2017 IEEE 11th International Symposium on Embedded Multicore/Many-core Systems-on-Chip

Key Frame Extraction from Video Based on Determinant-Type of Sparse Measure and DC Programming

Yujie Li*, Benying Tan†, Shuxue Ding†, Incheon Paik† and Atsunori Kanemura*‡
*National Institute of Advanced Industrial Science and Technology (AIST), Tsukuba, Japan
{yujie-li, atsu-kan}@aist.go.jp
†The University of Aizu, Aizu-Wakamatsu, Fukushima, Japan
{m5201105, sding, paikic}@u-aizu.ac.jp
‡Advanced Telecommunications Research Institute International (ATR), Kyoto, Japan

***Abstract*—Video is human's favorite multimedia data type due to its abundant amount of information and intuitive experience compared with text, audio, and image. With rapid progress of computer and network technologies, the amount of video data increases fast, massive storage and frequent retrieval inevitably lead to huge spatio-temporal cost, and how to manage the massive video data efficiently becomes a challenging issue. Key frame extraction is considered as one of the most critical issues in video processing. In this paper, we introduce a novel key frame extraction method based on sparse modeling. Assume that each video frame signal can be expressed as a linear combination of the representative key frames and formulate the problem of finding the representatives as a sparse vector problem. We consider a sparsity measure that is based on the determinant of the Gram matrix of the signals. Based on this measure, we propose a novel key frame extraction formulation based on sparse modeling with the determinant measure of sparsity. The formulation can be expressed as the difference of two convex functions, making the objective function neither convex nor concave. Thus the formulation cannot be easily solved by standard convex optimization methods. Difference of convex (DC) programming is introduced to solve the optimization problem.**

***Index Terms*—Sparse representation, key frame extraction, determinant, difference of convex (DC) programming.**

I. Introduction

With the rapid progress of computer and network technologies, the amount of videos has been growing at an exponential rate; the need for easier video browsing has increased considerably. How to manage, classify, and retrieve the massive video data efficiently has become a challenging issue in the field of multimedia management, and then content-based video retrieval technology (CBVR) appears. Figure 1 shows typical four layers of video structure; a corresponding CBVR system includes several technologies, such as shot boundary detection, video summary, video abstract, scene analysis. Among these technologies, key frame extraction plays a critical role in video indexing, query, and browse. Key frame extraction can be expressed as a connection between shot detection and advanced semantic information acquisition of the video. Because of these importance, key frame extraction has received more and more attentions in recent years [1].

Key frames are a subset of the frames extracted from different video shots, and can be theoretically defined as the frames with the most informative and representative features that reflect the most visual contents in a video. According to this definition, the purpose of a key frame extraction algorithm is to select correct and proper key frames from whole frames of each video. Key frame extraction can compress the amount of video storage, and decrease the video indexing complexity. Through key frames extracted from the video, the work of quick browse and query would speed up and simplify.

For an application like video summarization, we gather key frames of interest based on video features, and then the key frames are used to extract and summarize information content in the video. The result of generating such video summaries can range from just a collection of key frames, which represent the essence of a video to generate a video clip summarizing the essential content of the video with temporal order in tact [2]. Another technology, shot detection based key frame selection will yield a summary that maintains temporal order but at the cost of increased redundancy.

A. Contributions

In our research, we focus on applying a sparse model on key frame extraction for videos. We introduce a determinant measure of sparsity to design sparse selection from the video, which is differentiable and can efficiently enforce sparsity. The key frame extraction problem formulation can be expressed as the approximation error subtracted by the determinant sparse measure. Since the problem formulation is uncertainly convex or concave, it cannot be solved by general convex optimization methods. Thus we introduce a difference of convex (DC) programming approach to solve the optimization problem of the difference of two convex functions. By solving the determinant measure problem, we can obtain sparse coefficients.

This study was supported in part by NEDO, Japan, JST CREST JPMJCR15E2, JST SICORP, and JSPS KAKENHI 26730130.

978-1-5386-3441-7/17 $31.00 © 2017 IEEE
DOI 10.1109/MCSoC.2017.8

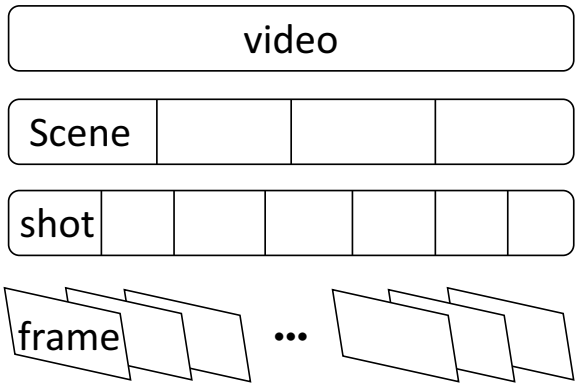

Fig. 1. Four layers of video structure.

The indices of key frames can be obtained by the rows of nonzeros in the sparse coefficient matrix.

In summary, firstly, we propose a novel key frame extraction method based on sparse representation with a determinant measure. Secondly, we introduce a DC programming approach to solve the proposed optimization problem of the key frame extraction problem.

B. Related Works

Over the past decade, researchers in computer graphics, computer vision, and robotics have worked with very large collections of data to model human behavior. We need to deal with massive collections of data, such as databases of images, videos, and text documents [3]. Compared with text, audio, and image, video is our favorite multimedia data type due to the abundant amount of information and intuitive experience. Sparse representation is broadly adopted for video processing and computer vision, e.g., face recognition, abnormal event detection [4], object tracking [5], image denoising, and video summarization [6]–[8]. Several novel and improved key frame extraction methods are presented recently. N. Ejaz et al. [9] proposed an aggregation mechanism to combine the visual features extracted from the correlation of RGB color channels, the color histogram, and the moments of inertia to extract key frames [10]. In 2012, X. Liu et al. [11] proposed a method based on maximum a posteriori (MAP) estimation of the positions of key frames. In the same year, E. Elhamifar et al. [12] have proposed sparse modeling representation selection (SMRS) to video processing, which has been shown to be efficient for summarization and classification of general videos like movie and sport scenes. Y. Li et al. have proposed a key-frame extraction method based on sparse modeling for activities of daily living, which integrates multi-modal sensor signals to temper noise and detect salient activities [3], [13].

C. Organization

An overview of our proposed approach to select key frames from a video is shown in Fig. 2. We propose a sparse model with a determinant measure to key frame extraction (Section 2). DC programming is introduced to solve the problem, and the proposed algorithm is described in Section 3. The results of the numerical experimental studies are described in Section 4, which clearly demonstrate the practical performance of the proposed algorithm. Finally, we present our conclusions in Section 5.

II. Model and Formulation

A signal model formulates a mathematical description of the family of interesting signals, which are distinguished from the rest of the signals. With the development of mathematics, linear representation methods have been well studied and have recently received considerable attention [14], [15]. A signal can be expressed as a linear combination of a few representatives, and we can formulate the problem of finding the corresponding representatives as a sparse multiple measurement vector problem [12]. Sparse representation [16]–[19] is a major category of linear representation methods. The purpose of sparse representation is to approximate a signal by a linear combination of atoms chosen from a dictionary matrix. Then, the signal can be represented as a linear combination of a few atoms.

A. Sparse Representation

We assume a data matrix $\mathbf{Y}$ which can be represented as a linear combinations of very few or sparse atoms $\mathbf{w}_i$ from the dictionary $\mathbf{W}$. Namely, the signal $\mathbf{Y}$ can be represented by the product of the dictionary matrix $\mathbf{W}$ and a corresponding sparse coefficient matrix $\mathbf{H}$ with the equation of $\mathbf{Y} = \mathbf{WH}$, where $\mathbf{H}$ contains the representation coefficients of signals $\mathbf{Y}$ with respect to the dictionary $\mathbf{W}$. Another popular way to represent the signal $\mathbf{Y}$ is approximate equality, which is expressed as $\mathbf{Y} \approx \mathbf{WH}$ satisfying $\|\mathbf{Y} - \mathbf{WH}\|_{\mathrm{F}}^2 \leq \epsilon$, where $\|\cdot\|_{\mathrm{F}}$ is the Frobenius norm. Most methods for sparse representation employ the l_0-norm as the sparsity constraint, namely $\|\mathbf{h}\|_0$, which counts the nonzero entries in a vector $\mathbf{h}$. The sparse representation with the l_0-norm constraint can be expressed as

$$\begin{aligned} &\min_{\mathbf{W},\mathbf{H}} \|\mathbf{Y} - \mathbf{WH}\|_{\mathrm{F}}^2 \\ &\text{s.t.} \quad \|\mathbf{H}\|_0 < l, \end{aligned} \tag{1}$$

where l is the number of nonzeros in columns of the coefficient matrix $\mathbf{H}$.

Generally, a common approach of sparse representation is using the l_0-norm as the measure of sparsity to solve the signal sparse coding problem [20]. However, the l_0-norm optimization problem is generally NP-hard, which is difficult to solve. Thus, approximate solutions are considered instead, and several efficient algorithms have been proposed [21]. Unfortunately, these methods cannot provide good enough estimates of signals and are not suitable for high-dimensional problems. In this paper, instead of using the l_0-norm, we introduce the determinant measure, which is differentiable and easy to optimize.

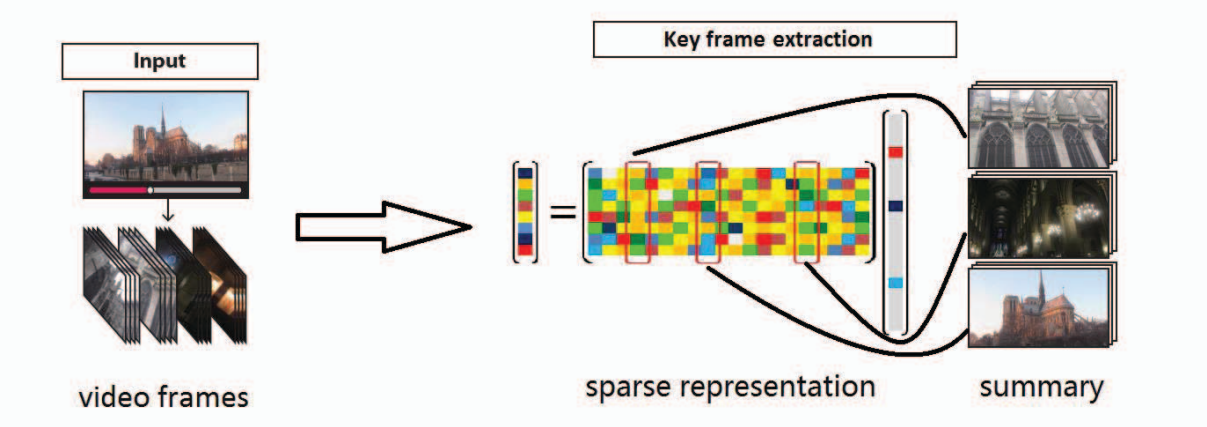

Fig. 2. Data flow in key frame selection based on sparse representation.

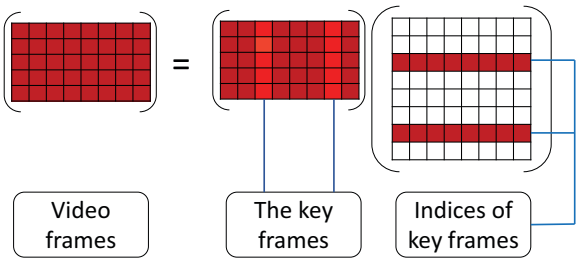

Fig. 3. Matrix representation of key frame extraction based on sparse representation.

B. Key frame extraction with determinant measure

To find representative points that coincide with actual data points and combine sparse representation with key frame extraction, we modify the dictionary learning framework, which first addresses the minimization problem due to the product of two unknown matrices, i.e., the dictionary matrix and the sparse coefficient matrix. Second, it enforces selecting sparse representations from the actual signal by using the collection of the signal data as the dictionary [12]. The indices of key frames are the indices of nonzero rows in the coefficient matrix, as illustrated in Fig. 3. As mentioned above, we intend to introduce a determinant-type of sparse measure to quantify the sparsity of signals. The determinant measure has good features [22], such as differentiable, convex, and well bounded. In addition, the determinant value of matrix interpolates monotonously between two extreme values, 0 and 1, with increasing sparsity degrees [23]. The detailed proof can be found in [24], [25].

Figure 4 illustrates three sparseness degrees of different matrices gauged by the determinant measure. The determinant values of the three matrices from left to right are 0.25, 0.5, and 1. We can find that the larger value the determinant measure is, the sparser the matrix is [26]. Thus we can impose the determinant measure as the sparsity measure of the sparse matrix.

From the objective function of sparse representation, we can achieve the sparsest coefficient matrix $\mathbf{H}$ when the objective function is the optimal minimum. However, in the case of the sparse representation with determinant measure, when $\det(\mathbf{H}\mathbf{H}^{\mathrm{T}})$ takes the maximum, $\mathbf{H}$ is sparsest. What is more, we set the parameter α as a positive constant. Determinant as the sparsity measure is required the rows of $\mathbf{H}$ satisfying sum-to-one. Considering these above factors, we can rewrite the objective function as follows.

$$\begin{aligned} &\min_{\mathbf{H}} F(\mathbf{H}) = \|\mathbf{Y} - \mathbf{W}\mathbf{H}\|_{\mathrm{F}}^2 - \alpha \det(\mathbf{H}\mathbf{H}^{\mathrm{T}}), \\ &\text{s.t.} \quad \mathbf{W} \in \Re_+^{m\times n}, \mathbf{H} \in \Re_+^{n\times N}, \forall i \ \|\mathbf{h}_i\|_2 = 1. \end{aligned} \tag{2}$$

Let $\widetilde{\mathbf{H}} = \mathbf{H}\mathbf{H}^{\mathrm{T}}$. It holds that

$$\widetilde{\mathbf{H}} = \begin{bmatrix} \mathbf{h}_1\mathbf{h}_1^{\mathrm{T}} & \cdots & \mathbf{h}_1\mathbf{h}_i^{\mathrm{T}} & \cdots & \mathbf{h}_1\mathbf{h}_n^{\mathrm{T}} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ \mathbf{h}_i\mathbf{h}_1^{\mathrm{T}} & \cdots & \mathbf{h}_i\mathbf{h}_i^{\mathrm{T}} & \cdots & \mathbf{h}_i\mathbf{h}_n^{\mathrm{T}} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ \mathbf{h}_n\mathbf{h}_1^{\mathrm{T}} & \cdots & \mathbf{h}_n\mathbf{h}_i^{\mathrm{T}} & \cdots & \mathbf{h}_n\mathbf{h}_n^{\mathrm{T}} \end{bmatrix} \tag{3}$$

To find the key frames of the video, we modify the original dictionary learning framework, which is the product of two unknown matrices, the dictionary matrix and the sparse coefficient matrix. The key frames are extracted from the actual video data by using the collection of the actual video data as the dictionary [12]. For this purpose, a sparse representation for key frame extraction can be formulated as follows.

$$\begin{aligned} &\min_{\mathbf{H}} F(\mathbf{H}) = \|\mathbf{Y} - \mathbf{Y}\mathbf{H}\|_{\mathrm{F}}^2 - \alpha \det(\mathbf{H}\mathbf{H}^{\mathrm{T}}), \\ &\text{s.t.} \quad \mathbf{H} \in \Re_+^{n\times N}, \forall i \ \|\mathbf{h}_i\|_2 = 1. \end{aligned} \tag{4}$$

Then, we introduce the DC programming approach to translate the proposed minimization formulation into a DC programming problem. We construct the DC function F as follows.

$$F(\mathbf{H}) = G_1(\mathbf{H}) - G_2(\mathbf{H}), \tag{5}$$

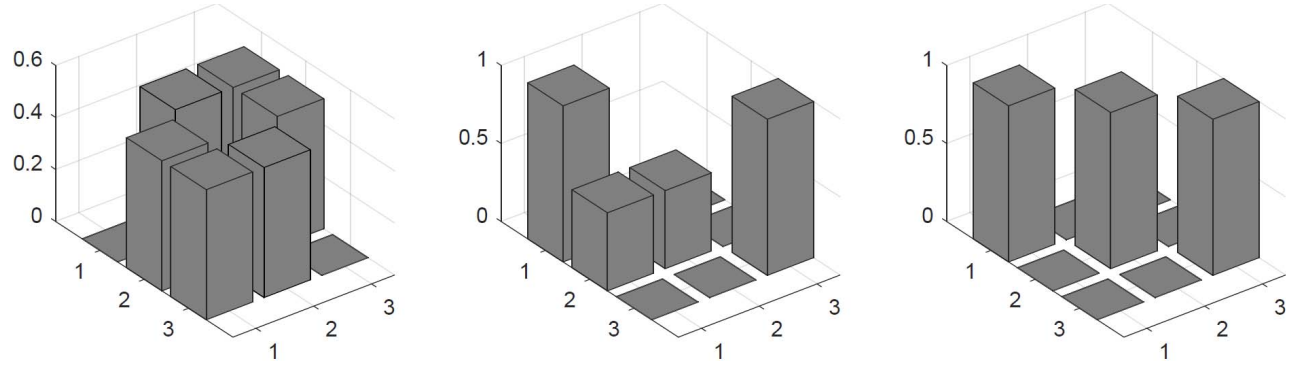

Fig. 4. Illustration of various degrees of sparseness (from left to right, the determinant values are 0.25, 0.5, and 1).

where $G_1(\mathbf{H}) = \|\mathbf{Y} - \mathbf{YH}\|_{\mathrm{F}}^2$ and $G_2(\mathbf{H}) = \alpha \det(\mathbf{HH}^{\mathrm{T}})$ are DC components of F. Then, the original problem can be reformulated as a DC program

$$\min \left\{ G_1(\mathbf{H}) + \chi_{\mathfrak{R}_+^n}(\mathbf{H}) - G_2(\mathbf{H}) : \mathbf{H} \in \mathfrak{R}_+^n \right\}, \tag{6}$$

where $\chi_{\mathfrak{R}_+^n}(\mathbf{H})$ is an indicator function, which is defined by

$$\chi_{\mathfrak{R}_+^n}(\mathbf{H}) = \begin{cases} 0, & \text{if } \mathbf{H} \in \mathfrak{R}_+^n \\ +\infty, & \text{otherwise} \end{cases} \tag{7}$$

The update rule of the coefficient matrix $\mathbf{H}$ can be expressed as follows.

$$\mathbf{H}^{(k+1)} = \max\left(0, \sum_{i=1}^{n} \frac{\mathbf{W}_{i:}^{\mathrm{T}}\mathbf{Y} - \sum_{l=1, l\neq i}^{N} (\mathbf{W}_{i:}^{\mathrm{T}}\mathbf{W}_{:l}\mathbf{H}_{l:} - 0.5\widetilde{\mathbf{H}}_{l:}^{(k)\mathrm{T}})}{[\mathbf{W}^{\mathrm{T}}\mathbf{W}]_{ii}}\right). \tag{8}$$

The details of the derivation of related formulas can be found in [27].

III. Algorithm

In this section, we describe the proposed algorithms for key frame extraction from video based on sparse modeling. We use the nonnegative sparse representation with DC programming (NSR-DC) algorithm to minimize our objective function and obtain the indices of key frames. The procedures are shown in Algorithm 1.

Algorithm 1

Require: Data matrix $\mathbf{V}$ from video

1: Scale each column of the data $\mathbf{V}$ to a unit l_2-norm.
2: Set the regularization parameter α.
3: Initialize $\mathbf{H}$ as a random matrix and scale each row of the coefficient matrix $\mathbf{H}$ to a unit l_2-norm.
4: Execute the NSR-DC algorithm to obtain the sparse coefficients $\mathbf{H}$.
5: Obtain the indices of the key frames of the video, which are non-zeros rows of the sparse coefficient matrix.

IV. Experiments and discussion

In our experiments, we used Matlab R2016b to code all programs, which were executed on a PC with a 3.30 GHz Intel Core i7 CPU and 16 GB of memory, under the Microsoft Windows 10 operating system. The database we use is SumMe[1]. The experiment parameter settings are same with those in Tan et al. [27].

A. Experimental results

In this section, we present the results of selected key frames from numerical experiments. We express the performance of the proposed algorithm applied to different videos from the database of SumMe.

In the experiments, as the determinant measure is used as the sparsity measure, there is a request to scale the rows of the coefficient matrix $\mathbf{H}$ to a unit l_2-norm. In the process of updating, if there is a negative number, we set it as 10^{-8} to replace zero [24].

We use the videos from the SumMe database [28] and apply our proposed algorithm to select the key frames from videos. The selected representatives of video "Fire Domino" are shown in Fig. 5, from which we can find the process of fire burning the tower of matchsticks. We also show the first two key frames selected from the original whole video frames. The selected key frames of video "Jumps" are shown in Fig. 6, from which we can find the steps of jumping from the track.

We apply our proposed algorithm to different videos from SumMe database (Fire Domin, Jumps, Car-over-camera, Cooking, Paluma-jump, Playing-ball, Statue of Liberty, Paintball, Playing-on-water-slide, Scuba, Bike Polo, Car-railcrossing, Bus-in-Rock-Tunnel, and Kids-playing-in-leaves). The numbers of selected key frames are expressed in Table I, from which we can see the effectiveness and performance of our proposed algorithm. For instance, we can select 24 key frames by the proposed algorithm in the whole 1590 frames, the compressive ratio is nearly 1.5% in the video

[1]Database is available on: www.vision.ee.ethz.ch/~gyglim/vsum/.

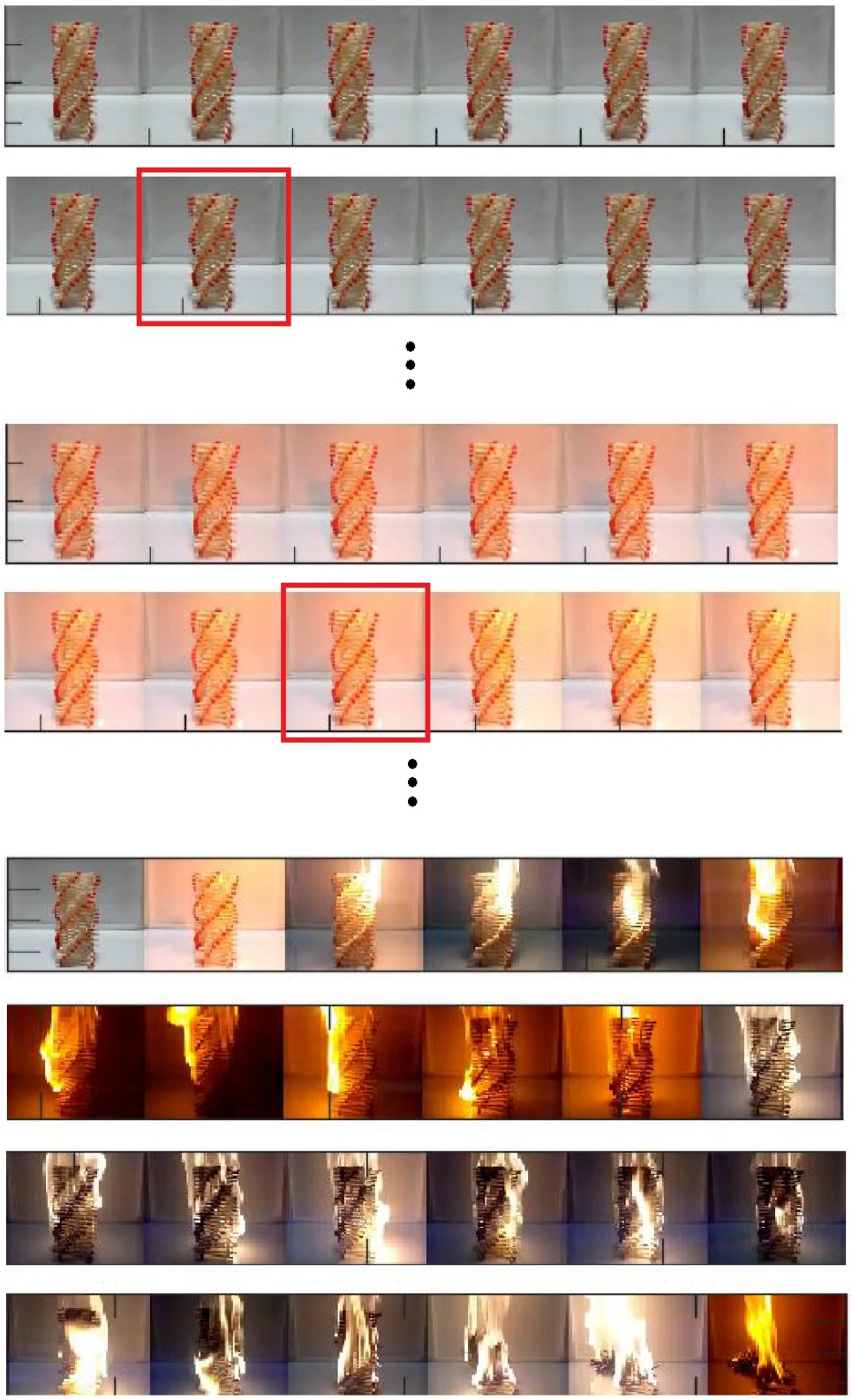

Fig. 5. The first two key frames selected from video frames and the whole key frames extracted from the video: Fire Domino.

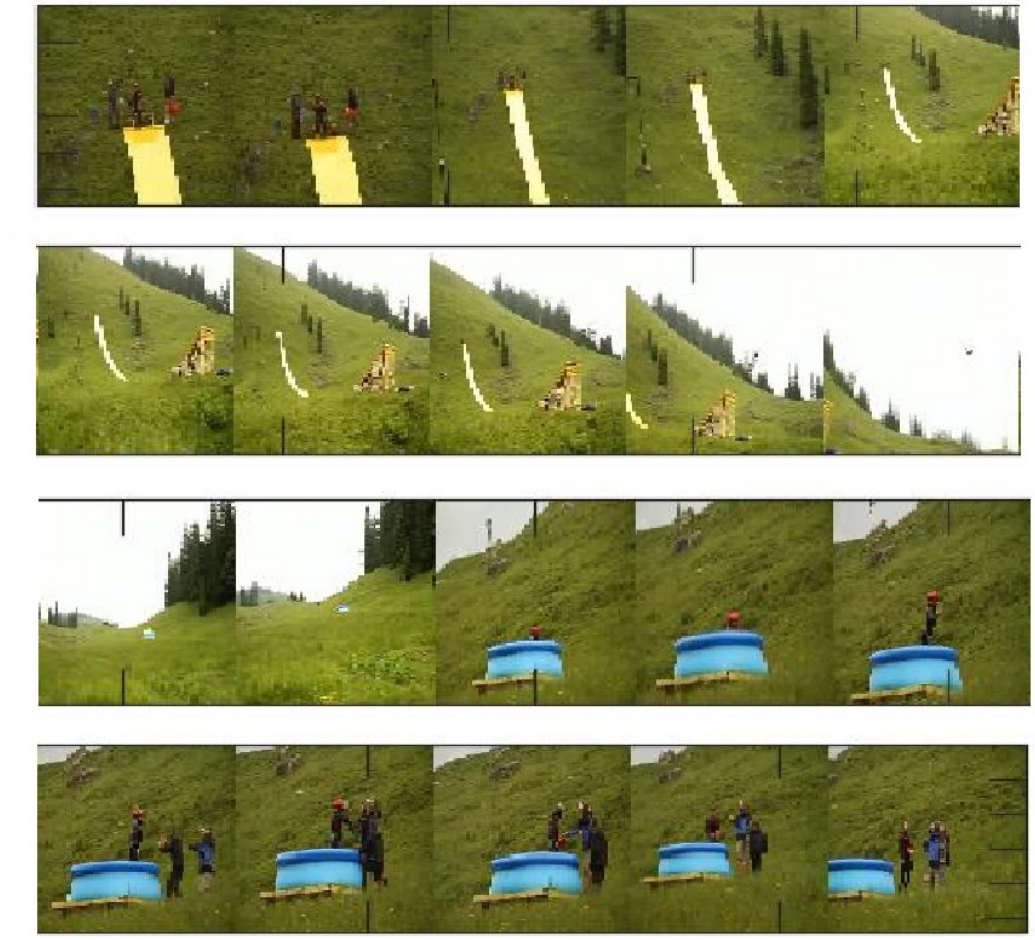

Fig. 6. The key frames from the video: Jumps.

of Fire Domin and select 20 key frames from total 950 frames of the video Jumps (compressive ratio is about 2%).

V. Conclusion

We have proposed a novel framework for key frame extraction of videos by sparse modeling representation selection based on the determinant measure. The algorithm utilizes the determinant-type measure as the sparsity measure and uses a DC programming approach to solve the optimization problem. The sparse coefficients can be obtained by the NSR-DC algorithm. The indices of the key frames are then produced, which prove to be more elegant and informative in representing the key frames of a video. In the future, we plan to improve our method by considering dimensionality reduction and decrease computational consumption.

References

[1] L. Pan, X. Shu, and M. Zhang, "A key frame extraction algorithm based on clustering and compressive sensing," *International Journal of Multimedia and Ubiquitous Engineering*, vol. 10, pp. 385–396, 2015.

[2] P. Mundur, Y. Rao, and Y. Y., "Keyframe-based video summarization using delaunay clustering," *International Journal on Digital Libraries*, vol. 6, p. 219232, 2006.

[3] Y. Li, A. Kanemura, H. Asoh, T. Miyanishi, and M. Kawanabe, "Key frame extraction from first-person video with multi-sensor integration," in *IEEE International Conference on Multimedia and Expo (ICME)*, 2017.

[4] Y. Cong, J. Yuan, and J. Liu, "Sparse reconstruction cost for abnormal event detection," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[5] Y. Cong, B. Fan, J. Liu, J. Luo, and H. Yu, "Speeded up low-rank online metric learning for object tracking," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 25, no. 6, pp. 922–934, 2015.

[6] Y. Cong, J. Yuan, and J. Luo, "Towards scalable summarization of consumer videos via sparse dictionary selection," *IEEE Transactions on Multimedia*, vol. 14, pp. 66–75, 2012.

[7] A. Kanemura, J. Yuan, and Y. Kawahara, "Finding structured dictionary representation by network-flow optimization," in *Workshop on Data Discretization and Segmentation for Knowledge Discovery (DDS)*, 2013.

[8] H. Wang, Y. Kawahara, C. Weng, and J. Yuan, "Representative selection with structured sparsity," *Pattern Recognition*, vol. 63, pp. 268–278, 2017.

[9] N. Ejaz, T. B. Tariq, and S. W. Baik, "Adaptive key frame extraction for video summarization using an aggregation mechanism," *Journal of*

TABLE I

THE NUMBER OF KEY FRAMES AND THE NUMBER OF WHOLE FRAMES OF DIFFERENT VIDEOS.

Video name	The number of key frames	The number of whole frames
Fire Domino	24	1590
Jumps	20	950
Car-over-camera	28	4234
Cooking	32	1275
Paluma-jump	28	2465
Playing-ball	77	3120
Statue of Liberty	46	3850
Paintball	9	5842
Playing-on-water-slide	54	2958
Scuba	332	2220
Bike Polo	62	3060
Car-railcrossing	36	4901
Bus-in-Rock-Tunnel	39	5130
Kids-playing-in-leaves	36	3074

Visual Communication and Image Representation, vol. 23, no. 7, pp. 1031–1040, 2012.

[10] Q. Xu, Y. Liu, X. Li, Z. Yang, J. Wang, S. M., and R. Scopigno, "Browsing and exploration of video sequences: A new scheme for key frame extraction and 3D visualization using entropy based jensen divergence," *Information Sciences*, vol. 278, pp. 736–756, 2014.

[11] X. Liu, M. L. Song, L. M. Zhang, and S. L. Wang, "Joint shot boundary detection and key frame extraction," in *International Conference on Pattern Recognition (ICPR)*, 2012.

[12] E. Elhamifar, G. Sapiro, and R. Vidal, "See all by looking at a few: Sparse modeling for finding representative objects," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[13] Y. Li, A. Kanemura, H. Asoh, T. Miyanishi, and M. Kawanabe, "Extracting key frames from first-person videos in the common space of multiple sensors," in *IEEE International Conference on Image Processing (ICIP)*, 2017.

[14] B. K. Natarajan, "Sparse approximate solutions to linear systems," *SIAM Journal on Computing*, vol. 24, pp. 227–234, 1995.

[15] M. Huang and et al, "Brain extraction based on locally linear representation-based classification," *NeuroImage*, vol. 92, pp. 322–339, 2014.

[16] M. Elad, *Sparse and Redundant Representations*. Springer, 2010.

[17] M. S. Lewicki and T. J. Sejnowski, "Learning overcomplete representations," *Neural Computation*, vol. 12, no. 2, pp. 337–365, 2000.

[18] K. Kreutz-Delgado, J. F. Murray, and B. D. Rao, "Dictionary learning algorithms for sparse representation," *Neural Computation*, vol. 15, pp. 349–396, 2003.

[19] Z. Li, S. Ding, and Y. Li, "A fast algorithm for learning overcomplete dictionary for sparse representation based on proximal operators," *Neural Computation*, vol. 27, no. 9, pp. 1951–198, 2015.

[20] L. Chaari, H. Batatia, N. Dobigeon, and J.-Y. Tourneret, "A hierarchical sparsity-smoothness bayesian model for ℓ_0+ ℓ_1+ ℓ_2 regularization," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2014, pp. 1901–1905.

[21] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," *Asilomar Conference on Signals, Systems, and Computers*, pp. 40–44, 1993.

[22] S. Nam, M. E. Davies, M. Elad, and R. Gribonval, "The cosparse analysis model and algorithms," *Applied and Computational Harmonic Analysis*, vol. 34, no. 1, pp. 30–56, 2013.

[23] R. Schachtner, G. Poppel, and E. W. Lang, "A nonnegative blind source separation model for binary test data," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 57, no. 7, pp. 1439–1448, 2010.

[24] Z. Tang, S. Ding, and Z. Yang, "Dictionary learning for sparse representation by nonnegative matrix factorization with constraint of determinant-type of maximization," *ICIC Express Letter*, vol. 4, no. 5, 2010.

[25] Z. Yang, Y. Xiang, S. Xie, S. Ding, and Y. Rong, "Nonnegative blind source separation by sparse component analysis based on determinant measure," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 23, no. 10, pp. 1601–1610, 2012.

[26] Y. Li, S. Ding, and Z. Li, "Dictionary learning with the cosparse analysis model based on summation of blocked determinants as the sparseness measure," *Digital Signal Processing*, vol. 48, pp. 298–309, 2016.

[27] B. Tan, Y. Li, S. Ding, and X. Li, "Recovering nonnegative sparse signals with a determinant-type of sparse measure and DC programming," in *IEEE International Conference on Applied Computer and Communication Technologies (IEEE ComCom)*, 2017.

[28] M. Gygli, H. Grabner, H. Riemenschneider, and L. Van Gool, "Creating summaries from user videos," in *European Conference on Computer Vision*, 2014.