Retrieval of Video Clips with Missing Frames using Sparse Bayesian Reconstruction

Pablo Ruiz*, S. Derin Babacan†, Rafael Molina* and Aggelos K. Katsaggelos‡
*Depto. de Ciencias de la Computación e I.A., Universidad de Granada, 18071 Granada, Spain
Email: {mataran,rms}@decsai.ugr.es
†Beckman Institute for Advanced Sc. and Tech.,University of Illinois at Urbana Champaign
405 N Mathews Ave, Urbana, IL 61801, USA. Email:dbabacan@illinois.edu
‡Dept. of Electrical Engineering and Comp. Sc.,Northwestern University
2145 Sheridan Road, Evanston IL 60208, USA. Email: aggk@eecs.northwestern.edu

***Abstract*—Fast and accurate algorithms are essential for the efficient search and retrieval of the huge amount of video data that is generated for different purposes and applications every day. The interesting properties of sparse representation and the new sampling theory named Compressive Sensing (CS) constitute the core of the new approach to video representation and retrieval we are presenting in this paper to deal with the search of noisy video clips with also possibly missing frames. Once the representation (where sparsity is expected) has been chosen and the observations have been taken, the proposed approach utilizes Bayesian modeling and inference to tackle the retrieval problem. In order to speed up the inference process the use of Principal Components Analysis (PCA) to provide an alternative representation of the frames is analyzed. Experimental results validate the proposed approach to the retrieval of video clips with missing frames as well as its robustness against noise.**

I. INTRODUCTION

With the rapidly increasing growth of digital video content, there is an equally growing need for efficient techniques to analyze, search and retrieve video content. Video retrieval is a key step in many applications including copyright protection, multimedia content search, security and surveillance. Fast and accurate algorithms in all these cases are needed for efficient video retrieval.

A number of methods have been developed for video retrieval. Generally, methods identify features distinguishing video frames and employ classification, indexing and searching based on these features. Among a large number of features, commonly used ones are color histograms [1], color and motion cues [2], [3], [4], visual features and semantic labels [5], and time interval statistics [6]. Surveys and comparisons can be found in [7], [8].

After identifying the distinguishing features, the second step in video retrieval is searching based on these features. Indexing and hashing are commonly used to improve the search efficiency. In [5] geometric hashing is used to build database indices, while [3], [4], [6] used tree-based indexing. A powerful data structure for indexing is the kd-trees. In [9], video trajectories over time are indexed using kd-trees with a dimensionality reduction using PCA. Random projections instead of PCA are utilized in [10], followed by several kd-trees for indexing.

In this paper, we present a new approach to video retrieval using *sparse representation and reconstruction*. The concept of sparsity has emerged in the last decade as a powerful modeling tool with a large number of potential applications [11], [12], [13]. This has been significantly motivated by the emergence of *compressive sensing* [14], [15]. Although compressive sensing and sparse reconstruction have originally aimed at the reconstruction of an original signal, the discriminative properties of sparse representations have been successfully utilized for several applications including face recognition and image classification [16], [17].

In this work, we demonstrate that the discrimination property of sparse representations can be employed very effectively for video retrieval. Specifically, we formulate the problem of searching a query video clip in a video database as a sparse reconstruction problem. We construct the video database directly from the existing video clips, such that no sophisticated feature extraction methods are needed as preprocessing. Moreover, we present a method to handle the problem of missing frames in the query clip, and show that our method is extremely robust to the cases where a large number (almost 80%) of the frames are removed. Finally, we demonstrate with experimental results that the proposed method provides very high retrieval performance in terms of both error rate and retrieval speed.

This paper is organized as follows. In section II, we present the proposed sparse representation framework and the Bayesian reconstruction algorithm for the video retrieval problem. The searching and classification procedure is explained in Section III. The dimensionality reduction step using PCA is presented in Section IV. We analyze the retrieval performance of the proposed system and its robustness to noise and missing frames in Section V and conclude in Section VI.

II. FRAME RETRIEVAL USING SPARSE REPRESENTATION

In this section, we study how to retrieve one frame in the database using sparse representation principles.

A. Sparse representation of frames

The video database can be represented as a matrix by concatenating the existing video clips as

$$\mathbf{A} = [\mathbf{a}_{1,1}, \mathbf{a}_{1,2}, \ldots, \mathbf{a}_{1,N_1}, \ldots, \mathbf{a}_{K,1}, \ldots, \mathbf{a}_{K,N_K}] \quad (1)$$

where $\mathbf{a}_{i,j},\ i = 1, \ldots, K, j = 1, \ldots, N_i$ represents the $j-$th frame in the $i-$th video. Each $\mathbf{a}_{i,j}$ is assumed to be a column vector of size M where $M = VH$ with V and H the vertical and horizontal dimensions of each frame respectively. For notational convenience, Eq. (1) is rewritten as

$$\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_N] \quad (2)$$

where $N = \sum_{i=1}^{K} N_i$.

Let $\mathbf{Y}$ be a video-clip in the database with only one frame:

$$\mathbf{Y} = [\mathbf{y}_1] \quad (3)$$

Then, it can be observed that $\mathbf{y}_1$ admits a sparse representation as

$$\mathbf{y}_1 = \mathbf{A}\mathbf{x}_1 \quad (4)$$

where $\mathbf{x}_1 = (0, \ldots, 0, 1, 0, \ldots, 0)^t$ is a sparse vector with all coefficients equal to zero except for the entry corresponding to the location of the frame $\mathbf{y}_1$ in the database, which is equal to 1. Hence, the position of $\mathbf{y}_1$ in the database is determined by $\mathbf{x}_1$.

For a given frame $\mathbf{y}_1$, solving for the corresponding $\mathbf{x}_1$ is an ill-posed problem as the system in (4) is highly under-determined which leads to non-uniqueness of the solutions. However, $\mathbf{x}_1$ has only one non-zero component, and therefore, our goal is to find the sparsest solution, finding the solution with most components equal to zero. This motivates, following [17], to seek for the solution of

$$\hat{\mathbf{x}}_1 = \arg\min_{\mathbf{x}} \parallel \mathbf{x} \parallel_0 \quad \text{subject to} \quad \mathbf{A}\mathbf{x} = \mathbf{y}_1, \quad (5)$$

where $\parallel \mathbf{x} \parallel_0$ is the l_0-quasinorm (the number of non-zero coefficients). However, as is well known, the solution of this optimization problem is NP-hard. Furthermore, there are other issues like noise, different image sizes or even occlusions that make us resort to the CS formulation of the problem.

With $\mathbf{y} = \mathbf{y}_1$, the noisy CS acquisition system can be modeled as

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{n}, \quad (6)$$

where $\mathbf{n}$ is the $M \times 1$ independent, Gaussian, zero-mean noise vector with variance equal to β^{-1}. The problem (5) can then be relaxed using the l_1-norm formulation as

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \{\|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2 + \tau\|\mathbf{x}\|_1\}, \quad (7)$$

where $\|\cdot\|_1$ denotes the l_1-norm. Solving (7) is much easier than (5) and has attracted much interest in the CS community.

B. Frame Retrieval Based on Bayesian Compressive Sensing

A number of methods have been proposed to solve the sparse optimization problem in Eq. (7), (see [11] and references therein, see also [18]). In this paper, we formulate the problem using the Bayesian framework following [19] which will also allow us to automatically estimate the regularization parameters, (see [19], [11] for references to parameter estimation). We provide here a brief review of solving (7) using a Bayesian approach.

In Bayesian modeling, all unknowns are treated as stochastic quantities with assigned probability distributions. The joint probability distribution of all quantities is given by

$$\mathrm{p}(\mathbf{x}, \boldsymbol{\gamma}, \beta, \mathbf{y}) = \mathrm{p}(\mathbf{y}|\mathbf{x}, \beta)\, \mathrm{p}(\mathbf{x}|\boldsymbol{\gamma})\, \mathrm{p}(\boldsymbol{\gamma})\, \mathrm{p}(\beta). \quad (8)$$

The observation noise is independent and Gaussian with zero mean and variance equal to β^{-1}, that is, with (6),

$$\mathrm{p}(\mathbf{y}|\mathbf{x}, \beta) = \mathcal{N}(\mathbf{y}|\mathbf{A}\mathbf{x}, \beta^{-1}). \quad (9)$$

It is shown in [19] that the l_1 regularization formulation in (7) is equivalent to using a hierarchical Laplace prior on the coefficients of $\mathbf{x}$, that is,

$$\mathrm{p}(\mathbf{x}|\boldsymbol{\gamma}) = \prod_{i=1}^{N} \mathcal{N}(x_i|0, \gamma_i), \quad (10)$$

$$\mathrm{p}(\gamma_i|\lambda) = \frac{\lambda}{2} \exp\left(-\frac{\lambda\gamma_i}{2}\right),\ \gamma_i \geq 0,\ \lambda \geq 0, \quad (11)$$

where $\boldsymbol{\gamma} = (\gamma_1, \gamma_2, \ldots, \gamma_N)$. Using this specification, the signal distribution $\mathrm{p}(\mathbf{x}|\mathbf{y}, \lambda, \beta)$ is estimated as a multivariate Gaussian distribution $\mathcal{N}(\mathbf{x}|\eta, \Theta)$ with parameters

$$\Theta = \left[\beta\mathbf{A}^t\mathbf{A} + \Lambda\right]^{-1}, \quad (12)$$

$$\eta = \Theta\, \beta\mathbf{A}^t\mathbf{y}, \quad (13)$$

with $\Lambda = \mathrm{diag}(1/\gamma_i)$. The hyperparameters $\boldsymbol{\gamma}$ are then estimated by forming the likelihood function

$$\mathcal{L} = -\frac{1}{2}\log|\mathbf{E}| - \frac{1}{2}\mathbf{y}^t\mathbf{E}^{-1}\mathbf{y} + N\log\frac{\lambda}{2} - \frac{\lambda}{2}\sum_i \gamma_i\,, \quad (14)$$

with $\mathbf{E} = \left(\beta^{-1}\mathbf{I} + \mathbf{A}\Lambda^{-1}\mathbf{A}^t\right)$, and maximizing it with respect to each γ_i and λ in an alternating fashion. This procedure results in the updates

$$\gamma_i = -\frac{1}{2\lambda} + \sqrt{\frac{1}{4\lambda^2} + \frac{<x_i^2>}{\lambda}}\ , \quad (15)$$

$$\lambda = \frac{N-1}{\sum_i \gamma_i/2}, \quad (16)$$

where $<x_i^2> = x_i^2 + \Theta_{ii}$. In summary, at each iteration of the algorithm, given an estimate of $\boldsymbol{\gamma}$ and λ, the estimate of the distribution of $\mathbf{x}$ is calculated using (12) and (13), followed by the estimation of the variances γ_i from (15) and the hyperparameter λ from (16). In addition, [19] proposed a greedy approach that finds the solutions much more efficiently without the need of solving the large linear system in (13). In our work, we use this greedy approach to find the solution of (7).

III. Video Retrieval

A. Searching for video clips without missing frames

In [21] we utilized the CS theory to retrieve a video clip of consecutive frames. More formally, let $\mathbf{y}$ be a video-clip in the database written in the vector form as

$$\mathbf{y} = \begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \\ \vdots \\ \mathbf{y}_S \end{bmatrix} \tag{17}$$

where each $\mathbf{y}_i$ represents the $i-$th frame, and S is the length of the video clip. Next, we built the following matrix for a query of length S

$$\tilde{\mathbf{A}} = \begin{pmatrix} \mathbf{a}_1 & \mathbf{a}_2 & \mathbf{a}_3 & \dots & \mathbf{a}_{N-(S-1)} \\ \mathbf{a}_2 & \mathbf{a}_3 & \mathbf{a}_4 & \dots & \mathbf{a}_{N-S} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{a}_S & \mathbf{a}_{S+1} & \mathbf{a}_{S+2} & \dots & \mathbf{a}_N \end{pmatrix} \tag{18}$$

by shifting the columns of $\mathbf{A}$ in (1). Using this matrix, it can be observed that $\mathbf{y}$ admits a sparse representation as

$$\mathbf{y} = \tilde{\mathbf{A}}\mathbf{x}_0 \tag{19}$$

where $\mathbf{x}_0 = (0, \dots, 0, 1, 0, \dots, 0)^t$ is a sparse vector with all coefficients equal to zero except for the entry corresponding to the location of the video clip $\mathbf{y}$ in the database, which is equal to 1. Hence, the position of $\mathbf{y}$ in the database is determined by $\mathbf{x}_0$.

The noisy CS acquisition system for this problem can be modeled as

$$\mathbf{y} = \tilde{\mathbf{A}}\mathbf{x} + \mathbf{n}, \tag{20}$$

where $\mathbf{n}$ is the $(SM) \times 1$ independent, Gaussian, zero-mean noise vector with variance equal to β^{-1}. The retrieval problem is the formulated using the l_1-norm formulation as

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}}\{\|\mathbf{y} - \tilde{\mathbf{A}}\mathbf{x}\|_2^2 + \tau\|\mathbf{x}\|_1\}, \tag{21}$$

where $\|\cdot\|_1$ denotes the l_1-norm.

B. Searching for video clips with missing frames

The method proposed above works well (see [21]) when there are no missing frames in the clip we are looking for. However, when there may be missing frames and their positions in the clip are not known the above $\tilde{\mathbf{A}}$ matrix can not be built and the proposed method is then not applicable. To deal with possibly missing frames we propose the location and classification procedures that are described next.

Let

$$\mathbf{Y} = [\mathbf{y}_1 \ \mathbf{y}_2 \dots \mathbf{y}_S] \tag{22}$$

be a query video-clip in the database which may contain missing frames. The location procedure consists on finding a candidate video-clip in the database that contains $\mathbf{Y}$. Following the proposed CS methodology, we can search for each frame independently and then examine if their sparse representations correspond to frames in a video clip in the database. However, we can reduce the computational time by noticing that the presence of the query video in the database is determined by the location of the first and last frames in the database.

Therefore we start by finding the two sparse vectors $\hat{\mathbf{x}}_1$ and $\hat{\mathbf{x}}_S$ that solve:

$$\hat{\mathbf{x}}_1 = \arg\min_{\mathbf{x}}\{\|\mathbf{y}_1 - \mathbf{A}\mathbf{x}\|_2^2 + \tau\|\mathbf{x}\|_1\} \tag{23}$$

and

$$\hat{\mathbf{x}}_S = \arg\min_{\mathbf{x}}\{\|\mathbf{y}_S - \mathbf{A}\mathbf{x}\|_2^2 + \tau\|\mathbf{x}\|_1\} \tag{24}$$

respectively. Then, if $\hat{\mathbf{x}}_1$ or $\hat{\mathbf{x}}_S$ do not satisfy the following initial conditions:

1) They only have one non-zero component.
2) The position marked by $\hat{\mathbf{x}}_1$ is preceding the position marked by $\hat{\mathbf{x}}_S$.

the video clip is classified as not present in the database. If these initial conditions are satisfied, we proceed to examine if $\mathbf{Y}$ is in the database.

We define the candidate video-clip to be retrieved as:

$$\mathbf{C} = [\mathbf{c}_1 \ \mathbf{c}_2 \dots \mathbf{c}_P] \tag{25}$$

where $\mathbf{c}_i \ i = 1, \dots, P$ are the frames in the database between the positions marked by $\hat{\mathbf{x}}_1$ and $\hat{\mathbf{x}}_S$.

It is very important to note that P and S do not have to be the same. The query video can have missing intermediate frames. Other systems as [9], [10], [21] strongly utilize the fact that the query video-clip has all intermediate frames, and, as we will see in experimental section, they fail when this does not happen.

Once we have located the tentative position of the first and last frames of the video query in the database we proceed to accept or reject the candidate video clip. We assume that all frames in $\mathbf{C}$ are independent realizations of a Gaussian distribution with mean μ and covariance matrix Σ which are estimated by using

$$\mu = \frac{1}{P}\sum_{i=1}^{P}\mathbf{c}_i, \quad \Sigma = \frac{1}{P-1}\sum_{i=1}^{P}(\mathbf{c}_i - \mu)(\mathbf{c}_i - \mu)^t. \tag{26}$$

Therefore if $\mathbf{Y}$ is in the database, then $\overline{\mathbf{y}} = \frac{1}{S}\sum_{i=1}^{S}\mathbf{y}_i$ will be close to μ when using the Mahalanobis distance. We define the Classification Coefficient $(\mathbf{CC})$, fix a threshold δ and decide that $\mathbf{Y}$ is in the database if and only if:

$$\mathbf{CC}(\mathbf{Y}) \doteq \sqrt{\frac{1}{S}(\overline{\mathbf{y}} - \mu)^t \Sigma^+ (\overline{\mathbf{y}} - \mu)} \leq \delta. \tag{27}$$

where Σ^+ is the Generalized Inverse of Moore-Penrose of Σ.

IV. Feature Extraction

In order to perform an efficient search, and due to the size of the frames, feature extraction is needed. We first assume that the frames in the database are downsampled to a reasonable size. This is an important step and a compromise between the size of the downsampled images and the feature extraction process has to be reached. If the downsampled frames are too

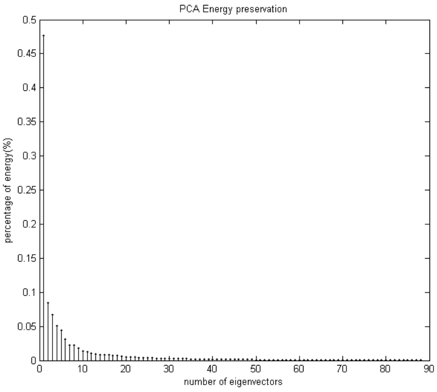

Fig. 1. Energy of the PCA components.

large, feature extraction is very time consuming, on the other hand if the downsampled images are too small, the frames can not be distinguished. See section V for the size of the downsampled images used in the experiments.

We then use a linear feature transformation. The projection from the image space to the feature space can be represented by a matrix $\mathbf{R} \in R^{T\times M}$ with $T \ll M$ which when applied to $\mathbf{A}$ produces

$$\mathbf{R}_{T\times M}\mathbf{A}_{M\times N} = \dot{\mathbf{A}}_{T\times N} \tag{28}$$

Then we can use the proposed retrieval procedure on $\dot{\mathbf{A}}$, which leads to a faster search. In this work we consider $\mathbf{R}$ to be the matrix associated to PCA, (see [9]). Notice that we could have also used a matrix of random projections $\Phi_{T\times M}$.

The PCA transformation retains much of the information in only a reduced set of principal components. The number of preserved dimensions, T, determines the energy loss during the PCA transformation. The energy represented by each PCA coefficient obtained from the test database used in the experiments, which consists of 567146 frames, is shown in Figure 1. Notice that for T=21 more than 90 % of the energy is preserved. Furthermore if the CS theorical conditions are met by $\mathbf{A}$ (see [17]), then $\hat{\mathbf{x}}$ can be recovered by l1-minimization with overwhelming probability if $T > 2\log(567146/T)$. In other words, around $T \approx 21$ would suffice to recover the only non-zero component. As we will see in the experimental section, when $T = 21$ the proposed system retrieves all the relevant clips in the database in the noiseless case.

V. Simulation Results

In our experiments we used the 2004 NIST TRECVID shot boundary test set. This data set has approximately 6 hours of video in 12 videos (each of about 30 mins long). We split it in two data sets. The positive video repository (or database) consists of 11 videos and the other video forms the negative data set.

The frames are downsampled with a scaling factor of 16 to produce 22x16 downsampled frames. Then the frames are projected using PCA transformation with $T = 21$. In our test, we select randomly 100 positive and 100 negative query videos. The query clip lengths are $S = 15, 30$, and 60 frames. All experiments were performed utilizing an Intel i7 2.8GHz notebook with 8 GB of RAM.

TABLE I
CPU time used to find a query.

Methods	15 Frames	30 Frames	60 Frames
KD-PCA [9]	0.04 s	0.06 s	0.12 s
Proposed Method	0.69 s	0.72 s	0.71 s
KD-RP [10]	2.25 s	2.50 s	4.00 s
SR-C [21]	7.57 s	22.17s	out of memory

A. Noise free and complete test cases

For noise-free test cases with all its frames our system retrieved all positive cases and rejected all negative one. The results are exactly the same as the ones reported in [9], [10] and [21].

B. Degraded test cases

In real world applications video clips can be corrupted by coding and communication losses, as well as, image formation variations. We evaluate the robustness of our system to both, noise and missing frames, using precision-recall curves. The precision-recall curve [20] is a typical way of characterizing retrieval performance. For a given threshold, let us assume that a is the number of relevant (present in the database) clips retrieved, b the number of relevant clips not retrieved, and c the number of non relevant clips retrieved. Then the precision and recall values are defined by $precision = a/(a+c)$ and $recall = a/(a+b)$. By changing the threshold value we obtain, for a given method, its precision-recall curve. Notice that as the threshold δ in Eq. (27) decreases the recall value is expected to decrease while the precision value is expected to increase.

To simulate the systems in [9], [10], [21] Gaussian noise is added to the query clip to evaluate their robustness to noise. However these systems do not consider test cases with missing frames. This is because they are not designed to retrieve videos with more frames than the query video, and therefore if the query video has missing frames, these systems can not retrieve it.

1) Noisy test cases.: We added Gaussian noise to query clips at PSNR of 25dB; this is the noisiest case considered in [9] and [10]. The comparison between our system and KD-PCA [9], KD-RP [10] and SR-C [21], when the query does not contain missing frames, is shown in Fig. 2, for the case $T = 30$. The proposed method obtains a $recall = 0.94$ and a $precision = 1$.

2) Test cases with missing frames.: We created queries with missing frames by randomly removing intermediate frames at the following percentages 20%, 50%, and 80%. For noise-free queries all positives cases are retrieved, and all negatives are rejected. For noisy videos with 80% missing frames, out of 30 and 60 frames query clips the $precision = 1$, i.e., all negatives cases are rejected, and $recall = 0.94$ and $recall = 0.96$, respectively. Finally, Fig. 3 shows the precision-recall curves

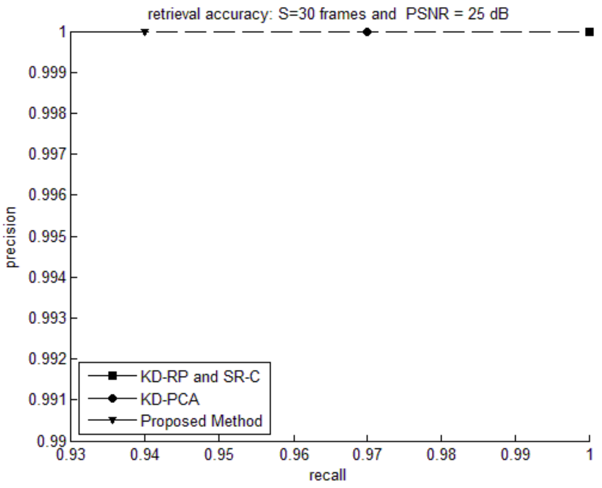

Fig. 2. Comparison of four systems of video retrieval for noisy test.

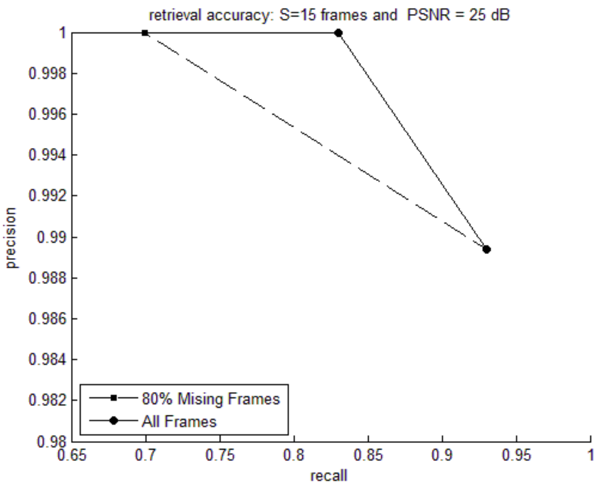

Fig. 3. Precission-Recall curves for noisy query videos with 15 frames and 80% of missing frames, and noisy query videos with all 15 frames.

for a video query of 15 frames and the same query with 80% of missing frames. The system produces more false positives when we remove frames.

VI. Conclusion

In this paper we have developed a new system for video retrieval based on the sparse representation framework. We formulate the video retrieval as a sparse reconstruction problem by constructing a database matrix and searching for the sparsest representation of a query clip using the database. We have shown that the proposed system is very effective and robust to noise and missing frames, and does not require sophisticated and data-dependent feature extraction methods. Moreover, the proposed system requires comparable and less computational resources to some of the state-of-the-art methods for video retrieval while providing very high retrieval accuracy.

Acknowledgment

This work has been supported in part by the “Consejería de Inovación, Ciencia y Empresa” of the “Junta de Andalucía” under contract P07-FMQ-02701 and the “Ministerio de Educación y Ciencia” under contract TIN2010-15137.

References

[1] A.M. Ferman, A.M. Tekalp and R. Mehrotra, ”Robust Color Histogram Descriptors for Video Segment Retrieval and Identification,” *IEEE Trans. on Image Processing*, vol. 11, no. 5, pp. 497-508, May 2002.

[2] S.-F. Chang Chen, W. Meng, H.J. Sundaram and H. Di Zhong, ”A Fully Automated Content-based Video Search Engine Supporting Spatiotemporal Queries,” *IEEE Trans. on Circuits and System for Video Technology*, vol. 8, no.5, pp. 602-615, Sept. 1998.

[3] C-T. Hsu and S.-J. Teng, ”Motion Trajectory Based Video Indexing and Retrieval,” *Proc. International Conference on Image Processing*, vol. 1, pp. I-605 - I-608, 2002.

[4] V. Mezaris, I. Kompatsiaris, N. V. Boulgouris and M. G. Strintzis, ”Real-time Compressed-domain Spatiotemporal Segmentation and Ontologies for Video Indexing and Retrieval,” *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 14, no. 5, pp. 606-621, May 2004.

[5] J. Fan, A.K. Elmagarmid, X. Zhu, W.G. Aref and L. Wu, ”ClassView: Hierarchical Video Shot Classification, Indexing, And Accessing,” *IEEE Trans. on Multimedia*, vol. 6, no. 1, pp. 70-86, Feb. 2004.

[6] C.G.M. Snoek and M. Worring, ”Multimedia event-based video indexing using time intervals,” *IEEE Trans. on Multimedia*, vol. 7, no. 4, pp. 638-647, Aug. 2005.

[7] A. Joly, O. Buisson and C. Frelicot, ”Content-based Copy Retrieval using Distortion-based Probabilistic Similarity Search,” *IEEE Trans. on Multimedia*, vol. 9, no. 2, pp. 293-306, Feb. 2007.

[8] J. Law-to, O. Buisson, L. Chen, M. H. Ipswich, V. Gouet-brunet, A. Joly, N. Boujemaa, I. Laptev, F. Stentiford and M. H. Ipswich, ”Video copy detection: a comparative study,” *ACM Int. Conf. on Image and Video Retrieval (CIVR)*, pp. 371-378, 2007.

[9] L. Gao, Z. Li and A.K. Katsaggelos, ”Fast Video Shot Retrieval with Luminance Field Trace Indexing and Geometry Matching,”*Proc. IEEE Int. Conf. on Image Processing (ICIP)*, pp. 1497-1500, 8-11, Oct. 2006.

[10] L. Gao, Z. Li and A. K. Katsaggelos, ”A Video Retrieval Algorithm using Random Projections,” *Proc. IEEE Int. Conf. on Image Processing (ICIP)*, pp. 797-800, 7-10 Nov. 2009.

[11] J.L. Starck, F. Murtagh, and J. Fadili, *Sparse Image and Signal Processing: Wavelets, Curvelets, Morphological Diversity*, Cambridge University Press, Cambridge, 2010.

[12] M. Elad, M.A.T. Figueiredo, and M. Yi, ”On the Role of Sparse and Redundant Representations in Image Processing,” *Proceedings of IEEE*, vol. 6, 972 - 982, June 2010.

[13] J. Wright, M. Yi, J. Mairal, G. Sapiro, T.S. Huang, and Y. Yan, ”Sparse Representation for Computer Vision and Pattern Recognition,” *Proceedings of IEEE*, vol. 6, 1031 - 1044, June 2010.

[14] E. J. Candès and T. Tao, ”Near-optimal signal recovery from random projections: Universal encoding strategies?,” *IEEE Trans. Inform. Theory*, vol. 52, 5406-5425, 2006.

[15] D. Donoho, ”Compressed sensing,” *IEEE Trans. on Information Theory*, vol. 52, 1289 - 1306, 2006.

[16] J. Mairal, F. Bach, J. Ponce, G. Sapiro and A. Zisserman, ”Discriminative Learned Dictionaries for Local Image Analysis,” *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, 1-8, 2008.

[17] J. Wright, A.Y. Yang, A. Ganesh, S.S. Sastry and Y. Ma, "Robust face recognition via sparse representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, pp. 210 - 227, Feb. 2009.
[18] S.D. Babacan, R. Molina and A.K. Katsaggelos, "Parameter Estimation in TV Image Restoration Using Variational Distribution Approximation," *IEEE Transactions on Image Processing*, vol. 17, pp. 326 - 339, Feb. 2008.
[19] S.D. Babacan, R. Molina and A.K. Katsaggelos, "Bayesian Compressive Sensing Using Laplace Priors," *IEEE Trans. on Image Processing*, vol. 19, no. 1, pp. 53-63, Jan. 2010.
[20] V. Raghavan, P. Bollmann, and G. S. Jung, "A critical investigation of recall and precision as measures of retrieval system performance," *ACM Trans. on Information Systems, vol. 7, 1989, pp. 205-229*.
[21] P. Ruiz, S.D. Babacan, L.Gao, Z. Li, R. Molina and A.K. Katsaggelos, "Video Retrieval Using Sparse Bayesian Reconstruction," *IEEE Int. Conf. on Multimedia and Expo (ICME), Barcelona (Spain), July 2011.*(Accepted for publication.)