

Low-Complexity Face Recognition using a Multilevel DWT and Two States of Continuous HMM to recognize Noisy Images

Hameed R. Farhan¹, Mahmud H. Al-Muifraje², Thamir R. Saeed²

¹Department of Electrical and Electronic Engineering, University of Kerbala, Kerbala, Iraq
hrf64ha@ieee.org

²Department of Electrical Engineering, University of Technology, Baghdad, Iraq
drmahmood6@gmail.com
thamir_rashed@uotechnology.edu.iq

Abstract—Face recognition has become an important subject in modern life, especially in security and surveillance applications. This work introduces a face recognition method, which is characterized by high-speed, low-complexity, and high-efficiency in a noisy environment. The performance of this method is greatly improved by using a median filter, such that each image is filtered to eliminate the influence of noise and light illumination. Multiple levels of discrete wavelet transform are applied to the filtered image to reduce size and eliminate further noise. Subsequently, the resultant image is scanned using a window with a predefined overlap in raster fashion to construct a sequence of observation vectors used as the basis of a model. The model consists of two states of a continuous hidden Markov model, a unique model in the face recognition field that has not yet been used by other researchers, which interprets the novelty and low complexity of the method. In spite of the presence of 0.15, 0.4, and 0.25 measurement values of impulse noise density in images stored in the ORL, Yale, and EURECOM Kinect face databases, respectively, the proposed work has achieved a recognition rate of 100%.

Keywords-Face recognition, Median filter, Discrete wavelet transform, Continuous hidden Markov model, Noisy images

I. INTRODUCTION

Face recognition has recently received considerable attention due to its important applications in many areas and fields such as credit card verification, security, and surveillance. The advantage of face recognition over other biometric techniques is the mechanism to recognize persons, which is frequently performed without requiring that the person have physical contact with the device, e.g. use of palm or fingers. The face image can be captured by a camera without informing the target person, especially in the case of criminal surveillance and apprehension.

The methods introduced to recognize faces vary relative to the procedure used for feature extraction and type of classifier used. The artificial neural network [1], principle component analysis (PCA) [2], independent component analysis [3], linear discriminant analysis (LDA) [4], support vector machine [5] and K-nearest neighbour [6] are the most widely used methods. In addition, the hidden Markov model (HMM) [7] has been successfully used in face recognition during the last two decades. The advantage of applying HMM to face recognition is the flexibility of the selection of training models, such that it is easy to add or remove individuals. There is no need for re-training the overall system; however, a model with updated images must be re-trained.

For dimensionality reduction, a particular feature extraction method is applied to form a sequence of observation vectors to model faces. Thereafter, for each individual, one HMM model is constructed by training a specified set of images for that individual. The training process depends on the estimation of the HMM parameters [8], such that the process is iterated several times before it converges. In the testing process, the algorithm computes the probability of the observations of unknown face image calibrated with the parameters derived from the training. The unknown face belongs to the person whose model computes the highest probability.

The disadvantages of using more states of the HMM are: high computational complexities, large memory requirements, and low processing speed. The proposed work addresses the unfavourable features by using only two states of the HMM.

In Section II, a brief description of the related work is introduced. Section III presents theoretical backgrounds of the main topics. The proposed method is described in Section IV. Experimental results are discussed in Section V. Finally, Section VI documents our conclusions and future work.

II. RELATED WORK

Several attempts to solve the face recognition problem using the HMM have been presented in the last 20 years. These approaches differ in the structure of the HMM and the technique to generate the observation vectors. In [9], the authors proposed the first HMM face recognition system. They used two types of a 1D-HMM; ergodic and left-to-right, with eight and five states, respectively. Pixel intensities are the elements of the observation vectors produced from a given image. The disadvantage of using pixel intensities is the associated high level of computational complexity; therefore, the tendency is to employ low-complexity methods. The discrete wavelet transform (DWT) [10], local binary pattern (LBP) [11], singular value decomposition (SVD) [12, 13, 14] and discrete cosine transform (DCT) [15] are the most widely methods used for dimensionality reduction.

Different types of wavelets have been used in the application of HMM to face recognition. In [10, 16, 17], the Haar wavelet was used to extract features from face images. The 3rd level of wavelet transform (WT) and a continuous 1D-HMM classifier with five states and two mixture components were used in [18]. The sensitivity of the system to white Gaussian noise was examined using two types of Daubechies wavelets: db4 and db10. In [19], the authors used a DWT instead of a DCT, because of their belief that the DCT may not be the best technique to describe an image and wavelets may be a better choice. Three types of wavelets were used in [20]: Haar, Biorthogonal 9/7, and Coif3 wavelets with structural HMM. In [21], the authors used a WT and a PCA with a five-state HMM classifier. A discrete Gabor wavelet transform (DGWT) was used in [22] to obtain observation vectors from face images with a seven-state 2D-HMM. The same authors used a DWT and a seven-state 1D-HMM in [23], but the experiments were conducted on a different database. In [24], the authors presented an HMM face recognition system based on Gabor wavelet for features extraction and an LDA for dimensionality reduction. They used 15 states for the ORL database and 9 states for the Yale database.

III. BACKGROUND

The theoretical backgrounds of main topics that are relevant to the proposed work are explained in the following sections.

A. Median Filter

A median filter is a special kind of nonlinear spatial filter that is used to improve image processing such as noise elimination. It surrounds the pixels by sliding the center point of a mask window over the image from top to bottom and left to right in steps of one pixel. Then, the window matrix is converted to a 1D vector, and all elements are arranged in descending or ascending order, where the mid-value replaces the center of the window [25]. The following example illustrates the operation of the median filter using a mask window of size (3 × 3) pixels. The following matrix represents a sample of block image before filtering:

$$I = \begin{bmatrix} 64 & 76 & 80 & 53 & 34 & 72 \\ 60 & 61 & 60 & 58 & 78 & 78 \\ 63 & 80 & 79 & 84 & 90 & 73 \\ 53 & 71 & 71 & 59 & 46 & 44 \\ 60 & 50 & 50 & 36 & 36 & 47 \\ 41 & 32 & 33 & 37 & 38 & 48 \end{bmatrix}$$

The mask window is overlaid on I , generating the pixel values indicated by the red numbers in positions ($I_{11} - I_{33}$), and the center value of the window is 61. Thereafter, the window is converted to a descending vector V :

$$V = [80 \ 80 \ 79 \ 76 \ 64 \ 63 \ 61 \ 60 \ 60]$$

The mid-value of V is 64, which represents the median value of that window, and it replaces the previous value (i.e. 61). After median filtering, the matrix I will be:

$$I = \begin{bmatrix} 0 & 60 & 58 & 53 & 53 & 0 \\ 61 & 64 & 76 & 78 & 73 & 72 \\ 60 & 63 & 71 & 71 & 73 & 46 \\ 53 & 63 & 71 & 59 & 47 & 44 \\ 41 & 50 & 50 & 38 & 44 & 38 \\ 0 & 33 & 33 & 36 & 36 & 0 \end{bmatrix}$$

B. Discrete Wavelet Transform (DWT)

The DWT was invented by Alfred Haar in 1909; it decomposes a data signal into approximation and detailed coefficients. The DWT applies two functions: scaling and wavelet [25]. The scaling function is associated with a low-pass filter $h[n]$, while a high-pass filter $g[n]$ corresponds to the wavelet function. The wavelet function $\psi(t)$ and scaling function $\varphi(t)$ defined by the Haar wavelet are, respectively, described as follows:

$$\psi(t) = \begin{cases} 1 & 0 \leq t < 0.5, \\ -1 & 0.5 \leq t < 1, \\ 0 & \text{elsewhere.} \end{cases} \quad (1)$$

$$\varphi(t) = \begin{cases} 1 & 0 \leq t < 1, \\ 0 & \text{elsewhere.} \end{cases} \quad (2)$$

Filter banks decompose the signal into different frequency levels using multi stages of high-pass and low-pass filters. The desired resolution is obtained by further splitting the signal at the $h[n]$ stage. The procedure of a 1D-DWT can be viewed as a convolution process between the signal $x[n]$ and the impulse responses $h[n]$ and $g[n]$ of the low-pass and high-pass filters, respectively. The approximation coefficients are obtained as follows [26];

$$y_l[n] = x[n] * h[n] \quad (3)$$

$$y_l[n] = \sum_{k=-\infty}^{\infty} x[n]h[n - k] \quad (4)$$

In the same manner, the detailed coefficients are generated using the following convolution equations:

$$y_h[n] = x[n] * g[n] \quad (5)$$

$$y_h[n] = \sum_{k=-\infty}^{\infty} x[n]g[n - k] \quad (6)$$

The outputs of each filter are then downsampled by 2, and the corresponding convolution expressions can be written as follows;

$$y_l[n] = \sum_{k=-\infty}^{\infty} x[n]h[2n - k] \quad (7)$$

$$y_h[n] = \sum_{k=-\infty}^{\infty} x[n]g[2n - k] \quad (8)$$

The image is a 2D signal and two separate 1D-DWT instances that must be used to achieve a 2D-DWT. Initially, the decomposition is applied to the rows of the image, and then, the outputs are decomposed along the columns. This operation splits the image into four sub-images, denoted by LL, LH, HL, and HH. The multi-resolution level is obtained by further splitting LL into new four sub-images, and so on, as shown in Fig. 1.

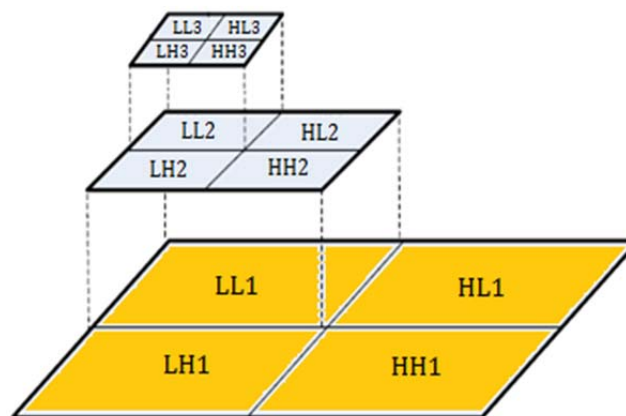


Fig. 1. 2D multilevel DWT

C. Hidden Markov Model (HMM)

The HMM is a one-dimensional model, which is associated with interconnected non-observable (hidden) states, manifested by the observable vector sequence. The parameters of an HMM are [7]:

- N : Number of states in the model, where the interconnections of the states depend on the topology of the model.
- M : Number of symbols in each state, or the range of values employed in the observation vectors. The individual symbols are denoted as $V = \{v_1, v_2, \dots, v_M\}$.
- A : the transition matrix ($N \times N$) which comprises the probabilities of the state transitions, where

$$A = \{a_{ij}\}, \quad 1 \leq i, j \leq N \tag{9}$$

$$\sum_{j=1}^N a_{ij} = 1 \tag{10}$$

- B : the emission matrix ($N \times M$) specifies the probability distribution of the M observation symbols in each of the N states such that:

$$B = \{b_j(k)\}, \quad 1 \leq j \leq N, \quad 1 \leq k \leq M \tag{11}$$

$$\sum_{j=1}^N b_j(k) = 1 \tag{12}$$

Each observation vector (O_t) consists of T elements, and $O = (O_1, O_2, \dots, O_T)$ that defines the observation sequence at T different observation instances.

- π : the initial state probability vector ($N \times 1$), $\pi = \{\pi_i\}$, where

$$\sum_{i=1}^N \pi_i = 1 \tag{13}$$

The model is indicated by the following notation:

$$\lambda = (A, B, \pi) \tag{14}$$

For continuous output observation densities of the hidden states, the observation matrix elements satisfy the following equation:

$$b_j(O) = \sum_{m=1}^K c_{jm} \mathcal{G}(O, \mu_{jm}, U_{jm}), \quad 1 \leq j \leq N, \tag{15}$$

where O is the observation vector, c_{jm} is the coefficient of the m^{th} mixture in state j , and the function \mathcal{G} is any log-concave or elliptically symmetric density (e.g. Gaussian), with mean vector μ_{jm} and covariance matrix U_{jm} . The mixture gain c_{jm} must satisfy the stochastic constraint:

$$\sum_{m=1}^K c_{jm} = 1 \tag{16}$$

D. HMM Face Recognition

1) *1D-HMM Face Recognition*: The configuration of a 1D-HMM model accepts vertical observation vectors (1D), which are obtained by sampling each face image using a sliding window. This approach was used in [9, 10, 12, 13, 16, 17, 18, 21, 23], where a sequence of observation vectors is constructed from each image. The size of the sequence depends on the size of the images, and the length of each vector varies according to the size of the scanning window. Fig. 2 shows an example of a two-state 1D-HMM topology.

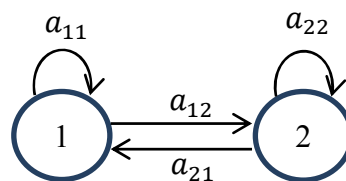


Fig. 2. State diagram of fully connected 2-state 1D-HMM

2) *2D-HMM Face Recognition*: The configuration of a 2D-HMM model is a composition of two 1D-HMM models, where each model is overlaid onto the other. It consists of a number of superstates, which represent the main 1D-HMM, and each superstate consists of a number of states of the 1D-HMM model. In [11, 15, 22], 2D-HMMs were employed with a variation in the number of states and also the number of superstates. Fig. 3 shows a structure of a left-to-right 2D-HMM.

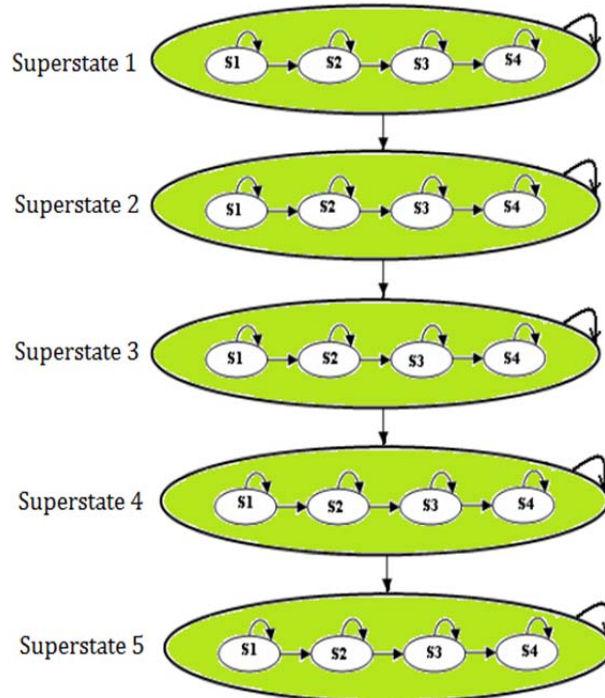


Fig. 3. State diagram of left-to-right 2D-HMM

IV. THE PROPOSED WORK

The proposed face recognition system consists of four steps: (1) pre-processing, (2) feature extraction, (3) training, and (4) testing. For each individual, a proper subset of images is selected from the database for training and the remaining images are used for testing. The accuracy of the system is usually affected by the training set selection; thus, the best recognition rate may be achieved by using different poses from the training images. The main operations of the system are shown in Fig. 4.

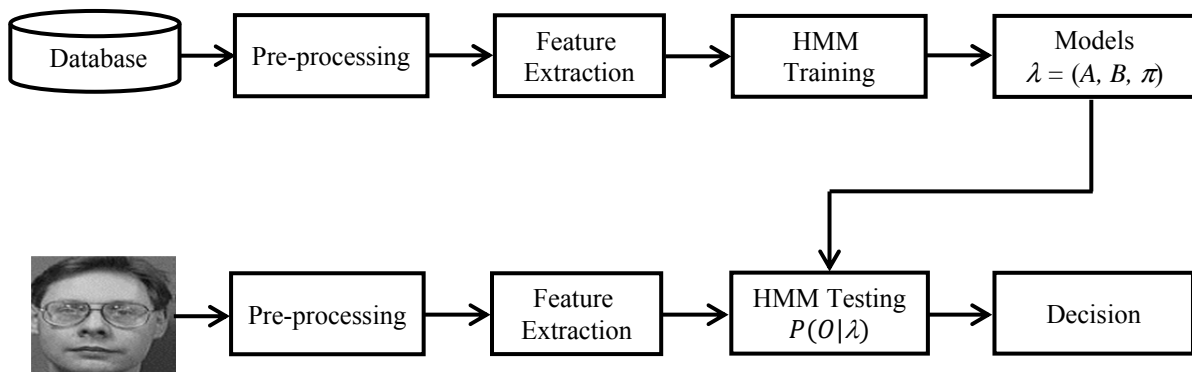


Fig. 4. Main operations of the proposed face recognition system

A. Pre-processing

A pre-processing step is used to enhance the operation of the system, which includes image cropping and median filtering. Image cropping is used only for large images to reduce the data to be processed while maintaining sufficient data to represent the entire face. The images in the ORL face database [27] remain uncropped because the size of these images is 112×92 , which may not be a large size, whereas the cropping is

performed on the images in the Yale database (243×320) [28] and the EURECOM Kinect database (256×256) [29]. Every image within the Yale and EURECOM Kinect databases contains a dispensable background that is not relevant to the face; the important image region is extracted and resized to 192×192 pixels. On the other hand, application of median filtering is very relevant as it contributes to the elimination of noise and any illumination in the images. This step greatly improves the performance of the system, especially in noisy environments.

B. Feature extraction

The filtered image is decomposed by applying either the third or fourth level of the DWT according to the size of the image. The third level is applied to images of size 112×92 (ORL database) and the resultant image size is 14×12 , whereas images of size 192×192 (Yale and EURECOM Kinect databases) are decomposed to the fourth level to produce images of size 12×12 . It should be noted that detailed coefficients are excluded and only the approximation coefficients are retained. For each image, a sequence of blocks is generated as shown in Fig. 5, using a sliding window of size 5×4 for the ORL database and 5×5 for the two other databases. The maximum overlap between two consecutive blocks is selected, with a height and width less than one pixel. Then, the blocks are converted to column vectors and sequentially concatenated. For a given person, the number of sequences is equal to the number of training images, arranged according to the accepted form by the HMM.

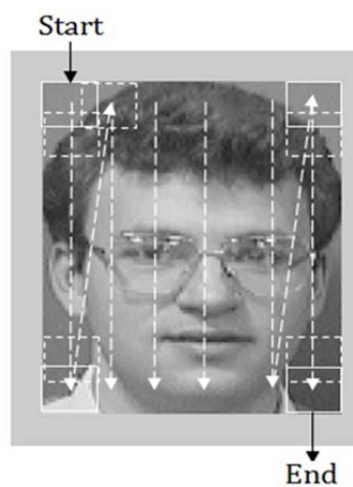


Fig. 5. Generation of blocks sequence

C. Training process

The proposed method is distinct from previous HMM face recognition methods, where it uses only two states of the ergodic HMM, as shown in Fig. 2. The continuous output observations are obtained using the Gaussian probability density function with one mixture component, such that $K = 1$ in (15) and (16). One model is trained for each individual using the Baum-Welch algorithm [30]. The model is initialized with random parameters according to the constraints cited in Sec. III-C. The training process converges when the possible difference between two successive iterations reaches any value below (10^{-5}); otherwise, it converges in four iterations.

D. Testing process

A sequence of observation vectors is obtained from the image to be tested after using a median filter and a DWT, as previously described. The Viterbi algorithm [31] is used to calculate the probability of this sequence of vectors using all HMM-trained models. The unknown image is identified as that person with the highest probability index.

V. EXPERIMENTAL RESULTS

The experiments are conducted on three sets of databases: ORL (Olivetti Research Laboratory), Yale, and EURECOM Kinect. The ORL database includes 400 images of size 112×92 (10 different images for each of 40 distinct persons). For some persons, the images were taken of the person while he/she exposed facial expressions (open / closed eyes, smiling / not smiling), facial details (glasses / no glasses), and tolerance for some side movement. The Yale face database contains 165 grayscale images to accommodate 15 persons, each of size 243×320 . For each individual, 11 images are taken with different facial expressions, e.g. happy, sad, sleepy, not wearing glasses, wearing glasses, surprised, and winking. The EURECOM Kinect dataset contains 468 images, each of size 256×256 , and represents the multimodal facial images of 52 people. For each person,

nine images were taken, each with a different facial expression, different lighting, and occlusion conditions: neutral, smiling, open mouthed, left profile, right profile, occlusion of eyes, occlusion of mouth, occlusion of paper, and lights turned on.

Several types of wavelet families are examined to determine the proper type that provides the best results. The wavelet families used are: Daubechies, Coiflet, Biorthogonal, Reverse biorthogonal, and Symlet. The test encompasses all subfamilies, and it appears that the Haar wavelet is the most suitable type. It is worth noting that the wavelet and scaling functions of the Haar wavelet described in (1) and (2), respectively, are exactly the same as those of db1, Biorthogonal 1.1 (bior1.1), Reverse biorthogonal 1.1 (rbio1.1) and Symlet1 (Sym1) wavelets. Therefore, these wavelets also produce a high recognition rate (100%).

A. Experiment 1

This experiment is performed to evaluate the sufficient number of training images that achieves the highest recognition rate. If the number of images for each person in the database is k and the number of training images is n , the number of testing images is $k - n$. The best recognition rates obtained from varying the number of training images taken from the ORL, Yale and RGB in session 2 of EURECOM Kinect databases are tabulated in Table I.

Table I. Experimental results of varying the number of training images

No. of training images	ORL Database		Yale Database		EURECOM Kinect Database	
	No. of testing images	Recognition Rate (%)	No. of testing images	Recognition Rate (%)	No. of testing images	Recognition Rate (%)
1	9	58	10	65.33	8	49.76
2	8	85.63	9	88.2	7	55.22
3	7	94.64	8	94.2	6	72.11
4	6	98.33	7	98.1	5	86.15
5	5	100	6	100	4	95.2
6	4	100	5	100	3	100

It is evident from Table I that the adequate number of training images for each person employed from the ORL and Yale databases that achieves a free of errors testing is five, whereas six training images for each individual in the EURECOM Kinect database are appropriate for achieving a recognition rate of 100%. This is explained by observing the nature of the images of each database. The ORL and Yale databases contain primarily frontal images with some facial and lighting effects, while the EURECOM Kinect database includes non-frontal images and partially occluded images. The recognition process is highly dependent on the nature of the training images, and the heterogeneous nature of the database poses may be such that a pose that is out of the (assumed) normal distribution may be selected to maintain a successful discriminant process.

B. Experiment 2

To examine the efficiency of the system in the case of addressing noisy images, impulse noise (also called salt and pepper noise) and white Gaussian noise are added to the testing images, whereas the training process is conducted in an environment that is more tightly controlled, the noise-free images are used. To address images with real noise, we were unable to compute the noise density that the image contained, so this attempt failed to determine the efficiency of the system, whereas the addition of noise to the images with a predefined noise density may contribute significantly to the determination of the effectiveness of the system to noise level. One particular test was conducted without using the median filter for the purpose of examination of its usefulness in noise reduction. Fig. 6 shows the results of using impulse noise and white Gaussian noise (zero mean and different variance) with and without median filtering.

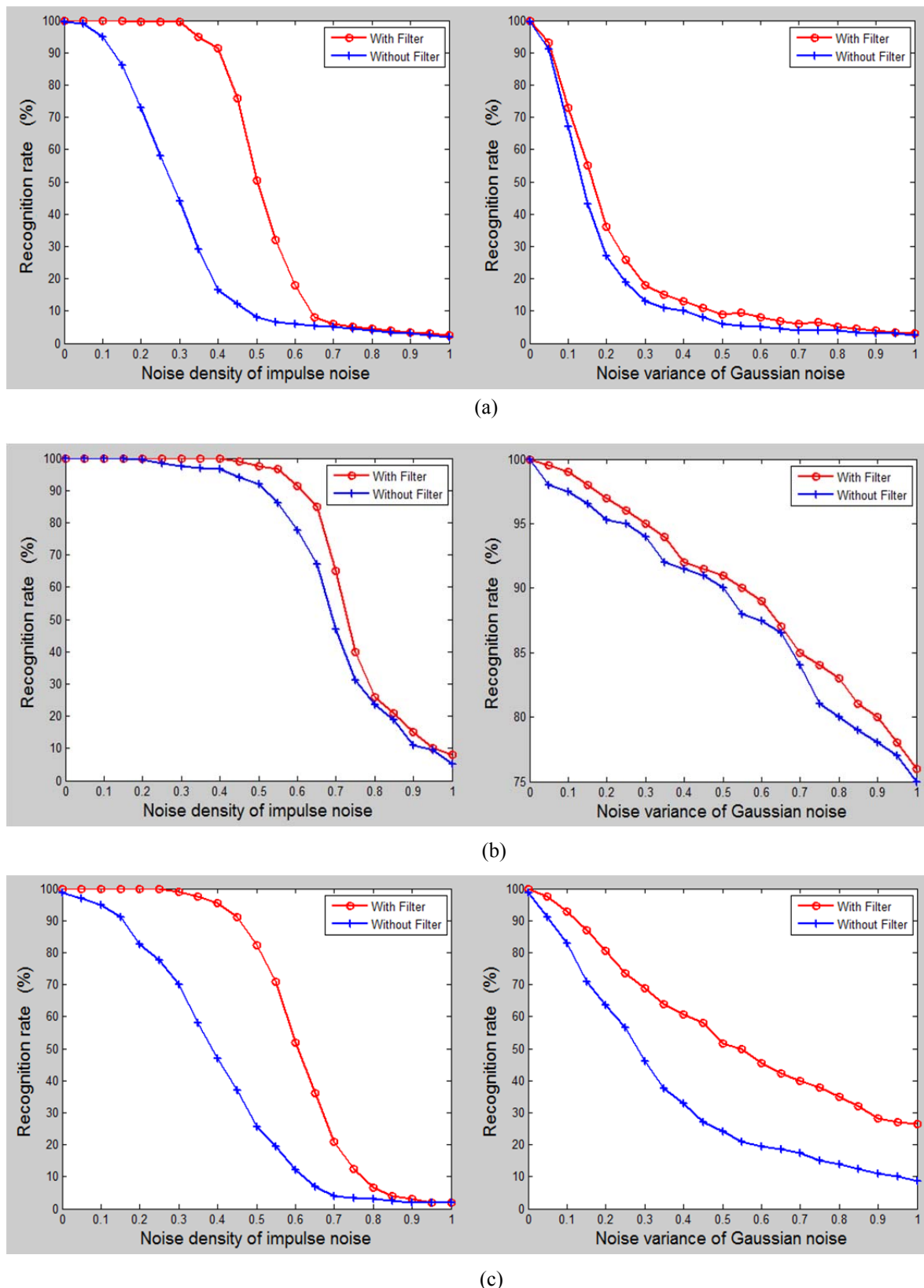


Fig. 6. Experimental results of noisy images with and without the use of median filter: (a) for ORL database, (b) for Yale database, and (c) for EURECOM Kinect database

According to the results illustrated in Fig. 6, it is clear that the median filter is more efficient in removing the effect of impulse noise from the images. The recognition rate remains at 100% while the noise density increased to 0.15 for images of the ORL database, 0.4 for images of the Yale database, and 0.25 for images extracted from the EURECOM Kinect database. The disparity among the rates between the databases appears as a result of the

differences in their sizes and images nature. It should be noted that the testing sets of the system include: 200 images from the ORL database (five images for each of 40 persons), 90 images from the Yale database (six images for each of 15 persons) and 156 images from the EURECOM Kinect database (three images for each of 52 persons). Impulse noise turns the pixel value on or off, i.e. the pixel value assumes a value of either 255 or zero, and, unless the noise density is high, these values do not represent the median of the mask window. To explain this observation, a mask window of size 3×3 was, via experiment, overlaid onto the image and the following pixel values were fetched from predefined positions:

$$W = \begin{bmatrix} 200 & 197 & 202 \\ 195 & 193 & 201 \\ 189 & 196 & 194 \end{bmatrix}$$

To obtain the median of W , it must be ordered by descending value to generate the vector V :

$$V = [202 \ 201 \ 200 \ 197 \ 196 \ 195 \ 194 \ 193 \ 189]$$

The index of the mid-value is 5, and the median value of W is 196. The window may be infected as a result of low noise density (number of infected pixels in W is less than five) or by high noise density (number of infected pixels in W is greater than or equal to five). In the case of low noise density, the median value either remains unchanged or it is rounded to the nearest value depending on its position in the arranged vector, and also on its state of infection. The following descending vectors show some possibilities that illustrate infection, such that the noise and median values are indicated by blue and red numbers, respectively:

$$V = [255 \ 255 \ 200 \ 197 \ 196 \ 193 \ 189 \ 0 \ 0]$$

$$V = [255 \ 255 \ 202 \ 200 \ 197 \ 196 \ 194 \ 189 \ 0]$$

$$V = [255 \ 202 \ 201 \ 197 \ 195 \ 194 \ 193 \ 0 \ 0]$$

$$V = [255 \ 201 \ 200 \ 195 \ 194 \ 193 \ 0 \ 0 \ 0]$$

In case of high noise density, it is probable that the median value is changed either to 0 or 255, as illustrated in the following vectors:

$$V = [255 \ 255 \ 255 \ 255 \ 0 \ 0 \ 0 \ 0 \ 0]$$

$$V = [255 \ 255 \ 255 \ 255 \ 255 \ 196 \ 0 \ 0 \ 0]$$

$$V = [255 \ 255 \ 202 \ 197 \ 0 \ 0 \ 0 \ 0 \ 0]$$

In some cases, despite the presence of high noise density, the median value either remains unchanged or it is altered to the nearest value as shown in the following vectors:

$$V = [255 \ 255 \ 255 \ 201 \ 197 \ 0 \ 0 \ 0 \ 0]$$

$$V = [255 \ 255 \ 255 \ 255 \ 196 \ 0 \ 0 \ 0 \ 0]$$

$$V = [255 \ 255 \ 255 \ 197 \ 195 \ 193 \ 0 \ 0 \ 0]$$

It can be concluded from the above example that the image may be infected by a certain degree of noise density, the median value of which is non-deterministic. Therefore, the experiment is repeated four times and the average rate is considered. Hence, it is clear that the median filter maintains the characteristics of the image despite the presence of some degree of noise density. This leads to the conclusion that the use of a median filter greatly improves the performance of the system, especially in the case of impulse noise.

White Gaussian noise differs somewhat from impulse noise as it modifies the pixels to values ranging between 0 and 255, such that the median value is probably changed. Therefore, the usefulness of a median filter in the case of white Gaussian noise is limited.

C. Comparative results

The advantage of the proposed work relative to other research can be observed through a comparison performed under the condition of the similarity between these methods, including the use of the HMM and the face database.

In the case of noisy images, the research presented in [18] used the third level of the WT in conjunction with the HMM, and several tests were performed on the ORL database. The authors used white Gaussian noise with a mean value of zero and a maximum variance value of 0.1 to examine the efficiency of their system. Fig. 7 shows a comparison between the results reported in [18] and the results of the proposed work with and without the use of a median filter.

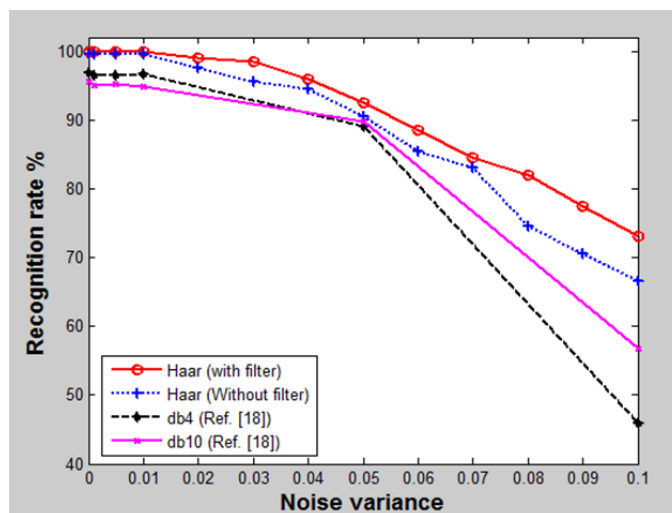


Fig. 7. Comparative results of noisy face recognition systems

From Fig. 7, it can be observed that the proposed work has achieved the best results, either using a median filter or without filtering. This leads to the conclusion that Haar is the best wavelet type for noise elimination. Noise may be regarded as a high frequency component, and can be filtered out using a low pass filter. As mentioned previously, the DWT splits the image into approximation (low frequency) coefficients and detailed (high frequency) coefficients. The approximation coefficients form the base for the feature extraction step, while the detailed coefficients are ignored. Therefore, the DWT can be considered as a good noise eliminator.

It is more suitable to discriminate the best HMM face recognition by a comparison with other work. Table II shows comparative results relative to the ORL database.

Table II. Comparative results of HMM face recognition methods on the ORL database

Ref. No.	Feature Extraction Method	No. of HMM States	Training Time per image (s)	Testing Time per image (s)	Recognition Rate (%)
[10]	WT	NA	NA	NA	100
[12]	SVD	7	0.63	0.28	99
[13]	SVD	3	0.024	0.052	100
[14]	*MV+SVD	2	0.02	0.042	100
[15]	DCT	5	5.2	0.25	91
[16]	DWT	5	1.13	0.3	100
[17]	DWT	5	1.3	0.8	100
[18]	WT	5	NA	NA	97.1
[19]	DWT	NA	NA	NA	98.5
[20]	DWT	NA	4.31	3.45	97
[21]	WT+PCA	5	NA	1.2	96.5
[22]	DGWT	7	NA	NA	99
[24]	GF+LDA	15	NA	NA	97.5
This Paper	DWT	2	0.02	0.05	100

* MV=Maximum variance

At first glance, one can observe from Table II that the processing time of [14] is much closer to that of this work; however, there are three major differences between our results and those presented in [14], as explained below:

1. In the case of updating the database, the entire system is required to be retrained in [14], whereas the proposed work trains only the model that requires an update while all other models remain unchanged.
2. In [14], the quantization and the weight vectors are chosen experimentally using a trial and error method, and this process consumes a large amount of time, contrary to the proposed work, which uses a different feature extraction method that requires no additional time.

- In case of noisy images, the proposed work outperforms the work introduced in [14] due to the use of a DWT, which is an efficient noise eliminator. Fig. 8 shows comparative results of using the two types of noise.

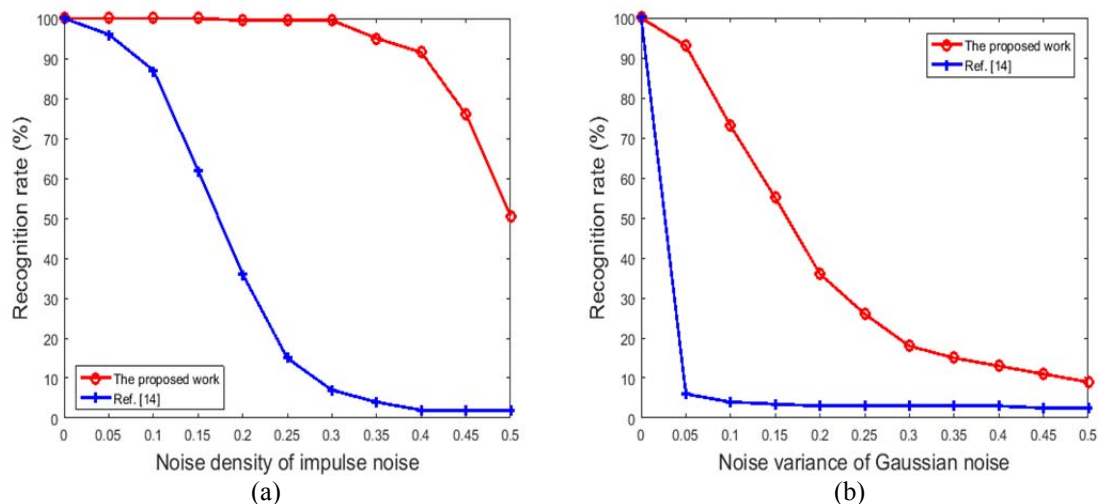


Fig. 8. Comparative results between the proposed work and the method employed in [14], where noisy images are tested: (a) for Impulse noise, and (b) for white Gaussian noise with zero mean

The high margin between the two curves in both cases illustrated in Fig. 8 is evident, and this demonstrates the fact that there is no doubt that the proposed work can be considered the best face recognition system in noisy environments.

Some of the HMM approaches employ the Yale face database; therefore, a comparison is made in Table III to show the effectiveness of the proposed algorithm applied to the Yale database.

Table III. Comparative results of HMM face recognition methods on Yale database

Ref. No.	Feature Extraction Method	No. of HMM States	No. of Training images	Training Time per image (s)	Testing Time per image (s)	Recognition Rate (%)
[12]	SVD	7	6	0.53	0.15	100
[24]	GF+LDA	9	6	NA	NA	100
This Paper	DWT	2	5	0.03	0.03	100

As can be observed in Table III, the proposed work achieves 100% recognition rate with five training images, whereas the other methods reach this rate with six training images. Additionally, the proposed work is more efficient, consuming fewer processor cycles and less memory due to the use of only two states of the HMM, compared to the other methods that contain seven or nine states.

Based on our knowledge, the EURECOM Kinect database has not yet been used in the HMM face recognition approach, as it has only recently been recognized as competitive within the face recognition field. In [29], the authors presented the EURECOM Kinect database and employed four types of face recognition methods: PCA, LBP, scale-invariant feature transform (SIFT), and local Gabor binary patterns (LGBP). Table IV shows a comparison with the results reported in [29] related to 2D RGB images of session 2.

Table IV. Comparative results on session 2 of EURECOM Kinect database

Ref. No.	Method	Recognition Rate (%)
[29]	PCA	53.57
	LBP	90.93
	SIFT	74.18
	LGBP	95.05
This Paper	DWT+HMM	100

In [29], only the recognition rates are listed and the processing times are unrecorded. Therefore, based on the available information, the proposed algorithm outperforms the methods used in [29], where it achieves the maximum recognition rate (100%), as is clear in Table IV. The training time of one image from the EURECOM Kinect database is 0.017 (s) including the median filtering and the DWT, and the elapsed time required to test a single image is 0.07 (s).

The experiments were performed using MATLAB (R2016a) included the Bayes Net Toolbox for the HMM [32]. The operating system was Windows 7 (64-bit) executed on a machine with a double-core processor of 2.53 GHz, 2528 Mhz speeds and 4 GB RAM.

VI. CONCLUSION

A noisy face recognition method is presented using the median filter and a multilevel DWT in combination with the HMM. The novelty of the algorithm can be viewed in the use of two states of the ergodic HMM, which is a unique model in the face recognition field and has not yet been used by other researchers. Moreover, the use of the median filter together with a multilevel DWT achieves 100% recognition rate in spite of the presence of 0.15, 0.4, and 0.25 noise densities on the images of the ORL, Yale, and EURECOM Kinect databases, respectively. In addition to noise elimination, a high reduction in image size is achieved using the third and fourth levels of the DWT. The processing time and the memory utilization are extremely minimized due to the reduction in image size and the number of states, which is interpreted as simplicity and low complexity of the system. Comparative results show the superiority of the proposed technique over others.

In the future, a discrete HMM will be used instead of the continuous model; we also plan to use only one state of HMM.

REFERENCES

- [1] M. Kasar, D. Bhattacharyya, and T. Kim, "Face Recognition Using Neural Network: A Review", *International Journal of Security and Its Applications*, vol. 10, no. 3, pp. 81-100, 2016.
- [2] R. Kaur and E. Himanshi, "Face recognition using Principal Component Analysis", 2015 IEEE International Advance Computing Conference (IACC), Bangalore, 12-13 June 2015, pp. 585-589, IEEE, 2015.
- [3] A. Singha, M. Bhowmik, P. Dhar, and A. Ghosh, "Analysis and Performance Evaluation of ICA-based Architectures for Face Recognition", in *Proc. of the 17th International Workshop on Combinatorial Image Analysis (IWCIA 2015)*, ISI, Kolkata, Springer-Verlag, LNCS, vol. 9884, pp. 323-336, 2015.
- [4] A. Sharma and K. K. Paliwal, "A two-stage linear discriminant analysis for face-recognition", *Pattern Recognition Letters*, vol. 33, no. 2012, pp. 1157-1162, Elsevier, 2012.
- [5] Z. Li and X. Tang, "Using Support Vector Machines to Enhance the Performance of Bayesian Face Recognition", in *IEEE Transactions on Information Forensics and Security*, vol. 2, no. 2, pp. 174-180, June 2007.
- [6] A. Suruliandi, K. Meena, and R. Rose, "Local binary pattern and its derivatives for face recognition", *IET Computer Vision*, vol. 6, no. 5, pp. 480-488, IET, 2012.
- [7] L. Rabiner, "A tutorial on hidden Markov models and selected application in speech recognition", in *Proc. IEEE*, vol. 77, no. 2, pp. 257-286, 1989.
- [8] L. Xia, H. Fang, and X. Liu, "Hidden Markov model parameters estimation with independent multiple observations and inequality constraints", *IET Signal Processing*, vol. 8, no. 9, pp. 938-949, IET, 2014.
- [9] F. Samaria, and F. Fallside, "Face identification and feature extraction using hidden Markov models", in Vernazza, G. (Ed.): "Image processing: theory and applications", Elsevier Science press, 1st edition, pp. 295-298, June 1993.
- [10] M. Bicego, U. Castellani, and V. Murino, "Using hidden Markov models and wavelets for face recognition", in *Proc. 12th Int. Conf. Image Analysis and Processing*, Mantova, Italy, Sept. 2003, pp. 52-56, IEEE, 2003.
- [11] J. Cao, and C. Tong, "Facial expression recognition based on LBP-EHMM", in *Congress on Image and Signal Processing*, Sanya, China, May 2008, vol. 2, pp. 371-375, IEEE, 2008.
- [12] H. Naimi and P. Davari, "A new fast and efficient HMM-based face recognition system using a 7-State HMM along with SVD coefficients", *Iranian Journal of Electrical & Electronic Engineering*, vol. 4, nos. (1&2), pp. 46-57, 2008.
- [13] H. Farhan, M. Al-Muifraje, and T. Saeed, "Face recognition using maximum variance and SVD of order statistics with only three states of hidden Markov model", *International Journal of Computer Applications*, vol. 134, no. 6, pp. 32-39, 2016.
- [14] H. Farhan, M. Al-Muifraje and T. Saeed, "Using only two states of discrete HMM for high-speed face recognition," in 2016 Al-Sadeq International Conference on Multidisciplinary in IT and Communication Science and Applications (AIC-MITCSA), Baghdad, Iraq, 9-10 May 2016, pp. 1-5, IEEE, 2016.
- [15] B. Vaseghi, and S. Hashemi, "Face verification using D-HMM and adaptive K-means clustering", in *Proc. 4th IEEE Int. Conf. Broadband Network and Multimedia Technology*, Oct. 2011, pp. 270-275, IEEE, 2012.
- [16] H. Le, and H. Li, "Simple 1D discrete hidden Markov models for face recognition", in *Proc. 8th Int. workshop*, Madrid, Spain, 18-19 Sep. 2003, in Garcia N., (Ed.): "Visual content processing and representation", *Lecture Notes in Computer Science*, vol. 2849, Springer-Verlag, Berlin, Heidelberg, pp. 41-49, 2003.
- [17] E. Abbas, and H. Farhan, "Face recognition using DWT with HMM", *Engineering & Technology Journal*, vol. 30, no. 1, pp. 142-154, 2012.
- [18] I. Makaremi, and M. Ahmadi, "A robust wavelet based feature extraction method for face recognition", in *Proc. IEEE Int. Conf. Systems, Man, and Cybernetics*, 11-14 Oct. 2009, San Antonio, USA, pp. 2173-2176, IEEE, 2009.
- [19] L. Shen, Z. Ji, and L. Bai, "DWT based HMM for face recognition", *Journal of Electronics*, vol. 24, no. 6, pp. 835-837, 2007.
- [20] P. Nicholl, A. Amira, D. Bouchaffra, and R. H. Perrott, "Multiresolution hybrid approaches for automated face recognition", in *Proc. 2nd NASA/ESA Conf. Adaptive Hardware and Systems*, Edinburgh, Scotland, 5-8 Aug. 2007, pp. 89-96, IEEE, 2007.
- [21] L. Peng, and L. Jiao, "Implement of face recognition system based on hidden Markov model", in *Proc. 6th Int. Conf. Natural Computation*, 10-12 Aug. 2010, Yantai, China, vol. 7, pp. 3344-3348, IEEE, 2010.
- [22] M. Srinivasan, and N. Ravichandran, "A new technique for face recognition using 2D-Gabor wavelet transform with 2D-hidden Markov model approach", in *Proc. Int. Conf. Signal Processing, Image Processing & Pattern Recognition*, Feb. 2013, Coimbatore, India, pp. 151-156, IEEE, 2013.

- [23] M. Srinivasan, and N. Ravichandran, "A 2D discrete wavelet transform based 7-state hidden Markov model for efficient face recognition", in Proc. 4th Int. Conf. Intelligent Systems Modeling & Simulation, Bangkok, Thailand, Jan. 2013, pp. 199–203, IEEE, 2013.
- [24] Z. Elgarrai, O. Meslouhi, H. Allali, et al., "Face recognition system using Gabor features and HTK toolkit", in Proc. 10th Int. Conf. Signal-Image Technology and Internet-Based Systems, Marrakech, Morocco, 23–27 Nov. 2014, pp. 32–36, IEEE, 2015.
- [25] R. Gonzalez, and R. Woods, "Digital image processing", Pearson Prentice Hall press, 3rd edition, 2008.
- [26] D. Ambika, and V. Radha, "A comparative study between discrete wavelet transform and linear predictive coding", World Congress on Information and Communication Technologies, 30 Oct.-2 Nov. 2012, Trivandrum, India, pp. 965–969, IEEE, 2013.
- [27] AT&T Laboratories, "The ORL face database", Cambridge, U.K. Available:
<http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html>
- [28] Computer Vision Laboratory, "The Yale database", Computer Science and Engineering Department, University of California, San Diego. Available: <http://vision.ucsd.edu/content/yale-face-database>
- [29] R. Min, N. Kose, and J. Dugelay, "KinectFaceDB: A Kinect database for face recognition", IEEE Transactions on Systems, Man, and Cybernetics, vol. 44, no. 11, pp.1534–1548, IEEE, 2014.
- [30] L. Welch, "Hidden Markov models and the Baum-Welch algorithm", IEEE Information Theory Society Newsletter, vol. 53, no. 4, pp. 1–24, IEEE, December 2003.
- [31] A. Viterbi, "A personal history of the Viterbi algorithm", IEEE Signal Processing Magazine, vol. 23, no. 4, pp. 120–142, IEEE, July 2006.
- [32] K. Murphy, "BNT-Hidden Markov Model Toolbox for Matlab". Available:
<http://www.cs.ubc.ca/~murphyk/Software/HMM/hmm.html>