Real Time Arabic Sign Language to Arabic Text & Sound Translation System

A. E. El-Alfi
Dep. of Computer Science
Mansoura University
Mansoura, Egypt

A. F. El-Gamal
Dep. of Computer Science
Mansoura University
Mansoura, Egypt

R. A. El-Adly
Dep. of Computer Science
Mansoura University
Mansoura, Egypt

***Abstract*- Sign Language is a well-structured code gesture, where every gesture has a specific meaning. Sign Language is the only mean of communication for deaf and dumb people. With the advancement of science and technology many techniques have been developed not only to minimize the problems of deaf people but also to implement in different fields.**

This paper presents a real time Arabic sign language to Arabic text translation system, acts as a translator between deaf and dumb with normal people to enhance their communication.

The proposed system has three phases: video processing, Pattern construction and discrimination, finally text and sound transformation. The system depends on building a dictionary for Arabic sign language gestures from two resources: standard Arabic Sign Language dictionary and gestures from different domain human experts.

Keywords - *Arabic Sign Language, Video Processing, Key Frames, Weighted Euclidian Distance.*

I. INTRODUCTION

Hearing impairment or deaf people cannot talk like normal people; so they have to depend on some sort of visual communication in most of the time. Dumb people are usually deprived of normal communication with other people in the society [1].

The communication among deaf, dumb and normal people depends only on the sign language, while the majority of normal people don't know this language. Sign language is not universal; it varies according to the country, or even according to the regions, a sign language usually provides sign for whole words and it can also provide sign for letters [2].

Arabic sign language (ArSL) has recently been recognized and documented. Many efforts have been made to establish the sign language used in Arabic countries. Jordan, Egypt, the Gulf States and Kingdom of Saudi Arabia (KSA), are trying to standardize the sign language and spread it among members of the deaf community and those concerned. Such efforts produced different sign languages each of which concerns with the specific country. However, Arabic speaking countries deal with the same sign alphabets [3, 4].

The gestures used in Arabic Sign Language Alphabets are depicted in figure 1.

Figure1: Arabic alphabet sign language

In general, we need to support the communications between the deaf, dumb and normal people and make this communication take place between the deaf and dumb communities with the general public possible [5].

Developments in both technological and video processing techniques help in providing systems suit the abilities of deaf, dumb people through the use of computers. This research presents one of these systems.

The proposed system aims to develop a real time translation system from ArSL to Arabic text and sound. The main steps required can be stated as follows; building two relational data bases (gestures and description data base, and the conjugate sound data base), video processing to extract the key frames, extracting the key words and finally displaying the sentence with the audio playing.

The paper is organized as follows: section2 presents proposed system framework; section3 illustrate proposed system description; experimental results are shown in section 4; and finally section 5 presents conclusion and future work.

II. PROPOSED SYSTEM FRAMEWORK

The proposed Real Time Arabic Sign Language Translation System (RTASLTS) consists of three main steps, as shown in figure 2.

- Video processing.
- Pattern construction and discrimination.

- Text and audio transformation.

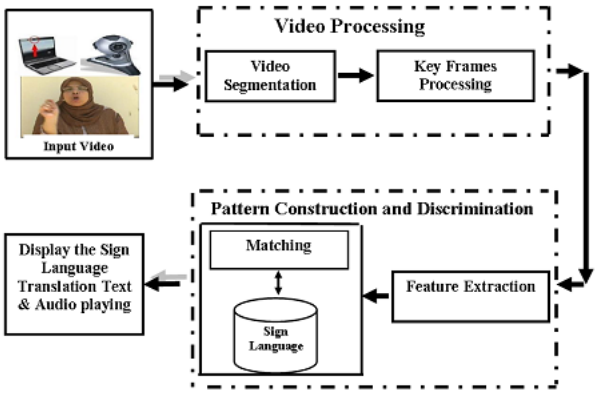

Figure.2 Proposed System Frame Work

The system has sign language data base containing 700 gestures, from five different persons 5 gestures per every one, to build different hand gestures. Also database contains the description for every alphabet.

The system goes throw many steps which are illustrated in the following flow chart, the next sections explain in details every step.

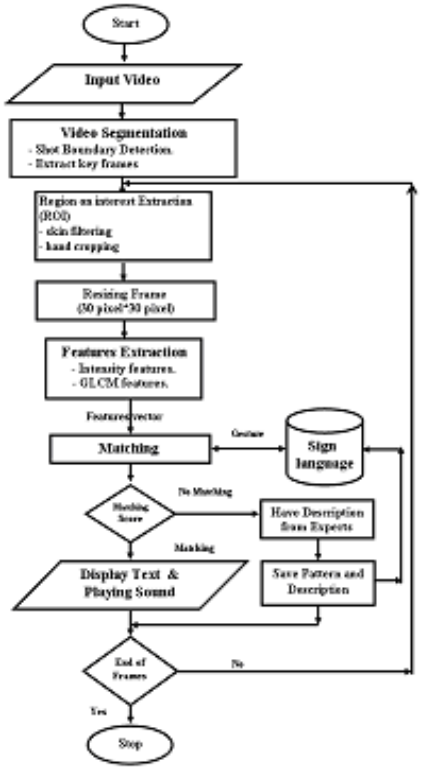

Figure3. proposed system flow chart

A. *Video Processing*

To extract valid information from video, without any loss of information, much attention is being paid to video processing technology.

As shown in the previous flow chart, the input video is segmented into frames. Figure 4 shows sample of input video. The next section illustrates this process.

Figure4. Input Video

1) *Video Segmentation*

Video segmentation is the most important phase in the proposed system; Key frame is very useful technique in this aspect. Extracting a small number of frames, that can abstract the content of video. Consequently, technologies for video segmentation and key-frame extraction have become crucial for the development of advanced digital video systems [6]. Video segmentation includes two main steps, namely shot boundary detection and key frames extraction, as shown in the following figure.

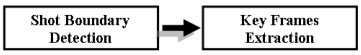

Figure 5. Video Segmentation Main Steps

These main two steps are carried out through several sub-steps which will be illustrated in the following sections.

a) *Shot Boundary Detection*

Shot Boundary Detection is an early step for most video applications which involve the understanding, indexing, characterization, and categorization of the video, as well as temporal video segmentation. The algorithm for Shot Boundary Detection is shown as follows [7, 8, 9];

Let F(k) be the k^{th} frame in video sequence, k = 1,2,.., F_v (F_v denotes the total number of video frames). The algorithm of shot boundary detection is described as follows:

Step 1: Partitioning a frame into blocks with m rows and n columns, and B(i, j,k) stands for the block at(i, j) in the k^{th} frame.

Step 2: Computing x^2 histogram matching difference between the corresponding blocks between consecutive frames in video sequence. H(i, j,k) and H(i, j, k +1) stand for the histogram of blocks at (i, j) in the k^{th} and $(k+1)^{th}$ frame respectively. Block's difference is measured by the following equation:

$$D_B(k,k+1,i,j) = \sum_{i=0}^{L-1}[(H(i,j,k) - H(i,j,k+1)]^2 / H(i,j,k). \quad (1)$$

Where, L is the number of gray level in an image.

Step 3: Computing the x^2 histogram difference between two consecutive frames:

$$D(k,k+1)=\sum_{i=1}^{m}\sum_{j=1}^{n} w_{ij}D_B(k,k+1+i,j) \quad (2)$$

Where, w_{ij} stands for the weight of block at (i, j).
Step 4: Computing Threshold Automatically: Computing the Mean and standard variance of x^2 histogram difference over the whole video sequence. Mean and standard variance are defined as follows:

$$MD=\sum_{k=1}^{F_{v-1}} D(k,k+1)/F_{v-1} \quad (3)$$

$$STD=\sqrt{\sum_{k=1}^{k_{v-1}} (D(k,k+1)-MD)^2/F_{v-1}} \quad (4)$$

Step 5: Shot boundary detection. Let threshold:
$T = MD + a \times STD.$
Where a is the constant. Say a=1.
If $D(i,i+1) \geq T$, the i_{th} frame is the end frame of previous shot.

b) Key Frame Extraction

Because of the development witnessed by multimedia information technology, the content and the expression forms of ideas have become increasingly complicated; and the way of effective organization and retrieval of the video data has become the emphasis of a large number of studies. This has also made the technology of key frame extraction the basis of video retrieval. The key frame, also known as the representation frame, represents the main content of the video, and using key frames to browse and query the video data makes the amount of processing minimal [10].
Moreover; key frames provide an organizational framework for video retrieval, and generally, key frame extraction follows the principle that quantity is more important than quality and removes redundant frames in the event that the representative features are unspecific. The following section illustrates the algorithm for extracted key frames [11]:
1) For finding a KEY frame from video, take first frame of each shot is reference frame and all other frames within shots are general frames. Compute the difference between all the general frames and reference frame in each shot with the above algorithm.
2) Searching for the maximum difference within a shot: $Max(i) = \{D(1,k)\}_{max}$, k= 2, 3…N. (5)
Where N is the total number of frames within the shot.
3) Now if the Max (i) > MD, then the frame with the maximum difference is called a key frame and otherwise with respect to the odd number of a shot's frames, the frame in the middle of shot is chose as key frame; in the case of the even number, any one frame between the two frames in the middle of shot can be chose as key frame.
Figure6 illustrates the application of the previous steps on an video sample to determine the general frames and figure 7 represents the extracted key frames.

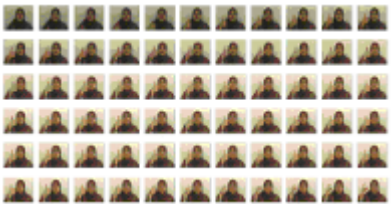

Figure 6 extracted general frames

Figure7. Extracted Key Frames

B. Pattern construction and discrimination

The pattern construction process has two steps, where the extracted key frames of the input video are processed, then their features are calculated to obtain pattern for each one. Pattern discrimination process include a comparison between each obtained pattern and a built in database patterns (which contains standard Arabic Sign Language dictionary and gestures from different domain human experts).
The next section illustrates Pattern construction and discrimination process in details.

1) Key Frames Processing

Key Frames Processing goes throw several steps as shown in figure8:

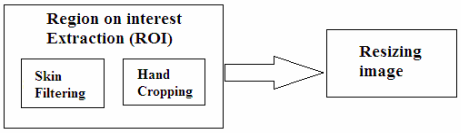

Figure 8. Key frames processing block diagram

a) Region on Interest Extraction

Each extracted key frame contains a lot of details; we need only the part which contains the hand that acts as region on interest (ROI). To extract the ROI from image, two steps are to be considered, the first is skin filtering and the second is hand cropping [12].

- Skin Filtering

The first phase of ROI step is skin filtering of the input image which extracts the skin colored pixels from the non-skin colored pixels. This method is very useful for hand detection.

Skin filtering is a process of finding regions with skin colored pixels from the background. This process has been used for detection of hand or two hands.
RGB image is converted to HSV (Hue, Saturation, and Value) color model through the following mathematical calculations [2]:

$$H = \begin{cases} 60\left(\frac{G-B}{\delta}\right) & if\ MAX = R \\ 60\left(\frac{B-R}{\delta}+2\right) & if\ MAX = G \\ 60\left(\frac{R-G}{\delta}+4\right) & if\ MAX = B \\ notdefiened & if\ MAX = 0 \end{cases} \quad (6)$$

$$s = \begin{cases} \frac{\delta}{MAX} & if\ MAX \neq 0 \\ 0 & if\ MAX = 0 \end{cases} \quad (7)$$

where $\delta = (MAX - MIN)$, $MAX = \max(R, G, B)$ and $MIN = \min(R, G, B)$.

A HSV color space based skin filter would be used on the current image frame for hand segmentation. The skin filter would be used to create a binary image with black background. To obtain this binary image, the resulting image after converted to HSV image: was filtered, smoothened and finally we obtain a gray scale image. Along with the desired hand image, other objects having skin colored was also taken into consideration which needs to get removed. This was done by taking the biggest BLOB (Binary linked object) [14]

The results obtained from performing skin filtering are given in figure9 [13].

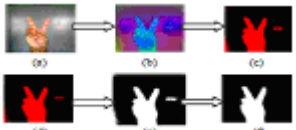

Figure9. Skin Filltering

a) RGB image, b) HSV image, c) Filtered image, d) Smoothened image, e) Binary image in grayscale, f) Biggest BLOB

- Hand Cropping

The following phase is represented in hand cropping. For recognition of different gestures, only the hand portion up to the wrist is required, and the unnecessary part is removed using the hand cropping technique. We can detect the wrist and hence eliminate the undesired region; and once the wrist is spotted, the fingers can be easily located in the wrist opposite region. The steps involved in this technique are summarized as follows [14, 15]

- The skin filtered image is scanned from all directions to determine the wrist of the hand, and then its position can be detected.

- Minimum and maximum positions of the white pixels in the image are found out in all other directions. Thus we obtain X_{min}, Y_{min}, X_{max}, Y_{max}, one of which is the wrist position.

Figure 10 represents a sample of images before and after performing hand cropping [2].

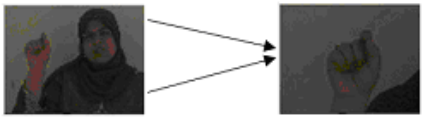

Figure10. Hand Cropping

2) *Features Extraction*

After the desired portion of the image is being cropped, the image is resized into 30×30 pixels and then feature extraction phase is carried out. Mathematical steps are applied for finding out features values in features vector.

The extracted features are: intensity histogram features and Gray Level Co-occurrence Matrix (GLCM) features.

a) Intensity Histogram Features

Histogram-based approach to texture analysis is based on the intensity value concentrations on all or part of an image represented as a histogram. The histogram of intensity levels is a simple summary of the image's statistical information and individual pixels are used to calculate the gray-level histogram. Therefore, the histogram contains the first-order statistical information about the image .Features derived from this approach include moments such as mean variance, skwness and kurtosis [15].

b) GLCM Features

Using only histograms in calculation will result in measures of texture that carry only information about distribution of intensities, but not about the relative position of pixels with respect to each other in that texture. Using a statistical approach such as co-occurrence matrix will help to provide valuable information about the relative position of the neighboring pixels in an image. The GLCM is a tabulation of how often different combinations of pixel brightness values (grey levels) occur in an image. GLCM texture considers the relation between two pixels at a time, called the reference and the neighbor pixel.

Some of the most common GLCM features are Contrast, Homogeneity, Dissimilarity, Angular Second Moment and Energy, Entropy [17].

After calculating the previous statistics, we obtain features vector contains 9 values (Mean, standard deviation, kurtosis, skewness, ASM, Energy, Correlation, Homogeneity and Contrast) for each frame.

Now matching process will be considered, where the obtained pattern containing the features vector is compared with the built in database patterns.

3) Matching

A feature vector corresponding to an image k can be denoted by:
$V^k = \{ V_1^k, V_2^k, V_3^k, \ldots\ldots, V_n^k \}$
Where, each component v_1^k typically an invariant moment function of the image. The set of all v^k_s constitute the reference library of the features' vectors. The image for which the reference vectors are computed and stored is asset

of patterns used for pattern recognition. The problem considered here is to match a features vector;
$V'=\{ V_1', V_2', V_3', \ldots\ldots, V_n' \}$
For matching, the following algorithm is applied:
Euclidean Distance Measure

$$d(v', v^k) = \sqrt{\sum_{i=1}^{n} (v_i', v_i^k)^2} \quad (8)$$

Performance of the Euclidian similarity measure function can be greatly improved if an expert knowledge about the nature of the data is available. If it is known that some values in the features vector hold more discriminatory information with respect to others, it is possible to assign proportionally higher weights to such vector components and as a result influence the final outcome of the similarity function. The formula of weighted Euclidean distance measure can be written as follows:

$$d(v', v^k) = \sqrt{\sum_{i=1}^{n} \rho_i (v_i', v_i^k)^2} \quad (9)$$

Where ρ_i denote the weight added to the component v_i to balance the variations in the dynamic range.

The value of k for which the function d is minimum is selected as the matched image index. The value of n denotes the dimension of the features vector and the N value denotes the number of images in database. The weight is given by:

$$\rho_i = N / \sum_{i=1}^{n} (v_i', v_i^k)^2 \quad (10)$$

After determining the matched pattern from system database, text and sound transformation is carried out.

C. *Text and sound transformation*

According to the built-in database which contains both a text describing a pattern and its conjugate sound, the obtained matching pattern is presented through its text and sound. Repeating this step for each input video key frame allows integration between their descriptions, and leads to formulating the text representing the translation of the input video. The descriptions obtained from the gesture sign language database for every key frame are concatenated to transform the video into a text; and synchronically, its corresponding sound is played.

III. PROPOSED SYATEM DESCRIPTION

Proposed system is implemented to translate the video Arabic Sign Language to Arabic text and sound. The system translates all signs using one hand or booth hands. The users/signers are not required to wear any gloves or to use any devices to interact with the system. The graphical user interface (GUI) for the proposed system was implemented using MATLAB 7.1. The next figures show samples of the system's screens.

Figure 11 represent the system main screen which include 6 buttons. The first one "Load video" displays the open file dialog box to load the input video from its location. The second button "Video segment" is used to apply video segmentation process. Button "Open general frames folder" allows the user to browse the folder containing total general frames. Button "Open key frames folder" allows the user to browse the folder containing extracted key frames. Button "Matching frames" for construction and discrimination of patterns, then applying matching technique and display the translation. Finally button "Maintenance" for updating the system database by adding the unrecognized gestures by domain expert.

The screen also includes video description, where information about the input video is presented. The information includes: location; type; duration in second; frames per second; size in MB and number of total video frames. Extracted features are displayed containing two choices to extract intensity histogram features or GLCM features. The system allows modification of the input video, by adding or removing frames and rebuilds the video through button "Video builder" and plays the modified video by clicking "Video preview".

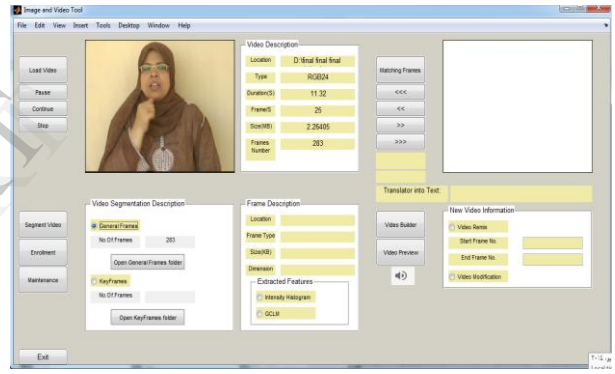

Figure11. System Main Screen

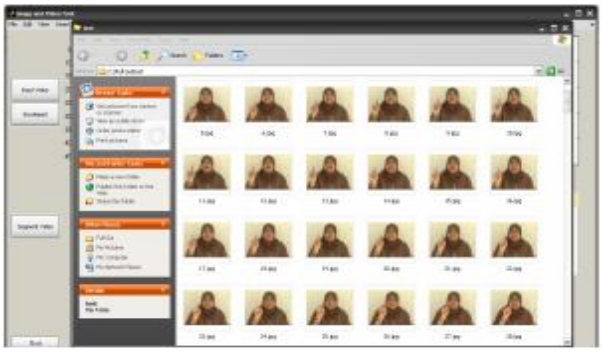

Figure12. Total General Frames

Figure13. Display Key Frames

Figure14. Total Key Frames

Figure15. Pattern Construction And Discrimination

Figure16. Detection of Unrecognized Gesture

Figure17. Final Stage of video Sign Language into Arabic Text and Sound

IV. EXPERIMENTAL RESULTS

The proposed system was applied on 10 video files of Arabic sign language, concerned with learning mathematics for first grade deaf and dumb students in primary schools. The types of these videos being AVI and the type of extracted frames being RGB, and the number of frames is 25 per second.

The evaluation of output translation was manually examined by different five experts in Arabic Sign Language. The experts' evaluations were widely used to evaluate the translation output, which with the possibility of obtaining different versions of correct translations which could only be checked by experts of the Arabic Sign Language.

For each video, the following parameters are calculated:
1. the number of correctly translated gestures and the ratio to the total gestures.

2. The number of unrecognized gestures, and the ratio to the total gestures.

Table1 illustrates the experimental data ; video duration, video size in MB, number of general frames, number of key frames, number and ratio of patterns correctly translated, and number and ratio of unrecognized patterns.

- *Performance Evaluation*

From the above-mentioned experiments, we can conclude that the designed system was able to perform a real time translation of Arabic sign language into Arabic text & sound with recognition rate of 97.4% and un recognition rate of 2.6% the number of unrecognized patterns are 16.these un matched of gestures are used to update the system database. Through addition of more unrecognized gestures by domain experts to the system database, error rate will be reduced.

Thus, the proposed system can be used on a large scale in supporting communication between deaf-dumb people and normal people.

Table 1. Experimental Data

Sign language video sample	Video Duration (in seconds)	Video size (in MB)	Total number of General frames	Total number of Key frames	Correct translated patterns	Correct translated patterns ratio	Unrecognized Patterns	Unrecognized Patterns ratio
1	11.32	2.26405	283	64	59	92.1857%	5	7.8125%
2	10.64	2.1429	266	46	45	97.82608%	1	2.173913%
3	12.56	2.51824	314	47	44	93.61702%	3	6.382979%
4	15.92	3.25682	399	81	79	97.53086%	2	2.469136%
5	18	3.65165	450	68	65	95.58823%	3	4.411765%
6	27.16	5.51396	679	85	85	100%	0	0%
7	10.92	2.18218	273	99	98	98.98989%	1	1.010101%
8	12.8	2.58603	320	45	45	100%	0	0%
9	25.2	5.13829	630	53	53	100%	0	0%
10	19.88	4.04692	497	68	67	98.529412%	1	1.470588%
Average						97.4%		2.6%

CONCLUSIONS AND FUTURE WORK

Arabic sign language (ArSL) has recently been recognized and documented. Many efforts have been made to establish sign language used in Arabic countries. This work is one of these efforts; it presents a proposed system to support the communication between deaf, dumb and normal people by translating Egyptian sign language video to its corresponding text and sound.

The proposed RTASLTS System consists of three main steps: Video processing, Pattern construction and discrimination, finally text and audio transformation.
Video processing step contains video segmentation throw shot boundary detection and key frames extraction. Pattern construction and discrimination step contains key frames processing throw region on interest extraction (skin filtering and hand cropping) and feature extraction (intensity histogram and GLCM).

RTASLTS was applied on 10 video files of Arabic sign language. We can conclude that the designed system was able to perform a real time translation of Egyptian Arabic sign language into Arabic text & sound with recognition rate of 97.4%. Through the addition of more unrecognized gestures by domain experts to the system database, error rate will be reduced.

Arabic speaking countries deal with the same sign alphabets, but not the same Arabic sign language. With more efforts we can obtain standard Arabic sign language translator, which deal with all Arabic countries not only for Egyptian sign language.

REFERENCES

[1] Ravikiran J; Kavi Mahesh, "Finger Detection for Sign Language Recognition", Proceedings of the International Multi Conference of Engineers and Computer Scientists, vol.1, 2009.
[2] Joyeeta Singha, Karen Das, " Indian Sign Language Recognition Using Eigen Value Weighted Euclidean Distance
Based Classification Technique", (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 4, No. 2,2013, available at: www.ijacsa.thesai.org .
[3] Nashwa El-Bendary, Hossam M. Zawbaa, Mahmoud S. Daoud,Aboul Ella Hassanien: "ArSLAT: Arabic Sign Language AlphabetsTranslator", International Journal of Computer Information Systems and Industrial Management Applications, Vol.3, 2011, available at: www.mirlabs.org/ijcisim/regular_papers_2011/Paper56.pdf.
[4] Catherine S. Fichten, Vittoria Ferraro, Jennison V. Asuncion and Caroline Chwojka: "Disabilities and e-Learning Problems and Solutions: An Exploratory Study", Educational Technology & Society, McGill University, Canada, 2009.
[5] SHOAIB AHMED .V: "MAGIC GLOVES (Hand Gesture Recognition and Voice Conversion System for Differentially Able Dumb People)", C. Abdul Hakeem College of Engineering and Technology,PHD, London, 2012.
[6] Shilpa.R.Jadhav, Anup.V. Kalaskar, Shruti Bhargava Department of: "Efficient Short Boundary Detection & Key Frame Extraction using Image Compression", International Journal of Electronics Communication and Computer Engineering, vol.2, 2011.
[7] Kintu Patel Oriental: "Key Frame Extraction Based on Block based Histogram Difference and Edge Matching Rate", International Journal of Scientific Engineering and Technology, Vol.1, No.1, 2011.
[8] Ganesh. I. Rathod; Dipali. A. Nikam, "An Algorithm for Shot Boundary Detection and Key Frame Extraction Using Histogram Difference", International Journal of Emerging Technology and Advanced Engineering, vol.3, 2013, available at: www.ijetae.com.
[9] Sandip T. Dhagdi, P.R. Deshmukh: "Keyframe Based Video Summarization Using Automatic Threshold & Edge Matching Rate",

International Journal of Scientific and Research Publications, vol.2, 2012, available at: www.ijsrp.org.

[10] Saurabh Thakare: "Intelligent Processing and Analysis of Image for Shot Boundary Detection", International Journal of Emerging Technology and Advanced Engineering, vol. 2, 2012, available at: ww.ijetae.com.

[11] Prajesh V. Kathiriya, Dhaval S. Pipalia, Gaurav B. Vasani, Alpesh J. Thesiya, Devendra J. Varanva:" X2 (Chi-Square) Based Shot Boundary Detection and Key Frame Extraction for Video ", International Journal Of Engineering And Science, Vol. 2, 2013.

[12] Jiong June Phu and Yong Haur Tay; "Computer Vision Based Hand Gesture Recognition Using Artificial Neural Network", Universiti Tunku Abdul Rahman (UTAR), MALAYSIA, available at: http://www.deafblind.com/asl.html.

[13] Joyeeta Singha, Karen Das: "Hand Gesture Recognition Based on Karhunen-Loeve Transform", Assam Don Bosco University, Mobile & Embedded Technology International Conference, India, 2013.

[14] Jagdish Lal Raheja, Karen Das and Ankit Chaudhary: "Fingertip Detection: A Fast Method with Natural Hand", International Journal of Embedded Systems and Computer Engineering, Vol. 3, No. 2, 2011.

[15] S.Selvarajah, S.R. Kodituwakku: "Analysis and Comparison of Texture Features for Content Based Image Retrieval", International Journal of Latest Trends in Computing, vol.2, 2011

[16] BISWAROOP GOSWAMI: "TEXTURE BASED IMAGE SEGMENTATION USING GLCM", JADAVPUR UNIVERSITY, PhD, 2013, available at: http://dspace.jdvu.ac.in/bitstream/123456789/23635/1/Acc.%20No.%20DC%20442.pdf.

[17] Ch.Kavitha, B.Prabhakara Rao, A.Govardhan : " Image Retrieval Based On Color and Texture Features of the Image Sub-blocks", International Journal of Computer Applications, vol.15, No.7, 2011.