Genetic Approach for Video Based Face Recognition

Om Prakash Sinha #1, Gargishankar verma*2

Deptt. Of Computer Science & Engineering , CIET, Raipur

Chhattisgarh, Bilaspur – India

Abstract

Extracting high-level features is an important field in video indexing and retrieving. Identifying the presence of a human in the video is one of these high-level features, which facilitate the understanding of other aspects concerning people or the interactions between people. In this paper, we are giving an approach for face recognition based on a video. Face recognition in videos is a superheated topic in computer vision and biometrics for many years. Compared to traditional face analysis, video-based face recognition has the advantages of more plentiful information to improve accuracy and robustness but also suffers from large-scale variations, low quality of facial images, illumination changes, pose variations, and occlusions. Related to applications, we divide the existing video-based face recognition approaches into two categories: video-image-based methods and video-video-based methods, which are surveyed and analyzed in this paper.

Keywords: - *Video Processing, Computer vision, Human detection, Face recognition*

I. INTRODUCTION

In recent years, face recognition is always a vibrant topic in the field of biometrics. Compared to traditional face recognition in still images, video-based face recognition has significant benefits listed as follows. Firstly, videos contain more abundant information than a single image. As a result, more robust and sturdy recognition can be achieved by fusing information of multi frames. Secondly, temporal information becomes available to be exploited in videos to improve the accuracy of face recognition. Finally, multi poses of faces in videos make it possible to explore shape information of face and combined it into the framework of face recognition. However, video-based face recognition is also a very challenging problem, which suffers from low-quality facial images, illumination changes, pose variations, occlusions, and so on. Due to its prominence and difficulties, many research institutes have concentrated on video-based face recognition with all kinds of strategies proposed, such as Massachusetts Institute of Technology [1], Carnegie Mellon University [2, 3], the University of Illinois at Urbana-Champaign [4, 5], University of Maryland [6–8], University of Cambridge [9–11], Toshiba [12, 13], Institute of Automation Chinese Academy of Sciences [14, 15]. Related to applications, we can divide video-based face recognition methods into two categories: video-image-based methods and video-video-based methods. The first category can be seen as an extension of still image-based face recognition. Face recognition can establish a subject's identity based on facial characteristics. Automated face recognition requires various techniques from different research fields, including computer vision, image processing, pattern recognition, and machine learning. In a typical face recognition system, face images from several subjects are enlisted into the system as gallery data, and the face image of a test subject (probe image) is matched to the gallery data using one-to-one or one-to-many schemes. The one-to-one and one-to-many matching's called verification and identification, respectively. Face recognition is one of the fundamental methods used by human beings to interact with each other. Attempts to match faces using a pair of photographs dating back to 1871 in a British court [9]. Techniques for automatic face recognition have been developed over the past three decades for automatic person recognition with still and video images. Face recognition has a wide range of applications, including law enforcement, civil applications, surveillance systems, and many more. Face recognition applications have also been extended to smart home systems where the recognition of the human face and expression is used for better interactive communications between humans and machines. Fig. 1.1 shows some biometric applications using the face. The face has several advantages that make it one of the most preferred biometric traits. First, the face biometric is easy to capture even at a long distance. Second, the face conveys not only the identity but also the inner feelings (emotion) of the subject (e.g., happiness or sadness) and the person's age. This makes face recognition a vital subject in human-computer interaction as well as a person's recognition. The face biometric is affected by several intrinsic (e.g., expression and age) and extrinsic (e.g., pose and lighting) variations. While there has been a significant improvement in face recognition performance during the past decade, it is still below acceptable levels for use in many applications.Recent efforts have focused on using 3D models, video input, and different features (e.g., skin texture) to overcome the performance bottleneck in 2D still face recognition [36].

Figure 1.1. Example applications using face biometric: (a) ID cards (from [1]), (b) face matching and retrieval (from [2]), (c) access control (from [2]), and (d) DynaVox EyeMax system (controlled by eye gaze and blinking, from [3]).

II. RELATED WORK

A: Feature Identification

Most of the studies in this field use face detection algorithms as the key idea. Jin [15] proposed a method to identify video shots with people based on face detection [38]. The category of the shot was considered to be "people", only if there is at least one image with more than one face within that shot. One of the three features chosen by Huang et al. [16] to be evaluated in the TREC video Evaluation (2003) was the "People" feature, Huang et al., state that for a segment of video to have people feature it should contain at least three human faces. Huang et al. used a skin tone filter to detect skin regions, followed by the Omni-face detection algorithm which was proposed by Wei and Sethi [12].

B: Human Detection

From the literature reviews done, it can be concluded that the most common way in human detection is via detecting the human face. The human face is the most unique part of the human body, and if it is accurately detected it leads to robust human existence detection.

C: Face Detection methods

Several studies were done in the face detection domain since 1970, and lots of surveys addressed the algorithms used in this field under different categories [5], [13], [14] but in general two main classes can be used to classify these algorithms namely, feature-based (e.g. Bottom-Up) and image-based (e.g. Appearance-Based and Template matching) approaches. Features-based techniques extract facial features from an image and manipulate its parameters such as angles, size, and distances. Image-based approaches rely on a training and learning set of examples of objects of interest. However, dealing with video introduces other approaches for face detection such as the motion-based approach. A brief description of the most common approaches and examples of algorithms used in each of them is given in the rest of this section.

C1: Knowledge-based (Top-Down) approach

In this method, the relationship between facial features is captured to represent the contents of a face and encode it as a set of rules. Coarse-to fine scale is used in lots of algorithms classified under this category, in which the coarsest scale is searched first and then proceeds with the others until the finest scale is reached.

C2: Feature invariant (Bottom-Up) approach

In this approach, the face's structural features do not change under different conditions such as varying viewpoints, pose angles, and/or lighting conditions. Common algorithms used under this category are the Color-based approach or the so-called skin-model-based approach. This approach makes use of the fact that the skin color can be used as an indication of the existence of humans using the fact that different skins from different races are clustered in a single region. Cezhnevets et al., [20], presented 4 pixel-based skin modeling techniques named Explicitly defined skin region, Non-parametric skin distribution modeling, Parametric skin distribution modeling, Dynamic skin distribution modeling.

C3: Facial features based approach

This method, in which global (e.g. skin, size, and shape) and/or detailed (e.g. eyes, hose, and lips) features are used, has become popular recently. Mostly, the global features first are used to detect the candidate area and then tested using the detailed features. Texture The human face differs from other objects in texture. This approach examines the likelihood of sub-image to belong to human face texture, using the Space Gray Level dependency (SGLD) matrix.

C4: Template matching methods

These methods are based on measuring the degree of similarity between the candidate sub-image and the predefined stored face pattern. The predefined image might be for the whole face pattern or the individual face features such as eyes, nose, and lips. Common algorithms used under this category are Predefined face templates, in which several templates for the whole, individual, or both (whole and individual) parts of a face are stored. Deformable Templates in which an elastic facial feature model as a reference model where the deformable template model of the object of interest, is fitted in[37].

C5: Appearance-Based Method

Unlike template matching methods, where the templates are predefined by experts, the Appearance-Based method learns the templates from a set of images, using statistical analysis and machine learning. Examples of algorithms used by these approaches are:

Eigenfaces, or so-called eigenvectors, in which different algorithms are used to approximate the eigenvectors of the autocorrelation matrix of a candidate image. [17]

Distributed-Based, where the distribution pattern of an object is learned using the positive and negative image sets of that object.

Neural Networks, where networks of neurons (simple Elements) called nodes are used to perform a function in parallel. The idea of neural networks comes from the central nervous system. However, these networks are trained to detect the presence of face by giving it face and no face samples.

Support Vector Machines, are learning machines that make binary classifications. The idea here is to maximize the margin between positive and negative sets of vectors and obtain an optimal boundary that separates the two sets of vectors. They were first suggested by Vapnik in 1960 [4].

Hidden Markov Model is a statistical model used to model the statistical properties of a signal. The Markov process is used to model the processed system and the Markov parameters are taken from the observed parameters.

D: Movement Detection

Unlike still images, video sequences hold more details about the history of moving objects (foreground), which helps in isolating the foreground from the background. Generally, the moving areas are detected by finding the changes that happen among the sequences of images [1], [2]. Most of the research done in movement detection applied pre-processing steps before applying the change detection algorithms, [2]. Such pre-processing steps involve geometric and intensity adjustments. The problem of variation in light intensity is solved by intensity adjustment in which the illumination effect is reduced to some degrees based on the method used. Elgammal et al. [1], state that transforming the RGB values, into chromatic color space makes the module insensitive to the small changes in the illumination. There are several ways for detecting a change in a video sequence [2]. Recent studies agree that the Image differencing method is more effective than others in change detection [3].

III PROPOSED METHODOLOGY

In this section of the paper, we have discussed the approach that we had implemented. The flow chart of the Frame Extraction Algorithm is shown in figure 3.1. For implementation and analysis, we have used Matlab software.

Step 1: Compress & Uncompress Video Format: Captured video that has not been altered by the capture device or software to compress the data. Uncompressed video streams take up a lot of disk space and more transmission bandwidth but it provides the best quality.

Step 2: Extract the number of frames from the input video.

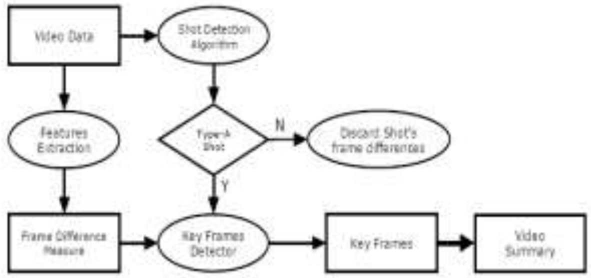

Figure 3.1 Frame Extraction Algorithms

Frame difference measure: In general, a single visual descriptor cannot capture all the pictorial details needed to estimate the changes in the visual content of frames and the visual complexity of a video shot. In defining what a good pictorial representation of a frame is, we must take into account both color properties and structural properties, such as texture. Instead, as stated in Section 2, many existing algorithms use only one feature. To overcome the frame representation problem, we compute three different descriptors: a color histogram, an edge direction histogram, and wavelet statistics. The features used have been selected for three basic properties: perceptual similarity (the feature distance between two images is large only if the images are not "similar"), efficiency (the features can be rapidly computed), and economy (small dimensions that do not affect efficacy). The use of these assorted visual descriptors provides a more precise representation of the frame and captures slight variations between the frames in a shot. Color histograms are frequently used to compare images because they are simple to compute, and tend to be robust regarding small changes in camera viewpoint. Retrieval using color histograms to identify objects in image databases has been investigated in [29] and [30]. An image histogram h( ) refers to the probability mass function of image intensities. Computationally, the color histogram is formed by counting the number of pixels belonging to each color. Usually, a color quantization phase is performed on the original image to reduce the number of colors to consider in computing the histogram and thus the size of the histogram itself. The color histogram we use is composed of 64 bins determined by sampling groups of meaningful colors in the HSV color space [35]. The use of the HSV color space allows us to carefully define groups of colors in terms of Hue, Saturation, and Lightness. The edge direction histogram is composed of 72 bins corresponding to intervals of 2.5 degrees. Two Sobel filters are applied to obtain the gradient of the horizontal and the vertical edges of the luminance frame image [32]. These values are used to compute the gradient of each pixel; those pixels that exhibit a gradient above a predefined threshold are taken to compute the gradient angle and then the histogram. The threshold has been heuristically set at the 4% of the gradient maximum value to remove from the histogram computation edges derived from background noise. Multiresolution wavelet analysis provides representations of image data in which both spatial and frequency information are present [33]. In multiresolution wavelet analysis, we have four bands for each level of resolution resulting from the application of two filters, a low-pass filter (L) and a high-pass filter (H). The filters are applied in pairs in the four combinations, LL, LH, HL, and HH, and followed by a decimation phase that halves the resulting image size. The final image, of the same size as the original, contains a smoothed version of the original image (LL band) and three bands of details (see Fig. 3.2).

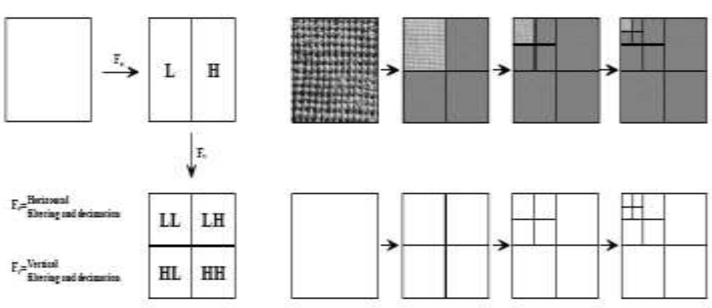

Figure 3.2 The filtering and decimation of the image along the horizontal and vertical directions. Four bands are created each a quarter the size of the whole image. b) The three-step application of the multiresolution wavelet. The wavelet filters are applied to the top left band containing the resized image.

Each band corresponds to a coefficient matrix that can be used to reconstruct the original image. These bands contain information about the content of the image in terms of general image layout (the LL band) and terms of details (edges, textures, etc..). In our procedure, the features are extracted from the luminance image using a three-step Daubechies multiresolution wavelet decomposition that uses 16 coefficients and produces ten sub-bands [34] (Fig. 3.2(b). Two energy features, the mean and standard deviation of the coefficients, are then computed for each of the 10 sub-band obtained, resulting in a 20-valued descriptor.

To compare two frame descriptors, a difference measure is used to evaluate the color histograms, wavelet statistics, and edge histograms. There are several distance formulas for measuring the similarity of color histograms. Techniques for comparing probability distributions are not appropriate because it is a visual perception that determines the similarity, rather than the closeness of the probability distributions. One of the most commonly used measures is the histogram intersection [29]. The distance between two color histograms (dH) using the intersection measure is given by:

$$d_H(H_t, H_{t+1}) = 1 - \sum_{j=0}^{63} \min(H_t(j), H_{t+1}(j))$$

where Ht and Ht+1 are the color histograms for frame F(t) and frame F(t+1) respectively.

The difference between two edge direction histograms (dD) is computed using the Euclidean distance as such in the case of two wavelet statistics (dW):

$$d_D(D_t, D_{t+1}) = \sqrt{\sum_{j=0}^{71} (D_t(j) - D_{t+1}(j))^2}$$

$$d_W(W_t, W_{t+1}) = \sqrt{\sum_{j=0}^{19} (W_t(j) - W_{t+1}(j))^2}$$

Where Dt and Dt+1 are the edge direction histograms and Wt and Wt+1 are the statistics of the wavelet for frame F(t) and frame F(t+1). The three resulting values (to simplify the notation we have indicated them as dH, dW, and dD only) are mapped into the range [0,1] and then combined to form the final frame difference measure (dHWD) as follow:

$$d_{HWD} = (d_H \cdot d_W) + (d_W \cdot d_D) + (d_D \cdot d_H)$$

The frame difference measure aims to accentuate dissimilarities to detect changes within the frame sequence. At the same time it is important that only when the frames are very different, the measure should report high difference values. As told before, the majority of the keyframe selection methods exploit just one visual feature which is not sufficient to effectively describe an image's contents. If we were to use, for example, only the color histogram, a highly dynamic sequence (e.g. one containing fast-moving or panning effects) with frames of the same color contents, would result in a series of similar frame difference values and the motion effects would be lost. Similarly, frames with the same color content but different from the point of view of other visual attributes are considered similar. The use of multiple features can overcome these issues but pose the problem of their combination. In content-based retrieval systems, the features are combined by weighing them with suitable factors which are usually task-dependent [31]. We choose instead to use a different approach: the explicit selection of weight factors is removed by weighing each difference against the other. Moreover, this allows us to register significant differences in the dHWD values only if at least two of the single differences exhibit high values (and thus two of the visual attributes emphasize the frame dissimilarity).

Keyframe selection: The keyframe selection algorithm that we propose dynamically selects the representative frames by analyzing the complexity of the events depicted in the shot in terms of pictorial changes. The frame difference values initially obtained are used to construct a curve of the cumulative frame differences which describes how the visual content of the frames changes over the entire shot, an indication of the shot's complexity: sharp slopes indicate significant changes in the visual content due to a moving object, camera motion, or the registration of a highly dynamic event. These cases must be taken into account in selecting the keyframes to include in the shot summary. They are identified in the curve of the cumulative frame differences as those points at the sharpest angles of the curve (curvature or corner points). The keyframes are those corresponding to the midpoints between each pair of consecutive curvature points. To detect the high curvature points we use the algorithm proposed by Chetverikov et al. [35]. The algorithm was originally developed for shape analysis to identify salient points in a 2D shape outline. The high curvature points are detected in two-pass processing. In the first pass, the algorithm detects candidate curvature points. The algorithm defines a "corner" a location where a triangle of a specified size and opening angle can be inscribed in a curve. Using each curve point P as a fixed vertex point, the algorithm tries to inscribe a triangle in the curve, and then determines the opening angle a(P) in correspondence of P. Different triangles are considered using points that fall within a window of a given size w centered in P; the sharpest angle is retained as a possible high curvature point. This procedure is illustrated in Fig. 3.3. Defining the distance between points P and O as dPO, the distance between points P and R as dPR, and the distance between points O and P as dOP, the opening angle corresponding to the triangle OPR is computed as:

$$\alpha = arccos \frac{d_{OP}^{\ 2} + d_{PR}^{\ 2} - d_{OR}^{\ 2}}{2 \cdot d_{OP} \cdot d_{PR}}$$

A triangle satisfying the constraints on the distances between points (we consider only the x-coordinates)

$$d_{min} \leq |P_x - O_x| \leq d_{max}$$
$$d_{min} \leq |P_x - R_x| \leq d_{max}$$

and the constraint on the angle values is called an admissible triangle. The first two constraints represent the operating window; the set of points contained in it is used to define the triangles. The third constraint is used to discard angles that are too flat. The sharpest opening angle of the admissible triangles is then assigned to P:

$$\alpha \leq \alpha_{max}$$
$$\alpha(P) = \min_{\alpha}\left\{\alpha = O\hat{P}R\right\}$$

If a point has no admissible triangles, the point is rejected assigning it an angle default value of p. In the second pass, those points in the set of the candidate high curvature points that are sharper than their neighbors (within a certain distance) are classified as high curvature points. A candidate point P is discarded if it has a sharper valid neighbor N, that is if:

$$\alpha(P) > \alpha(N)$$

A point *N* is defined to be a neighbor of *P* if the following constraint is valid:

$$|P_x - N_x| \leq d_{max}$$

In our implementation we have defined the minimum points distance dmin as always equal to 1; consequently, the only two parameters that influence the results of the algorithm are dmax and amax. The most important parameter is amax which controls the set of admissible angles: a high value of amax will result in more points included in the set of candidate high curvature points, while a lower value indicates that only very sharp angles must be considered. This is the same as considering worthy of attention only slopes corresponding to sharp changes in the curve of the cumulative dHWD frame differences as shown in figure 3.3.

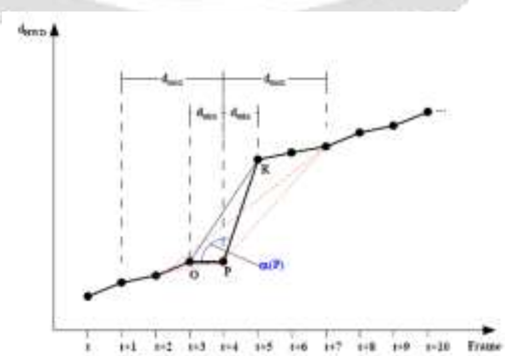

Fig. 3.3. Detection of the high curvature points algorithm.

Once the high curvature points have been determined, keyframes can be extracted by taking the midpoint between two consecutive high curvature points. Fig. 3.4 is an example of how the algorithm works. The top image shows an example shot: the algorithm detects a high curvature point within the curve of cumulative frame differences. The first and last frames of the shot are implicitly assumed to correspond to high curvature points. The frames corresponding to the midpoints between each pair of consecutive high curvature points are selected as keyframes (Fig. 3.4 center, triangles represent the high curvature points detected while the circles represent the keyframes selected). If a shot does not present a dynamic behavior, i.e. the frames within the shot are highly correlated; the curve does not show evident curvature points, signifying that the shot can be summarized by a single representative frame. Fig. 3.4 bottom shows the keyframes extracted from the example shot. The summary contains the relevant elements of the frame sequence in terms of low-level features. Unlike some methods, such as those that extract, for example, keyframes based on the length of the shots, our algorithm does not have to process the whole video. Another advantage is that it can extract the keyframes on the fly: to detect a high curvature point we can limit our analysis to a fixed number of frame differences within a predefined window. Consequently, the curvature points can be determined while computing the frame differences, and the keyframes are extracted as soon as a second high curvature point has been detected. The high curvature points analysis for keyframes extraction is similar to the approach proposed in [35] and also used in [27] where a polygonal curve representing the evolution of the frame is iteratively simplified by removing curve points. Unlike this approach, which requires that all the curve points should be globally analyzed at each step to select the candidate point to be removed, our detection of high curvature points can be made sequentially and locally.

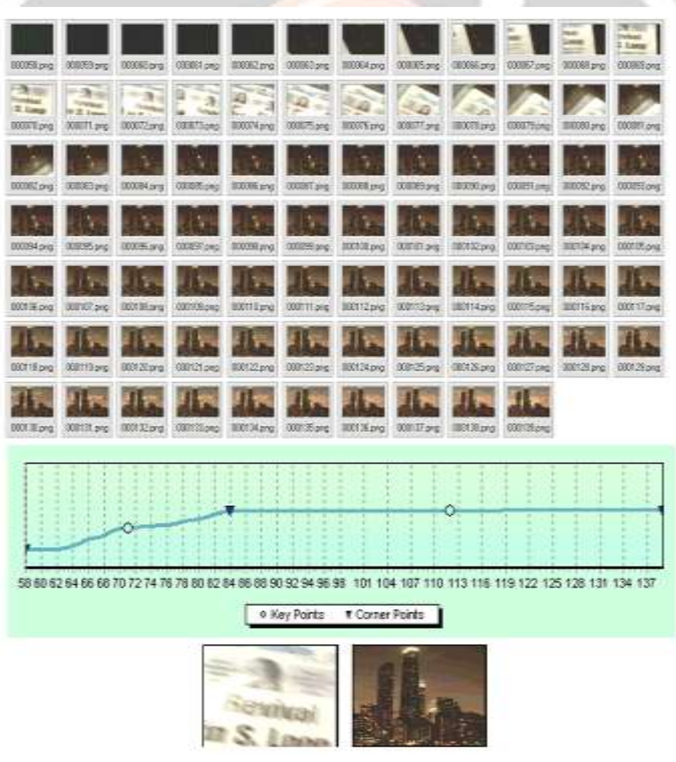

Fig. 3.4. An example of key frame selection with the proposed method. Top: the shot to analyze. Center: the corner points detected (triangles) and the key frames selected (circles). Bottom: the two key frames, number 71 and 112, extracted from the example shot are shown at the bottom.

As stated above, the corner point algorithm has two parameters: the size of the window within which the curvature angles are computed (dmax), and the maximum value of the angle considered in determining the point's curvature angle (amax). We have found experimentally that a size 3 window and angles of less than 140 degrees provide a fair tradeoff between the number of computations required and the number of keyframes extracted from each shot. It is interesting to note that, in theory, the shot segmentation phase is not strictly required. Suppose that a segmentation algorithm is not available, and thus the video is a single sequence of frames. Since cuts are abrupt changes in the visual content of the video sequence, our keyframe selection algorithm can still detect them as corner points. The keyframes extracted are the same as those extracted when the video is segmented. The keyframe extraction algorithm does not detect fading or dissolving affects them and, if their visual evolution does not present sharp changes, they will not have corner points assigned. However, the selection of keyframes within these gradual transition sequences cannot be entirely avoided. Take for example the case of the video that starts with a fade-in followed immediately by a cut, a keyframe will be selected in the set of frames corresponding to the fade sequence. Accordingly, the segmentation algorithm serves to remove these kinds of shots, improving the summarization results by ensuring the selection of informative frames only.

Step 3: Save the frames for training and testing purposes Save all the frames into the test data folder for further operation.

Step 4: Select the no of frames for training purposes (convert .bmp to .jpg & save it to training folder): Select the numbers of frames and convert it into .jpg format for training purpose

Step 5: Apply a Genetic Approach to recognize the person whose flow chart is shown in figure 3.5.

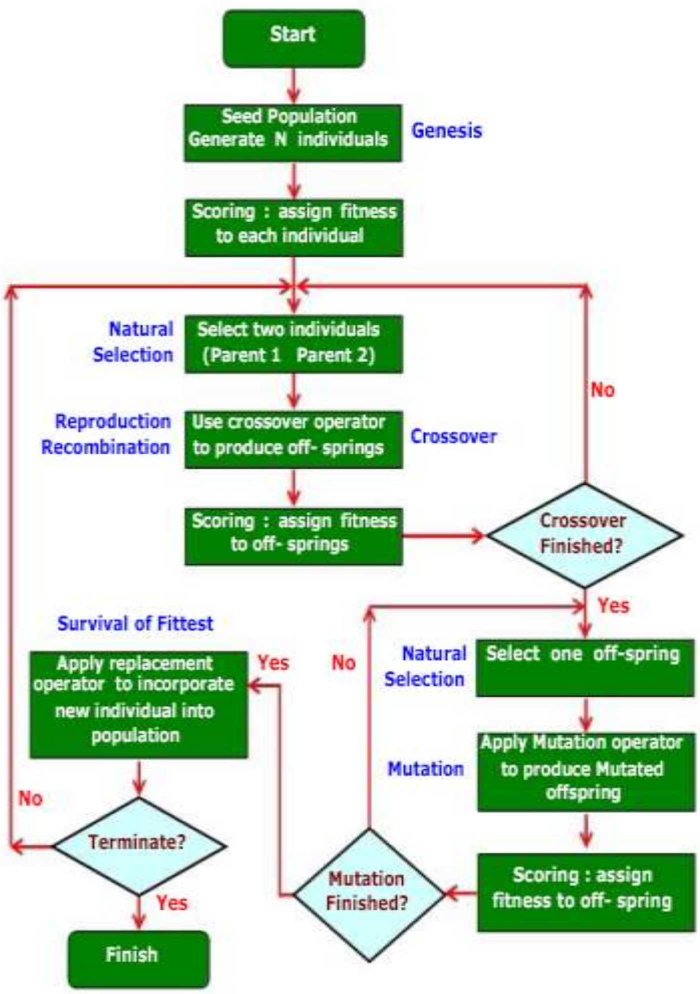

Fig: 3.5 Flow chart of Genetic Algorithm

IV EXPERIMENTAL RESULT

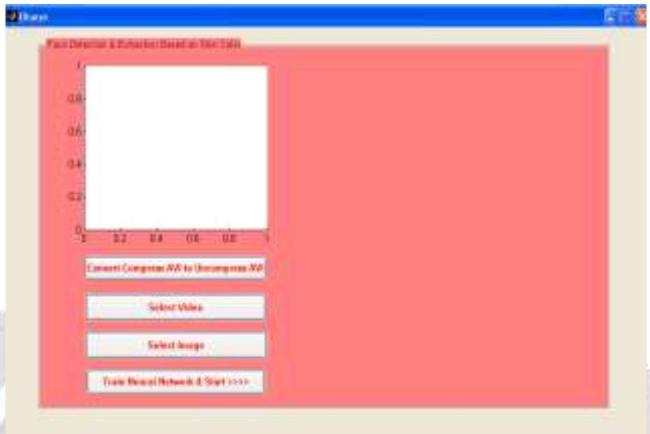

Figure 4.1 Layout of system

Figure 4.2 Input Video

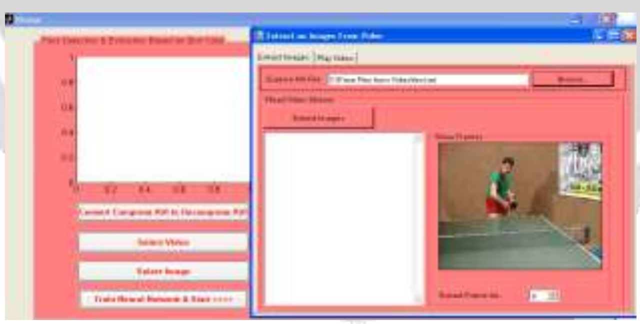

Figure 4.3 Extractions of Frames from Video

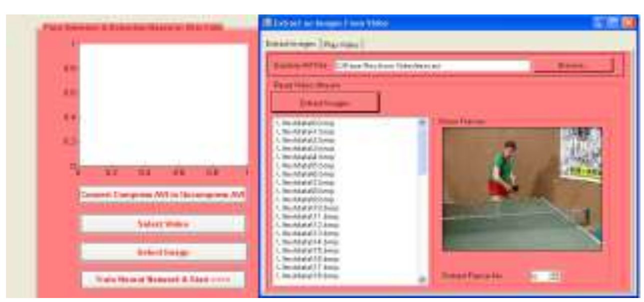

Figure 4.4 Extractions of Frames from Video

Figure 4.5 No of Frames from video

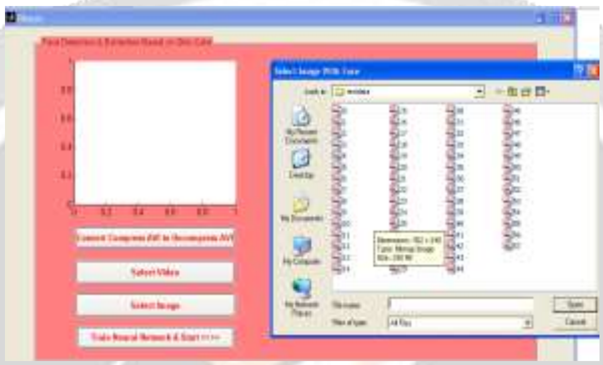

Figure 4.6 Input Image for Recognition

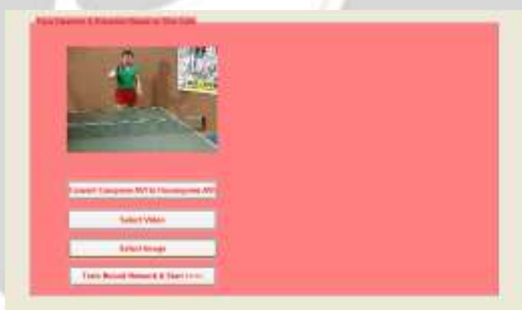

Figure 4.7 Input Image

Figure 4.8 Input Image Recognize with Test Image.

V Conclusion

We have presented methods for detecting and identifying characters in video across wide variations in pose and appearance. In this paper, we have presented an innovative algorithm for keyframe extraction. By analyzing the differences between pairs of consecutive frames of a video sequence, the algorithm determines the complexity of the sequence in terms of changes in visual content as expressed by different frame descriptors. Similarity measures are computed for each descriptor and combined to form a frame difference measure. The algorithm, which does not exhibit the complexity of existing methods based, for example, on clustering or optimization strategies, can dynamically and rapidly select a variable number of keyframes within each sequence. Another advantage is that it can extract the keyframes on the fly: keyframes can be determined while computing the frame differences as soon as a two high curvature point has been detected. The performance of this algorithm has been compared with that of other keyframe extraction algorithms based on different approaches.

REFERENCES

[1]. Richard J. Radke, Andra.S., Al- Kofahi.O., and Roysam.B**, (2005)"** Image Change Detection Algorithms: A Systematic Survey". IEEE Transactions on Image Processing, VOL. 14, NO. 3.
[2].http://www.cast.uark.edu/local/brandon thesis/Chapter_IV_change.htm last accessed on [3th of September 2008]
[3]. Burbidge.R., Buxton.B., "An Introduction to Support Vector
Machines for Data Mining", available from: http://www.martinsewell.com/dataminin g/BuBu.pdf
[4]. Peer .P., Solina.F., (1999). "An Automatic Human Face Detection Method", Project J2-8829, available from [http://lrv.fri.unilj. si/~peterp/publications/cvww99.pdf], last accessed on [1st of May 2007]
[5]. Gonzalez, Rafael C. & , Woods.R.E. (1992), Digital Image Processing /Rafael C. Gonzalez, Richard E. Woods . - Reading, Mass.; Wokingham : Addison-Wesley, (1992)
[6]. Kovac. J., Peer. P. and Solina. F., (2004) "Human Skin Colour Clustering for Face Detection" available from: [http://www.princeton.edu/
~aabdalla/ele 79/Human%20Skin%20Colour%20Cl
ustering%20for%20Face%20Detection. pdf] last accessed on [1st of September 2008]
[7]. Peer. P., Solina. F., Batagelj. B., Juvan.S., & Kovaˇc. J., (2003) "Colour-Based Face Detection in The “15 Seconds Of
Fame”
[8]. Low, B. K & Hjelm°as1, E. (2001), "Face Detection: A Survey"
[9]. Jin, R. & Hauptmann, A. G., **"**Learning to Identify Video Shots With People Based on Face Detection", July 6-9. 2003
[10]. Huang, X., Wei, G. & Petrushin, V. A., "Shot Boundary
Detection and High level Features Extraction for the TREC Video Evaluation 2003"
[11]. http://java.sun.com/products/javamedia/jmf/2.1.1 html, last accessed on [9th of September 2008]
[12]. Over, P., Ianeva, T., Kraaijz, W., & Smeaton, F. A."TRECVID 2005 – An Overview", March 27, 2006.
[13]. Kuchi, .P, Gabbur, P., Bhat, P. S., & David, S. S. "Human Face Detection and Tracking using Skin Colour Modeling and Connected Component Operators", (2002)
[14]. http://java.sun.com/products/ja va-media/jmf/, last accessed on [6th of October 2008]
[15]. Vezhnevets, V., Sazonov, V., & Andreeva, A., "A Survey on Pixel- Based Skin Colour Detection Techniques"
[16]. Gomez, G., Morales, E.F. "Automatic Feature Construction and a Simple Rule Induction Algorithm for Skin Detection"
[17]. http://en.wikipedia.org rhunen-Lo%C3%A8ve_transform last accessed on [29th of October 2008]
[18]. http://en.wikipedia.org/wiki/Ei eigenface last accessed on [20th of October 2008]
[19]. http://goldennumber.net/ last accessed on [9th of October 2008]
[20]. http://www.cl.cam.ac.uk/dtg/attarchive/facesataglance.html
last accessed on [9th of October 2008] .
[21] Haritaoglu, I., D. Harwood and L.S. Davis, “W4: A Real-Time System for Detecting and Tracking People in 2 ½ D.” in 5th European Conference on Computer Vision. 1998. Freiburg, Germany: Springer.
[22] J. Kittler, M. Hatef, R. P. W. Duin, and J. Matas. On Combining Classifiers. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 20, No. 3, March 1998.
[23] J.C. Platt, “Probabilistic Outputs for Support Vector Machines and Comparisons to Regularized Likelihood Methods,” in Advances in Large Margin Classifiers, MIT Press, Cambridge, MA, 1999.

[24] Oliver, N., B. Rosario, and A. Pentland. "A Bayesian Computer Vision System for Modeling Human Interactions." in Int'l Conf. on Vision Systems. 1999. Gran Canaria, Spain: Springer.
[25] V.N. Vapnik, The Nature of Statistical Learning Theory, Springer-Verlag, New York, 1995.
[26] W. E. L. Grimson and C. Stauffer, "Adaptive background mixture models for real-time tracking." in Proc. IEEE Conf. CVPR, vol. 1, 1999, pp 22-29.
[27] M. Soysal and A. A. Alatan. Combining MPEG-7 Based Visual Experts For Reaching Semantics. International Workshop VLBV03, Madrid, Spain, 18-19 September 2003.
[28] M. Soysal and A. A. Alatan. Combining Region-based MPEG-7 Experts for Reaching Semantics. 6th COST 276 Workshop, Thessaloniki, Greece, May 6-7, 2004.
[29] Wang J., Plataniotis K.N. and Venetsanopoulos A.N., Selecting discriminant eigenfaces for face recognition,Pattern Recognition Letters, 26(10)(2005) 1470-1482.
[30]Martinez, A.M. Recognizing imprecisely localized, partially occluded, and expression variant faces from a single sample per class. IEEE Trans. Pattern Analysis and Machine Intelligence 25(6) (2002) 748-763.
[31] Martinez A.M., Recognizing Expression Variant Faces from a Single Sample Image per Class, Proceedings of IEEE Computer Vision and Pattern Recognition (CVPR), (2003)353-358.
[32] Martinez A.M., Matching Expression Variant Faces, Vision Research, 43(9) (2003) 1047-1060.
[33]Beymer D. and Poggio T., Face Recognition from One Example View,Science, 272(5250) (1996).
[34]Vetter, T., Synthesis of novel views from a single face image. International Journal of Computer Vision, 28(2) (1998) 102-116.
[35]Niyogi, P.; Girosi, F.; Poggio, T. Incorporating prior information in machine learning by creating virtual examples, Proceedings of the IEEE, 86(11)(1998) 2196-2209.
[36] Md Hasan, K., Md.Ahsan, S., Mamun, A., Shahnewaz, S., Human Face Detection Techniques: A Comprehensive Review and Future Research Directions, https://www.mdpi.com/journal/electronics, 2021.
[37] Ismail, N.; Sabri, M.I.M. Review of existing algorithms for face detection and recognition. In Proceedings of the 8th WSEAS International Conference on Computational Intelligence, Man-Machine Systems and Cybernetics, Kuala Lumpur, Malaysia, 14–16 December 2009; pp. 30–39.
[38]Kumar, A.; Kaur, A.; Kumar, M. Face detection techniques: A review. Artif. Intell. 2019, 52, 927–948.