# KIT at MediaEval 2015 – Evaluating Visual Cues for Affective Impact of Movies Task

Marin Vlastelica P.*, Sergey Hayrapetyan*, Makarand Tapaswi*, Rainer Stiefelhagen

Computer Vision for Human Computer Interaction, Karlsruhe Institute of Technology, Germany

marin.vlastelicap@gmail.com, s.hayrapetyan@hotmail.com, tapaswi@kit.edu

## ABSTRACT

We present the approach and results of our system on the MediaEval Affective Impact of Movies Task. The challenge involves two primary tasks: affect classification and violence detection. We test the performance of multiple visual features followed by linear SVM classifiers. Inspired by successes in different vision fields, we use (i) GIST features used in scene modeling, (ii) features extracted from a deep convolutional neural network trained on object recognition, and (iii) improved dense trajectory features encoded using Fisher vectors commonly used in action recognition.

## 1. INTRODUCTION

As the number of videos grow rapidly, automatically analyzing and indexing them is a topic of growing interest. One interesting area is to analyze the affect such videos have on viewers. This can lead to improved recommendation systems (in case of movies) or help improve overall video search performance. Another task is to predict the amount of violent content in the videos, thus supporting automatic filters for sensitive videos based on viewer age. The MediaEval 2015 task – "Affective Impact of Movies" [9] studies these two areas.

The affect task is posed as a classification problem on a two-dimensional arousal-valence plane, where each dimension is discretized to 3 values (classes). On the other hand, the detection task is presented as a detection problem. Please refer to [9] for task and dataset details.

## 2. APPROACH

In this section we describe the features and classifiers we use to analyze the affective impact of movies.

### 2.1 Development splits

The development set consists of 6144 short video clips obtained from 100 different movies. To analyze the movies we use a 5-fold cross-validation on the dataset. The data is split into 5 sets with two goals in mind: (i) the source movies in the training and test splits are different; (ii) the distribution of class labels (positive/neutral/negative) is maintained close to the original complete set.

*indicates equal contribution


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

In this way, we achieve 5 fairly independent splits for training and testing our models. The splits include differing number of movies in the training and test sets ranging from 65/35 to 91/9.

### 2.2 Descriptors and models

We focus primarily on simple visual cues to estimate the affect of videos and detect violence in them. To this end, we use three feature types and use linear SVM classifiers.

For the image-based descriptors, we extract exemplar images from the video, sampled at every 10 frames. To compensate for shot changes within the video clips we do not average the features across the video and use them directly to train our models. The video-level label is assumed to be shared across all images of the clip.

**GIST** We use GIST features that were developed in the context of scene recognition [7]. We expect these features to provide good performance on the valence task. The features are extracted on each part of an image broken down using a $4 \times 4$ grid to yield a 512 dimensional descriptor. We then train multi-class linear SVM classifiers on these features for the affect tasks (arousal and violence) and another linear SVM for the violence detection task.

**CNN features** Since the ImageNet winning method proposed by Krizhevsky, et al. [6] in 2012, deep convolutional neural networks (CNNs) have revolutionized computer vision. These networks have a large number of parameters and are trained end-to-end (from image to label) using massive datasets. The initial layers of the convolution act as low-level feature extractors, while the higher level fully connected layers start learning about object shapes.

Inspired by DeCAF [2], we use the BVLC Reference CaffeNet model provided with the Caffe framework [4] as a feature extractor. The model contains 5 convolutional layers, 2 fully connected layers and a soft-max classifier. We use the output of the last fully connected layer to obtain 4096 dimensional features for the images from video clips. Linear SVMs are trained on these features for all tasks. Owing to the complexity of the model and its ability to capture a large number of variations, we expect these features to perform well for all tasks.

**Improved Dense Trajectories** Dense trajectories are an effective descriptor for action recognition. [10] recently proposed additional steps to obtain Improved Dense Trajectories (IDT). Unlike dense trajectories, these features estimate and correct camera motion and thus obtain trajectories primarily on the foreground moving objects (often human ac-

tors). As violence in videos is often characterized by rapid motion, we anticipate these features to work well for violence detection.

Several descriptors are computed for each trajectory – Histogram of Oriented Gradients (HOG), Histogram of Optical Flow (HOF), Motion Boundary Histogram (MBH) and overall trajectory characteristics to obtain a 426 dimensional representation for each trajectory. These features are projected via PCA to 213 dimensions and finally encoded using state-of-the-art Fisher vector encoding [8]. This results in a 109,056 dimensional feature representation for the entire video. Finally, as before, we train linear SVMs using these features.

## 2.3 Software environment

The descriptors and models were developed and trained in Python and Matlab. We used the scikit-learn machine learning framework [1] in Python, which uses the liblinear SVM library [3] as the backend. For extracting features from deep neural networks we used the Caffe framework [4] which provides a simple interface for classification and feature extraction with convolutional neural networks (CNN). We extract IDT features using the provided code and implement Fisher vector encoding in Matlab.

# 3. EVALUATION

We now present and discuss the results obtained by the visual features. The best classifier parameters were obtained by cross-validation splits on the dev set.

The affect task that includes valence and arousal is treated as a multi-class classification problem (three classes each). The metric for these is the overall class prediction accuracy (acc). The violence task is a detection problem and uses average precision (ap) to evaluate different methods.

We present the results of various run submissions in Table 1. The run submissions are as follows:

- Run 1: GIST features + linear SVMs
- Run 2: IDT features + linear SVMs
- Run 3: CNN features + linear SVMs
- Run 4: Fusion-1 + linear SVMs
- Run 5: Fusion-2 + linear SVMs

Note that Run 3 and 5 constitute external runs (Ext) since they use pre-trained CNN models. All other submissions are trained solely on the development data.

We see that CNN features (Run 3) outperform the first two runs on valence and violence which involve single features. Contrary to expectations, IDT features (Run 2) perform best on arousal classification. This can be explained by passive videos often have very little motion, while active have higher.

While we expect IDT features to perform well on violence, videos annotated as violent need not have active motion and violence and can often be shots of a post-crime scene. CNN features seem to work better in this case.

**Fusion runs** Run 4 and 5 constitute fusion of different features. Run 4, the Fusion-1 scheme uses the features provided along with the dataset (IAV - image/audio/video concatenated and trained as one model), GIST and IDT. Run 5, Fusion-2 scheme includes the above along with CNN features (thus making it an external data run).

In order to fuse the different features, we choose the best models for each feature type. We then perform late fusion, where the final score for each video is a weighted combination of the individual feature predictions. We try a grid of discrete weights to generate a large number of combinations and pick the best scoring model based on cross-validation.

**Table 1: Evaluation on the test set. The first three runs use a single feature, while the latter use late fusion.**

| | Ext. | Valence (acc) | Arousal (acc) | Violence (ap) |
|---|---|---|---|---|
| Run 1 | - | 35.5 | 30.8 | 7.1 |
| Run 2 | - | 36.0 | **46.7** | 8.6 |
| Run 3 | ✓ | **38.5** | 44.7 | **10.2** |
| Run 4 | - | 35.7 | 46.7 | 10.7 |
| Run 5 | ✓ | **38.5** | **51.9** | **12.9** |

Both fusion schemes perform equal to or better than the single features. For Fusion-1 scheme, we see that IDT features get the highest weight, followed by the IAV (dataset features). In the case of Fusion-2, CNN and IDT features are weighted higher.

**Error analysis** We present a short analysis of the errors we encountered in the development set. For violent video detection, some of the difficult samples include black-and-white videos with rapid blinking. In case of affect analysis, for both valence and arousal classification cartoon scenes were often deemed colorful and classified as positive (or active) while their ground truth was neutral or negative (or passive).

# 4. CONCLUSION

We conclude that the CNN features are the best single features for studying the affective impact of movies task. Fine tuning the model, or training a model to perform video classification as in [5] could further improve the performance. Fusing the models results only in a slight improvement indicating that using other modalities such as meta-data and audio might help improve performance.

# 5. REFERENCES


[1] Python scikit-learn: machine learning framework. http://scikit-learn.org/.

[2] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. DeCAF: A Deep Convolutional Activation Feature for Generic Visual Recognition. In *International Conference on Machine Learning (ICML)*, 2014.

[3] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. LIBLINEAR: A Library for Large Linear Classification. *Journal of Machine Learning Research*, 9:1871–1874, 2008.

[4] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional Architecture for Fast Feature Embedding. *arXiv preprint arXiv:1408.5093*, 2014.

[5] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei. Large-scale video classification with convolutional neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[6] A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In *Neural Information Processing Systems (NIPS)*, 2012.

[7] A. Oliva and A. Torralba. Modeling the Shape of the Scene: A Holistic Representation of the Spatial Envelope. *International Journal of Computer Vision (IJCV)*, 42(3):145–175, 2001.

[8] F. Perronnin, J. Sánchez, and T. Mensink. Improving the Fisher kernel for large-scale image classification. In *European Conference on Computer Vision (ECCV)*, 2010.

[9] M. Sjöberg, Y. Baveye, H. Wang, V. L. Quang, B. Ionescu, E. Dellandréa, M. Schedl, C.-H. Demarty, and L. Chen. The MediaEval 2015 Affective Impact of Movies Task. In *MediaEval 2015 Workshop*, 2015.

[10] H. Wang and C. Schmid. Action Recognition with Improved Trajectories. In *IEEE International Conference on Computer Vision (ICCV)*, 2013.