Video Fingerprinting: Features for Duplicate and Similar Video Detection and Query-based Video Retrieval

Anindya Sarkar, Pratim Ghosh, Emily Moxley and B. S. Manjunath

Department of Electrical and Computer Engineering
University of California, Santa Barbara, CA 93106

ABSTRACT

A video "fingerprint" is a feature extracted from the video that should represent the video compactly, allowing faster search without compromising the retrieval accuracy. Here, we use a keyframe set to represent a video, motivated by the video summarization approach. We experiment with different features to represent each keyframe with the goal of identifying duplicate and similar videos. Various image processing operations like blurring, gamma correction, JPEG compression, and Gaussian noise addition are applied on the individual video frames to generate duplicate videos. Random and bursty frame drop errors of 20%, 40% and 60% (over the entire video) are also applied to create more noisy "duplicate" videos. The similar videos consist of videos with similar content but with varying camera angles, cuts, and idiosyncrasies that occur during successive retakes of a video. Among the feature sets used for comparison, for duplicate video detection, Compact Fourier-Mellin Transform (CFMT) performs the best while for similar video retrieval, Scale Invariant Feature Transform (SIFT) features are found to be better than comparable-dimension features. We also address the problem of retrieval of full-length videos with shorter-length clip queries. For identical feature size, CFMT performs the best for video retrieval.

Keywords: video fingerprinting, Fourier-Mellin Transform, SIFT, ordinal features, query-based video retrieval

1. INTRODUCTION

Video search and retrieval is a topic of active research. Here, we search for similar and duplicate videos, where the similar videos are different retakes of a single scene, and various image processing operations are performed on the different videos to create the different "similar" videos. A common approach to solving the search and retrieval problem is by the use of video fingerprinting - video "keyframes" are intelligently chosen and image features, computed per keyframe, are stored as the "video" fingerprint. A good fingerprint represents the video efficiently in the form of a compact feature that facilitates fast search without sacrificing retrieval accuracy. Understandably, there is a trade-off between feature compactness and retrieval accuracy. Here, we have experimented with the following features - Scale Invariant Feature Transform (SIFT), Compact Fourier Mellin Transform (CFMT), ordinal histogram and YCbCr histogram based feature. We present precision-recall analysis to compare detection accuracy using different features as "fingerprints." The goodness of the fingerprint has also been considered in the query-based retrieval scenario where a certain fraction of a full-length video is presented as the query. Based on the stored fingerprints, the original video can be retrieved even with a short-length query.

An early work in video copy identification based on a keyframe based approach was by Joly et al.[1] The approach was to first extract keyframes from a video and then compute the local descriptors from each keyframe, based on the Harris corner detector. The noise attacks that were considered were spatial Gaussian noise addition, gamma variations, spatial resizing and vertical shifts. In a later work,[2] they improved the indexing structure for video fingerprints, based on Statistical Similarity Search. Lee and Doo[3] proposed a fingerprinting approach based on the centroids of gradient orientations (CGO) - the video frames are divided into a fixed number of

Further author information: (Send correspondence to Anindya Sarkar)
Anindya Sarkar: E-mail: anindya@ece.ucsb.edu, Telephone: 1-805-893-5682

non-overlapping blocks and a CGO based descriptor is computed for each of the non-overlapping blocks. A constraint in their method is that the number of extracted frames should be same for the two sequences being compared. The attacks considered were brightness changes, red channel variations, Gaussian blurring, histogram equalization, frame rate change and frame resizing.

The development of "ordinal" features[4,5] gave rise to very compact signatures and they were also used for video sequence matching.[6] Li et al[7] used a binary signature to represent each video - they merged color histogram with ordinal signatures for feature representation. Yuan et al[8] also used a combination of ordinal histograms and cumulative color histograms for robust similarity search and copy detection. They also proposed an efficient index structure for the short video clip search.[9]

A compact and robust fingerprint based on a novel space-time color feature was proposed by Yang et al.[10] Here, each shot is divided into a fixed number of equal size segments, each of which is represented by a blending image obtained by averaging of the pixel values of all the frames in a segment. Each blending image is divided into blocks of equal size, and then two color signatures are extracted per block. The fingerprint was shown to be effective for content identification. A recent paper by Zhao et al[11] was based on near-duplicate keyframe identification based on matching, filtering and learning of local interest points. The proposed approach aws shown to outperform other popular methods, including those employing Locality Sensitive Hashing (LSH).[12]

2. PROBLEM FORMULATION

2.1 Duplicate and Similar Video Detection

We first outline the process used to generate the duplicate and similar videos, created from BBC rushes used for the TRECVID-2007 video summarization task.[13]

2.1.1 Duplicate Videos

We use image processing techniques on the decoded video frames to generate the duplicate videos. The image processing operations applied to create duplicate videos are:

1. blurring using a 3×3 and 5×5 window
2. gamma correction by -20% and 20%
3. addition of AWGN (additive white Gaussian noise) using SNR of -20, 0, 10, 20, 30 and 40 dB
4. JPEG compression using quality factors of 10, 30, 50, 70 and 90

Thus, given a certain video, we end up with 16 (15 generated through the various processing methods and 1 original) duplicate videos.

Apart from the image processing based duplicates, we also introduce frame drop-based errors - the frame drops can be random (frame drops may occur throughout the video) or bursty (frame drops may be concentrated at a certain region of the video). We have experimented with frame drops of 20%, 40% and 60% of the original video. Thus, for a given video, we have 64 ($(1 + 15) \times (1 + 3)$) duplicates - for each of the bursty and non-bursty error cases.

2.1.2 Similar Videos

The BBC rushes dataset was specifically chosen for facilitation of creation of similar videos. The videos in this dataset contain retakes of various shots. Each original video that is considered is a scene with one or more retakes. We consider the different retakes of a single scene as "similar" videos.

If the i^{th} scene has N_i number of retakes, we obtain N_i similar videos. Now, for every retake, we generate D duplicates as described in Sec. 2.1.1, where D=64. For the i^{th} scene, with N_i retakes, there are $N_i \times D$ similar videos. Each video has two to six retakes; thus, the number of similar videos generated will be at least 128 (2×64).

For browsing, a compact signature is generated per "significant" frame. If an image-based signature is to be stored and the video has a large number of frames, the size of the signature can be reduced only if "significant" keyframes are selected. A video summarization technique is used to identify keyframes, and a feature is extracted to characterize each keyframe. The collection of these keyframe features constitute the video signature.

We experiment with the following keyframe features:

1. **Compact Fourier-Mellin Transform (CFMT)**-based signature[14–16] at multiple dimensions
2. **YCbCr histogram**-based feature computed over a video segment
3. **Scale Invariant Feature Transform (SIFT)**-based feature[17] at multiple dimensions

The ordinal histogram feature[4–6] is an example of a feature that is computed per video, rather than per keyframe, and has been shown to be useful for video fingerprinting. A brief comparison is made between the keyframe-based signatures, and a video-based signature as calculated using ordinal features.

For comparing signatures computed from different videos, the signatures generated are of equal length (though actual video sizes do differ). The signature for all videos is based on eight keyframes.

2.2 Retrieval of "Large" Video Using "Small" Query Clip

We use a repository of 64 "large" (30000-50000 frames) videos from the BBC rushes dataset. For building the input query, we use only four of these videos. We hand-annotate the repeated shots in the videos used for querying. For the query, we combine the signatures of three or four distinct scenes in the video. As mentioned in Section 2.1, for each distinct scene, we use a eight-frame summary. So, for a video query comprised of three scenes, there will be 24 ($8 \times 3 = 24$) keyframes in all. The size of the input query is either a 24-frame or a 32-frame feature depending on whether three to four distinct scenes are included.

The "large" video database has videos with a varying number of scenes. While considering the large videos, the connection between the video length and the signature size is as follows. Say, for a given video of N frames, we wish its video summary to be 4% of its length. Using a keyframe based approach (Fig. 1) for summarization and using a window of 15 frames on either side of a keyframe, we obtain $K = \frac{0.04N}{31}$ keyframes. For each keyframe, a d-dimensional feature is computed. Then, the total signature size for the whole video = $K \times d$.

For the YCbCr feature, to obtain a $K \times d$ dimensional signature, we consider K equal sized windows for feature generation (Fig. 2), each window having N/K frames.

3. CFMT FEATURE

The Fourier-Mellin transform (FMT) has been studied extensively in the context of watermarking[18, 19] and invariant object recognition.[15, 20, 21] All these methods exploit the fact that this transform generates a rotation, translation, and scale-invariant representation of the images. The FMT was first introduced in[14] and our implementation is based on the fast approximation described in.[15]

The classical FMT of a 2D function f, $T_f(k, v)$ is defined as:

$$T_f(k, v) = \frac{1}{2\pi} \int_0^{\infty} \int_0^{2\pi} f(r, \theta) r^{-iv} e^{-ik\theta} d\theta \frac{dr}{r} \tag{1}$$

where (k, v) and (r, θ) are respectively the variables in Fourier-Mellin and polar domain representation of the function f. Ghorbel [16] suggested the concept of an Analytical Fourier-Mellin Transform (AFMT), an important modification to the problem associated with the existence of standard FM integral (the presence of $\frac{1}{r}$ term in the definition necessarily requires f to be proportional to r around the origin such that when $r \to 0$, then $f \to 0$). The AFMT, $T_{f,\sigma}(k, v)$, is defined as:

$$T_{f,\sigma}(k, v) = \frac{1}{2\pi} \int_0^{\infty} \int_0^{2\pi} f(r, \theta) r^{\sigma - iv} e^{-ik\theta} d\theta \frac{dr}{r} \tag{2}$$

where σ, a strictly positive parameter, determines the rate at which f tends toward zero near the origin.

Let $f_1(x, y)$ be an image and its rotated, scaled and translated version $f_2(x, y)$ is given by the equation:

$$f_2(x, y) = f_1(\alpha(x \cos\beta + y \sin\beta) - x_o, \alpha(-x \sin\beta + y \cos\beta) - y_o) \tag{3}$$

where the rotation and scale parameters are β and α respectively, and $[x_o, y_o]$ is the translation. It can be shown that for rotated and scaled images, the magnitudes of the AFM transforms, $|T_{f_1,\sigma}|$ and $|T_{f_2,\sigma}|$, (corresponding to f_1 and f_2, respectively) are related by the equation:

$$|T_{f_2,\sigma}(k, v)| = \alpha^{-\sigma}|T_{f_1,\sigma}(k, v)| \tag{4}$$

An AFMT leads to a scale and rotation invariant representation after proper normalization by $1/\alpha^{-\sigma}$. Finally, the CFMT representation can be made translation invariant by computing the AFMT on the Fourier transformed image (considering only the magnitude part).

Once the AFM coefficients are extracted, Principal Component Analysis (PCA) and Lloyd-Max non-uniform scalar quantization are applied to obtain a compact representation, the CFMT descriptor. Each dimension of the CMFT descriptor is quantized to 256 levels.

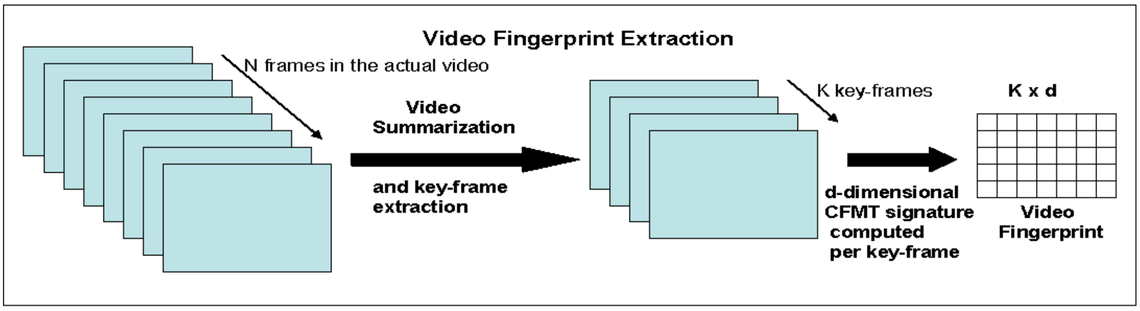

Video Fingerprint Extraction
N frames in the actual video
Video Summarization
and key-frame extraction
K key-frames
d-dimensional CFMT signature computed per key-frame
K x d
Video Fingerprint

Figure 1. Generation of CFMT-Based Signature

4. YCBCR HISTOGRAM FEATURE

The YCbCr histogram is computed over a certain window, as in Figure 2. For a N-frame video and K keyframes, we use a window having $P = \frac{N}{K}$ frames for histogram computation. For a window of P frames, we allocate five bins for each of the Y, Cb, and Cr axes, thus making it a 125-dimensional feature per window. So, the effective size of the signature for the entire video, considering K keyframes = $K \times 125$.

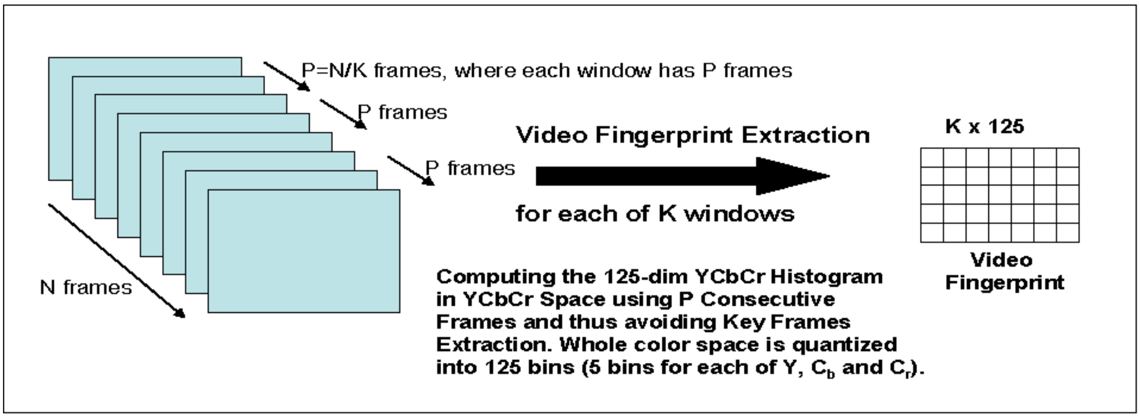

P=N/K frames, where each window has P frames
P frames
P frames
N frames
Video Fingerprint Extraction
for each of K windows
K x 125
Video Fingerprint
Computing the 125-dim YCbCr Histogram in YCbCr Space using P Consecutive Frames and thus avoiding Key Frames Extraction. Whole color space is quantized into 125 bins (5 bins for each of Y, C_b and C_r).

Figure 2. Generation of YCbCr Histogram-Based Signature

In contrast with the CFMT-based signature, keyframe extraction is not required here as the YCbCr histogram is computed over a window. So, for the duplicate and similar videos, where we had allocated eight keyframes for the CFMT based approach, we have eight adjacent, non-overlapping equal-sized windows for the YCbCr case that span the entire video.

5. SIFT FEATURE

SIFT descriptors were introduced in[17] as features that are relatively robust to scale, perspective, and rotation changes, and their accuracy in object recognition has led to extensive use in image similarity. Computationally however they are costly since characterization of an image using SIFT descriptors generally solicits thousands of features for a single image which must be pairwise compared, as shown in Figure 3. It is not the dimensionality of the descriptor (typically 128) that bottlenecks SIFT image comparison, but the number of descriptor comparisons, so dimensionality reduction techniques such as PCA-SIFT[22] are insufficient for large datasets such as constitute video frames. One way to speed the comparison is quantizing the descriptors to a finite "vocabulary" as shown in Figure 4(a). Then, each image is represented as a weighted vector of the quantized descriptor ("word") frequencies. The dimensionality of the signature is a tunable parameter, and can be thought of as the number of "words" in the vocabulary. With a larger vocabulary, the words are more representative of the actual descriptors (less quantization error), and therefore in general a larger vocabulary provides signatures that quantify similarity more accurately.

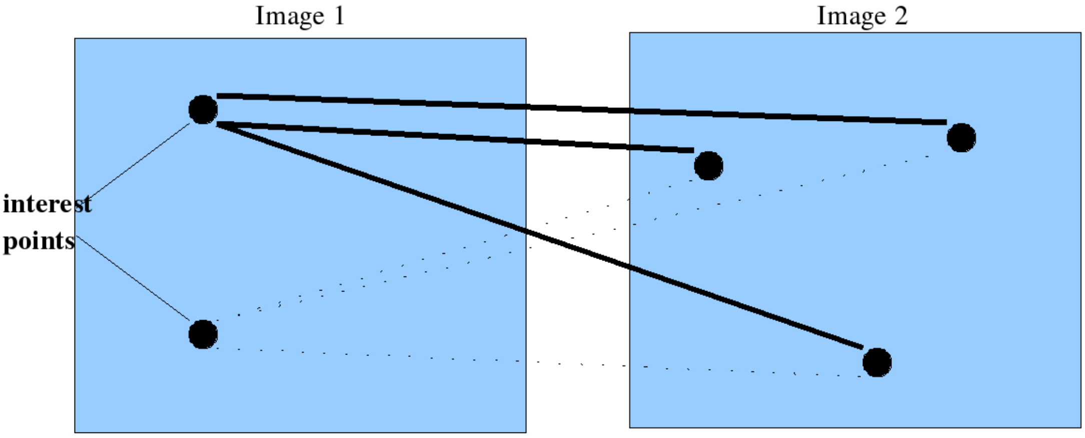
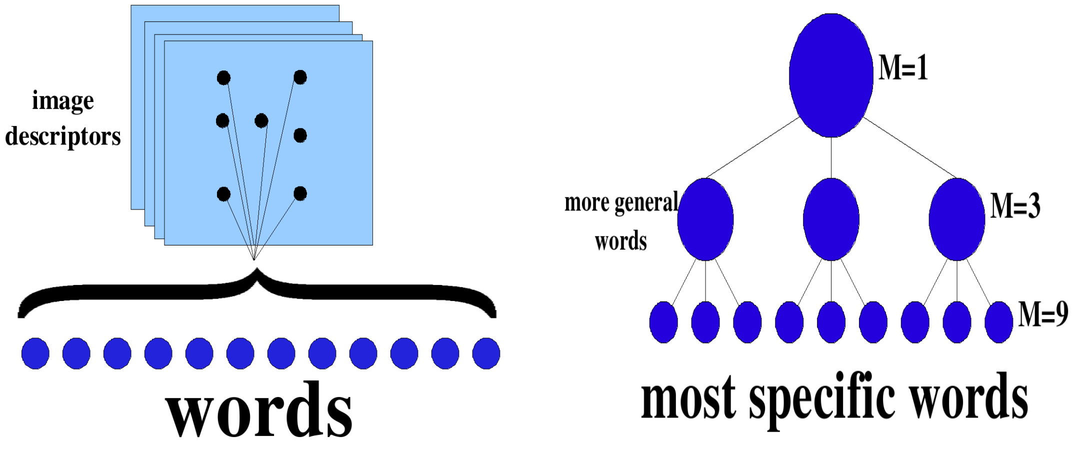


Figure 3. Pairwise SIFT descriptor comparison at numerous interest points.

One drawback of a straight vocabulary is its neglect of the relationship between words. Nister and Stewenius[23] propose a scheme where the vocabulary is created using hierarchical k-means clustering. The descriptor set is clustered into K clusters. Each of these K clusters is further divided into K clusters, producing $K \times K$ groups, and so on. Each cluster, at all resolutions, constitutes a word. Figure 4(b) demonstrates the vocabulary tree.

A straight vocabulary can be thought of as a vocabulary tree with depth 0 and multiple root nodes. *At extremely low dimensions, as necessary for a video fingerprint, it is likely that a straight vocabulary is more representative than a vocabulary tree.* The additional words provide a better description of the image than the relationship among words does. For instance, in Figure 4(b), the three words in the tree that capture the similarity of the leaf nodes are unlikely to provide more information than the addition of three more specific words.

The authors of[11] use SIFT for keyframe matching for video copy detection, employing a technique of one-to-one descriptor matching, rather than a vocabulary technique as explained above. Only descriptors, that are symmetrically first nearest neighbors, are considered for a match. This represents a "voting" technique for image matching that requires many descriptor comparisons (e.g., 5000×5000 comparisons in the initial example above, with each descriptor being compared having dimension 128). The vocabulary approach described above is much less computationally intensive than pairwise descriptor matching and is therefore more suitable for video search and retrieval.



(a) Straight Vocabulary

(b) Vocabulary Tree

Figure 4. **Straight Vocabulary vs. Vocabulary Tree** (a) Vocabulary created by clustering - depicted here for same dimensionality as tree (12). (b) Tree created using hierarchical k-means on SIFT features to a depth of 2. M is the number of nodes at each level. Using a branching factor of 3, the final vocabulary size=3+9=12.

6. ORDINAL HISTOGRAM FEATURE

The ordinal histogram feature was introduced for visual similarity detection in[4,5] and for video sequence matching in.[6] E.g. if a $3^2 = 9$-dimensional signature is desired, a given image is first re-sized to a 3×3 matrix. Let the image after being resized be: $\begin{bmatrix} 10 & 30 & 70 \\ 20 & 50 & 80 \\ 40 & 60 & 100 \end{bmatrix}$

Thus, scanning the image row-wise, from left to right, the ordinal based signature will be [1 3 7 2 5 8 4 6 9], denoting the arrangement of the nine numbers in ascending order. The ordinal features have been used to find similar regions in videos, with the assumption that the variation of the ordinal features is similar when studied across successive frames for similar scenes. Thus, if we use a keyframe-based approach, then the ordinal signature for each frame can be stored. However, since the image is resized, the feature loses out on details while being compact.

For video search purposes, a feature which is based on the entire video is generally more compact than a keyframe-based feature. A histogram is taken over all the frames to obtain a full video signature. In,[8,9] a 72-dimensional histogram has been used as a video signature. Each video frame is separately considered along R, G, and B axes. Along each axis, a 24-dimensional histogram is constructed considering all the frames. This gives a total signature size of $24 \times 3 = 72$. The way the 24-dimensional histogram is constructed is as follows: the image is resized to a 2×2 matrix for which the numbers 1,2,3,4 can occur in 4!=24 combinations. The 24 histogram bins represent the 4!=24 combinations.

7. SIGNATURE DISTANCE COMPUTATION

A proper distance measure must be chosen to compare video signatures. For the keyframe-derived signature, if $d(X,Y)$ denotes the distance between two $K \times d$ signatures X and Y (d-dim vectors being computed for K keyframes), then

$$d(X,Y) = \sum_{i=1}^{K} \left\{ \min_{1 \leq j \leq K} ||X(i) - Y(j)||_1 \right\} \tag{5}$$

where $||X(i) - Y(j)||_1$ refers to the L_1 distance computed between the i^{th} frame of X and the j^{th} frame of Y. Thus, the distance relation is not symmetric: $d(X,Y) \neq d(Y,X)$ in general. The motivation behind using this distance function, as in (5), is that for every vector in the first signature (X), we look for the best match out of all the vectors in the second signature (Y) and repeat this process for all the frames in X. Thus, if frame drops occur or some video frames are corrupted by noise, they will not adversely affect the distance between two duplicate videos, which would have been the case if an F-norm like distance is used, which compares the two signatures component-wise (assuming that the two videos follow the same temporal sequence). Also, for two video signatures, it is not obvious which frame in the first corresponds with which frame in the other. Therefore, we opt for a closest-overlap distance rather than a sequential frame-to-frame distance. This is based on the assumption that videos consisting of a reordering of scenes from the same video should be regarded as duplicates, though when viewed along the time axis, the videos may look different due to the scene reordering.

8. DUPLICATE AND SIMILAR VIDEO DETECTION RESULTS

Precision-recall values have been used to compare the performance of the signatures derived from YCbCr features, SIFT features, and CFMT features.

Let $A(H,\Gamma)$ be the set of H retrievals based on the smallest distances from the query video, Γ, in the signature space and $S(\Gamma)$ be the number of M videos in the database relevant to the query Γ. Then, precision P is defined by the number of videos retrieved relevant to a query divided by the number of retrievals, H.

$$P(H,\Gamma) \stackrel{\text{def}}{=} \frac{|A(H,\Gamma) \bigcap S(\Gamma)|}{H}$$

Recall which is defined as

$$R(H,\Gamma) \stackrel{\text{def}}{=} \frac{|A(H,\Gamma) \bigcap S(\Gamma)|}{M}$$

is the proportion of relevant images retrieved from $S(\Gamma)$. A precision-recall curve is obtained by plotting the average precision and recall values over a large number of queries Γ at a varying number of retrievals. Ostensibly, as the number of retrievals is increased, recall will increase as precision will decrease.

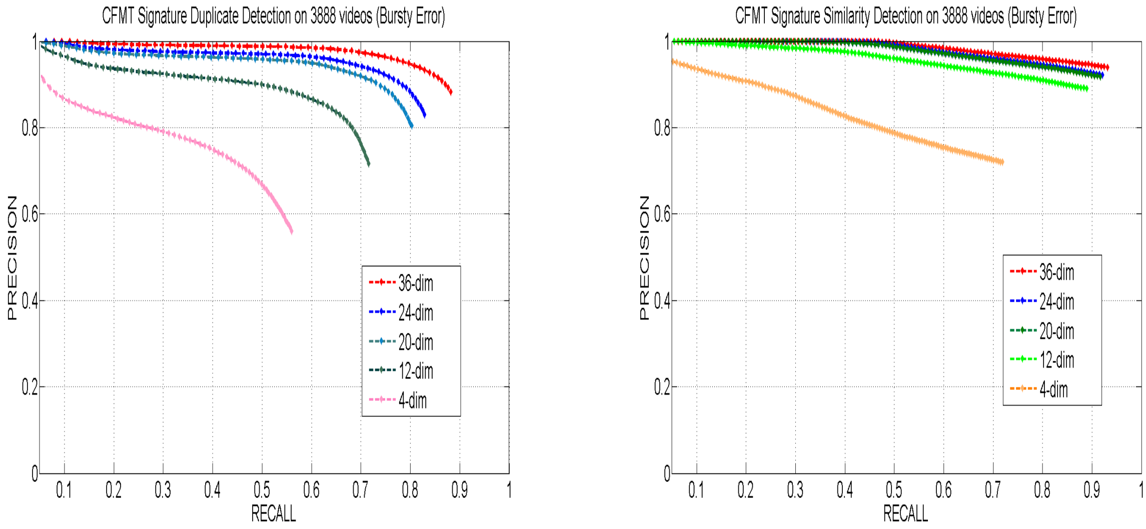
We generate precision-recall (P-R) graphs for the signatures generated from CFMT, YCbCr histogram, SIFT, and ordinal histogram features. The feature dimensions are varied to illustrate the trade-off between feature compactness and detection performance.

We present results comparing P-R performance for bursty and non-bursty errors for both similarity and duplicate detection, for the following set of features:

1. CFMT for dimensions 36/24/20/12/4 (Fig. 5 and 6)
2. YCbCr and ordinal histogram (compared with other descriptors in Fig. 9)
3. SIFT for dimensions 781/341/33/31/21/12 (Fig. 7 and 8)
4. CFMT vs best performing SIFT for duplicate detection (Fig. 9(a))
5. SIFT vs best performing CFMT for similarity detection (Fig. 9(b))

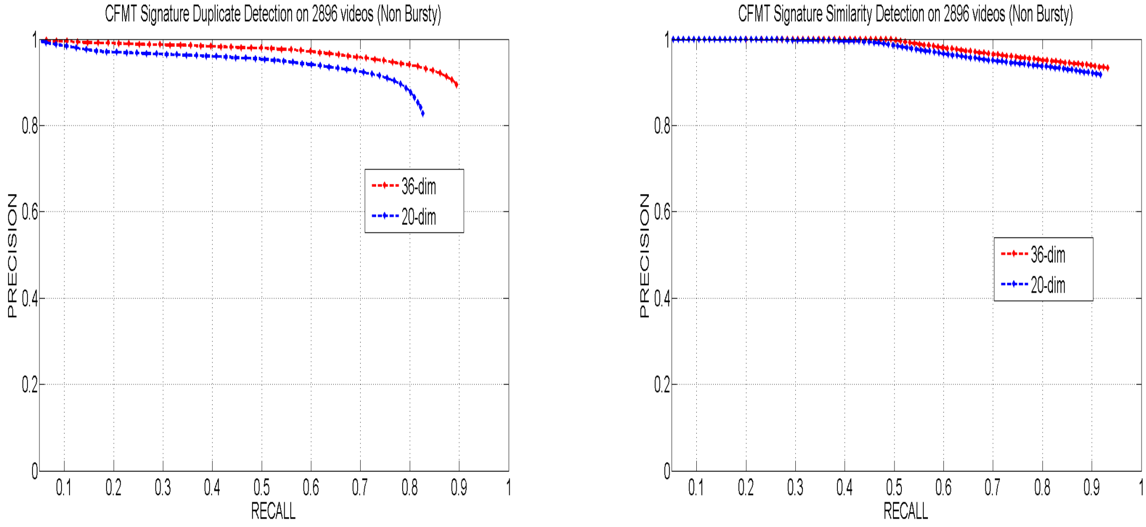
From the graphs, it becomes apparent that the precision-recall drop-off is quite small for duplicate detection when using a CFMT-based signature, in both bursty and nonbursty error cases. This indicates that even for a large number of retrievals, only few of the retrievals may be incorrect.

SIFT-based signatures perform quite well on similar video retrieval. It is notable, however, that even for the highest dimensional SIFT-signature (11111), its performance for duplicate video retrieval is worse than that of the low-dimensional CFMT signature (36), as shown in Fig. 9(a).



(a) (b)

Figure 5. Comparing the performance of different dimensional CFMT for duplicate (a) and similarity (b) detection for bursty errors



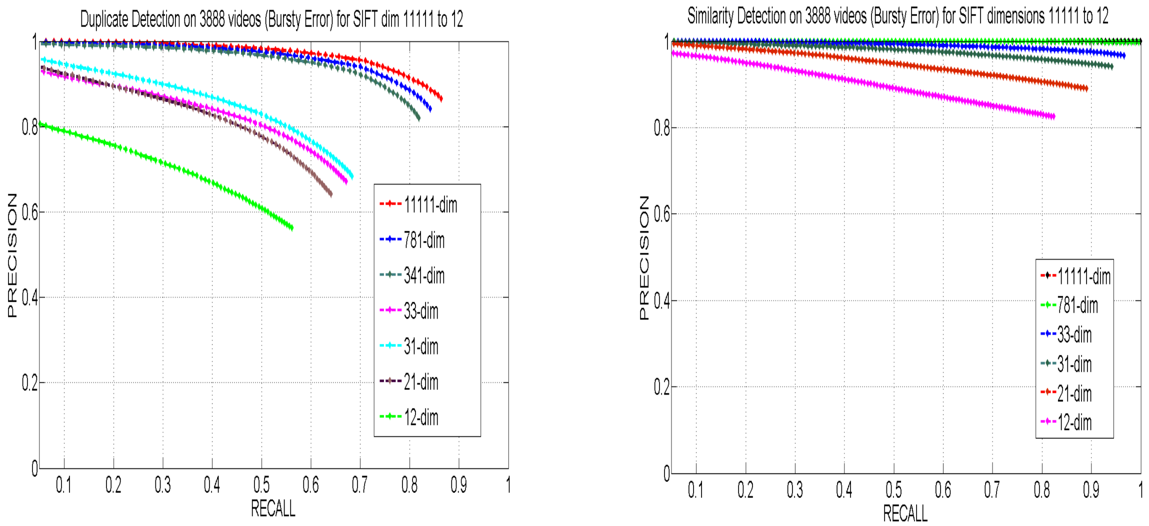
(a) (b)

Figure 6. Comparing the performance of different dimensional CFMT for duplicate (a) and similarity (b) detection for non-bursty errors

9. FULL-LENGTH VIDEO RETRIEVAL WITH CLIP QUERYING

We also study the retrieval results from a full-length video database (64 videos in the BBC rushes dataset as mentioned in Sec. 2.2) while querying with a clip of one of the videos.

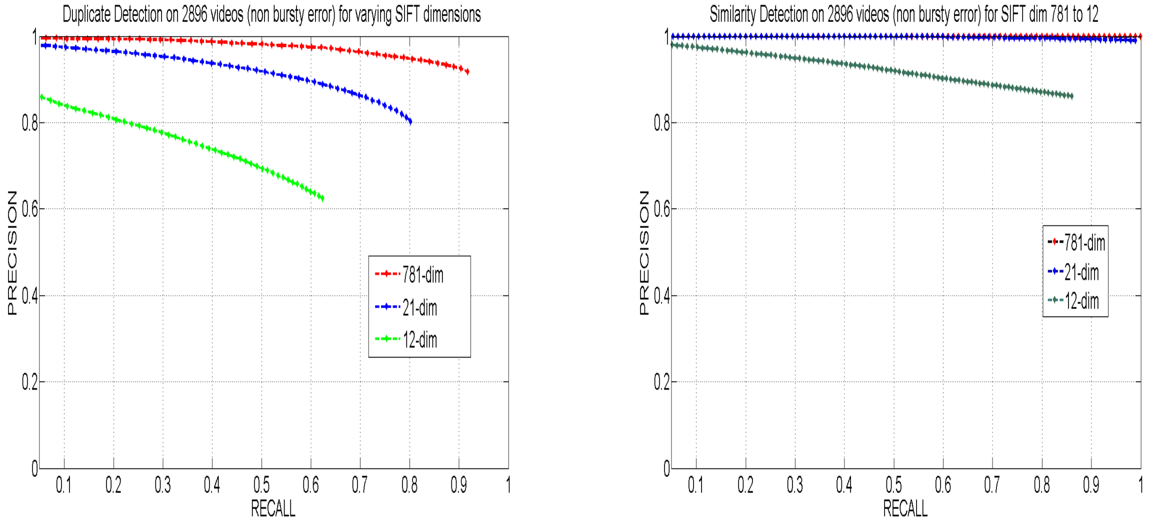
Four videos are used for querying: MRS45905, MRS147040, CU497924 and CU501380. For MRS45905, MRS147040 and CU497924, we consider queries consisting of 32 keyframes, corresponding to four distinct scenes. For CU501380, we consider queries representing 3 distinct scenes, given by 24 keyframes. Let the input query X_{query} be a $32 \times d$ signature, where each of the 32 representative frames is represented by a d-dimensional signature. While comparing this signature with that of a large video, which is represented by N_{large} keyframes and has a signature X_{large} ($N_{large} \times d$), the distance between these signatures, $\mathcal{D}(X_{query}, X_{large})$, is computed as follows:



(a)

(b)

Figure 7. Comparing the performance of different dimensional SIFT features for duplicate (a) and similarity (b) detection for bursty errors



(a)

(b)

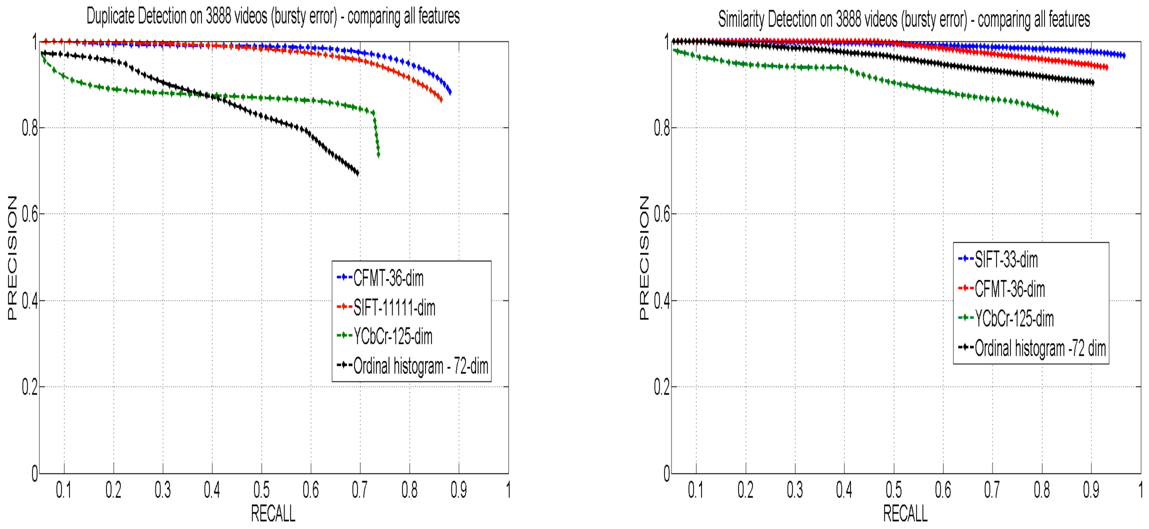
Figure 8. Comparing the performance of different dimensional SIFT features for duplicate (a) and similarity (b) detection for non-bursty errors

$$\Delta(i) = \min_{j} ||X_{query}(i) - X_{large}(j)||_1, \; 1 \leq i \leq 32 \tag{6}$$

$$\mathcal{D}(X_{query}, X_{large}) = \sum_{i=1}^{32} \Delta(i)/32 \tag{7}$$

where $||X_{query}(i) - X_{large}(j)||_1$ is the L_1 distance between the i^{th} frame for X_{query} and the j^{th} keyframe for X_{large}. We have used L_1 distance between two frame-based vectors, for CFMT, YCbCr histogram, and ordinal histogram based features.

Now, after finding the distance $\mathcal{D}(X_{query}, X_{large})$ for all the large videos, we arrange them in increasing order



(a) (b)

Figure 9. Performance of different descriptors for duplicate (a) and similarity (b) detection. **CFMT is superior for duplicate detection, and SIFT is superior for similarity detection.**

of distance and consider the position of the video, from which the clip is actually taken, in the list, which is called the "rank." In an ideal case, for perfect retrieval, the "rank" will be 1. Here also, we use the same set of image processing operations and frame losses (20, 40, and 60%), as mentioned in Sec. 2.1 on three or four distinct-scene videos, for creating the query videos. We have 12096 queries created in this manner for each of the four videos. The average rank is computed over these 12096 queries.

Another aspect to note is the summary size for the large videos. If the summary size is increased, it will be easier to find similar regions between a query and the summary of the larger video. We experiment with summary sizes of lengths 1%-4% of the entire video. The "rank" values for the ordinal histogram based feature for the videos MRS45905, MRS147040, CU497924 and CU501380 are 1.46, 15.14, 21.58 and 2.22, respectively. As seen from Tables 1 and 2, the CFMT feature provides the best retrieval rank for features of comparable dimensions.

Table 1. The retrieval results are presented for 4 videos for 1% summary lengths

video name	CFMT-36	CFMT-20	CFMT-12	YCbCr-125	SIFT-781	SIFT-31	SIFT-21
MRS45905	1	1.01	1	7.92	1.01	3.83	13.26
MRS147040	1	1.01	1	1.60	1	2.67	1.49
CU497924	1.03	1.36	1.03	1.71	1	1.00	2.15
CU501380	1	1	1	1.92	1	1	1

Table 2. The retrieval results are presented for 4 videos for 4% summary lengths

video name	CFMT-36	CFMT-20	CFMT-12	YCbCr-125	SIFT-781	SIFT-31	SIFT-21
MRS45905	1	1.09	1.23	1.78	1	2.52	3.94
MRS147040	1	1	1.06	2.11	1	1	1.45
CU497924	1	1.21	1.59	4.70	1	1.41	8.44
CU501380	1	1	1.47	1.99	1	1	1

10. ANALYSIS

The results show that while CFMT features provide quick, accurate retrieval for **duplicate** videos, SIFT features are superior for detecting **similar** videos. This analysis seems in accordance with the general practice of CFMT use for duplicate image retrieval, but SIFT for object recognition in images, where objects may be similar to each other but are not identical. The applicability of the duplicate video detection method is to identify copied or pirated videos and prevent copyright infringements, where given the original, the pirated copies (even if the copies span only a part of the original, akin to our frame-drop based scenario, or contain a temporal reordering of the scenes in the original video) can be detected. Future research will focus on the scalability of our similarity detection method to capture non-retake based similar videos (similar in content - e.g. episodes of a certain television serial).

In other analysis which is not presented here, it was revealed that perhaps strategic keyframe selection, based on efficient video summarization, is unnecessary, as the retrieval results (retrieval of large video using small clips as query) using uniform keyframe selection for full-length video signature creation rival those of strategic selection for three of four videos. Further analysis is called for in this direction, using a larger number of videos.

ACKNOWLEDGMENTS

This research is supported in part by a grant from ONR # N00014-05-1-0816.

REFERENCES

1. A. Joly, C. Frelicot, and O. Buisson, "Robust content-based video copy identification in a large reference database," in *Springer: Lecture Notes in Computer Science - 2728*, pp. 414–424, 2003.
2. A. Joly, O. Buisson, and C. Frelicot, "Statistical similarity search applied to content-based video copy detection," *International Conference on Data Engineering* , p. 1285, 2005.
3. S. Lee and C. D. Yoo, "Video fingerprinting based on centroids of gradient orientations," in *Proc. of ICASSP*, pp. II–401–404, 2006.
4. D. N. Bhat and S. K. Nayar, "Ordinal measures for image correspondence," *IEEE Transactions on Pattern Analysis and Machine Intelligence* , pp. 415–423, 1998.
5. D. N. Bhat and S. K. Nayar, "Ordinal measures for visual correspondence," *Proc. of CVPR* , pp. 351–357, 1996.
6. R. Mohan, "Video sequence matching," in *Proc. of ICASSP*, pp. 3697–3700, 1998.
7. Y. Li, J. S. Jin, and X. Zhou, "Matching commercial clips from TV streams using a unique, robust and compact signature," in *Proc. of Digital Image Computing: Technqiues and Applications*, 2005.
8. J. Yuan, L. Y. Duan, Q. Tian, S. Ranganath, and C. Xu, "Fast and robust short video clip search for copy detection," in *Springer: Lecture Notes in Computer Science - 3332*, pp. 479–488, 2004.
9. J. Yuan, L. Y. Duan, Q. Tian, and C. Xu, "Fast and robust short video clip search using an index structure," in *Proceedings of the 6th ACM SIGMM international workshop on Multimedia Information Retrieval*, pp. 61–68, 2004.
10. X. Yang, Q. Tian, and E. C. Chang, "A color fingerprint of video shot for content identification," in *Proceedings of the 12th annual ACM international conference on Multimedia Systems*, pp. 276–279, 2004.
11. W. L. Zhao, C. W. Ngo, H. K. Tan, and X. Wu, "Near-duplicate keyframe identification with interest point matching and pattern learning," *IEEE Transactions on Multimedia* **9**, pp. 1037–1048, 2007.
12. A. Gionis, P. Indyk, and R. Motwani, "Similarity search in high dimensions via hashing," in *Proc. of Int. Conf. on Very Large Data Bases*, pp. 518–529, 1999.
13. "http://ir.nist.gov/tv2007/tv7rushes/tv7.bbc.devel/." BBC rushes dataset for TRECVID video summarization task - 2007.
14. D. Casasent and D. Psaltis, "Scale invariant optical transform," *Opt.Eng.* **15**(3), pp. 258–261, 1976.
15. S. Derrode and F. Ghorbel, "Robust and efficient Fourier-Mellin transform approximations for gray-level image reconstruction and complete invariant description," *Computer Vision and Image Understanding: CVIU* **83**(1), pp. 57–78, 2001.
16. F. Ghorbel, "A complete invariant description for gray-level images by the harmonic analysis approach," in *Pattern Recognition Letters*, **15**, pp. 1043–1051, October 1994.
17. D. Lowe, "Distinctive image features from scale-invariant keypoints," in *International Journal of Computer Vision*, **20**, pp. 91–110, 2003.

18. C.-Y. Lin, M. Yu, J. A. Bloom, I. J. Cox, M. L. Miller, and Y. M. Lui, "Rotation scale and translation resilient watermarking for images," *IEEE Transaction on Image Processing* **10**, pp. 767–782, May 2001.
19. D. Zheng and J. Zhao, "LPM-based RST invariant digital image watermarking," in *IEEE CCECE 2003. Canadian Conference on Electrical and Computer Engineering, 2003.*, **3**, pp. 1951–1954, May 2003.
20. N. Gotze, S. Drue, and G. Hartmann, "Invariant object recognition with discriminant features based on local fast-fourier mellin transform," in *International Conference on Pattern Recognition*, **1**, 2000.
21. S. Raman and U. Desai, "2-D object recognition using Fourier Mellin transform and a MLP network," in *IEEE International Conference on Neural Networks 1995 Proceedings*, **4**, pp. 2154–2156, May 1995.
22. Y. Ke and R. Sukthankar, "PCA-SIFT: A more distinctive representation for local image descriptors," 2004.
23. D. Nister and H. Stewenius, "Scalable recognition with a vocabulary tree," in *Proc. of CVPR*, pp. 2161–2168, (Washington, DC, USA), 2006.