ROBUST AND COMPACT VIDEO DESCRIPTOR LEARNED BY DEEP NEURAL NETWORK

Yue nan Li, Xue piao Chen

School of Electronics and Information Engineering, Tianjin University, China

ABSTRACT

In this paper, we propose to extract robust video descriptor by training deep neural network to automatically capture the intrinsic visual characteristics of digital video. More specifically, we first train a conditional generative model to capture the spatio-temporal correlations among visual contents and represent them as an intermediate descriptor. A nonlinear encoder, with the functions of dimension reduction and error correcting, is then trained to learn a compressed yet more robust representation of the intermediate descriptor. The cascade of the conditional generative model and the encoder constitutes the building block of the deep network for learning video descriptor. As a post-processing component, the top layers of the network are trained to optimize the robustness and discriminative capability of the output descriptor. Experimental results on benchmark databases confirm that the descriptor learned by deep neural network shows excellent robustness against photometric, geometric, temporal and combined distortions, and it can attain an F_1 score of 0.982 in content identification, which is much higher than hand-engineered descriptors.

Index Terms— Video content identification, Video fingerprinting, Video hashing, Deep neural network.

1. INTRODUCTION

The upsurge of content sharing web sites during the past few years has resulted in a rapid growth of digital video, and video is dominating the Internet traffic. In spite of the abundance of video resource, without effective content identification technique, we may not able to see an increased accessibility of video information. Video content identification, which aims at searching the exact copies and transformed versions of a specific piece of video content, is the enabling technique for video indexing, retrieval, tracing, etc. In particular, as one of the most effective alternatives to digital watermarking, content identification plays a fundamental role in digital right management, without of which copyright-compliant content sharing may remain a vision. Moreover, some novel applications of video content identification are being explored by the multimedia industry, hoping to prompt user experience or foster more value-added services. A central problem in content identification is to seek a concise description of visual contents, and a good video descriptor should have the following properties:

- Robust: Video descriptor should be insensitive to video processing operations and intentional manipulations.
- Discriminative: The descriptors extracted from visually irrelevant videos should exhibit distinct difference.
- Compact: For the ease of storage and database searching, video descriptor should have the least amount of redundancy.
- Computationally efficient: Video descriptor should be easy-to-compute.

The process of computing video descriptor is coined as video fingerprinting or video hashing in the literature [1]. The most straightforward approach for generating video descriptor is to concatenate the descriptors independently extracted from representative frames. However, directly extending image descriptors, such as statistics [2], gradient of intensities [3] and chromatic correlation [4], to video may suffer from high redundancy and the sensitivity to temporal distortions. A more effective approach for generating video descriptor is to characterize both the spatial and temporal variations of visual contents. For example, some video descriptors are generated by encoding the intensity differences between spatially and temporally nearby blocks [5] and the trajectory of key points along the time axis [6]. Moreover, three-dimensional transform [7], tensor decomposition [8] and optical flow [9], were also explored to extract spatio-temporal visual features. Systematic reviews on video descriptors can be found in [1] and [10].

Most existing video fingerprinting algorithms are manually designed. However, making an informative and compact representation of the rich visual contents of video requires tremendous expert knowledge. Moreover, it is very difficult, if not impossible, for hand-engineered descriptors to capture abstract and high-level visual characteristics. Recently, learning based approaches have shown great potentials in large-scale indexing and searching [11]. In this paper, we propose a data-driven algorithm that uses deep neural network to learn descriptor from raw video. The proposed work breaks down the task of learning video descriptor into a number of sub-problems and train neural networks to tackle each of them,

This work was supported by the National Natural Science Foundation of China under Grants 61572352 and 61202164, Tianjin Research Program of Application Foundation and Advanced Technology under Grant No.14JCQNJC01500.

including capturing the spatio-temporal correlations among visual information, making compact and invariant representation of learned features, and balancing the robustness and discriminative capability of the final descriptor. The comparative experiments over public databases demonstrate that the proposed data-driven descriptor can achieve more accurate content identification than hand-engineered ones.

The reminder of this paper is organized as follows. Section 2 describes the architecture of the deep neural network and the training algorithm. Experimental results are presented in Section 3, and conclusions are summarized in Section 4.

2. METHOD

2.1. Modeling spatio-temporal correlations

The correlations among neighboring pixels and successive frames reflect the local structures in each frame and their time dynamics, which is a kind of most discriminative and stable visual characteristic. However, it is very challenging to capture and represent such abstract visual characteristic. To tackle this challenge, we take the Conditional Restricted Boltzmann Machine (CRBM) [12] as one of the key components for constructing the deep feature-learning network. CRBM can simultaneously model the statistical correlations of visual information along the spatial and temporal directions. More specifically, the spatial correlations among pixels are modeled by the connection between the visible and hidden layers at the same time instant, while the temporal ones among successive frames are modeled by the connections between the layers at different time steps. Let us denote the N_V-dimensional vectorized representation of the t-th frame as $\boldsymbol{v}_t \in \mathbb{R}^{N_V}$, and it is set as the visible layer at time t, as shown in Fig.1. CRBM uses an energy function to define the joint probability of visible and hidden units conditioned on m past frames,

$$p(\boldsymbol{v}_t, \boldsymbol{h}_t | \boldsymbol{v}_{t-1}, \cdots, \boldsymbol{v}_{t-m}) = \exp(-E(\boldsymbol{v}_t, \boldsymbol{h}_t))/Z, \quad (1)$$

where $\boldsymbol{h}_t \in \{0,1\}^{N_H} (N_H < N_V)$ represents the states of hidden units, and $Z = \sum_{\boldsymbol{h}_t} \sum_{\boldsymbol{v}_t} \exp(-E(\boldsymbol{v}_t, \boldsymbol{h}_t))$ is the partition function. The energy in (1) is defined as

$$\begin{aligned} E(\boldsymbol{v}_t, \boldsymbol{h}_t) = \frac{1}{2}\|\boldsymbol{v}_t - (\sum_{k=1}^{m} \boldsymbol{A}_k \boldsymbol{v}_{t-k} + \boldsymbol{a})\|_2^2 \quad (2) \\ -(\sum_{k=1}^{m} \boldsymbol{B}_k \boldsymbol{v}_{t-k} + \boldsymbol{b})^T \boldsymbol{h}_t - \boldsymbol{v}_t^T \boldsymbol{W} \boldsymbol{h}_t. \end{aligned}$$

As illustrated in Fig.1, $\boldsymbol{W} \in \mathbb{R}^{N_V \times N_H}$, $\boldsymbol{A}_k \in \mathbb{R}^{N_V \times N_V}$ and $\boldsymbol{B}_k \in \mathbb{R}^{N_H \times N_V}$ $(k = 1, \cdots, m)$ are weight matrices, $\boldsymbol{a} \in \mathbb{R}^{N_V}$ and $\boldsymbol{b} \in \mathbb{R}^{N_H}$ are the bias terms associated with the visible and hidden layers, respectively. It is obvious from Fig.1 that $\boldsymbol{v}_{t-1}, \cdots, \boldsymbol{v}_{t-m}$ provide dynamic biases to $\boldsymbol{v}_t$ and

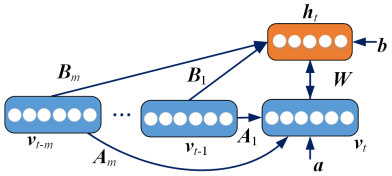

Fig. 1. Architecture of CRBM.

$\boldsymbol{h}_t$. According to (1) and (2), we can derive that the conditional distribution of the j-th $(j = 1, \cdots, N_V)$ element of $\boldsymbol{v}_t$ given $\boldsymbol{h}_t, \boldsymbol{v}_{t-1}, \cdots, \boldsymbol{v}_{t-m}$ follows a Gaussian distribution:

$$\begin{aligned} & p([\boldsymbol{v}_t]_j | \boldsymbol{h}_t, \boldsymbol{v}_{t-1}, \cdots, \boldsymbol{v}_{t-m}) \quad (3) \\ & = \mathcal{N}([\boldsymbol{a} + \sum_{k=1}^{m} \boldsymbol{A}_k \boldsymbol{v}_{t-k} + \boldsymbol{W} \boldsymbol{h}_t]_j, 1), \end{aligned}$$

where $[\cdot]_j$ represents the j-th element of a vector. The conditional distribution of the state of each hidden unit can be computed as

$$p([\boldsymbol{h}_t]_j = 1 | \boldsymbol{v}_t, \cdots, \boldsymbol{v}_{t-m}) = f\left(\left[\sum_{k=1}^{m} \boldsymbol{B}_k \boldsymbol{v}_{t-k} + \boldsymbol{b} + \boldsymbol{W}^T \boldsymbol{v}_t\right]_j\right) \quad (4)$$

where $f(z) = 1/(1+\exp(-z))$ is the sigmoid activation function. The criterion for training the CRBM is to learn a set of parameters $\{\boldsymbol{W}, \boldsymbol{a}, \boldsymbol{b}, \boldsymbol{A}_k, \boldsymbol{B}_k\}$ $(k = 1, \cdots, m)$ that can minimize the following negative log-likelihood.

$$\begin{aligned} \mathcal{L}_{\text{CRBM}} &= -\ln p(\boldsymbol{v}_t | \boldsymbol{v}_{t-1}, \cdots, \boldsymbol{v}_{t-m}) \quad (5) \\ &= -\ln \sum_{\boldsymbol{h}_t} \exp(-E(\boldsymbol{v}_t, \boldsymbol{h}_t)) + \ln \sum_{\boldsymbol{h}_t} \sum_{\boldsymbol{v}_t} \exp(-E(\boldsymbol{v}_t, \boldsymbol{h}_t)). \end{aligned}$$

We minimize $\mathcal{L}_{\text{CRBM}}$ via stochastic gradient descent. For the sake of computational efficiency, the gradient of the second term in (5) are approximated via Gibbs sampling, as in training the traditional RBM. The proposed algorithm takes the activation of hidden units as the N_H-dimensional intermediate descriptor of each frame. From (4), we see that the activation of the j-th hidden unit equals to the conditional probability that it is in the 'on' state.

2.2. Compressing intermediate descriptor

It is worth noting that the above training procedure does not explicitly concern the robustness of the descriptor. Moreover, a two-layer CRBM is not able to aggregate the input video to a compact descriptor. As a result, another neural network is trained to complement the CRBM. We desire that the second neural network can simultaneously reduce the redundancy in the intermediate descriptor and discover the informa-

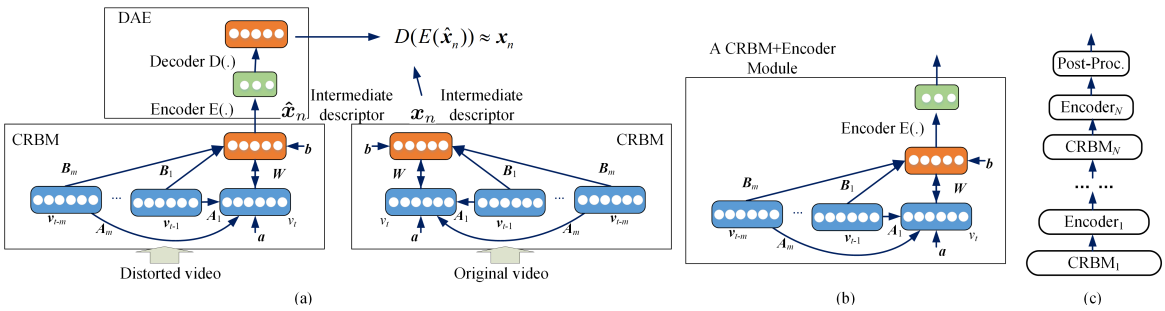

Fig. 2. The process of constructing the deep neural network for learning video descriptor. (a) train a pair of CRBM and denoising auto-encoder, (b) stack the encoder on top of the CRBM to form a CRBM-Encoder module, (c) after training N CRBM-Encoder modules, train a post-processing network and stack it on top of them.

tion that is invariant to distortions. To this end, we train a denoising auto-encoder [13] using pairs of intermediate descriptors extracted from original and distorted videos, as shown in Fig.2(a). The input-hidden and hidden-output connections of the DAE form a encoder $E(\cdot)$ and a decoder $D(\cdot)$, respectively, and the size of the hidden layer is much smaller than that of the input layer. Denote the n-th training example by $(\boldsymbol{x}_n, \hat{\boldsymbol{x}}_n)$, where $\boldsymbol{x}_n$ is the descriptor of an original training video and $\hat{\boldsymbol{x}}_n$ is its distorted version. The objective for training the DAE is to find a compressed representation of $\hat{\boldsymbol{x}}_n$ from which $\boldsymbol{x}_n$ can be recovered:

$$\mathcal{L}_{\text{DAE}} = \frac{1}{2}\sum_n \|D(E(\hat{\boldsymbol{x}}_n)) - \boldsymbol{x}_n\|_2^2 + \frac{\lambda_{\text{DAE}}}{2}\sum_{l=1}^{2}(W_{i,j}^{(l)})^2, \quad (6)$$

Given the weight matrices and bias terms $\{\boldsymbol{W}^{(l)}, \boldsymbol{b}^{(l)} | l = 1, 2\}$, the mapping defined by the encoder and the decoder can be expressed as $E(\boldsymbol{x}) = f(\boldsymbol{W}^{(1)}\boldsymbol{x} + \boldsymbol{b}^{(1)})$, $D(E(\boldsymbol{x})) = f(\boldsymbol{W}^{(2)}E(\boldsymbol{x}) + \boldsymbol{b}^{(2)})$, respectively. It is obvious from (6) that the encoder $E(\cdot)$ has the capabilities of dimensionality reduction and error correcting; hence, the input-to-hidden connections of the DAE is stacked on top of the CRBM, as Fig.2(b) shows. After training a pair of CRBM and encoder, we freeze their parameters, and the outputs of the encoder are collected for training the next pair. By sequentially training multiple CRBM-Encoder modules and concatenating them together, we get a deep neural network.

2.3. Balancing robustness and discriminative capability

To make accurate content identification, video descriptor should exhibit a good balance among robustness and discriminative capability. The aforementioned feature learning network is trained to encode the spatio-temporal visual characteristics of video to a compact descriptor, but the tradeoff between robustness and discriminative capability is not addressed. We finally train a double-layer network to refine the descriptor generated by the neural network formed by CRBM+Encoder modules, aiming to strike the optimal balance between robustness and discriminative capability. The post-processing network is trained using pair-wise descriptors: $(\boldsymbol{v}_{n,1}, \boldsymbol{v}_{n,2}, y_n)$, where n is index of the example, $\boldsymbol{v}_{n,1}$ and $\boldsymbol{v}_{n,2}$ are the intermediate descriptors of two training videos output by the top CRBM+Encoder module, and y_n is the label ($y_n = +1$ for perceptually similar pair, and $y_n = -1$ otherwise). Let $\phi(\cdot)$ be the mapping defined by the post-processing network, we define the following cost function:

$$\mathcal{L}_{\text{Post}} = \frac{1}{2}\sum_n y_n \|\phi(\boldsymbol{v}_{n,1}) - \phi(\boldsymbol{v}_{n,2})\|_2^2 + \frac{\lambda_{\text{Post}}}{2}\sum_{l=1}^{2}(W_{i,j}^{(l)})^2. \quad (7)$$

We minimize (7) using the stochastic gradient descent based back-propagation.

After training a series of CRBM-Encoder modules and the post-processing layers, we stack them together to form the descriptor-learning network. Take the network with N modules for example (as illustrated in Fig.2(c)), its architecture can be expressed as: CRBM$_1$ $\rightarrow$ Encoder$_1$ $\rightarrow \cdots \rightarrow$ CRBM$_N$ $\rightarrow$ Encoder$_N$ $\rightarrow$ Post-Processing, where each subnetwork is trained using the output of the one immediately follows it.

3. EXPERIMENTAL RESULTS

The performance of the deep learning based video fingerprinting algorithm was evaluated by content identification experiments, and our experiments were conducted on two public benchmark databases: the Youtube test set (600 sequences) [16] and the TRECVID test set (201 sequences) [17]. Nine kinds of photometric, geometric, temporal and joint spatio-temporal manipulations were applied on testing sequences, resulting in 13,617 distorted copies (depends on the number of distortion parameters). Table.1 presents the detailed information of each distortion. All the distortions were implemented using Matlab except compression. By taking every original and distorted video as query, 14,418 rounds of content identification were carried out.

Table 1. CONTENT-PRESERVING DISTORTIONS

Distortion	Description
Compression	Encoder: XVid, frame rate: 25fps, bit rate: 256kbps, fixed resolution: 480×320
Median Filtering	Filter size $\in [10, 20]$
Gaussian Noise	Zero mean, variance $\in [0.1, 0.5, 1]$
Rotation+Cropping	$\theta \in [2, 5, 10]$
Histogram Equalization	Number of gray levels $\in [16, 32, 64]$
Frame Dropping	Delete 25% frames and then linearly interpolate
Frame Resizing	Ratio $\in [0.2, 4]$
Joint Spatio-temporal Distortion 1	Combine median filtering (filter size=10) and frame dropping (25%)
Joint Spatio-temporal Distortion 2	Combine rotation ($\theta = 5$) and frame dropping (25%)

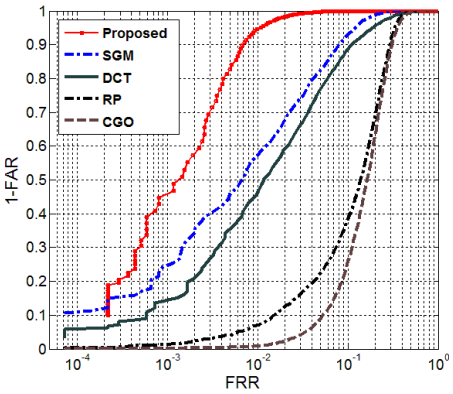

Fig. 3. Comparisons on ROC curves.

The deep neural network was trained using the Hollywood2 Human Actions and Scenes data set [18], and the training sequences were not included in the test set. The architecture of the network is 1024-300-100-80-50-40-30, which can be decomposed into two CRBM+Encoder modules:(1024-300-100),(100-80-50), and a post-processing module (50-40-30). The orders of the first and the second CRBM were set to $m = 3$, and λ_{DAE}=$\lambda_{\text{Post}} = 10^{-5}$. Before feeding a video into the neural network, we first smoothed it via low-pass filtering and temporal averaging, and then normalized it to $32 \times 32 \times 20$. The network maps each normalized frame to a descriptor of length 30.

Table 2. COMPARISONS ON THE F_1 SCORES

Dist.	Proposed	CGO	SGM	DCT	RP
Comp.	0.907	0.801	0.929	0.889	0.845
Flt.	0.994	0.887	0.978	0.991	0.966
Noise	0.999	0.813	0.890	0.896	0.624
Rot.	0.988	0.534	0.939	0.838	0.950
Hist. Eq.	0.980	0.839	0.836	0.851	0.519
Drop.	0.999	0.996	0.994	0.999	0.999
Res.	0.999	0.903	0.994	0.998	0.948
Joint Dist. 1	0.996	0.941	0.985	0.997	0.981
Joint Dist. 2	0.993	0.516	0.968	0.888	0.965
Overall	0.982	0.783	0.915	0.894	0.793

The proposed hashing algorithm was compared with four representative video fingerprinting algorithms: the structural graphical model based (SGM) [14], the 3D discrete cosine transform based (DCT) [15], the radial projection based (RP) [2], and the centroids-of-gradient-orientations based (CGO) [3], among which SGM and DCT are spatial-temporal algorithms, and the other two are key-frame based. All the comparative algorithms were implemented using the source codes provided in [16], and input sequences were normalized to fixed size (SGM, DCT and RP: $75 \times 75 \times 20$; CGO: $120 \times 120 \times 20$). In our experiments, descriptors were computed for the first 500 frames of each testing sequence, and two sequences are classified as being perceptually similar if the Euclidean distance between their descriptors is smaller than a threshold τ. The results of content identification were compared with ground-truth to compute the false rejection and false acceptance rates (FRR and FAR). By sweeping τ in a wide range, we compute the F_1 score and plot the receiver operating characteristic (ROC) curves, as displayed in Table 2 and Fig.3, respectively. We see that the proposed work achieves the best performance in content identification, and it is the only one whose F_1 scores are higher than 0.9 in all cases. Compared with other testing algorithms, the most distinct feature of the proposed one lies in its capability of capturing the statistical correlations of visual information. As a result, the output descriptor can resist a wider spectrum of distortions. Take histogram equalization for example, the F_1 score of the proposed algorithm far surpasses that of the second best. Histogram equalization, especially the one with quit few bins, can dramatically change pixel intensities. However, it does not alter the statistical correlations among pixels, which can account for the fact that some geometrically dominant local structures are still clearly visible after histogram equalization. Since the deep neural network is trained to learn an invariant representation of such abstract visual attributes, the learned descriptor can exhibit much higher robustness than those hand-engineered ones.

The proposed deep learning based video descriptor is also computationally efficient. As measured on a PC equipped with 32G RAM and a 3.2GHz GPU, computing the descriptor of a 500-frame video sequence takes 1.52s on average.

4. CONCLUSIONS

We have presented a novel data-driven algorithm for computing video descriptor, where visual features are learned by modeling the time dynamics and spatial correlations of video. Compared with several representative algorithms, we have found our data-driven approach shows substantial advantage in terms of content identification accuracy. Our future work will focus on developing a fine-tuning algorithm that to a top-down optimization of the whole network.

5. REFERENCES

[1] J. Lu, "Video fingerprinting for copy identification: from research to industry applications," in *Proc. SPIE Media Forensics and Security*, Feb. 2009, vol.7254, pp.1–15.

[2] C. D. Roover, C. D. Vleeschouwer, F. Lefèbvre, 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.

[3] S. Lee and C. D. Yoo, "Robust video fingerprinting for content-based video identification," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 7, pp. 983–988, Jul. 2008.

[4] Y. Lei, W. Luo, Y. Wang and J. Huang, "Video sequence matching based on the invariance of color correlation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 9, pp. 1332–1343, Sept. 2012.

[5] J. C. Oostveen, T. Kalker, and J. Haitsma, "Visual hashing of digital video: applications and techniques," in *Proc. SPIE Applications of Digital Image Processing XXIV*, July 2001, vol.4472, pp.121–131.

[6] S. Satoh,M. Takimoto, and J. Adachi, "Scene duplicate detection from videos based on trajectories of feature points," in *Proc. Int. Workshop on Multimedia Information Retrieval*, 2007, 237–244.

[7] B. Coskun, B. Sankur, and N. Memon, "Spatio-temporal transform based video hashing," *IEEE Trans. Multimedia*, vol.8, no.6, pp.1190–1208, Dec. 2006.

[8] M. Li and V. Monga, "Robust video hashing via multilinear subspace projections," *IEEE Trans. Image Process.*, vol.21, no.10, pp.4397–4409, Oct. 2012.

[9] M. Li and V. Monga, "Twofold video hashing with automatic synchronization," *IEEE Trans. Inf. Forens. Sec.*, vol. 10, no. 8, pp. 1727–1738, Aug. 2015.

[10] J. Liu, Z. Huang, H. Cai, H. T. Shen, C. W. Ngo, and W. Wang, "Near-duplicate video retrieval: current research and future trends," *ACM Comput. Surv.*, vol.45, no.4, pp.44:1–23, Aug. 2013.

[11] J. Wang, W. Liu, S. Kumar and S. F. Chang, "Learning to hash for indexing big data–a survey," *Proc. IEEE*, vol. 104, no. 1, pp. 34–57, Jan. 2016.

[12] G. W. Taylor, G. E. Hinton, and S. T. Roweis, "Modeling human motion using binary latent variables," in *Proc. Advances in Neural Information Processing Systems*, 2007, vol.19.

[13] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, P. A. Manzagol,"Stacked denoising autoencoders: learning useful representations in a deep network with a local denoising criterion," *J Mach. Learn. Res.*, vol.11, pp.3371–3408, Dec. 2010.

[14] M. Li and V. Monga, "Compact video fingerprinting via structural graphical models," *IEEE Trans. Inf. Forens. Sec.*, vol. 8, no. 11, pp. 1709–1721, Nov. 2013.

[15] M. M. Esmaeili, M. Fatourechi, and R. K. Ward, "A robust and fast video copy detection system using content-based fingerprinting," *IEEE Trans. Inf. Forens. Sec.*, vol.6, no.1, pp.213–226, Mar. 2011.

[16] Test set and Matlab codes of video fingerprinting algorithms [Online]. Available: http://signal.ee.psu.edu/SGMVideoHashing.htm.

[17] TRECVID test set [Online]. Available: http://www-nlpir.nist.gov/projects/tv2011/pastdata/copy.detection/201/.

[18] Hollywood2 human actions and scenes data set [Online]. Available: http://www.di.ens.fr/ laptev/actions/hollywood2/.