BBN VISER TRECVID 2011 Multimedia Event Detection System

Pradeep Natarajan, Prem Natarajan, Vasant Manohar, Shuang Wu, Stavros Tsakalidis,
Shiv N. Vitaladevuni, Xiaodan Zhuang, Rohit Prasad
Speech, Language, and Multimedia Business Unit
Raytheon BBN Technologies, Cambridge, USA

Guangnan Ye, Dong Liu, I-Hong Jhuo, Shih-Fu Chang
Department of Electrical Engineering
Columbia University, New York, USA

Hamid Izadinia, Imran Saleemi, Mubarak Shah
Department of Electrical Engineering and Computer Science
University of Central Florida, Orlando, USA

Brandyn White, Tom Yeh, Larry Davis
Department of Computer Science
University of Maryland, College Park, USA

Abstract

We describe the Raytheon BBN (BBN) VISER system that is designed to detect events of interest in multimedia data. We also present a comprehensive analysis of the different modules of that system in the context of the MED 2011 task. The VISER system incorporates a large set of low-level features that capture appearance, color, motion, audio, and audio-visual co-occurrence patterns in videos. For the low-level features, we rigorously analyzed several coding and pooling strategies, and also used state-of-the-art spatio-temporal pooling strategies to model relationships between different features. The system also uses high-level (i.e., semantic) visual information obtained from detecting scene, object, and action concepts. Furthermore, the VISER system exploits multimodal information by analyzing available spoken and videotext content using BBN's state-of-the-art Byblos automatic speech recognition (ASR) and video text recognition systems. These diverse streams of information are combined into a single, fixed dimensional vector for each video. We explored two different combination strategies: *early fusion* and *late fusion*. Early fusion was implemented through a fast kernel-based fusion framework and *late fusion* was performed using both Bayesian model combination (BAYCOM) as well as an innovative a weighted-average framework. Consistent with the previous MED'10 evaluation, low-level visual features exhibit strong performance and form the basis of our system. However, high-level information from speech, video-text, and object detection provide consistent and significant performance improvements. Overall, BBN's VISER system exhibited the best performance among all the submitted systems with an average ANDC score of 0.46 across the 10 MED'11 test events when the threshold was optimized for the NDC score, and <30% missed detection rate when the threshold was optimized to minimize missed detections at 6% false alarm rate.

Description of Submitted Runs

BBNVISER-LLFeat: Uses a combination of 6 high-performing, multimodal, and complementary low-level features, namely, appearance, color, motion based, MFCC, and audio energy. We combine these low-level features using an early fusion strategy. The threshold is estimated to minimize the NDC score.

BBNVISER-Fusion1: Combines several sub-systems, each based on some combination of low-level features, ASR, video text OCR, and other high-level concepts using a late-fusion, Bayesian model combination strategy. The threshold is estimated to minimize the NDC score.

BBNVISER-Fusion2: Combines same set of subsystems as BBNVISER-Fusion1. Instead of BAYCOM, it uses a novel weighted average fusion strategy. The fusion weights (for each sub-system) are estimated for each video automatically at runtime.

BBNVISER-Fusion3: Combines all the sub-systems used in BBNVISER-Fusion3 with separate end-to-end systems from Columbia and UCF. In all, 18 sub-systems were combined using weighted average fusion. The threshold is estimated to minimize the probability of missed detection in the neighborhood of ALADDIN's Year 1 false alarm rate ceiling.

Keywords: Low-level visual features; Spatio-temporal pooling; Automatic speech recognition; Videotext OCR, Feature fusion; Early fusion; Late fusion; BAYCOM.

1 Introduction

Analysis of web videos is a challenging task since such videos are typically acquired under uncontrolled conditions with wide variations in viewpoint, lighting, and camera motion. Web videos often include overlaid audio and text content that is unrelated to the content of the video itself (e.g., rock music accompanying war footage). Further, processing large volumes of such data introduces several difficult system engineering challenges. *Bag-of-words* approaches [Csurka et al. 2004] have been effective for such tasks and have been used widely such as in Columbia's top performing MED'10 system [Jiang et al. 2010]. We take this basic approach further, by rigorously evaluating several low-level visual features for modeling appearance (e.g. SIFT [Lowe 2004]), color (e.g. RGB-SIFT [van de Sande et al. 2010]), and motion (e.g. STIP [Laptev 2005]), and also MFCC based audio features. In addition, we developed novel bi-modal features to model correlations between different audio and visual features. We explored the utility of high-level (sometimes called *semantic*) information from different modalities. Such high-level information includes scene concepts, objects, and actions from the visual stream, automatically generated transcripts of spoken content from the audio stream, and both in-scene and overlaid text in videos. We then combined these different features using several fusion techniques. Our primary findings can be summarized as follows:

- Low-level audio features and low-level visual features for modeling appearance, color, and flow complement each other and show consistent and impressive performance.

- Spatio-temporal pooling for modeling spatial and temporal relationships between low-level features shows significant performance gain over straightforward bag-of-words approaches that ignore such relationships.

- High-level visual information from scene, object, and action recognition produce performance gains, but are not consistent across all MED’11 events.

- Speech recognition is extremely effective in retrieving videos with relevant spoken content. However, the occurrence of speech varies widely between different events and even among different videos within a specific event. Therefore, when speech is available, the use of ASR transcripts improves the overall detection performance of the system.

- Video OCR is similar to ASR in terms of relative impact on system performance, but videotext is even sparser than speech content.

- Feature fusion expectedly produces significant gains over any individual feature. Further, BAYCOM is effective in optimizing performance at a specific operating point while weighted average fusion optimizes the entire DET curve.

The rest of the paper is organized as follows – in Section 2 we describe our low-level feature system in detail. In Section 3, we describe our bi-modal features. In Section 4, we discuss the high-level features that were studied. In Sections 5 and 6, we describe our ASR and videotext OCR systems. In Section 7, we present the different feature fusion strategies. We conclude with a discussion of experimental results in Section 8.

2 Bag-of-words Feature Representation

The bag-of-words based feature representation for videos consists of 4 steps – first, a set of points are sampled from the video, either on a dense, uniform grid or by detecting 2D/3D corners in the video. Second, we extract a descriptor vector from the spatio-temporal neighborhood of each detected point by aggregating gradient/flow patterns and produce a set of feature vectors that represent the local appearance/motion at different locations in the video. Next, these feature vectors are projected to a *codebook* of feature vectors, typically learnt using *k*-means clustering. Finally, the projections are aggregated to obtain a single, fixed-dimensional feature vector to represent the video. In our work, we evaluated several combinations of low-level features, coding, and pooling techniques. We now discuss the features and their combinations in detail.

2.1 Low-level Features

We considered 4 classes of low-level visual and audio features, namely – appearance, color, motion, and audio features.

2.1.1 Appearance Features

Appearance features model local shape patterns by aggregating quantized gradient vectors in grayscale images. We used the following appearance features in our system:

SIFT [Lowe 2004]: These features are among the most widely used in vision and use a difference of Gaussians (DoG) approach to detect interest points at different scales. Then a 128 dimensional feature descriptor is extracted at each point to capture local image gradients. These descriptors are scale invariant and robust to affine distortion.

SURF [Bay et al. 2008]: The speeded-up robust features (SURF) are similar in principle to SIFT, but are several times faster to extract. They are extracted using sums of 2D Haar Wavelet response and are potentially more robust to image transformations compared to SIFT.

D-SIFT [Boureau et al. 2010]: This is a dense version of SIFT where, instead of detecting interest points, the 128-dimensional feature vectors are extracted by uniformly sampling over the image. D-SIFT typically generates 3X the number of points produced by SIFT and has been shown to outperform SIFT for image classification [Boureau et al. 2010].

CHoG [Chandrasekhar et al. 2011]: The compressed histogram-of-oriented gradient features use a low bit-rate feature descriptor with a 20X reduction in bit-rate corresponding to SIFT and other features. They have shown competitive performance in image retrieval tasks.

2.1.2 Color Features

These features, proposed in [van de Sande et al. 2010], extend SIFT features by splitting the image into color planes, computing descriptors in each plane, and then concatenating them for each detected interest point. In our system, we considered 3 different color features – **RGB-SIFT** and **OpponentSIFT**, which split the image in the RGB and Opponent color spaces respectively, and **C-SIFT**, which uses C-invariants to eliminate intensity in the opponent space.

2.1.3 Motion Features

We used the following features for modeling optical flow patterns in video:

STIP [Laptev 2005]: The Space-Time Interest Points (STIP) extends the notion of spatial interest points into the spatio-temporal domain and the resulting features often correspond to the interesting events in video. The Harris interest point operator is extended to detect local structures in space-time, where the image values have significant local variations in both space and time. Spatio-temporal extents of the detected events are estimated and their scale-invariant spatio-temporal descriptors are computed from image gradients and optical flow.

D-STIP: These features are similar to STIP, except that the points are sampled from a uniform spatio-temporal grid. They are the slowest features in our system to extract and take 3X as long as STIP for extraction.

2.1.4 Audio Features

We extract the following low-level features from the audio stream:

MFCC: We first transform the raw audio into a 45 dimensional feature stream using the following steps. Features were extracted from overlapping frames of audio data, each 29 ms long, at a rate of 100 frames per second. Each frame was windowed with a Hamming window and a power spectrum was computed for the frequency band 80-6000 Hz. From this, 14 Mel-warped cepstral coefficients were computed. Each segment of speech is normalized by the mean cepstrum and peak energy non-causally, removing any long term bias due to the channel. In addition, the feature vectors were scaled and translated such that for each video, the data has zero mean and unit variance. These base cepstral features with their first and second derivatives, together with the energy and its first and second derivatives, compose the 45-dimensional feature vector.

FDLP: The Frequency Domain Linear Prediction (FDLP) model, in contrast to the short-term analysis by MFCC, is based on linear prediction on different frequency bands, and describes the perceptually dominant peaks and removes the finer-scale details. Followed by short-term temporal energy integration, conversion to cepstral features, and concatenation within temporal context windows, the resultant FDLP feature has 588 dimensions. FDLP has been shown to perform well when channel distortion varies, and therefore is applicable to MED'11 in which the audio channels are highly heterogeneous.

Audio Transients: Another audio feature complementary to the MFCC feature is the transient event feature. The transient event feature differs from MFCC with respect to the fact that it does not have uniformly-spaced frames and instead focuses on audio transients. Particularly, spectrograms are extracted with window lengths between 2 and 80ms – high-magnitude in any frequency bins in any of these proposes the candidate transient event times. At each event time, a spectrogram with window length of 25ms is extracted from a neighborhood of 250ms, which is reshaped into a vector representation. Each video clip has a set of these vectors, to be modeled using bag-of-word approaches.

2.1.5 Unsupervised Feature Learning

We also investigated unsupervised feature learning directly from the data. The features described so far are hand-coded. Instead, we used an extension of the Independent Subspace Analysis (ISA) algorithm to learn invariant spatio-temporal features from unlabeled video data.

We tried this method on the UCF11 dataset. In the first run, the block size of 10×10×16 and 16×16×20 were used for the first and second ISA levels, respectively. The accuracy for 11 actions is 60% with 25-fold cross validation. In the second run, the block sizes for two levels were tuned to 8×8×10 and 16×16×15. Finally, we combined the histograms of these two runs and obtained an accuracy of around 72%. These features also showed promising performance in the MED'11 task and we will further explore them further in future work.

2.2 Coding and Pooling Strategies

For each of the low-level features described, we tested several strategies for projecting the extracted descriptors to a codebook, and then for aggregating the projections to get a single feature vector for the video. Following the notations in [Boureau et al. 2010], let x_i be the set of low-level descriptors extracted from the video, where i=1…N. Let the video be partitioned into M regions of interest (e.g., the 21=16+4+1 cells in the three level spatial pyramid used in [Lazebnik et al. 2006]). Let f and g denote some coding and pooling operators. Then, the vector z representing the whole image is obtained by coding and pooling over each region and then concatenating:

$$\boldsymbol{\alpha}_i = f(x_i), \qquad i = 1, \dots, N \tag{1}$$

$$\boldsymbol{h}_m = g\left(\{\boldsymbol{\alpha}_i\}_{i \in N_m}\right), \qquad m = 1, \dots, M \tag{2}$$

$$z^T = [\boldsymbol{h}_1^T \dots \boldsymbol{h}_M^T] \tag{3}$$

In usual bag-of-words, we use *hard quantization* and *average pooling*, i.e. f minimizes distance to a codebook learnt by an unsupervised algorithm like k-means, and g averages over the pooling region. This can be formally represented as:

$$\boldsymbol{\alpha}_i \in \{0,1\}^K, \boldsymbol{\alpha}_{i,j} = 1 \;\; \text{iff } j = (_k \leq K \;\; \square^{\mathbf{argmin}} \|x_i - \mathbf{d}_k\|_2^2 \tag{4}$$

$$\boldsymbol{h}_m = \frac{1}{|N_m|} \sum_{i \in N_m} \square\, \alpha_i \tag{5}$$

In recent work [Jiang et al. 2010], *soft quantization*, where the descriptors are assigned with soft weights to the codebooks, has shown significantly better performance. Formally:

$$\boldsymbol{\alpha}_{i,j} = \frac{\exp\left(-\beta\|x_i - \mathbf{d}_j\|_2^2\right)}{\sum_{k=1}^{K} \exp\left(-\beta\|x_i - \mathbf{d}_k\|_2^2\right)} \tag{6}$$

In our experiments, we found the combination of soft quantization with average pooling to produce the best results for a range of features. Further, we saw significant gains by using spatial pooling, which is consistent with several earlier results [Lazebnik et al. 2006][van de Sande et al. 2010][Jiang et al. 2010]. In addition to these strategies, we also tested the utility of recent *sparse coding* techniques for D-SIFT features. Here, the feature descriptors are projected to a dictionary $\emptyset$ by optimizing:

$$\min_{\downarrow} \alpha \;\; \|\mathbf{x} - \emptyset_{\downarrow} \alpha \;\; \|_{\downarrow} 2, \qquad \text{s.t. } \| \alpha \|_{\downarrow} 0 \leq k \tag{7}$$

We considered a novel *α-histogram* pooling where the *α*-values were aggregated using a histogram instead of average/max pooling. Without spatial pooling, this combination improved performance over soft quantization+average pooling for D-SIFT features. However, with spatial pooling, soft quantization+average pooling had better performance. We plan to combine sparse coding with spatial pooling in future and carefully study the performance impact.

3 Joint Audio-Visual Bi-Modal Words

Joint audio-visual patterns often exist in videos and provide strong multi-modal cues for detecting events. In order to discover the visual-audio correlation, we apply a bipartite graph to model relation across the quantized words extracted from the visual and the audio modalities. We then apply graph partitioning to construct bi-modal words that reveal the joint patterns across modalities. In recent literature, bipartite graph has been used successfully in various applications [Liu et al. 2010][Pan et al. 2010]. To the best of our knowledge, this is the first work to apply bipartite graph to model the correlation between audio and visual codebooks. It offers several distinct advantages – the dimensionality of the features can be greatly reduced (from about 14,000 to 8,000) and the bi-modal words provide strong cues and discriminative power for detecting the MED'11 events. The process for bi-modal word construction is illustrated in Figure 1 and described as follows.

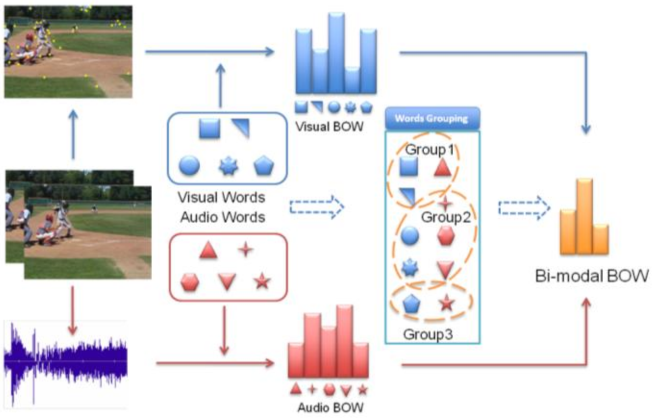

Figure 1: Bi-Modal word construction pipeline.

First, we apply BoW to build audio words and visual words through the standard *k*-means clustering method separately. Then, a bipartite graph is constructed to capture certain relations (e.g. co-occurrence, causality, etc.) between the audio words and visual words. After that, the spectral clustering technique is used for graph partitioning. Finally, the original individual word in each modality (audio or visual) is re-quantized into the discovered bi-modal word clusters which are then used as the final feature.

To consider different scales of the temporal information, we apply two versions of the bi-modal, video-level and clip-level. For the video-level representation, the link relation in the bipartite graph is computed by measuring the co-occurrence statistics of audio and visual words in the entire video. For the clip-level, each video is first segmented into short clips and the link relation in the bipartite graph is computed from the co-occurrence statistics of audio and visual words in that clip only. As shown in the experimental section, bi-modal features have strong performance and also produce gains when combined with other features.

4 High-level Visual Features

We included high visual information from 2 sources – object detection and scene concepts. We developed a novel representation for leveraging object detections that included spatial information. The scene concepts were detected at the image level.

4.1 Object Detection

We used detections from a state-of-the-art object detector developed by Pedro Felzenszwalb at the University of Chicago [Felzenszwalb et al. 2010]. We trained and tested detectors for several objects including cars, pedestrian, hat, cake, bicycles, tires etc. A key challenge was – "*How should the output of such high-level concept detectors be integrated into the MED classification framework?*" In order to address this challenge, we developed a novel representation called the *spatial probability map* which captures the spatial distribution of an object's presence in a video. Overall, we found car detections to produce consistent gains for the "*Getting vehicle unstuck*" event, but did not find significant improvement when we used other detectors.

4.2 Scene Concepts

Recent work shows that concept-based mid-level feature provides extra information for event classification. We applied the Classemes models provided in [Torresani et al. 2010] to generate scene concept features. These models were trained over a large scale concept pool (around 3000 concepts) defined in LSCOM. The concept scores generated by classifiers are used as the feature for training the final event model. In our experiments, we found the scene concept features to be competitive with the SIFT features and produced marginal gains when combined with the low-level feature system.

5 Automatic Speech Recognition

Human language content is often present in consumer videos in the form of the spoken content in the audio track. Such content could potentially provide useful information for detecting events of interest. For example, in videos of tutorials about making dishes and documentaries about particular expeditions, the accompanying spoken narrative provides information about the category of the video. Besides, the semantic information from human language is, typically, complementary to the information from low-level visual features.

Our approach for using the spoken language information in the audio track involves the following three modules.

First, within the video clips, the speech segments are identified by a speech activity detection (SAD) system. The SAD system employs two Gaussian mixture models (GMM), for speech and non-speech observations respectively. A small subset of 101 video clips is annotated for speech segments, which are used for training the speech GMM. Besides the non-speech segments in this set, we also use 500 video clips with no speech at all to enrich the non-speech model, in order to handle the heterogeneous audio data in MED'11.

Second, we apply BBN's large-vocabulary automatic speech recognition (ASR) system to the speech data to produce a transcript of the spoken content. This system is adapted from a BBN ASR system trained on 1700-hour broadcast news. In particular, we adapt the lexicon and language model using MED'11 descriptor files, relative web text data, and the small set of 101 video clips with annotated speech transcription. The acoustic models are adapted during ASR decoding for each video clip in an unsupervised fashion.

Finally, to leverage the hypothesized speech transcripts in event detection, we use the distribution of a set of event-discriminating keywords within each video clip. The hypothesized speech transcripts, stop words removed, are normalized and then stemmed by the Porter stemmer. We identify event-discriminating keywords by choosing those that score the highest according to a revised TF-IDF criteria: $(n/t)^{\alpha}\log(d/h)$, where n is the number of times a word appears in a video clips belonging to a particular event category; t is the total number of words in that category; d is the total number of categories considered; h is the number of categories containing the word; α is an exponential weight. For each video clip, the counts of these keywords are normalized to form a histogram of keywords within that clip. In the MED'11 submission, α is chosen to be 0.25 and the top 2251 keywords are selected, forming a histogram of length 2251.

6 Video Text OCR

We also included input from videotext OCR into our overall system. Detecting salient text such as "*making a sandwich*" can be of significant use in the event detection task and also for the event recounting task. However, unlike speech, the occurrence of text in video is quite sparse. Hence, it is difficult to learn sophisticated dictionary and language models from the available training data. Therefore, we created a small 128-word dictionary based on words occurring in the event kit textual descriptions. In order to reduce the false positives from videotext, we also constructed bigrams for each event.

Given a video, we ran our videotext detection/OCR engine on every 5th frame. We eliminated all the special characters detected by the OCR engine to get the final OCR output for event detection. We then matched each word in the output with the dictionary words using the Levenshtein edit distance measure and assigned a score of (1-*NED(w,d)*) for each word (*w*) and dictionary word (*d*) pair, where, *NED(w,d)*, Normalized Edit Distance, is the ratio of the Levenshtein edit distance to the length of the word.

For the specific bigrams chosen for an event, we computed the *bigram score* for words occurring in a frame as:

$$bigram\_score([w_1,w_2],[d_1,d_2]) = NED(w_1,d_1) * NED(w_2,d_2) \tag{8}$$

Finally, for each event we computed the probability of an event by taking the maximum of the bigram scores corresponding to that event over the entire video. These scores were then used during system combination. Through our experiments on our internal dev and test partitions, we observed that inclusion of videotext produced about 4% relative reduction in the P_{MD} without increasing the P_{FA} of the system.

7 Classifier Learning and Feature Fusion

Using the features described so far, we build multiple sub-systems by training kernel based classifiers for each event. During this process, we jointly optimize the classifier parameters and the detection threshold. Given a test video, we obtain classification scores for each of these sub-systems. We then apply a late fusion strategy to combine these scores and obtain a final detection score. During training, we also estimate a detection threshold for the late fusion system. In this section, we will describe each of these steps.

7.1 Kernel Based Early Fusion

We trained different subsystems by combining different features from the same class, such as appearance, color, motion, etc. We first computed χ^2 kernels for each feature and then combined them using a weighted sum. The weights were automatically learnt from training data based on the approach in [Viswanathan et al. 2010]. Further, we used standard parameter estimation techniques to optimize the performance of each sub-system.

7.2 Detection Threshold Estimation

We considered two approaches for threshold estimation – the first optimized for NDC within a specific P_{FA} and P_{MD} boundary, while the second minimized the missed detection rate assuming a maximum tolerable false alarm rate.

7.2.1 Bounded NDC Optimization

In order to estimate the detection threshold, we ran *k*-fold validation trials on an internally created training set which is a subset of the MED’11 training data (DEVT and Event Kits). Next, we used the outputs of the *k*-fold trials (with $k = 3$ for our initial run) to compute a set of P_{FA} and MNDC values by applying detection-score thresholds ranging from 0 to 1, in increments of 0.001. Subsequently, the set of <Threshold, $P_{FA,}$ MNDC> tuples was sorted in increasing order of P_{FA}, and the estimated threshold for each event is set to the value that corresponds to the lowest value of a “moving average” of MNDC scores computed using a fixed-width window of P_{FA} values.

7.2.2 Missed Detection Rate Optimization

In this approach, the threshold is computed based on the percentage of videos retrieved by the system. Since the ALADDIN program goals for Year-1 are 75% P_{MD} at 6% P_{FA}, we estimated the threshold (using training data only) to minimize P_{MD} while staying within the maximum P_{FA} target. We used this threshold estimation approach in our primary MED’11 evaluation system.

7.3 System Combination

Once we train the different sub-systems and estimate their detection thresholds, we considered two possible strategies for combining the different sub-systems, namely, Bayesian model combination (BAYCOM) and weighted average fusion, which are described in the following sub sections.

7.3.1 Bayesian Model Combination (BAYCOM)

In the first system combination approach, we combine different sub-system outputs using a Bayesian decision theoretic approach (BAYCOM). Let *M* be the number of models to combine, and r_i denote the output generated by model *i*. Here, r_i consists of the classification c_i generated by system *i*, along with the associated confidence score s_i, i.e. r_i =(c_i, s_i). Let the event *c* mean “hypothesis *c* is correct” and *C* be the set of unique classes proposed by all systems. Then, the model selects the optimal hypothesis according to:

$$c^* = (_{c} \in C \quad \square^{\mathrm{argmax}} P(c|r_1, \dots, r_M) \qquad (9)$$

In our system, we used class specific conditional probabilities and overcame data sparseness by smoothing the conditional probabilities with class independent probabilities. BAYCOM is effective in optimizing performance at a specific operating point, and produces the smallest number of errors in terms of the number of missed detections and false alarms at the detection threshold. However, it converts the individual system scores to a bimodal distribution and causes the performance to degrade rapidly at other points in the DET curve.

7.3.2 Weighted Average Fusion

In the second approach for combining different sub-systems, we used a variant of weighted average fusion where, in addition to computing a global system level weight, we adaptively weight each system’s output on a video by video basis. The first is a system level weight (w_1), which was calculated from the ANDC scores of each system based on our internal partitions. The second is a video specific weight (w_2), calculated from the optimal threshold for the system found during our threshold analysis, and the confidence score for a given test video.

Given these weights, the output score P for a video j is simply given by:

$$P(j) = \frac{\sum_i w_1(i) w_2(i,j) p_{ij}}{\sum_i w_1(i) w_2(i,j)} \qquad (10)$$

8 Experiments and Results

In this section, we analyze the performance of each of our systems and their associated components. For the MED'11 evaluations, we submitted 4 variants of our system, each corresponding to a different combination of features, fusion, and threshold estimation strategy. For the reader's convenience, we have repeated the summary description of each variant below.

1) BBNVISER-LLFeat: This combines 6 high performing low-level features, including 3 color features, 1 appearance feature, 1 motion feature, and 1 audio feature. These features are combined using the weighted kernel based early fusion strategy (Section 7.1), and the detection threshold was optimized for the NDC score (Section 7.2.1).

2) BBNVISER-Fusion1: For the second submission, we used 7 sub-systems. First we created sub-systems based on ASR, low-level audio, motion, appearance, and color features. We also trained a system based on features extracted by Columbia University (SIFT, STIP, MFCC, scene concepts, and bimodal audio-visual features). In addition to these 6 sub-systems, we included the BBNVISER-LLFeat system. Then for each video, we combined the scores from these 7 systems using a Bayesian model combination (BAYCOM) strategy described in Section 7.3.1. Then, we combine the fused system with output from video text OCR as described in Section 6. The detection threshold was optimized for the NDC score (Section 7.2.1).

3) BBNVISER-Fusion2: For the third submission, we used the same 7 sub-systems as BBNVISER-Fusion1, but combined them using the weighted average fusion strategy described in Section 7.3.2.

4) BBNVISER-Fusion3: For the final submission, we combined 18 different sub-systems. This includes the 7 sub-systems used in submissions (2) and (3). We also included the BBNVISER-Fusion2 system. Further, we trained two additional subsystems by combining BBNVISER-LLFeat with the feature based on car detections (from Section 4.1), and by combining BBNVISER-LLFeat with car detections and ASR. Finally, we included 7 sub-systems based on end-to-end runs done at Columbia University and a sub-system from an end-to-end run done at UCF that was based on deep learning of low-level features (Section 2.1.5). These 18 sub-systems were combined using weighted average fusion (Section 7.3.2) and the threshold was optimized to minimize missed detections with ~6% false alarm rate (Section 7.2.2).

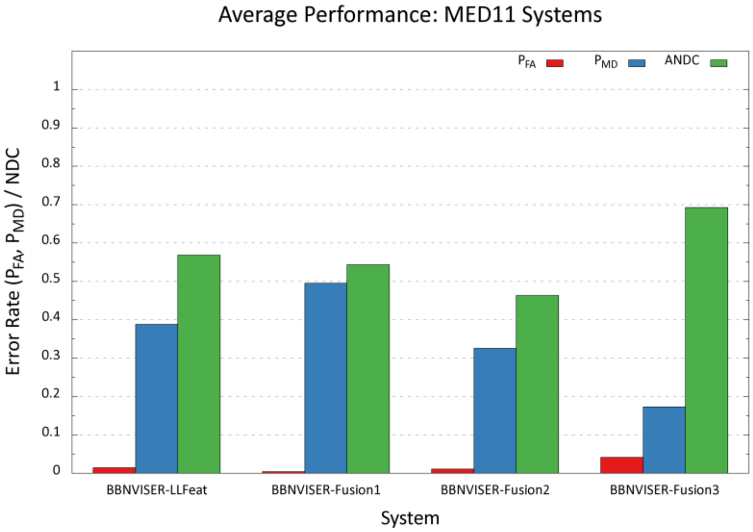

Figure 2: Summary of BBNVISER MED'11 submissions.

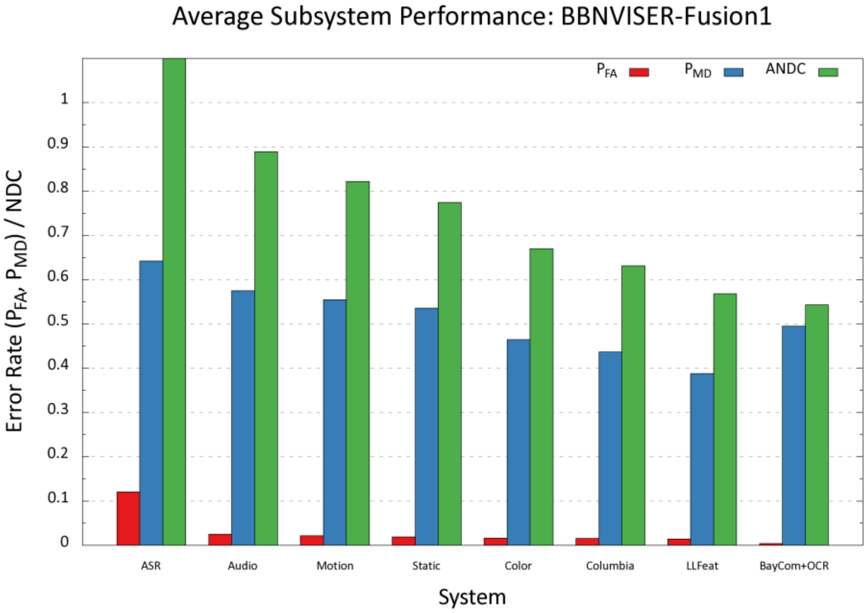

Figure 3: Performance of individual subsystems used in BBNVISER-Fusion1 system.

Figure 2 summarizes the P_{MD}, P_{FA}, and ANDC scores of our 4 MED'11 submissions. Overall, BBNVISER-Fusion2 has the best ANDC score and also lower P_{MD}, P_{FA} rates compared to the baseline BBNVISER-LLFeat system. This demonstrates the importance of feature fusion as well as high-level information from ASR, object/scene concept/action concept detections, and video text OCR.

The BBNVISER-Fusion1 system using BAYCOM has better ANDC score than BBNVISER-LLFeat, with lower P_{FA} and P_{MD}. This system has the smallest total number of errors in terms of missed detections and false alarms at the detection threshold. In effect, it optimizes the fusion system at the detection threshold, but has sub optimal performance at other points in the DET curve. Figure 3 illustrates the performance of different sub-systems used in BBNVISER-Fusion1 submission.

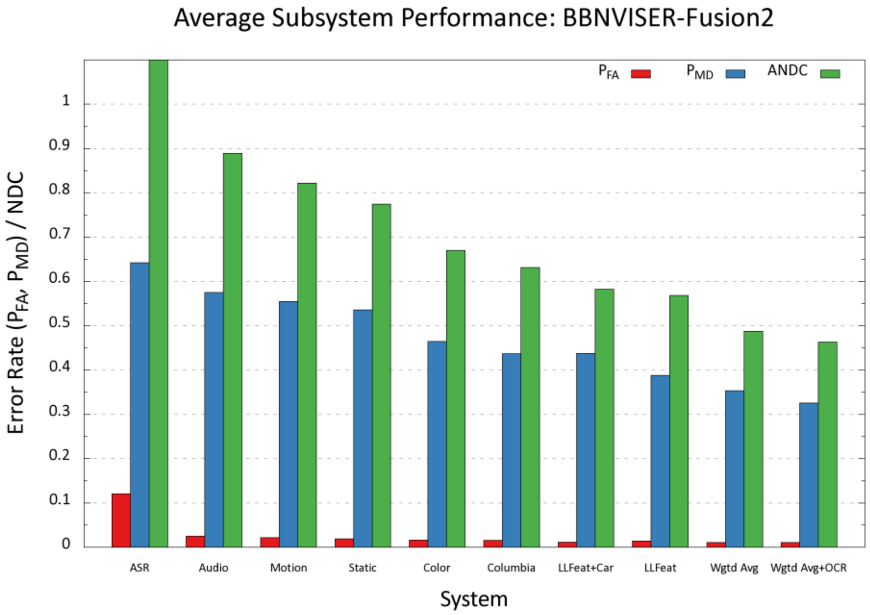

Figure 4: Performance of individual subsystems used in BBNVISER-Fusion2 system.

The BBNVISER-Fusion2 system has the best ANDC score across all 60 submissions for the MED'11 task. Further, it has P_{FA}<6% and P_{MD}<75% for all 10 events, P_{FA}<4% and P_{MD}<50% for 9/10 events, P_{FA}<2.8% and P_{MD}<35% for 5/10 events, and P_{FA}<2% and P_{MD}<25% for 2/10 events. Figure 4 summarizes the performance of different sub-systems used in the fusion. Combining the 8 sub-systems used, improves the ANDC score by ~0.08 over the best individual system, and the inclusion of videotext OCR improves it further by 0.04, demonstrating the utility of feature fusion as well as video text OCR. The weighted average fusion strategy not only improves performance at the detection threshold but also over the entire DET curve. It also has better performance compared to BAYCOM at the detection threshold. This demonstrates the utility of using a weighted average fusion strategy with video specific weights.

The BBNVISER-Fusion3 system (Figure 5) has similar performance to BBNVISER-Fusion2, but the threshold is optimized to minimize missed detection rate with a false alarm rate of near 6%. This system has the lowest MNDC score across all 60 submissions for MED'11, and the lowest missed detection rates for all 10 events among the systems submitted by BBN.

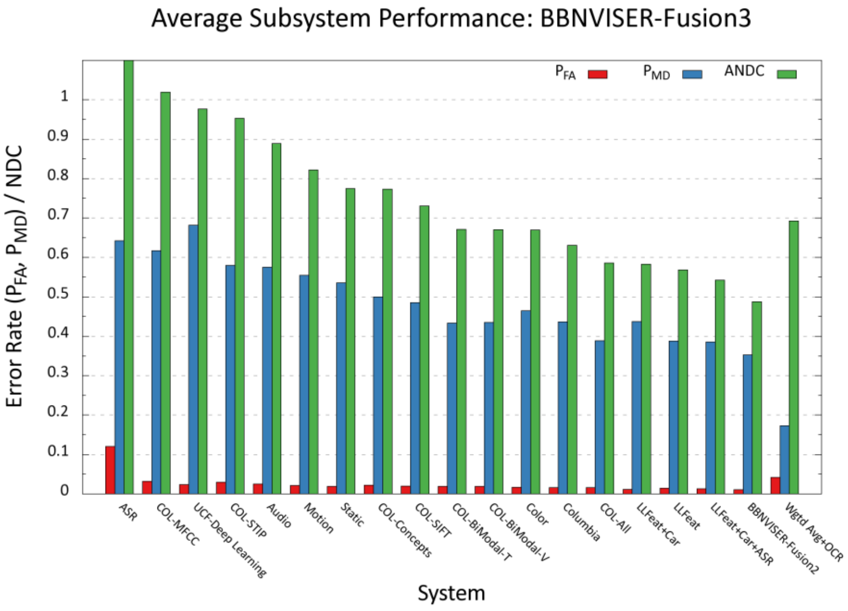

Figure 5: Performance of individual subsystems used in BBNVISER-Fusion3 system.

Figures 6 and 7 summarize the relative performance of the 4 BBN submissions compared to all 60 submissions for MED'11, in terms of ANDC and MNDC, respectively. BBNVISER-Fusion2 has the best ANDC score, while BBNVISER-Fusion3 and BBNVISER-Fusion2 have the top 2 MNDC scores. Further, BBNVISER-LLFeat has strong performance on both metrics. This indicates that low-level features have strong performance stand alone. However, high-level information from ASR, video text, and vision processing produce large gains over the low-level features. We plan to build on these results by developing stronger representations and systems for capturing such high-level information from videos.

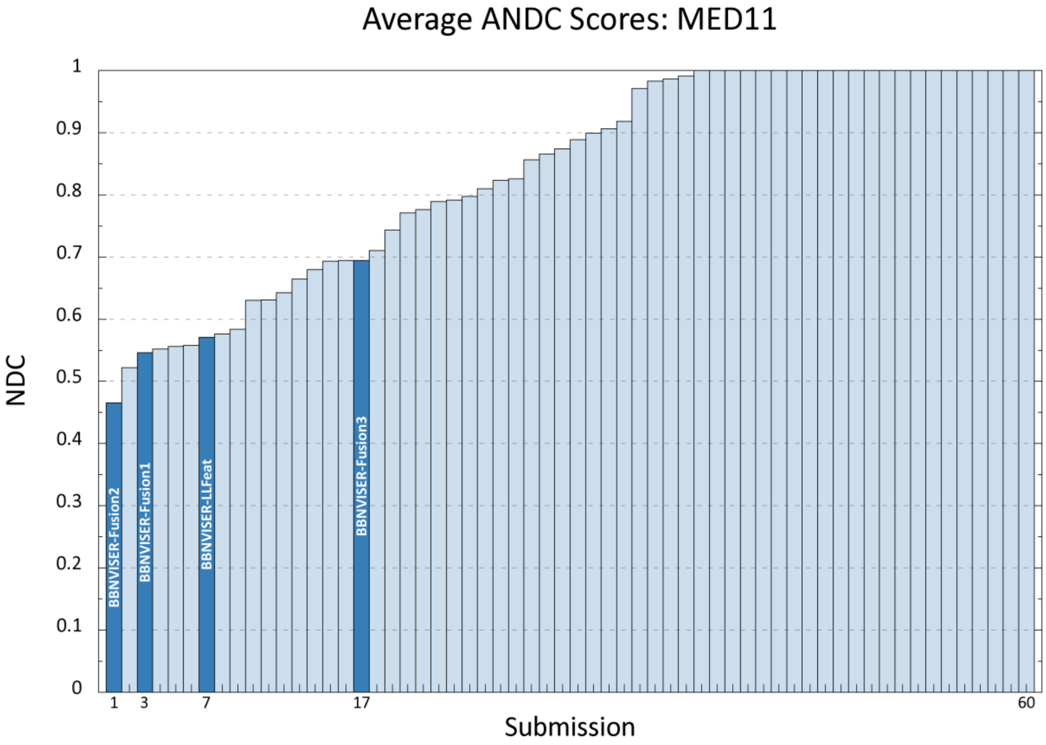

Figure 6: Performance of our MED runs and all 60 official submissions. The vertical axis shows the performance measured by average actual normalized detection cost (average ANDC) across the 10 MED'11 events.

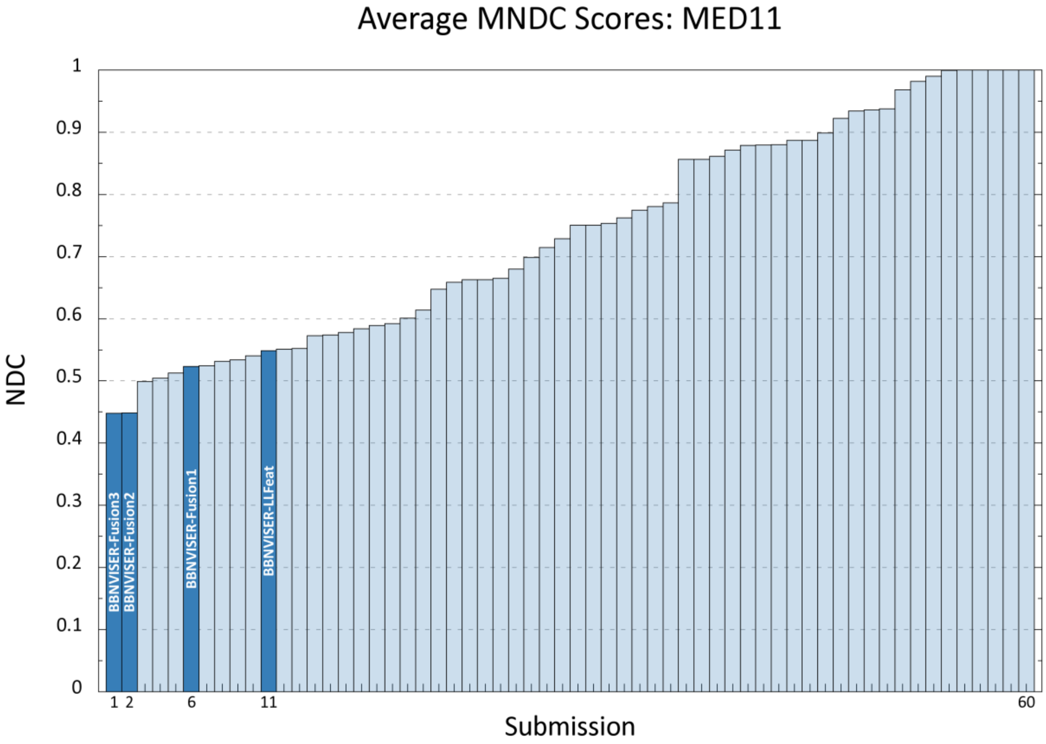

Figure 7: Performance of our MED runs and all 60 official submissions. The vertical axis shows the performance measured by average minimum normalized detection cost (average MNDC) across the 10 MED'11 events.

9 Acknowledgments

Supported by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior National Business Center contract number D11PC20071. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright annotation thereon.

Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DoI/NBC, or the U.S. Government.

References

[Csurka et al. 2004] G. Csurka, C. Dance, L.X. Fan, J. Willamowski, and C. Bray, “Visual Categorization with Bags of Keypoints,” in *Proc. of ECCV International Workshop on Statistical Learning in Computer Vision*, 2004.

[Lazebnik et al. 2006] S. Lazebnik, C. Schmid, and J. Ponce, “Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories,” in *Proc. CVPR*, 2006.

[Laptev et al. 2008] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld, “Learning Realistic Human Actions from Movies,” in *Proc. CVPR*, 2008.

[Jiang et al. 2010] 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.

[Lowe 2004] D. Lowe. Distinctive image features from scale-invariant keypoints. International Journal on Computer Vision, 60:91–110, 2004.

[Mikolajczyk et al. 2004] K. Mikoljczyk and C. Schmid. Scale and affine invariant interest point detectors. International Journal of Computer Vision, 60:63–86, 2004.

[Laptev 2005] I. Laptev. On space-time interest points. International Journal of Computer Vision, 64:107–123, 2005.

[Liu 2011] Jingen Liu, Mubarak Shah, Benjamin Kuipers, and Silvio Savarese, Cross-View Action Recognition via View Knowledge Transfer, IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), Colorado Springs, 2011

[Pan et al. 2010] S. J. Pan, X. Ni, J.-T. Sun, Q. Yang, and Z. Chen. Cross-domain sentiment classification via spectral feature alignment. international conference on World Wide Web, New York, NY, USA, 2010.

[Torresani et al. 2010] Lorenzo Torresani, Martin Szummer, Andrew Fitzgibbon. Efficient Object Category Recognition Using Classemes. European Conference on Computer Vision, 2010

[van de Sande et al. 2010] 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, volume 32 (9), pages 1582-1596, 2010.

[Boureau et al. 2010] Y. Boureau, F. Bach, Y. Le Cun, and J. Ponce, Learning mid-level features for recognition. In CVPR, pages 2559-2566, 2010.

[Bay et al. 2008] H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool. Surf: Speeded up robust features. CVIU, 110(3):346-359, 2008.

[Chandrasekhar et al. 2011] V. Chandrasekhar, G. Takacs, D. Chen, S. Tsai, Y. Reznik, R. Grzeszczuk, and B. Girod, "Compressed histogram of gradients: a low bitrate descriptor", International Journal on Computer Vision, Vol. 94, No. 5, May 2011.

[Felzenszwalb et al. 2010] P. Felzenszwalb, R. Girshick, D. McAllester, D. Ramanan. Object Detection with Discriminatively Trained Part Based Models. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 32, No. 9, September 2010

[Viswanathan et al. 2010] S. V. N. Vishwanathan, Zhaonan Sun, Nawanol Theera-Ampornpunt, and Manik Varma. Multiple Kernel Learning and the SMO Algorithm. In Advances in Neural Information Processing Systems 23, pp. 2361–2369, 2010.

[Natarajan et al. 2011] P. Natarajan, S. Tsakalidis, V. Manohar, R. Prasad, and P. Natarajan, "Unsupervised Audio Analysis for Categorizing Heterogeneous Consumer Domain Videos," in Interspeech, Florence, Aug. 2011.

[Vitaladevuni et al. 2011] S. Vitaladevuni, P. Natarajan, R. Prasad, and P. Natarajan, "Efficient Orthogonal Matching Pursuit using sparse random projections for scene and video classification," To appear ICCV, Barcelona, 2011.

[Manohar et al. 2011] V. Manohar, S. Tsakalidis, P. Natarajan, R. Prasad, and P. Natarajan, "Audio-Visual Fusion Using Bayesian Model Combination for Web Video Retrieval," To appear ACM Multimedia, Scottsdale, AZ, Nov. 2011.