection of input attributes that contains n attributes: $A = \{a_1, \dots, a_i, \dots, a_n\}$, and y to denote the class variable or goal attribute. (2) The base inducer is an induction algorithm that acquires a training set and constructs a classifier that describes the abstract relationship between the input and target attributes. We use the notation $M = I(S)$ to denote a classifier M that was induced on a training set S by inducer I . (3) Diverse Classifier Generator—this portion is in charge of creating diverse classifiers. (4) Combiner—the combiner is in charge of integrating the classifications created by the various classifiers.

By substituting eBootstrap for the traditional bootstrap method in the proposed classification model, the eBagging ensemble classifier improves the conventional bagging technique [45]. The critical distinction is that training sets are created by providing a higher probability of selection to difficult-to-classify instances that were misclassified by the prior learner. The boosting method (i.e., the AdaBoost algorithm) attempts to correctly distinguish the difficult-to-classify cases while ignoring the easy-to-classify examples. However, eBagging approach is twofold distinct from boosting. To begin, eBagging produces training sets from the initial dataset in parallel, while boosting is an iterative method. Second, eBagging does not allocate weight values to individual instances

for the purpose of boosting. Other than that, eBagging copies all tough examples directly into all training sets.

The eBagging classification is a four-step process [45]: (1) Pre-training: a prior classifier is applied to the initial dataset, dividing it into two parts: one containing correctly classified instances and the other containing incorrectly classified instances (incorrectly classified). (2) eBootstrapping: a technique for generating several training sets by explicitly moving misclassified instances and resampling with substitution from classified instances. As a result, each subset of data contains complicated instances. This step adds diversity to the dataset and enables the learning algorithm to work on difficult-to-classify cases, providing us with a fair starting point. (3) Training: Various subsets of the training patterns are used to train the base classifiers. (4) Combining (Aggregating): The base classifiers execute classification operations, and the final prediction is rendered using plurality voting on the ensemble subset outputs. If classifiers dispute, the voting process will be used to exclude the various classifiers' incorrect mistakes [45–48]. Fig. 7 demonstrates the four steps of the eBagging algorithm and Fig. 8 illustrates some different smile categories.

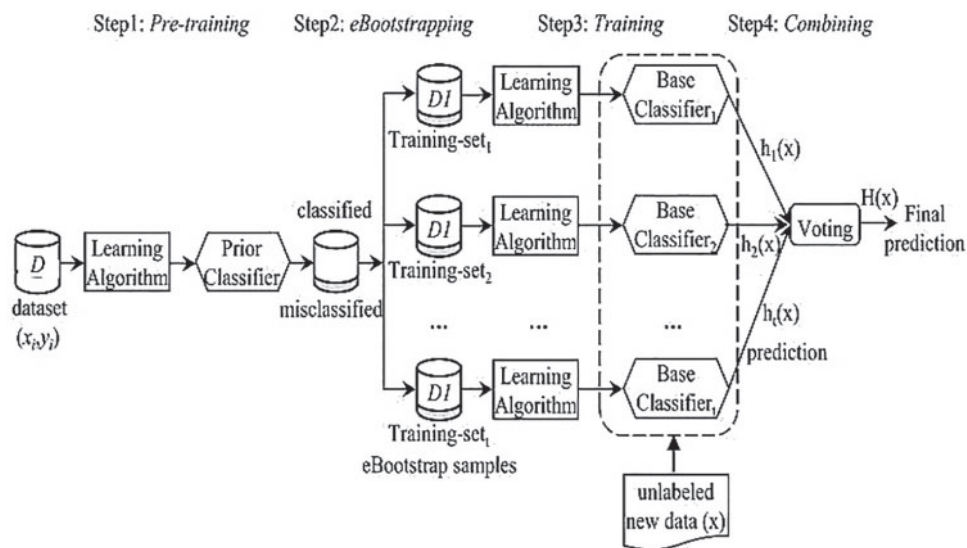


Figure 7: eBagging framework [45]

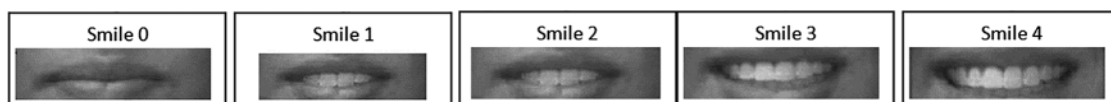


Figure 8: Different smiles for a person

4 Experimental Results

The proposed facial expression recognition system is tested with benchmark datasets that includes Japanese Female Facial Expressions (JAFPE), Extended Cohn-Kanade (CK+), CK+48 Dataset [46]. JAFPE is a Japanese database containing 7 facial expressions with a 256×256 -pixel resolution of 213 images. With 10,414 images with a resolution of 640×490 pixels, the

CK+ database has 13 expressions. The CK+48 dataset has 7 facial expressions with a resolution of 48×48 pixels with 981 images. Features are extracted from ROIs using histogram and lips, teeth, and eyes areas which produce a 21-dimensional feature vector. Herein, 80 percent are selected for training and 20 percent are for testing for each dataset considered. The prototype classification methodology was developed in a modular manner and implemented and evaluated on a Dell™ Inspiron™ N5110 Laptop device, manufactured by Dell computer Corporation in Texas. Intel(R) Core(TM) i5–2410 M processor running at 2.30 GHz, 4.00 GB of RAM, Windows 7 64-bit.

The eBagging classifier was compared to single (without an ensemble strategy), normal bagging ensemble, and AdaBoost learners in this research. Support Vector Machine (SVM), k-Nearest Neighbors (kNN), Decision Tree (C4.5), and Naive Bayes (NB) algorithms were used individually as a basis learner for ensemble methods [48]. Both recognition rate and accuracy, as well as win/tie/loss status and average error rates, are used to evaluate the suggested model's performance. The classifier parameters of SVM, C4.5 and NB classifiers were left as default Weka parameters. The number of neighbors, N for kNN classifier was selected as $\log_2(n)$ where n indicates the number of instances in the respective dataset and k represents the number of classes in each benchmarked dataset. The number of iterations to be performed (ensemble size) were determined as Weka's default parameter, 10. For additional details, see [46].

Three distinct situations were considered when comparing the efficiency of classifiers: classification accuracy for benchmark datasets (shown in Tab. 1), win/tie/loss status for pairwise comparisons of the implemented methods (displayed in Tab. 2), and the average error rates relative to each other. Tab. 1 compares the classification accuracy of the applied approaches (eBagging, standard Bagging, single learner, and AdaBoost) using C4.5, NB, kNN, and SVM as the base learner. The results indicate that eBagging consistently achieves the highest average classification accuracies of 95.37%, 88.61%, and 72.06% for the CK+, CK+48, and JAFFE datasets, respectively, by using the corresponding base learners. The datasets with the best classification accuracy are highlighted in bold. It should be remembered that the proposed scheme does not do well on the JAFFE dataset, owing to the fact that this benchmark database contains an inadequate number of images for each class. In general, using an eBagging classifier needs additional data for proper preparation. When one of the methods from C4.5, SVM, NB, or kNN is used as the base learner in the generation of ensembles, it is obvious that eBagging is the winner. Additionally, while C4.5 is used as the ensemble classifier, eBagging performs well at classifying instances.

Table 1: Comparison table of the applied classifiers in terms of classification accuracies (%)

Decision Tree (C4.5)				SVM			
Method/Dataset	CK+	CK+48	JAFFE	Method/Dataset	CK+	CK+48	JAFFE
eBagging	98.73	91.36	70.05	eBagging	96.87	90.07	65.67
Bagging	95.58	87.80	66.35	Bagging	96.42	87.34	64.05
Single	84.64	78.26	56.55	Single	96.50	77.02	53.56
AdaBoost	90.68	83.31	62.25	AdaBoost	96.80	81.90	60.35

(Continued)

Table 1: Continued

NB				KNN			
Method/Dataset	CK+	CK+48	JAFFE	Method/Dataset	CK+	CK+48	JAFFE
eBagging	95.78	89.12	64.91	eBagging	89.55	84.14	58.05
Bagging	93.18	86.77	62.58	Bagging	87.00	79.60	54.98
Single	82.08	75.28	52.38	Single	74.80	67.60	54.90
AdaBoost	88.68	81.38	60.20	AdaBoost	80.85	72.45	51.07

Table 2: Pairwise comparisons of the applied methods by showing in each cell (wins-ties-loses) between the method in that row and the method in that column for the respective classifier for CK+48 Dataset (20% of samples for testing)

C4.5	eBagging	Single	Bagging	Adaboost
eBagging		184-0-12	160-0-36	131-0-65
Single			37-11-148	36-6-154
Bagging				66-11-119
AdaBoost				
SVM	eBagging	Single	Bagging	Adaboost
eBagging		143-6-47	101-8-87	125-0-71
Single			59-12-125	59-53-84
Bagging				95-12-89
AdaBoost				
NB	eBagging	Single	Bagging	Adaboost
eBagging		142-0-54	148-6-42	89-11-96
Single			65-30-101	60-48-88
Bagging				95-18-83
AdaBoost				
kNN	eBagging	Single	Bagging	Adaboost
eBagging		148-0-48	166-0-30	168-0-28
Single			77-12-107	72-78-46
Bagging				114-6-76
AdaBoost				

The results confirm the research hypothesis that using eBagging classifier based on discriminative features will enhance the classification accuracy. In addition to the results of classification accuracy, it is essential to expand experimental work on the pairwise comparisons of the performed algorithms. Tab. 2 represents the (win-tie-loss) status of the paired algorithms where each cell is read by looking at the algorithm in the relevant row and then in the relevant column. When one of the approaches from C4.5, SVM, or kNN is used as the base learner in the generation of ensembles, it is obvious that eBagging is the winner. Additionally, while NB is used as the ensemble classifier, both eBagging and AdaBoost do well at classifying instances.

The average error rates (averaged over all datasets) derived from pairwise comparisons of the implemented algorithms as shown in Tab. 3. The average error rate estimation can be shown with the following illustration (eBagging vs. single classifier): The ratio of the mean error rate of the eBagging algorithm and a single classifier when C4.5 is implemented is determined for each dataset. Since computing the ratio values for each sample, the mean value of the ratios provides the average error rate for the compared algorithms. In this scenario, baseline =100 indicates that the compared approaches do nearly equally well at classifying instances. The average error rate is less than the baseline in the enhancement scenario, indicating that the first approach outperforms the second implemented algorithm. Apart from this, eBagging significantly improves performance across the majority of comparisons among all base learners. Simultaneously, the standard bagging algorithm is improved by performing eBagging regardless of the base classifier chosen. Although eBagging outperforms AdaBoost in terms of classification precision, when the NB and C4.5 classifiers are used as the ensemble's base classifiers (in around half of the cases), the average enhancement rate falls below the baseline. In this scenario, the average error rates are 115 and 109, respectively, while eBagging is used as an ensemble technique rather than AdaBoost. This is because AdaBoost significantly improves the classification performance of several datasets, while eBagging correctly classifies a considerably larger number of datasets.

Table 3: Average error rates (%) of the compared classifiers in terms of pairwise comparisons

	C4.5	SVM	NB	kNN
eBagging vs. Single classifier	77	84	87	79
Bagging vs. Single classifier	91	100	99	103
AdaBoost vs. Single classifier	80	97	96	103
eBagging vs. Bagging	87	91	93	77
eBagging vs. AdaBoost	115	98	109	77
AdaBoost vs. Bagging	90	87	96	104

Robustness to noise is a beneficial property, since noise in data is often present. We investigated the impact of classification noise on the efficiency of the eBagging technique in this experimental study. To investigate the influence of classification noise, we applied random class noise to the three datasets. To include p % classification noise, p % of the data instances were arbitrarily selected without replacement and their class names were modified to be inaccurate (alternated to class label chosen uniformly from the other labels). The overall classification efficiency of eBagging and bagging techniques at four different noise levels (0%, 4%, 8%, and 10%) is shown in Tab. 4, along with the amount of wins and ties for eBagging and bagging methods. As the noise ratio is raised, the classification accuracies of the above approaches decrease more dramatically. However, as noise levels rise, eBagging retains certain advantages over noise. We may infer from this analysis that eBagging is indeed superior to Bagging in the presence of data noise.

The final series of experiments validated the proposed model's efficiency in comparison to the state-of-the-art models mentioned in Tab. 5 using the CK+ dataset. The findings corroborate the proposed model's dominance. Despite the proposed model's convergence with the 3D Shape-based recognition model's performance, the suggested model is descriptor-independent (geometric descriptor), and it employs a number of translation- and scale-invariant functions. By and large, the 3D Shape descriptor performs poorly while the data collection contains more noise, i.e.,

target groups overlap. Furthermore, employing a deep neural network needs adjusting network configuration parameters that, in turn, need more effort.

Table 4: The number of wins obtained from classification accuracy of eBagging and bagging methods in terms of added classification noise for CK+48 Dataset (20% of samples for testing)

Noise ratio	C4.5			NB			kNN			SVM		
	eBagging	Bagging	Tie	eBagging	Bagging	Tie	eBagging	Bagging	Tie	eBagging	Bagging	Tie
0%	160	36	0	148	42	6	166	30	0	101	87	8
4%	160	36	0	107	89	0	136	60	0	132	58	6
8%	148	48	0	124	72	0	130	66	0	136	60	0
10%	166	30	0	114	82	0	114	82	0	144	52	0

Table 5: Comparisons with different methods on CK+

Method	Accuracy rate
3D shape + GA [49]	97.6
Gabor [50]	93.8
Neural network [51]	94.4
Deep neural network [52]	97.8
The proposed work	98.01

5 Conclusions

Facial expression classification is a very challenging and open area of research. This paper developed a simple yet effective smile classification approach based on a combination of row transform-based features extraction algorithm and eBagging ensemble classifier. Utilizing the row transformation helps to remove some unnecessary coefficients from the extracted features' vector to reduce the computational complexity. By taking a weighted average of the decisions made by the poor learners, eBagging assists in training a highly reliable classifier. The model's objective is to achieve the lowest possible recognition error, the shortest possible run time, and the simplest layout. For various samples, the model achieves a identification accuracy of 98%. Four widely used classifiers, namely SVM, NB, kNN, and C4.5, are used as base classifiers in the laboratory experiments, which were validated using statistical testing. According to the experimental results, eBagging outperforms its competitors by correctly classifying data points while minimizing training error. Additionally, as eBagging is used, the average error rate decreases substantially when compared to single classifiers and standard bagging algorithms, and in half of the scenarios, it results in close results with AdaBoost. As a result, the proposed model based on the eBagging classifier has a high likelihood of being applicable to classifying facial expression samples. Additionally, the tests demonstrate that eBagging outperforms Bagging through three datasets, as long as the data contains minimal or no noise. The proposed model is characterized by simplicity in implementation, in contrast to the deep learning-based classification methods that depend on

adjusting multiple variables to achieve reliable accuracy. On the other hand, the limitation of this work appeared in JAFFE dataset because of the insufficient number of samples. In the future, a mobile application shall be created to find expressions in each video frame automatically. Furthermore, speech detection includes both audios from a speaker tone and video responses can further improve detection accuracy.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

- [1] R. Andrian and S. Supangkat, "Comparative analysis of deep convolutional neural networks architecture in facial expression recognition: A survey," in *Proc. of the Int. Conf. on ICT for Smart Society*, Bandung, Indonesia, pp. 1–6, 2020.
- [2] M. Nestor, D. Fischer and D. Arnold, "Masking our emotions: Botulinum toxin, facial expression, and well-being in the age of COVID-19," *Journal of Cosmetic Dermatology*, vol. 19, no. 1, pp. 2154–2160, 2020.
- [3] Z. Geng, C. Cao and S. Tulyakov, "Towards photo-realistic facial expression manipulation," *International Journal of Computer Vision*, vol. 128, no. 10, pp. 44–2761, 2020.
- [4] F. Zhang, T. Zhang, Q. Mao and C. Xu, "Geometry guided pose-invariant facial expression recognition," *IEEE Transactions on Image Processing*, vol. 29, no. 1, pp. 4445–4460, 2020.
- [5] I. Gogić, M. Manhart, I. S. Pandžić and J. Ahlberg, "Fast facial expression recognition using local binary features and shallow neural networks," *Visual Computer*, vol. 36, no. 1, pp. 97–112, 2020.
- [6] S. Escalera, E. Puertas, P. Radeva and O. Pujol, "Multi-modal laughter recognition in video conversations," in *Proc. of the IEEE Conf. on Computer Vision*, Miami, FL, USA, pp. 110–115, 2009.
- [7] B. Gong, Y. Wang, J. Liu and X. Tang, "Automatic facial expression recognition on a single 3D face by exploring shape deformation," in *Proc. of the 17th ACM Int. Conf. on Multimedia*, Beijing, China, pp. 569–572, 2009.
- [8] S. F. Cotter, "Mobiexpressnet: A deep learning network for face expression recognition on smart phones," in *Proc. of the IEEE Int. Conf. on Consumer Electronics*, Las Vegas, NV, USA, pp. 1–4, 2020.
- [9] S. Law, C. I. Seresinhe, Y. Shen and M. Gutierrez-Roig, "Street-frontage-net: Urban image classification using deep convolutional neural networks," *International Journal of Geographical Information Science*, vol. 34, no. 1, pp. 681–707, 2020.
- [10] A. Alarifi, A. Tolba, Z. Al-Makhadmeh and W. Said, "A big data approach to sentiment analysis using greedy feature selection with cat swarm optimization-based long short-term memory neural networks," *Journal of Supercomputing*, vol. 76, no. 6, pp. 4414–4429, 2020.
- [11] A. M. Ashir, A. Eleyan and B. Akdemir, "Facial expression recognition with dynamic cascaded classifier," *Neural Computing and Applications*, vol. 32, pp. 6295–6309, 2020.
- [12] M. J. Cossetin, J. C. Nievola and A. L. Koerich, "Facial expression recognition using a pairwise feature selection and classification approach," in *Proc. of the Int. Joint Conf. on Neural Networks*, Vancouver, BC, Canada, pp. 5149–5155, 2016.
- [13] O. A. Hassen, N. A. Abu, Z. Abidin and S. M. Darwish, "A new descriptor for smile classification based on cascade classifier in unconstrained scenarios," *Symmetry*, vol. 13, no. 5, pp. 1–16, 2021.
- [14] S. L. Happy and A. Routray, "Automatic facial expression recognition using features of salient facial patches," *IEEE Transactions on Affective Computing*, vol. 6, pp. 1–12, 2015.
- [15] K. Mistry, L. Zhang, S. C. Neoh, C. P. Lim and B. Fielding, "A micro-gA embedded PSO feature selection approach to intelligent facial emotion recognition," *IEEE Transactions on Cybernetics*, vol. 47, pp. 1496–1509, 2017.

- [16] A. Samara, L. Galway, R. Bond and H. Wang, "Affective state detection via facial expression analysis within a human-computer interaction context," *Journal of Ambient Intelligence and Humanized Computing*, vol. 10, pp. 2175–2184, 2019.
- [17] S. Mitra and T. Acharya, "Gesture recognition: A survey," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 37, pp. 311–324, 2007.
- [18] O. Hassen, N. Abu and Z. Abidin, "Human identification system: A review," *International Journal of Computing and Business Research*, vol. 9, pp. 1–26, 2019.
- [19] I. Buciu, C. kotropoulos and I. Pitas, "ICA and gabor representation for facial expression recognition," in *Proc. of the Int. Conf. on Image Processing*, Barcelona, Spain, pp. II–855, 2003.
- [20] M. Minsky and S. A. Papert, *Perceptrons: An Introduction to Computational Geometry*, Cambridge, MA, USA: MIT press, 2017.
- [21] I. Goodfellow, Y. Bengio, A. Courville and Y. Bengio, *Deep Learning*, vol. 1. Cambridge, MA, USA: MIT Press, 2016.
- [22] H. Sikkandar and R. Thiyagarajan, "Soft biometrics-based face image retrieval using improved grey wolf optimization," *IET Image Processing*, vol. 14, no. 1, pp. 451–461, 2020.
- [23] M. Woźniak and D. Połap, "Bio-inspired methods modeled for respiratory disease detection from medical images," *Swarm and Evolutionary Computation*, vol. 41, pp. 69–96, 2018.
- [24] M. Woźniak and D. Połap, "Adaptive neuro-heuristic hybrid model for fruit peel defects detection," *Neural Networks*, vol. 98, pp. 16–33, 2018.
- [25] H. R. Kanan, K. Faez and M. Hosseinzadeh, "Face recognition system using ant colony optimization-based selected features," in *Proc. of the IEEE Symposium on Computational Intelligence in Security and Defense Applications*, Honolulu, HI, USA, pp. 57–62, 2007.
- [26] N. Karaboga, "A new design method based on artificial bee colony algorithm for digital IIR filters," *Journal of the Franklin Institute*, vol. 346, pp. 328–348, 2009.
- [27] J. I. Ababneh and M. H. Bataineh, "Linear phase FIR filter design using particle swarm optimization and genetic algorithms," *Digital Signal Processing*, vol. 18, pp. 657–668, 2008.
- [28] S. C. Chu and P. W. Tsai, "Computational intelligence based on the behavior of cats," *International Journal of Innovative Computing, Information and Control*, vol. 3, no. 1, pp. 163–173, 2007.
- [29] M. Aziz and A. Ewees and A. E. Hassanien, "Multi-objective whale optimization algorithm for content-based image retrieval," *Multimedia Tools and Applications*, vol. 77, pp. 26135–26172, 2018.
- [30] R. A. Ibrahim, M. A. Elaziz and S. Lu, "Chaotic opposition-based grey-wolf optimization algorithm based on differential evolution and disruption operator for global optimization," *Expert Systems with Applications*, vol. 108, pp. 1–27, 2018.
- [31] M. Aziz and A. E. Hassanien, "Modified cuckoo search algorithm with rough sets for feature selection," *Neural Computing and Applications*, vol. 29, pp. 925–934, 2018.
- [32] M. Aziz, A. Ewees and A. E. Hassanien, "Multi-objective whale optimization algorithm for content-based image retrieval," *Multimedia Tools and Applications*, vol. 77, pp. 26135–26172, 2018.
- [33] L. B. Krithika and G. L. Priya, "Graph based feature extraction and hybrid classification approach for facial expression recognition," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, no. 2, pp. 2131–2147, 2021.
- [34] X. Fan and T. Tjahjadi, "A dynamic framework based on local zernike moment and motion history image for facial expression recognition," *Pattern Recognition*, vol. 64, pp. 399–406, 2017.
- [35] M. Roomi, A. Naghasundharam, S. SathishKumar and R. Sugavanam, "Emotion recognition from facial expression-a target oriented approach using neural network," in *Proc. of the Indian Conf. on Computer Vision, Graphics and Image Processing*, Kolkata, India, pp. 1–4, 2004.
- [36] C. Fuentes, V. Herskovic, I. Rodríguez, C. Gereá, M. Marques *et al.*, "A systematic literature review about technologies for self-reporting emotional information," *Journal of Ambient Intelligence and Humanized Computing*, vol. 8, pp. 593–606, 2017.
- [37] B. Langeroodi and K. Kojouri, "Automatic facial expression recognition using neural network," in *Proc. of the Int. Conf. on Image Processing, Computer Vision, and Pattern Recognition*, Las Vegas, NV, USA, pp. 1–5, 2011.

- [38] A. Mollahosseini, D. Chan and M. Mahoor, "Going deeper in facial expression recognition using deep neural networks," in *Proc. of the IEEE Winter Conf. on Applications of Computer Vision*, Lake Placid, NY, USA, pp. 1–10, 2016.
- [39] R. Walecki, O. Rudovic, V. Pavlovic, B. Schuller and M. Pantic, "Deep structured learning for facial expression intensity estimation," *Image and Vision Computing*, vol. 259, pp. 143–154, 2017.
- [40] G. Yolcu, I. Oztel, S. Kazan, C. Oz and F. Bunyak, "Deep learning-based face analysis system for monitoring customer interest," *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, pp. 237–248, 2020.
- [41] K. Zhao, W.-S. Chu and H. Zhang, "Deep region and multi-label learning for facial action unit detection," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, Las Vegas, NV, USA, pp. 3391–3399, 2016.
- [42] P. Viola and M. Jones, "Robust real-time object detection," *International Journal of Computer Vision*, vol. 4, no. 1, pp. 34–47, 2001.
- [43] C. Silva, L. Schnitman and L. Oliveira, "Detection of facial landmarks using local-based information," in *Proc. of the 19th Brazilian Conf. on Automation*, Campina Grande, Brazil, pp. 1–5, 2012.
- [44] I. Garali, M. Adel, S. Bourennane and E. Guedj, "Histogram-based features selection and volume of interest ranking for brain PET image classification," *IEEE Journal of Translational Engineering in Health and Medicine*, vol. 6, pp. 1–12, 2018.
- [45] G. Tuysuzoglu and D. Birant, "Enhanced bagging (eBagging): A novel approach for ensemble learning," *International Arab Journal of Information Technology*, vol. 17, no. 4, pp. 515–528, 2020.
- [46] Y. Lu and Q. Liu, "Lip segmentation using automatic selected initial contours based on localized active contour model," *EURASIP Journal on Image and Video Processing*, vol. 7, pp. 1–12, 2018.
- [47] T. M. Khoshgoftaar, J. V. Hulse and A. Napolitano, "Comparing boosting and bagging techniques with noisy and imbalanced data," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 41, pp. 552–568, 2011.
- [48] P. Hieu and S. Olafsson "Bagged ensembles with tunable parameters," *Computational Intelligence*, vol. 35, no. 1, pp. 184–203, 2019.
- [49] L. Jeni, A. Lőrincz, Z. Szabó, J. Cohn and T. Kanade, "Spatio-temporal event classification using time-series kernel based structured sparsity," in *Proc. of the European Conf. on Computer Vision*, Zurich, Switzerland, pp. 135–150, 2014.
- [50] M. Mahoor, M. Zhou, K. Veon, S. Mavadati and J. Cohn, "Facial action unit recognition with sparse representation," in *Proc. of the Automatic Face Gesture Recognition and Workshops*, Santa Barbara, CA, USA, pp. 336–342, 2011.
- [51] B. Langeroodi and K. Kojouri, "Automatic facial expression recognition using neural network," in *Proc. of the Int. Conf. on Image Processing, Computer Vision, and Pattern Recognition*, pp. 1–5, 2011.
- [52] A. Mollahosseini, D. Chan and M. Mahoor, "Going deeper in facial expression recognition using deep neural networks," in *Proc. of the IEEE Winter Conf. on Applications of Computer Vision*, Lake Placid, NY, USA, pp. 1–10, 2016.