Profile based Video segmentation system to support E-learning

S. C. Premaratne, D. D. Karunaratna, G. N. Wikramanayake
K. P. Hewagamage and G. K. A. Dias.
University of Colombo School of Computing,
35, Reid Avenue, Colombo 7, Sri Lanka.

E-mail: saminda@webmail.cmb.ac.lk, {ddk ,gnw ,kph ,gkd }@ucsc.cmb.ac.lk

Abstract

Use of video clips for e-learning is very limited due to the high usage of band width. The ability to select and retrieve relevant video clips using semantics addresses this problem. This paper presents a Profile based Feature Identification system for multimedia database systems which is designed to support the use of video clips for e-learning. This system is capable of storing educational video clips with their semantics and retrieving required video clip segments efficiently on their semantics. The system creates profiles of presenters appearing in the video clips based on their facial features and uses these profiles to partition similar video clips into logical meaningful segments. The face recognition algorithm used by the system is based on the Principal Components Analysis (PCA) approach. However PCA algorithm has been modified to cope with the face recognition in video key frames. Several improvements have been proposed to increase the face recognition rate and the overall performance of the system.

1. Introduction

In today's rapidly changing electronic world (e-world) the key to maintain the appropriate momentum in organizations and academic environments is knowledge. Therefore, continuous, convenient and economical access to training material assumes the highest priority for the ambitious individual or organization. This requirement is met by electronic learning (e-learning). E-learning is one of the fastest growing areas of the high technology sector today and is a highly cost-effective and adaptable medium for education and training.

E-learning offers potentially universal access to content, regardless of location, and it can transform education and training from a passive consumption experience to a more flexible and learner-centric experience [3]. As a result of the increasing availability of e-learning, the market for training in workplace readiness and problem-solving is growing rapidly.

Establishing virtual universities and colleges and digital libraries, developing online courses and content are all important activities to support e-learning. They enable remote access to a vast volume of educational material anytime for e-learners, who can then spend their limited time on understanding and processing material on their own pace. A large volume of digital documents that can be used for e-learning are currently available on the internet in different forms such as text files, image files, voice clips, video clips, question databases etc.. In addition, the distance learning systems augment this volume of digital video documents on the internet every day.

Integration of heterogeneous data as content for e-learning applications is crucial, since the amount and versatility of processable information is the key to a successful system. Multimedia database systems can be used to organize and manage heterogeneous multimedia e-learning content [8]. At the same time, the large amount of visual information, carried by video documents requires efficient and effective indexing and searching tools. The development of standards for video encoding such as the XML-based MPEG-7 standard introduced by the moving pictures expert group (MPEG) coupled with the increased power of computing made that content-based manipulation of digital video information feasible [5].

Another important aspect that determines the success of a e-learning system is how efficiently the system uses the available bandwidth. One solution to this problem is to provide facilities for the user to browse and select what he actually required before delivering the material. This can be done by categorizing and clustering various types of educational materials by using ontologies and indices.

In this paper our focus is on video based educational material where presenters deliver educational content. We employ a set of tools developed by us to segment video clips semantically into shots by using low level features. Then we identify those segments where presenters appear and extract the relevant information in key face frames. These information are then encoded and compared with a database of similarly encoded images. The feature information in video frames of a face is represented as an eigenvector which is considered as a profile of a particular person [17]. These profiles are then used to construct an index over the video clips to support efficient retrieval of video shots.

Once the profiles for the presenters are created a semi-automatic semantic annotation process is used to annotate meta-data with the video shots. Majority of automatic metadata authorization procedures reported in the literature are based on the video's physical features such as color, motion, or brightness data [22, 23,]. However in our system we use profiles to annotate semantics to video clips automatically. The system also provides features to extend the metadata associated with profiles later at any time as they become available. The annotated metadata is saved in a XML database. We use XML databases for metadata because it allows both multimedia educational objects and metadata to be stored and handled uniformly by using the same techniques.

The remainder of this paper is organized as follows. The system architecture is shown in Section two. Section three reviews a number of techniques related to our work. Section four explains the technique for segmenting face regions and describes the use of PCA (Principle Component Analysis) for our work. The implementation of the system is shown in Section five and the results obtained are shown in Section six. Finally, in Section seven gives our conclusions and address the future work based on this project.

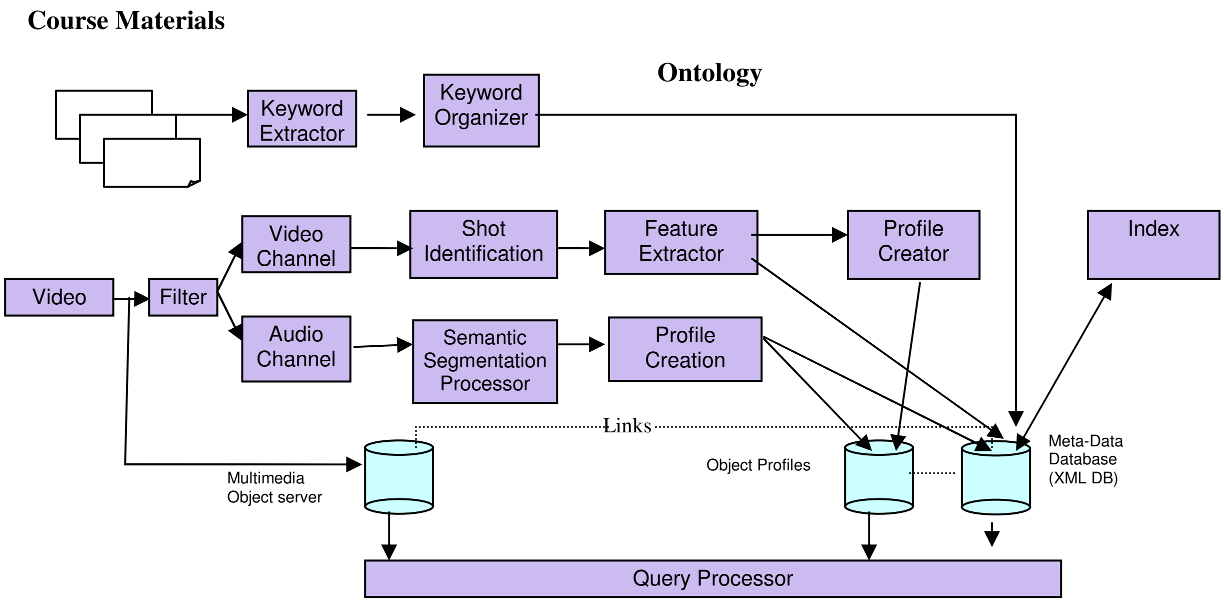

Figure 1: System Architecture

2. System Architecture

The overall architecture of our system is shown in Figure 1. The main components of our system are the keyword extractor, keyword organizer, Feature extractor, Profile creator and the query processor.

Various types of course materials such as course notes, PowerPoint presentations, quizzes, past examination papers and video clips are the main input to this system. The system stores this educational material in a multimedia object server. The keyword extractor extracts keywords from the main course materials. The keyword organizer assists the construction of an ontology in a database out of the keyword generated by the keyword extractor. The feature extractor extracts audio and video features from the video clips and the profile creator creates profiles of presenters from the information generated by the feature extractor. These profiles are then used to create indices on the video clips. Finally the query processor process enables the end users to browse and retrieve educational material stored in the object server by using the ontology and the indices.

2.1 Video Segmentation

Video segmentation can be done either manually or automatically. Manual segmentation is usually time-consuming but more accurate. Many approaches to automate segmentation of video sequences have been proposed in the past [21, 22, 23]. Earlier approaches exploited mostly the motion information in order to extract moving objects from a scene [15]. However, most of the contemporary techniques have merged motion information with information obtained from edge extraction and/or texture analysis to increase the accuracy [22, 23].

In our system a video is analyzed by segmenting it into shots, selecting key-frames, and extracting audio-visual descriptors from the shots (See Figure 2). This allows the video to be searched at the shot-level using content-based retrieval approaches.

Our approach initially uses a semi-automatic method on a training data set to construct profiles of presenters. These profiles are subsequently used to automatically assign semantics to the video shots. We have primarily investigated models that apply broadly to video content, such as presenter vs. slide show, change of presenter, change of speaker and change of lecture etc. While the models allow the video content to be annotated automatically using this small vocabulary, the integration of the different search methods together like content-based and model-based allow more effective indexing and retrieval (See Figure 1).

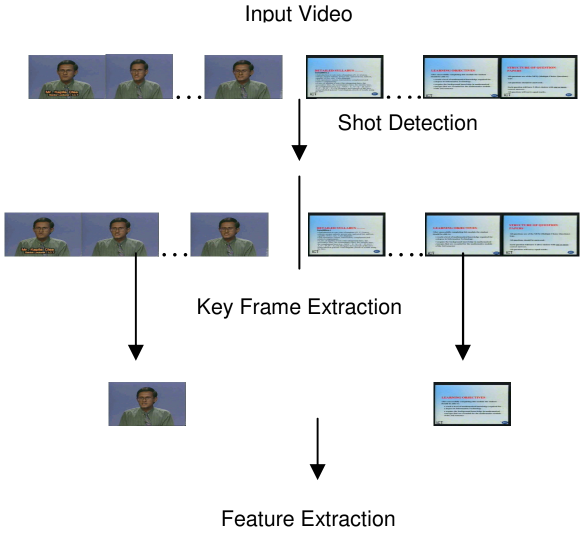

Figure 2 Segmentation of video clips

Our system extracts the following types of descriptors for each of the key-frames.

1. Color histogram
2. Edge histogram

2.2 Profile Creation.

The system initially uses a set of video clips from a video library to compute the eigenvectors of presenters [17]. An eigenvector computed for a presenter in this way can be thought of as a point in the possible eigenspace. Due to various reasons the eigenvectors compute for the same presenter by using different shots may result in multiple non equal eigenvectors. These eigenvectors can be thought of as a set of features that together characterize the variation between face images. In such cases a single eigenvector is created by correlating the individual eigenvectors created for that presenter by considering the fact that faces possess similar structure (eye, nose and mouth position, etc). One of the main reasons for using eigenfaces for our research is that it needs a lower dimensional space to describe faces.

2.3 Audio Segmentation.

In addition to automatic analysis and modeling of the features of the video content, we also investigated the use of speech indexing to combine our approach for video retrieval.

In the audio stream, initial segmentation was carried out through the use of the Bayesian Information Criterion (BIC) [16]. The technique used in this system is based on the variable window scheme proposed by Tritschler & Gopinath [16]. The Expectation Maximization algorithm was applied for the training of the Gaussian Mixture Models (GMM) for the known speakers [12]. Mel Frequency Cepstral Coefficients (MFCC) features were extracted from the "unknown" audio and tested against the GMM speaker model. The output of this procedure is a likelihood value for the speaker in the given audio stream.

2.4 Multimedia Object Server

All the multimedia objects are indexed and saved on a XML database. We are using Apache Xindice 1.0 as our multimedia object server and MPEG-7 Description Schemes schemas to store the multimedia metadata [8]. The Description Schemes (DS) provide a standardized way of describing in XML the important concepts related to audio-visual content description and content management in order to facilitate searching, indexing, filtering, and access. A relational database is used to store the profiles and catalogues.

3. Face Detection and Recognition

In the field of multimedia, the focus of research has been not just detection but also identification of faces, people or some specific objects in video images or video footages. A face recognition system can be thought of as being comprised of two stages:

1. Face Segmentation
2. Face Recognition

The first step of any face processing system is detecting the locations in images where faces are present. However, face detection from a single image is a challenging task because of variability in scale, location, orientation (up-right, rotated), and pose [1]. In general single face detection methods are classified into the following four categories:

1. Knowledge-based methods
2. Feature invariant approaches
3. Template matching methods
4. Appearance-based methods

However these methods have overlap category boundaries. The algorithms of the first category are simple. In general, algorithms of this type are used to detect faces in real time when the volume of data involved is small [4]. Most of the time, the algorithms of the second and fourth categories are implemented on expensive workstations dedicated to image processing and employee real time processing [6].

There are many approaches for face recognition ranging from the Principal Component Analysis (PCA) approach (also known as eigenfaces) [17], Elastic Graph Matching (EGM) [9], Artificial Neural Networks [10, 14], to Hidden Markov Models (HMM) [2]. All these systems differ in terms of the feature extraction procedures and/or the classification techniques used.

Michael C. Lincoln and Adrian F. Clark of the University of Essex have proposed a scheme for independent face identification in video sequences [11]. In their research an “unwrapped” texture map is constructed from a video sequence using a texture-from-motion approach. A drawback with unwrapped texture map is the recognition will be only comparable to the best front-face-only frames. Unlike this technique, eigenfaces are robust against noise and poor lighting conditions. Also eigenfaces are relatively insensitive to small variation in scale, rotation and expression.

Using neural networks for face recognition is another popular approach. Steve Lawrence has developed a face recognition system based on Self Organizing Maps (SOMs) and Convolutional Neural Networks (CNN) [10]. Their system consists of an SOM fed into a Convolutional Neural network. The problem with the SOM is that it arbitrarily divides input space into a set of classes of which the designer has no control or knowledge. Another problem with the neural networks find is a result of their inability to deal with the high dimensionality of the problem. For an example, when we consider a image of size 128 * 128 pixels requires a neural net with 16,384 input neurons for processing. Furthermore, to train such a neural network, and ensure robust performance requires an extremely large training set (much bigger than 16,384). This is often not possible in real-world applications where only one or two images of an individual may be available.

Proposed in 1991 by Turk and Pentland, this was a successful system for automatic recognition of human faces [17]. This method can be classified as appearance-based methods, which uses the whole face region as the raw input to a recognition system. The objective of an appearance-based face recognition algorithm is essentially to create low-dimensional representations of face images to perform recognition. In contrast, geometric feature-based methods attempt to distinguish between faces by comparing properties and relations between facial features, such as eyes, mouth, nose and chin. As a consequence, success of these methods depends on the feature extraction and measurement process.

4. Profile Construction Algorithm

Motivated by the work of Paul Viola and Michael Jones [18], we use a new image representation called an integral image that allows for very fast feature evaluation. We use a set of features which are reminiscent of Haar Basis functions. In order to compute these features very rapidly at many scales we used the integral image representation for key frames. The integral image is computed from an image using a few operations per pixel. Once computed, any one of these Haar-like features are computed at any scale or location very fast [6].

We use AdaBoost to construct a classifier by selecting a small number of important features [19]. Feature selection is achieved through a simple modification of the AdaBoost procedure: the weak learner is constrained so that each weak classifier returned depends on only a single feature. As a result each stage of the boosting process, which selects a new weak classifier, can be viewed as a feature selection process.

The complete face detection cascade has 32 classifiers, which total over 80,000 operations. Nevertheless the cascade structure results in extremely rapid average detection times.

Figure 3 shows some face detection samples from different video segments. Operating on 352 x 288 pixel image frames, it takes less then 1 second to detect faces. So the approach is extremely efficient and fast. After detecting the faces, the face segments are passed in to the face recognition system based on PCA.

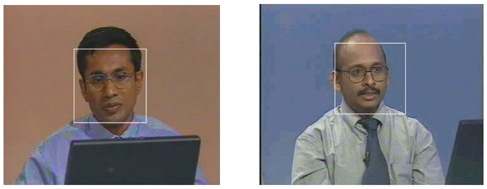

Figure3: Detected face samples

Our method of face recognition is based on profiles, which is created by using principle component analysis (PCA) [17]. Among the best possible known approaches for face recognition, Principal Component Analysis (PCA) has been object of much effort. In PCA, the recognition system is based on the representation of the face images using the so called eigenfaces. In the eigenface representation, every training image is considered a vector of pixel gray values (i.e. the training images are rearranged using row ordering).

An eigenvector of a matrix is a vector such that, if multiplied with the matrix, the result is always an integer multiple of that vector. This integer value is the corresponding eigenvalue of the eigenvector. This relationship is described by the equation below.

$$A \times u = \lambda \times u$$

Where u is an eigenvector of the matrix $A\ (n \times n)$ and λ is the corresponding eigenvalue.
Eigenvectors possess following properties:

- They can be determined only for square matrices
- There are n eigenvectors (and corresponding eigenvalues) in an $n \times n$ matrix.
- All eigenvectors are perpendicular, i.e. at right angle with each other.

The system functions by projecting face images onto a feature space that spans the significant variations among known face images. The significant features are known as "eigenfaces" because they are the eigenvectors (principal components) of the set of faces. Face images are collected into sets. Every set (or class) includes a number of images for each person, with some variations in expression and in the lighting. Some of the eigenfaces that are stored in our database is shown in Figure 4.

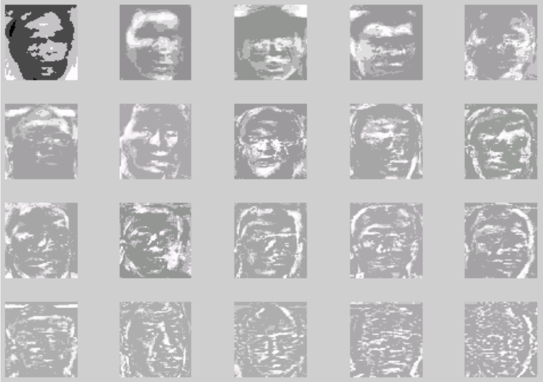

Figure 4: Eigenfaces from profile database

There is an average eigenface for each class as well and this is considered as a profile of person.

If there is M total eigenvectors, the average matrix Ψ has to be calculated and then subtracted from the original faces(Γ_i) and the result stored in the variable Φ_i :

$$\Psi = \frac{1}{M}\sum_{n=1}^{M}\Gamma_n \quad (1)$$

$$\Phi_i = \Gamma_i - \Psi \quad (2)$$

Then the covariance matrix C is calculated according to,

$$C = \frac{1}{M}\sum_{n=1}^{M}\Phi_n\Phi_n^T \quad (3)$$

Then the eigenvectors (eigenfaces) and the corresponding eigenvalues are calculated. The eigenvectors (eigenfaces) are normalized so that they are unit vectors of length 1. From M eigenvectors (eigenfaces), only M' are chosen, which have the highest eigenvalues. The higher the eigenvalue, the more characteristic features of a face does the particular eigenvector describe. Eigenfaces with low eigenvalues are omitted, as they explain only a small part of characteristic features of the faces [17]. After M' eigenfaces are determined, the "training" phase of the algorithm is finished.

There is a problem with the algorithm described in equation 3. The covariance matrix C has a dimensionality of $N^2 \times N^2$, so we would have N^2 eigenfaces and eigenvalues. For a 128 × 128 image that means that one must compute a 16,384 × 16,384 matrix and calculate 16,384 eigenfaces. Computationally, this is not very efficient as most of those eigenfaces are not useful for our task.

$$C = \frac{1}{M}\sum_{n=1}^{M}\Phi_n\Phi_n^T = AA^T \qquad (4)$$

$$L = A^T A \quad L_{n,m} = \Phi_m^T \Phi_n \qquad (5)$$

$$u_l = \sum_{k=1}^{M} v_{lk}\Phi_k \quad l = 1,\ldots,M \qquad (6)$$

Where ***L*** is a ***M × M*** matrix, ***v*** are ***M*** eigenvectors of ***L*** and ***u*** are eigenfaces.

The covariance matrix *C* is calculated using the formula *C* = AA^T. The advantage of this method is that one has to evaluate only M numbers and not N^2. Usually, ***M*** << N^2 as only a few principal components (eigenfaces) is be relevant. The amount of calculations to be performed is reduced from the number of pixels ($N^2 \times N^2$) to the number of images in the training set (***M***). We use only a subset of ***M*** eigenfaces, the ***M'*** eigenfaces with the largest eigenvalues.

The process of classification of a new (unknown) face Γ_{new} to one of the classes (known faces) proceeds in two steps. First, the new image is transformed into its eigenface components. The resulting weights ***w*** form the weight vector Ω_{new}^T:

$$\omega_k = u_k^T(\Gamma_{\text{new}} - \Psi) \quad k = 1\ldots M' \qquad (7)$$

$$\Omega_{\text{new}}^T = [\ \omega_1 \quad \omega_2 \quad \ldots \quad \omega_{M'}\ ] \qquad (8)$$

The Euclidean distance between two weight vectors $d(\Omega_i, \Omega_j)$ provides a measure of similarity between the corresponding images ***i*** and ***j***. If the Euclidean distance between Γ_{new} and other faces exceeds on average some threshold value ***θ***, we assume that Γ_{new} is no face at all. $d(\Omega_i, \Omega_j)$also allows one to construct "clusters" of faces such that similar faces are assigned to one cluster.

Let an arbitrary instance ***x*** be described by the feature vector

$$x = [a_1(x), a_2(x), \ldots, a_n(x)] \qquad (9)$$

Where $a_r(x)$ denotes the value of the ***r*** th attribute of instance ***x***. Then the distance between two instances x_i and x_j is defined to be $d(x_i, x_j)$:

$$d(x_i, x_j) = \sqrt{\sum_{r=1}^{n}(a_r(x_i) - a_r(x_j))^2} \qquad (10)$$

Each of the most frontal faces is normalized into a 128 x 128 image using the eye positions, and then converted to a point in the 16-dimensional eigenspace.

5. Implementation

Figure 5 shows the structure of an educational video clip segments in which several presenters appearing. As shown in the diagram, face features and voice features are extracted from the video clips by analyzing the audio and video channels separately. The system also employs video-caption recognition to obtain face-voice-name association if captions are available on the video clips, otherwise the user is expected to enter this meta-data manually. In many cases, a video caption is attached to a face and usually represents a presenter's name. So video-caption recognition provides rich information for face-voice-name association.

Given the extracted faces voices and names, the indexing system combines the corresponding data together and creates the required indices to support information retrieval. Finally the query processor responds to different types of user queries by using these indices and the multimedia object server as shown.

6. Evaluation

The Techniques that we have explained in this paper have been evaluated by developing a prototype system. A collection of video clips already used to deliver educational content to one of our external degree program over the TV is used as the input to our system. From this collection we first created a medium size database with profiles of 10 people. For each person we have chosen 10 face video frames with different imaging conditions. After the construction of this initial profile database, a random sample of 65 key frames were selected from our video collection and tested with our system. A small number presented poor imaging conditions which our algorithms were not designed to accommodate. These conditions included very dark lighting different camera angles and head orientation more that 30 degrees.

Our system achieves a recognition rate of 92% when we tested on 10 face classes (see Figure 6) and it dropped to 70% when we added another 10 face classes to our database. Recognition results of up to 80.5% were obtained for 20 face classes that contain straight looking faces.

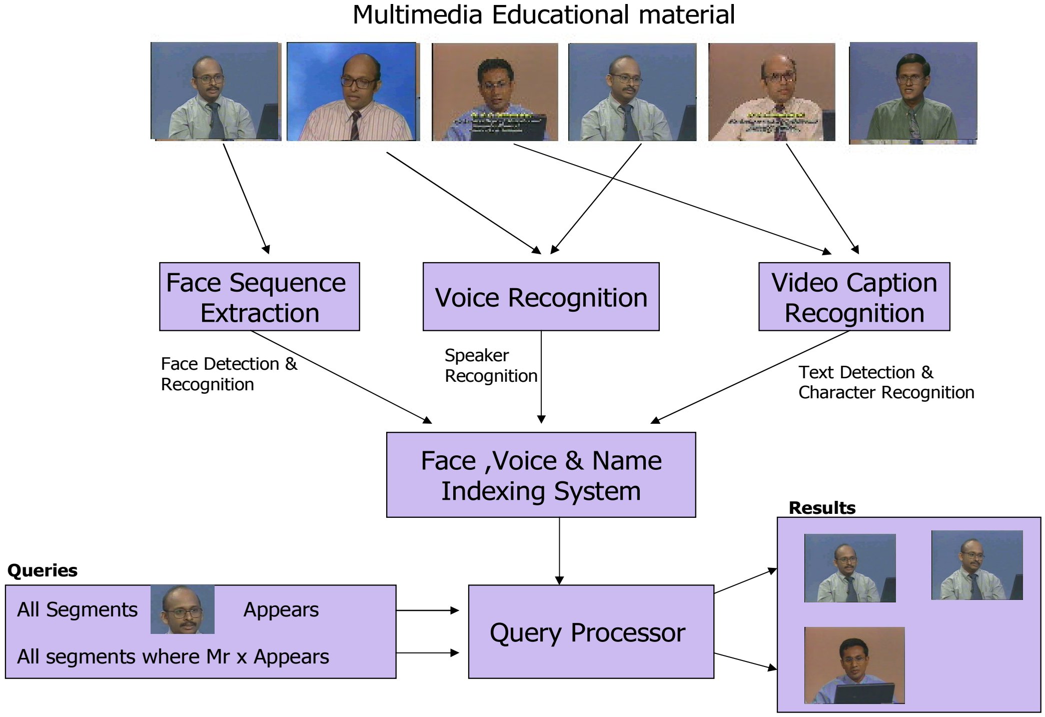
Multimedia Educational material
Face Sequence Extraction
Voice Recognition
Video Caption Recognition
Face Detection & Recognition
Speaker Recognition
Text Detection & Character Recognition
Face ,Voice & Name Indexing System
Results
Queries
All Segments
Appears
All segments where Mr x Appears
Query Processor

Figure 5: Implementation

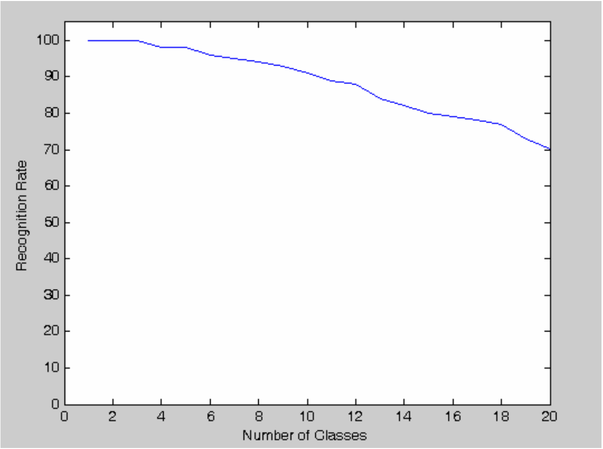

Figure 6: Results for 20 face classes

7. Conclusion and Future Work

Recognition of faces from a video sequence is still one of the most challenging problems in face recognition because video is of low quality and the frame images are small. We have proposed a simple and efficient technique to detect and recognize human faces in a video sequence but are two major challenges: the illumination and pose problems. Pose discrimination is not difficult but accurate pose estimation is hard to acquire.

We tested the performance of our implementation by varying the number of face classes for different number of eigenfaces. We observe that the algorithm is sensitive to the number of face classes. The recognition rate decreases when we increase the number of classes, because in eigenspace some face classes can overlap for some faces that have similar facial features.

In order to increase the recognition rate, methods that decrease the number of classes should be explored. One of these methods can be constructing a hierarchical tree structure. If we consider the top level nodes as main face classes, each node must have a small number of child nodes which contains sub classes with attributes of facial features extracted in different poses. This method will improve the pose problem in face recognition for some extent.

Nevertheless, as far as face recognition in video sequences is concerned, much work still remains to be done.

8. Acknowledgement.

This work is supported by the Japan International Cooperation Agency (JICA) and Asian Development Bank (ADB). The authors would like to thank the all the reviewers for insightful comments. The authors also acknowledge each individual appearing in our face database.

References

[1] Adini, Y., Moses, Y., and Ullman, S. (1993). Face Recognition: the Problem of compensating for Changes in Illumination Direction. *Technical Report CS93-21, Mathematics & Computer Science, Weizmann Institute of Science.*

[2] Bicego, M., Castellani, U., and Murino, V. (2003). Using Hidden Markov Models and Wavelets for face recognition. *Image Analysis and Processing, 2003.Proceedings of 12th International Conference*,52-56.

[3] E-Learning: Putting a World-Class Education at the Fingertips of all Children. (2000) *The National Educational Technology Plan, U.S Department of Education, December.*

[4] Fr¨oba, B., Ernst, A., and K¨ublbeck, C. (1998). Real-Time Face Detection. *Department of Applied Electronics, Fraunhofer Institute for Integrated Circuits, Germany.*

[5] ISO/IEC JTC1/SC29/WG11 Coding of Moving Pictures and Audio. *International Organization for Standardization.* http://zola.byu.edu/vad/byu2.pdf.

[6] Kawato, S., and Ohya, J. (2000). Two-step approach for real-time eye tracking with a new filtering technique. *International Conference on Systems, Man and Cybernetics.*

[7] Kobla, V., Doermann, D., Lin, K., and Faloutsos, C. (1997). Compressed domain video indexing techniques using DCT and motion vector information in MPEG video. *Storage and Retrieval for Image and Video Databases (SPIE)*, 200-211.

[8] Kosch, H. (2000). MPEG-7 and Multimedia Database Systems. *SIGMOD Records, ACM Press*, 34–39.

[9] Lades, M., Vorbriiuggen, J. C., Buhmann, J., Lange, J., Malsburg, C., Wiiurtz, R. P., and Konen, W. (1993). Distortion Invariant Object Recognition in the Dynamic Link Architecture. *IEEE Transactions on Computers*, 300-311.

[10] Lawrence, S., Giles, L. C., Tsoi, A. C. and Back, A. D. (1997). Face Recognition: A Convolutional Neural Network Approach. *IEEE Transactions on Neural Networks,* 98 -113.

[11] Lincoln, M. C., and Clark, A. F. (2001). Pose-Independent Face Identification from Video Sequences. *VASE Laboratory, University of Essex Colchester CO4 3SQ, UK, AVBPA,* 14-19

[12] Liu, M., Chang, E., and Dai, B. (2002). Hierarchical Gaussian Mixture Model for Speaker Verification. *Proceedings International Conference on Spoken Language Processing.*

[13] Lorente, L., and Torres, L. (1998). Face Recognition of Video Sequences in a MPEG-7 Context Using a Global Eigan Approach. *International Workshop on Very Low Bit-rate Video Coding, Urbana, Illinois.*

[14] Palanivel, S., Venkatesh B. S., and Yegnanarayana, B. (2003). Real Time Face Recognition System Using Autoassociative Neural Network Models. *IEEE International Conference on Acoustics, Speech and Signal Processing*, Hong Kong, 833-836.

[15] Park, S., Park, J., and Aggarwal, J. K. (2003). Video Retrieval of Human Interaction Using Model-Based motion Tracking and Multi layer Finite State Automata. *Image and Video Retrieval, Second International Conference, CIVR 2003, Urbana-Champaign, IL, USA,* 394-403.

[16] Tritschler, A., Gopinath, R. A. (1999) Improved Speaker Segmentation and Segments Clustering Using the Bayesian Information Criterion. Sixth European Conference on Speech Communication and Technology.

[17] Turk, M., and Pentland, A. (1991). Eigenfaces for Recognition. *Journal of Cognitive Neuroscience,* 71-86.

[18] Viola, P., and Jones, M. (2001). Fast and Robust Classification using Asymmetric AdaBoost and a Detector Cascade. *Neural Information Processing Systems.*

[19] Viola. P., and Jones, M. (2001). Robust Real-time Object Detection. *Second International Workshop on Statistical and computational theories of vision Canada*.

[20] Wang H. and Chang, S. (1996). Automatic face region detection in MPEG video sequences. *Conference Electronic Imaging and Multimedia Systems, part of SPIE's Photonics, China.*

[21] Yeo, B., Liu, B. (1995). Rapid scene analysis on compressed video. IEEE Transactions on Circuits & Systems for Video Technology, 533-44.

[22] Zabih, R., Miller J., and Mai, K. (1995). A feature-based algorithm for detecting and classifying scene breaks. Proc. ACM Multimedia, 189-200.

[23] Zhang, H., Kankanhalli A., and Smoliar, W. (1993). *Automatic partitioning of full-motion video", Multimedia Systems*, 10-28.