Journal of Information Technology Research
Volume 15 • Issue 1

Deep Stacked Autoencoder-Based Automatic Emotion Recognition Using an Efficient Hybrid Local Texture Descriptor

Shanthi Pitchaiyan, National Institute of Technology, Tiruchirappalli, India
Nickolas Savarimuthu, National Institute of Technology, Tiruchirappalli, India
https://orcid.org/0000-0002-0703-3839

ABSTRACT

Extracting an effective facial feature representation is the critical task for an automatic expression recognition system. Local binary pattern (LBP) is known to be a popular texture feature for facial expression recognition. However, only a few approaches utilize the relationship between local neighborhood pixels itself. This paper presents a hybrid local texture descriptor (HLTD) that is derived from the logical fusion of local neighborhood XNOR patterns (LNXP) and LBP to investigate the potential of positional pixel relationship in automatic emotion recognition. The LNXP encodes texture information based on two nearest vertical and/or horizontal neighboring pixel of the current pixel whereas LBP encodes the center pixel relationship of the neighboring pixel. After logical feature fusion, the deep stacked autoencoder (DSA) is established on the CK+, MMI, and KDEF-dyn dataset, and the results show that the proposed HLTD-based approach outperforms many of the state-of-the-art methods with an average recognition rate of 97.5% for CK+, 94.1% for MMI, and 88.5% for KDEF.

KEYWORDS

Deep Neural Network, Emotion, Facial Expression, Feature Fusion, Hybrid Local Texture Descriptor, Stacked Autoencoder

INTRODUCTION

Emotions are the automatic process of the brain directly caused by events in the environment. Emotions are complicated to define but share some information with others. Emotions can be expressed in different shapes, such as speech, facial expression, and actions. In human interaction, the intuition of facial expressions creates a communication channel along with voice, which conveys primary evidence about the internal emotional state of the person in conversation. Facial expression is caused by a coordinated pattern of muscle movements, triggered by a particular brain area. Facial expression understanding via the machine can modernize user interfaces such as robotics, car driving, etc.,. In the last few decades, many methods were proposed for Automatic Facial Expression Recognition (AFER) towards feature extraction and recognition. Nevertheless, AFER is still a challenging task with high accuracy because of the minor interpersonal variation along with the substantial intra-personal distinctions arising from illumination, posture, expression, and other aspects.

DOI: 10.4018/JITR.2022010103

In general, the AFER system contains three primary stages, namely preprocessing, features extraction, and classification. The essential stage for AAFER is to extract standard features from the given image to effectively discriminate different emotions. Facial expressions involve changes in local texture. Appearance feature-based approaches are good at capturing transient features, and it can be extracted from the entire face or the specific facial regions. The region-based feature extraction omits a useful correlation among different features and increases the feature space so that computational cost is high with the risk of overfitting. The most widely used appearance-based methods are Gabor wavelets(Bartlett et al., 2005), Local Binary Pattern (LBP) (Huang, Wang, & Ying, 2010), and its variants. Among these methods, LBP is studied extensively in the application of AFER. This type of code generation method considers the relationship between center and neighboring pixels but neglecting the relationship among the neighboring pixels. Due to this fact, these methods are less discriminative to distinguish different textures as they might be insufficient to define the local substructure feature accurately. Moreover, the presence of intensity variations and noise in the local region may produce false feature codes.

This paper introduces a new intensity variation based hybrid local texture descriptor to overcome the limitations as mentioned above and to increase the discriminative capability of feature pattern for AFER. The proposed hybrid local texture feature descriptor encodes the sectional shape by mean of the relationship of neighboring pixels with the center pixel as well as the relationship among neighboring pixels. This type of feature fusion excludes featureless variations in the local region and also reduces the feature space. In recent years, deep learning approaches perform better in AFER and become more popular in the computer vision community (Chen et al., 2018; Jain, S. Kumar, A. Kumar, Shamsolmoali, &Zareapoor, 2018; Xie, & Hu, 2019). However, it requires extensive training data for proper training of deep network model. Due to the unavailability of sufficient data in the current expression dataset, most of the deep learning methods have employed image augmentation at the cost of high computational power (Christou, & Kanojiya, 2019;D. Liang, H. Liang, Yu,& Zhang, 2019; Nguyen et al., 2019). Without such augmented data, the effectiveness of the proposed hybrid feature is evaluated using a deep-stacked autoencoder. Deep Stacked autoencoder (DSA) is one of the deep learning models that has been broadly applied to various applications (Chen et al., 2018; Zeng et al.,2018). The best part of DSA is that it can extract useful, reliable, and specific characteristics of features after identifying and eliminating redundant features in an unsupervised fashion.

Further, an AFER is a challenging task due to the inter-person and intra-expression variations. The external factors such as noise, illumination, and environmental change are also raising the complexity of the AFER. Here, a DSA-based deep learning model is designed to improve accuracy by learning valuable hybrid features. The proposed person-independent facial expression recognition system shows better performances using DSA with and without noise. The highlights of the proposed work are as follows:

- A hybrid local texture descriptor (HLTD is introduced to integrate the gains of two local features namely LBP and LNXP.
- The higher-level feature mapping by stacked autoencoder compress the feature space further which in turn reduce the computational cost and improve the accuracy.
- The outcome of the experiments using a deep stacked autoencoder framework on a diverse subset of features proves the effectiveness of the proposed system.
- The presented DSA-based approach to recognize facial expressions under a noise environment effectively.

The rest of the paper is structured as follows. The related works section briefly reviews the related works and the subsequent section presents the proposed methodology. The result and discussion section report the experimental outcome on the CK+, MMI, and KDEF-dyn datasets. The conclusion section gives the conclusion of the proposed approach and provides scope for further study.

RELATED WORKS

Many image classification algorithms are more specific in determining, understanding, describing, and refining useful patterns or features. In a feature-oriented approach, facial expression information is obtained in terms of geometric or appearance features (Akputu, Seng, Lee,&Ang, 2018; Barman,& Dutta, 2019; Majumder, Behera, & Subramanian, 2016). The geometric method uses the absolute and relative geometrical relationship of facial components for feature extraction. Two well-known model-based facial feature point detection methods are an Active Shape Model (ASM) (Cootes, Taylor, Cooper, & Graham,1995; Shbib, & Zhou, 2015) and an Active Appearance Model (AAM) (Edwards, Cootes, & Taylor, 1998; Maximiano da Silva, &Pedrini, 2016). The main drawback of the statistical model is the model fitting. An automatic feature point detection method is hard to implement in the real-time data due to the variations in illumination and head pose change. The expression difference among different persons varies in the same database. The difference is even more significant for different databases. In such cases, developing a generalized geometrical model for AFER is a challenging task. Furthermore, geometric feature-based methods ignore skin texture change, which is essential for facial expression modeling to capture micro muscle movements.

In appearance feature-based approach, features express the facial appearance in terms of edges, corners, statistical features, etc., Among the various appearance-based approach, Principal Component Analysis (PCA) (Turk, & Pentland, 1991),Histograms Oriented Gradients feature (HOG) (Nazir, Jan, & Sajjad, 2018), Independent Component Analysis (ICA) (Guo, Zhang, Deng, & Wei, 2013),Scale Invariant Feature Transformation (SIFT) (Kalsum, Anwar, Majid, Khan, & Ali, 2018), Local Directional Number Pattern (LDN) (Rivera, Castillo, & Chae, 2015), are the most popular features extraction methods. The multi-scale multi-resolution gabor filter is used widely with good recognition accuracy, but it is computationally intensive. The conventional LBP has been used as a useful texture feature in face analysis (Shan, Gong, & McOwan, 2009; Tan, & Triggs, 2007), which preserve the facial information effectively. Recently, Illumination invariant merged binary pattern coding proposed by (Munir, Hussain, Khan, Nadeem, & Arshid, 2018) along with PCA, extract the subcode pattern based on the position of neighboring pixel and the final combined feature set proves that full-face analysis performs better than the zone-based recognition in the real-time dataset. Similarly, the Histogram Oriented Gradient (HOG) with Graph Signal Processing (GSP) reduces the redundant feature, but there is a trade-off between the feature size and initial computation cost (Meena, Joshi, & Sharma, 2018). Recently the feature fusion of HOG and OC-LBP is applied successfully in face recognition (Singh & Chhabra, 2018), and it proves that the handcrafted feature fusions are more suitable for low computational devices. So, an efficient handcrafted feature formally considers the right trade-off between computational efficiency and accuracy.

Usually, texture contains numerous types of patterns. The combination of many patterns can provide effective texture representation since they collectively represent the information in a different aspect. Therefore, to enhance the discriminant power of the facial feature in AFER, multiple features from different extraction methods can be used together. For example, a hybrid approach of PCA and LBP is introduced in (Luo, Wu, & Zhang,2013) to combine the local and global grayscale features of facial expression. In the Serial Feature Fusion strategy concatenate the histogram of different featured images based on the union-vector (Yang, Ban, Li, & Yang, 2018). The combination of SIFT and SURF features introduced in (Kalsum et al., 2018) uses a spatial bag to generate a fixed-length feature for all samples. But, the multiple feature fusion techniques are computationally intensive when the number of features increases. To reduce the computational cost without losing informative texture details, the Gradient Local Ternary pattern proposed in (Holder, & Tapamo, 2017), combine the histogram of positive and negative gradient difference and using PCA the feature space dimension is reduced. The Gabor and LDN based feature descriptor proposed in (Zhang et al., 2018) adopt the mean pooling technology for feature concatenation and reduction. Recently, the Local Prominent Directional Pattern (LPDP) is proposed to differentiate various types of textures using the statistics

of the weighted distribution of neighboring pixels directions. The discriminative capability of LPDP is reduced due to the less difference among expressions of the elder subjects.

In recent years, Deep Neural Network (DNN) (Sharan, & Moir, 2017; Zhang et al., 2016) has extended attention in artificial intelligence and machine learning, and many related algorithms have been applied in image recognition problems effectively (Shi, & Pun, 2017; Y. Wang, X. Wang, & Liu, 2016). Instead of using shallow learning model (Al Rahhal et al., 2016) with single layer nonlinear transformation, a DNN based model extract and learn the high-level features from the data directly. These higher-level feature representations can describe the semantics of the data and increase the robustness of intra-class variability. But significant computational power is required to train the model, and additional effort is needed to preprocess the data. Some existing methods combine or fuse different features to produce a complete representation feature, and the classification using a deep neural network improves the recognition accuracy than the single feature alone.

In (Majumder et al.,2016), the regional LBP, as well as geometric features, are used for the feature fusion using three-level of the autoencoder. The first two autoencoders are used to adjust the feature size of different natured features and the final autoencoder used for classification. The HOG and PCA feature-based deep learning techniques proposed by (Zeng et al., 2018) utilize the handcrafted features for 7 and 8 class expressions. In general, the model gets confused when there is a similarity that exists in the shape and appearance of features for the same expression. Another feature fusion approach (Fan, & Tjahjadi, 2019) combines handcrafted features (PHOG and shape feature) with dynamic deep features to improve recognition accuracy. The unavailability of sufficient data to train a deep neural network is a significant issue. So that most of the deep learning methods have employed image augmentation at the cost of high computational power. From the above study, it is proved that the feature fusion technique enhances the discriminative information and improves the overall performance of the system.

Most of the existing deep learning models focus on extracting higher-level feature mapping. But disregard the lower level features extracted from the local facial regions. This work mainly focuses on exploiting the potential of features extracted based on local neighborhood relationships and build a better performing deep neural network model using Stacked Autoencoder. Unlike the previous approach, local feature extraction is performed independently from the deep neural network to highlight the significance of local features in expression analysis. Here, three types of handcrafted features are considered for analysis, which includes LBP, LNXP, and the Hybrid Local Texture Descriptor (HLTD). The deep neural network exploits the reduced features supplied by the last layer of the DSA and trains a softmax classifier, which implements the transfer learning approach. The conventional neural network algorithms are always liable to the gradient diffusion and local maximum, which in turn provide a reduced recognition rate. Thus, DSA learns the data with initial weight in an unsupervised manner, and during the backpropagation, weights are updated based on the error to improve accuracy. The contributions of this approach are twofold. The first one is to verify the capability of the closest neighboring pixel relationship that is neglected by the other encoding schemes in AFER. The new hybrid local texture descriptor is introduced by logically fusing LBP and LNXP. The second contribution is the evaluation of the hybrid feature using deep-stacked autoencoder to improve the accuracy of the proposed system.

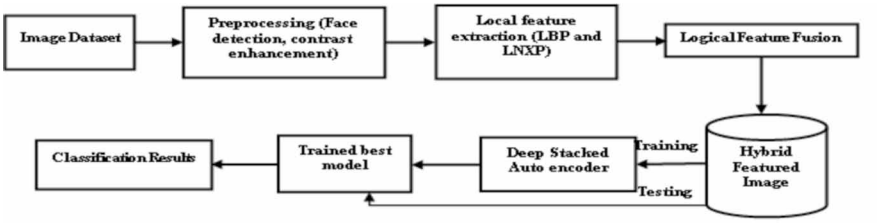
THE HYBRID LOCAL TEXTURE DESCRIPTOR (HLTD) BASED AFER

The proposed AFER system involves the following steps: Feature extraction, data fusion, deep stacked autoencoder based classifier. Figure1 shows the different phases involved in the proposed approach.

Automatic Extraction of Local Texture Descriptor

LBP is one of the most dominant texture descriptors since its introduction. It is proposed by Ojala, Pietikäinen, & Mäenpää, (2002) with the two local underlying complementary assumptions that

Figure 1. Phases of the proposed system



texture has pattern and strength. The local substructure size is decided based on P and R where P is a number of the neighboring pixels located at the radius of R. For the given a grayscale image with the intensity value $I_i = I\left(x_i, y_i\right)$ at the location x_p, y_p, the LBP operator encodes the 3×3 local substructure of size $R = 1$ and $P = 8$. The pixel value at the pth sampling point I_p is given by Equation 1:

$$I_i = I\left(x_i, y_i\right), i = 0,1,\cdots,P-1 \tag{1}$$

The binary operator $S\left(I_i, I_c\right)$ of the LBP encoding can be given by Equation 2 and 3:

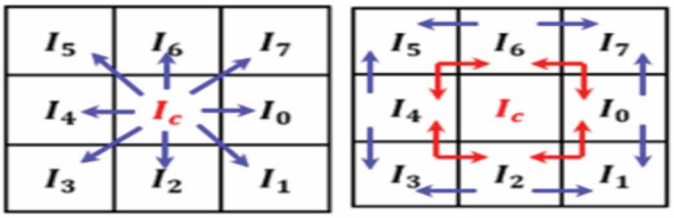
$$S\left(I_i - I_c\right) = \begin{cases} 1 & I_i \geq I_c \\ 0 & otherwise \end{cases}; where\, i = 0,1\ldots,P-1 \tag{2}$$

$$LBP_P^R\left(I_i, I_c\right) = S\left(I_i - I_c\right)2^i + S\left(I_{i-1} - I_c\right)2^{i-1} + \cdots + S\left(I_0 - I_c\right)2^0 \tag{3}$$

Then the LBP_P^R features are calculated by taking the corresponding decimal equivalent. According to (Ojala et al., 2002), a subset of the basic LBP pattern is adequate to represent texture details of an image. This subset pattern is said to be a uniform LBP pattern if it contains almost two transitions, 0 to 1 or 1 to 0 (Topi, Timo, Matti, & Maricor, 2000). The uniform LBP encodes the presence of micropatterns only, and the feature dimension is 59. However, it lost 20% of the texture details, which leads to higher false expression recognition (Zhang et al.,2016). The region-based uniform LBP feature extraction increases the feature dimension when all the sub-region feature vectors are concatenated, but it did not precisely represent the expression uniqueness when the number of blocks is small. Here, the LBP pattern is extracted from the whole face to retain the shape information of the facial region.

The conventional LBP only represents the center pixel relationship with the neighboring pixel of the selected window. The contribution of the relationship among the closest adjacent pixel in texture description is analyzed differently in the proposed approach. This paper employed a new local texture descriptor called Local Neighborhood XNOR pattern, which represents the relationship among closest neighboring pixels with the current pixel. Figure 2 shows the encoding difference between conventional LBP and LNXP. Like LBP, from the sub-structure, an 8-bit binary code is calculated using the current pixel as the threshold. The two closest neighboring pixels are located vertical and/or horizontal to the i^{th} sampling point. The neighboring pixel direction is different for the corner pixels such as I_1, I_3, I_5 and I_7. For the remaining pixels, neighboring pixels will be in the same

Figure 2. LBP and LNXP neighboring pixel



directions that are either in the vertical or horizontal direction. Its positions from the i^{th} sampling point is represented as $(i+1) mod P$ and $(P-1+i) mod P$. For example, the neighboring pixel position for I_0 is $I_{(0+1) mod 8} = I_1$ and $I_{(8-1+0) mod 8} = I_7$. The identified closest neighboring pixels are equated against I_i, and the outcome of the binary operator $S\left(I_{(i+1) mod P}, I_i\right)$ and $S\left(I_{(P-1+i) mod P}, I_i\right)$ is represented by Equation4 and 5:

$$S\left(I_{(i+1) mod P} - I_i\right) = \begin{cases} 1 & if\ I_{(i+1) mod P} \geq I_i \\ 0 & otherwise \end{cases} \tag{4}$$

$$S(I_{P-1+i) mod P} - I_i = \begin{cases} 1 & if\ I_{(P-1+i) mod P} \geq I_i \\ 0 & otherwise \end{cases} \tag{5}$$

where i=0,1,2. . ., P-1.

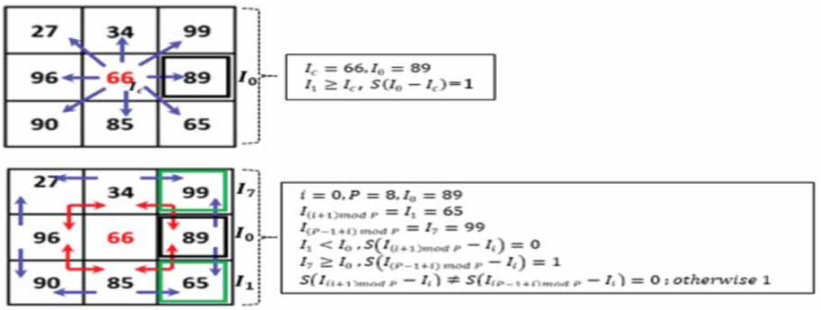
The bitwise XNOR operation is performed between the outcome of Equation 4 and 5 in order to retain the useful information. This binary operation. The eight-bit LNXP is calculated for the selected window using the following Equation 6 and 7:

$$\begin{aligned} & S\left(I_{(i+1) mod P} - I_i\right) \odot S\left(I_{(P-1+i) mod P} - I_i\right) \\ & = \begin{cases} 1 & if\ S\left(I_{(i+1) mod P} - I_i\right) == S\left(I_{P-1+i) mod P} - I_i\right) \\ 0 & otherwise \end{cases} \end{aligned} \tag{6}$$

$$\begin{aligned} & LNXP_P^R\left(I_{(i+1) mod P}, I_i, I_{(P-1+i) mod P}\right) = \left[S\left(I_{(i+1) mod P} - I_i\right) \odot S\left(I_{(P-1+i) mod P} - I_i\right)\right] 2^i \\ & + \left[\left(I_{i\ mod P} - I_{i-1}\right) \odot S\left(I_{(P-2+i) mod P} - I_{i-1}\right)\right] 2^{i-1} + \ldots + \left[\left(I_1 - I_0\right) \odot S\left(I_{P-1} - I_0\right)\right] 2^0 \end{aligned} \tag{7}$$

After encoding, the binary code is converted into its corresponding decimal value of the binary operator $S\left(I_{(i+1) mod P}, I_i, I_{P-1+i) mod P}\right)$ using Equation 7. Figure 3 shows the LBP and LNXP code generation.

Figure 3. LBP and LNXP code generation



Feature Fusion

The texture is the visual information of the surface and reveals the real deviations upon a surface, and it is distinct as a specific group of the various patterns. As stated early, LBP is robust in representing simple features, such as edges, corners, and spots. Thus, LBP can be viewed as a substructure pattern code to provide a specific description of the given surface. The discrimination power of the feature can be improved when the combination of many patterns represents the texture. With this intuition, a hybrid approach is proposed using Hybrid Local Texture Descriptor (HLTD) for AFER. Usually, feature fusion approaches concatenate different types of features together. Such a method is simple, but the resultant feature may not perform better (Fu, Cao, Guo, & Huang,2008). The cause of such performance is an uneven dimension of different natured features, especially with the geometric and appearance feature.

Algorithm 1. Hybrid texture feature extraction

Input: Facial Image $I_{W \times H}$, where W and H are the width and height of an image, respectively. From each local patch, neighboring pixels P =8 is identified at the radius R=1.

Output: Hybrid Local Texture Descriptor (HLTD)

Step 1: For each image in the given dataset do
Step 2: Perform face detection
Step 3: Adjust the contrast using Equation.1
Step 4: Downscale the facial image to the specified size
For w= 2 to W-1 **do**
For h= 2 to H-1 **do**
While do
Identify local neighborhood $I_i = I\left(w \pm 1, h \pm 1\right)$
If $I_i \geq I_c$ **then**
=
End If
Ifthen
=
Else If then
=
End If
End While
End For
End For
Step 5: Return

A simple concatenation feature fusion results in poor recognition accuracy due to the significant unbalanced dimension of various features. In the proposed feature fusion method, both local feature dimensions are the same, so that logical feature fusion is carried out with the expectation to reduce the size and improve the recognition accuracy. This logical feature fusion technique eliminates irrelevant information while retaining the shape information, which is common for both features. This feature level fusion strategy is based on the logical operator. Suppose LNXP and LBP are two feature spaces defined on the given dataset. For the random sample, the two feature vectors LNXP and LBP are logically fused using bitwise AND by Equation 8:

$$\gamma = \left(\alpha \cdot \beta\right) \tag{8}$$

The logically combined feature is n-dimension provided and is also n-dimension. All logically fused feature vectors of all samples define the overall feature space. It is mathematically represented as follows:

$$HLTD_P^R = LBP_P^R\left(I_i, I_c\right) \cdot LNXP_P^R\left(I_{(i+1) mod\, P}, I_i, I_{(P-1+i) mod\, P}\right)$$

$$HLTD_P^R = \sum_{p=0}^{P-1}\left[s\left(I_i - I_c\right) \cdot \left(s\left(I_{(i+1) mod\, P} - I_i\right) \odot s\left(I_{(P-1+i) mod\, P} - I_i\right)\right)\right] 2^i \tag{9}$$

$$HLTD_P^R = \begin{cases} 1 & if\, s\left(LBP_P^R\left(I_i, I_c\right)\right) and\, s\left(LNXP_P^R\left(I_{(i+1) mod\, P}, I_i, I_{(P-1+i) mod\, P}\right)\right) = 1 \\ 0 & otherwise \end{cases} \tag{10}$$

For the given LBP=11000011 and LNXP =11111101, the HLTD is calculated as 11000001 (logical AND of LBP and LNXP). The following Algorithm 1 explains the steps of feature extraction and fusion.

Feature Analysis

The discriminant power of the proposed hybrid feature is shown in Figure 4. The evaluation is made between the LNXP, LBP, and the hybrid feature. The proposed encoding scheme provides a better description of a different structure than the LBP and LNXP encoding scheme and is less sensitive to dynamic contrast change. The LBP, LNXP, and hybrid encoding of 3 different local substructures are shown in Figure 4. The patches (a) and (b) have diverse structures, whereas the patches (b) and (c) have related structures but changed in contrast. The LBP generate the same code for the two different structures, whereas LNXP and the hybrid feature represent the structural difference between (a) and (b) with two different code. Also, the hybrid coding scheme shows better discriminant capability for the low contrast patch than the LBP. The noise sensitivity analysis is verified by the example shown in Figure 5(a) and 5(b). The encoding result of the given local patch is described with a null or minimum level of difference by LNXP. This example exhibits that LNXP is more stable under noise than LBP.

Further, a non-parametric statistical test is carried out to avoid possible erroneous assumptions about feature distribution under a noise environment. For this purpose, zero-mean Gaussian noise is added to the set of images randomly collected from the working dataset, and the noise variances vary from 1 to 10 to generate the noisy images at each noise variance level. Now the three types of features are extracted from the noisy images. The dissimilarity of histograms obtained from noisy and the corresponding noiseless featured images for each level of noise

Figure 4. The encoding process of three different local substructure using three features

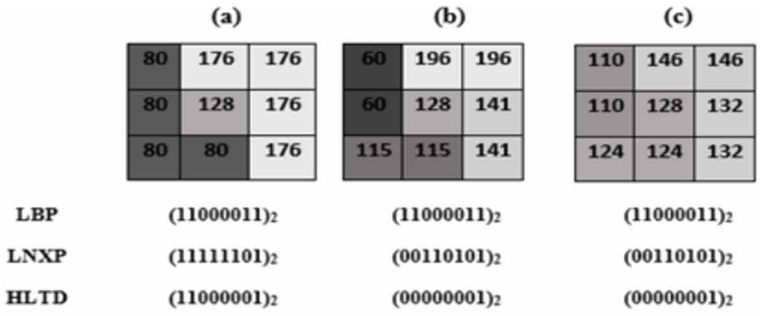
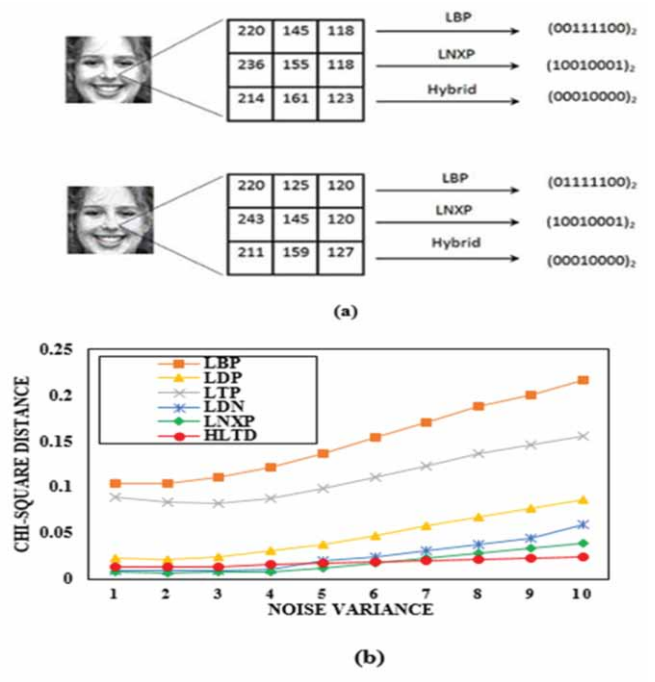


Figure 5. Feature analysis under noise (a) Local patch encoding with noise (b) Histogram distance between noisy and noiseless images at different noise level variance



variance is calculated using chi-square distance Equation11. Chi-squared χ^2 distance is used widely in histogram-based feature comparison from facial images data. The χ^2 distance for two (histogram) vectors $\vec{h}_i$ and $\vec{h}_b^{'}$ is defined as:

$$\chi^2\left(\vec{h}_i,\vec{h}_j\right)=\sum_{b=1}^{D}\frac{\left(\left(\vec{h}_i\right)_b-\left(\vec{h}_i\right)_b^{'}\right)^2}{\left(\vec{h}_i\right)_b+\left(\vec{h}_i\right)_b^{'}} \tag{11}$$

where $\vec{h}_i$ and $\vec{h}_i^{'}$ are the histogram of the noisy and noiseless image and b denotes the b^{th} in of $\vec{h}_i$ histogram. Equation12 can be rewritten in a matrix-based formulation as:

$$\chi^2\left(\vec{h}_i,\vec{h}_j\right)=\left(\sqrt{\left(diag\left(\vec{h}_i+\vec{h}_j\right)\right)^{-1}}\left(\vec{h}_i-\vec{h}_j\right)\right)^T\left(\sqrt{\left(diag\left(\vec{h}_i+\vec{h}_j\right)\right)^{-1}}\left(\vec{h}_i-\vec{h}_j\right)\right) \tag{12}$$

where $\sqrt{\cdot}$ denotes the element-wise square root of a matrix.

The dissimilarity measure is taken for a different feature, and its average is shown in Figure5(b) for each noise variance. From the result, it is observed that the divergence for the hybrid feature is higher than LNXP and lower than LBP up to the particular level of noise variance, and the dissimilarity is not much increased further even with the higher noise variance. However, the LBP and LNXP dissimilarity distance gradually increased when the noise variance level increased. Since the hybrid feature integrates advantages of LBP and LNXP, it tolerates small changes in the local region and in that way, ensures consistent codes. This type of feature analysis proves that local substructure change due to the additive noise will not disturb the discriminative capability of the hybrid feature.

Deep Stacked Sparse Autoencoder

This paper also demonstrates how the deep neural networks are used for dimensionality reduction and recover the spatial information by employing a sequence of the sparse autoencoder. A set of sparse autoencoders are stacked to design the deep neural network, and it will learn the higher-level features in an unsupervised manner, and for the classification, the softmax classifier is used. During the backpropagation, internal parameters are updated and feedback to the entire network. Conventional Deep Neural Network took enormous time for training and produced bad generalization. Hinton, OsinderoandTeh (2006) address this issue with an objective that unsupervised learning using initial weight and achieve generalization by applying backpropagation to fine-tune the initial weights.

The sparse autoencoder (SA) is the one type deep learning model, on a bio-inspired hierarchical neural network and has an inherent capability to extract higher-level features from the data. A typical SA comprises three layers, namely, the input layer, one hidden layer, and the reconstruction layer, and it is shown in Figure 6. It performs two functions, namely encoding function, and decoding function. During the encoding phase, the input data $x_i, i=1,2,\ldots,N$ is represented by the hidden layer h_i using a non-linear activation function, and is shown by Equation 13and14:

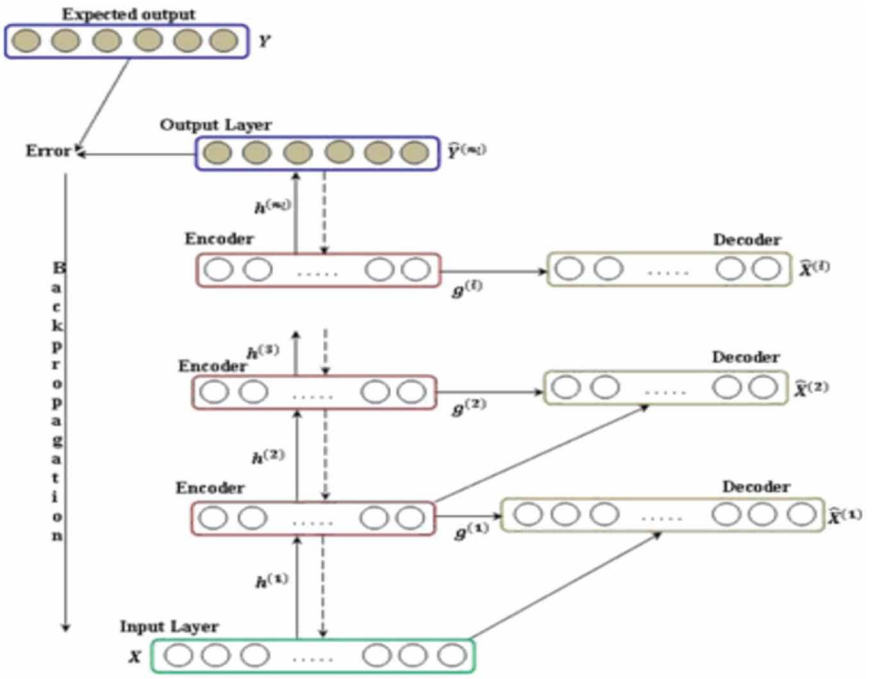
$$h_i=f\left(x_i\right)\approx\hat{x}_i \tag{13}$$

$$f\left(x_i\right)=sigm\left(W_0x_i+b_0\right) \tag{14}$$

where W_0 and b_0 represent the initial weight and bias of the network, respectively. The logistic sigmoid function $\left(1+exp\left(x\right)\right)^{-1}$ is mostly chosen as the activation function for encoding and decoding. In the decoding stage, $\hat{x}_i$ is reconstructed to the input feature by the same approximation function and is shown as follows:

$$\hat{x}_i=g\left(h_i\right)=sigm\left(W_1h_i+b_1\right) \tag{15}$$

Figure 6. A typical Stackedautoencoder structure



where W_1 and b_1 denote as the updated weight and bias for the next stage, respectively.

The quality of reconstruction performed by the decoding function is calculated as the loss function $L\left(X,\hat{X}\right)$. The main objective of the sparse autoencoder is to minimize the expected loss with some constraints, such as by restricting the size of the hidden unit. The sparse autoencoder proves that the network can still discover useful patterns by imposing more constraints even with a large number of hidden units. To maintain the average activation of the hidden unit $\left(\hat{\rho}_j\right)$, the sparsity parameter $\left(\rho\right)$ is introduced to enforce $\hat{\rho}_j = \rho$, so that value is overfitting, that is nearly zero. The deviation of $\hat{\rho}_j$ from ρ is avoided by adding penalty term based on Kullback-Leibler divergence function and can also be written as the probability density functions of two distributions by Equation16:

$$\sum_{j=1}^{s_2} KL(\rho \,||\, \hat{\rho}_j) = \sum_{j=1}^{s_2} \rho log \frac{\rho}{\hat{\rho}_j} + \left(1-\rho\right) log \frac{1-\rho}{1-\hat{\rho}_j} \tag{16}$$

The final overall cost function of the sparse autoencoder mathematically ex-pressed by Equation17 and 18:

$$J_{sparse}\left(W,b\right) = J\left(W,b\right) + \beta \sum_{j=1}^{s_{l+1}} KL(\rho \,||\, \hat{\rho}_j) \tag{17}$$

$$J\left(W,b\right)=\frac{1}{m}\sum_{i=1}^{N}\frac{1}{2}\left(h_{W,b}\left(x_i-\hat{x}_i\right)^2+\frac{\lambda}{2}\sum_{i=1}^{s_l}\sum_{j=1}^{s_{l+1}}W_{j,i}^{l}+\beta\sum_{j=1}^{s_{l+1}}KL(\rho\,||\,\hat{\rho}_j)\right) \tag{18}$$

The first two terms in Equation 16 are an average sum-of-squares error and a regularization term that tends to reduce the magnitude of the weight and helps to avoid overfitting. These terms control the sparsity penalty weight. In the deep neural network, many sparse autoencoders are trained in isolation. The encoders from the autoencoder have been used to extract features. The encoders from the autoencoders can be stacked together with the softmax layer to form a deep-stacked autoencoder based neural network. The softmax classifier is the multi-class simplification of the logistic sigmoid function.

For multi-class classification problems, the given training set $-\left(x_1,y_1\right),\left(x_2,y_2\right),\cdots,\left(x_m,y_m\right)\}$, is associated with the target that $y_i\in\left\{1,2\cdots,K\right\}$ where K is the number of classes. The cross-entropy loss function is used for a 1-of-k coding scheme is denoted by Equation19.

$$E\left(\theta\right)=-\sum_{i=1}^{n}\sum_{j=1}^{k}\hat{y}_{ij}lny_j\left(x_i,\theta\right) \tag{19}$$

where $\hat{y}_{ij}$ is the feedforward step outcome that indicates the i^{th} sample assign to the j^{th} class, and $y_j\left(x_i,\theta\right)$ can be inferred as the probability of assigning i^{th} sample to j^{th} the class that is, $(\hat{y}_j=1\mid x_i)$ and θ s the parameter vector. The softmax function is the activation function of the output layer, and it is represented by Equation 20:

$$y_r\left(x\right)=\frac{\exp\left(a_r\left(x\right)\right)}{\sum_{j=1}^{k}\exp\left(a_r\left(x\right)\right)} \tag{20}$$

where $0\le y_r\le 1$ is the class prior probability, $\sum_{j=1}^{k}y_r=1$ and a_r is the conditional probability of the sample that belongs to class *r*. The results for the stacked autoencoder based deep neural network can be fine-tuned by performing backpropagation on the whole network. During back-propagation, the parameters weight (W) and bias (b) of the stacked encoder are updated by Equation21-24:

$$W^l=W^l-\eta\left[\left(\frac{1}{m}\Delta W^l\right)+\lambda W^l\right] \tag{21}$$

$$b^l=b^l-\eta\left[\frac{1}{m}\Delta b^l\right] \tag{22}$$

$$\Delta W^l=\Delta W^l+d^{(l+1)}\left(a^{(l)}\right)^T \tag{23}$$

$$\Delta b^{l} = d^{(l+1)} \tag{24}$$

where $d^{n_l} = -\left(y - a^{n_l}\right) f^{'}\left(z^{(n)}\right)$ is the error term at the output layer and for the intermediate layer $d^{l} = \left(\left(W^{l}\right)^{T} d^{l+1}\right) f'\left(z^{(n)}\right), l = n_l - 1, n_l - 2, \ldots 3, 2$, and η is the learning rate. This process continues until the error rate decreases; otherwise, stop and return the classification result.

Process of the Proposed Scheme Via Deep Stacked Autoencoder

This paper introduces the automatic facial expression recognition via a deep stacked autoencoder based neural network to improve recognition accuracy using HLTD. Recently, high-dimensional feature-based facial analysis performs better in most cases. Based on this observation, this paper introduces a high dimensional hybrid feature obtained by the fusion of LBP and LNXP. For AFER, preprocessing is required to make the given data suitable for subsequent steps, which in turn increases the overall system performance. At first, face detection is carried out based on the object detection algorithm (Viola, & Jones, 2004)to segregate the region of interest from the background, followed by contrast adjustment using CLAHE (Zuiderveld, 1994). All images are downsampled to the size of 48X48 to solve the range of faces caused by image acquisition conditions.

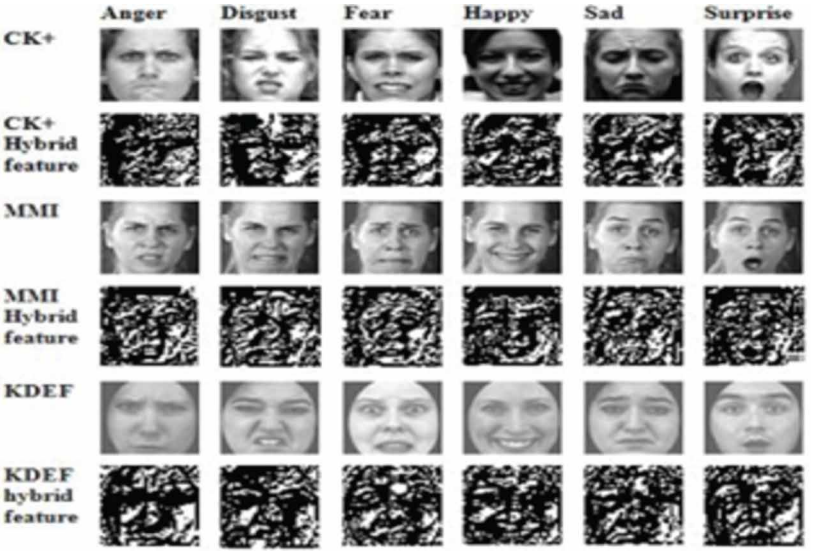
After preprocessing, two types of texture features are extracted to evaluate the proposed approach. The resultant feature space is very high in dimension. Though the high-dimensional feature learning takes more time and space, the bitwise AND operation is performed over LBP and LNXP to make it compact and efficient to apply. The resultant hybrid featured image is then fed into the deep stacked autoencoders.During the training stage, the higher-order feature is encoded by the first stage of autoencoder in an unsupervised manner. Based on the correlation exist on the input data, mapping is carried out by the hidden layers efficiently. In the second stage, the entire network is fine-tuned by the backpropagation algorithm in a supervised manner.

RESULT AND DISCUSSION

In this section, hybrid feature extraction and the classification results are reported on the publically available and well known CK+ (Lucey et al., 2010), MMI (Valstar, & Pantic, 2010) and KDEF-dyn (Calvo, Fernández-Martín, Gutiérrez-García, & Lundqvist, 2018; Lundqvist, Flykt, & Öhman, 1998) dataset using the method outlined in section 3. Finally, the classification results are compared against state of the art methods reported in the literature. In this experiment, the onset and apex state frames of the six basic emotions are selected for further processing. Initially, feature extraction algorithms are applied to the preprocessed facial image. The feature extraction results are displayed in Figure 7. The performance of the HLTD is evaluated by dividing the dataset into N-fold more explicitly with approximately equal size, and N is chosen based on the number of subjects involved in the given dataset (Rathee, & Ganotra, 2017). The samples from the one fold are used for testing, and the rest of the folds are used for training. Therefore, the training and testing sets are disjoint sets. Finally, the overall accuracy of the proposed approach is reported with the mean accuracy of N-fold recognition results.

In the proposed deep learning framework, DSA consist of five layers, namely the input layer, three hidden layers, and the output layer. The greedy layer-wise training is used to identify optimal parameters. For that, the input layer is trained with hybrid featured image and weight, and bias values are returned to the next level. Meanwhile, the first layer transforms the raw input data into a vector containing the activation of the hidden units and then trains the second layer. The same process is repeated for the successive layers by connecting each layer output to the input of the subsequent layer. In this way, the parameters of each layer trained, and the best parameters are maintained for the rest of the model. The backpropagation based fine-tuning is done to yield better results by updating the

Figure 7. Sample images from working dataset (odd rows) and its corresponding featured image(even rows)



parameters of all layers at once. Since this framework is designed for classification, the decoding part of the stacked autoencoder is omitted. The parameter details of DSA is given in Table 1.Parameters are fixed after many trails and hit method. The maximum epochs for pre-training and fine-tuning are set to 400.

Results on CK+ Dataset

The Extended Cohn-Kanade (CK+) data set contains 593 videos of 123 persons, where 31% are male, and 69% are female with the age group between 18-50 years. The image sequences contain seven facial expressions, namely anger, disgust, fear, happy, sad, surprise, and neutral. In this part, the logically fused hybrid featured images are used as the input for the DSA, and the results after fine-tuning are reported in Tables2 and 3. The encoding part of the autoencoder would be very different if the data were completely random. But if there are correlated features in the input data, then the algorithm will be able to determine some of those correlations quickly. Based on this intuition, hybrid featured images are given as input for DSA. The average recognition rate of the hybrid feature is 97.5% for 6-class recognition and 86.3% for 7- class recognition, respectively.

From Table 2, it can be observed that all six emotions are classified with high accuracy. Among six expressions, anger, disgust, and happy achieve accuracy of 100%, due to the presence of distinctive hybrid features on those expressions. At the same time, the fear and surprise emotions provide consistent results due to misclassification among expression with sad. In general, misclassification among expressions occurs due to the minor variations in muscle movements for the different

Table 1. Deep stacked autoencoder parameters settings

Number of hidden layers	**Number of hidden units per layer**	**L2WeightRegularization**
3	400-200-50	0.004-0.002-0.002
Sparsity Regularization	**Sparsity Proportion**	**Max Epochs**
4	0.5-0.3-0.3	400-200-50

Table 2. Confusion matrix of six class recognition on CK+ dataset using HLTD

		Predicted Label					
		Anger	**Disgust**	**Fear**	**Happy**	**Sad**	**Surprise**
Truth Label	Anger	100	0	0	5	0	0
	Disgust	0	100	0	0	0	0
	Fear	0	0	96	0	0	4
	Happy	0	0	0	100	0	0
	Sad	5	0	0	0	90	5
	Surprise	0	0	0	0	3.3	96.7
	Avg. Accuracy: 97.5%						

Table 3. Confusion matrix of seven class recognition on CK+ dataset using HLTD

		Predicted Label						
		Anger	**Disgust**	**Fear**	**Happy**	**Sad**	**Surprise**	**Neutral**
Truth Label	Anger	86.4	0	0	0	0	0	13.6
	Disgust	0	87.9	3.1	0	4.5	0	4.5
	Fear	0	0	95.5	0	0	0	4.5
	Happy	0	2.9	2.9	91.2	0	0	2.9
	Sad	5.55	0	0	0	88.9	0	5.55
	Surprise	0	0	0	0	3.1	90.6	6.3
	Neutral	20.7	0	0	3.45	3.45	6.9	65.5
	Avg. Accuracy: 86.3%							

expressions of the same person and the inter-person variation for the same expression. For the seven classes, the average accuracy reduced slightly after the inclusion of neutral expression, and it is shown in Table 3.Noticeably, expressions are misclassified with the neutral state so that the type I error rates occurred up to 13.6%, 12.1%, 11.1%, 9.4% and 8.8% for anger, disgust, sad and surprise, respectively. The type I error is the ratio between false positive and the sum of false positive and true negative. However, the highest recognition rate is achieved for fear with 95.5%, and other expressions also achieve reasonable recognition rate. Also, the neutral expression recognition rate is comparatively low due to the misclassification with anger so that 26.9% type I error occurred. The seven class average recognition rate is 11.2% lower than the six class expressions. One of the main reason for the performance degradation is insufficient training samples for some expressions.

Results on MMI Dataset

The MMI face dataset contains a frontal and profile view with various facial expressions. In our experiments, the frontal view images (Part II) are considered, which consists of 238 videos of 28 subjects. Each of the sequences is grouped into six basic emotions (anger, disgust, fear, happiness, sadness, and surprise). Table 4 shows the result of a hybrid feature based 6-class person independent experiment on the MMI dataset. As implemented in the CK+ dataset, samples of the single subject's images are selected for testing, and the remaining subject's images are used for training. The overall recognition accuracy with the hybrid feature is 94.1%.

Table 4. Confusion matrix of six class recognition on MMI dataset using HLTD

		Predicted Label					
		Anger	**Disgust**	**Fear**	**Happy**	**Sad**	**Surprise**
Truth Label	Anger	97.4	1.8	0.8	0	0	0
	Disgust	2.75	94.5	2.75	2.75	0	0
	Fear	2.75	3.05	91.6	0	0.8	0.8
	Happy	0	0.8	0	99.2	0	0
	Sad	5.4	0.8	4.7	0.8	88.3	0
	Surprise	0.8	0.8	4.7	0	1.8	91.9
	Avg. Accuracy: 94.1%						

Like the CK+ dataset, the highest recognition is achieved for happy with 99.2%, and the lowest rate is achieved for sad with 88.3%.summarizeApart from these two expressions, the rest of the emotions provide more than 90% results due to very low misclassification among the expressions. From the result, it is observed that the highest misclassification occurs in angry, with 11.7% type I error. Because both expressions involve micro muscle movements around eyebrow and mouth regions, and remaining regions have a similar shape and appearance. Also, poor representation or pose given by the subject is one of the reasons for misclassifications. In all cases, increasing the number of samples per expression can further improve the recognition rate. Hence to improve the detection rate, a secondary feature like forehead wrinkle, nose side wrinkle may be incorporated. According to Table 5, the inclusion of neutral expression increase the misclassification among other expression and get the average recognition rate of 93.1%. Due to the good recognition of neutral, the overall recognition rate not reduced much.

Moreover, the subjects wearing eyeglasses the primary source of the presence of inappropriate information in feature vector while depicting the facial expressions characteristics; hence, including this useless information may produce uncertainty in the feature description. These are some of the causes of the lower accuracy of HLTD in MMI using a stacked autoencoder. However, improving the performance of the proposed method in such conditions can be another motivating research problem and leave it here as future work.

Table 5. Confusion matrix of seven class recognition on MMI dataset using HLTD

		Predicted Label						
		Anger	**Disgust**	**Fear**	**Happy**	**Sad**	**Surprise**	**Neutral**
Truth Label	Anger	95.6	1.89	0	1.89	0.62	0	0
	Disgust	0	92.8	4.5	1.8	0.9	0	0
	Fear	3.2	2.5	86.3	1.6	1.6	4.8	0
	Happy	1.6	0	3.3	95.1	0	0	0
	Sad	2.97	1.98	4.96	0.99	89.7	0	0
	Surprise	0.87	0	2.6	0	1.73	94.8	0
	Neutral	0	0	0	0	0	0	100
	Avg. Accuracy: 93.1%							

Results on KDEF-Dyn Dataset

The dynamic version of the Karolinska Directed Emotional Faces (KDEF-Dyn) has been often used in emotion recognition. It contains video clips of 40 models(20 females and 20 males) displaying each of seven expressions with morphing software is selected for feature extraction. After preprocessing, two types of local features are extracted from the detected face, and the feature fusion is carried out to obtain the HLTD. The hybrid featured images are fed into the DSA for classification. Table 6 shows the result of the six emotion classification. The average recognition rate of the proposed approach on KDEF-dyndataset is 88.5%, with the highest recognition accuracy of 98% for happy and the lowest recognition accuracy of 75% for fear.

From the results, it is observed that misclassification occurs among anger, disgust, and fear due to the existence of similar muscle movements around all primary components of the face. Similarly, happy and surprise are misclassified as fear because these expressions involve a similar muscle movement in the mouth. Table 7 shows the performance of the proposed approach in seven emotion recognition. The additional expression reduces the average recognition rate to 80.9%, which is 7.6% lower than the six class recognition rate. The highest recognition rate is 98% for neutral, and the lowest recognition rate is 63.7% for fear in seven class emotion classification. When the neural expression, for classification, the discriminating power of hybrid feature reduced so that the misclassification among expression increased except neutral and happy.

Table 6. Confusion matrix of six class recognition on KDEF-dyndataset using HLTD

		Predicted Label					
		Anger	**Disgust**	**Fear**	**Happy**	**Sad**	**Surprise**
Truth Label	Anger	95.7	1.1	0	0	0	3.2
	Disgust	8.4	85.2	4.6	0	0.9	0.9
	Fear	1.92	1.92	75	1.92	9.62	9.62
	Happy	0	0	1	98	0	1
	Sad	0	5.96	8.94	0	85.1	0
	Surprise	0	0	6.4	0	0	93.6
	Avg.Accuracy: 88.5%						

Table 7. Confusion matrix of seven class recognition on KDEF-dyndataset using HLTD

		Predicted Label						
		Anger	**Disgust**	**Fear**	**Happy**	**Sad**	**Surprise**	**Neutral**
Truth Label	Anger	76.7	13.17	0	0	11.14	0	0
	Disgust	10.77	86	3.23	0	0	0	0
	Fear	0.98	3.92	63.7	0.98	18.62	11.76	0
	Happy	0.94	2.8	0	92.5	2.8	0.94	0
	Sad	9.79	0	17.4	0	72.8	0	0
	Surprise	0	0	15.5	0	0	84.5	0
	Neutral	0	0	0	2	0	0	98
	Avg. Accuracy: 80.9%							

To further confirm the dominance of the proposed approach under noise, zero-mean Gaussian noise is added to generate noisy images with a variance that varies from 1 to 10. From the noisy images, HLTD is extracted by the logical fusion of LBP and LNXP. The featured image is given to DSA for classification with the same parameter. Table 8 is showing how the noise factor affects the overall recognition rate with a distinctive feature and hybrid feature. Although noise can severely affect the recognition accuracy in all cases, it has been shown by the experiments that the hybrid feature performs well than individual features in the noise environment. The combination of these two features can efficiently integrate the gains mutually by preserving local features of the facial image differently. Therefore, the combination of LBP and LNXP descriptor is an effective method for AFER in different noise environments.

Feature Level Performance Validation

Several experiments are carried out to confirm the performance of the proposed approach. The first individual feature contribution towards the six class emotion classification is performed on three datasets. Then the hybrid featured image is classified using the DSA-based softmax classifier. Table 9 shows the average recognition rate of three types of features, and the results clearly show that the performance of the hybrid feature notably better than individual feature vector with 97.5% for CK+ dataset, 94.1% for MMI and 88.5% for KDEF dataset. The summary of the recognition rate of all the three types of features (LBP alone, LNXP alone, and hybrid feature) is shown in Figure 8(a),8(b), and 8(c) for three types of datasets. For the same experimental setup, LBP and LNXP feature achieves the average recognition accuracy of 77.2% and 90.7% respectively in the CK+ dataset, which is 20.3% and 6.8% lower than the hybrid feature, respectively. Except sad, all other expressions achieve satisfactory results.

Similarly, in the MMI dataset, the individual feature performance is lower than the hybrid feature with 85.4% and 82.8% average accuracy using LBP and LNXP features, respectively. The hybrid feature achieves 94.1% average accuracy, which is 8.7% and 11.3% higher than LBP and LNXP features. Like the CK+ dataset, the sadness recognition rate is lower than the other expressions because the muscle movements all over face for sad is very minimal so that the misclassification rate is higher than other emotions. For KDEF dataset also hybrid feature achieves 7% and 12.7% more recognition rate than LBP and LNXP features, respectively. Here, much confusion occurs in fear with happy, sad, and surprise. Other than that, the remaining expression showed good recognition accuracy. Thus, the proposed hybrid feature shows its dominance over LBP and LNXP methods. The comparison of the average recognition rate of three types of features illustrates the superiority of the HLTD on the three public datasets using DSA.

Performance Comparison With Existing Work

In this subsection, the proposed method results are compared with other state-of-art methods. The comparison task becomes complicated due to the unknown fact of the input data and experimental setup. However, the proposed approach performance is compared against the existing methods that adopted similar protocols on CK+ and MMI dataset. Table 10 shows the average accuracy attained

Table 8. Expression recognition rate on KDEF-dyndataset with varying noise

Descriptor	Without Noise	With Noise	
		σ=2.24	σ=3.16
LBP	81.5	76	60.5
LNXP	75.8	53.9	53.5
HLTD	88.5	78.2	72.6

Figure 8. Performance of three types of features on (a) CK+ (b) MMI (c) KDEF-dyn dataset

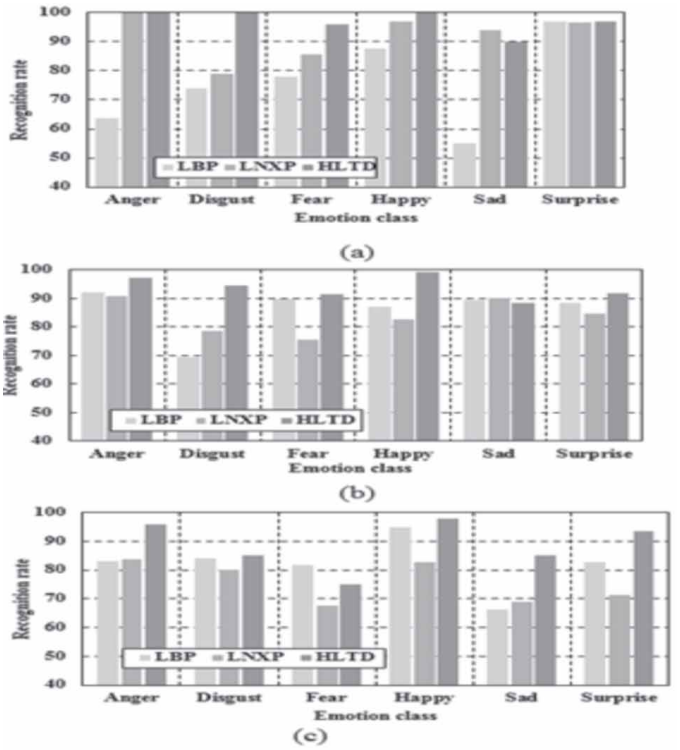


Table 9. Average recognition rate using three types of a feature on three datasets

	LBP	LNXP	HLTD
CK+	77.2	90.7	97.5
MMI	85.4	82.8	94.1
KDEF	81.5	75.8	88.5

by all the challenging methods on the CK+ and MMI datasets. For CK+, many approaches achieve good recognition accuracy. The reason is CK+ images are frontal face images taken under a controlled environment. In contrast, the classification accuracy obtained on the MMI dataset by all the existing methods is comparatively lower than the CK+ dataset, as MMI facial images are taken under more challenging situations.

Among all the comparative methods, the two highest accuracies are achieved by Majumder et al.,(2016) and Barman and Dutta (2019) using a hybrid feature that combines geometric and texture features. In Majumder et al.,(2016) approach, feature fusion is carried out using three autoencoders to balance the different nature of features, but the resultant high dimensional feature space increases the computational complexity. In Barman and Dutta (2019) approach, the D-T signature feature gets 98.7% average accuracy on CK+, but the experimental results seem to be person dependent. In Meena et al.,(2018) approach, there is a trade-off between the feature size and initial computation cost. Other local texture feature-based approaches like NEDPs (Iqbal, Abdullah-Al-Wadud, Ryu, Makhmudkhujaev, & Chae, 2018), Improved GLTP (Holder, & Tapamo,

Table 10. Comparison of the proposed approach with state-of-art methods

Approach	Feature	CK+	MMI
Iqbal et al., 2018	NEDPs	93.78	69.01
Akputu et al., (2018)	Gabor	88	-
Rathee and Ganotra, 2017	Geometric feature with PCA	90	-
Holder and Tapamo, 2017	Improved GLTP	86.5	-
Majumder et al., 2016	Geomantic and regional LBP	**98.95**	**97.55**
Chen et al., 2018	Deep Features	89.03	-
Fan et al., 2018	W-PHOG TOP	84.9	76.6
Jain et al., 2018	Hybrid deep feature	-	92.07
Makhmudkhujaev et al., 2019a	LDSP frequent feature	94.5	70.63
Mistry, Zhang, Neoh, Lim, & Fielding, 2016	hvnLBP	90.6	-
Meena et al., 2019	HOG with GSP	97.61	-
Barmanand Dutta, (2019)	Distance and texture signature feature	98.6	94.3
Proposed Approach	**Hybrid Local Texture Descriptor**	97.5	94.1

2017), WPHOG TOP (Fan, Yang, Ye, & Yang, 2018), LDSP (Makhmudkhujaev, Iqbal, Ryu, & Chae, 2019) and HOG with GSP (Meena et al., 2019) extract histogram oriented features from the specific region or divide the whole face into a different block. This histogram-based feature description is simple, but, it needs adequate illustrations, and excludes the spatial information inside each region. For the MMI dataset, the average recognition accuracy of existing methods lower than the CK+ dataset. Like CK+, in MMI, also, Majumder et al. (2016) and Barman and Dutta (2019) achieve the topmost accuracy of 97.55% and 94.3%, respectively, which is slightly better than our proposed method. The subjects wearing glasses generate some un-wanted texture may arise discrepancy in the feature description, which leads to lesser accuracy in MMI. Even though the existing techniques use relatively more complex feature representations than our approach, the proposed HLTD shows better accuracy than those methods.

Further, the deep feature-based methods also obtain satisfactory results with some additional preprocessing. Usually, deep learning-based methods require more training data and more computing power to provide a better result in AFER. In a deep learning based approach, the training sample size is increased with the help of data augmentation, which generates simulated data. Here, the proposed approach robustness is proved without such additional data with good recognition accuracy. Nevertheless, the block-based texture descriptor retains the spatial information, the number of sample code required to describe the micro-texture information decreases. Due to this type of sampling error, the histogram-based description cannot increase spatial information. Hence, instead of generating a histogram, the proposed hybrid featured image is given directly to the stacked autoencoder to get higher feature level abstraction. From the experimental results, it is observed that the proposed HLTD outperforms the existing work for six and seven class expressions on three datasets.

The feature vector computation time of the proposed method is calculated using an un-optimized MATLAB R2017b code on a desktop machine with octal-core CPU running at 3.5 GHz. For a single image, LBP takes 0.0289 seconds, and LNXP takes 0.0229, whereas the hybrid feature takes 0.0131 seconds, which is faster and provides a notable gain in recognition accuracy than the single feature. According to the experimental results, the proposed method can be applied for real-time applications.

Percent Error Analysis With Existing Work

The percent error is a measure of the degree of closeness of a measured or calculated value to its actual value (literature value), and it is expressed by Equation (25):

$$\%Error = \frac{\left(Accepted\,value - Expermental\,value\right)}{Accepeted\,value} \times 100 \tag{25}$$

The negative percent error shows that the proposed model performs better than existing works. Table 11 and 12 show the average recognition rate percent error of the proposed method with state

Table 11. Percent error comparison on CK+ dataset

Approach	Avg.Recognition rate	% Error
NEDPs (Iqbal et al., 2018)	93.78	-3.97
Gabor (Akputu et al., (2018)	88	-10.80
Geometric feature with PCA (Rathee and Ganotra, 2017)	90	-8.33
Improved GLTP (Holder and Tapamo, 2017)	86.5	-12.72
Geomantic and regional LBP (Majumder et al., 2016)	**98.95**	**1.47**
Deep Features (Chen et al., 2018)	89.03	-9.51
W-PHOG TOP (Fan et al, 2018)	84.9	-14.84
LDSP frequent feature (Makhmudkhujaev et al., 2019a)	94.5	-3.17
hvnLBP (Mistry, Zhang, Neoh, Lim, & Fielding, 2016)	90.6	-7.62
HOG with GSP (Meena et al., 2019)	**97.61**	**0.11**
Distance and texture signature feature (Barmanand Dutta, (2019)	**98.6**	**1.12**
LBP	77.2	-26.30
LNXP	90.7	-7.50

Table 12. Percent error comparison on MMI dataset

Approach	Avg.Recognition rate	% Error
NEDPs (Iqbal et al., 2018)	69.01	-36.357
Geomantic and regional LBP (Majumder et al., 2016)	**97.55**	**3.536648**
W-PHOG TOP (Fan et al, 2018)	76.6	-22.846
Hybrid deep feature (Jain et al., 2018)	92.07	-2.20484
LDSP frequent feature (Makhmudkhujaev et al., 2019a)	70.63	-33.2295
Distance and texture signature feature (Barmanand Dutta, (2019)	**94.3**	**0.212089**
LBP	85.4	-10.1874
LNXP	82.8	-13.6473

of the art methods. In both datasets, a positive percent error is realized for three approaches with less than 1.5%. Besides, those methods achieve a better recognition rate than the proposed method with the high computational cost in terms of feature space.

CONCLUSION

This paper presents an efficient hybrid feature based facial expression recognition framework using deep stacked autoencoder. The proposed hybrid local texture descriptor (HLTD) is derived from the logical fusion of LBP and LNXP. The LNXP represents the mutual relationship among the closest neighborhood pixel whereas the LBP encodes the neighboring pixel relationship with the central pixel. The logical feature fusion of LBP and LNXP gains the advantage of both features by considering two different statistical and structural models in texture analysis. In the next step, higher-level feature mapping using DSA reduces the feature space, which in turn reduces the computational cost and improves the recognition rate via fine-tuning. The effectiveness of three features is analyzed individually and collectively on CK+, MMI, and KDEF dataset. Experimental outcomes on three datasets exhibit that the HLTD outperforms than the descriptors from which it is derived and achieves an average recognition rate of 97.5% for CK+, 94.1% for MMI, and 88.5% for KDEF in six class. Also, the extensive experiments performed on the image having noise demonstrate that the hybrid feature performs better than the other two feature descriptors in a noisy environment. Further, the stacked autoencoder based deep neural network framework with backpropagation algorithm exhibit impressive performance consistently in all cases without augmentation and additional computational cost. In the future, this work can be further extended to investigate the robustness of the proposed feature descriptor in real-time data with the help of deep learning models.

ACKNOWLEDGMENT

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-pro t sectors.

REFERENCES

Akputu, O. K., Seng, K. P., Lee, Y., & Ang, L. M. (2018). Emotion recognition using multiple kernel learning toward E-learning applications. *ACM Transactions on Multimedia Computing Communications and Applications*, *14*(1), 1–20. doi:10.1145/3131287

Al Rahhal, M. M., Bazi, Y., AlHichri, H., Alajlan, N., Melgani, F., & Yager, R. R. (2016). Deep learning approach for active classification of electrocardiogram signals. *Information Sciences*, *345*, 340–354. doi:10.1016/j.ins.2016.01.082

Barman, A., & Dutta, P. (2019). Facial expression recognition using distance and texture signature relevant features. *Applied Soft Computing*, *77*, 88–105. doi:10.1016/j.asoc.2019.01.011

Bartlett, M. S., Littlewort, G., Frank, M., Lainscsek, C., Fasel, I., & Movellan, J. (2005, June). Recognizing facial expression: machine learning and application to spontaneous behavior. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)* (Vol. 2, pp. 568-573). IEEE. doi:10.1109/CVPR.2005.297

Calvo, M. G., Fernández-Martín, A., Gutiérrez-García, A., & Lundqvist, D. (2018). Selective eye fixations on diagnostic face regions of dynamic emotional expressions: KDEF-dyn database. *Scientific Reports*, *8*(1), 17039. doi:10.1038/s41598-018-35259-w PMID:30451919

Chen, L., Zhou, M., Su, W., Wu, M., She, J., & Hirota, K. (2018). Softmax regression based deep sparse autoencoder network for facial emotion recognition in human-robot interaction. *Information Sciences*, *428*, 49–61. doi:10.1016/j.ins.2017.10.044

Christou, N., & Kanojiya, N. (2019). Human facial expression recognition with convolution neural networks. In *Third International Congress on Information and Communication Technology* (pp. 539-545). Springer. doi:10.1007/978-981-13-1165-9_49

Cootes, T. F., Taylor, C. J., Cooper, D. H., & Graham, J. (1995). Active shape models-their training and application. *Computer Vision and Image Understanding*, *61*(1), 38–59. doi:10.1006/cviu.1995.1004

Edwards, G. J., Cootes, T. F., & Taylor, C. J. (1998). Face recognition using active appearance models. In *European conference on computer vision*. Springer.

Fan, X., & Tjahjadi, T. (2019). Fusing dynamic deep learned features and handcrafted features for facial expression recognition. *Journal of Visual Communication and Image Representation*, *65*, 102659. doi:10.1016/j.jvcir.2019.102659

Fan, X., Yang, X., Ye, Q., & Yang, Y. (2018). A discriminative dynamic framework for facial expression recognition in video sequences. *Journal of Visual Communication and Image Representation*, *56*, 182–187. doi:10.1016/j.jvcir.2018.09.011

Fu, Y., Cao, L., Guo, G., & Huang, T. S. (2008, July). Multiple feature fusion by subspace learning. In *Proceedings of the 2008 international conference on Content-based image and video retrieval* (pp. 127-134). ACM. doi:10.1145/1386352.1386373

Guo, X., Zhang, X., Deng, C., & Wei, J. (2013). Facial Expression Recognition based on Independent Component Analysis. *Journal of Multimedia*, *8*(4). Advance online publication. doi:10.4304/jmm.8.4.402-409

Hinton, G. E., Osindero, S., & Teh, Y. W. (2006). A fast learning algorithm for deep belief nets. *Neural Computation*, *18*(7), 1527–1554. doi:10.1162/neco.2006.18.7.1527 PMID:16764513

Holder, R. P., & Tapamo, J. R. (2017). Improved gradient local ternary patterns for facial expression recognition. *EURASIP Journal on Image and Video Processing*, *2017*(1), 42. doi:10.1186/s13640-017-0190-5

Huang, M. W., Wang, Z. W., & Ying, Z. L. (2010, October). A new method for facial expression recognition based on sparse representation plus LBP. In *2010 3rd International Congress on Image and Signal Processing* (Vol. 4, pp. 1750-1754). IEEE. doi:10.1109/CISP.2010.5647898

Iqbal, M. T. B., Abdullah-Al-Wadud, M., Ryu, B., Makhmudkhujaev, F., & Chae, O. (2018). Facial expression recognition with neighborhood-aware edge directional pattern (NEDP). *IEEE Transactions on Affective Computing*.

Jain, N., Kumar, S., Kumar, A., Shamsolmoali, P., & Zareapoor, M. (2018). Hybrid deep neural networks for face emotion recognition. *Pattern Recognition Letters*, *115*, 101–106. doi:10.1016/j.patrec.2018.04.010

Kalsum, T., Anwar, S. M., Majid, M., Khan, B., & Ali, S. M. (2018). Emotion recognition from facial expressions using hybrid feature descriptors. *IET Image Processing*, *12*(6), 1004–1012. doi:10.1049/iet-ipr.2017.0499

Liang, D., Liang, H., Yu, Z., & Zhang, Y. (2019). Deep convolutional BiLSTM fusion network for facial expression recognition. *The Visual Computer*, 1–10.

Lucey, P., Cohn, J. F., Kanade, T., Saragih, J., Ambadar, Z., & Matthews, I. (2010, June). The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops* (pp. 94-101). IEEE. doi:10.1109/CVPRW.2010.5543262

Lundqvist, D., Flykt, A., & Öhman, A. (1998). The Karolinska directed emotional faces (KDEF). *CD ROM from Department of Clinical Neuroscience, Psychology section. Karolinska Institutet*, *91*, 630.

Luo, Y., Wu, C. M., & Zhang, Y. (2013). Facial expression recognition based on fusion feature of PCA and LBP with SVM. *Optik (Stuttgart)*, *124*(17), 2767–2770. doi:10.1016/j.ijleo.2012.08.040

Majumder, A., Behera, L., & Subramanian, V. K. (2016). Automatic facial expression recognition system using deep network-based data fusion. *IEEE Transactions on Cybernetics*, *48*(1), 103–114. doi:10.1109/TCYB.2016.2625419 PMID:27875237

Makhmudkhujaev, F., Abdullah-Al-Wadud, M., Iqbal, M. T. B., Ryu, B., & Chae, O. (2019). Facial expression recognition with local prominent directional pattern. *Signal Processing Image Communication*, *74*, 1–12. doi:10.1016/j.image.2019.01.002

Makhmudkhujaev, F., Iqbal, M. T. B., Ryu, B., & Chae, O. (2019). Local directional-structural pattern for person-independent facial expression recognition. *Turkish Journal of Electrical Engineering and Computer Sciences*, *27*(1), 516–531. doi:10.3906/elk-1804-58

Maximiano da Silva, F. A., & Pedrini, H. (2016). Geometrical Features and Active Appearance Model Applied to Facial Expression Recognition. *International Journal of Image and Graphics*, *16*(04), 1650019. doi:10.1142/S0219467816500194

Meena, H. K., Joshi, S. D., & Sharma, K. K. (2019). Facial Expression Recognition Using Graph Signal Processing on HOG. *Journal of the Institution of Electronics and Telecommunication Engineers*, 1–7. doi:10.1080/03772063.2019.1565952

Mistry, K., Zhang, L., Neoh, S. C., Lim, C. P., & Fielding, B. (2016). A micro-GA embedded PSO feature selection approach to intelligent facial emotion recognition. *IEEE Transactions on Cybernetics*, *47*(6), 1496–1509. doi:10.1109/TCYB.2016.2549639 PMID:28113688

Munir, A., Hussain, A., Khan, S. A., Nadeem, M., & Arshid, S. (2018). Illumination invariant facial expression recognition using selected merged binary patterns for real world images. *Optik (Stuttgart)*, *158*, 1016–1025. doi:10.1016/j.ijleo.2018.01.003

Nazir, M., Jan, Z., & Sajjad, M. (2018). Facial expression recognition using histogram of oriented gradients based transformed features. *Cluster Computing*, *21*(1), 539–548. doi:10.1007/s10586-017-0921-5

Nguyen, H. D., Yeom, S., Lee, G. S., Yang, H. J., Na, I. S., & Kim, S. H. (2019). Facial emotion recognition using an ensemble of multi-level convolutional neural networks. *International Journal of Pattern Recognition and Artificial Intelligence*, *33*(11), 1–17. doi:10.1142/S0218001419400159

Ojala, T., Pietikäinen, M., & Mäenpää, T. (2002). Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *24*(7), 971–987. doi:10.1109/TPAMI.2002.1017623

Rathee, N., & Ganotra, D. (2017). Modelling Facial Features for Emotion Recognition and Synthesis. *Journal of the Institution of Electronics and Telecommunication Engineers*, *63*(6), 845–852. doi:10.1080/03772063.2017.1329639

Rivera, A. R., Castillo, J. R., & Chae, O. (2015). Local directional texture pattern image descriptor. *Pattern Recognition Letters*, *51*, 94–100. doi:10.1016/j.patrec.2014.08.012

Sadeghi, H., & Raie, A. A. (2019). Histogram distance metric learning for facial expression recognition. *Journal of Visual Communication and Image Representation*, *62*, 152–165. doi:10.1016/j.jvcir.2019.05.004

Shan, C., Gong, S., & McOwan, P. W. (2009). Facial expression recognition based on local binary patterns: A comprehensive study. *Image and Vision Computing*, *27*(6), 803–816. doi:10.1016/j.imavis.2008.08.005

Sharan, R. V., & Moir, T. J. (2017). Robust acoustic event classification using deep neural networks. *Information Sciences*, *396*, 24–32. doi:10.1016/j.ins.2017.02.013

Shbib, R., & Zhou, S. (2015). Facial expression analysis using active shape model. *International Journal of Signal Processing*. *Image Processing and Pattern Recognition*, *8*(1), 9–22. doi:10.14257/ijsip.2015.8.1.02

Shi, C., & Pun, C. M. (2017). 3D multi-resolution wavelet convolutional neural networks for hyperspectral image classification. *Information Sciences*, *420*, 49–65. doi:10.1016/j.ins.2017.08.051

Singh, G., & Chhabra, I. (2018). Effective and Fast Face Recognition System Using Complementary OC-LBP and HOG Feature Descriptors With SVM Classifier. *Journal of Information Technology Research*, *11*(1), 91–110. doi:10.4018/JITR.2018010106

Tan, X., & Triggs, B. (2007, October). Enhanced local texture feature sets for face recognition under difficult lighting conditions. In *International workshop on analysis and modeling of faces and gestures* (pp. 168-182). Springer. doi:10.1007/978-3-540-75690-3_13

Topi, M., Timo, O., Matti, P., & Maricor, S. (2000, September). Robust texture classification by subsets of local binary patterns. In *Proceedings 15th International Conference on Pattern Recognition. ICPR-2000* (Vol. 3, pp. 935-938). IEEE. doi:10.1109/ICPR.2000.903698

Turk, M. A., & Pentland, A. P. (1991, June). Face recognition using eigenfaces. In *Proceedings. 1991 IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (pp. 586-591). IEEE. doi:10.1109/CVPR.1991.139758

Valstar, M., & Pantic, M. (2010, May). Induced disgust, happiness and surprise: an addition to the mmi facial expression database. In *Proc. 3rd Intern. Workshop on EMOTION (satellite of LREC): Corpora for Research on Emotion and Affect* (p. 65). Academic Press.

Viola, P., & Jones, M. J. (2004). Robust real-time face detection. *International Journal of Computer Vision*, *57*(2), 137–154. doi:10.1023/B:VISI.0000013087.49260.fb

Wang, Y., Wang, X., & Liu, W. (2016). Unsupervised local deep feature for image recognition. *Information Sciences*, *351*, 67–75. doi:10.1016/j.ins.2016.02.044

Xie, S., & Hu, H. (2018). Facial expression recognition using hierarchical features with deep comprehensive multipatches aggregation convolutional neural networks. *IEEE Transactions on Multimedia*, *21*(1), 211–220. doi:10.1109/TMM.2018.2844085

Yang, L., Ban, X., Li, Y., & Yang, G. (2018). Multiple features fusion for facial expression recognition based on ELM. *International Journal of Embedded Systems*, *10*(3), 181–187. doi:10.1504/IJES.2018.091775

Zeng, N., Zhang, H., Song, B., Liu, W., Li, Y., & Dobaie, A. M. (2018). Facial expression recognition via learning deep sparse autoencoders. *Neurocomputing*, *273*, 643–649. doi:10.1016/j.neucom.2017.08.043

Zhang, T., Zheng, W., Cui, Z., Zong, Y., Yan, J., & Yan, K. (2016). A deep neural network-driven feature learning method for multi-view facial expression recognition. *IEEE Transactions on Multimedia*, *18*(12), 2528–2536. doi:10.1109/TMM.2016.2598092

Zhang, Z., Lu, G., Yan, J., Li, H., Sun, N., & Li, X. (2018). Compact local Gabor directional number pattern for facial expression recognition. *Turkish Journal of Electrical Engineering and Computer Sciences*, *26*(3), 1236–1248.

Zuiderveld, K. (1994, August). Contrast limited adaptive histogram equalization. In Graphics gems IV (pp. 474-485). Academic Press Professional, Inc. doi:10.1016/B978-0-12-336156-1.50061-6

Shanthi Pitchaiyan received her B.E. degree from Periyar Maniammai College of Technology for Women, Thanjavur, Tamil Nadu, India in 2002 and M.E. degree from M.A.M College of Engineering, Tiruchirappalli, Tamil Nadu, India in 2012. Currently she is a research scholar in the Department of Computer Applications, National Institute of Technology, Tiruchirappalli (NITT), Tamil Nadu, India. Her research interest includes computer vision, machine learning and image processing.

Nickolas Savarimuthu is working as Professor in the Department of Computer Applications, National Institute of Technology, Trichy, Tamilnadu, India. He received his M.E. Computer Science from REC, Trichy in 1992 and Ph.D in the year 2007 from NIT, Trichy. He is the Professor In-Charge of the Massively Parallel Programming Laboratory, NVIDIA CUDA Teaching Centre, NIT, Trichy. His research interest includes Evolutionary Algorithms, Data Mining, Big Data Analytics, Distributed Computing, Cloud Computing and Software Metrics.