BAG OF SUB-GRAPHS FOR VIDEO EVENT RECOGNITION

Najib BEN AOUN, Mahmoud MEJDOUB and Chokri BEN AMAR

REGIM-Lab: REsearch Groups on Intelligent Machines, University of Sfax,
National Engineering School of Sfax (ENIS), BP 1173, Sfax, 3038, Tunisia

ABSTRACT

Recognizing video events has been a very active field of interest. The diversity of videos captured in complex environments and under difficult conditions makes the event recognition a challenging task. In this paper, we present a video event recognition method which exploits the power of graphs for representing the structural organization of the features and the success of the Bag-of-Words approach. Our method combines the Scale Invariant Feature Transform and the Space-Time Interest Point features to characterize the video. To model the spatio-temporal relations among these features, a graph-based representation is used for each video. Then, the video is indexed based on a histogram of frequent sub-graphs. To evaluate our method, we have used the Columbia Consumer Video dataset. The experimental results show the efficiency of the proposed method.

Index Terms— Video event recognition, Spatio-temporal features, Bag-of-sub-Graphs, Graph-based video modeling.

1. INTRODUCTION

Many efforts have been devoted to recognize the human actions[1, 2], hand gestures [3] and especially the video events [4, 5, 6, 7]. The Bag-of-Words (BoW) approach was commonly followed due to promising results that it provides [4, 5, 8, 9, 10, 2, 1]. It consists on representing each video with a histogram of visual words obtained by quantifying the extracted features. In [4], the static Scale Invariant Feature Transform (SIFT) [11] and the Space-Time Interest Point (STIP) features [12] are used as visual features and the Mel-Frequency Cepstral Coefficients (MFCC) is used as audio features to describe the video events. Then, three BoW are conducted on these features of all the videos. Consequently, a 5000 dimensional SIFT histogram, a 5000 dimensional STIP histogram and a 4000 dimensional MFCC histogram are concatenated together to form the final video descriptor.

The main drawback of the BoW approach is the lack of structural organization of the features. These weaknesses can be surpassed by interconnecting the features using structured models such as the graphs [5, 13, 6, 1, 14]. In [5], Ye et al. have used the same audio and visual features as [4] following the BoW approach to form audio and visual codebooks. Then, the audio words are connected to the visual words with a bipartite graph which is partitioned in order to discover the audio-visual bi-modal words. Each bi-modal word is a group of audio and/or visual words that frequently co-occur together. So, the audio-visual bi-modal codebook will be formed and used to index the videos. Graph-based representation is also used in [6] where the video is represented with a set (string) of feature graphs that respect the spatio-temporal ordering. Graph matching is conducted in order to determine the classes of the events present inside the video. Najib et al. [13] have also used the graph-based representation. Video frame are modeled with a spatial graph where graph vertices represent the visual features of the frame segmented regions. So, the video is indexed with a binary histogram indicating the presence/absence of the frequent sub-graphs discovered with the graph-based Substructure pattern mining (gSpan) algorithm [15]. As a final descriptor, the video is indexed by the combination of the spatial graph-based histogram and a block-matching based motion feature [16, 17] that describes the temporal information of the video.

Despite these recent attempts, exploiting the graph-based representation for video modeling remains limited. In this paper, we propose a graph-based video event recognition (GVER) system. Our system exploits the efficiency of the graphs for representing the structural organization of the video features and the success of the BoW approach for its indexing. The SIFT and the STIP features are extracted from the detected interest points and combined together to characterize the video. Then, the k-means algorithm is conducted to quantize the features into k clusters. Thereafter, the video is represented with a spatial graph set and a temporal graph set interconnecting its features. Based on its graph representation, the video is indexed with a histogram of frequent sub-graphs, extracted from the graph database. This is done in a similar way to the BoW approach considering that the frequent sub-graphs represent the visual words. From this comes the idea to call our approach: Bag-of-sub-Graphs (BoG). Finally, like [4], a one-versus-all SVM with χ^2 kernel is applied to classify the video events.

In comparison with our previous work [13], four main contributions can be revealed in this work: (1) rather than using spatial and temporal features computed separately from the video, spatio-temporal features are used to detect the

978-1-4799-2893-4/14/$31.00 ©2014 IEEE

video motion and appearance around local 3D-interest points; (2) in [13], spatial graphs are constructed from segmented frame regions while in this work, they are constructed from detected spatio-temporal interest points. Consequently, this spatial graphs will not only surmount the problem of region segmentation but also integrate the temporal information in the video. (3) In this work, the spatial graphs are coupled with temporal graphs in order to extend the graph-based modeling from the frame level to the video level so to give an overall description of the video; (4) For the graph-based video indexing, we have improved the histogram encoding from a binary histogram to a numerical histogram that gives the number of occurrences of frequent sub-graphs.

In comparison with the state of the art methods, the contributions of this work can be summarized as follows: (1) Unlike [5, 6], we use a combination of spatial and temporal graphs. (2) To overcome the aforementioned drawbacks of the BoW based methods [4, 5], we have used spatio-temporal features and a model on top of them which provides the spatio-temporal relationships. (3) Our method can be considered as a generalization of the method used in [5, 6] where relatively simple graph structures are used to model the video. In contrast, our approach allows a more general structure with a higher and variable order as well as flexible topography. (4) Unlike most methods [5, 6], only significant and relevant sub-graphs are retained to model the video. (5) Unlike [6] that employ sequential graph matching to retrieve the most similar videos, we convert the graph matching problem to a vector space one by representing a graph with a histogram of frequent sub-graphs. This enables us to apply learning algorithms [4, 18] on these histograms for event classification.

The rest of this paper is organized as follows. In section 2, we describe in detail our GVER method which feat the efficiency of the graph representation to model and index the video events. In section 3, we evaluate our method on the Columbia Consumer Video (CCV) dataset [4]. The experimental results show the effectiveness of the proposed method. Finally, we conclude our paper by giving a summary of the presented work and proposing some future extensions.

2. OUR GRAPH-BASED VIDEO EVENT RECOGNITION METHOD

Figure 1 illustrates the framework of our GVER system. Our system is composed of two phases: training phase and testing phase. In the training phase, from each training video, features are extracted and used to construct a set of spatial graphs (one graph for each video frame) and a set of temporal graph. As a result, two graph databases are formed. Then, from each graph database, frequent sub-graphs are discovered and used to model each video with a histogram of frequent spatial sub-graphs and a histogram of frequent temporal sub-graphs. The two histograms are normalized and horizontally concatenated together to form the final video descriptor.

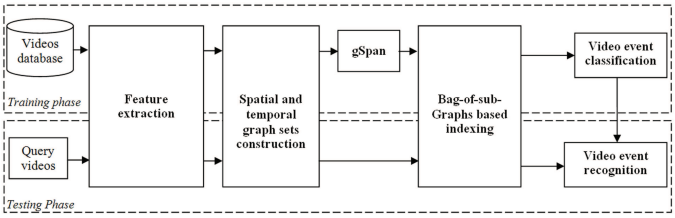

Fig. 1. Illustration of our GVER system

Consequently, the video will be indexed with a Bag-of-sub-Graphs. Finally, the training video descriptors are classified with SVM [4] to build a model for each video event. In the testing phase, the same process is followed to compute the graph-based video descriptor. Then, the video events are recognized using the event models already built by SVM in the training phase. Our GVER system is described in detail in the next sections.

2.1. Feature extraction

The combination of the SIFT and the STIP visual features has been proved to be efficient to recognize video event [4, 5]. So, we have used it in our system. After, the detection of spatio-temporal interest points with the Harris-3D method [19], a 128 dimensional SIFT is extracted at each point to capture the spatial local gradients. This feature is invariant to scale and robust to affine distortion. Besides, from each interest points, the STIP features are computed in order to describe the space-time variations. STIP feature extraction consist on extracting the Histograms of Oriented Gradients (HOG), describing the local appearance, and the Histograms of Optical Flow (HOF), describing the motion in the video, from the 3D volume around the interest points. We have used the Laptev method [19] to extract a 72 dimensional HOG and a 72 dimensional HOF and concatenate them to form a 144 dimensional feature as the final descriptor for each interest point.

2.2. Graph-based video representation

Each video is represented with spatial and temporal video graphs sets. Spatial and temporal video graph vertices are the spatio-temporal interest points already detected with the Harris-3D method. Each graph vertex is labeled by the class of its corresponding spatio-temporal interest point. To obtain the class of an interest point, all interest point descriptors (SIFT+STIP) of the training videos are clustered with the k-means algorithm into k clusters. So, the class of an interest point is the nearest cluster of the its descriptor.

To construct the Spatial Video Graph Set (SVGS), we spatially connect all the spatio-temporal interest points of the same video frame together (see Figure 2). Afterward, the graph edge connecting two spatio-temporal interest points

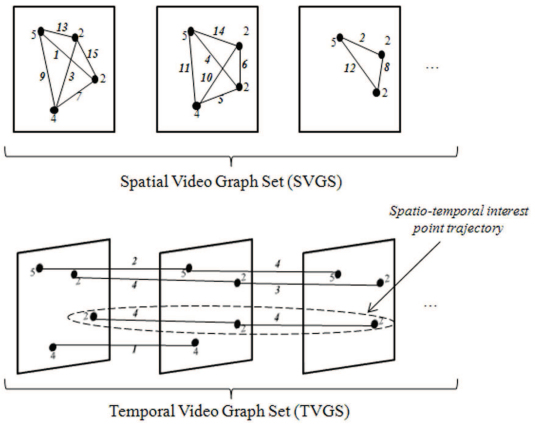

Fig. 2. An example of a SVGS and a TVGS of a video

(two vertices) is associated to an edge vector. This latter is composed of the displacements dx and dy, respectively, between the x-coordinates and the y-coordinates of the two participating vertices. Then, to label the edges, the k-means algorithm is applied to all edge vectors associated to the training videos and each edge is labeled by the closest cluster to its corresponding vector.

Having regard to The Temporal Video Graph Set (TVGS), each spatio-temporal interest point is connected to the one which has the same label in the next frame (see Figure 2). The search of the spatio-temporal interest point in the next frame is performed inside a window of $n \times n$ pixels. Afterward, the graph edge is labeled with the temporal displacement between two participating vertices. This displacement is quantized into 4 directions: top-left, top-right, bottom-left and bottom-right. Consequently, the TVGS will be formed by collecting all the trajectories of the spatio-temporal interest points.

2.3. Video indexing using frequent sub-graphs

Until this phase, each video is represented by a SVGS and a TVGS. Two graph databases will be then formed from the SVGSs and the TVGSs associated to the training videos: the database of the spatial graphs (DSG) and the database of the temporal graphs (DTG). Afterward, frequent spatial and temporal sub-graphs are discovered from them by applying the gSpan algorithm [15] (see Figure 3). Then, each video will be indexed by two histograms (histogram of the frequent spatial sub-graphs and histogram of frequent temporal sub-graphs) which are then horizontally concatenated together to form the final video descriptor.

In order to reduce the high complexity of the graph matching (graph isomorphism), frequent sub-graphs are discovered from the graphs and used to identify the similarity between them [13]. The discovery of the frequent sub-graphs is based on the minimum support (minSup), which is the number of

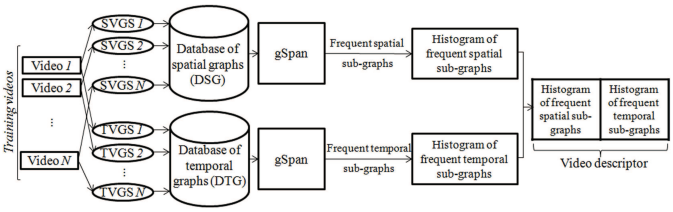

Fig. 3. Video indexing using frequent spatial and temporal sub-graphs

appearances that a sub-graph must exceed to be frequent in a database. Different frequent subgraphs discovery algorithms have been developed [15, 20, 21, 22]. The gSpan algorithm [15] outperforms other algorithms in the computational time and is capable of mining large frequent sub-graphs in a big graph set. This makes it an appropriate algorithm for our GVER system.

To compute the video histogram of the frequent spatial subgraphs, we proceed as follows:

- For each graph in the SVGS, we form a frequent spatial sub-graphs histogram. It consists in counting the number of occurrences, in the graph, of each frequent spatial sub-graphs discovered by gSpan.
- The histograms are added up over all the graphs in the SVGS to form the video histogram of the frequent spatial sub-graphs.

The same procedure is followed to compute the histogram of frequent temporal sub-graphs. The two histograms are horizontally concatenated together to form the video descriptor which is then rescaled. The rescaling is conducted to standardize the descriptor element values between 0 and 1.

Finding the frequencies of a sub-graph in a graph is a challenging task since it involves sub-graph isomorphism (a sub-graph isomorphism problem consists in deciding if there exists a copy of a sub-graph in a target graph) which is an NP-complete problem [13]. To surmount the sub-graph isomorphism problem, the Maximum Common Sub-graph (MCS) method is used [13]. It finds the Common Sub-graph (CS), which is the largest common substructure between the graph and the sub-graph. Then, it computes the maximum clique in CS (the maximum clique is the largest group of vertices in a graph that are all connected to each other another). Consequently, if the maximum clique has at least the size of the sub-graph, the sub-graph is considered to be present in the graph.

In our work, to compute the number of occurrences N_o of a frequent sub-graph S_G in a graph G, we begin by computing the CS between them. CS is composed of:

- Edges which exist, with the same labels and the same composing vertice labels, in S_G and in G (commonly present).

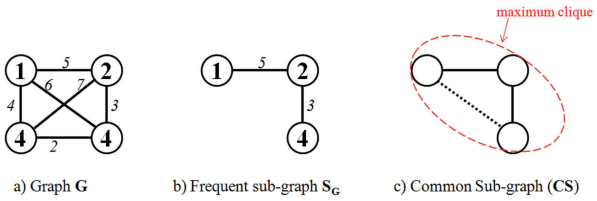

Fig. 4. Detecting a maximum clique from a CS between a graph G and a non complete frequent sub-graph S_G.

- Edges, with the same composing vertice labels, which do not exist in S_G and exist in G (added to the CS for calculation reasons in order to be able to use the maximum clique detection method [23]).

The Figure 4 shows an example of CS between a graph G and a frequent sub-graph S_G. In CS, the edges in bold are those which are commonly present, the dashed edge is the one which is added for computation reasons.

In the next step, we detect in CS the maximum cliques that have a size equal to or greater than the size of S_G. N_o corresponds then to the number of the detected maximum cliques. In CS of the Figure 4, we have one maximum clique having a size (3) which is superior to the size of S_G (2). So, the frequency of S_G in G is equal to 1.

3. EXPERIMENTS

The evaluation of our GVER system is done on a benchmark dataset for video event recognition: the Columbia Consumer Video dataset (CCV) [4]. It is composed of 9317 unedited consumer videos (about 210 hours in total) which make it one of the largest datasets publicly available in the Internet. The importance of these videos, shared in the YouTube website net, is that they contain important events for consumers. The dataset is divided into a training set of 4659 and a testing set of 4658 videos and it is labeled with 20 semantic categories of consumer concepts (15 of them are video events).

In our experiment, we have extracted local SIFT and STIP features. Then, to label the graphs, the SIFT+STIP features and the graph edge vectors are quantized using the k-means algorithm. The number of frequent sub-graphs is obtained by experimentally fixing the *minSup* in gSpan. For DSG, we denote by k_{vs}, k_{es} and N_{fs} respectively the number of vertice labels, the number of edge labels and the number of frequent spatial sub-graphs. For DTG, we denote by k_{vt} and N_{ft} respectively the number of vertice labels and the number of frequent temporal sub-graphs. Using cross validation, we tune these parameters in order to have their optimal values. For each couple (k_{vs}, k_{es}), we change the minSup value. For each minSup, we obtain a value of N_{fs}. For DSG, the optimal values of k_{vs}, k_{es} and N_{fs} are respectively 2000, 1000 and 7700. This setting is validated on the test dataset since

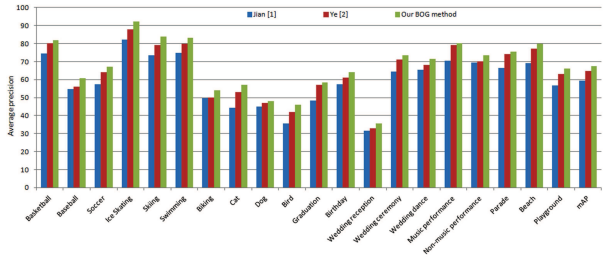

Fig. 5. Per-category performance comparison on CCV dataset. This figure is best viewed in color.

it gives the best mean Average Precision (mAP). For DTG, the optimal values of k_{vt} and N_{ft} are respectively equal to 1500 and 4400. For the construction of the TVGSs, we have taken, as search window size, *n=15* since it has given experimentally the best result by applying cross validation in the training dataset. To classify the videos, we have adopted the one-versus-all multi-class SVM with χ^2 kernel used in [4].

Figure 5 presents a comparison between our GVER system and the state-of-the-art methods for the CCV dataset in terms of event average precision (AP) and mAP. It can be observed that most events show relatively low recognition rates. This can be explained by the fact that the CCV dataset contains particularly challenging video conditions. Using the combination of STIP and SIFT features following the BoW approach, the method proposed in [4] has reached a mAP equal to 55.1%. By including the MFCC features, they have obtained a 59.5% mAP. The graph and BoW based descriptor used in [5] has given a 64.6% mAP using the three features (SIFT+STIP+MFCC). In our BoG method, the combination of frequent spatial and temporal sub-graph histograms results on a 12100-dimensional video descriptor. Using this video descriptor, we have reached a mAP equal to 67.58%. As it can be noticed, our video descriptor is not only less dimensional than the one used in [4, 5] which used a 14000-dimensional descriptor, but also it has given better result. Besides, without using the audio feature, our system outperforms the other methods [4, 5] which have used the audio modality to recognize the video events. This encouraging result illustrates the ability of our BoG method to build useful event models.

4. CONCLUSION

In this work, we demonstrated that graph-based video representation is efficient for recognizing video events. The proposed method exploits the benefits of graphs to model the structural organization of the local features and the success of the BoW approach to recognize video events. In our future works, due to the promising results given by our BoG method, we will extend our work to the video detection task. Besides, we will test more visual features and try to exploit the audio information using features such as MFCC.

5. REFERENCES

[1] N. Ben Aoun, M. Mejdoub, and C. Ben Amar, "Graph-based approach for human action recognition using spatio-temporal features," *Journal of Visual Communication and Image representation*, vol. 25, no. 2, pp. 329–338, 2014.

[2] M. Sekma, M. Mejdoub, and C. Ben Amar, "Human action recognition using temporal segmentation and accordion representation," in *15th International Conference on Computer Analysis of Images and Patterns*, 2013, pp. 563–570.

[3] T. Bouchrika, M. Zaied, O. Jemai, and C. Ben Amar, "Neural solutions to interact with computers by hand gesture recognition," *Multimedia Tools and Applications*, 2013.

[4] Y-G. Jiang, G. Ye, S-F. Chang, D. Ellis, and A.C. Loui, "Consumer video understanding: A benchmark database and an evaluation of human and machine performance," in *ACM International Conference on Multimedia Retrieval*, 2011.

[5] G. Ye, I-H. Jhuo, D. Liu, Jiang Y-G., D.T. Lee, and S-F Chang, "Joint audio-visual bi-modal codewords for video event detection," in *2nd ACM International Conference on Multimedia Retrieval*, 2012.

[6] U. Gaur, Y. Zhu, B. Song, and A. Roy-Chowdhury, "A string of feature graphs model for recognition of complex activities in natural videos," in *13th International Conference on Computer Vision*, 2011, pp. 2595–2602.

[7] A. Wali, N. Ben Aoun, H. Karray, C. Ben Amar, and M.A. Alimi, "A new system for event detection from video surveillance sequences," in *12th International Conference on Advanced Concepts for Intelligent Vision Systems*, 2010, vol. II, pp. 110–120.

[8] M. Mejdoub, L. Fonteles, C. Ben Amar, and Marc Antonini, "Fast indexing method for image retrieval using tree-structured lattices," in *International Workshop on Content-based multimedia indexing*, 2008, pp. 365–372.

[9] M. Mejdoub and C. Ben Amar, "Classification improvement of local feature vectors over the knn algorithm," *Multimedia Tools and Applications*, vol. 64, pp. 197–218, 2013.

[10] M. Dammak, M. Mejdoub, M. Zaied, and C. Ben Amar, "Feature vector approximation based on wavelet networks," in *International Conference on Agents and Artificial Intelligence*, 2012, pp. 394–399.

[11] D. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal on Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[12] I. Laptev and T. Lindeberg, "Space-time interest points," in *9th International Conference on Computer Vision*, 2003, vol. 1, pp. 432–439.

[13] N. Ben Aoun, H. Elghazel, and C. Ben Amar, "Graph modeling based video event detection," in *7th International Conference on Innovations in Information Technology*, 2011, pp. 114–117.

[14] N. Ben Aoun, H. Elghazel, M-S. Hacid, and C. Ben Amar, "Graph aggregation based image modeling and indexing for video annotation," in *International Conference on Computer Analysis of Images and Patterns*, 2011, pp. 324–331.

[15] X. Yan and J. Han, "gspan: graph-based substructure pattern mining," in *IEEE International Conference on Data Mining*, 2002, pp. 721–724.

[16] N. Ben Aoun, M. El'Arbi, and C. Ben Amar, "Multiresolution motion estimation and compensation for video coding," in *10th International Conference on Signal Processing*, 2010, vol. II, pp. 1121–1124.

[17] N. Ben Aoun, M. EL'ARBI, and C. Ben Amar, *Advances in Wavelet Theory and Their Applications in Engineering, Physics and Technology*, chapter Wavelet Transform Based Motion Estimation and Compensation for Video Coding, pp. 23–40, InTech, 2012.

[18] O. JEMAI, M. ZAIED, C. BEN AMAR, and ALIMI M. A., "Fast learning algorithm of wavelet network based on fast wavelet transform," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 25, no. 8, pp. 1297–1319, 2011.

[19] I. Laptev, "On space-time interest points," *International Journal of Computer Vision*, vol. 64, pp. 107–123, 2005.

[20] M. Kuramochi and G. Karypis, "Frequent subgraph discovery," in *IEEE International Conference on Data Mining*, 2001, pp. 313–320.

[21] J. Huan, W. Wang, and J. Prins, "Efficient mining of frequent subgraphs in the presence of isomorphism," in *IEEE International Conference on Data Mining*, 2003, pp. 549–552.

[22] A. Inokuchi, T. Washio, , and H. Motoda, "An apriori-based algorithm for mining frequent substructures from graph data," in *Principles of Data Mining and Knowledge Discovery*, 2000, pp. 13–23.

[23] K. Janez and J. Dusanka, "An improved branch and bound algorithm for the maximum clique problem," *MATCH Communications in Mathematical and in Computer Chemistry*, vol. 58, pp. 569590, 2007.