IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received March 15, 2018, accepted April 7, 2018, date of publication April 10, 2018, date of current version May 2, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2825310

Enhanced Human Face Recognition Using LBPH Descriptor, Multi-KNN, and Back-Propagation Neural Network

MOHANNAD A. ABUZNEID, (Member, IEEE), AND AUSIF MAHMOOD, (Senior Member, IEEE)
Department of Computer Science and Engineering, University of Bridgeport, Bridgeport, CT 06604, USA

Corresponding author: Mohannad A. Abuzneid (mohannad@my.bridgeport.edu)

ABSTRACT Face recognition has become a fascinating field for researchers. The motivation behind the enormous interest in the topic is the need to improve the accuracy of many real-time applications. The complexity of the human face and the changes due to different effects make it more challenging to design as well as implement a powerful computational system for human face recognition. In this paper, we presented an enhanced approach to improve human face recognition using a back-propagation neural network (BPNN) and features extraction based on the correlation between the training images. A key contribution of this paper is the generation of a new set called the T-Dataset from the original training data set, which is used to train the BPNN. We generated the T-Dataset using the correlation between the training images without using a common technique of image density. The correlated T-Dataset provides a high distinction layer between the training images, which helps the BPNN to converge faster and achieve better accuracy. Data and features reduction are essential in the face recognition process, and researchers have recently focused on the modern neural network. Therefore, we used a local binary pattern histogram descriptor to prove that there is potential improvement even using traditional methods. We applied five distance measurement algorithms and then combined them to obtain the T-Dataset, which we fed into the BPNN. We achieved higher face recognition accuracy with less computational cost compared with the current approach by using reduced image features. We test the proposed framework on two small data sets, the YALE and AT&T data sets, as the ground truth. We achieved tremendous accuracy. Furthermore, we evaluate our method on one of the state-of-the-art benchmark data sets, Labeled Faces in the Wild (LFW), where we produce a competitive face recognition performance.

INDEX TERMS Local binary patterns histogram (LBPH), Haar-cascade detection, K-nearest-neighbor, back-propagation neural network (BPNN), labeled faces in the wild (LFW).

I. INTRODUCTION

Human face recognition is a challenging task because of the variability of facial expressions, personal appearances, variant poses, and illumination, as shown in Figure 1 [1]–[4]. In addition, due to the variability in lighting intensity and direction, the number of light sources, and the orientation of the camera, as shown in Figure 2, it is a challenging task to design a face recognition system in real time with a high accuracy recognition rate. Changes in the human face have less of an effect compared to the pose variation and illumination [5]. Reducing the image dimension is necessary to improve the classification processing time since the object recognition system requires an enormous volume for the computing process. LBPH is one of the most popular conventional methods; it is used for robust data representation, as well as histograms, for features reduction [6]–[11].

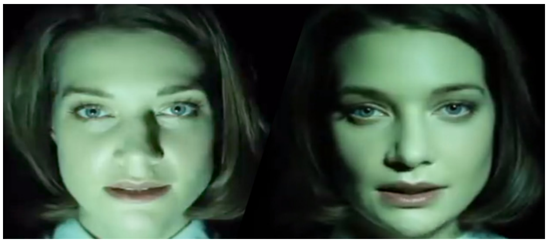

FIGURE 1. Under a variant lighting environment, the face of a single person can look different based on the light source.

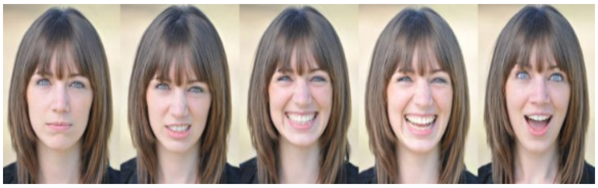

FIGURE 2. Variant facial expression of the same person.

We achieved a strong representation of the face by retaining the majority of dissimilarities in the image features after reducing the dimensionality of the image.

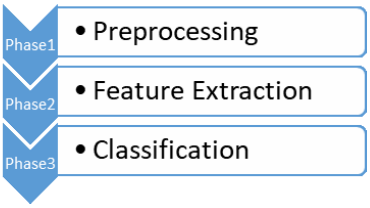

FIGURE 3. Face recognition system process.

Classical human face recognition systems are divided into three phases as shown in Figure 3: The first step is preprocessing, which consists of many types of operations, such as image registration, scaling, face normalization, reducing the effect of background noise, detection and resizing, all of which affect the face recognition accuracy. Feature extraction is the second phase, which can be achieved by using powerful transformation approaches. The image dimension can be reduced to a smaller dimension by retaining significant features. Some of the image descriptors are based on representative methods such as Gabor wavelets and LBP. Ahonen *et al*. [12] presented LBP descriptor, which provides a strong representation of the human face and improves the face recognition by binary-encoding the gray center pixel differences with eight neighboring pixels and then reducing the image dimension by concatenating the histograms of the binary codes. Variant methods are inherited from the LBP, such as Local Ternary Patterns (LTP) [13], which enhanced the LBP against noise. Trefny and Matas [14] proposed direction coded LBP (DLBP) and transition LBP (TLBP) to extract the features using novel encoding strategies. However, Gabor wavelets encodes the face image in a multi-scale and multi-orientation [15], [16]. Therefore, LBP is better at small encoding scales, while Gabor wavelets descriptor is better at the broad encoding scales. LBP outperformed most of the global extraction feature methods such as Principal Component Analysis (PCA) [17]–[19], independent component analysis (ICA) [20]–[22] and Linear Discriminate Analysis (LDA) [23]–[26], in addition to the PCA-inherited methods such as Diagonal PCA [27], Curvelet-based PCA [28], Kernel PCA [29], 2-DPCA [24] and Kernel FLD.

The final phase is the classification that exploits powerful classifiers such as BPNN and the fully connected NN [30]–[32], Support Vector Machine (SVM) [33], Euclidean distance classifier [34], Mahalanobis distance classifier [35], Hidden Markov Models [36], and extreme learning machine [37].

The main contribution of this work is an enhanced human face recognition using LBPH, multi-KNN, and BPNN. The strength of our approach is based on adding a step after the features extraction and dimension reduction to obtain a clear distinction T-Dataset, which will be used to train the BPNN. The novelty consists in a new T-Dataset achieved by taking into consideration the correlation between the training images, unlike existing methods that rely only on the density of the images.

This system starts with some of the preprocessing operations, which helps to reduce the processing time. Thereafter, we used the LBPH method to reduce the image dimension by selecting significant features. The new T-Dataset is obtained using five distance methods. In the final phase, we feed the T-Dataset to our BPNN for offline training. We tested our framework on three datasets, Yale, ORL, and LFW. We have achieved a higher recognition rate accuracy.

The paper is organized as follows. In section II, we present an overview of LBPH, BPNN and the distance methods. In section III, we implement the classical human face recognition system. In section IV, we explain the proposed framework in detail. In section V, we present our experimental results. In Section VI, we present our conclusion and future work.

II. MATERIALS and METHODS

A. LOCAL BINARY PATTERNS HISTOGRAM (LBPH)

Correlation methods require substantial computation time and enormous amounts of storage. Therefore, features reduction and face representation are needed in the face recognition system. LBPH is usually the preferred method in computer vision, image processing, and pattern recognition; it is appropriate for feature extraction because it describes the texture and structure of an image. We represent the face image and reduce the image dimension by applying the LBPH method, extracting the features texture of the image by dividing the image into local regions and extracting the binary pattern for each local region. The original LBP operator, which works on eight neighbors of a pixel, was introduced by Ojala *et al*. [38]. The image is divided into small regions called cells. Each pixel in the cell is compared with each of its eight neighbors. The center pixel value will be used as the threshold value [6]–[11]. The eight-neighbors-pixel will be set to one if its value is equal to or greater than the center pixel; otherwise, the value is set to zero. Accordingly, the LBP code for the center pixel is generated by concatenating the eight neighbor pixel values (ones or zeroes) into a binary code, which is converted to a 256-dimensional decimal for convenience as a texture descriptor of the center pixel. The original LBP operator is shown in Figure 4.

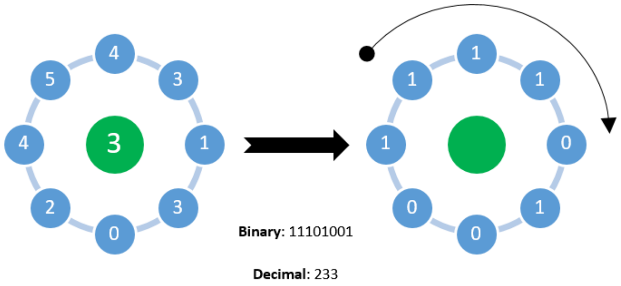

FIGURE 4. Original LBP Operator.

The mathematical formulation of LBP operator is given by:

$$LBP\,(x) = \sum\nolimits_{i=1}^{8} s(G(x^i) - G(x))2^{i-1} \qquad (1)$$

$$s\,(t) = \begin{cases} 1 & t \geq 0 \\ 0 & t < 0 \end{cases} \qquad (2)$$

We used a modified LBP operator called uniform pattern. The pattern is the number of bitwise transitions from 1 to 0 or vice versa. The LBP is called uniform if its uniformity measure is at most 2. For example, the patterns 11111111 (0 transitions), 01111100 (2 transitions) and 11000111 (2 transitions) are uniform, while the patterns 10001000 (3 transitions) and 11010011 (4 transitions) are not. For dimension reduction, we used the histogram to reduce the image features from a 256-dimensional decimal to a 59- dimensional histogram, which contains information about the local patterns. The histogram uses a separate bin for each uniform pattern, and one separate bin for all non-uniform patterns. In the 8-bit binary number, we have 58 uniform patterns; therefore, we used 58 bins for them and one bin for all non-uniform patterns. The global description of the face image is obtained by concatenating all regional histograms. The overall value of LBPH can be represented in a histogram as (3):

$$H\,(k) = \sum_{i=0}^{n}\sum_{j=1}^{m} f\left(LBP_{P,R}\,(i,j)\,,k\right), k \in [0,k] \qquad (3)$$

where P is the sampling points and R is the radius.

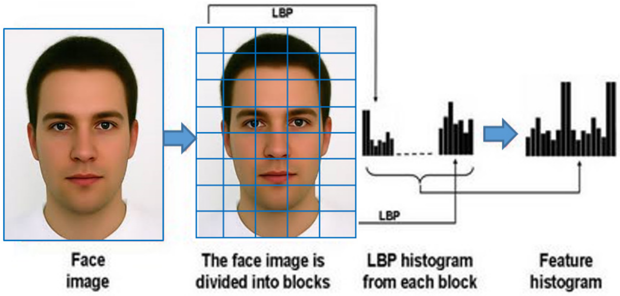
LBP
LBP
Face image
The face image is divided into blocks
LBP histogram from each block
Feature histogram

FIGURE 5. Face description with local binary patterns.

Figure 5 shows the process of getting the feature vector for each image, which will be fed to the classifier.

B. CLASSIFICATION METHODS

The K-Nearest-Neighbors (KNN) is one of the methods used in computer vision. Most of the KNN use Euclidean distances. However, it produces less accurate results than the other methods. Each distance method provides different levels of accuracy based on the problem domain. Therefore, the first contribution is to combine some of them to improve face recognition accuracy. The Mahalanobis distance method provides higher accuracy results than Minimum Distance depending on the covariance matrix between the two vectors (a and b) in the (4) [39].

$$Mahalanobis\,(a,b) = \sqrt{(ai-bi)^T\, S^{-1}\,(ai-bi)} \qquad (4)$$

where $\boldsymbol{S^{-1}}$ is the covariance matrix inverse.

Correlation distance classifier was introduced by Székely, Rizzo, and Bakirov in 2007 [40]. A valuable property is the measure of dependence equal zero and is sensitive to a linear relationship between two vectors.

$$Correlation\,(a,b) = \frac{Cov(a,b)}{\sigma_a\sigma_b} \qquad (5)$$

where $\boldsymbol{Cov}$ is the covariance and $\boldsymbol{\sigma_a}$ and $\boldsymbol{\sigma_b}$ are the standard deviations of $\boldsymbol{a}$ and $\boldsymbol{b}$.

The Euclidean distance method is considered the basis of many methods of similarity and dissimilarity. We use (6) to calculate the Euclidean distance between corresponding elements of the two vector space.

$$Euclidean\,(a,b) = \sqrt{\sum_{i=1}^{N}(a_i-b_i)^2} \qquad (6)$$

The Canberra distance method is a numerical measure of the distance between two points in a vector space, which is presented in (7):

$$Canberra\,(a,b) = \sum_{i=1}^{N}\frac{|a_i-b_i|}{|a_i|+|b_i|} \qquad (7)$$

The Manhattan distance method is another method to measure the distance between two vectors and is introduced in (8):

$$Manhattan\,(a,b) = \sum\nolimits_{i=1}^{N}|a_i-b_i| \qquad (8)$$

We used different distance methods to provide a variant dataset to improve the training in the neural network.

C. BACK-PROPAGATION NEURAL NETWORK

Computer vision requires powerful classification methods to achieve a high recognition system rate with low computing time and resources. BPNN classification is widely used for training the NN since BPNN is simple, efficient at computing the gradient descent, and straightforward to implement. Determining the size of the NN, the number of samples and the weights is a challenging task, and it is essential to fit the NN output. The BPNN is divided into three types of layers;

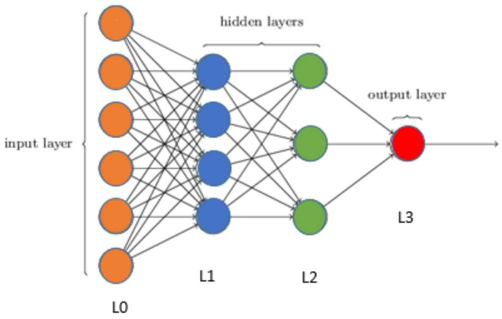

FIGURE 6. Three layers BPNN neural network.

the input layer, one or more hidden layers, and the predictable output layer as shown in Figure 6.

The common backpropagation algorithm can be described as follows:

1. The weights $w_{ij}^{[l]}$ and the thresholds $\vartheta_j^{[l]}$ are randomly initialized.

2. Compute the output of all layers using (9) after feeding the prepared training dataset $\boldsymbol{I_p}$ and the output dataset $\boldsymbol{O_p}$ to the NN.

$$y_{jp}^{[lC1]} = f\left(\sum_{i=1}^{N1} w_{ij}^{[l+1]} y_{ip}^{[l]} + \vartheta_j^{[l+1]}\right) \tag{9}$$

3. In each layer, compute the square root error as follows:

Equation (10) is used to calculate the square error at the output layer:

$$err_{jp}^{[L]} = f'\left(y_{jp}^{[L]}\right)\left(d_p - y_{jp}^{[L]}\right) \tag{10}$$

In the ith hidden layer (i = L-1, L-2 ...i):

$$err_{jp}^{[l]} = f'\left(y_j^{[l]}\right)\sum_{k=l}^{N_{l+1}} err_{kp}^{[l+1]} w_{jk}^{[l+1]} \tag{11}$$

4. The change in the weights between the input and the output will be calculated based on (12) and (13).

$$\vartheta_{ij}^{[l]}(n+1) = \vartheta_i^{[l]}(n) + \eta \cdot err_{jp}^{[l]} \tag{12}$$

$$w_{ij}^{[l]}(n+1) = w_{ij}^{[l]}(n) + \eta \cdot err_{jp}^{[l]} \cdot y_{ip}^{[l-1]} \tag{13}$$

5. Go back to step 2 if the mean-squared error is more than the threshold; otherwise, stop and print the weight value. There are many neuron activation functions used in the NN, and the sigmoidal function was used in our proposed system; it is shown in (14), and its derivatives are shown in (15).

$$f(x) = \frac{1}{1+e^{-x}} \tag{14}$$

$$f'(x) = (x)(1-f(x)) \tag{15}$$

J. Toms improved the backpropagation algorithm using the hybrid neuron because it is hard to reach the minimum mean-squared-error using the sigmoidal activation function in the big size NN system.

$$f(x) = \lambda \cdot s(x) + (1-\lambda) \cdot h(x) \tag{16}$$

where $h(x)$ is the hard-limiting function, which is defined in (17), and the derivatives are defined in (18)

$$h(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x < 0 \end{cases} \tag{17}$$

$$f'(x) = \lambda s(x)(1-s(x)) \quad \lambda \neq 0 \tag{18}$$

NN is often trapped in the local-minimum, and the learning speed is updated according to (19), where SSE is the Sum-Squared-Error. To make the NN faster and reach zero error, a coefficient α is added to the steepness of the sigmoidal function as defined in (20).

$$\lambda(n) = e^{-1/SSE} \tag{19}$$

$$f(x) = \frac{1}{1+e^{-\alpha x}} \tag{20}$$

The derivative is:

$$f'(x) = \alpha \cdot f(x)(1-f(x)) \tag{21}$$

III. CLASSICAL FACE RECOGNITION SYSTEM USING LBPH AND KNN

We implemented the existing classical face recognition system using LBPH and KNN as a reference point. Figure 7 explains the framework in detail. We used five distance methods, correlation, Euclidean, Canberra, Manhattan, and Mahalanobis, to find the distance between the testing images and the training images, and then we found the whole system accuracy based on that. The drawback of the classical method is the computing time since we have to compare the test image with all the training images with O(t(n)) complexity, where t(n) is the computing time of the distance. This experiment was applied on the datasets ORL and YALE with three different scenarios: 90% training and 10% testing, 70% training and 30% testing, and finally 50% training and 50% testing. The recognition rate (RR) is calculated using (22). Table 1 shows the accuracy result of this experiment.

$$RR(\%) = \frac{\textit{Number of correct match}}{\textit{Number of training set}} * 100 \tag{22}$$

TABLE 1. The accuracy of the classical face recognition system using LBPH and KNN.

	Recognition Rate (%)					
Method	50% 50%		70% 30%		90% 10%	
LBP +KNN Using:	Yale	ORL	Yale	ORL	Yale	ORL
Euclidean Distance	91.1	93	95.6	93.3	100	94
Correlation Distance	90	94	97.8	94.5	100	95
Canberra Distance	91.1	95	100	96	93.3	98
Manhattan Distance	94.4	96.5	95.6	97.3	100	99
Mahalanobis Distance	94.4	97	100	97.5	100	98

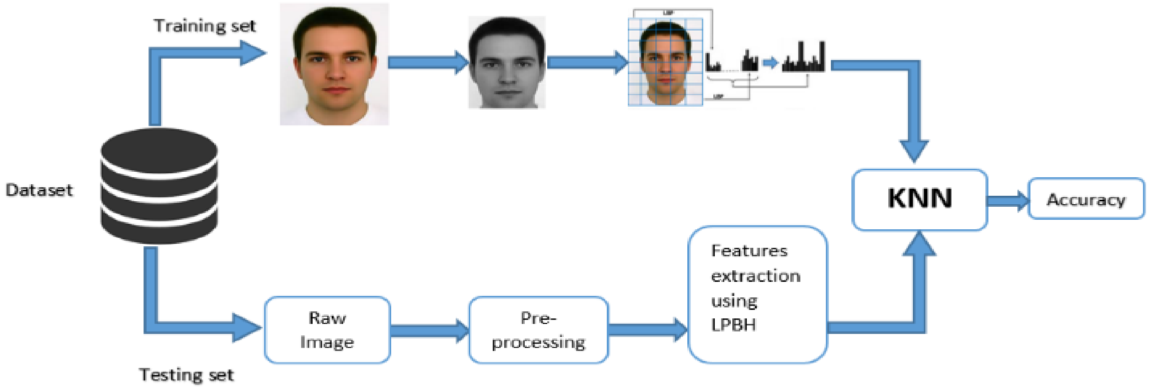

FIGURE 7. Classical recognition system using the KNN classifier method.

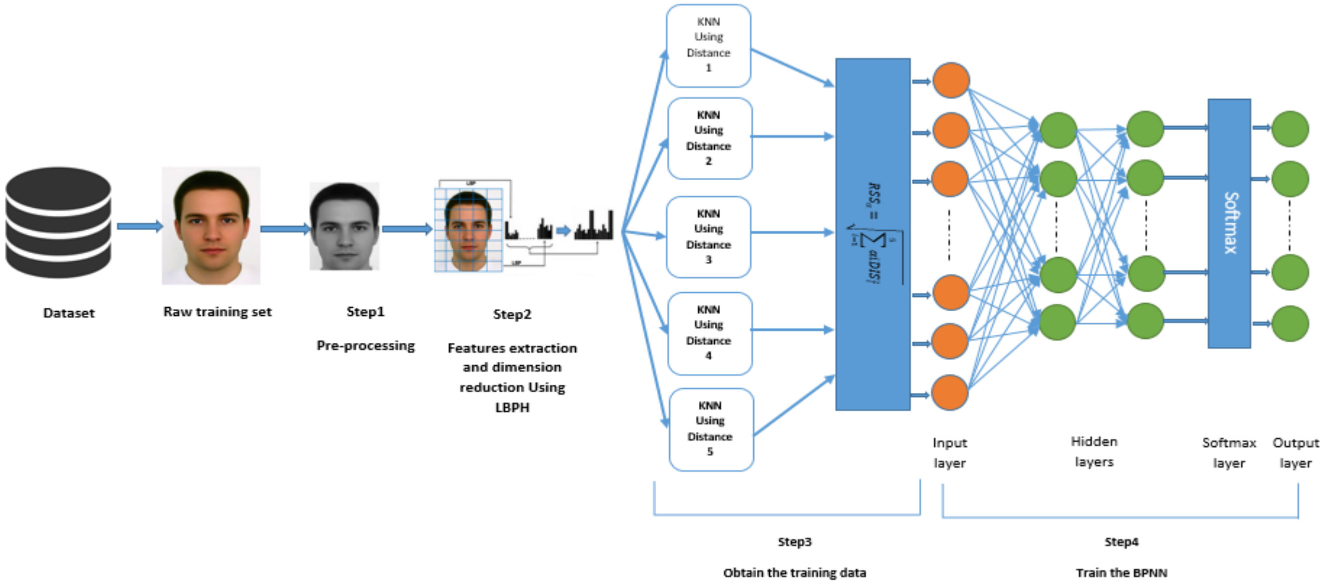

FIGURE 8. Proposed recognition system using LBPH descriptors, multi-KNN, and BPNN neural network.

IV. PROPOSED METHOD

We proposed in this work an enhanced human face recognition using LBPH descriptors, multi-KNN, and BPNN neural network. Figure 8 shows the proposed framework in detail. Our main contribution is based on the fact that obtaining a robust T-Dataset will help the BPNN to converge quickly with improved accuracy. We gathered a robust T-Dataset relying on the correlation between the training images, not the density of images. Our method is divided into five steps. In step one, we applied some of the preprocessing methods on the raw training images, including resizing and cropping using Haar-cascade detection, to eliminate the face background effect. Noise and illumination were reduced by converting the images to grayscale images and using histogram equalization to build a robust face recognition system; Figure 9 shows some of the preprocessing methods.

In Step 2, we extracted the most important local features from each image using the $LBP_{8,2}^{U2}$ descriptor and combined them into a global description using the histogram method. Here is how it is done:

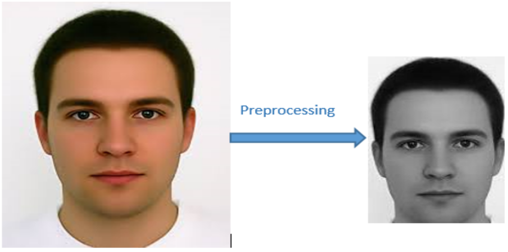

FIGURE 9. An example of preprocessing methods including cropping, resizing and Histogram equalization.

- We divided the images into 25 small cells after we tried different grid sizes. We found that the 5x5 grid gives us better performance with a reasonable time. Smaller grid sizes such as 4x4 provide fewer features ($4 \times 4 \times 59 = 944$) compared to ($5 \times 5 \times 59 = 1475$ features), which leads to less accuracy and perhaps to an under-fitting problem with the neural network training. A larger grid size provides more features; however, it increases the computing time with slight improvement in accuracy.

- We applied the LBP method on image pixels by thresholding the 3 × 3 neighborhood of each pixel with the center value and considering the result as a binary number.
- Finally, we applied the histogram method to concatenate the new cells description and obtain a new representation (25 cell * 59 dimension = 1475) for each training image, which helps to reduce the computation time.

Step 3 was added as an extra step to obtain a robust T-Dataset, which we used as an input to our BPNN instead of using the LBPH descriptor of each training image. As mentioned earlier, the T-Dataset is gathered based on the correlation between the new representations of all training images.

- Based on the LBPH presentation of each image, which we obtained from step 2, we calculated the distance between each training image with all other training images using five distance methods (Correlation, Euclidean, Canberra, Manhattan, and Mahalanobis).
- We tried different scenarios to achieve higher accuracy. First, we trained the BPNN using each distance method separately, and we achieved variant accuracy as shown in table 2. In another scenario, we combined the five distances using the square-root of the sum of the squares (RSS) (23) to provide a robust distinction T-Dataset in a reduced dimension.

$$RSS = \sqrt{\sum\nolimits_{i=1}^{5} DIS_i^2} \tag{23}$$

where DISi is one of the distance methods.

TABLE 2. Experiment results using LBPH descriptor, Multi-KNN and BPNN with 50% training set and 50% testing set.

Distance Method	Recognition Rate (%)	
	Yale	ORL
Euclidean	91.1	93.5
Correlation	91.1	95
Canberra	92.2	95.5
Manhattan	96.6	96.5
Mahalanobis	95.6	97
Proposed: $\sqrt{\sum_{i=1}^{5} DIS_i^2}$	96.7	97.5
Proposed: $\sqrt{\sum_{i=1}^{5} \alpha_i DIS_i^2}$	97.7	98

However, based on the classical face recognition experiment, each distance algorithm has an advantage over the other algorithms in different dimensions. Therefore, we modified (23) to (24) by adding a strength factor α to improve the accuracy result in the final scenario. Table 1 shows that the Mahalanobis and Manhattan distances have an advantage over the other distance methods. Therefore, we assign the strength factors as: Mahalanobis and Manhattan = 0.3, Canberra = 0.2, Correlation and Euclidean = 0.1.

$$RSS_\alpha = \sqrt{\sum_{i=1}^{5} \alpha_i DIS_i^2} \tag{24}$$

where $\sum_{i=1}^{5} \alpha_i = 1$.

- The KNN method is used to find the expected output for each training image, and we selected K=1 to avoid majority voting, which leads to incorrect votes since the dataset has an identical or nearly identical images. Our decision is based on the nearest neighbor, and we considered a match to have occurred if the nearest neighbor matches the source image as shown in Figure 10(b). Otherwise, it is considered a mismatch as shown in Figure 10(c).

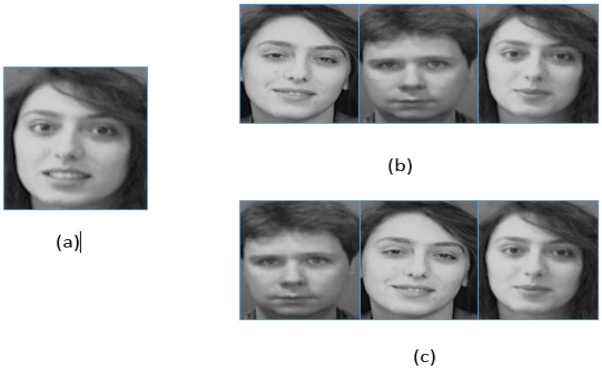

FIGURE 10. An example of the matching case and mismatching case image using KNN with Mahalanobis distance where K=1. (a) Test image (b) Matching case. (c) Mismatching case.

Table 3 shows an example of how to obtain the T-Dataset (column 6) and the expected output (column 7) for one of the training images (image X). We assume the training dataset has 200 images that represent 40 persons.

In Step 4, the BPNN parameters are set up and then the training begins. Our BPNN architecture contains an input layer followed by two fully connected hidden layers, followed by a soft-max classification layer.

- Set the number of layers and neurons.
- Set the number of iterations.
- Set the threshold value.
- Set the input matrix and the expected output from the previous step.
- Randomly initialize the weights and bias.
- Strat the training.

Finally, we test the accuracy of the system by:

- Applying steps 1 to 3 for each testing image.
- Feeding the testing image data to the trained BPNN and obtaining the predicted output.

TABLE 3. An example of how to obtain the new training data (column 6) and the expected output (column 7) for one of the training images (image X).

		Column1	Column2	Column3	Column4	Column5	Column6	Column7
	Distance **Between image X and**	Using Correlation	Using Euclidean	Using Canberra	Using Manhattan	Using Mahalanobis	Combine the result using: $RSS_\alpha = \sqrt{\sum_{i=1}^{5} \alpha_i DIS_i^2}$	expected Output Based on the KNN (K=1)
person1	Image1	0.39	0.34	0.71	0.34	0.56	1.091	1 (best match)
	Image2	0.31	0.32	0.67	0.42	0.49	1.033	
	Image3	0.41	0.28	0.77	0.29	0.39	1.037	
	Image4	0.29	0.29	0.64	0.27	0.48	1.533	
	Image5	0.72	0.54	0.85	0.48	0.79	1.544	
Person2	Image6	0.19	0.77	0.61	0.56	0.32	1.190	0
. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .
Person40	Image197	0.34	0.41	0.57	0.32	0.74	1.121	0
	Image198	0.51	0.57	0.74	0.21	0.64	1.259	
	Image199	0.41	0.27	0.78	0.41	0.71	1.233	
	Image200	0.39	0.54	0.92	0.69	0.69	1.497	

TABLE 4. Comparison Of the performance Of the proposed method to existing methods on the ORL dataset.

Reference	Year	Method	Number of Training Images per person	%Accuracy
[43]	2015	PCA+BPNN	N/A	88
[44]	2015	LDA	5	89.5
[45]	2015	Gabor + NMF	5	95
[46]	2014	PCA,LDA,DCT,ICA	5	85.5,88.5,91.5,87.5
[47]	2013	FKNN	5	87
[48]	2012	WT+PCA	8	95
[49]	2012	CASNN,FFNN	5	86.5,80
[50]	2011	PCA-DCT-Corr-PIFS	N/A	86.8
Proposed	**2018**	**LBPH, multi-KNN, and BPNN**	**5**	**98**

- Based on the image label, we know whether the prediction is correct.
- Finally, the overall system accuracy is calculated.

V. EXPERIMENTS

We applied our method on the small public datasets ORL [40] and Yale [41] as concepts of truth and to understand the benefits of using a strong descriptor, which provides training data with a clear distinction.

Then, we applied our proposed method on one of the state-of-the-art datasets, the LFW. We evaluated the proposed method on a restricted evaluation category.

A. EXPERIMENT ON YALE DATASET AND ORL DATASET

Olivetti Research Laboratory Dataset (ORL) [41] represents images of 40 different people with ten different pictures of each person. A total of 400 face images are divided into 200 images for training and 200 images for testing. The 400 images are in grayscale, and the size is 92 × 112 pixels with variant expressions, timing, pose, and gender. Figure 11 shows ORL sample images.

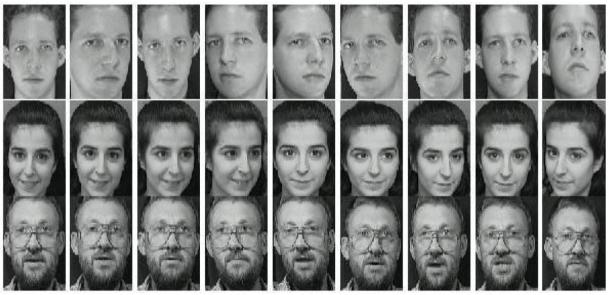

FIGURE 11. A sample of the ORL dataset.

The Yale face dataset has a total of 165 face images that represent 15 different persons with 11 images per person [42].

Different facial expressions, genders, and light configurations are shown, as well as images with and without eyeglasses. The 165 images are in a grayscale domain, and the images are resized to 92 x 112 pixels after the face is cropped using Haar-cascade detection. We used 75 images to train the NN and the remaining images to test the system.

In this experiment, the BPNN has four layers, an input layer (200 inputs in the ORL experiment and 75 in the Yale experiment) and two fully-connected hidden layers (100 nodes in the ORL experiment and 40 in the Yale experiment). The output layer (40 classes in the ORL experiment and 15 classes in the Yale Experiment) is followed by a softmax classifier. The training rate is 0.1, and we trained the network up to 8000 iterations and took 25 hours to train the BPNN. The experiments ran on a personal laptop with Intel CoreI7 CPU @ 3GHz.

We achieved a higher recognition rate using an LBPH descriptor, multi-KNN, and BPNN. Table 2 shows the result for each distance separately; then, it shows the result of combining the distance methods using (23) then (24). We achieved 97.7% accuracy on the Yale dataset with only two mismatches out of 165 images and 98% accuracy with four mismatches out of 200 testing images on a 50% training set and 50% testing scenario. Table 4 shows a comparison between the proposed framework and the other existing methods on the ORL dataset.

B. EXPERIMENT ON LFW DATASET

In 2007, LFW was created by Huang *et al.* [51]; it has had an enormous impact on the face recognition field. The LFW dataset contains 13,322 images of human faces captured from the web using a Viola-Jones face detector, and each image is labeled with the name of the person. The image size is 250 × 250 pixels, and most of them are in color. The dataset represents 5749 individuals, and 1680 of them have two or more images; the remaining individuals have only one image. LFW captured the images under unconstrained conditions (expression, lighting source, and pose). Figure 12 shows LFW sample images.

FIGURE 12. A sample of the LFW dataset.

Step	Pseudocode
1) Preprocessing	For each image in the training dataset and testing dataset: A1) Convert the image to grayscale. B1) Detect the face using the Haar-cascade detection method. C1) Crop the face. D1) Resize the image to the size we want. E1) Apply the histogram equalizer.
2) Feature extraction and domain reduction	For each image in the training dataset and testing dataset: A2) Divide the image into 25 cells (5*5). B2) Apply the LBP approach on each cell. C2) Apply the histogram on each cell. D2) The new representation for each cell will be 59 points. E2) Concatenate the new representation for the 25 cells to find the new representation for the entire image (59*25= 1475 points).
3) Obtaining the training dataset to train the network:	For each training image in the training dataset: A3) Find the distance between the training image and all training images using five distance methods. B3) Apply $RSS = \sqrt{\sum_{i=1}^{5} DIS_i^2}$ or $RSS_\alpha = \sqrt{\sum_{i=1}^{5} \alpha_i DIS_i^2}$. C3) Obtain the training data that we will use to train the NN (step3 in Figure 8). D3) Find the expected output for each image based on the KNN where K=1.
4) Training the NN:	A4) Set the number of layers and neurons. B4) Set the number of iterations. C4) Set the threshold value. D4) Set the input matrix (from step C3), and the expected output (from step D3). E4) Randomly initialize the weights and bias F4) Train the NN.
5) Testing	For each test image: A5) Apply step 1, 2, and 3. In step 3, there is no need to find the expected output. B5) Feed the new representation of the test image to the trained NN. C5) Obtain the predicted output that indicates which test image belongs to which person. D5) Based on the label, we will know whether the prediction is correct.

FIGURE 13. The Pseudocode of the proposed approach.

The LFW is divided into two views: View 1 is used for training and method implantation and contains 2200 pair images for training and 1000 pair images for testing. View 2 includes 6000 pair images divided into 10 sets for cross-validation, which is used for the performance evaluation. The evaluation set used in this experiment is "restricted images," with no outside data, using a leave-one-out cross-validation structure [51]. The total performance of the proposed method is calculated using the mean classification accuracy û and standard error of the mean SE.

To eliminate the background effect, we detected the face using the Haar-cascade detection algorithm; then, we cropped the center face area and resized it to 92 x 112. All the images are converted to grayscale followed by a histogram equalizer.

As we mentioned in the proposed method section, the new T-Dataset generated from the multi-LBPH descriptor is to be fed as the input data for our BPNN, which has two hidden layers and a softmax classification output layer. The advantages of using the multi-LBPH descriptor and KNN on the LFW are as follows: (1) A robust representation based on the correlation between training images is used. (2) There is no need for large amounts of data since we have robust distinction data. (3) The training time is faster since we used a small data description for each image. Figure 13 shows the pseudocode of the proposed approach; however, the testing process in the LFW experiment is slightly different. We find the match image for both images separately, and if both predict the same person, verification occurs.

TABLE 5. The result comparison of the proposed method and state-of-the-art methods on the LFW dataset under the restricted setting. No outside data.

Method	û ± S_E
Eigenfaces [1] , original	0.6002 ± 0.0079
Nowak[52], original	0.7245 ± 0.0040
Nowak, funneled[52]	0.7393 ± 0.0049
Hybrid descriptor-based[53], funneled	0.7847 ± 0.0051
3x3 Multi-Region Histograms [54]	0.7295 ± 0.0055
Pixels/MKL[55], funneled	0.6822 ± 0.0041
V1-like/MKL[55], funneled	0.7935 ± 0.0055
APEM (fusion)[56], funneled	0.8408 ± 0.0120
MRF-MLBP[57]	0.7908 ± 0.0014
Fisher vector faces[58]	0.8747 ± 0.0149
Eigen-PEP[59]	0.8897 ± 0.0132
MRF-Fusion-CSKDA[60]	0.9589 ± 0.0194
Proposed	**0.9571 ± 0.0082**

Table 5 shows the results of our proposed method and a comparison with existing state-of-the-art methods. We achieved **95.71%** accuracy, which is comparable to state-of-the-art methods, and we achieved the second highest accuracy after that in [60]. However, our standard error is less than that in [60], which leads to higher overall accuracy.

The training was finished within a week on a personal laptop with Intel CoreI7 CPU @ 3GHz using Microsoft visual studio C#.

VI. CONCLUSION

We proposed an enhanced framework for human face recognition using an LBPH descriptor, multi-KNN, and BPNN neural network. The novel LBPH descriptor and multi-KNN has helped to provide a training dataset with distinction patterns based on the correlation between the original training images. The newly obtained T-Dataset helped the BPNN to converge faster and with higher accuracy. This was achieved by combining the distance methods, as each distance method has an advantage over the other methods, which strengthens the whole system. We achieved higher accuracy and reduced the computation time. In addition, we outperformed current state-of-the-art frameworks. Table 4 and 5 show a comparison of the proposed framework and the other methods on the ORL dataset and LFW dataset. We have not applied modern NN to prove that we can achieve higher accuracy even with traditional features extraction and domain reduction methods using a correlated training dataset between images. However, in future work we plan to use different feature extraction methods such as convolutional NN and compare them to the current results.

REFERENCES

[1] R. Chellappa, C. L. Wilson, and S. Sirohey, "Human and machine recognition of faces: A survey," *Proc. IEEE*, vol. 83, no. 5, pp. 705–740, May 1995.

[2] H. Fang *et al.*, "Facial expression recognition in dynamic sequences: An integrated approach," *Pattern Recognit.*, vol. 32, no. 3, pp. 740–748, 2014.

[3] Z. Zeng, M. Pantic, G. I. Roisman, and T. S. Huang, "A survey of affect recognition methods: Audio, visual, and spontaneous expressions," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 1, pp. 39–58, Jan. 2009.

[4] M. Chihaoui, A. Elkefi, W. Bellil, and C. B. Amar, "A survey of 2D face recognition techniques," *MDPI Comput.*, vol. 5, no. 4, p. 21, 2016.

[5] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 711–720, Jul. 1997.

[6] T. Ahonen, A. Hadid, and M. Pietikäinen, "Face recognition with local binary patterns," in *Proc. Eur. Conf. Comput. Vis.*, 2004, pp. 469–481.

[7] T. Chen, X. Zhao, L. Dai, L. Zhang, and J. A. Wang, "Novel texture feature description method based on the generalized Gabor direction pattern and weighted discrepancy measurement model," *MDPI Symmetry*, vol. 8, no. 11, p. 109, 2016.

[8] Y. Xiao, J. Wu, and J. Yuan, "CENTRIST: A multi-channel feature generation mechanism for scene categorization," *IEEE Trans. Image Process.*, vol. 23, no. 2, pp. 823–836, Feb. 2014.

[9] C. Heng, S. Yokomitsu, Y. Matsumoto, and H. Tamura, "Shrink boost for selecting multi-LBP histogram features in object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 3250–3257.

[10] X. Zhao and S. Zhang, "Facial expression recognition based on local binary patterns and kernel discriminant Isomap," *MDPI Sensors*, vol. 11, no. 10, pp. 9573–9588, 2011.

[11] L. Huang, C. Chen, W. Li, and Q. Du, "Remote sensing image scene classification using multi-scale completed local binary patterns and Fisher vectors," *MDPI Remote Sens.*, vol. 8, no. 6, p. 483, 2016.

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

[13] X. Tan and B. Triggs, "Enhanced local texture feature sets for face recognition under difficult lighting conditions," *IEEE Trans. Image Process.*, vol. 19, no. 6, pp. 1635–1650, Jun. 2010.

[14] J. Trefny and J. Matas, "Extended set of local binary patterns for rapid object detection," in *Proc. Comput. Vis. Winter Workshop*, 2010, pp. 1–7.

[15] B. Zhang, S. Shan, X. Chen, and W. Gao, "Histogram of Gabor phase patterns (HGPP): A novel object representation approach for face recognition," *IEEE Trans. Image Process.*, vol. 16, no. 1, pp. 57–68, Jan. 2007.

[16] W. Zhang, S. Shan, W. Gao, X. Chen, and H. Zhang, "Local Gabor binary pattern histogram sequence (LGBPHS): A novel non-statistical model for face representation and recognition," in *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 1. Oct. 2005, pp. 786–791.

[17] M. Turk and M. Pentland, "Eigenfaces for recognition," *J. Cognit. Neurosci.*, vol. 3, no. 1, pp. 6–71, 1991.

[18] S. Wang and P. Liu, "A new feature extraction method based on the information fusion of entropy matrix and covariance matrix and its application in face recognition," *MDPI Entropy*, vol. 17, no. 7, pp. 4664–4683, 2015.

[19] M. Abuzneid and A. Mahmood, "Performance improvement for 2-D face recognition using multi-classifier and BPN," in *Proc. IEEE Long Island Syst., Appl., Technol. Conf. (LISAT)*, New York, NY, USA, Apr. 2016, pp. 1–7.

[20] M. S. Bartlett, J. R. Movellan, and T. J. Sejnowski, "Face recognition by independent component analysis," *IEEE Trans. Neural Netw.*, vol. 13, no. 6, pp. 1450–1464, Nov. 2002.

[21] B. A. Draper, K. Baek, M. S. Bartlett, and J. R. Beveridge, *Recognizing Faces With PCA and ICA. Computer Vision and Image Understanding: Special Issue on Face Recognition*. San Diego, CA, USA: Academic, 2003, pp. 115–137.

[22] I. Dagher and R. Nachar, "Face recognition using IPCA-ICA algorithm," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 6, pp. 996–1000, Jun. 2006.

[23] J. Lu, K. N. Plataniotis, and A. N. Venetsanopoulos, "Face recognition using LDA-based algorithms," *IEEE Trans. Neural Netw.*, vol. 14, no. 1, pp. 195–200, Jan. 2003.

[24] W.-S. Zheng, J.-H. Lai, and P. C. Yuen, "GA-Fisher: A new LDA-based face recognition algorithm with selection of principal components," *IEEE Trans. Syst. Man, Cybern. B, Cybern.*, vol. 35, no. 5, pp. 1065–1078, Oct. 2005.

[25] A. M. Martínez and A. C. Kak, "PCA versus LDA," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 2, pp. 228–233, Feb. 2001.

[26] H. M. Moon, D. Choi, P. Kim, and S. B. Pan, "LDA-based face recognition using multiple distance training face images with low user cooperation," in *Proc. IEEE Int. Conf. Consum. Electron. (ICCE)*, Las Vegas, NV, USA, Jan. 2015, pp. 7–8.

[27] D. Zhang, Z. Zhou, and S. Chen, "Diagonal principal component analysis for face recognition," *Pattern Recognit.*, vol. 39, no. 1, pp. 140–142, 2006.

[28] T. Mandal, Q. Wu, and Y. Yuan, "Curvelet based face recognition via dimension reduction," *Signal Process.*, vol. 89, no. 3, pp. 2345–2353, 2009.

[29] M.-H. Yang, "Kernel Eigenfaces vs. kernel Fisherfaces: Face recognition using kernel methods," in *Proc. 5th IEEE Int. Conf. Autom. Face Gesture Recognit. (RGR)*, May 2002, pp. 215–220.

[30] H. Boughrara, M. Chtourou, C. B. Amar, and L. Chen, "Face recognition based on perceived facial images and multilayer perceptron neural network using constructive training algorithm," *IET Comput. Vis.*, vol. 8, no. 6, pp. 729–739, 2014.

[31] M. J. Er, W. Chen, and S. Wu, "High-speed face recognition based on discrete cosine transform and RBF neural networks," *IEEE Trans. Neural Netw.*, vol. 16, no. 3, pp. 679–691, May 2005.

[32] Y. Xu, F. Liang, G. Zhang, and H. Xu, "Image intelligent detection based on the Gabor wavelet and the neural network," *MDPI Symmetry*, vol. 8, no. 11, p. 130, 2016.

[33] K. Lee, Y. Chung, and H. Byun, "SVM-based face verification with feature set of small size," *Electron. Lett.*, vol. 38, no. 15, pp. 787–789, Jul. 2002.

[34] R. Duda, P. Hart, and D. Stork, *Pattern Classification*, 2nd ed. New York, NY, USA: Wiley, 2001.

[35] N. A. Mahmon, N. Ya'acob, and A. L. Yusof, "Differences of image classification techniques for land use and land cover classification," in *Proc. IEEE CSPA*, Mar. 2015, pp. 90–94.

[36] H. Othman and T. Aboulnasr, "A separable low complexity 2D HMM with application to face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 10, pp. 1229–1238, Oct. 2003.

[37] A. A. Mohammed, R. Minhas, Q. M. Wu, and M. A. Sid-Ahmed, "Human face recognition based on multidimensional PCA and extreme learning machine," *Pattern Recognit.*, vol. 44, nos. 10–11, pp. 2588–2597, 2011.

[38] T. Ojala, M. Pietikäinen, and D. Harwood, "A comparative study of texture measures with classification based on featured distributions," *Pattern Recognit*, vol. 29, no. 1, pp. 51–59, 1996.

[39] K. Perumal and R. Bhaskaran, "Supervised classification performance of multispectral images," *J. Comput.*, vol. 2, no. 2, pp. 124–129, 2010.

[40] G. J. Szekély, M. L. Rizzo, and N. K. Bakirov, "Measuring and testing independence by correlation of distances," *Ann. Stat.*, vol. 35, no. 6, pp. 2769–2794, 2007.

[41] *ORL Dataset*. Accessed: Jan. 23, 2017. [Online]. Available: http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html

[42] *Yale Face Dataset*. Accessed: Jan. 23, 2017. [Online]. Available: http://vision.ucsd.edu/content/yale-face-database

[43] Rajath Kumar M. P., Keerthi Sravan R., and K. M. Aishwarya, "Artificial neural networks for face recognition using PCA and BPNN," in *Proc. IEEE Region Conf. TENCON*, Macao, China, Nov. 2015, pp. 1–6. [Online]. Available: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7373165&isnumber=7372693

[44] W. Ge, W. Quan, and C. Han, "Face description and identification using histogram sequence of local binary pattern," in *Proc. 7th Int. Conf. Adv. Comput. Intell. (ICACI)*, Wuyi, China, Mar. 2015, pp. 415–420.

[45] F. Purnomo, D. Suhartono, M. Shodiq, A. Susanto, S. Raharja, and R. W. Kurniawan, "Face recognition using Gabor wavelet and non-negative matrix factorization," in *Proc. SAI Intell. Syst. Conf. (IntelliSys)*, London, U.K., Nov. 2015, pp. 788–792.

[46] F. A. Bhat and M. A. Wani, "Performance comparison of major classical face recognition techniques," in *Proc. 13th Int. Conf. Mach. Learn. Appl.*, Detroit, MI, USA, 2014, pp. 521–528.

[47] T. Ayyavoo and J. S. Jayasudha, "Face recognition using enhanced energy of discrete wavelet transform," in *Proc. Int. Conf. Control Commun. Comput. (ICCC)*, Thiruvananthapuram, India, Dec. 2013, pp. 415–419.

[48] C.-L. Fan, X.-T. Chen, and N.-D. Jin, "Research of face recognition based on wavelet transform and principal component analysis," in *Proc. 8th Int. Conf. Natural Comput.*, Chongqing, China, 2012, pp. 575–578.

[49] A. J. Dhanaseely, S. Himavathi, and E. Srinivasan, "Performance comparison of cascade and feed forward neural network for face recognition system," in *Proc. Int. Conf. Softw. Eng. Mobile Appl. Modeling Develop. (ICSEMA)*, Chennai, India, 2012, pp. 1–6.

[50] M. A. Lone, S. M. Zakariya, and R. Ali, "Automatic face recognition system by combining four individual algorithms," in *Proc. Int. Conf. Comput. Intell. Commun. Netw.*, Gwalior, India, 2011, pp. 222–226.

[51] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," Dept. Comput. Sci., Univ. Massachusetts, Amherst, MA, USA, Tech. Rep. 07-49, Oct. 2007. [Online]. Available: http://vis-www.cs.umass.edu/~gbhuang/

[52] A. Shekhovtsov, I. Kovtun, and V. Hlavac, "Efficient MRF deformation model for non-rigid image matching," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2007, pp. 1–6.

[53] J. Zhang, M. Marszałek, S. Lazebnik, and C. Schmid, "Local features and kernels for classification of texture and object categories: A comprehensive study," *Int. J. Comput. Vis.*, vol. 73, no. 2, pp. 213–238, Jun. 2007.

[54] E. Nowak and F. Jurie, "Learning visual similarity measures for comparing never seen objects," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2007, pp. 1–8.

[55] L. Wolf, T. Hassner, and Y. Taigman, "Descriptor based methods in the wild," in *Proc. Faces Real-Life Images Workshop ECCV*, Oct. 2008, pp. 1–14.

[56] C. Sanderson and B. C. Lovell, "Multi-region probabilistic histograms for robust and scalable identity inference," in *Advances in Biometrics* (Lecture Notes in Computer Science), vol. 5558, M. Tistarelli and M. S. Nixon, Eds. Berlin, Germany: Springer-Verlag, 2009, pp. 199–208. [Online]. Available: http://conradsanderson.id.au/pdfs/sanderson_icb_2009.pdf and https://link.springer.com/content/pdf/bfm%3A978-3-642-01793-3%2F1.pdf

[57] S. R. Arashloo and J. Kittler, "Efficient processing of MRFs for unconstrained-pose face recognition," in *Proc. IEEE 6th Int. Conf. Biometr., Theory, Appl. Syst. (BTAS)*, Sep./Oct. 2013, pp. 1–8.

[58] N. Pinto, J. J. DiCarlo, and D. D. Cox, "How far can you get with a modern face recognition test set using only simple features?" in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 2591–2598.

[59] H. Li, G. Hua, X. Shen, Z. Lin, and J. Brandt, "Eigen-PEP for video face recognition," in *Proc. 12th Asian Conf. Comput. Vis. (ACCV)*, 2014, pp. 17–33.

[60] S. R. Arashloo and J. Kittler, "Class-specific kernel fusion of multiple descriptors for face verification using multiscale binarised statistical image features," *IEEE Trans. Inf. Forensics Security*, vol. 9, no. 12, pp. 2100–2109, Dec. 2014.

MOHANNAD A. ABUZNEID (M'16) received the B.S. degree in computer engineering from Yarmouk University in 2006 and the M.S. degree in computer engineering from the University of Bridgeport (UB), Bridgeport, CT, USA, in 2008.

He is currently pursuing the Ph.D. degree with the School of Engineering, UB.

He is currently a Technical Analyst with the Yale-New Haven Health System. He is also a Graduate Assistant with the School of Engineering, UB. His research interests have included image processing, deep learning, computer vision, and face recognition. He actively participated as a member of the Technical Program Committee of the International Joint Conference on Computer, International, and System Science, and Engineering, which was organized by the School of Engineering, UB, from 2010 to 2014.

Mr. Abuzneid has been a member of the Jordanian Engineers Association since 2006. He received the Special Achievement Award from UB in 2008. He was also recognized by the Honor Society of the Upsilon Pi Epsilon Organization in 2008.

AUSIF MAHMOOD (SM'82) received the M.S. and Ph.D. degrees in electrical and computer engineering from Washington State University, USA. He is currently the Chair Person of the Computer Science and Engineering Department and a Professor with the Computer Science and Engineering Department and the Electrical Engineering Department, University of Bridgeport, Bridgeport, CT, USA. His research interests include parallel and distributed computing, computer vision, deep learning, and computer architecture.