MIC-TJU at MediaEval Violent Scenes Detection (VSD) 2014

Bowen Zhang, Yun Yi, Hanli Wang, Jian Yu
Department of Computer Science and Technology
Tongji University, Shanghai 201804, P. R. China
{102310,13yiyun,hanliwang,yujian}@tongji.edu.cn

ABSTRACT

The task of Violent Scenes Detection requires creating a system to detect segments which contain physical violence in both movies and videos found on the web, which is a very challenging task due to camera jitters in hand-shot videos and free shot boundary in movies and web videos. In this paper, we present a novel system by combining shot boundary detection, feature extraction in both audio and video domains, Bag-of-Words model and Support Vector Machine. The key part of system lies in trajectory based features that are calculated around robust optical flows. These flows are extracted by a novel salient keypoint trajectory algorithm. According to our results, a good detection performance can be achieved by using trajectory based features combined with dense SIFT and MFCC.

1. INTRODUCTION

Violent Scenes Detection (VSD) is a challenging task which requires teams to build a high performance system to automatically detect video segments containing violence. VSD 2014 contains two different sub-tasks: main task and generalization task. A brief introduction to the dataset for training and testing as well as evaluation metrics of these two sub-tasks is given in [4]. In this paper, we discuss the techniques and algorithms employed by our system, as well as the system architecture and evaluation results.

2. SYSTEM DESCRIPTION

The architecture of the proposed system is shown in Fig. 1. We adopt the Bag-of-Words (BoW) framework with Gaussian Mixture Model (GMM), Fisher Vector (FV) and Support Vector Machine (SVM). A threshold based video shot boundary detector is firstly used to detect video shot boundaries [6]. After that, we extract features from audio and video. FV are then used to encode video and audio features into a single high dimensional vector using a codebook generated by a GMM. Since it is observed that fusion has a great influence on the final results, different fusion methods are used to fuse vectors from different features. Because SVM with linear kernel shows good performances with FV, it is employed as the classifier of our system [1][5].

MediaEval 2014 Workshop, October 16-17,2014,Barcelona,Spain

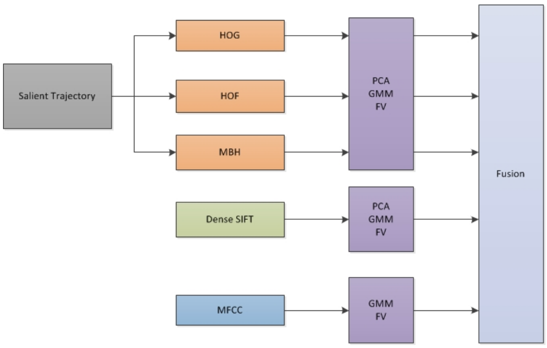

Figure 1: Overview of MIC-TJU system for VSD 2014.

2.1 Shot Boundary Detection

In VSD 2014, there are no video shot boundaries provided, neither for movies nor web videos. This causes difficulties for feature extraction and encoding. In order to address this issue, we employ the shot boundary detection method presented in [6], which adopts difference of histograms using an adaptive threshold. Specifically, the difference of histograms between two adjacent frames is firstly computed. We set a range of 15 frames ahead of the current frame to compute standard variance (STD) and mean. If the STD is lower than a specific value namely T_{vb}, it means that there are few fluctuations in these 15 frames. These frames can be used to adapt video shot boundary thresholds. In this work, T_{vb} is set to 500,000, which empirically shows good results. In order to enhance the robustness of shot boundary detection, we use a method based on two thresholds to detect both hard cuts and gradual changes. The lower threshold is used to detect gradual changes and the higher one is for hard cuts. These two adaptive thresholds are computed based on the aforementioned mean of previous differences of histograms. A hard cut will be detected if the difference of histograms between the current frame and the previous frame exceeds the corresponding threshold for hard cut detection.

2.2 Feature Extraction

For feature extraction, two different kinds of video features are used including trajectory based features and one appearance feature.

2.2.1 Video Features

Firstly, salient keypoint trajectories are generated to track

Table 1: Configuration of runs of MIC-TJU.

Run	Trajectory based Features	Appearance Feature	Audio Feature	Fusion	Weights
1	HOG,HOF,MBH	-	MFCC	Late Fusion	4:1
2	HOG,HOF,MBH	Dense SIFT	MFCC	Double Fusion	4:1
3	HOG,HOF,MBH	Dense SIFT	MFCC	Double Fusion	1:1
4	HOG,HOF,MBH	Dense SIFT	MFCC	Late Fusion	4:1:1
5	HOG,HOF,MBH	Dense SIFT	MFCC	Late Fusion	1:1:1

human actions at multiple spatial scales [5]. Then, camera motion elimination [5] is utilized to further improve the robustness of the trajectories. To encode human motions accurately and efficiently, the Histogram of Oriented Gradient (HOG), Histogram of Optical Flow (HOF) and Motion Boundary Histogram (MBH) are employed with the FV model being utilized to aggregate these three features [5]. The dimensions of these three descriptors are 96 for HOG, 108 for HOF and 192 for MBH, respectively. On the other hand, regarding the appearance feature, we use densely extracted SIFT features. We compute SIFT descriptors every 60 video frames at multiple scales on a dense grid (*i.e.*, 21×21 patches with 4 pixel steps and 5 scales) [3].

After the extraction of descriptors, these feature vectors are normalized with the signed square root, and then, PCA is individually applied to each of these three feature vectors (HOG, HOF and MBH) to reduce to half of the original dimension. Then, FVs are computed to construct a codebook for each descriptor. We compute one FV over the complete video, and apply signed square root normalization which is able to significantly improve the recognition performance in combination with linear SVM.

As far as classification is concerned, linear SVM is employed in this work and early fusion is performed to generate the final feature vector by concatenating the aforementioned four feature vectors (HOG, HOF, MBH and dense SIFT) into a single one. In our implementation, the standard linear LIBSVM is used with the penalty parameter C equal to 100, which has shown to exhibit good performances.

2.2.2 *Audio Features*

Due to auditory clues in segments which contain violent scenes, features of audio segments should be considered. Therefore, we adopt the popular Mel-Frequency Cepstral Coeffcients (MFCC) algorithm [2]. The time window for each MFCC is 32 ms and there is 50% overlap between two adjacent windows. To fully utilize the discrimination ability of MFCC, we integrate delta and double-delta of MFCC vector into the original MFCC vector to generate a 60-dimensional MFCC vector. In order to represent a whole audio file as a single vector, we adopt the classic BoW framework, where FV and GMM are used. Linear LIBSVM is used as the classifier for audio features with the penalty parameter C equal to 100.

2.3 Experimental Setup

The configuration of our submitted five runs are summarized in Table 1. Regarding the late fusion, an arithmetic sum of scores outputted from SVM for video features (trajectory based features and appearance feature) and audio feature is calculated; for double fusion, first we do early fusion of video features, and then late fusion of video and audio features. The weight setting segmented by colon in Table 1 stands for the weights applied to different kinds of features during late fusion.

3. RESULTS AND DISCUSSIONS

We submit five runs with the results given in Table 2 using the MAP2014 measure. The comparison of run1 and run4 show that the dense SIFT feature can help improve the recognition performance in the generalization task. However, there is a performance drop in the main task. The reason for this is that the late fusion strategy and weights assignment are sub-optimized for dense SIFT in the main task. By comparing run2 vs. run3 as well as run4 vs. run5, we conclude that different weights assignment will affect the recognition performances, and the optimum weight setting differs for different datasets. In general, we obtain better results in the generalization task than the main task. One reason for this is that the video shots in the generalization task do not change as frequent as that in the main task, which improves the performance of trajectory based features. It also indicates that the main task is more challenging than the generalization task.

Table 2: Results of MIC-TJU on MAP2014.

Run	Main Task	Generalization Task
1	44.17%	56.01%
2	43.07%	56.52%
3	44.60%	55.56%
4	39.23%	56.62%
5	38.50%	56.00%

4. REFERENCES

[1] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM Trans. Intell. Syst. Technol.*, 2(3):27:1–27:27, Apr. 2011.

[2] S. Davis and P. Mermelstein. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Trans. Acoust., Speech, Signal Processing.*, 28(4):357–366, Aug. 1980.

[3] D. Oneata, J. Verbeek, and C. Schmid. Action and event recognition with fisher vectors on a compact feature set. In *ICCV'13*, pages 1817–1824, 2013.

[4] M. Sjöberg, B. Ionescu, Y.-G. Jiang, V. L. Quang, M. Schedl, and C.-H. Demarty. The mediaeval 2014 affect task: Violent scenes detection. In *MediaEval 2014 Workshop*, 2014.

[5] H. Wang and C. Schmid. Action recognition with improved trajectories. In *ICCV'13*, pages 3551–3558, 2013.

[6] D. Zhang, W. Qi, and H. J. Zhang. A new shot boundary detection algorithm. In *PCM'01*, pages 63–70. 2001.