NII-UIT at MediaEval 2015 Affective Impact of Movies Task

Vu Lam
University of Science,
VNU-HCM
lqvu@fit.hcmus.edu.vn

Sang Phan
National Institute of
Informatics, Japan
plsang@nii.ac.jp

Duy-Dinh Le
National Institute of
Informatics, Japan
ledduy@nii.ac.jp

Shin'ichi Satoh
National Institute of
Informatics, Japan
satoh@nii.ac.jp

Duc Anh Duong
University of Information
Technology, VNU-HCM
ducda@uit.edu.vn

ABSTRACT

Affective Impact of Movies task aims to detect violent videos and affective impact on viewers of that videos [9]. This is a challenging task not only because of the diversity of video content but also due to the subjectiveness of human emotion. In this paper, we present a unified framework that can be applied to both subtasks: (i) induce affect detection, and (ii) violence detection. This framework is based on our previous year's Violent Scene Detection (VSD) framework. We extended it to support affect detection by training different valence/arousal classes independently and combine them to make the final decision. Besides using internal features from three different modalities: audio, image, and motion, in this year, we also incorporate deep learning features into our framework. Experimental results show that our unified framework can detect violent videos and its affective impact with a reasonable accuracy. Moreover, using deep features can significantly improve the detection performance of both subtasks.

1. INTRODUCTION

Detecting affective impact of movies requires combining multimedia features. For example, a violent video of car-chase can be detected by searching for evidences such as fast moving of cars or possibly the sound of gun shooting. To this end, we have developed a framework that supports combining features from multiple modalities for violent scene detection. We consider the induced affect detection as a multi-class classification task. Therefore, our framework can be applied to predict the valence and arousal class of a video as well. In general, our framework consists of three main components: feature extraction, feature encoding, feature classification. An overview of our framework is shown in Fig 1.

2. FEATURE EXTRACTION

2.1 Image Features

At first, we scale the original video into 320x240 pixels and then sample frames from video at every second. We use the standard SIFT feature with Hessian Laplace interest point detector to extract features from each frame [6]. Each frame is represented using the Fisher Vector encoding [7]. We use the average pooling strategy to aggregate frame-based feature into the final video representation, which has 40,960 dimensions.

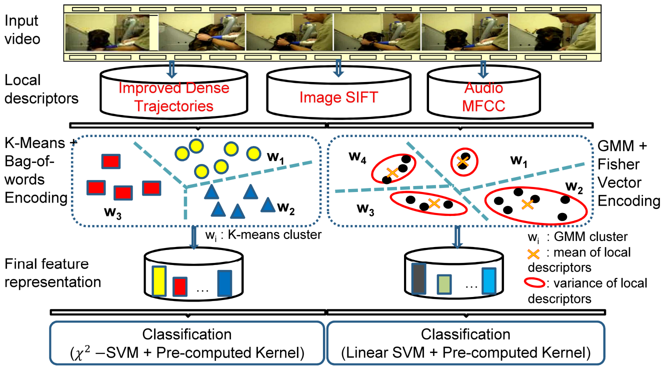

Figure 1: Our framework for extracting and encoding local features.

2.2 Motion Feature

We use the Improved Trajectories [10] to extract dense trajectories. A combination of Histogram of Oriented Gradients (HOG), Histogram of Optical Flow (HOF) and Motion Boundary Histogram (MBH) is used to describe each trajectory. We encode HOGHOF and MBH features separately using the Fisher Vector encoding. The codebook size is 256, trained using a Gaussian Mixture Model (GMM). The feature representation of each descriptor after applying PCA has 65,536 dimensions.

2.3 Audio Feature

We use the popular Mel-frequency Cepstral Coefficients (MFCC) for extracting audio features. We choose a length of 25ms for audio segments and a step size of 10ms. The 13-dimensional MFCC vectors along with each first and second derivatives are used for representing each audio segment. Raw MFCC features are also encoded using Fisher vector encoding. We use a GMM to train the codebook with 256

MediaEval 2015 Workshop, Sept. 14-15, 2015, Wurzen, Germany

Table 1: Submitted violence detection runs and official results.

Run	Features	Validation Results (mAP)	Official Results (mAP)
1	HOGHOF+MBH+MFCC	0.2200	0.2039
2	HOGHOF+MBH+SIFT+MFCC	0.2094	0.2087
3_ext	HOGHOF+MBH+MFCC+VDFULL	0.2457	0.2380
4_ext	HOGHOF+MBH+MFCC+VDFULL+HBM	**0.2499**	0.2196
5_ext	HOGHOF+MBH+MFCC+VDFULL+VDFC6 +VDFC7+FOHGOH+HBM+TFIS+CCFM	0.1930	**0.2684**

Table 2: Submitted induced affect detection runs and official results.

Run	Features	Decision Strategy	Validation Results (mAP)		Official Results (Accuracy)	
			Valence	Arousal	Valence	Arousal
1	HOGHOF+MBH+SIFT+MFCC	MAXREL	0.4148	0.3998	39.823	35.723
2	HOGHOF+MBH+SIFT+MFCC	MAX	0.4148	0.3998	41.653	**55.908**
3_ext	HOGHOF+MBH+SIFT+MFCC +VDFULL+VDFC6+VDFC7	MAXREL	0.4376	0.3958	**42.956**	55.677
4_ext	HOGHOF+MBH+SIFT+MFCC +VDFULL+VDFC6+VDFC7	MAX	0.4376	0.3958	42.914	55.656

clusters. For audio features, we do not use PCA. The final feature descriptor has 19,968 dimensions.

2.4 Deep Learning Feature

We use the popular DeepCaffe [3] framework to extract image features. We used the pre-trained deep model provided by Simonyan and Zisserman [8]. This model was trained on ImageNet 1,000 concepts [2]. As suggested in [4], we selected the neuron activations from the last three layers for the feature representation. The third and second-to-last layer has 4,096 dimensions, while the last layer has 1,000 dimensions corresponding to the 1,000 concept categories in the ImageNet dataset. We denote these features as VDFC6, VDFC7, and VDFULL in our experiments.

2.5 Features from Past VSD Tasks

For the violent detection task, we also consider using featurs from past VSD tasks as external features. In particular, we use the features that were extracted in the VSD 2014 task for training the violent detector. These features include SIFT, Dense Trajectories (HOGHOF and MBH descriptors) and Audio MFCC which achieved the runner-up performance in VSD 2014 [5]. We denote these features as FOHGHOF, HBM, TFIS and CCFM in our experiments.

3. CLASSIFICATION

LibSVM [1] is used for training and testing our affective impact detectors. For features that are encoded using the Fisher vector, we use linear kernel for training and testing. For deep learning feature, χ^2 kernel is used.

We divide the training videos into two subset. The first 3,072 videos are used for training the model, while the remaining 3,072 videos are used for validation. To learn the decision threshold of each detector, we sample this threshold in the range from 0 to 1 with the step size of 0.01, and select the value that maximizes the F1 score.

In order to generate the decision for valence or arousal detection, we need to make the decision from the predictions of all valence or arousal classes. To this end, we propose using two strategies: (1) MAX: select the class that has the highest prediction; (2) MAXREL: select the class that has the highest relative improvement from the learned threshold.

4. SUBMITTED RUNS

At first, we use the late fusion with average weighting scheme to combine features from different modalities. After that we select the runs that have the top performance on the validation set to submit. The list of submitted runs for each subtask and its validation results can be seen on Table 1 and Table 2.

5. RESULTS AND DISCUSSIONS

The official results for each subtask are shown on the last column of Table 1 and Table 2. For the violence detection task, we observe that the results of combining multiple features are more stable. For example, on the validation set, the run that combines all available features has the lowest performance. However, on the test set, this run achieves the best performance. This can be due to the fact that we only select one split for validation. For both subtasks, combining with deep learning features can significantly improve the detection performance. For the induced affect detection task, we found that the strategy using the max detection score tends to have more stable performance. The best valence detection performance is obtained by combining all internal features with all deep learning feature using the max relative improvement strategy.

6. ACKNOWLEDGEMENTS

This research is partially funded by Vietnam National University Ho Chi Minh City (VNU-HCM) under grant number B2013-26-01.

7. REFERENCES

[1] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011.

[2] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition (CVPR)*, pages 248–255. IEEE, 2009.

[3] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the ACM International Conference on Multimedia*, pages 675–678. ACM, 2014.

[4] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[5] V. Lam, D. Le, S. Phan, S. Satoh, and D. A. Duong. NII-UIT at mediaeval 2014 violent scenes detection affect task. In *Working Notes Proceedings of the MediaEval 2014 Workshop, Barcelona, Catalunya, Spain, October 16-17, 2014.*, 2014.

[6] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[7] J. Sánchez, F. Perronnin, T. Mensink, and J. Verbeek. Image classification with the fisher vector: Theory and practice. *International journal of computer vision*, 105(3):222–245, 2013.

[8] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[9] M. Sjöberg, Y. Baveye, H. Wang, V. L. Quang, B. Ionescu, E. Dellandrea, M. Schedl, C.-H. Demarty, and L. Chen. The mediaeval 2015 affective impact of movies task. *In MediaEval 2015 Workshop, Wurzen, Germany, Septemper 14-15 2015.*

[10] H. Wang and C. Schmid. Action recognition with improved trajectories. In *International Conference on Computer Vision (ICCV)*, pages 3551–3558. IEEE, 2013.