Spatial Semantics and Classifier Cascades

The ANU 2011 Multimedia Event Detection System

Lexing Xie*, Xuming He†*

*Australian National University, †National ICT Australia (NICTA)

Abstract

In this paper, we describe the system developed by the Australian National University (ANU), National ICT Australia (NICTA) for multimedia event detection applied to the TRECVID-2011 video retrieval benchmark. Our system uses five audio and visual features, leverages training events with cascaded classifier training, and sees performance improvements with spatial semantic features. A summary of our submitted runs can be found below:

Run number	Short hand	Description
pRun01	01_combined_all	Combination from all features, with classifier bagging, and cascade training from SIFT.
cRun02	02_combined_nocas	Same as Run01 except without cascade training.
cRun03	03_combined_visual	Run01 without the two audio features.
cRun04	04_combined_single	Combination from all features, without classifier bagging, with SIFT.

The best run from our system ranks fourth in mean-ActualNDC, and third in mean-F1 metric, averaged over all ten events among sixty runs from nineteen teams.

1 Introduction

Multimedia event detection presents many challenging problems in computer vision, machine learning, multimedia ontologies and large-scale learning and data processing. In particular, unconstrained environments makes motion estimation and object extraction difficult; the diverse relationships between the visual and the audio track pose challenge for audio-visual integration; and the small number of training instances available makes leveraging across different event classes desirable.

In our MED 2011 system, we have explored two new aspects. We made use of visual concept annotations in different spatial granularities as well as at the global level. These

has contributed to a strong baseline in addition to color SIFT, short- and long- term audio features. Furthermore, we learn cascade event classifiers and use classifier bagging. This is to explore the intercorrelation among events, especially those from training events. Such training strategy consistently improves detection across all events. We will describe features, system components and evaluations the following sections.

2 Visual Features

We uniformly sample one frame every two seconds from the original videos, and extract the following three visual features for each video frame.

We first extract the interest point features based on Color-SIFT [5] from each frame, and cluster them into a codebook of 1000 visual words using k-means. Each frame is represented as a three-level spatial pyramid of bag of features, with 2×2 at the second level and 1×3 at the third level. For each video clip, we average the histograms across all the frames and output a 8000-dimensional feature.

We use two types of image semantic classifier results as features. The first type is from the IBM Multimedia Analysis and Search System [1]. There are 280 one-vs-all classifiers, independently trained from a collection of web images, covering categories including objects, scenes, people, image types, and so on. We use the output score of these SVM-based classifiers, obtaining an 280-dimensional feature for each video frame.

The second type of semantic features is based on the high-dimensional object feature used in Objectbank [3]. For each frame, the object-bank feature encodes the spatial/scale distribution of a set of object categories based on their detection scores. We choose 158 object categories which are relevant to the task of event recognition, and 12 scales for each image, which leads to about 13000-dimensional feature per frame. We represent each video clip by taking maximum value of each feature across all the frames.

3 Audio Features

The audio track often contains information complimentary to the video track, and is intuitively useful for events with prominent aural cues such as *Birthday party*, *Parade* and so on. Both types of features are extracted using the Yaafe toolbox [4]. We resample all audio tracks to 32,000 hertz, individual features are extracted over 32ms windows moving by 16 ms.

Our system uses both audio transient and texture features. Our short-term transient audio feature is similar to those used in the Columbia 2010 system [2]. We extract 13-band Mel-Frequency Cepstral Coefficient (MFCC) and delta-MFCC features over a short time window, we cluster the combined features into 4096 codewords using k-means, and use the code-word histogram as the representation for each video. Our long-term audio feature consists of a larger number of audio features, including: MFCC and its first and second-order derivatives, spectral features (spectral shape statistics, flatness, variation, rolloff, slope, flux, decrease), energy features (zero-crossing rate, loudness, perceptual sharpness, energy) and complex

domain onset detection. Details of each feature can be found in the toolbox reference [4]. These features are meant to capture a larger variety of sound properties over a longer period of time, and we compute the the mean and standard deviation of each feature dimension over 4 seconds with a hop of 2 seconds. We then cluster the audio texture into 256 codewords with K-means, and represent each video clip as a histogram.

4 Classifier Training

We learn SVM classifiers on each of the audio or visual features, each represented as one vector for each video. We perform kernel and hyper-parameter search on linear, polynomial and RBF kernels. Objectbank features use linear kernels due to its high dimensionality, Color SIFT histogram uses an intersection kernel that yields better performance than RBF kernel. We split the MED evaluation data into 80% for training (including hyperparameter selection), and 20% for validation and threshold tuning.

We randomly sub-sampled negative video clips to be no more than 10 times the number of positive clips for each event. Furthermore we created 5-6 bagged classifiers with different sets of negative video clips (sampling with replacement), averaged the output scores among them to reduce classifier variance.

To utilize the information from similar event classes, we designed a two-level cascading classification system based on a manually designed event class hierarchy. Specifically, we assign two "super classes" to each of the ten target event class, as shown in Table 1. The assignment is made based on semantic similarity, automatic or semi-automatic assignment can be explored. We train a first-level classifier for each super-class triplet versus the rest of the training set using SVMs. We set a conservative threshold such that more than 90% target classes (E06-15) will be passed into the next stage. The second-level classifier focuses on detecting the target class (i.e., the first sub-class) within each super-classes by filtering out other sub-classes and background.

Target Class "Super-classes"	E006 E004, E008	E007 E009, E014	E008 E004, E012	E009 E007, E012	E010 E002, E003
Target Class "Super-classes"	E011 E002, E015	E012 E004, E008	E013 E001, E008	E014 E007, E005	E015 E005, E011

Table 1: The list of superclasses for cascade training.

5 Evaluations

We evaluate each of the audio/visual features, as well as different combinations using three different metrics. Two retrieval metrics that does not need a decision threshold: AP (average

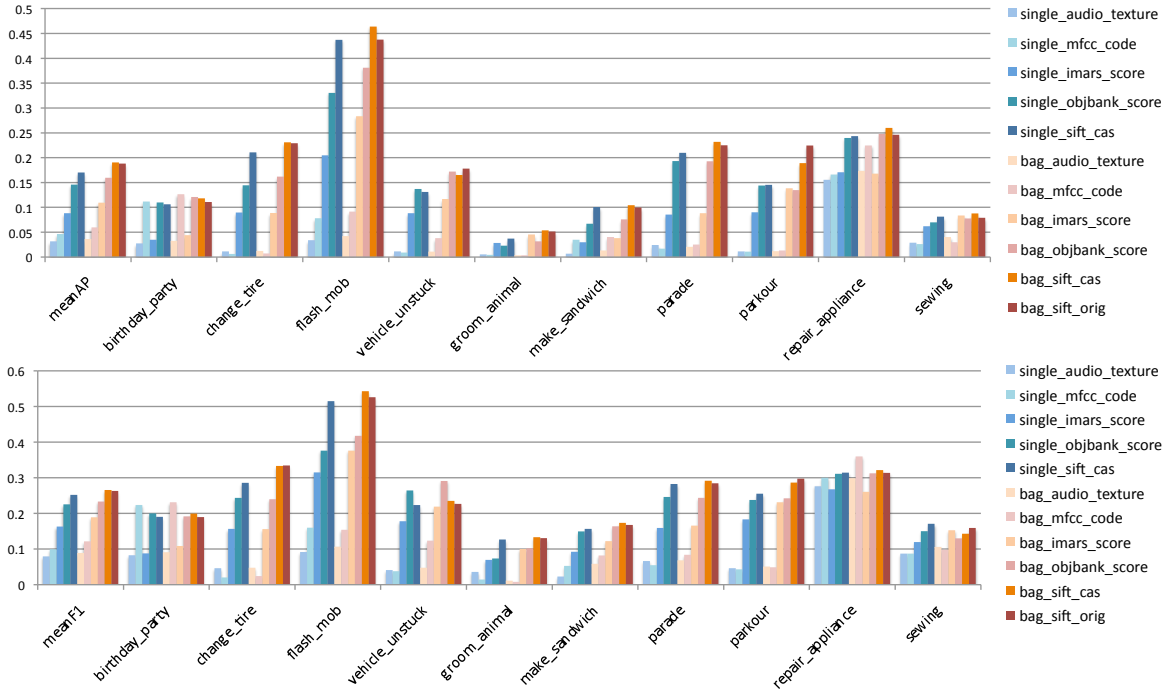

Figure 1: AP and F1 of the individual audio and visual runs. Top plot: average precision; bottom plot: F1.

precision), maximum F1, and the official MED evaluation metric normalized detection cost. Our detection thresholds are tuned on the validation using the F1 metric.

5.1 Performance of constituent audio-visual runs

Figure 1 compares classifiers trained on two types of audio features (MFCC, audio texture) and three types visual features (imars-concepts, object-bank, plus color SIFT with or without cascade training) on the test set. We can see that color SIFT is the best-performing single feature (AP=0.170), object-bank is the next (AP=0.146). MFCC features performs better than the audio texture features (AP 0.059 vs 0.036), suggesting that audio texture feature can be further fine-tuned. Classifier bagging, presented as red-shaded runs, consistently helps with an AP improvement $\sim$0.02, and cascade training on color SIFT features helps (AP from 0.188 to 0.190). The evaluation using AP and F1 leads to the same conclusions.

There is apparently large variations among different event classes. We can see that events with strong visual scene cues tend to do well (*flash mob gathering*), so as those with a distinct audio cue *birthday party*). While events that rely on objects with diverse appearances (*grooming an animal*) seem the most difficult.

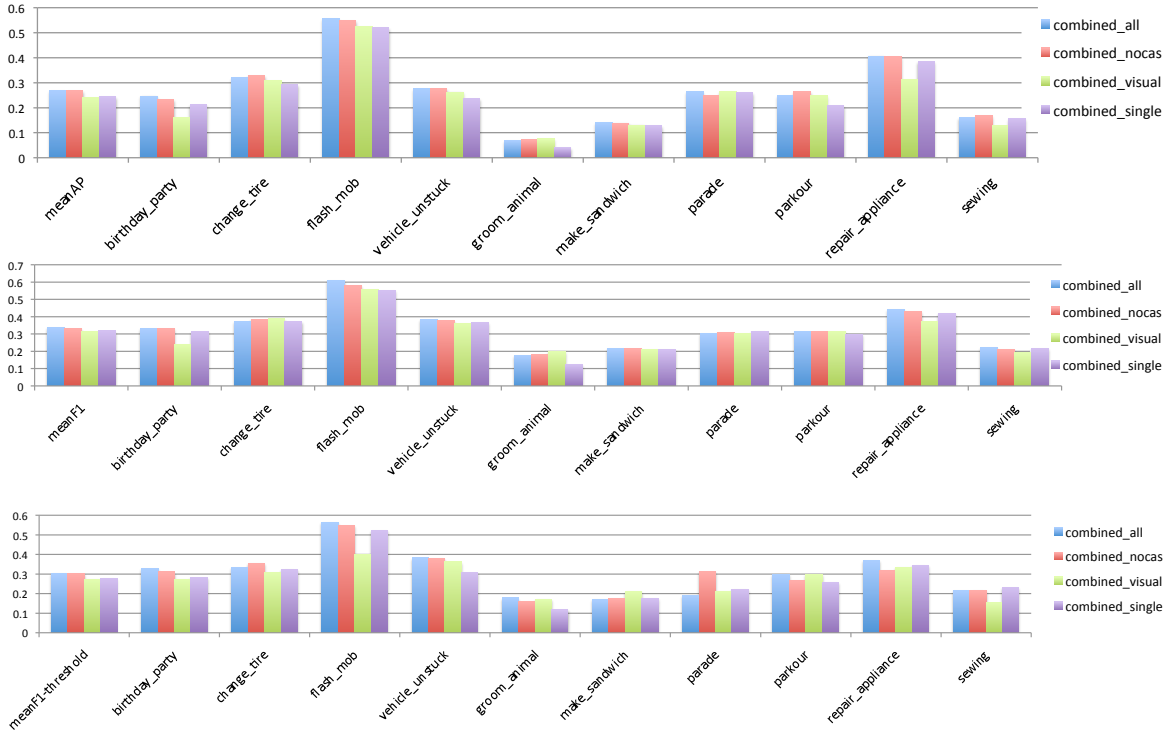

Figure 2: AP and F1 of the submitted runs. Top: Average Precision. Middle: F1 metric. Bottom: F1 metric with submission thresholds (mean-F1-threshold here correspond to Figure 3 middle plot).

5.2 Performance of combined runs

Figure 2 compares our four combined (and submitted runs) using three different metrics: AP, F1, and F1 at the submission thresholds. We can see that each of the five involved feature helped achieve the final performance, the combined AP is 0.2694 for Run01, up from 0.1905 for the best single feature (SIFT cascade). In particular, all three variants we tested among the submitted runs: cascade training, audio features, and classifier bagging all helped (Run01 perform best). There is no re-ordering of runs with the AP or max-F1 metric, but Runs 1 and 2, 3 and 4 flipped respectively upon score thresholding (Figure 2 bottom and Figure 3 middle).

5.3 Evaluation among submitted runs

Figure 3 presents the performance of the four ANU runs agains all sixty submitted runs. We can see that the relative order of runs varies greatly with respect to the evaluation metric. For example, our primary run (Run01) places 5^{th}, 4^{th} and 17^{th} on the three metrics used, and Run02 places 5^{th}, 10^{th} and 16^{th}. More analysis involving statistical testing will be needed.

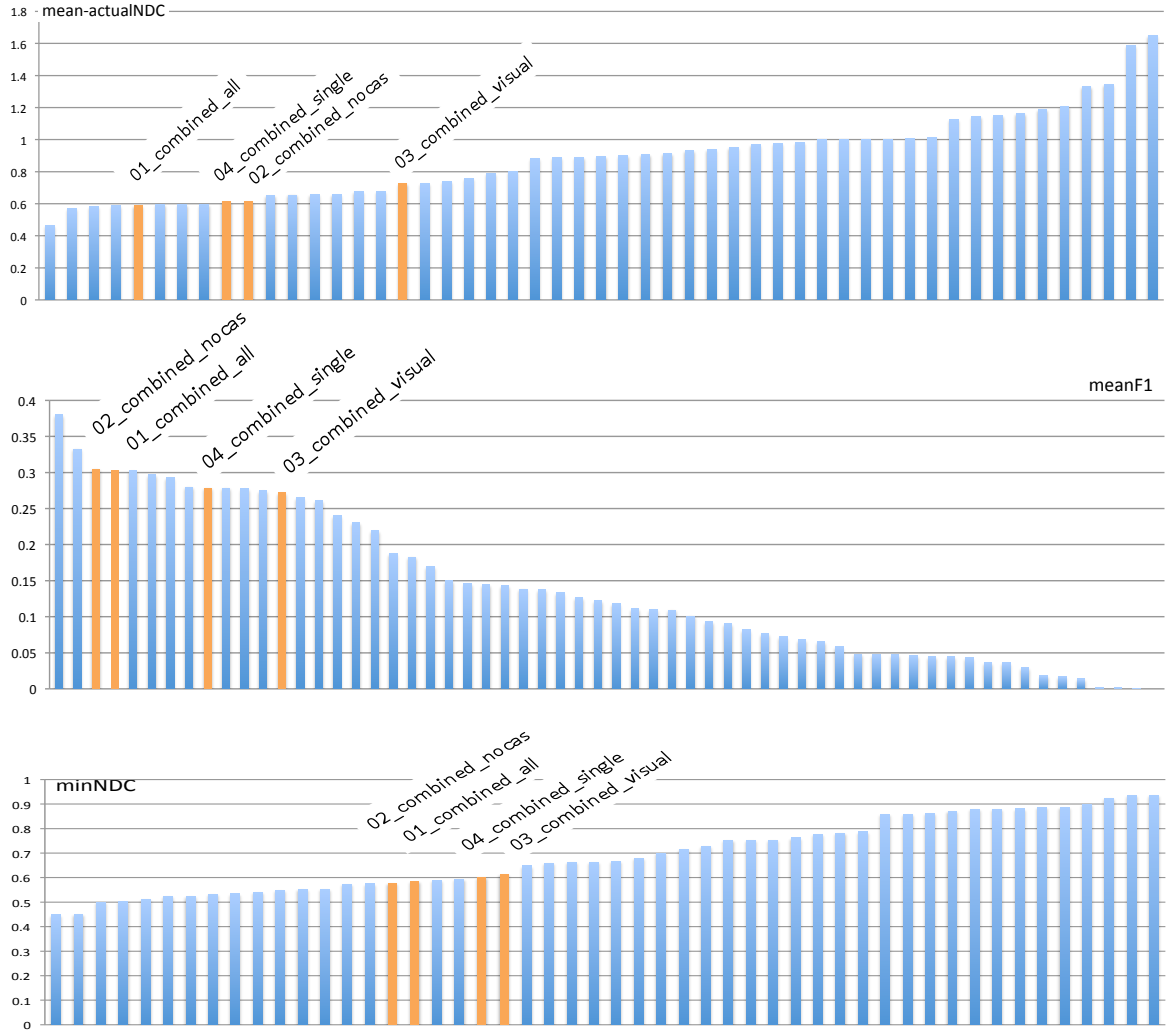

Figure 3: ANU runs (in orange) among 60 submitted runs across 19 teams, sorted by performance. Top: actual NDC, lower is better. Middle: F1 metric, higher is better. Bottom: minimum NDC, diagnostic metric, lower is better.

6 Conclusions

Our team's approach for the TRECVID2011 MED task include: a diverse set of audio and visual features that each contribute to the overall performance, spatial and global semantic features, as well as classifier bagging and cascade training that leverage the training events effectively. Future work include better use of training events, and localizing events within a long video clip.

References

[1] M. Hill, G. Hua, A. Natsev, JR Smith, L. Xie, B. Huang, M. Merler, H. Ouyang, and M. Zhou. IBM Research TRECVID-2010 video copy detection and multimedia event detection system. In *Proc. TRECVID Workshop*, 2010.

[2] Y.G. Jiang, X. Zeng, G. Ye, S. Bhattacharya, D. Ellis, M. Shah, and S.F. Chang. Columbia-UCF TRECVID2010 multimedia event detection: Combining multiple modalities, contextual concepts, and temporal matching. In *NIST TRECVID Workshop*, 2010.

[3] L.J. Li, H. Su, E.P. Xing, and L. Fei-Fei. Object bank: A high-level image representation for scene classification and semantic feature sparsification. In *Neural Information Processing Systems (NIPS)*, 2010.

[4] B. Mathieu, S. Essid, T. Fillon, J. Prado, and G. Richard. Yaafe, an easy to use and efficient audio feature extraction software. In *Proceedings of the 11th International Society for Music Information Retrieval Conference (ISMIR 2010)*, 2010.

[5] K. E. A. van de Sande, T. Gevers, and C. G. M. Snoek. Evaluating color descriptors for object and scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1582–1596, 2010.