We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900
Open access books available

186,000
International authors and editors

200M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS
BOOK CITATION INDEX
INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Robust Face Detection through Eyes Localization using Dynamic Time Warping Algorithm

Somaya Adwan

Factuly of engineering, department of electrical engineering, University of Malaya
Kuala Lumpur
Malaysia

1. Introduction

Detection of faces and facial patterns in static or video images is an important but challenging problem in computer vision, as faces may present in different scales, orientations, positions and poses in an uncontrolled background. To date, numerous approaches have been implemented for face and facial pattern detection [1-4]. These approaches have been grouped into four broad categories [2]. These are knowledge-based, feature based, appearance based and template based methods.

In this chapter, we present a simple yet robust algorithm for template matching method for face detection through eyes localization by using Dynamic Time Warping (DTW) algorithm with different poses and variations. By using DTW, we overcome some of the main limitations of Classical template matching.

We elaborate through discussion and experiments' results that how our image processing strategy applied to DTW algorithm effectively increase the performance and detection rate. Using such an approach does not require translation of human knowledge about the face and facial patterns to computer representation, training data set and training mechanisms or face geometry, and does not require large numbers of model templates to handle different pose and variation changes shown by facial features. In the coming section, short study of Dynamic Time Warping algorithm (DTW) is presented. Proceeding to the next section we discuss the necessary steps for our image processing strategy. This is followed by presenting our approach of image partitioning, and in the next section we present dynamic time warping algorithm application in our work. Towards the end of this paper, results and discussion are put forward and conclusion is presented.

2. Visiting Dynamic Time Warping review

Dynamic Time Warping (DTW) is a fast and efficient algorithm for measuring similarity between two sequences. It is used to find the optimal alignment between two time series, if one time series may be warped non-linearly along its time axis. This warping between the two time series can be used to determine the similarity which may vary in time or speed. Similarity is measured by aligning two sequences and computing a distance between them

[5]. This technique has made it possible to make a matching between two curves that are "subject not only to alteration by the usual additive random error but also to variations in speed from one portion to another" [5]. Dynamic Time Warping (DTW) was initially introduced to recognize spoken words [7]. Since then it has been used and proved useful in different applications such as: handwriting recognition [6, 8], signature verification [9-12], fingers print verification [13], face recognition [14, 15], speech and gesture recognition [16], protein structure [17], gene expression [18, 19], chromosome classification [20], object detection and classification [21, 22], curve matching [23, 24] control system [25], image and shape matching [24,26] database and data mining [27- 29, 31, 32, 34-37), brain activity classification [38], voice control [39] and so on.

The DTW algorithm uses a dynamic programming technique to align time series with a given template so that a total distance measure is minimized [25]. To align these sequences (the reference/template vector X of length M and the test vector Y of length N), a local cost matrix is defined where each cell (i,j) represents the pair wise distances between the ith component of X and the corresponding component j^{th} of vector Y. Generally the Euclidean distance is suggested as a distance cost function in DTW algorithm to build a local cost matrix [30], the cost (distance) function has a small value when sequences are similar and large value if they are different. After computing the local cost matrix between two 1-dimenasional features' vectors X and Y, a warping path is constructed by building a cumulative/global cost matrix D(i,j) [6,7]. The cumulative distances matrix D(i,j) is defined as the sum of the local distance d(i,j) found in the current cell and the minimum of the cumulative distances of the adjacent cells. A warping path is a concatenation of cells starting from (1,1) to the cell (M,N). The objective is to search an optimal path for which a least cost is associated. This path provides the low cost areas between the two feature-vectors (see section 3.5 for more details of building the local and global matrices).

In DTW, contsriants are applied for constructing optimal warping path [7, 27]. Constraints are widely used to speed up DTW [7, 27, 30, 33] by reducing the number of paths to be considered during the computation. "These constraints serve to reduce the search space- the space of possible warping paths. Searching through all possible warping paths is computationally expensive. Therefore, out of concern for efficiency, "it is important to restrict the space of possible warping paths" [27]. Several well-known constraints have been applied to the problem to restrict the moves that can be made from any point in the path. Sakoe and Chiba applied a pattern matching algorithm with a nonlinear time-normalization effect using dynamic programming to spoken word recognition [7]. They introduced several kinds of reasonable constraints as outlined below [27]:

- Monotonicity: The alignment path doesn't go back in time index, $i_{k-1} \leq i_k$ and $j_{k-1} \leq j_k$. This guarantees that features are not repeated in the alignment.
- Continuity: The alignment doesn't jump in time index, $i_k - i_{k-1} \leq 1$ and $j_k - j_{k-1} \leq 1$. This guarantees that important features are not omitted.
- Boundary: The alignment starts at the top-left and ends at the bottom-right. This guarantees that the sequences are not considered only partially.
- Warping window: A good alignment path is unlikely to wander too far from the diagonal. This guarantees that the alignment doesn't try to skip different features or get stuck at similar features, and allowable points must fall within a given window, $|i_k - i_{k-1}| \leq w$; where w is a positive integer window width (threshold).

A new restriction called a slope constraint was introduced in their work and it was claimed that symmetric form and slope constraint are effective, which resulted in optimum performance in comparison to several dynamic programming algorithms [7]. Slope constraint was also used later by Wang et al. [23], to prevent unrealistic warping.

Berndt and Clifford [27] applied the constraints introduced in Sakoe and Chiba work [7] using symmetric form to find patterns in time series for archiving purpose. They used tracing backward technique to find the optimal warping path. Daniel et al. [33] proposed the threshold DTW (TDTW) algorithm based on the DTW technique, for two dimensional spatial activity recognition, by introducing a user defined threshold to the diagonal matching condition of the DTW. Their work showed that the TDTW is less affected to noise and they accomplished higher classification accuracy than DTW. David Clifford et al. [36] used DTW for aligning chromatogram signals. A variable penalty was introduced into the DTW that was added to the distance metric to reduce the number of non-diagonal paths. The percentage of non-diagonal moves taken during the usual DTW ranges from 52% to 70%. After adding the variable penalty to DTW, This range was reduced and it ranges from 1% to 5%. It was shown in the experiments that penalized DTW is highly significant and achieves good alignment of peaks in chromatograms.

These constraints are best visualized in [27, 23, 40]. Using constraints helps speed up DTW execution time by a constant factor but the DTW algorithm time complexity remains $O(N^2V)$ [40].

Some of the principal advantages of using Dynamic Time Warping include its capability to handle different scale and translation, its ability to compare two sequences of signals with different elongations. It does not require complex mathematical models.And "it is more robust against noise and provides scaling along the time axis. This ability allows DTW to identify similarities far more accurately and so enhances the functionality of the applications that use it" [31]. Furthermore, a simple pattern representation template is enough for similarity matching and detecting purposes with DTW [28]. However, DTW algorithm poses limitations in terms of time and space complexity [28, 40]. The time and space complexity of DTW is approximately $O\ (NMV)$, where N and M are sequences lengths ($O\ (N^2V)$ if the two sequences is equal in length), and V is the number of templates to be considered. This means that practical applications may be quite slow. To overcome these limitations, researchers proposed different methods to make DTW faster and to reduce the complexity of time and calculations required for alignment process.

Another significant issue with DTW is to achieve the higher accuracy for classification problems. This issue is dealt on application by application or domain by domain basis. Research efforts are being put forward to address this issue [48,49].

2.1 Dynamic Time Warping and face detection

Dynamic time warping gained significant attention in the last decade as a classification algorithm and similarity measure technique for different applications in different domains. Researchers have also attempted to use DTW with different existing algorithm within the domain of face localization and face recognition.

Luis et *al.* [43] introduced a face localization algorithm using DTW supplemented with skin color filter in RGB space to reduce the computational complexity. The template used in DTW is an average face created by manually extracting and aligning 40 different human faces with a constant size of 80 × 83 from 66 still images. The averaged face is quantized and

smoothed using median filter prior to extracting the feature vectors. The feature vectors, in their work, consist of vertical projection of the template and two horizontal vectors one for the eyebrows and eye region and the later for the nose and mouth region. Prior to DTW classification algorithm, they had segmented the image using a morphological skin color filter in RGB space. Then they had extracted the large regions, which contain the skin color as a candidate face region and restricted the DTW classification algorithm to the extracted regions that had reduced the total computational complexity of their approach. Finally the DTW is used to calculate the Pearson correlation coefficient as a similarity distance measure for every extracted feature vector in stepwise. Alexander et *al.* [15] proposed DTW and LSTM ANN algorithms for face recognition. For DTW, they used the Euclidian distance as similarity measure. The templates consists of 50 face images of the size of 19 × 19 pixels converted into 1-D 361 element feature vector. The test set consists of 50 face images for the same persons in the templates and the DTW classification algorithm is applied to calculate the similarity measure for every test image with all the templates. To validate their approach, they repeated the experiments in three scenarios:

1. The templates and the test images are free of noise.
2. The templates are free of noise and the test images are degraded with Gaussian noise (σ^2=0.1).
3. The templates and the test images are degraded with Gaussian noise (σ^2=0.1).

The first scenario showed a 100% detection rate and the path sizes matrix showed a perfect alignment of the templates and the test faces. The performance of the algorithm was reduced to 94% when applied to the second scenario; however, it shows the robustness of the algorithm in the presence of noise. The last scenario the performance is further reduced to 70%. All the scenarios are repeated using MSE as similarity measure and the performances were reported as 92%, 72%, and 34% respectively. For LSTM neural network, they used 50 faces converted to 1×361 feature vector as training set to represent the main classes. Eigen faces were extracted using PCA to reduce the dimensionality of the input space. The experiment is performed in two scenarios:

1. Using 10 PCA, and
2. Using 20 PCA.

Selecting the learning rate to be 0.02 and training the LSTM ANN for 50000 epochs in both scenarios to reach 0.05 mean square error. In the first scenario the detection rate was 96% and 88% for the second one. Computational complexity of their approach was not investigated. The proposed LSTM ANN exhibited high performance and robustness.

3. Method for face detection

The eyes region has the most edge information in a facial image [42], and hence, it can be considered as a vital signature of a human face. Thus, the eyes region was chosen in this study as a facial pattern to apply the proposed method.

To have an eye region as a facial pattern template, a face image of a person from a data set (reference) has been cropped to eyes. Significant information (features) is then extracted from this particular template and transformed into a 1-Dimendioanl (1-D) series sequence. In addition, this facial pattern is used as a reference to detect eyes in the input images from the data set. On the other hand, an input face image from the image data set is gone through a partitioning process (see sub-section 3.2.3). This process produces virtual partitions of an input image. Each partition is equivalent to the template image in size. For

each partitioned region, significant information is extracted and converted into a 1-D vector sequence to be aligned with the reference sequence (from template image) by DTW. The partition which belongs to a sequence that best matches the reference sequence is selected as the eyes/face region of the facial image. The proposed image processing strategy consists of the following steps:

i.   Edge Detection,
ii.   Histogram Equalization,
iii. Image Partitioning, and
iv.   Features-Vectors Extraction.

Figures 1 and 2 present the functional flow of the image processing strategy and the image processing strategy respectively. All the steps stated above are described in the following sections.
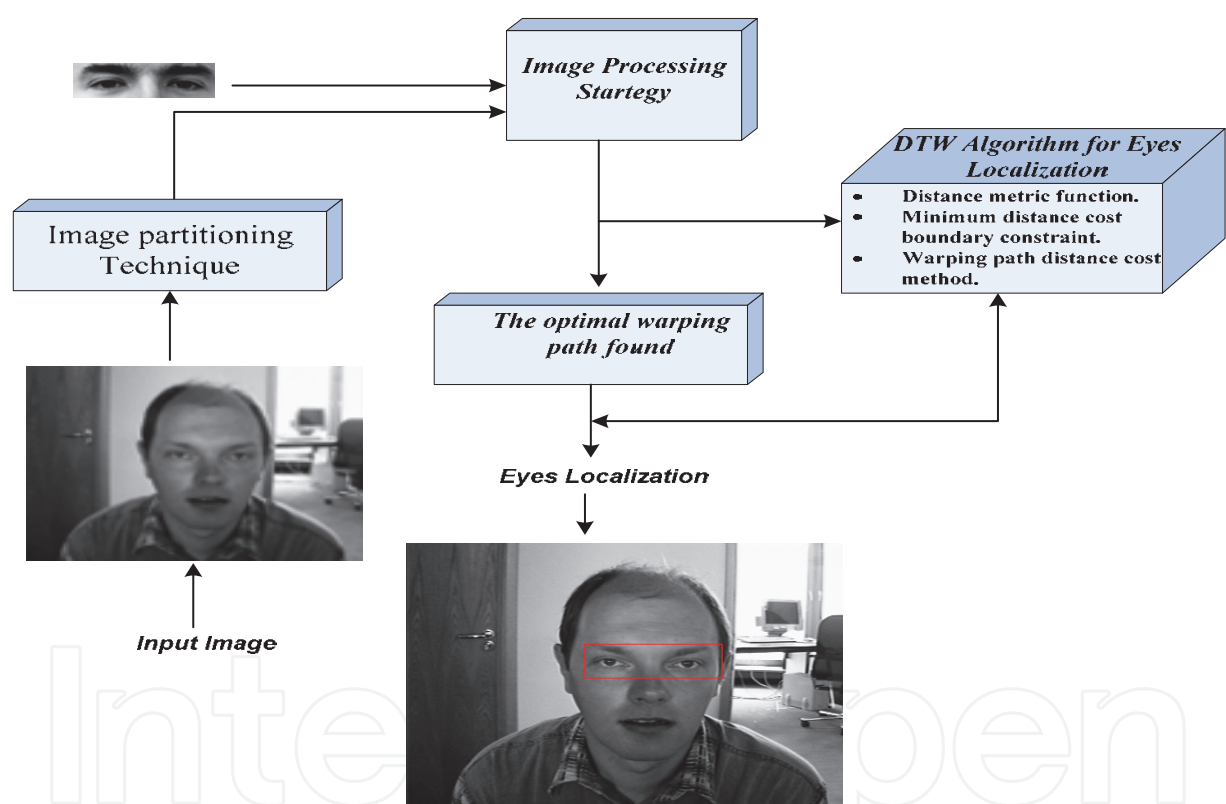
Fig. 1. Functional flow of Eye region detection process

3.1 Edge detection and cropping rules

The first step of proposed strategy is edge detection. An input image is processed for edge detection to find the region boundaries. Standard Sobel operators are used with 3X3 filters for edge detection. Once the boundaries of the image are determined, a cropping step is applied.

Cropping an image is important in reducing the size of the data and iterations required to scan full image by DTW. For this purpose, a cropping rule is set instead of the manual cropping. An image is cropped till the first boundary region determined in the previous step is obtained. Figure 3 shows the original input image and the edge image for the same image, as well as the result of the cropping rules.
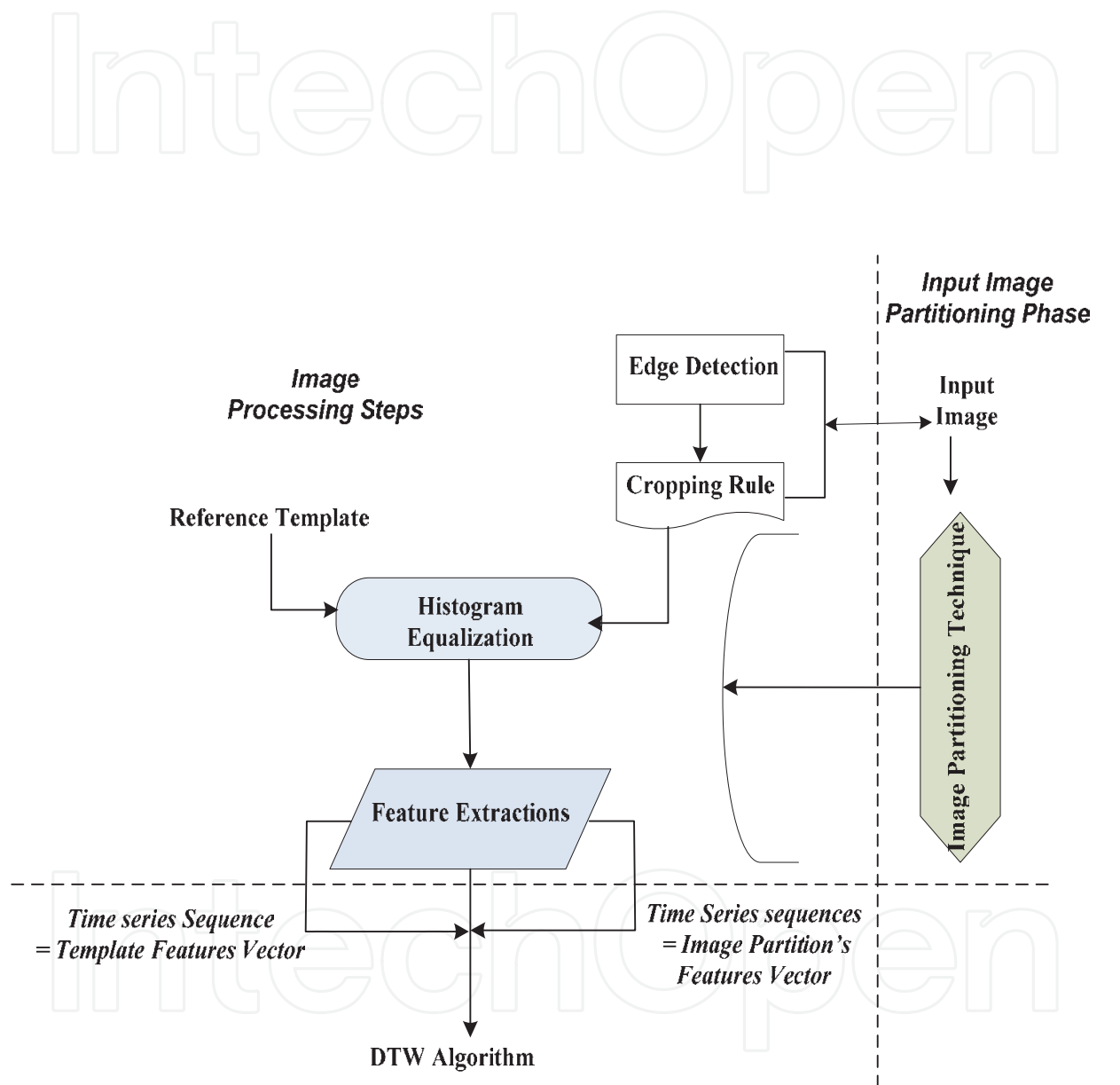
Fig. 2. Image processing strategy

(a)

(b)

(c)
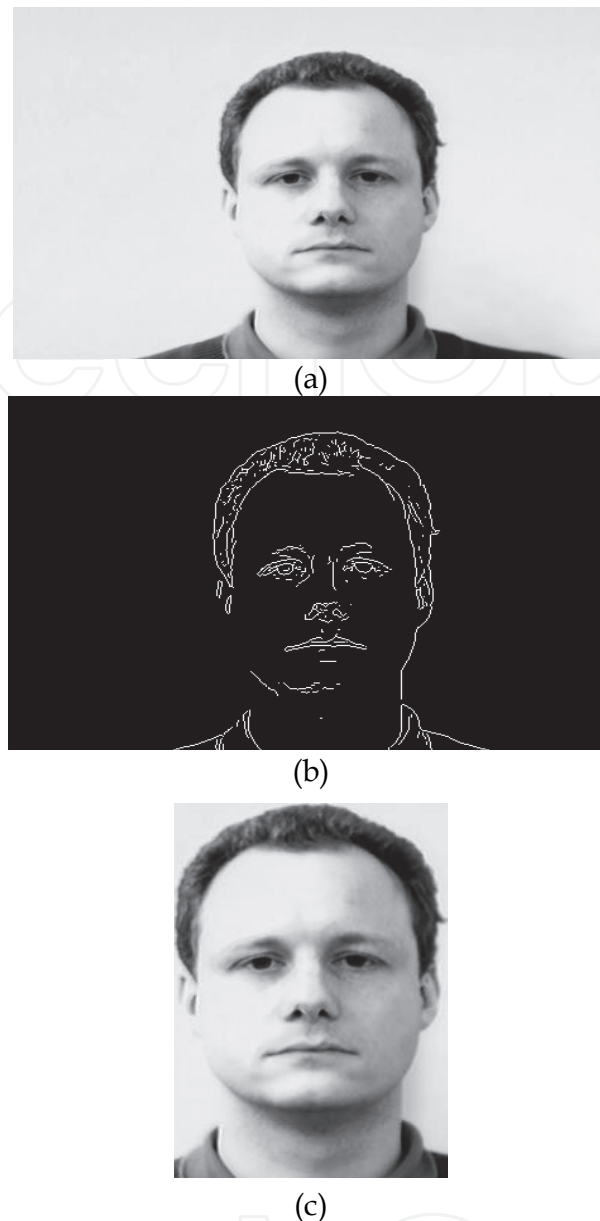
Fig. 3. (a) The original image, (b) the image after the application of Sobel operator, (c) the cropped image.

3.2 Histogram equalization

Intensities in the images are highly sensitive to external factors, such as illuminations, pose, rotations of the images, etc. These external factors affect the distribution of intensities in the histogram of the images [44], which in turn affects the detection rate to a great extent. In order to make the intensities relatively insensitive to a particular contrast, and brightness of the original image, etc., a histogram equalization is applied with a flat envelop on the image to re-distribute the intensities throughout the range. Figure 4 shows the histogram of an image before applying the histogram equalization step. In comparison to the histogram shown in Figure 4, the histogram of the same image is shown in Figure 5 after applying the histogram equalization. It can be seen in Figure 5 that the intensities of the image have been distributed throughout the range of the histogram.
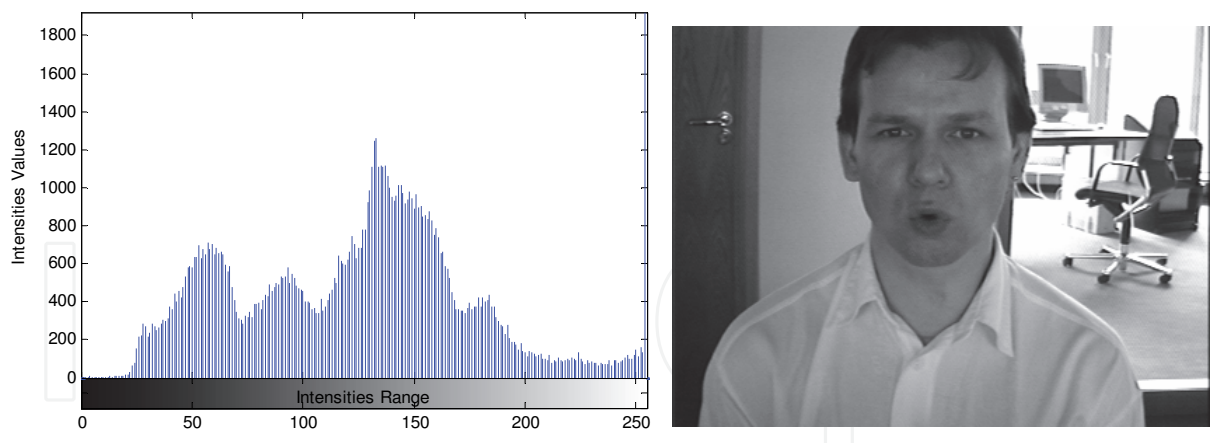
Fig. 4. A histogram distribution of an image before histogram equalization
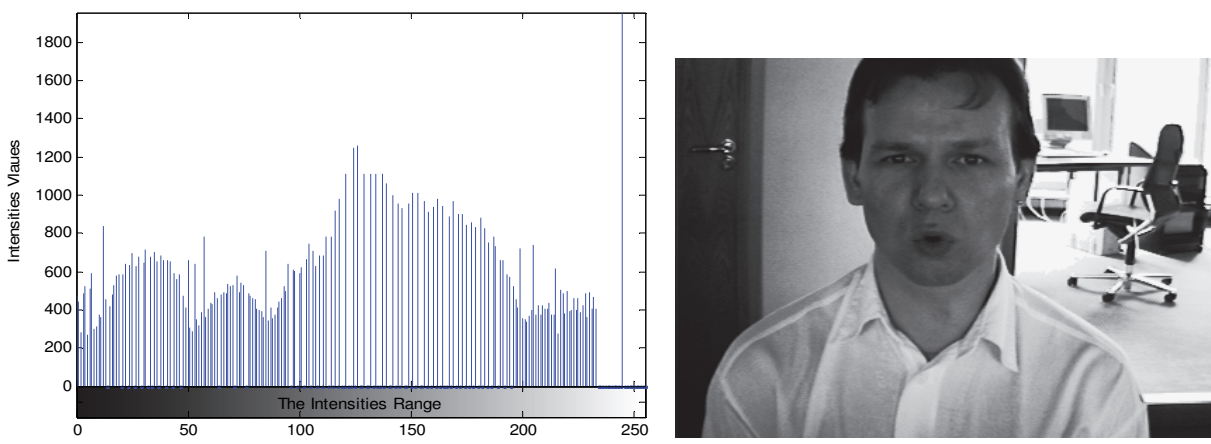
Fig. 5. A histogram distribution of an image after histogram equalization

3.3 Image partitioning

Image partitioning is an important step of the proposed strategy for face detection using DTW. In this step, virtual image partitions are extracted from an image. It is applied only on the test (input) images, which have been cropped through the steps of the image processing strategy. The image partitioning performed by our heuristic is given in Figure 6 below.

All partitions are equal in the size of the template image. Partitioning of an image is performed from the beginning of the first row and column and these are continued till the last row and column. All image partitions are further processed through the steps of the proposed image processing strategy, as shown in Figure 1. These image partitions are transformed into 1-D feature vectors (see section 3.4). It is important to note that at this point, DTW algorithm is used to measure the similarity between the template features vector and the vectors of the partition features to detect facial patterns and faces.

Partitioning process has been elaborated with the example given in Figure 7. In this figure, the first input image partition is virtually extracted (equivalent in the size of the template) from the first row and the first column, as shown by the sky blue colour. The second partition is extracted from the first row and the second column, which is shown in white (see Figure 7). In this way, the partitioning of an input test image goes through each column and row in a test image to find the optimal path between the two sequences. Thus, horizontally, a column and vertically a row becomes a *step size*.

```
//initialization
T= template image;
I= test image;
k=0;

[Trow,Tcol]=size(T);
[Irow,Icol]=size(I);

imax=Irow-Trow;
jmax=Icol-Tcol;

//iterations
for every i from 1 to imax
for every j from 1 to jmax
increment k
partition    I →Ipart=I(i:i+Trow,j:j+Tcol);

repeat for i
repeat for j
```

Fig. 6. Heuristic for image partitions
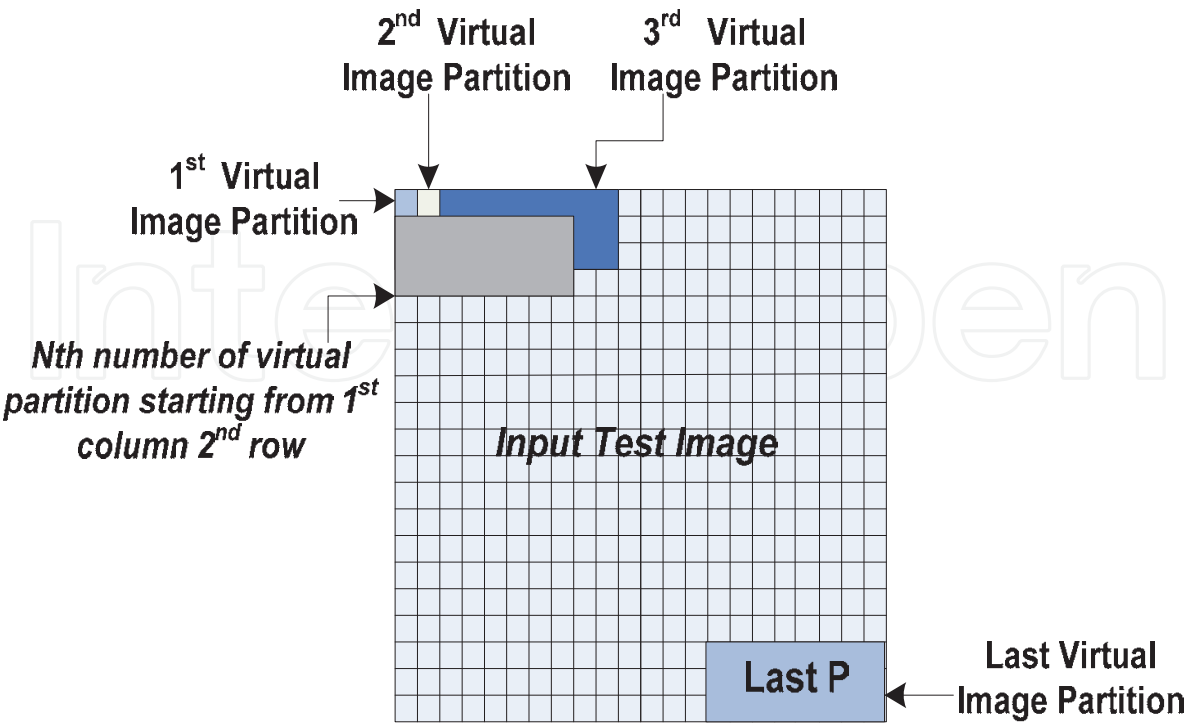
Fig. 7. Image Partitioning Process

3.4 Features vector extraction

To extract a particular feature from the image using 1-D DTW algorithm, horizontal and vertical projections are computed by integrating the columns and rows to produce 1-D vectors. The horizontal and vertical integral projections are presented in Equations 1 and 2, respectively.

$$Horizontal\ relief\ (y - \text{relief}) = \left[\sum_{j=1}^{N} I_{1j} \quad \sum_{j=1}^{N} I_{2j} \cdots \sum_{j=1}^{N} I_{Mj}\right] \tag{1}$$

$$Vertical\ relief\ (x - \text{relief}) = \left[\sum_{i=1}^{M} I_{i1} \quad \sum_{i=1}^{M} I_{i2} \cdots \sum_{i=1}^{M} I_{iN}\right] \tag{2}$$

In Equations 1 and 2, N is the width of the template/input partition image (the total number of columns available), M is the height of the template/input partition image (total number of rows). Figure 8 and 9 show the horizontal and vertical relief of the image (eyes region) respectively.
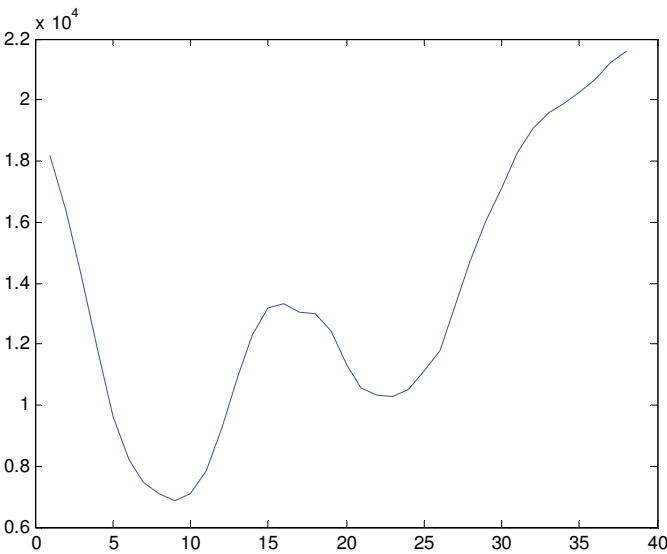
Fig. 8. Horizontal projections for image template
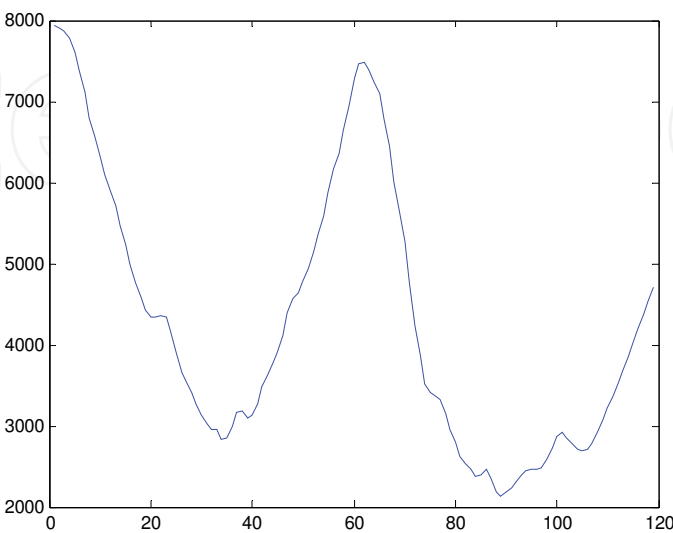
Fig. 9. Vertical projections for image template

The feature vector is formed by concatenating the $y_{\text{-relief}}$ and $x_{\text{-relief}}$ vectors. From Equations 1 and 2, the feature vector f can be obtained, as follows:

$$feature\ vector\ f = x - \text{relief} \oplus y - \text{relief} = [f_1 \quad f_2 \cdots f_k] \tag{3}$$

Where $\oplus$ denotes vector concatenation and k=N+M.

Equation 3 represents the feature vector extracted from a 2-D image. In the present work, the last element of the row sequence was joined to the first element of the column sequence. It was found through the experiment that an advantage of concatanating the horizontal and vertical projections in the features would include the cross correlation between the values in the rows and columns in the resulted accomulative distance matrix (in DTW algorithm). This row-column cross correlation has been found to make the classification less sensitive to the variation in the pose and orientation of the images. Figure 10 shows the result of cocatnation the vertical and horisontal vectors of the image template (eyes in this case).
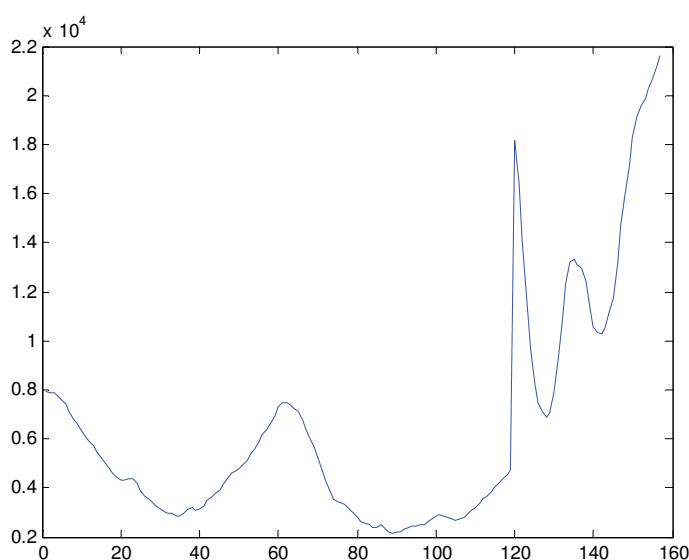
Fig. 10. Concatenated projections for image template

3.5 Dynamic Time Warping algorithm

Assume that there are two image vectors, I (the input image partition) and T (representing template image) each having noise n_I and n_T, the vectors are:

$$I_n = I + n_I = [I_1 + n_{I_1}\ I_2 + n_{I_2} \dots I_m + n_{I_m}] \tag{4}$$

$$T_n = T + n_T = [T_1 + n_{T_1}\ T_2 + n_{T_2} \dots T_k + n_{T_k}] \tag{5}$$

In Equations 4 and 5, m denotes the number of columns (the length) in the input image's partition I, and k denotes the number of column in the template image T (both vectors I and T have same length in the proposed method), whereas I_n and T_n are the input image partition and template image vectors with noise, respectively.

To align the image sequences, first a grid, I × T, local cost matrix is defined, where each cell represents the distance between the corresponding indices of the two feature-vectors. Normally, Euclidean distance is chosen to calculate the distance between the two elements in the matrix, d. The local distance matrix d can be computed as follows:

$$d(I,T) = |I - T| \tag{6}$$

The d matrix is of the size of m×k, presenting the cost values between each element of vector I and T, as shown in Equation 3.7 below:

$$d = \begin{bmatrix} I_1 + n_{I_1} - T_1 - n_{T_1} & \cdots & I_1 + n_{I_1} - T_k - n_{T_k} \\ \vdots & \ddots & \vdots \\ I_m + n_{I_m} - T_1 - n_{T_1} & \cdots & I_m + n_{I_m} - T_k - n_{T_k} \end{bmatrix} \tag{7}$$

Rearranging the terms in each element in Equation 7, the local distance matrix can be written as shown in Equation 8:

$$d(I_n, T_n) = d(I,T) + d(n_I, n_T) \tag{8}$$

After computing and building the local distance cost matrix (d) between two 1-D features' vectors I and T, a warping path (WP) is constructed by building a cumultaive/global cost matrix D(i,j). As previously mentioned (see section 2), the cumulative distance matrix D(i,j) is defined as the sum of the local distance d(i,j) found in the current cell and the minimum of the cumulative distances of the adjacent cells. A warping path is a concatenation of the cells starting from (1,1) to the cell (m,k). Let's define the minimum operator as follows:

$$M_{ij} = min[D_{i-1,j}, D_{i,j-1}, D_{i-1,j-1}] \tag{9}$$

Where M_{ij} is the minimum of the three cells in the global cost matrix D, from where the cell can be reached (depicted in Figure 11).
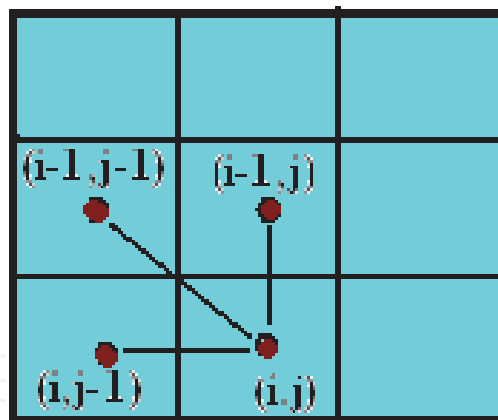
Fig. 11. The cell (i, j) and the three cells surrounding it.

Therefore, the global distance matrix (or the accumulative distance) can be formed from *d* (*I, T*) in Equation 7, as presented in Equation 10.

$$D = \begin{bmatrix} 0 & \infty \ldots & \infty \\ \infty & d_{1,1} + M_{2,2} \ldots & d_{1,k} + M_{2,k+1} \\ \vdots & \vdots & \vdots \\ \infty & d_{m,1} + M_{m+1,2} \ldots & d_{m,k} + M_{m+1,k+1} \end{bmatrix} \tag{10}$$

Applying the minimum operator and the steps to create the global distance matrix will result in building matrix D. Equations 11-21 represent the process of building the global cost matrix D.

The first element in the D matrix is equivalent to zero.

The entire first row and first column of the global matrix D is filled by infinity to ensure that the first element D (1,1) is the minimum. The second element of D, D (2, 2) is filled as follows:

$$D(2,2) = d(1,1) + M_{2,2} \tag{11}$$

$$D(2,2) = d(1,1) + minimum(D_{1,2}, D_{2,1}, D_{1,1}) \tag{12}$$

$$D(2,2) = d(1,1) + minimum(inf, inf, 0) \tag{13}$$

$$D(2,2) = I_1 - T_1 + 0 = I_1 - T_1 \tag{14}$$

Proceeding in the same manner for the second column of D and the second row of D:

$$D(3,2) = d(2,1) + M_{3,2} \tag{15}$$

$$D(3,2) = d(2,1) + minimum(D_{2,2}, D_{3,1}, D_{2,1}) \tag{16}$$

$$D(3,2) = d(2,1) + minimum(A_1 - B_1, inf, inf) \tag{17}$$

$$D(3,2) = I_2 - T_1 + I_1 - T_1 = \sum_{l=1}^{2} I_l - 2T_1 \tag{18}$$

In the same manner, the contents of the second column can be written as:

$$D(m,2) = \sum_{l=1}^{m-1} I_l - (m-1)T_1 \tag{19}$$

Once again, the second row elements of the matrix D can be expressed in the same manner, as:

$$D(2,k) = (k-1)I_1 - \sum_{w=1}^{k-1} T_w \tag{20}$$

The rest of D elements can be computed on the same manner, D will be:

$$D = \begin{bmatrix} 0 & \infty \dots & \infty \\ \infty & I_1 - T_1 \dots & (k-1)I_1 - \sum_{w=1}^{k-1} T_w \\ \vdots & \vdots & \vdots \\ \infty & \sum_{l=1}^{m-1} I_l - (m-1)T_1 \dots & d_{m,k} + M_{m+1,k+1} \end{bmatrix} \tag{21}$$

Constraints are applied for constructing the optimal warping path (Sakoe & Chiba 1978), (see section 2 for more details).

- Monotonicity: The alignment path does not go back in the time index, $i_{l-1} \le i_l$ and $j_{w-1} \le j_w$.
- This guarantees that the features are not repeated in the alignment (see Figure 3.12).
- Continuity: The alignment does not jump in the time index, $i_l - i_{l-1} \le 1$ and $j_w - j_{w-1} \le 1$. This guarantees that the important features are not omitted. Continuity condition is shown in Figure 12, along with the monotonicity constraint.
- Boundary: The alignment starts at the beginning of the matrix and ends at the bottom-right. This guarantees that the sequences are not considered only partially. The blue boxes in Figure 12 show that the path starts at D (1,1) and ends at D (m,k).
- Warping window: A good alignment path is unlikely to wander too far from the diagonal. This guarantees that the alignment does not try to skip different features or

get stuck at similar features, and allowable points must fall within a given window, $|i_l - i_{l-1}| \leq w$; where w is a positive integer window width (threshold). Windowing constraint appears in Figure 12, as the two dark red lines surrounded the purple diagonal line.

Figure 12 illustrates the global cost matrix together with the warping path and the constraints applied to the paths.
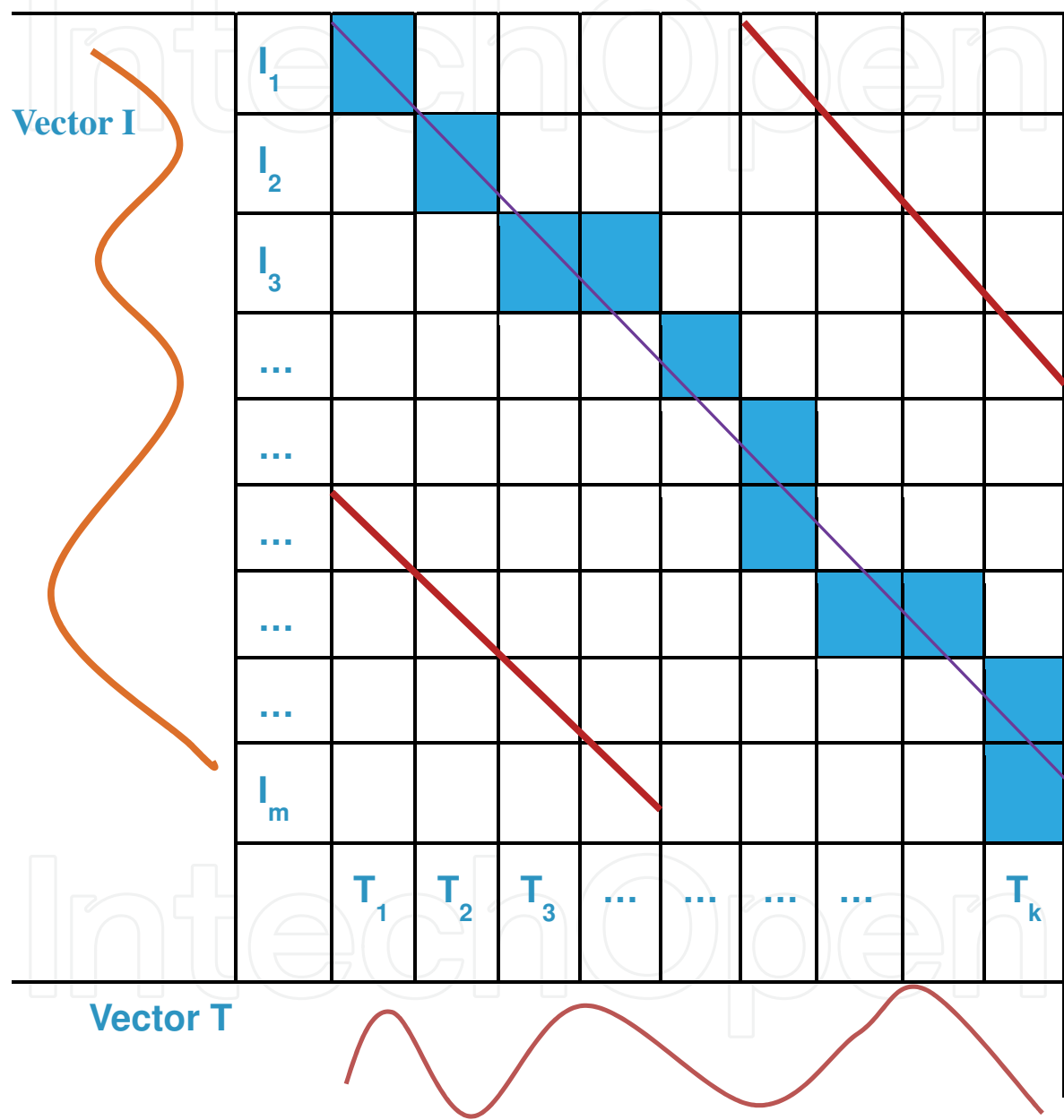
Fig. 12. Alignment between the two image vectors I and T

4. Working example of the Dynamic Time Warping algorithm for face detection

The proposed algorithm is implemented in MatLab 9 running on a computer with 2.0GHz dual core OPTERON processor and 1.87GB RAM. To evaluate the proposed techniques and

method, two publicly available databases were used in this study. One database, containing face images with a resolution of 512 x 342 pixels under controlled lighting conditions and different rotations, is available from University of Bern, Switzerland [47]. Additionally, face images have changes in their facial expressions and immediate changes in head poses (right, left, up down and straight). The second one is the BioID face images database with the resolution of 384 x 286 pixels [46]. This database has face images with a complex background, along with different expressions and various illumination conditions, poses, and rotations. We conducted the experiments in different settings to evaluate the efficacy of our approach. In first setting, we applied the technique for eyes localization using DTW without our image processing strategy; the detection rate was very low. In second setting, we applied our image processing strategy with histogram equalization and DTW algorithm on the same dataset, the detection rate improved within 35%. Figure 13 shows the arbitrary eyes template chosen for our experiment.
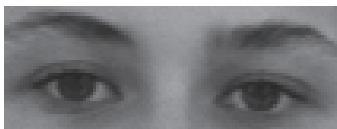
Fig. 13. The arbitrary template of eyes

With our experimental analysis and observation, histogram equalization reduced the effect of light on intensity distribution.

Comparing to other template matching methods [45] our system is simple fast and easy to implement with no training data set and shape model is needed, and template of eyes is manually cropped. The proposed method is showing acceptable detection rate and less time complexity comparing to other classifier and distance measurement systems.

Using the proposed image processing strategy, features vector were extracted (see section 3.4). With these extracted features, histogram equalization was applied with a flat envelop for both the template image and input test image(s) so as to re-distribute the intensities throughout the range. Histogram equalization distributes the intensities of an image throughout a range of histogram. The better distribution of the intensities has helped to detect facial patterns and face in bright background with existing illumination and lighting conditions. For example, the face image shown in Figure 14(a) was processed in Experiment setting 1 for the eye detection. Figure 14(a) shows that the eyes were not detected before histogram equalization was applied. Figure 14(b) illustrates the alignment between both the template and input image template vectors. Before that figure 14(c) depicts the histogram distribution for the image shown in Figure 14(a). It is obvious that both vector sequences are not well aligned. With our experimental analysis, histogram equalization was applied to reduce the effect of light on intensity distribution. The same facial image is shown in Figure 15(a) after applying histogram equalization. From the figure, it can be seen that the eyes have been detected correctly. Furthermore, both the vectors have been well aligned, as shown Figure 15(b) below:

(a)

(b)

(c)
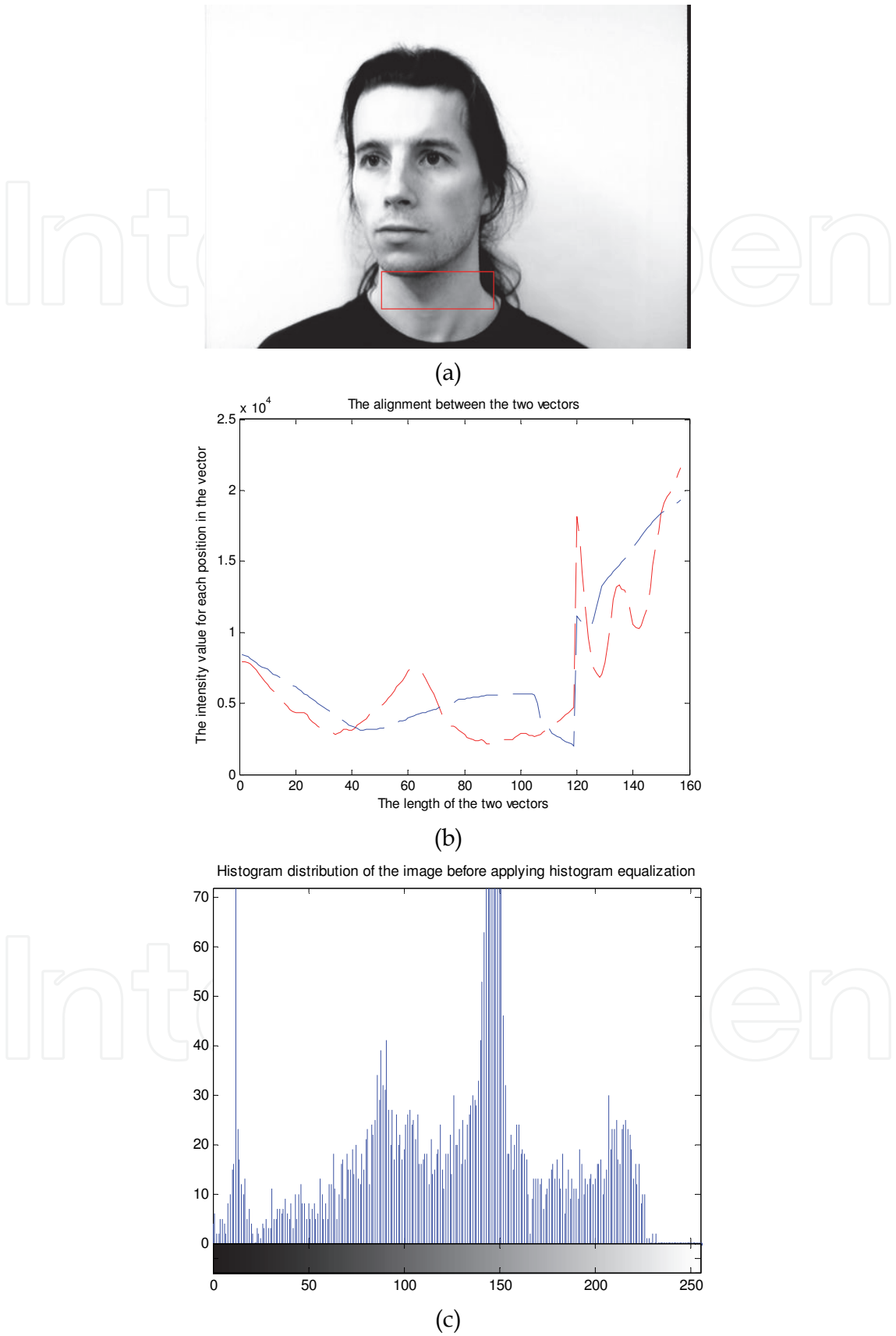
Fig. 14. Before applying image processing strategy

(a)

(b)

(c)
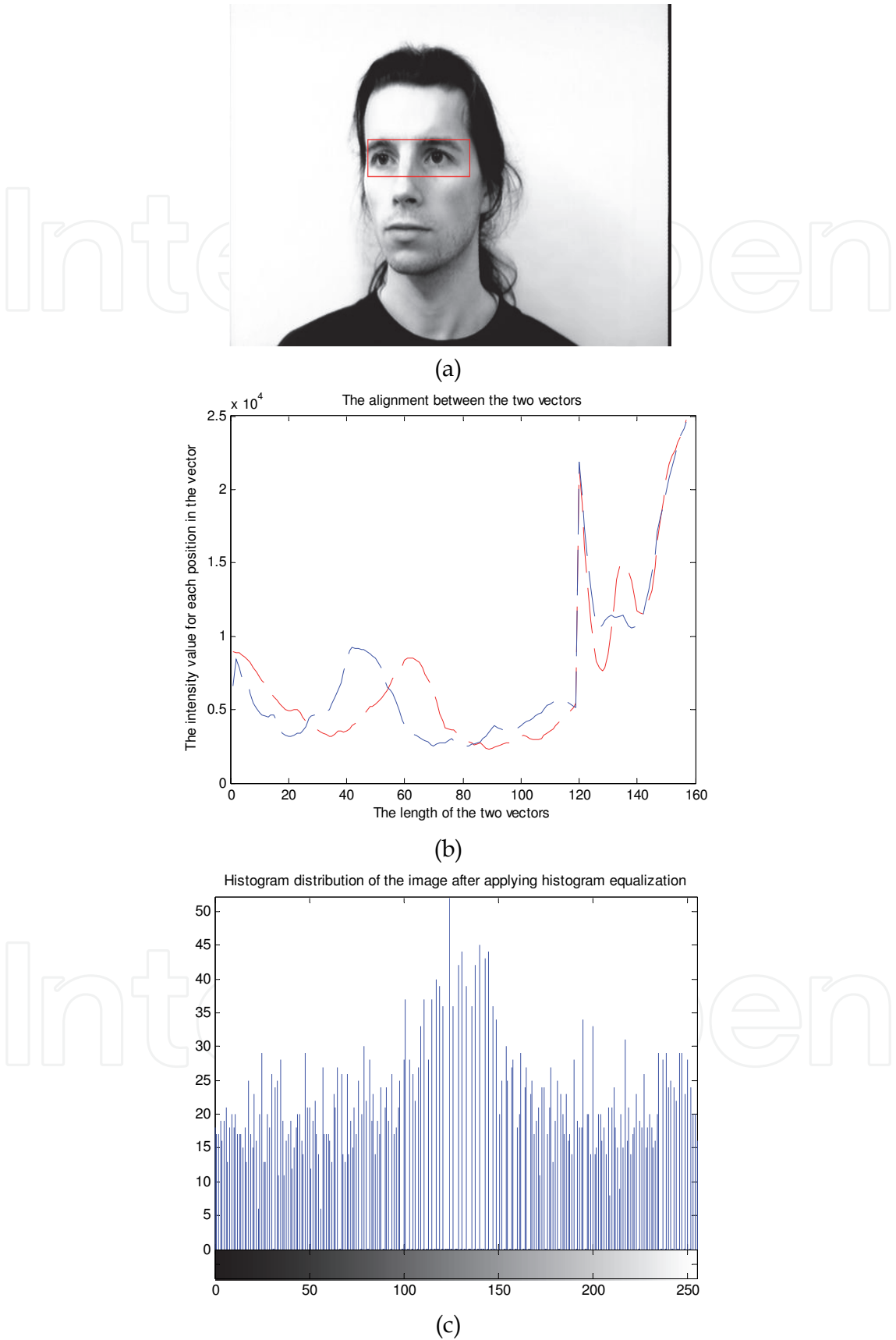
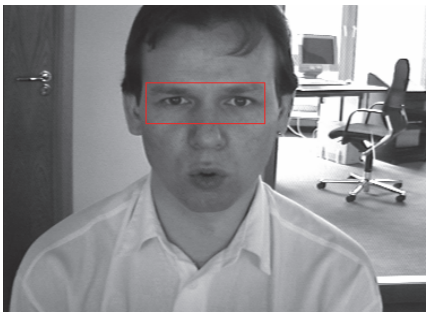Fig. 15. Correct detection of the eyes after applying image processing strategy

Figure 16 and 17 show some of the image faces in different poses and lighting conditions, which are detected correctly and the alignment between the template image and the test images corresponding to these images, respectively.
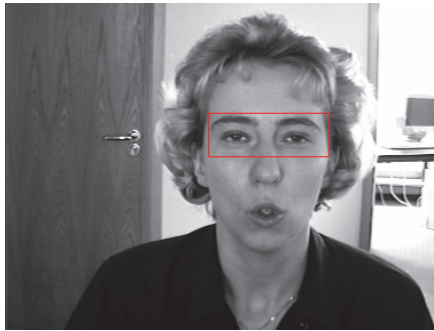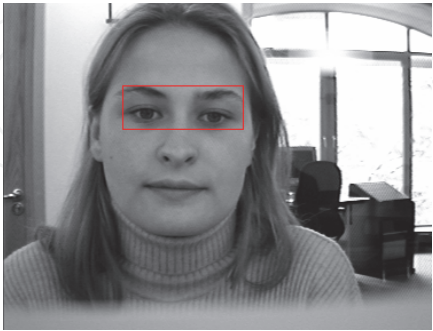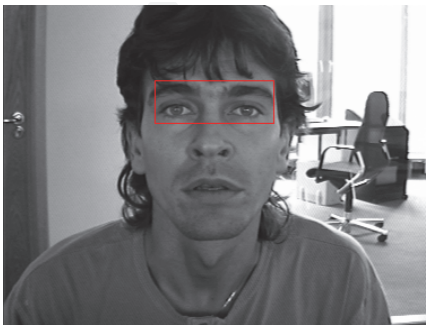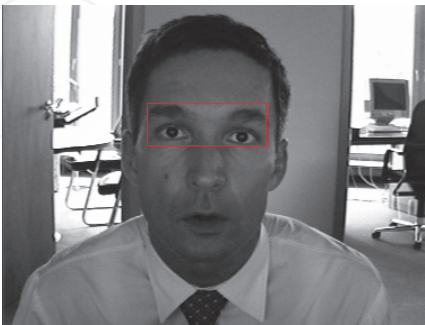

(a)   (b)

(c)   (d)

(e)   (f)

Fig. 16. (a-f) Face Images with positive eyes localized. (BioID images database [22])
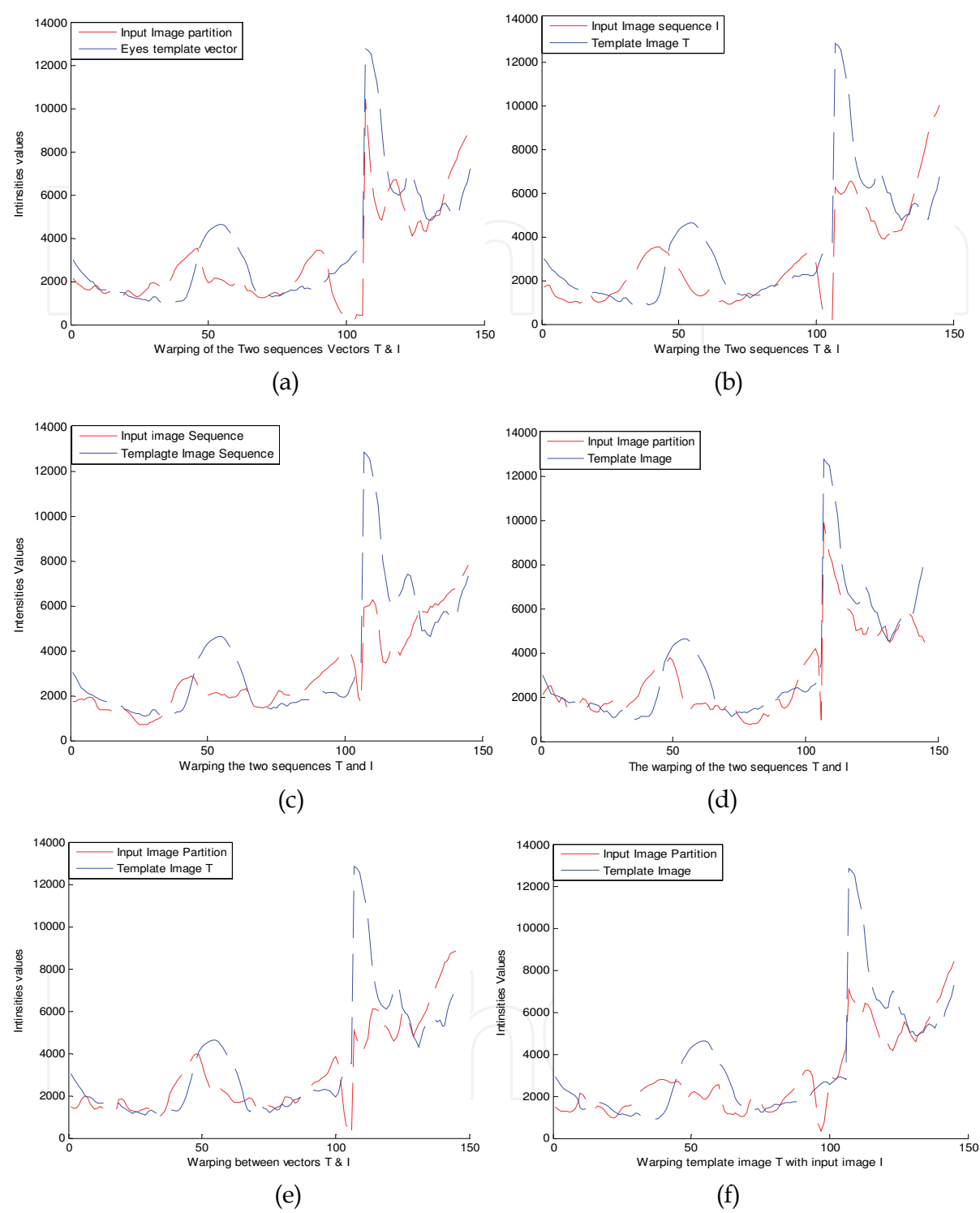
Fig. 17. (a-f) the corresponding graph of the warping path for the same image presented in Figure 16.

5. Conclusions

We have presented our approach of template-based matching method and Dynamic time warpping to efficiently enhance the detection and localization of eyes in complex

---

[14] Hichem S., and Nozha B., "Robust Face Recognition Using Dynamic Space Warping", Lecture Notes in Computer Science 2359(2002): 121-132.

[15] Alexandre L. M. L., Débora C. C., Denis H. P. S., José H. S. and Nelson D. A. M., "Novel Approaches for Face Recognition: Template-Matching using Dynamic Time Warping and LSTM Neural Network Supervised Classification", (Presented in the *15th International Conference on Systems, Signals and Image Processing, 2008. IWSSIP 2008.*, pp.241-244, 25-28 June 2008).

[16] Scott M., George L., "Elastic matching in linear time and constant space", (DAS '10, June 9-11, 2010, Boston, MA, USA).

[17] Han-Wen H.,Wen-H., Cheng-K. and Jeffrey J. P. T., "A Novel Feature with Dynamic Time Warping and Least Squares Adjustment for Protein Structure Alignment", Asian Journal of Health and Information Sciences 1(3)(2006): 261-275.

[18] Aach J, Church GM, "Aligning gene expression time series with time warping algorithms", Bioinformatics. 17(6)(2001):495-508.

[19] Jo C. and Elena T., "Gene Time *Ex*pression Warper: a tool for alignment, template matching and visualization of gene expression time series", Bioinformatics Applications Note 22 (2) (2006): 251–252.

[20] Benoˆt L., Chang C.S. , Ong S.H., Soek-Ying N., Nallasivam P., "Chromosome classification using dynamic time warping", Journal of Pattern Recognition Letters 29(2008): 215–222.

[21] Aparna L. R., Grimson W. E. L., And William M. W. I., "Object Detection and Localization by Dynamic Template Warping", *International Journal Of Computer Vision* 36(2)(2000)131-147.

[22] Santosh K.C, "Use of Dynamic Time Warping for Object Shape Classification through Signature", Kathmandu University Journal of Science, Engineering And Technology 6(I) (2010): 33-49.

[23] Kongming W. and Theo G., "Alignment of curves by dynamic time warping", Annals of Statistics 25(3) (1997): 1251-1276.

[24] Longin J. L., Vasileios M., Qiang W., Deguang Y., "An elastic partial shape matching technique." Pattern Recognition 40(2007): 3069-3080.

[25] Joan C., Joaquim M., Fco. I. G., "Pattern Recognition Based On Episodes and DTW, Application To Diagnosis Of A Level Control System", (Presented at the 16th International Workshop on Qualitative Reasoning, 2002).

[26] Hansheng L., Govindaraju, "Direct Image Matching by Dynamic Warping", (Presented in the Computer Vision and Pattern Recognition Workshop CVPRW '04: 76- 76, 27-02 June 2004).

[27] Berndt D. and Clifford J., "Using dynamic time warping to find patterns in time series", (Presented at AAAI-94 Workshop on Knowledge Discovery in Databases, 1994).

[28] Keogh E., "Exact indexing of dynamic time warping", *Knowledge and Information Systems* 7 (3) (2005): 358-386.

[29] Keogh E.and Pazzani M., "Derivative Dynamic Time Warping", (Presented at the SIAM International Conference on Data Mining, Chicago, 2001).

[30] Ratanamahatana A. and Keogh E., "Making Time-series Classification More Accurate Using Learned Constraints", (Presented in the SIAM Intl. Conf. on Data Mining, pp. 11-22, Lake Buena Vista, Florida, 2004).

[31] Yasushi S., Masatoshi Y., and Chrisos F., "FTW: Fast Similarity Search under the Time Warping Distance". (Presented In *PODS*, pp. 326–337, 2005).

[32] Yutao S., Nikos M., and David W. C., "Fast and Exact Warping of Time Series Using Adaptive Segmental Approximations", Machine Learning 58 (2005): 231–267.

[33] Daniel E. R., Svetha V., and Wanquan L., "Threshold dynamic time warping for spatial activity recognition", *International Journal of Information and Systems Sciences* (3) (2007): 392-405.

[34] Daniel L., "Faster Sequential Search with a Two-Pass Dynamic-Time-Warping Lower Bound", arXiv:0807.1734v4 [cs.DB], Oct 2008.

[35] Johannes A., Thomas B., Hans-Peter K., Peer K., and Matthias R., "Periodic Pattern Analysis in Time Series Databases. In Database Systems For Advanced Applications", Lecture Notes In Computer Science, 5463 (2009): 354-368.

[36] Clifford D., Stone G., Montoliu I., Rezzi S., Martin F., Guy P., Bruce S., And Kochhar S., "Alignment Using Variable Penalty Dynamic Time Warping", *Anal. Chem* 81(2009): 1000-1007.

[37] Xavier A., Robert M., Nuria O., "Partial Sequence Matching Using an Unbounded Dynamic Time warping Algorithm", (ICASSP 2010).

[38] Chaovalitwongsea W. A. and Pardalos P. M., "On the Time Series Support Vector Machine Using Dynamic Time Warping Kernel For Brain Activity Classification", Cybernetics and Systems Analysis 44(1) (2008): 125-138.

[39] Marek I., Agnieszka M., "Voice control", (Presented in the XXIV Symposium *Vibrations in Physical Systems,* Poznan – Bedlewo, May 12-15, 2010).

[40] Salvador S., Chan P., "FastDTW: Toward Accurate Dynamic Time Warping In Linear Time and Space", (Presented in the *3rd KDD workshop on mining temporal and sequential data*, pp.70-80, 2004).

[41] Cha Z. and Zhengyou Z., "A Survey of Recent Advances in Face Detection", (Microsoft Research, Microsoft Corporation. June 2010).

[42] Kawaguchi T., Rizon M., "Iris Detection Using Intensity And Edge Information", Journal of Pattern Recognition 36(2) (2007): 549 –562.

[43] Luis, E., Raul, P., Jonathan V., "Face Localization In Color Images Using Dynamic Time Warping and Integral Projections", (Presented in the International Joint Conference On Neural Networks, August 12-17, 2007, Orlando, Florida, USA).

[44] Schwenker F., Andreas, S., Palm G., Kestler H. A., "Orientation Histograms for Face Recognition", Artificial Neural Networks in Pattern Recognition 4087(2006): 253 – 259.

[45] Qiong W., Jingyu Y., "Eye Detection in Facial Images with Unconstrained Background", Journal of Pattern Recognition Research 1(2006): 55-62.

[46] Jesorsky O., Kirchberg K., Frischholz R. The BioID face database. *HumanScan,* 2001,[Online]. Available: http://www.bioid.com/downloads/facedb/index.php.

[47] B. Achermann, The face database of University of Bern [http://iamwww.unibe.ch/fkiwww/staH/achermann.html], Institute of Computer Science and Applied Mathematics, University of Bern, Switzerland, 1995.

[48] Somya A., Hamzah A., "Dynamic Time Warping Approach Toward Face Patterns Detection", International Journal Of Academic Research 3(1)(2011): 89-95.

[49] Somya A., Hamzah A., "A New Approach for an Efficient DTW in Face Detection through Eyes Localization", Journal of Electronics and Electrical Engineering 2(108) (2011): 103-108.
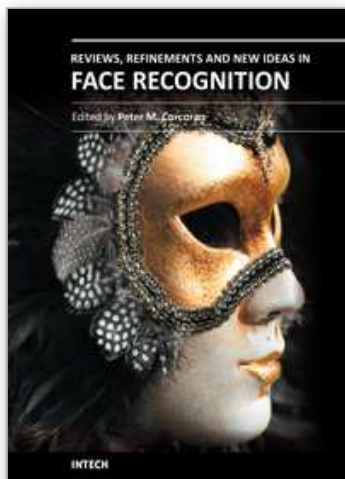
Reviews, Refinements and New Ideas in Face Recognition

Edited by Dr. Peter Corcoran

As a baby one of our earliest stimuli is that of human faces. We rapidly learn to identify, characterize and eventually distinguish those who are near and dear to us. We accept face recognition later as an everyday ability. We realize the complexity of the underlying problem only when we attempt to duplicate this skill in a computer vision system. This book is arranged around a number of clustered themes covering different aspects of face recognition. The first section on Statistical Face Models and Classifiers presents reviews and refinements of some well-known statistical models. The next section presents two articles exploring the use of Infrared imaging techniques and is followed by few articles devoted to refinements of classical methods. New approaches to improve the robustness of face analysis techniques are followed by two articles dealing with real-time challenges in video sequences. A final article explores human perceptual issues of face recognition.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

INTECH
open science | open minds