Unsupervised Feature Based Key-frame Extraction Towards Face Recognition

Jana Selvaganesan[1] and Kannan Natarajan[2]
[1]Department of ECE, Mookambigai College of Engineering, India
[2]Department of CSE, Jayaram College of Engineering and Technology, India

Abstract: *A convenient and most effective method of querying a video database for robust face recognition is by using key-frames extracted from the image sequence. In this paper we present a clustering based approach that bypasses the need for shot detection or segmentation, to extract the key-frames from the video using the local features, for the purpose of face recognition. Local features which are insensitive to noise, displacement, scale, rotation and illumination, are extracted from arbitrary points on the images based on Speeded Up Robust Feature (SURF) algorithm. The frames are then clustered using sequential K-means algorithm. A representative frame from each cluster called the key-frame is then determined for subsequent use in video based face recognition. The proposed method has been demonstrated with experimental results obtained using Honda/UCSD (name of a standard database available for face recognition research) dataset 1.*

Keywords: *Feature extraction, frame clustering, key-frame, SURF, unsupervised learning.*

Received December 12, 2013; accepted January 27, 2015

1. Introduction

Video-based face recognition has been an active research area for over a decade as it provides more information than still based face recognition. Developing a robust face recognition system for uncontrolled environments is still a challenging task.

A convenient and most effective method of querying a video database for robust face recognition in uncontrolled environments is by using key-frames. A number of key-frame extraction algorithms are available in literature [7, 14, 17], most of which are either shot based or segment based and are applicable to structured videos namely news and sports videos. The extraction of key-frame can be obtained either using sequential techniques or using cluster based techniques [5]. In cluster based techniques, a feature space is chosen using which the frames are formed into groups called clusters. A frame which closely matches with the center frame of the cluster is chosen as the key-frame. In the process of comparison of two frames, the measure of dissimilarity between the frames depends on the proper selection of feature(s) or descriptor(s) which effectively describes the visual content of the image.

For fully automatic and real time extraction of key-frames from video, we require features which are invariant to scale, rotation, illumination, expression and occlusion. Local features serve this purpose. There are many techniques of local feature extraction given in literature [19]. Yakhu and Suvonvorn [20] retrieved different shots of a person from a movie by using local features extracted from pre-defined landmarks.

Extraction of local features from images can be made using Scale Invariant Feature Transform (SIFT) as proposed by Lowe [15]. Learning of features can be implemented using supervised or unsupervised learning technique. In supervised learning technique, a frame has to be manually assigned the cluster label which closely matches with the pose cluster. This makes the process of extraction complex and time consuming. Unsupervised learning algorithms overcome this drawback and allow representing raw input data concisely [4]. In this paper, we present an approach to automatically extract key-frames from the input video using Speeded Up Robust Features (SURF) features, with the objective of improving the performance of video based face recognition systems in uncontrolled environments.

The rest of this paper is organized as follows: In section 2, we extensively discuss the papers related to our topic of unsupervised key-frame extraction. In section 3, we present our approach of unsupervised key-frame extraction using local features. In section 4, feature extraction using SURF is discussed. In section 5, we describe the technique of automatic clustering of the frames from a face video using K means sequential clustering. In section 6, we describe the method of extracting the key-frame from each cluster. In section 7, we discuss the experimental results obtained using sample videos from Honda/UCSD dataset1 database using MATLAB implementation. Conclusions and Future work are discussed in section 8.

2. Background

In the recent past, key-frame extraction has been the basic technology for much video processing research, as it significantly reduces the amount of data to be processed. In this section we discuss some of the papers related to key-frame extraction and unsupervised clustering of video frames. Kelm *et al.* [12] proposed shot based key-frame extraction approach using minimum motion intensity concept. Dang *et al.* [8] in their approach for key-frame extraction, utilized the dissimilarity measure between two frames based on image epitome. Chatzigiorgaki and Skodras [5] in their approach implemented a sequential search algorithm to extract key-frames in Moving Picture Experts Group (MPEG) videos. Yakhu and Suvonvorn [20] utilized the values of face image covariance matrix to extract key-frames from video sequence. Patel [18] proposed a key-frame extraction approach in which the histogram difference of frames was used. Zhuangy *et al.* [23] developed an unsupervised clustering algorithm for extracting key-frames from a video using the color histogram of the frames. Hadi *et al.* [11] proposed an algorithm using k-medoid clustering technique for video summarization by extracting multiple key-frames. Leordeanu and Collins [13] developed an unsupervised learning algorithm for learning object models in video sequences, considering pairwise co-occurences of dependent object features and independent nature of features that are unrelated. Here, we also review some of the papers related to unsupervised learning of features from the images. Yang *et al.* [21] proposed an unsupervised learning approach to represent the face, using local features and categorized the faces using nearest neighbor algorithm with the objective of partitioning still images in a gallery set. Zhang *et al.* [22] proposed an unsupervised clustering approach called J-based scene clustering for analyzing scenes in sports video using Fisher Discriminant function. Cinbis *et al.* [6] introduced an unsupervised metric learning technique to identify faces in a query video. Motivated by the above works and with the knowledge that choice of the feature or descriptor for effective description of the image content is vital in key-frame extraction, we propose a method to learn the local features detected and described by SURF algorithm in an unsupervised manner for extracting the key-frames for subsequent use in face recognition in uncontrolled environment.

3. Methodology

A block diagram representing the steps involved in clustering of frames using SURF features and extraction of key-frames is shown in Figure 1. A video file is given as input and the first frame of the video sequence is considered as the reference frame. For the reference frame, the feature vector describing the invariant feature points is obtained using SURF detector and descriptor. The feature vector of the subsequent frame is then obtained. The similarity between the current frame and the reference frame is computed using the Squared Euclidean Distance (SED) measure. Based on the similarity measure, the input sequence of frames is clustered using sequential clustering algorithm.

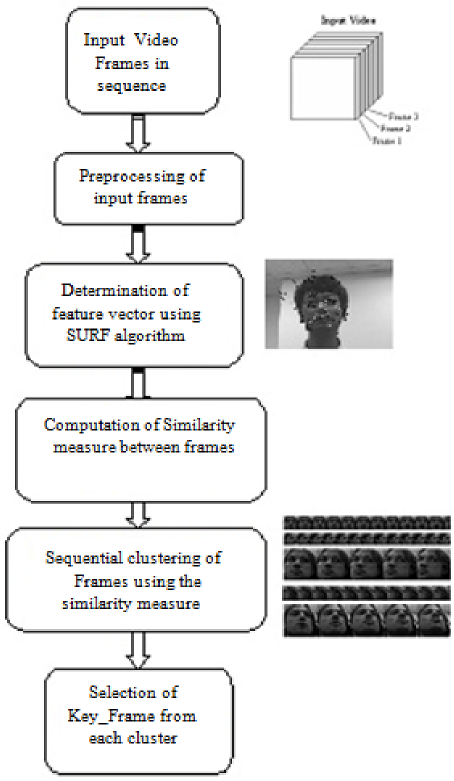

Figure 1. Block diagram of the proposed system.

From the clustered video frames a representative frame, here the first frame of the cluster, is chosen as the key-frame. The invariant features of the representative face can then be used for face recognition from uncontrolled environments in query video. As face representation plays a vital role in the automatic face recognition process, the focus of this paper is in the efficient unsupervised clustering of video frames using the similarity measure between the local invariant features extracted from the faces in the frames and the subsequent extraction of key-frames from these clusters.

4. Feature Extraction

Existing video based face recognition techniques rely on known pose, illumination and local features extracted from predefined facial landmarks [9]. Face recognition algorithms that use global features does not provide robust performance, as global features suffers from the inability to capture discriminative features which are insensitive to variations in pose, illumination, expression and partial occlusion. Local features from interest points overcome these limitations of global features and form an excellent descriptor. In literature a number of feature detectors and descriptors are available. Laplacian of Gaussian, difference of Gaussian and determinant of hessian are some of the common blob feature detectors insensitive to scale, rotation and illumination. While SURF uses determinant of hessian for detecting interest points in an image, SIFT uses difference of Gaussian for interest point detection [15]. As computation of SURF features is faster than that of SIFT features, we propose to learn the facial features using SURF algorithm. A brief overview of SURF detectors and descriptors is given in this section.

4.1. SURF Features

SURF is a robust interest point detector and descriptor which is insensitive to noise, displacement, scale, rotation and illumination. It is comparatively superior to other detectors and descriptors such as SIFT in terms of repeatability, distinctiveness, robustness and computation speed [2]. Basic steps involved in SURF implementation are selection of key-points or interest points at corners, blobs, T-junctions; representing the interest point and its neighbourhood by a feature vector referred to as descriptors; and matching the descriptor vectors of different images based on Euclidean distance.

4.2. Interest Point Detection using Integral Images

Detection of interest point is based on integer approximation to the determinant of hessian-matrix which can be computed using integral images [2]. An integral image $I_{int}(x)$ at a location $x = (x, y)^T$ is given by the sum of all pixels within the rectangular region formed by the origin and x, in the image I.

$$I_{int}(x) = \sum_{i=0}^{i \le x} \sum_{j=0}^{j \le y} I(i, j) \tag{1}$$

The blob-like structures are detected at points where the determinant of the hessian matrix is maximum. Hessian matrix is a square matrix consisting of second order partial derivatives of a function and for a point $x(x, y)$ in an image I it is given by:

$$H(x) = \begin{bmatrix} Lxx(x) & Lxy(x) \\ Lxy(x) & Lyy(x) \end{bmatrix} \tag{2}$$

Where $Lxx(x)$ is a second order partial derivative in the direction x and $Lxy(x)$ is the mixed partial second order derivative in the x and y directions. $L(x)$ are derivatives of an image smoothed by Gaussian function and it is given by $L(x)=g(\sigma_I)\otimes I(x)$, σ being the scale factor. The maximum of the determinant of hessian matrix and the maximum of the trace of hessian matrix determines the interest point. The Determinant (D) and Trace (TR) of hessian matrix is given by Equations 3 and 4 respectively.

$$D = \sigma_I^2 \, (L_{xx}L_{yy}(x) - L^2_{xy}(x)) \tag{3}$$

$$TR = \sigma_I \, (L_{xx} + L_{yy}) \tag{4}$$

Using a 9x9 box filter, which are Gaussian approximations with a scale factor of 1.2, the blob responses at point x can be computed [10]. The trace of the hessian matrix is significant during the matching process, as those point pairs with the same sign are only considered. Figure 2 shows the detected interest points in the sample image in a frame of the video input.

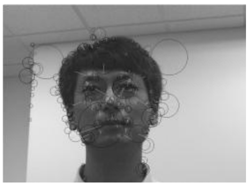

Figure 2. Sample frame from 'fuji.avi' with 178 interest points.

4.3. Interest Point Description and Matching

The description of the interest point is based on the distribution of the intensity within the rectangular neighbourhood of the detected point. Haar wavelet responses along x and y direction are computed using the Haar wavelet filters shown in Figure 3. For the extracted feature to be insensitive to image rotation, the interest points are assigned a reproducible orientation. The extracted features comprising the interest points and their location, vary from one image to another and hence are unique to an image.

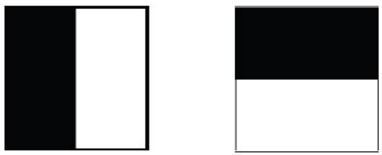

Figure 3. Haar wavelet filter along x and y direction.

To preserve the spatial information, the region is divided into 4×4 square sub-regions (as shown in Figure 4) and Haar wavelet response is computed at equally spaced sample points. The response in horizontal direction is represented as dx and that in vertical direction by dy. Haar wavelet responses dx and dy are summed up in each sub region after weighing with a Gaussian function centered at the interest point. To include the sign of intensity change, sum of the absolute values of responses |dx| and |dy| are also considered. The 4x4 subregions are represented by a 4 dimensional vector v comprising of (Σdx, Σdy, Σ|dx|, Σ|dy|) components. Concatenating this for an entire region corresponds to a 64 length descriptor.

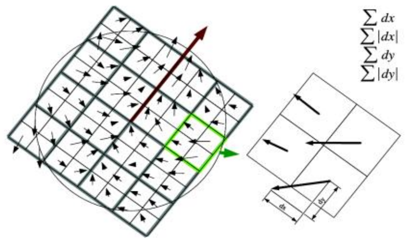

Figure 4. Subregion for Haar wavelet response computation.

5. Automatic Clustering using Surf Features

From the training video, faces are detected using Viola-Jones face detection algorithm and are then made insensitive to variations in illumination using histogram equalization [20]. The local features are then extracted by detecting the interest point using SURF features and are matched with the faces detected in the subsequent frames. To determine a stable feature, similarity measure between frames are computed for a total of NF frames and these stable features can subsequently be used to recognize faces from the test video.

5.1. Face Matching using SURF Features

Face matching can be determined by correspondingly matching the key points using SURF features. In Bicego *et al.* [3] and Luo *et al.* [16], sub region matching strategies are adapted using sub region similarities and global similarities but at the expense of computation cost. Implementing geometric constraints in the point matching method as adapted by Du *et al.* [10], the matching speed and robustness can be increased. Using this methodology interest points in two frames are compared. If the ratio of two minimum-distance point pair for each interest point in the first frame is smaller than a predefined threshold, then the point pair with minimum distance is chosen as the matched pair. A Measure of Similarity (MSim) based on the number of matched points P, average value of Squared Euclidean Distance (SED) E_{av} and the average value of the ratio of minimum distances DR_{av} of all the matched pairs is given by Equation 5:

$$MSim = \begin{cases} \dfrac{E_{av} + DR_{av}}{2} \; \forall n = 1 \ldots P .. if P \geq 10 \\ \dfrac{E_{av} + DR_{av}}{2} + 1 \forall n = 1 \ldots P .. if P < 10 \end{cases} \tag{5}$$

Where

$$E_{av} = \frac{1}{P} \sum_{n} SED \tag{6}$$

$$DR_{av} = \frac{1}{P} \sum_{n} DR \tag{7}$$

The SED between two interest point descriptor IP1and IP2 is given by Equation 8:

$$SED = sum\,((IP1 - IP2)^2) \tag{8}$$

The distance ratio of matched points is given by Equation 9.

$$DR = \frac{First\ minimum\ distance}{Second\ minimum\ distance} \tag{9}$$

Figure 5 shows an example of frame matching considering 50 matching point pairs.

Figure 5. Sample frame matching.

5.2. Sequential K-Means Frame Clustering

A number of data clustering algorithms are available in literature [1]. Owing to its simplicity, faster speed of computation and ability to scale well to large amounts of centroids, sequential K-means algorithm is used to cluster the frames based on the similarity measure. The first frame is chosen as the first cluster center and the distance of the subsequent frames to the first frame is computed using the similarity measure. If the similarity score is less than a specified threshold S, the frame is grouped in cluster-1. If the distance of the current frame is greater than S, then a new cluster is initiated with the current frame as cluster-2 center. The number of clusters and hence the number of frames in each cluster can be controlled by choosing the threshold parameter S. Clustering enables selecting representative frame for each cluster.

6. Key-Frame Extraction

The subsequent step after clustering the frames is extraction of key-frame from each cluster. Clusters which have sufficient number of frames are considered for key-frame selection. If a cluster has a minimum of P/Q frames then it is considered as key-cluster. Here P is the total number of frames to be clustered and Q is the total number of clusters. From each key-cluster the first frame of the cluster is chosen as the key-frame.

7. Experimental Results

Sample videos from Honda/UCSD dataset1 is used for experiment purpose. The algorithm was implemented and tested using MATLAB in a 1.86GHz Pentium system. The characteristics of the five videos used for testing the proposed key-frame extraction using local features are given in Table 1.

Table 1. Summary of the video inputs used for testing.

Video Name	Resolution	TNF	Frames/Sec
James.avi	640 x 480	314	15
Mushiake.avi	640 x 480	275	15
Behzad.avi	640 x 480	384	15
Fuji.avi	640 x 480	311	15
Yokoyama	640 x 480	329	15

A sample of 100 consecutive frames from several input files are used for testing the proposed clustering technique. The colour image frames are converted to monochrome image frames. Histogram equalization is done for normalizing the illumination and the frames are resized to 256x256. The interest points are detected for an upright orientation and a hessian response threshold of 0.0001. Considering a sample data ‘yokoyama.avi’, the number of interest points obtained when the hessian response threshold was varied from 0.00001 to 0.001 is shown in Figure 6.

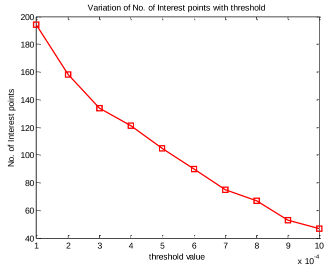

Figure 6. Variation of number of interest points with hessian response threshold for sample input ‘yokoyama.avi’.

With smaller threshold, higher number of interest points is obtained and subsequently more numbers of point pairs are matched between two frames. The similarity measure for NF frames is a NFxNF symmetric matrix. The lower the measure value, higher the similarity between the frames. The similarity measure considering 6 frames for the sample input ‘fuji.avi’ is given in Table 2.

Table 2. Similarity measure for 6 frames, frame 10 to frame 15 of the input video ‘fuji.avi’.

Frame index	**1**	**2**	**3**	**4**	**5**	**6**
1	0	0.20	0.32	0.37	0.44	0.43
2	0.18	0	0.23	0.37	0.44	0.45
3	0.30	0.26	0	0.30	0.41	0.43
4	0.37	0.38	0.29	0	0.33	0.42
5	0.43	0.45	0.38	0.30	0	0.33
6	0.45	0.47	0.42	0.42	0.32	0

The K-means sequential clustering with a similarity threshold for the k^{th} frame, chosen as 0.5 is applied to 100 frames of the input video sample. The number of clusters and the number of frames within a cluster is a variable quantity for different input data. The clustered result for sample inputs is given in Table 3. Results of key-frame extraction on different video clips under different thresholds considering 100 frames of the input video, with the criteria ‘cluster size greater than P/Q’ where P is the total number of frames and Q is the number of clusters is given in Table 4. The variation of number of clusters with the threshold for the various input videos are shown in Figure 7.

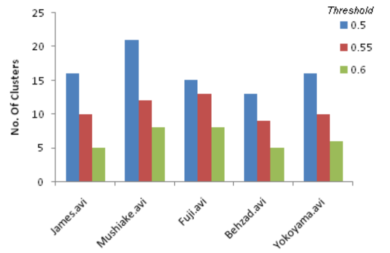

Figure 7. Variation of number of clusters under different thresholds for 100 frames of the sample video inputs.

Table 3. Results of clustering 100 frames of the input video with a similarity threshold (S) of 0.5 and 0.6 with the key-frames extracted from each cluster with the criteria ‘cluster Size greater than P/Q’. P is the total number of frames and Q is then number of clusters. TNKF (Total Number of Key Frames.) denotes the total number of key-frames.

Video Name	**Threshold**	**Key-Frames Extracted**	**TNKF**
James.avi	0.5	1, 28, 63, 69, 81	5
	0.6	1, 27, 65	3
Mushiake.avi	0.5	1, 31, 54, 59, 67, 78, 83	7
	0.6	1, 54, 75	3
Behzad.avi	0.5	1, 27, 53, 68, 81	5
	0.6	1, 52, 80	3
Fuji.avi	0.5	1, 20, 30, 48, 70, 90	6
	0.6	1,27,50.67,84	5
Yokoyama.avi	0.5	1,46, 71	3
	0.6	1,43,67	3

Table 4. Results of key-frame extraction on different video clips under different thresholds, considering 100 frames of the input video, with the criteria ‘cluster Size greater than P/Q’. P is the total number of frames and Q is the number of clusters.

Threshold	**James.avi (No. of Clusters/No. of Key-Frames)**	**Mushiake.avi (No. of Clusters/No. of Key-Frames)**	**Fuji.avi (No. of Clusters/No. of Key-Frames)**	**Behzad.avi (No. of Clusters/No. of Key-Frames)**	**Yokoyama.avi (No. of Clusters/No. of Key-Frames)**
0.5	16/5	21/7	15/6	13/5	16/3
0.55	10/4	12/4	13/8	9/3	10/4
0.6	5/3	8/3	8/5	5/3	6/3

The variation of number of key-frames extracted under different threshold for the input videos is shown in Figure 8. The clustered result for 100 frames of the input ‘james.avi’ when the threshold is 0.6 is shown in Figure 9. It can be seen from Figure 7 that the number of clusters formed is more when the threshold is less. Also, the density of clustering increases with threshold.

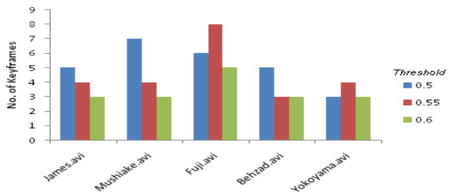

Figure 8. Variation of number of key-frames under different thresholds for 100 frames of the sample video inputs.

Figure 9. Clustered frames from the sample video input 'james.avi' considering 100 frames with a threshold of 0.6.

7.1. Comparison With Other Key-Frame Extraction Techniques

A review of key-frame extraction techniques has been done in section 2. Key-frame extraction technique is application dependent. Most of the algorithms are shot based. The proposed method bypasses the need for shot detection and utilizes SURF features for clustering the frames, considering the application of face recognition. A brief summary of some of the techniques is given in Table 5.

Table 5. Summary of some of the key-frame extraction techniques reviewed.

Reference Number	Method used	Application/characteristic
[5]	Sequential search algorithm.	Video copy detection.
[8]	Dissimilarity measure based on image epitome.	Consumer video.
[11]	K-medoid clustering.	Multiple representative frames.
[12]	Minimum motion intensity.	Web sharing videos.
[18]	Histogram difference of frames.	Shot based videos.
[20]	Covariance matrix.	Surveillance video.
Our method	Unsupervised clustering using SURF features.	Bypasses shot detection; for face recognition purpose.

8. Conclusions and Future Works

Facial biometrics play an important role in surveillance and security, as it can be obtained non-intrusively without the knowledge and cooperation of the human being involved, unlike fingerprint recognition or iris recognition. Prior to face recognition, the primary step involved is the extraction of invariant features in the spatio-temporal domain. This paper described a method of automatically clustering the frames in a video using SURF features and K-means sequential algorithm for the purpose of face recognition, bypassing the need for shot detection or segmentation. It was observed that the number of clusters formed is more when the threshold is less. From the clustered faces a representative face containing the salient features necessary for efficient face recognition is chosen. This will further be extended to include the temporal information for batch processing of frames during clustering and recognition of an identity. Key-frame extraction technique may vary depending on application. As key-frames play an important role in video processing, the proposed key-frame extraction technique provided a means of extracting the frame with significant content, for the purpose of face recognition.

References

[1] Abu-Abbas O., "Comparisons Between Data Clustering Algorithms," *The International Arab Journal of Information Technology*, vol. 5, no. 3, pp. 320-325, 2008.

[2] Bay H., Ess A., Tuytelaars T., and Gool L., "Speeded-Up Robust Features (SURF)," *Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346-359, 2008.

[3] Bicego M., Lagorio A., Grosso E., and Tistarelli M., "On the Use of SIFT Features for Face Authentication," *in Proceeding of Conference on Computer Vision and Pattern Recognition Workshop*, New York, pp. 17-22, 2006.

[4] Blum M., Springenberg J., Wulfing J., and Riedmiller M., "On the Applicability of Unsupervised Feature Learning for Object Recognition in RGB-D Data," *Workshop on Deep Learning and Unsupervised Feature Learning*, New York, pp. 1-9, 2011.

[5] Chatzigiorgaki M. and Skodras A., "Real-Time Key-Frame Extraction Towards Video Content Identification," *in Proceeding of International Conference on Digital Signal Processing*, Santorini, pp. 934-939, 2009.

[6] Cinbis R., Verbeek J., and Schmid C., "Unsupervised Metric Learning for Face Identification in TV Video," *in Proceeding of International Conference on Computer Vision*, Barcelona, pp. 1559-1566, 2011.

[7] Gianluigi C. and Raimondo S., "An Innovative Algorithm for Key-Frame Extraction in Video Summarization," *Journal of Real-Time Image Processing*, vol. 1, no. 1, pp. 69-88, 2006.

[8] Dang C., Kumar M., and Radha H., "Key-frame Extraction from Consumer Videos Using Epitome," *in Proceeding of International Conference on Image Processing, September*, Orlando, pp. 93-96, 2012.

[9] Kumar D., Rai C., and Kumar S., "An Experimental Comparison of Unsupervised Learning Techniques for Face Recognition," *International Journal of Computer and Information Science and Engineering*, vol. 1, no. 3, pp. 158-166, 2007.

[10] Du G., Su F., and Cai A., "Face Recognition Using SURF Features," *in Proceeding of Pattern Recognition and Computer Vision*, 2009.

[11] Hadi Y., Essannouni F., Oulad R., and Thami R., "Unsupervised Clustering by K-Medoids for Video Summarization," *in Proceeding of International Symposium on Communications, Control and Signal Processing*, Marrakech, pp. 1-4, 2006.

[12] Kelm P., Schmiedeke S., and Sikora T., "Feature-based Video Key-frame Extraction for Low Quality Video Sequences," *in Proceeding of 10th Workshop on Image Analysis for Multimedia Interactive Services*, England, pp. 25-28, 2009.

[13] Leordeanu M. and Collins R., "Unsupervised Learning of Object Features from Video Sequences," *in Proceeding of IEEE International Conference on Computer Vision and Pattern Recognition*, San Diego, pp. 1142-1149, 2005.

[14] Liu G. and Zhao J., "Key-frame Extraction from MPEG Video Stream," *in Proceeding of the Second Symposium International Computer Science and Computational Technology*, Huangshan, pp. 7-11, 2009.

[15] Lowe D., "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91-110, 2004.

[16] Luo J., Ma Y., Takikawa E., Lao S., Kawade M., and Lu B., "Person-Specific SIFT Features For Face Recognition," *in Proceeding of International Conference on Acoustic, Speech and Signal Processing*, Hawaii, pp. 593-596, 2007.

[17] Nasrollahi K. and Moeslund T., "Finding and Improving the Key-frames of Long Video Sequences for Face Recognition," *in Proceeding of IEEE 4th International Conference on Biometrics Theory, Applications and Systems*, Washington, pp. 1-6, 2010.

[18] Patel K., "Key-frame Extraction Based on Block based Histogram Difference and Edge Matching Rate," *International Journal of Scientific Engineering and Technology*, vol. 1, no. 1, pp. 23-30, 2011.

[19] Qian J. and Yang J., "A Novel Feature Extraction Method For Face Recognition Under Different Lighting Conditions," *in Proceeding of the 6th Chinese Conference on Biometric Recognition*, China, pp. 17-24, 2011.

[20] Yakhu S. and Suvonvorn N., "Key-frame Extraction from Video Sequence Using a Face Quality Index Applied to Surveillance System," *in Proceeding of ECTI Conference on Application Research and Development*, Thailand, pp. 409-413, 2009.

[21] Yang S., Bebis G., Hussain M., Muhammad G., and Mirza A., "Unsupervised Discovery of Visual Face Categories," *International Journal on Artificial Intelligence Tools*, vol. 22, no. 1, pp. 1-27, 2013.

[22] Zhang W., Ye Q., Xing L., Huang Q. and Gao W., "Unsupervised Sports Video Scene Clustering And Its Applications To Story Units Detection," *in Proceeding of Visual Communications and Image Processing*, Beijing, pp. 446-455 2005.

[23] Zhuangy Y., Rui Y., Thomas S. and Mehrotra S., "Adaptive Key-frame Extraction Using Unsupervised Clustering," *in Proceeding of International Conference on Image Processing*, Chicago, pp. 866-870, 1998.

Jana Selvaganesan obtained her BE degree in Electronics and Communication Engineering from Bharathidhasan University, ME in Communication Systems from Anna University Chennai and is pursuing PhD at Anna University of Technology Trichy. Her research interests include Signal Processing and Computer Vision.

Natarajan Kannan obtained his BE degree in Computer Science and Engineering from Bharathiar University, Coimbatore, ME in Computer Science and Engineering from Annamalai University, Chidambaram and Ph.D. in Computer Science and Engineering from Bharathidasan University, Trichy. He has rich teaching experience and has published numerous papers in national and international journals and international conferences. Currently he is the Principal of Jayaram College of Engineering and Technology, Trichy. His research interests include Computer Vision, Networks and Software Testing.