

PERFORMANCE ANALYSIS OF FINGERPRINTING EXTRACTION ALGORITHM IN VIDEO COPY DETECTION SYSTEM

Ms.R.Gnana Rubini¹, Prof.P.Tamije Selvy², Ms.P.Anantha Prabha³

¹PG Student, ²Assistant Professor(SG), ³Assistant Professor
^{1,2,3}Department of Computer Science and Engineering, Sri Krishna College of
Technology, Coimbatore, India

¹gnanrajrad@gmail.com, ²tamijeselvey@gmail.com, ³ap.prabha@gmail.com

ABSTRACT

A video fingerprint is a recognizer that is derived from a piece of video content. The video fingerprinting methods obtain unique features of a video that differentiates one video clip from another. It aims to identify whether a query video segment is a copy of video from the video database or not based on the signature of the video. It is difficult to find whether a video is a copied video or a similar video, since the features of the content are very similar from one video to the other. The main focus of this paper is to detect that the query video is present in the video database with robustness depending on the content of video and also by fast search of fingerprints. The Fingerprint Extraction Algorithm and Fast Search Algorithms are adopted in this paper to achieve robust, fast, efficient and accurate video copy detection. As a first step, the Fingerprint Extraction algorithm is employed which extracts a fingerprint through the features from the image content of video. The images are represented as Temporally Informative Representative Images (TIRI). Then, the second step is to find the presence of copy of a query video in a video database, in which a close match of its fingerprint in the corresponding fingerprint database is searched using inverted-file-based method. The proposed system is tested against various attacks like noise, brightness, contrast, rotation and frame drop. Thus the performance of the proposed system on an average shows high true positive rate of 98% and low false positive rate of 1.3% for different attacks.

KEYWORDS

Video copy detection, Content-based fingerprinting, multimedia fingerprinting, video copy retrieval

1. INTRODUCTION

Image Mining deals with extraction of implicit knowledge, image data relationship or other patterns not explicitly stored in images and uses ideas from computer vision, image processing, image retrieval, data mining, machine learning, databases and AI. The fundamental challenge in image mining is to determine how low-level, pixel representation contained in an image or an image sequence can be effectively and efficiently processed to identify high-level spatial objects and relationships. Typical image mining process involves preprocessing, transformations and feature extraction, mining to discover significant patterns out of extracted features, evaluation and interpretation and obtaining the final knowledge. Various techniques are also applied to image mining and include object recognition, learning, clustering and classification.

Video Copy Detection is based on detecting video copies from a video sample. Thus, copyright violations can be avoided. In video copy detection based on the content[4], the signature which defines the video in terms of content. The function of the video copy retrieval algorithms based on its content extract the fingerprint[10] through the features of the visual content of video. Then the fingerprint is used to compare with fingerprints from videos in a database. The problem associated with this type of algorithms is difficult to find whether a video is a copied video or a similar video. The features of the content are very similar from one video to the other and appear as a copied image. For example, film archives.

Video fingerprinting methods extract several unique features of a digital video that can be stored as a fingerprint of the video content. Fingerprints are feature vectors that can uniquely characterize the video signal. The goal of a video fingerprinting system is to judge whether two videos have the same contents by measuring distance between fingerprints extracted from the videos. To find a copy of a query video in a video database, one can search for a close match of its fingerprint in the corresponding fingerprint database, which is extracted from the videos in the database. The closeness of two fingerprints represents a similarity between the corresponding videos; two perceptually different videos should have different fingerprints. The overall structure of the fingerprinting system is shown in Fig. 1.

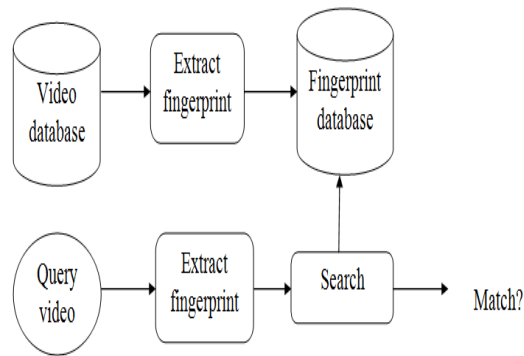


Figure 1. Fingerprinting system.

1.1 Properties of Fingerprints

The video fingerprints generally need to satisfy the following properties:

- a) *Robustness*: The fingerprints extracted from a degraded video should be similar to fingerprints of the original video.
- b) *Pairwise independence*: When two different videos are considered, the fingerprint extracted from those videos should also be different.
- c) *Database search efficiency*: For large-scale database applications, the fingerprints should be efficient for DB search.

1.2. Types of Fingerprints

The existing video fingerprint extraction algorithms can be classified into four groups based on the features that they extract: color-space-based, temporal, spatial, and spatio-temporal fingerprinting. *Color-space-based fingerprints* are derived from the histograms of the colors in specific regions over a particular time and/or space within the video. The color features are popular because the features change with different video formats, at the same time these are not applicable to black and white videos. *Temporal fingerprints* are derived from the characteristics of a video sequence over time. Although these features perform work well with long video

sequences, they do not perform well for short video clips since they do not contain sufficient discriminant temporal information. Because short video clips occupy a large share of online video databases, temporal fingerprints alone do not suit online applications. *Spatial fingerprints* are features derived from each and every frame or from a key frame. They are widely used for both video and image fingerprinting. Spatial fingerprints are subdivided into global and local fingerprints. Global fingerprints represent the global properties of a frame or a subsection of it (e.g., image histograms), while local fingerprints represent local information around some interest points within a frame (e.g., edges). The space time interest points correspond to points where the image values have significant local variation in both space and time. A spatial-temporal fingerprinting is based on the differential of luminance of partitioned grids in spatial and temporal regions.

2. RELATED WORK

Sunil lee, *Member, IEEE*, and Chang D. Yoo, [4] states that the centroid of gradient orientations is chosen due to its pairwise independence and robustness against common video processing steps that include lossy compression, resizing, frame rate change, etc. A threshold used to find a fingerprint match derived by modeling this fingerprint. The performance of this fingerprint system is compared with that of other widely-used features.

M. Malekesmaeili, M. Fatourechi, and R. K.Ward, [5] proposes an approach for generating representative images of a video sequence that carry the temporal as well as the spatial information. These images are denoted as TIRIs, Temporally Informative Representative Images[5]. Performance of the approach is demonstrated by applying a simple image hashing technique on TIRIs of a video database.

A. Gionis, P. Indyk, and R. Motwani, suggested a novel scheme for approximate similarity search is examined based on hashing[2]. The hashing technique used here is based on locality-sensitive hashing. The basic idea is to hash the points from the database so as to ensure that the probability of collision is much higher for objects that are close to each other than for those that are far apart. The query time improves even by allowing small error and storage overhead. This technique provides a better result for large number of dimensions and data size.

B. Coskun, B. Sankur, and N. Memon [1] proposes two robust hash algorithms for video are based both on the Discrete Cosine Transform (DCT), one on the classical basis set and the other on a novel randomized basis set (RBT). The robustness and randomness properties of the hash functions are resistant to signal processing and transmission impairments, and therefore can be instrumental in building database search, broadcast monitoring and watermarking applications for video. The DCT hash is more robust, but lacks security aspect, as it is easy to find different video clips with the same hash value.

Roover *et al.* extracted the variance of the pixels in different radial regions passing through the center of the key frames [9] based on a set of radial projections of the image pixel luminance values. A drawback of the key-frame-based techniques is the sensitivity of the key frames to frame dropping and noise [9]. This might adversely affect copyright protection applications as a pirate can change the hash by manipulating the key frames or shot boundaries.

3. PROPOSED SCHEME

This paper relies on a fingerprint extraction algorithm followed by a fast approximate search algorithm. Fingerprints are feature vectors that can uniquely characterize the video signal. Video

fingerprinting methods extract several unique features of a digital video that can be stored as a fingerprint of the video content. Video fingerprinting is technology that has proven to be effective in identifying and comparing digital video data. The goal of a video fingerprinting system is to judge whether two videos have the same contents by measuring distance between fingerprints extracted from the videos. The fingerprint extraction algorithm[4] extracts compact content-based fingerprint[8] from special images constructed from the video. Each such image represents a short segment of the video and contains temporal as well as spatial information about the video segment[7]. These images are denoted by temporally informative representative images (tiri) [5],[11]. The image also contains information about possible existing motions in the video. The fast search algorithms used for finding the match between the videos are inverted file based method. To find whether a query video (or a part of it) is copied from a video in a video database, the fingerprints of all the videos in the database are extracted and stored in advance as shown in Fig. 2. The search algorithm[6] searches the stored fingerprints to find close enough matches for the fingerprints of the query video.

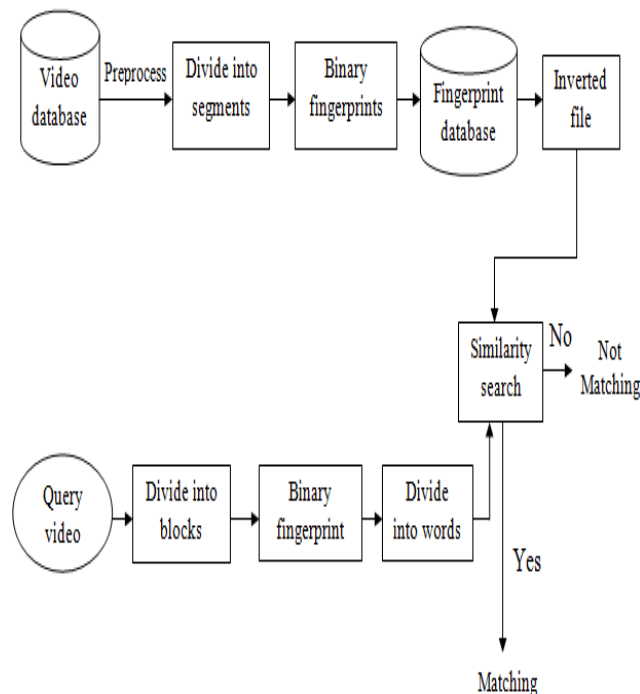


Figure 2. Overall process of fingerprinting system.

3.1 Feature Extraction

3.1.1 Generating TIRI-DCT

The fingerprint algorithm would be robust to changes in the frame size by applying pre-processing. Down-sampling can increase the robustness of a fingerprinting algorithm to these changes. This step consists of resizing of the video into fixed $W \times H$, where $W \times H$ is the frame size. After this pre-processing step, the video is segmented into fixed short segments. Therefore the frames obtained from the videos are resized.

The preprocessed images are converted to grayscale images, in order to obtain luminance value of the pixels of all available frames. This will be used to compute the weighted sum of the frames, i.e., the pixels of representative images[5]. The weighted average method used here is an

exponential method. This exponential weighting function produces perceptually better results and also generates images that best capture the motion of video. The pixels of representative images are used for the generation of DCT (Discrete Cosine Transform)[1]. The steps involved in TIRI-DCT are as follows:

- a) Extract frames from given video input.
- b) Pre-process the frames obtained.
- c) Convert RGB image into grayscale image.
- d) Compute pixels of TIRI as the weighted sum of frames using

$$I'_{m,n} = \sum_{k=1}^J w_k I_{m,n,k}$$

where $I'_{m,n}$ - pixels of TIRI.

w_k - weighted average.

$I_{m,n,k}$ - luminance value of $(m,n)^{th}$ pixel of k^{th} frame.

- e) Generate DCT for TIRI.

3.1.2 Binary Fingerprint

TIRI-DCT is segmented into number of blocks. The DCT-based hash, which uses low frequency 2D-DCT coefficients of the TIRIs is used because of its better detection characteristics. The features of input video are derived by applying a 2D-DCT on the blocks of size $n \times n$ from each TIRI. From each of these blocks, 1st horizontal and 1st vertical coefficients are extracted. These coefficients[5] of each block of representative images are concatenated and will be used to calculate the median as shown in Fig. 3.

The binary fingerprints[4] are generated by comparing the median value(threshold) and the values of coefficients as follows: If the values of coefficients are greater than or equal to median value, then the value 1 is assigned to binary hash. If the values of coefficients are less than median value, then the value 0 is assigned to binary hash.

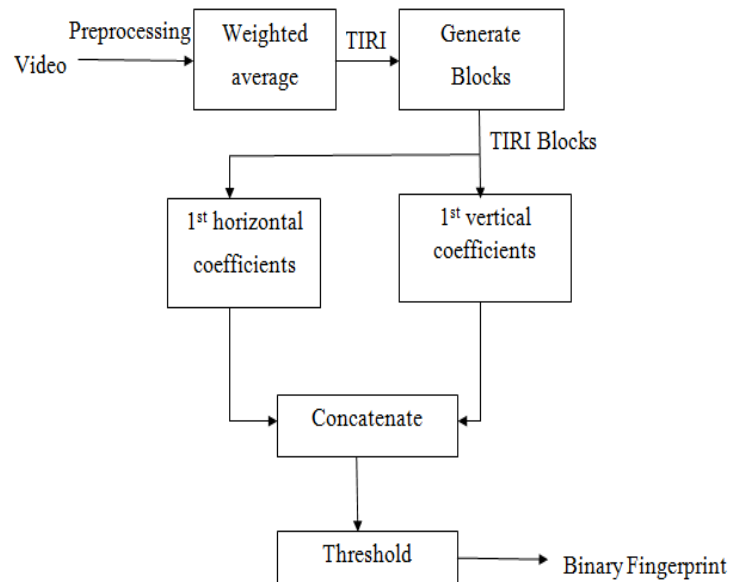


Figure 3. Steps involved in generating binary fingerprint.

The steps involved in generating binary fingerprint are:

- a) TIRI-DCT is segmented into number of blocks.
- b) Extract 1st horizontal coefficients of DCT.
- c) Extract 1st vertical coefficients of DCT.
- d) Concatenate horizontal and vertical coefficients.
- e) Compute median m from the concatenated values.
- f) Compare coefficients with median.
- g) If the values of coefficients are greater than or equal to median value, then the value 1 is assigned to it.
- h) If the values of coefficients are greater than or equal to median value, then the value 1 is assigned to it.

3.2 Inverted File Based Similarity Search

The binary fingerprints are divided into n words with m equal number of bits. Each of those m bits are termed as words, which are used to create table of size, $2^m * n$ where 2^m represents the possible values of words and n represents position of word. The horizontal and vertical dimensions of the table represent the possible values and position of a word respectively. To generate this table[3], consider the first word of each fingerprints and add the index of the fingerprint to the entry in the first column corresponding to the value of this word. This process is continued for all the words in each fingerprint and all the columns in the inverted file table[9].

Once a inverted file index has been created, it can be used to match a fingerprint of query video against the collection. To find a query fingerprint in the database, first the fingerprint is divided into words. The query is then compared to all the fingerprints that start with the same word. The indices[2] of these fingerprints are found from the corresponding entry in the first column of the inverted file table.

The Hamming distance[12] is calculated between fingerprints in database and query fingerprint. If the distance is less than threshold value, then the query video will be announced as matching, otherwise as not matching with the database. The steps involved are as follows:

- a) Binary fingerprints are divided into n words of equal bits.
- b) The horizontal dimension of the table represents the position of a words.
- c) The vertical dimension of the table represents the possible values of words.
- d) Add index for each word of the fingerprint to the entry in column corresponding to the value of the word.
- e) Hamming distance is calculated between fingerprints in database and query fingerprint.
- f) If the distance is less than threshold value, then the query video will be announced as matching.
- g) Otherwise, it will be announced as not matching.

In the fingerprint matching process, two videos are declared similar if the distance between their fingerprints is below a certain threshold.

4. EXPERIMENTAL RESULT

The performance of the proposed video fingerprinting method is evaluated using the fingerprint Database generated. The length and the resolution of the videos in the DB

range up to 4 minutes, and from 384 X 288. The frame rate is 25 fps for all videos. The videos are chosen to be in avi(Audio Video Interleaved) format. This system announces whether the query video matches video in database using inverted file based similarity search method.

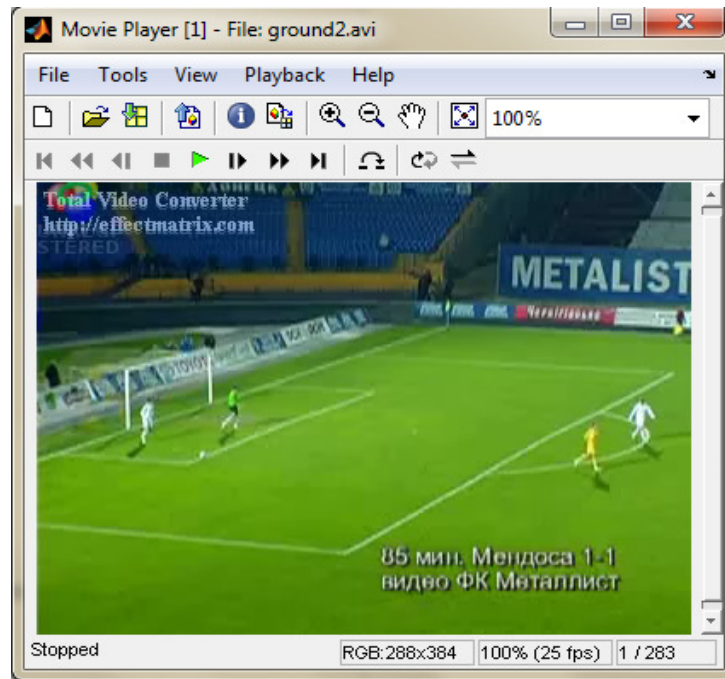


Figure 4. Input video

The Fig. 4 shows the video of avi format from input video database. By iteration, all videos in video database can be read. Depending on the frame rate and duration of the video, number of frames present in it varies. Frames of all videos are extracted.

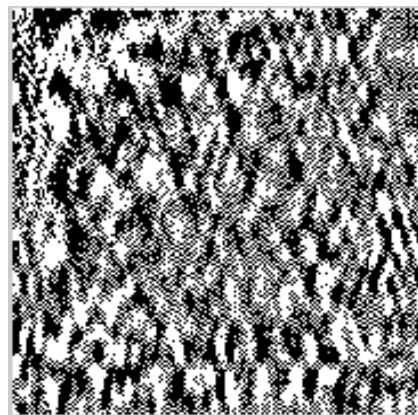


Figure 5. DCT of video

For all the frames extracted from the video, preprocessing is performed. The preprocessing technique used in this process is down sampling. Here, the sizes of frames are reduced uniformly for all videos in database. The reduced size of frame is 128*128. The RGB images are converted into grayscale images, in order to obtain luminance value of the pixels of all available frames. This will be used to compute the weighted sum of the frames, i.e., the pixels of representative

images. The Fig. 5 shows DCT being applied to representative image of video and segmented into number of blocks. From each blocks, horizontal and vertical coefficients are extracted.

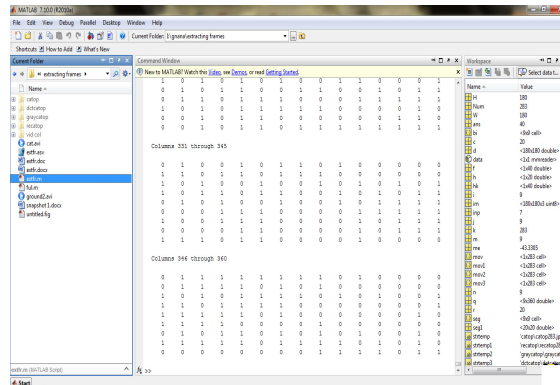


Figure 6. Binary fingerprint of video

Concatenate all coefficients of each block of DCT video to find the value of median, which will be used to compare with coefficients to generate binary fingerprint. If the values of coefficients are greater than or equal to median value, then the value 1 is assigned to it. If the values of coefficients are less than median value, then the value 0 is assigned to it. The Fig. 6 shows binary fingerprint of video obtained.

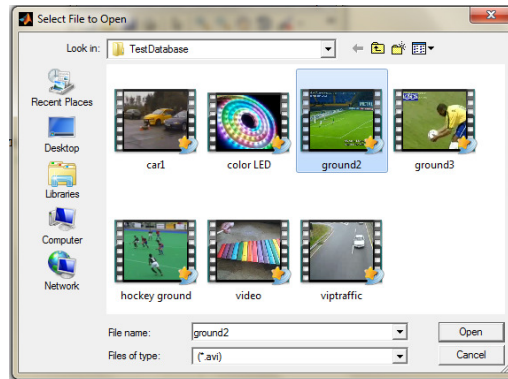


Figure 7. Selection of query video

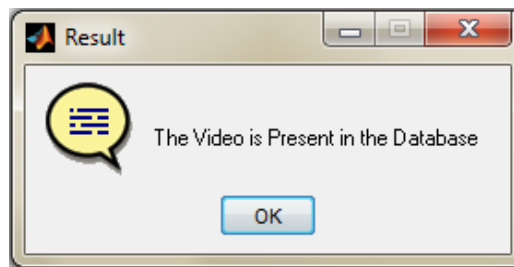


Figure 8. Announcement of match in database

The Fig. 7 shows the selection of query video, for which match in database will be found. The binary fingerprint of query video is obtained by following similar procedure as that in generation of binary fingerprint of a video in video database. The Fig. 8 shows announcement of matching of query video with video in database based on the distance between the fingerprints in database and

query fingerprint using inverted file based similarity search method. If the DB position with the minimum distance exactly corresponds to the input fingerprint sequence in the processed video, it is assumed that the input fingerprint sequence is correctly identified.

Attacks	Noise	Rotation	Brightness	Contrast	Frame drop
TPR(%)	98.15	99.10	98.73	97.91	96.46
FPR(%)	1.64	0.87	0.90	1.52	1.78
F-Score	0.98	0.98	0.99	0.98	0.99

Table 1. Performance of Inverted File Based Similarity Search

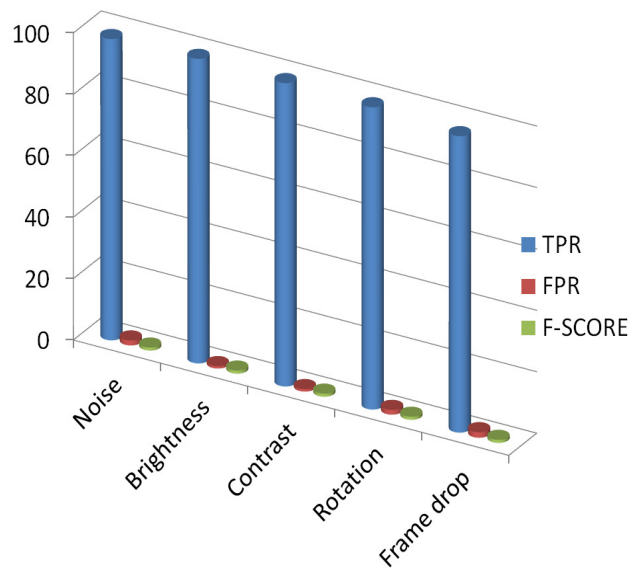


Figure 9. Accuracy based on various attacks

The attacks are mounted independently on the videos to generate the queries. Table 1 represents performance of inverted file based similarity search based on true positive rate (TPR) and false positive rate (FPR). Percentage of accuracy in detecting similarity between the query video and videos in the database using inverted file based similarity search is shown in Fig. 9. Thus the inverted file based similarity search provides better performance in detecting the exact match between the videos.

5. CONCLUSION

The proposed fingerprinting algorithm, TIRI-DCT extracts robust, discriminant, and compact fingerprints from videos in a fast and reliable fashion. These fingerprints are extracted from TIRIs containing both spatial and temporal information about a video segment. The proposed fast approximate search algorithm, the inverted file based method, which is a generalization of an existing search method is fast in detecting whether the query video is present in video database. By using inverted file based similarity search for detecting the similarity among the videos, the performance of the system yield high true positive rate and low false positive rate. Future work

includes implementation of cluster based similarity search method and product quantizer method detects whether the query video is present in video database. The performance of all the above similarity search methods will be evaluated in order to find the best similarity search technique in an efficient manner.

REFERENCES

- [1] B. Coskun, B. Sankur, and N. Memon, "Spatiotemporal transform based video hashing," *IEEE Trans. Multimedia*, vol. 8, no. 6, pp.1190–1208, Dec. 2006.
- [2] A. Gionis, P. Indyk, and R. Motwani, "Similarity search in high dimensions via hashing," in *Proc. Int. Conf. Very Large Data Bases (VLDB)*, San Francisco, CA, 1999, pp. 518–529, Morgan Kaufmann Publishers Inc..
- [3] A. Hampapur and R. M. Bolle, Videogrep: Video copy detection using inverted file indices IBM Research Division Thomas. J. Watson Research Center, Tech. Rep., 2001.
- [4] S. Lee and C. Yoo, "Robust video fingerprinting for content-based video identification," *IEEE Trans. Circuits Syst. Video Technol.*, vol.18, no. 7, pp. 983–988, Jul. 2008.
- [5] M. Malekesmaeili, M. Fatourehchi, and R. K. Ward, "Video copy detection using temporally informative representative images," in *Proc. Int. Conf. Machine Learning and Applications*, Dec. 2009, pp. 69–74.
- [6] Mani Malek Esmaeili, Mehrdad Fatourehchi, and Rabab Kreidieh Ward, *Fellow, IEEE*, "A robust and fast video copy detection system using content-based fingerprinting," *IEEE Trans. on Information Forensics and Security*, vol. 6, no. 1, Mar. 2011.
- [7] J. Law-To, L. Chen, A. Joly, I. Laptev, O. Buisson, V. Gouet-Brunet, N. Boujemaa, and F. Stentiford, "Video copy detection: A comparative study," in *Proc. ACM Int. Conf. Image and Video Retrieval*, New York, NY, 2007, pp. 371–378, ACM.
- [8] R. Radhakrishnan and C. Bauer, "Content-based video signatures based on projections of difference images," in *Proc. MMSP*, Oct. 2007, pp.341–344.
- [9] C. De Roover, C. De Vleeschouwer, F. Lefebvre, and B. Macq, "Robust video hashing based on radial projections of key frames," *IEEE Trans. Signal Process.*, vol. 53, no. 10, pp. 4020–4037, Oct. 2005.
- [10] G. Willems, T. Tuytelaars, and L. Van Gool, "Spatio-temporal features for robust content-based video copy detection," in *Proc. ACM Int. Conf. Multimedia Information Retrieval*, New York, NY, 2008, pp. 283–290, ACM.
- [11] M. Malekesmaeili and R. K. Ward, "Robust video hashing based on temporally informative representative images," in *Proc. IEEE Int. Conf. Consumer Electronics*, Jan. 2010, pp. 179–180.
- [12] M. L. Miller, "Audio fingerprinting: Nearest neighbour search in high dimensional binary spaces," in *IEEE Workshop on 2002, Multimedia Signal Processing, 2002*, 2002, pp. 182–185.

Authors

Ms.R.Gnana Rubini has received Bachelor of Technology degree in Information Technology under Anna University, Chennai in 2010. She is currently pursuing Master of Engineering degree in Computer Science and Engineering under Anna University, Coimbatore, India. Her areas of interest are Data Mining and Image Processing.



Prof. P.Tamije Selvy received B.Tech (CSE), M.Tech (CSE) in 1996 and 1998 respectively from Pondicherry university. Since 1999, she has been working as faculty in reputed Engineering Colleges. At Present, she is working as Assistant Professor(SG) in the department of Computer Science & Engineering, Sri Krishna College of Technology, Coimbatore. She is currently pursuing Ph.D under Anna University, Chennai. Her Research interests include Image Processing, Data Mining, Pattern Recognition and Artificial Intelligence.



Ms.P.Anantha Prabha obtained B.E in Electronics and Communication Engineering and Master of Engineering in Computer Science and Engineering from V.L.B. Janakiammal College of Engineering and Technology, Coimbatore, India in 2001 and 2008 respectively. She has been working in various Engineering Colleges for 9 years. She is currently working as an Assistant Professor in Sri Krishna College of Technology, Coimbatore. Her areas of interest are Clouding Computing, Mobile Computing and Image Processing.

