Sports Video Classification Using Bag of Words Model

Dinh Duong, Thang Ba Dinh, Tien Dinh, and Duc Duong

Faculty of Information Technology, University of Science
Ho Chi Minh City, Vietnam
dinhseva@gmail.com,
{dbthang,dbtien,daduc}@fit.hcmus.edu.vn

Abstract. We propose a novel approach classify different sports videos given their groups. First, the SURF descriptors in each key frames are extracted. Then they are used to form the visual word vocabulary (codebook) by using K-Means clustering algorithm. After that, the histogram of these visual words are computed and considered as a feature vector. Finally, we use SVM to train each classifier for each category. The classification result of the video is the production of the scores output from all of the key frames. An extensive experiment is performed on a diverse and challenging dataset of 600 sports video clips downloaded from Youtube with a total of more than 6000 minutes in length for 10 different kinds of sports.

Keywords: image classification, sports video classification, bag of words, SVM, K-means, SURF.

1 Introduction

These days, with the rapid growth of technology, video cameras can be purchased with a surprisingly low cost. The consequence of this fact is the tremendous existing amount of videos from both broadcast and personal sources. Automatic video classification becomes a crucial task in video analysis because it is impossible for people to annotate such a vast amount of video resources. In this paper, we focus in classifying sports video shots into different categories, which is important for many applications such as content-based video search, sports strategy analysis, video highlights recommendation. This is a very challenging problem as sports videos are very dynamic and often share similar motion characteristics.

It is worth to emphasize that, most of the current approaches in video classification is inspired by image classification which is still one of the most challenging problems in computer vision. The approaches can be formulated in two categories: discriminative and generative. Discriminative methods often use Bag-of-Words (BoW) representation [6,7], where visual words are local features such as SIFT [1], SURF [2]. Grauman and Darrel also presented the Spatial pyramid matching (SPM) [8] which is efficient for whole image classification. The generative methods, on the other hand, focus on the topic models as in [9].

J.-S. Pan, S.-M. Chen, N.T. Nguyen (Eds.): ACIIDS 2012, Part III, LNAI 7198, pp. 316–325, 2012.

To address the video classification problem, people often choose to narrow down the domain such as: horror movie scenes [10], violent scenes [13]. Some others choose to classify type of camera views in a specific genre of video such as soccer videos [14]. Some approaches try to fuse different cues such as caption, audio, visual information [13, 14].

In this paper, we address the problem of categorizing sports videos due to its popularity and challenges. Takagi [11] focused on camera motion in the video sequence to categorize 6 different sport types. Ling-Yu Duan *et. al.* [12] used top down video shot classification based on pre-defined video shot classes, each of which has a clear semantic meaning. They tested on 4 types of sports and get 85% – 95% of accuracy rate. Recently, Zhang and Guan [15] proposed a large scale video genre classification method using SIFT descriptors in a modified latent Dirichlet allocation (mLDA) framework. The classifier is then built using k-NN algorithm. The method was tested in 23 sports dataset and achieve from 55%-100% accuracy ratios for different categories.

Here we propose a novel method based on Bag-of-Words (BoWs) from which visual words are represented by SURF descriptors. In the experiments, we collected a diverse and challenging dataset with a total of more than 6000 minutes in length. We have tested and analyzed our model intensively on this dataset. The recall and precision analysis shows robust results in all types of sports.

The rest of the paper consists of 3 sections. In section 2, the details of our algorithm are described. Then, Section 3 shows the experiment settings and results followed by the conclusion and future work discussion in Section 4.

2 Our Approach

Our approach is based on bag of visual words model, which originally comes from document representation in terms of form and semantic. The bag of words model has been widely used in classification, recognition, content-based image retrieval and detection [6,7]. Inspired by the image classification algorithm proposed by Csurka et. al. [6], which has been proved to effectively in static image dataset, we propose to classify sports video by extracting key frames, and classify each of them. The final classification result of the video is the production of the scores output from all of the key frames. Our method contains the following 4 main steps:

- Descriptors of video frames are detected and extracted by using SURF approach.
- The descriptors are then used to form up the visual word vocabulary (codebook) by using a cluster algorithm.
- A histogram representation is formed to count the number of visual words appeared in each frame.
- A multi-class classifier considers histogram representation of a frame as a feature vector. It, then, determines which class to assign the test frame to.

Fig. 1 illustrates the 4-step model for sports video classification.

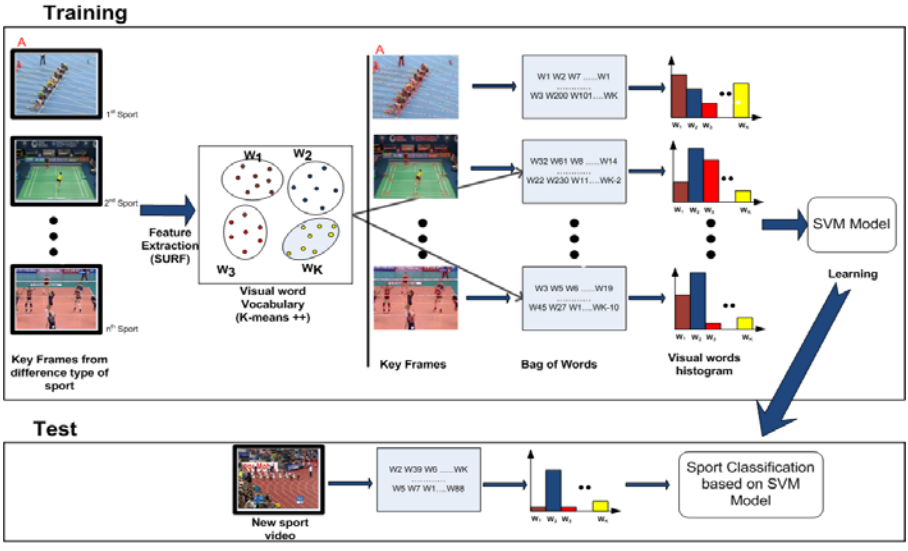

Fig. 1. Illustration of our four steps method base on Bag of Words model

2.1 Descriptors Extraction

Firstly, in the training dataset, key frames are collected base on our pre-defined shot views. Key frames then become input of this step. It then outputs the key points set and their descriptors. Key point is a point in the image, which has a rich local information and is stable under local and global perturbed activities in the image domain, such as affine transformations, scale changes, and rotation. Many key point detectors including Harris, Harris-Laplace, DoG and the descriptors such as SIFT [1], and SURF [2] providing very impressive results. In our framework, we adopt SURF because of its good performance comparing to other methods while having reasonable running time.

Key point detector

SURF detector is based on Hessian Matrix and rely on the determinant of the Hessian for selecting the location and the scale. Given a point $x = (x, y)$ in an image *I*, the Hessian matrix $H(x,\sigma)$ in *x* at scale σ is defined as follows:

$$H(x,\sigma) = \begin{bmatrix} L_{xx}(x,\sigma) & L_{xy}(x,\sigma) \\ L_{xy}(x,\sigma) & L_{yy}(x,\sigma) \end{bmatrix} \quad (1)$$

Where $L_{xx}(x,\sigma)$ is the convolution of the Gaussian second order derivative $\frac{\partial^2}{\partial x^2} g(\sigma)$ with the image *I* in point *x*. It is also similar for $L_{xy}(x,\sigma)$ and $L_{yy}(x,\sigma)$.

The maxima of the determinant of Hessian matrix are then interpolated in scale and image space. Fig. 2 shows an example of the key points detected by SURF.

Fig. 2. Detected key points on tennis, football and swimming

Key point descriptor

The key point descriptor in SURF requires 3 steps. First, it constructs a circular region around the key points and then uses Haar wavelet to compute the orientation in both x and y directions. Finally, SURF descriptors are extracted using square regions. Square regions are split into 4 x 4 sub regions, producing the standard SURF descriptor, SURF-64. Furthermore, another version of SURF descriptor is SURF-128, in which a couple of similar features are added. Because of the advantages of SURF-128 such as more distinctive and not much slower than SURF-64, we apply SURF-128 for key point descriptors in our experimental studies.

2.2 Visual Word Vocabulary Calculation

Assuming that the input contains *n* descriptors, this step aims to partition these *n* descriptors into *k* clusters. Each cluster is called a "visual word". A set of *k* clusters forms the visual vocabulary. After that, the vocabulary is used to represent a video frame, *i.e.* an image. The main idea is that each visual word is represented a region of interest in a frame. Fig. 3 shows how the K-means approach groups similar features in sports frames. To achieve robust performance, a good clustering method and an appropriate codebook size need to be determined. If a codebook size is too small (*i.e.* the value of k is too small), a number of different features could be generally grouped into one cluster. Obviously, this cannot form a good vocabulary and makes the results worse. In contrast, a too large codebook size leads to similar features could be scattered into many clusters making non-sense final results. The codebook size problem is examined carefully in our experiment section. The results are then used to determine a best codebook size for the dataset. Discussions on the relationship between the kind of sports and the codebook size are also mentioned.

Here we use K-means as our clustering method due to its good performance reported in [3]. However, in our empirical experience, when dealing with a large dataset, K-means encounters a number of limitations and the clustering does not give reasonable results. Based on our analysis, we claim that in large datasets, K-means with arbitrary initial cluster-centers cause some blank clusters, which leads to a very bad performance for the system. To avoid this issue, we apply K-means ++ [4] instead. In our experiment results, K-means ++ is proven to be a suitable method for large datasets.

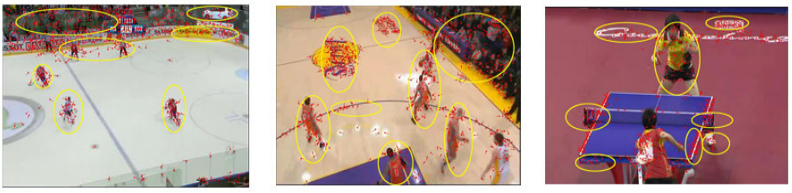

Fig. 3. Similar features are grouped by K – means++

2.3 Histogram Representation

After setting up the visual word vocabulary, in this step, each frame in the training set is reconstructed according to these codebook words. It is then represented by a histogram, which actually counts the number of visual words appearing in the frame (Fig. 1). For instance, in a 100m sprint video, after two steps mentioned above, the features of a frame, named *A*, has been extracted. Assuming that *A* contains n key points, its key point descriptors set is formed as a set of n vectors. Each vector has 128 dimensions based on the SURF-128 descriptor. With each vector, we find its nearest visual word using a distance metric and it means the nearest visual word found happens to appear in *A*. After repeating for all n vectors, we reconstruct *A* as a histogram of visual words appearing in it. Each bin of the histogram is the number of occurrences of a visual word in *A*. For example, in Fig. 1, visual words: 1, 2, 7,…, 3, 200, 101, *k* belong to *A*. In our experiments, three distance metrics: L1, L2 and CHI2 (χ^2) are compared to find the most appropriate metric.

Denote $S_1 = (u_1, u_2, ..., u_m)$ and $S_2 = (w_1, w_2, ..., w_m)$ is the vector representation of two key point descriptors S_1 and S_2

The formula for each distance metric is showed bellow:

$$\text{L1: } D(S_1, S_2) = \sum_{i=1}^{m} |ui - wi| \tag{2}$$

$$\text{L2: } D(S_1, S_2) = \sqrt{\sum_{i=1}^{m} (ui - wi)^2} \tag{3}$$

$$\text{CHI2: } D(S_1, S_2) = \frac{1}{2} \sum_{i=1}^{m} \frac{(u_i - w_i)^2}{u_i + w_i} \tag{4}$$

2.4 Multi-class Classifier

In the last step, we use SVM as a multi–class classifier for our training dataset. The input of this step is a set of <visual word histogram, label> where the label determines the class of a frame. We use one-against-one as approach for multiclass problem where each classification gives one vote to the winning class and the frame is labeled with the class having most votes. In our experiments, we compare three SVM

Kernels: LINEAR, RBF (Radial Basis Function) and POLY (Polynomial) to find the best kernel for our dataset.

3 Experimental Studies and Results

The dataset is a collection of 600 sports videos. The dataset has a total of more than 6000 minutes in length and contains 10 different classes of sports, including 100m sprint, badminton, basketball, fencing, football, hockey, swimming, table tennis, tennis and volleyball. We randomly collected all of these videos from youtube with variation in size and quality. The videos are collected from a large variety of tournament and matches for all sports classes. For example, football videos are collected from UEFA Euro 2012 qualifying, Copa America 2011, Premier League 2010/11, Series A 2010/11, Primera Liga 2010/11, UEFA Champion League 2010/11 and Seagame 24. For each sport, we set up pre-defined shot views based on the camera views. Then, these shot views are used to build the training and testing dataset (Fig. 4). In the first two experiments, we have evaluated the distance metric and codebook size presented in Section 2 with the classification recall and precision reported. The third experiment is aimed to compare the results of three SVM Kernels. The last experiment studies the performance of each kind of sports with the variation of codebook size. Recall and Precision values at each experiment are taken at threshold 0.5.

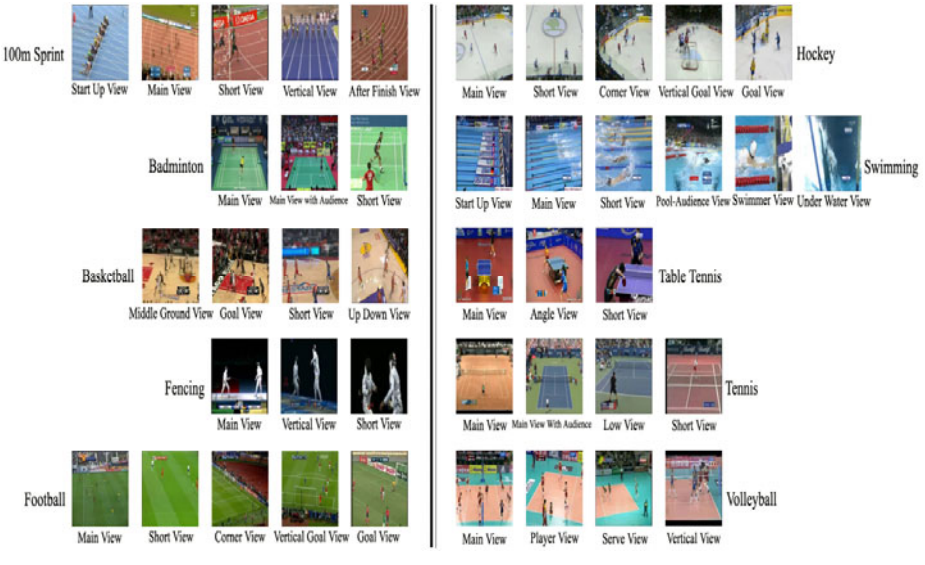

Fig. 4. Pre-defined Shot Views for 100m-Sprint, Badminton, Basketball, Fencing, Football, Hockey, Swimming, Table Tennis, Tennis and Volleyball

3.1 Experiment 1

In the first experiment, we evaluated the performance of three distances: *L1*, *L2* and *CHI2* to the classification. We repeated the experiment with different Codebook sizes.

The SVM-kernel was set to RBF. Tab. 1 reports the average recall and precision values of *10* sport classes for each distance. The results showed that *L2* is the best distance in all cases. The second best is *CHI2* followed by the *L1* distance.

Table 1. Comparing the performance of three distances: L1, L2 and CHI2

Metric	Recall	Precision
L1	92.41%	90.13%
L2	94.04%	96.16%
CHI2	92.84%	90.60%

3.2 Experiment 2

The second experiment is aimed to find the impact of the codebook size to the performance of the classification. In this experiment, *L2* was chosen as the distance metric. The experiment repeated on *9* codebook sizes. In each codebook size, the mean values of three SVM-kernels: Linear, RBF, and Poly are obtained. Tab. 2 shows the recall and precision mean values of 10 sport classes for each codebook size. The results show that the best codebook size is 3850. It is also noticed that 1650 performs a good results. Even a small size of 440 also provides acceptable results. These two codebook sizes are studied in more details in the experiment 4. The running time corresponding to each codebook size is also provided. The last column in Tab. 2 reports the testing of average running time in seconds for each frame. Depending on the application, the best parameters are chosen to compensate the trade-off between the performance and the running time.

Table 2. Comparing the impact of the codebook size

Codebook Size	Recall	Precision	Average time (sec)
110	90.01%	88.86%	0.28
220	92.41%	94.85%	0.41
440	93.48%	95.93%	0.52
880	93.56%	96.29%	0.86
1650	94.32%	97.18%	1.45
2200	94.01%	97.15%	1.86
3300	93.83%	96.93%	2.73
3850	94.28%	97.42%	3.45
4400	94.04%	97.43%	4.88

The last column shows test result on the average running time, taken in a condition of Intel core i3, 2.67ghz and 4Gb of RAM.

3.3 Experiment 3

In this experiment, we compare the results of different SVM-kernels. L2 is the distance metric used. The test procedure is repeated for all of the codebook size. The result is represented by two values indicating the recall and precision. Tab. 3 reports that RBF is always the best kernel for all size. When the dictionary size increases, the performance of LINEAR is improved significantly. This is reasonable according to the [5] experiment conclusion stating that when number of features is large, the data is no needed to map to a higher dimensional space, which means the nonlinear mapping does not improve the performance so much.

Table 3. Comparing the performance of three SVM-kernels: LINEAR, RBF and POLY

Codebook Size	LINEAR		RBF		POLY	
	Re	Pre	Re	Pre	Re	Pre
110	88.8%	87.7%	90.7%	90.0%	90.6%	88.9%
220	91.5%	93.8%	93.1%	95.5%	92.6%	95.2%
440	93.2%	96.0%	93.9%	96.0%	93.3%	95.8%
880	93.2%	96.0%	94.1%	96.7%	93.4%	96.2%
1650	94.3%	97.5%	94.7%	97.3%	94.0%	96.8%
2200	94.1%	97.5%	94.6%	97.3%	93.4%	96.7%
3300	94.2%	97.1%	94.8%	97.3%	92.5%	96.4%
3850	95.1%	97.8%	95.4%	97.8%	92.4%	96.7%
4400	94.9%	97.8%	95.1%	97.8%	92.2%	96.6%

3.4 Experiment 4

Finally, we study the performance of each sport with the variation of codebook sizes. We use *L2* is the distance metric, and RBF is the SVM Kernel. The experiment is repeated with 4 sizes of codebook: 110, 440, 1650 and 3850 (the best code book size). As the result, in Tab. 4, we can observe that badminton, basketball, fencing and table tennis have achieved very high results even if the codebook size is small. When we increase the size, the result does not improve much. This can be explained by the following analysis: these four sports have 2 common characteristics. First, they all contain few numbers of pre-defined shot views. Second, their frames are quite similar with a high rate of repetition. Thus, when represented by the bag of words model, the dictionaries with small size can handle these sports with a high recall and precision values. The best result belongs to Fencing with recall and precision rates are up to 99% and 100%. The poorest is tennis with the recall of 91% and the precision of 96%.

Fig. 6 reports the confusion matrix of 10 sport classes at the best experiment parameters. It shows that some of the most confusing cases are between badminton and tennis, tennis and football, and hockey and volleyball. These confusions are caused by the similarity in sports views and features (shown in Fig. 5). In Fig. 5a, the boundary of the ground and the net is not clear. Thus, a small number of features are taken. This could be confused with short views in football videos. In Fig. 5b, the

confusions could come from the picket along the play ground making the system recognize it as the volley ball net-picket. In Fig. 5c, the shot is very similar to short view of tennis (shown in Fig. 4).

a. Tennis - Football b. Hockey - Volleyball c. Badminton - Tennis

Fig. 5. Common confusion cases

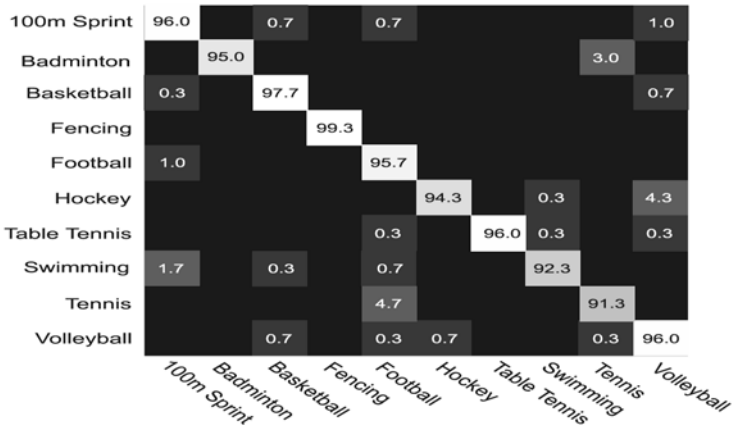

Fig. 6. Confusion matrix of 10 sports at our best experiment parameters

Table 4. Performance of each sport with the variation of codebook size

Code book size	Badminton		Basketball		Fencing		Table Tennis	
	Re	Pre	Re	Pre	Re	Pre	Re	Pre
110	92%	100%	98%	96%	96%	100%	95%	93%
440	94%	100%	98%	97%	99%	100%	97%	99%
1650	94%	100%	98%	97%	99%	100%	96%	100%
3850	95%	100%	98%	98%	99%	100%	96%	100%

4 Conclusions and Future Works

The paper has presented a novel approach to classify sport videos. The proposed method consists of 4 main steps, including descriptors detected and extracted by SURF, visual word vocabulary formed up, the histogram representation constructed and the multi-class classifier used. We have collected a large real-world dataset with a high diversity including 600 videos with a total of more than 6000 minutes for 10

different kinds of sports. As shown in the experiment results, our system shows the Bag of Words model is highly appropriated with sports video shot classification problem. Extensive experiment setups are demonstrated to the advantages of different parameters such as: codebook sizes, classifier kernel. In future, we are going to integrate more sports into the dataset. We will try to improve our model to speed up the running time as well as avoid confusions. We also would like to integrate with state-of-the-art shot boundary detection to automatically provide the shots for classification.

References

1. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision 60, 91–110 (2004)
2. Bay, H., Ess, A., Tuytelaars, T., Gool, L.V.: Speeded-Up Robust Features (SURF). Computer Vision and Image Understanding 110(3), 346–359
3. Bosch, A., Zisserman, A., Muñoz, X.: Scene Classification via pLSA. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3954, pp. 517–530. Springer, Heidelberg (2006)
4. Arthur, D., Vassilvitskii, S.: k-means++: the advantages of careful seeding. In: Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 1027–1035 (2007)
5. Hsu, C.-W., Chang, C.-C., Lin, C.-J.: A Practical Guide To Support Vector Classification. Technical Report, Department of Computer Science and Information Engineering, National Taiwan University, Taipei (2003)
6. Csurka, G., Dance, C.R., Fan, L., Willamowski, J., Bray, C.: Visual Categorization with Bags of Keypoints. In: ECCV 2004 (2004)
7. Sivic, J., Russell, B.C., Efros, A.A., Zisserman, A., Freeman, W.T.: Discovering object categories in image collections. In: ICCV 2005 (2005)
8. Grauman, K., Darrel, T.: The pyramid match kernel: discriminative classification with sets of image features. In: ICCV 2005 (2005)
9. Fei-Fei, L., Perona, P.: A Bayesian Hierarchical Model for Learning Natural Scene Categories. In: CVPR 2005 (2005)
10. Moncrieff, S., Venkatesh, S., Dorai, C.: Horror film genre typing and scene labeling via audio analysis. In: ICME 2003 (2003)
11. Takagi, S., Hattori, S., Yokoyama, K., Kodate, A., Tominaga, H.: Sports Video Categorizing Method Using Camera Motion Parameters. In: 2003 International Conference on Multimedia and Expo. (ICME 2003), vol. 2 (2003)
12. Duan, L.-Y., Xu, M., Tian, Q.: Semantic Shot Classification in Sports Video. In: Proc. of SPIE Storage and Retrieval for Media Database 2003, pp. 300–313 (2003)
13. Nam, J., Alghoniemy, M., Tewfik, A.H.: Audio-visual content-based violent scene characterization. In: ICIP 1988 (1998)
14. Lang, C., Xu, D., Jiang, Y.: Shot Type Classification in Sports Video Based on Visual Attention. In: ICCINC 2009 (2009)
15. Zhang, N., Guan, L.: An efficient framework on large-scale video genre classification. In: MMSP 2010 (2010)