# IBM Research and Columbia University TRECVID-2013 Multimedia Event Detection (MED), Multimedia Event Recounting (MER), Surveillance Event Detection (SED), and Semantic Indexing (SIN) Systems

Lisa Brown<sup>†</sup> Liangliang Cao<sup>†</sup> Shih-Fu Chang<sup>†</sup> Yu Cheng<sup>†</sup> Alok Choudhary<sup>†</sup> Noel Codella<sup>†</sup> Courtenay Cotton<sup>\*</sup> Dan Ellis,<sup>\*</sup> Quanfu Fan<sup>†</sup> Rogerio Feris<sup>†</sup> Leiguang Gong<sup>†</sup> Matthew Hill<sup>†</sup> Gang Hua<sup>†</sup> John Kender<sup>‡</sup> Michele Merler<sup>†</sup> Yadong Mu<sup>\*</sup> Sharath Pankanti<sup>†</sup> John R. Smith<sup>†</sup> Felix X. Yu<sup>\*</sup>

# Abstract

For this year's TRECVID Multimedia Event Detection task [11], our team studied a semantic approach to video retrieval. We constructed a faceted taxonomy of 1313 visual concepts (including attributes and dynamic action concepts) and 85 audio concepts. Event search was performed via keyword search with a human user in-the-loop. Our submitted runs included Pre-Specified and Ad-Hoc event collections. For each collection, we submitted 3 exemplar conditions: 0, 10, and 100 exemplars. For each exemplar condition, we also submitted 3 types of semantic modality retrieval results: visual only, audio only, and combined.

The current IBM-Columbia MER system exploits nine observations about human cognition, language, and visual perception in order to produce an effective video recounting of an event. It designed and tuned algorithms that both locate a minimal persuasive video segment, and script a minimal verbal collection of concepts, in order to convince an analyst that the MED decision was correct. With little loss of descriptive clarity. the system achieved the highest speed-up ratio amongst the ten teams competing in the NIST MER evaluation.

For SED, we seek to explore temporal dependencies between events for enhancing both evaluation tasks, i.e automatic event detection (retrospective) and interactive event detection with human in the loop (interactive). Our retrospective system is based on a joint-segmentation-detection framework integrated with temporal event modeling while the interactive system performs risk analysis to guide the end user for effective verification. We achieve better results on the retrospective and interactive tasks than last year.

For SIN, we submitted 4 full concept detection runs, and 2 concept pair runs. In the first 3 concept detection runs, we changed our data sampling strategy between using balanced bags via majority undersampling for ensemble fusion learning, balanced bags via minority oversampling, and unbalanced bags. For the 4th run we used a rank normalized fusion of the first 3 runs. Concept pair runs consisted of the sum of individual concept classifiers with and without sigmoid normalization of the dataset.

# 1 Multimedia Event Detection (MED)

#### 1.1 Overview

For this year's MED task, we sought to develop a semantic event detection methodology that unifies the approach across all 3 search conditions: 0EX, 10EX, and 100EX. Our system, simply summarized, allows semantic keyword search over the video corpus. Semantic concepts are detected for both visual and audio modalities. Upon supplied training examples in the form of videos (for the 10EX and 100EX cases), our system analyzes the videos to recommend a set of semantic concepts for the keyword search event retrieval. This is used a starting point for the human user to perform a given event search. When no event video exemplars are available (0EX condition), the user employs our interface to select the most intuitively relevant concepts from our taxonomy to perform semantic keyword based search. In this manner, the approach for all search retrieval conditions has been unified under one elegant and user-friendly design.

In the subsequent subsections, each stage of our system is described: VCD generation, ECD generation, and event search. Finally, we summarize our submitted runs.

<sup>\*</sup>Columbia University, Dept. of Electrical Engineering

<sup>&</sup>lt;sup>†</sup>IBM T. J. Watson Research Center

<sup>&</sup>lt;sup>‡</sup>Columbia University, Dept. of Computer Science

<sup>&</sup>lt;sup>§</sup>Supported by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior National Business Center contract number D11PC20070. 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.

#### 1.2 Video Content Description (VCD)

In this section, we describe the methods employed to model and extract the semantic content from videos. The methods are broken into different components described in each of the subsequent subsections. The first set of components concern video processing and both the variety of visual and audio low-level features used to model the semantics. The second set of components address modeling the semantics using our unique faceted taxonomy approach in conjunction with the extracted low-level features.

#### 1.2.1 Low-level Feature Extraction

#### 1.2.2 Video processing

We decode the video clip, and uniformly save one frame every two seconds. These frames are then used to extract static visual low-level descriptors. We extract over 100 types of static image features from each of the sampled frames. These features capture a wide range of image information including color, texture, edge, local appearances and scene characteristics. We chose 0.5 fps as a sampling rate based on the data set size in order to yield a number of frames that we could process in a reasonable time.

Our system uses a subset of these low-level features to determine the semantic content of video frames, from which further event modeling is performed. Semantic content is extracted at a slower rate of 0.25 fps due to the added complexity of evaluating the models.

#### 1.2.3 Low-Level Descriptors for Semantic Analysis

A combination of local and global descriptors are extracted for the analysis of the visual semantic content of video frames. Features extracted include some standard ones, such as LBP, GIST, Color Histogram, etc, as well as a new proprietary multi-scale LBP and Fourier Polar Pyramid features.

Each descriptor is evaluated at various spatial granularities that include global, center, cross, grid, horizontal parts, horizontal center, vertical parts, vertical center, and pyramids – each of which is a fixed division of the image frame into square blocks (numbering from 1 up to 25), and then concatenating the descriptor vectors from each block. Such spatial divisions has been repeatedly shown robust performance in image/video retrieval benchmarks such as TRECVID [14].

Local Binary Patterns (LBP) [16] are extracted across two image granularities: global, and a pyramid consisting of 1x1 followed by 2x2. The LBP histogram is extracted from the greyscale and color versions of the image as a histogram of 8bits local binary patterns, each of which is generated by comparing the grayscale value of a pixel with those of its 8 neighbors in circular order, and setting the corresponding bit to 0 or 1 accordingly. A pattern is called uniform if it contains at most two bitwise transitions from 0 to 1. The final histogram for each region in our granularity contains 59 bins, 58 for uniform patterns and 1 for all the non-uniform ones. In addition, we've implemented a new multi-scale variant of the LBP descriptor this year, which has shown improvements in performance. Currently, the multiscale LBP features are our best performing low-level features.

In addition to the LBP descriptors, we extracted 13 different visual descriptors at 8 granularities and spatial divisions, including Color Histogram, Color Correlogram, Color Moments, Wavelet Texture, Edge Histogram, and GIST. SVMs are trained on each feature and subsequently linearly combined in an ensemble classifier. Details on features and ensemble classifier training can be found in our prior report [1, 2].

We have also developed a proprietary feature referred to as a Fourier Polar Pyramid. It incorporates ideas from both spatial pyramids and from the Curvelets feature transform. The basic idea is to construct a spatial pyramid in Fourier space, under the polar coordinate system, across all 3 color channels red, green, blue, in addition to a grayscale color channel. Pyramid levels in the radial dimension consist of 1, 2, 4, and 8 partitions. For each of these paritions, we also construct a pyramid in the angular dimension, of partitions 1, 2, 4, 8, 16, and 32 segments (see Figure 1). Due to the property of image symmetry in Fourier space, only the top half of the polar Fourier circle is sampled for the feature vector. In addition, we have added a prefiltering step to the original image that multiplies a circular mask to improve the rotational invariance of the discrete Fourier transform. In total, the dimensionality of our new feature vector is 3900 for the global granularity, and 19,500 for the layout granularity. For efficiency purposes, on the MED task, we reduced the complexity of the Fourier Polar Pyramid to radial partitions of 1, 2, 4, and 8. Angular segments were reduced to 1, 2, 4, 8, and 16. This resulted in a feature of 868 dimensions.

#### 1.2.4 Spatio-Temporal Features

In order to train dynamic action concepts, we employed state of the art spatio-temporal descriptors based on dense-trajectory features. In particular, we adopted the Motion Boundary Histogram (MBH) [18] descriptor within the bag of visual words framework with a 5000-dimensional codebook.

#### 1.2.5 Audio Features

We calculated two types of low-level audio features over nonoverlapping 2 second windows for each video. The first features are based on conventional MFCCs, and constitute the mean and full covariance matrix of the MFCCs. We use 20 MFCC dimen-



Figure 1: Depiction of the overlapping polar pyramid partitions sampled in Fourier-Mellin space (light blue boxes) used to compute the Fourier Polar Pyramid.

sions in our basic feature (for a finer description of the spectrum than is typically used in speech recognition); we also calculate the deltas and double-deltas. MFCCs are calculated over 32 ms windows every 16 ms, and the delta and double-delta features are calculated over blocks of 9 adjacent feature frames (i.e., around 300 ms). The full representation of each window then consists of 60 mean values for each dimension of the direct, delta, and double-delta features, plus the 1830 ( $61 \times 60/2$ ) unique elements of the covariance matrix calculated over the 125 frames contained within each 2 sec block. In practice, we used only the 60 means plus the first 3 leading diagonals of the covariance matrix for 60+60+59+58 = 237 feature dimensions. Each dimension was mean and variance normalized across the training set before creating a Euclidean distance matrix between 2 sec clips to be used as the basis of SVM training.

The second feature type was based on the Auditory Image Model of [9], which captures the fine temporal structure of the audio signal across a set of frequency bands, chops this "auditory image" into a number of different subregions, uses vector quantization to capture the information in each subregion, then performs classification on the concatenated VQ codeword histograms. We replaced the detailed (and rather slow) auditory front-end with a simplified approximation of a linear band-pass filterbank followed by running autocorrelation, and performed VQ on the resulting a PCA reduction of the resulting "correlogram" image in four separate frequency regions (each of 6 bands). With 1000 entries in each codebook, each 2 sec window was represented by a normalized 4000 dimensional histogram. In our experiments, this simplified model performed essentially the same as the full Lyon model, and a little worse than the MFCC features. However, combining MFCC-based and auditory-model-based features gave a substantial improvement of around 15% relative, indicating their complementary information. We used Chi-squared distance to turn this into a



Figure 2: The Faceted Taxonomy approach offers many benefits to other concept labeling methodologies.

distance matrix for use with the SVM.

Both kinds of features can be automatically and exactly aggregated to larger time spans, for instance to calculate the overall features for an entire video. This is implemented within our feature file retrieval routine.

#### 1.2.6 Semantic Taxonomy

For the MED13 event detection task, our team utilized a taxonomy of visual concepts/categories based on the IBM Multimedia Analysis and Retrieval System (IMARS) taxonomy [5]. The IMARS taxonomy is a set of federated multiple facets of concept trees using four conceptual constructs: entity (node), facet (node), is-a (link) and facet-of (link). Adopting the facet node type and "facet-of" link type allows greater flexibility in modeling mutually non-exclusive concepts, which represent different view perspective of a same entity (e.g. people - number of people, age of people). Sibling concepts (nodes) with an entity parent node in the taxonomy tree are mutually exclusive. By inferencing the structure and semantic relationships, the taxonomy system can perform efficient labeling of training images by associating images with the each entity node in the hierarchy, and allocates negative training examples accordingly with the recognition of exclusiveness of entity nodes and non-exclusiveness of facet nodes (Fig. 2).

This year, we have expanded our semantic taxonomy to 8 top level facets, with 1447 total concepts. These cover Image Type (graph, natural photo, color, etc), Setting (Indoor, outdoor, art studio, etc), People (Human, Adult, Child, Police Officer, etc.), Objects (Vehicle, Ball, etc), Audio Concepts (Dog barking, speech, etc), Animals, Human Activities, and Human Actions (dynamic). The framework utilized to expand the taxon-



Figure 3: Semantic Expansion Cycle. There are 4 stages: 1) Develop the Faceted Taxonomy Representation, 2) Expand concept coverage, 3) Train classifiers, 4) Assess classifier utility.

omy is depicted in Fig. 3. Starting with our representation, we mine concepts to expand our coverage, train classifiers from those new concepts, and assess the utility of the resultant models through a combination of detectibility, observability, and information gain.

#### 1.2.7 Visual Semantic Modeling

#### **Modeling Pipeline**

Visual semantic modeling is carried out by the IBM Multimedia Analytics and Retrieval System (IMARS). IMARS is a machine learning system designed for the extraction of semantic content from images and videos. The system has been in development for a period of over ten years, and is unique in its ability to evaluate a multitude of modeling strategies to determine the best approach for each semantic concept. In addition, the structure of the system gives it the ability to arbitrarily scale to large learning problems.

Instead of training very large models by concatenating all features in early fusion, and training models from all available data, our system trains smaller Unit Models [19]. For each concept, Unit Models are trained on a single feature, a single image granularity (such as whole image, which results in a descriptor matching the dimensionality of the feature, or a 5x5 grid resulting in a descriptor 25 times the dimensionality of the original feature), and a random subsample of data, referred to as a bag (Fig. 4). For each Unit Model, the system tries a variety of modeling strategies and kernel parameters (29 or more), selecting the most effective approach via n-fold cross-validation. The most discriminative Unit Models for each concept are selected to

Figure 4: IBM Multimedia Analytics and Retrieval System (IMARS) learning approach. Training data is partitioned into Learning and Validation sets, which are further divided by features and data samples, referred to as "bags." Unit Models are trained for each bag, and an ensemble fusion approach using forward model selection determines the best combination of unit models to discriminate for the given concept.

be fused into an ensemble classifier based on their performance on a held-out validation set.

Building models in this manner yields several desirable properties:

- The first is that data imbalance is markedly reduced. When a Unit Model is sub-sampled, a maximum data imbalance threshold is enforced. The whole of the majority class is covered by the generation of many Unit Models, each with a different sampling of examples.
- 2. The second is that the learning problem is much more efficient when training many smaller models, instead of one large model, since the computational complexity of training a model is polynomial in nature

$$O\left(k \cdot \left(\frac{n}{k}\right)^c\right) << O\left(n^c\right) \tag{1}$$

, especially for large k and large c.

- 3. The third is that since each Unit Model is an independent training task, we can easily parallelize to arbitrary scale.
- 4. The fourth is that since each Unit Model is trained over an individual feature, the process of forward model selection



Figure 5: Example top scoring semantic retrieval results on a random sampling of 80,000 frames from the MED training data. Note how the semantic classifiers can extract some subtle differences between concepts, such as that a steering wheel shown in the red box is a "Car Part" yet not a "Car Tire" even though it has a round shape.



Figure 6: Example top scoring semantic retrieval results on a random sampling of 80,000 frames from the MED training data.

is also implicitly performing feature selection, as it determines the optimal combination of models to combine for each concept.

Some example retrieval results using the trained semantic models are shown in Figures 5 and 6. Note how our classifiers tend to be able to differentiate some subtle characteristics, such as a steering wheel being a car part, and not a care tire, even though both objects are round in shape.

#### **Probablistic Score Normalization**

This year we researched and implemented a method for probabilistic SVM score normalization. The purpose is to roughly map an SVM score to the probability that the score actually yields an instance of the positive class (Fig. 7). This type of score normalization factors in the performance of an SVM classifier. For example, if Dog and Cat both score +0.75, but the AP of Dog is 0.8 and the AP of Cat is 0.6 on a held-out test set, these two scores can mean very different things.

In preliminary experiments, we see improvements both to



Figure 7: Probability normalization process. A validation set is balanced, and the probability of a positive instance is computed for each half. The probabilities are then fit to a sigmoid with the average score in each window. The resultant sigmoid function serves as a mapping from SVM score to probability.

multiclass accuracy and boolean expression event retrieval performance:

We used a 3 concept query for E029 (Winning Race Without a Vehicle) on the 100EX Event Kits dataset, and compared performance of this query using several normalization methods:

- 1. Linear Norm (Original Normalization Method): Sport Facility + Track Scene + Track Racing = AP 0.398693
- 2. Probability: Sport Facility + Track Scene + Track Racing = AP 0.425122

In addition, we performed a multiclass experiment on mutually exclusive sports concepts within the IMARS taxonomy. A held out test dataset of 3020 images was utilized. The multiclass decision was made using a MAX operator over classifier outputs, after normalization. Performance is shown as multiclass accuracy:

- 1. Linear Norm: 0.68245033 (2061 / 3020)
- 2. Probability: 0.70794702 (2138 / 3020)



**Over all model:**  $K(x_i, x_j) = \sum \alpha_m K_m(x_i, x_j) = \sum \alpha_m \phi(x_i) \phi(x_j)$ 





Figure 8: Linear kernel approximation method.

#### **Non-Linear Kernel Approximation**

This year we implemented a method that approximates the behavior of non-linear SVM kernels using linear kernels with additional dimensions. In such a manner, we have been able to maintain the performance of non-linear kernels with incredible performance improvements.

The idea of our kernel approximation based is very simple: first map each feature into a higher dimension space using explicit kernel mapping [17], and train a linear model with all the concatenated features using LibLinear [4]. Figure 8 illustrates this method.

#### 1.2.8 Visual Attribute Modeling with Label Proportions

Attribute-based representation has shown great promises for visual recognition due to its intuitive interpretation and crosscategory generalization property. For MED, attributes can be seen as adjectives describing properties of objects, scenes, human, actions etc. Such adjectives are very useful for describing unseen categories. However, classic attribute learning requires extensive manual labeling of the images. In this work, we propose to model attributes based category-attribute proportions.

Given a category-attribute proportion matrix  $\mathbf{A}$ , in which  $A_{i,j}$  characterizes the percentage of images in category *i* contain-

Figure 9: Examples of discovered attributes and their top ranked images from the IMARS taxonomy: (a) dry, (b) cold, (c) reading, (d) bathing.

ing attribute j, we use the recently proposed proportion-SVM [20] to learn the attributes. In such case, the categories are the IMARS concepts.

Specifically, for MED13, we consider learning the scene attributes. We use a vocabulary of 102 attributes defined by [12]. Some examples are:

glossy matte sterile moist/ damp dry dirty rusty warm cold natural man-made open area semi-enclosed area enclosed area far-away horizon no horizon rugged scene mostly vertical components mostly horizontal components symmetrical cluttered space scary soothing stressful.

Then, a  $717 \times 102$  dimensional category-attribute proportion matrix is computed from the SUN attribute dataset [12]. We then apply the information for learning IMARS scene attributes. Our assumption is that the proportions are approximately the same for IMARS and SUN datasets. We then apply the the method in [20]. Figure 9 presents some results of attributes classes in the form of top ranked images from the IMARS taxonomy.

Table 1: The 45 semantic terms extracted from the brief annotations of the BBC Sound Effects library.

stairs	ambience	speech
pavement	babies	cars
walking	sports	car
women	steam	boat
running	cat	transport
warfare	vocals	urban
wooden	siren	train
footsteps	animals	crowd
country	rhythms	electric
wood	animal	traffic
men	birds	street
horses	rural	children
emergency	electronic	household
construction	horse	crowds
futuristic	aircraft	voices

#### 1.2.9 Audio Semantic Modeling

We trained a total of 100 semantic audio models on our lowlevel audio features. The particular models were defined primarily by the availability of suitable training data. 55 of the models were the same as 2011, and were based on earlier cliplevel labeling efforts based on YouTube data selected to consist of unedited consumer videos [8, 6], and on the MED2010 data segmented in to 10 sec chunks. The remaining 45 models were based on the 60 CD BBC Sound Effects Library, which consists of 2238 sound files covering a wide range of conditions, each lasting anywhere from a second to several minutes. Each track in the BBC collection comes with a one-line description including several keywords describing the sound; we generated a list of candidate semantic classes by choosing the 100 most common words in these descriptions, and training preliminary classifiers for each one. We then sorted the words by the accuracy of these preliminary classifiers, and further filtered it down to 45 by choosing concepts that were semantically meaningful, reasonably successfully detected, and not redundant with other selected terms. The full list of 45 terms is in table 1.

Since our semantic classification features are to be provided at 2 sec resolution, we need to train on frames of this size. The training labels we have, however, are at the level of video or sound clip – typically much longer. For instance, a video tagged as containing "Music Performance" may include several windows of non-music sound prior to the performance beginning. Training a "Music Performance" classifier on these frames of generic background noise might hurt its discrimination.

Table 2: Mean Average Precision for labels aggregated to the clip level from the iterative relabeling of 2 sec audio clips, for the 10 sec segmented MED2010 corpus.

Epoch	mAP
1	0.523
2	0.537
3	0.559
4	0.562
5	0.559

To address this, we developed a Multiple-Instance Learning (MIL) procedure. MIL refers to the scenario in which data points belong to "bags", with labels that indicate whether a particular bag contains any items of that class. Thus a bag with a negative label will consist only of negative examples, but a bag with a positive label will in general contain a mixture of both positive and negative examples (like the frames in our Music Performance video). Our procedure first trains classifiers assuming all frames in the positive bags are true positives, and attempts to discriminate them from the frames in the negative bags. Every item in each positive bag is then submitted to a classifier (with cross-validation, so a classifier is never applied to frames used in its training), and any frames that fall below some threshold in classifier score are relabeled as negative, with the constraint that at least one frame in the bag (and any other frames whose classifier scores are very close to this "best" frame) must retain positive labels. Ideally, this will remove negative examples from the positive pool; classifiers are then retrained, and the process repeats until no further increases are observed on held-out development data (where a simple combination rule is used to produce a clip-level label from individual frames).

Our approach to choosing the threshold was to create histograms of the classifier scores from the negative and positive frames, calculate the cumulative distribution functions in opposite senses (i.e.,  $P(\text{score } < \theta)$  for the negative frames, and  $P(\text{score} > \theta)$  for the positive frames), then choose the threshold  $\theta$  where they intersect. By scaling one function prior to finding the intersect, the threshold can be made to remove frames from the positive bags more or less aggressively. Table 2 gives an example of the changes in test set mean Average Precision for iterative relabeling of the five frames in each 10 sec clip of our segmented MED2010 set; performance at the clip level improves for 4 epochs, then gets worse, so the labels for the 4th epoch are used as the basis for the final 2 sec-resolution semantic classifier. Note that we do not have any ground-truth labels at 2 sec resolution, so we cannot directly measure the frame-level classifier performance.

Audio classifiers for the 100 semantic classes were trained on both types of raw audio features, MFCC statistics and Auditory Model histograms (Section 1.2.5). Because a single video may contain hundreds of 2 sec windows, and because SVM distance matrix calculation is  $O(n^2)$  in the number of frames, this training was far more computationally expensive than training whole-video classifiers. Further, because the histogram features relied on a chi-squared distance measure, which is around 10 times slower to compute than the euclidean distance used for the statistics, it was not possible to complete labeling of the MED12 development data in the available time; at 2 sec level, our 100 audio semantic features are based only on MFCC statistic features. At whole-video level, however, we were able to train separate sets of semantic classifiers on both low-level audio features. The outputs of these classifiers - SVM distance-to-boundary scores, which have been found to be largely comparable between different classifiers - were then combined by simple summing to create a set of fused audio semantic features.

To evaluate these semantic features, and to compare them to the raw features, we used a task based on the first release of the MED2011 DEV data and the example videos for events 001 to 015. This gives a combined pool of 6354 videos, which were broken into 5 cuts, with classifiers trained on 4/5ths of the data and tested on the remaining 1/5th. For this test, all features were at the whole-video level. 15 independent per-event classifiers were trained using the different feature sets. The results are shown in Figure 10. We see that the raw MFCC and Auditory Model (sbpca) features have different strengths, with the Auditory Model features doing particularly well for E004 Wedding Ceremony, E008 Flash Mob Gathering, and E012 Parade. The semantic features broadly reflect the raw feature performance, but do better in some cases, such as E001 Attempting a Board Trick and E009 Getting a Vehicle Unstuck (at least for MFCCs). Finally, the fused semantic features (sema100sum) are generally successful in capturing or improving on the best individual feature in each category, delivering a 15% relative improvement in mean Average Precision, from 0.19 for the raw MFCC features to 0.22 for the fused semantic features.

#### 1.2.10 Dynamic Actions Semantic Modeling

The 51 Dynamic Action concepts were modeled on top of the 5000-dimensinal bow representation of MBH descriptors. Training clips were employed from the publicly available datasets HMDB [7] and UCF101 [15]. Each concept was modeled as a histogram intersection kernel SVM, with probabilistic output as detailed in Section 1.2.7.



Figure 10: Audio-based Average Precision results for MED2011 event classifiers (6354 video set), comparing classifiers based on raw features (MFCCs and Auditory Model (sbpca)), and on 100-class semantic audio features based on each set of raw features (sema100mfcc, sema100sbpca), and fused by summing (sema100sum).

# 1.2.11 Temporal Granularities and Video Level Semantic Aggregation

We experimented with Semantic representations extracted at four different temporal granularities:

- 1. *Frames*: semantics are represented at the level of the frames which are extracted from the video. We sampled frames at a rate of 0.5 frames per second.
- 2. *Keyframes*: result of low level histogram descriptors based shot detection
- 3. *Intervals*: result of a Semantic based analysis and of variable length for each semantic
- 4. *Video*: result of an aggregation from frame level to entire video clip

In all the submitted run we employed the Video level representation using the *top-quartile* aggregation method, which is computed as the average of the top quarter responses of each semantic classifier over the frames in the video. While active research is being performed on the other temporal granularities and also aggregation methods (for example, Max aggregation) for the purposes of event retrieval, we went forward with topquartile due to its predictable performance.

### 1.3 Event Content Description (ECD) Generation

For the 10EX and 100EX conditions, in which we have event training clips to be used, we consider the problem of generating ECD from VCDI in the framework of the following figure. Especially we are interested in the aspect of semantic modeling.

Multiple concepts search (added in the order of frequency)							
concept_seleted	+Activity	+Activity+Dominant Color	+Activity+Domini Color+General Setting	aant +Activity+Dominant Color+General Setting+Human Sport	+Activity+Dominant Color+General Setting+Human Sport+Non-Building View	+Activity+Dominant Color+General Setting+Human Sport+Non-Building View+Non-Chart Type	+Activity+Dominant Color+General Setting+Human Sport+Non-Building View+Non- Chart Type+Non_Animal
true detection	100	100	100	100	100	100	100
false alarm	900	900	900	900	900	900	849
Multiple concepts search (reverse order of backgFreq)							
concept_seleted	+Commo Object	+Common Object+Upper Human Body (Head and (Freedom)	Common bject+Upper uman Body fead and Inc	Common Object+Upper uman Body (Head and toulder)+Adult+Unknows doors	+Common Object+Upp Human Body (Head and Shoulder)+Adult+Unkno Indoors+Animal Groom	+Common Object+Upper Human Body (Head and Shoulder)+Adult+Unknown wm Indoors+Animal ing Grooming+Conventional La	1 and

Figure 11: Different tasks correspond to different choices of optimal features.

To handle the diversity of video contents, we consider several representations for a video, namely frame-level, keyframe-level, and interval, and video level. For each level, we take semantic VCDI with the same format, and create different ECD generators.

The simplest ECD generator is to count the frequency of different concepts in event kits. For example, we count in the Birthday party event the frequency of concepts "Adult", "Individual", "Activity" are 0.863, 0.988, 0.904, respectively. As a result, the ECD investigator will rank these concepts in the order of frequency.

The benefit of counting method lies in the fact that it is simple and intuitive, and we can easily use this method for incremental and distributed computing. However, this simple method overlooks the correlation between concepts, as well as the distribution in the background set. We will develop two other methods.

The second method is entropy based method. It considers the dependency and overlapping of multiple concepts, and selects the concept which maximizes the conditional entropy for the given event. The following figure illustrates how we select concept using entropy based method.

The third method treats the ECD model as a combination of decision trees, where each node denotes a selection of semantic concept. By traversal in the tree, we can find multiple paths from node to leaves. Every path corresponds to a logical expression. This method is easy to interpret, and fit for the scalable learning. However, we need make more efforts to make use of the logical expression in MER and MED retrieval.

The fourth method is to try greedy method to search for the semantic queries. An example is illustrated in Fig. 11.

The last method is to combine logical expression for event query generator. We realize that not every possible logical expression can be used for the retrieval system. On the one hand, a complicated event query will be difficult to interpret intuitively. On the other hand, a very long event query will make the retrieval time very long. The non-friendly event query interface will prevent the user from effectively interacting with the system and hence reduce the use of the system. Even worse, if we cannot attract users to use our system, we will not be able to leverage human intelligence to design effective event models.

As a solution, we currently use a limited number of logical operators: weighted combination, average, AND, OR. Although the limited operators may decrease the performance a little bit, we believe it is better to focus on user interface than to improve the performance marginally.

Comparing AND and Weighted Operators: AND (average) and Weighted Average are two popular operators in our systems. We think it is worth the efforts to compare the performance of two operators. Take E012 as an example, the AND operator is "Parade AND Podium outdoor AND Street scene AND Dragonboat AND Social and religious event".

We see the following observations in our experiments: The performance of weighted operator is slightly better than the AND operator, however the difference is not significant when there are only a few examples. There is a gap between the performance on training set and testing set.

# 1.4 Event Search

We centered the attention of our Event Search system around the human user. We implemented semantic concept keyword based search to enable event-based search by users. The user interacts with our system through an interface (represented in Figure 12) where he can insert various combinations of semantic query terms, from the vocabulary of 1447 concepts in our audiovisual taxonomy.

Among the functionalities of the Event search interface are keyword auto-completion and, most importantly, the possibility of composing multiple keywords searches using logical expressions, namely AND, OR and expressions grouping via parenthesis (see detail in Figure 12 (a) ). For any given query, the interface returns the ranked list of videos in a given collection, where each video is represented by a mosaic image icon of its keyframes. Also, the score of the classifiers being queried is reported below each clip mosaic. By hovering over the video icon, a animated gif of the video keyframes is played, giving the user a quick glimpse of the content of the full video. By clicking on the video icon, the user is presented to the video VCD browser view (Figure 12 (b)), which allows to further verify the Semantic evidence that prompted the video in such rank position, as well as a quick summary of the top ranked concepts for each facet in our taxonomy on the given video clip. The user can also find clips which are semantically similar (i.e., for which the top scoring semantics for each facet have similar scores) to a given one by clicking on the LIKE button under its icon. The system will automatically generate a weighted logical expression on such semantics in order to retrieve similar clips.

The system allows disjunctive combinations of conjunctive queries. The terms and clauses can be optionally weighted or thresholded. Example queries are as follows:

- Car AND Towing\_a\_vehicle
- Zoo X Zebra X Animal\_grooming X Animal X Animal\_feeding X -1 Adult X Brush\_hair\_ACTIONS X Clean\_ACTIONS X Barn X -1 Auto\_part X Audio\_animals X Audio\_animal
- Home\_appliance AND Kitchen AND Kitchen\_appliance AND machine\_operation\_and\_maintenance\_activity AND Home\_Appliance AND Adventure\_Land\_Sport AND Disposal AND laundromat

Multiple conjunctive operations are supporting including AND (average), X (product) and MIN (minimum). All operations are performed over probabilistically normalized concept classifier scores, scaled between 0.0 and 1.0 (Section 1.2.7), and aggregated to video level representations as described in Section 1.2.11.

In the cases of 10EX or 100EX, where event kit videos can be used to aid the search, the event content description generation module is used, as described in Section 1.3, to suggest a query expression, which can then be edited by the user.

# 1.5 Submitted Runs

We submitted 9 pre-specified event search runs, and 9 ad-hoc event search runs for each given event. These included the 3 exemplar cases: 0 examplars, 10 examplars, and 100 examplars. In addition, for each event and exemplar condition, we submitted results for each of three modalities: visual only, audio (non-ASR) only, and combined, according to which concepts of the semantic taxonomy concepts we employed to generate event search queries.

In the following we describe the details of how the submitted runs were produced.

#### **Pre-Specified:**

0EX: a different person from our group was given the list
of concepts in the taxonomy and randomly assigned event
queries to produce by using the system interface loaded
with the MEDTEST collection, without any ground truth
information about the events visible in the interface. The
user would then proceed to produce the query by qualitatively analyzing the top page (with 50 items) of the ranked

results produced by the weighted logical expression he generated as query. The preduced queries were then used to score the PROGTEST collection and produce the submitted results.

• 10EX and 100EX: for these conditions, the ECD Generator was used on the EventKit and Event Background collection to produce an automatically generated query, to be used as suggested starting point for a user who was then free to edit and modify such query by using the system interface on the unlabeled MEDTEST collection, as in the 0EX condition. The only difference between our 0EX and 10(0)EX runs was only in the suggested query strings.

Below is the complete list of pre-specified events submitted runs.

- IBM-Columbia\_MED13\_FullSys\_PROGAll\_PS\_0Ex\_1: Pre-Specified, 0 exemplar condition, visual + audio combined.
- IBM-Columbia\_MED13\_FullSys\_PROGAll\_PS\_10Ex\_1: Pre-Specified, 10 exemplar condition, visual + audio combined.
- IBM-Columbia\_MED13\_FullSys\_PROGAll\_PS\_100Ex\_1: Pre-Specified, 100 exemplar condition, visual + audio combined.
- IBM-Columbia\_MED13\_VisualSys\_PROGAll\_PS\_0Ex\_1: Pre-Specified, 0 exemplar condition, visual only.
- IBM-Columbia\_MED13\_VisualSys\_PROGAll\_PS\_10Ex\_1: Pre-Specified, 10 exemplar condition, visual only.
- IBM-Columbia\_MED13\_VisualSys\_PROGAll\_PS\_100Ex\_1: Pre-Specified, 100 exemplar condition, visual only.
- IBM-Columbia\_MED13\_AudioSys\_PROGAll\_PS\_0Ex\_1: Pre-Specified, 0 exemplar condition, audio only.
- IBM-Columbia\_MED13\_AudioSys\_PROGAll\_PS\_10Ex\_1: Pre-Specified, 0 exemplar condition, audio only.
- IBM-Columbia\_MED13\_AudioSys\_PROGAll\_PS\_100Ex\_1: Pre-Specified, 0 exemplar condition, audio only.

#### **Ad-Hoc:**

• 0EX: for this condition, the system was used exactly in the same manner as the pre-specified case.

#### Home Classifiers Events Queries GT Random Help

ity AND 20 Cla ssifiers@Car AND 20 C ring AND Class ifiers@Sedar

10 Classifiers@Transportation\_activity AND 20 Classifiers@Car AND 20 Classifiers@Driving AND Classifiers@Sedan Search Clear

Add related concepts: AUTO RACING I BACKTRALL COURT OUTDOOR | FOR STATUDOR | FIRE STATION | General State State Court outbook | State Stat **Clear)** 17000r | LAND VEHICLE | light truck | manufactured DLL PLAZA | TOWING A VEHICLE | truck | urban scene | V OR TOLL PLAZA TOWING A VI

#### Search Results

10 Classifi 🖬 🖷 Page 1 of 312 🕨 🗎 Go to page 1 💌 Results 1 to 80 of 24956





np4 (0.668) LIKE

Semantic Concepts: [ Sort by: Score Name ] [ Display: Al Above threshold ] [ Show: Scores Discrimination scores ]							
Actions facet	Activity facet	Animal facet	Audio facet	Object facet	People facet	Setting facet	Type facet
	Non human event [0.98]	Non animal [0.99]	Audio other [1]	Object [1]	Non view of human [0.99]	Non building view [1]	Dominant color [1]
	Auto racing [0.96]	Whale [0.78]	Audio ski [1]	Sedan (0.94)	Non people [0.99]	Boat deck [0.99]	Color [0.97]
	Non human activity [0.96]	Crab [0.75]	Audio wedding [0.96]	Truck [0.93]	Non human [0.98]	Sky scene (0.99]	Non chart type [0.96]
	Bobsledding [0.95]	Killer whale [0.73]	Audio museum [0.95]	Land vehicle [0.93]	Automobile driver [0.9]	Pilothouse indoor [0.98]	No horizon [0.91]
	Non sport [0.94]	Goldfish [0.69]	Audio biking [0.94]	Ship [0.92]	Human body view [0.85]	Non sky scene [0.98]	Rubber plastic [0.86]
	Motor sport [0.93]	Pigeon [0.56]	Audio beach [0.93]	Fire truck [0.91]	People with affiliation [0.81]	Harbor [0.98]	Gray [0.84]
	Changing tire [0.93]	Duck [0.53]	Audio traffic [0.92]	Vehicle [0.9]	Veterinarians [0.75]	Arrival gate outdoor [0.98]	Orange [0.8]
	Non human action or	Swan [0.52]	Audio one person [0.86]	Unknown land vehicle type	Human hands [0.67]	Poolroom establishment	Red [0.78]
	gesture [0.92]		Audio warfare [0.86]	[0.89]	Mechanic [0.66]	[0.97]	Glass [0.75]
	Fishing [0.92]		Audio street [0.81]	Car [0.89]	Boy [0.64]	Runway [0.97]	Stressful [0.73]
	Disturbance [0.9]		Audio bird [0.8]	Bus (0.88)	Full human body [0.6]	Outdoor [0.97]	Metal [0.71]
	Human activity [0.9]		Audio musicperformance	Light truck [0.87]	Prisoner [0.59]	Game room [0.96]	Electric indoor lighting
	Driving a car [0.89]		[0.79]	Dinette vehicle [0.86]	Motorcycle cop [0.51]	Poolroom home [0.96]	[0.7]
	Surf casting [0.89]		Audio nonmusicperformance	Cargo ship [0.86]		Nuclear power plant outdoor	Matte [0.66]
	Bathing baby [0.88]		[0./1]	Ambulance vehicle [0.84]		[0.96]	Vinyl linoleum [0.66]

Figure 12: MED Event Search interface. (a) the user can insert any combination of semantic keywords and the interface will display a retuned list of results with keyword scores under each mosaic image. (b) by clicking on the mosaic icon, the user is prompted to a detailed view of the taxonomy facets top scoring concepts for the given clip.

- 10EX: for this condition, we ingested the EventKits (10 per event) and Event Background videos into the system, together with the ground truth. Multiple users were then randomly assigned to event searches, to be performed on such collection. The event ground truth for any given semantic query was exposed to the user as a feedback to refine/improve such query under the form of: 1. color coded visual information (green box around mosaic thumbnails of true positives in the first page of returned search results, read box for negative ones); 2. Average Precision (AP) score at full rank of the given search. The rationale behind the use of such interface with exposed ground truth was to provide the means to a user to produce a query both semantically meaningful (since a human was picking the keywords to insert in each query) and well performing (since the user could adapt his query to improve AP score and number of true positives in the first page of returned results. Figure 13 reports the detail of the feedback-based interface for the EventKits + Background collection given a query for Event E037 - Parking a vehicle. In this particular case, we also exploited redundancy, under the form of multiple users being assigned the same event search task. The qualitative results on the MEDTEST collection (visible through the system interface) and the quantitative results (in terms of AP at full rank) a newly formed collection containing the EventKit videos (10 per event) as positives and the ResearchSet videos as negatives was used to select or combine queries from multiple users.
- 100EX: for this condition, differently from the prespecified case, we directly employed the queries automatically generated by the ECD Generator applied to the EventKits (with 100 videos per event) and Event Background collections.

Below is the complete list of ad-hoc events submitted runs.

- IBM-Columbia\_MED13\_FullSys\_PROGAll\_AH\_0Ex\_1: Ad-Hoc, 0 exemplar condition, visual + audio combined.
- IBM-Columbia\_MED13\_FullSys\_PROGAll\_AH\_10Ex\_1: Ad-Hoc, 10 exemplar condition, visual + audio combined.
- IBM-Columbia\_MED13\_FullSys\_PROGAll\_AH\_100Ex\_1: Ad-Hoc, 100 exemplar condition, visual + audio combined.
- IBM-Columbia\_MED13\_VisualSys\_PROGAll\_AH\_0Ex\_1: Ad-Hoc, 0 exemplar condition, visual only.
- IBM-Columbia\_MED13\_VisualSys\_PROGAll\_AH\_10Ex\_1: Ad-Hoc, 10 exemplar condition, visual only.



Figure 14: Our system in green. Both quality and quickness are mutually attainable.

- IBM-Columbia\_MED13\_VisualSys\_PROGAll\_AH\_100Ex\_1: Ad-Hoc, 100 exemplar condition, visual only.
- IBM-Columbia\_MED13\_AudioSys\_PROGAll\_AH\_0Ex\_1: Ad-Hoc, 0 exemplar condition, audio only.
- IBM-Columbia\_MED13\_AudioSys\_PROGAll\_AH\_10Ex\_1: Ad-Hoc, 10 exemplar condition, audio only.
- IBM-Columbia\_MED13\_AudioSys\_PROGAll\_AH\_100Ex\_1: Ad-Hoc, 100 exemplar condition, audio only.

# 2 Multimedia Event Recounting (MER)

The current IBM-Columbia MER system is based on the close connection between our ontology of semantic classifiers and the functional aspects of natural language. Our system is informed throughout by human psychology and user studies. We outline the nine people-oriented principles behind our MER processing pipeline, and give the reasons and evidence for each of our nine design decisions.

These decisions have led to a system that produces above average clarity of explanation (NIST MER evaluation criterion 2), while using less time for an analyst that is less than one-sixth video time (NIST MER evaluation criterion 3). This places the time performance of the the IBM-Columbia system well above all other NIST MER competitors. Fig. 14, which shows our system circled in green, illustrates that our clarity score (horizontal axis) and our time efficiency (vertical axis, representing



Figure 13: MED Event Search interface with ground truth feedback for EventKits (10 examplars per event)+Backgoround collection. The feedback on the query [10Transportation\_activity AND 20Car AND 20Driving AND Sedan] is provided under the form of 4 true positives (highlighted in green) results reported in the top 10 of the ranked list, as well as the AP score at full rank of 0.244.

speedup on a logarithmic scale) indicates how both quality and quickness are mutually attainable. In fact, the lines represent the linear regression of the two against each other: overall, the faster systems were also the clearest.

For each video that our MER system was presented for recounting, we were given the ECD terms used to define the event, plus a matrix of classifier scores to represent the response of the MED system. Rows of the matrix represented normalized classifier outputs, and columns of the matrix represented two-second intervals of the video. Our goal was to determine the best local snippets of the video that would support the overall global MED determination of event presence.

We exploited the following observations:

1) Semantic ontologies are like natural language: People tend to view and describe the world using a number of categories, at various levels of generality. Our visual, sound, and motion classifiers have been embedded into an ontology tree that has approximately 1400 concepts. This tree is actually a forest of "facets", each of which reflecting a generic category of concepts, such as "people", "actions", "settings", etc. Each of these dozen facets have been designed to reflect the structure of what are called "thematic roles" in linguistics. Roughly, these correspond to subject, verb, prepositional phrases of place, etc. Each facet then has further multiple levels of specificity, for example, "object", "animal", "vertebrate", "mammal", "dog", etc. This provided our MER system with a wide range of tradeoffs for selecting the proper verbal output, based on classifier accuracy. People prefer specific, accurate descriptions over generic, inexact descriptions, and so does our MER.

2) MER has different goals than MED: We had noted in prior work that people generally do not edit YouTube videos, and so there is usually great visual and audio redundancy within a video. And, at the same time, there are few if any shot boundaries within a video. We therefore found it helpful in design to view MED and MER as achieving different purposes. MED finds videos that are examples of events, based on global information that is distributed over the entire video's length. In contrast, MER persuades people of the correctness of our detection, based on local information that is usually perceivable within seconds. MER therefore seeks the minimal amount of local evidence that is consistent with the global MED determination. This enabled a very high degree of temporal compression, as the NIST competition confirmed.

3) People use small queries: Through experimentation, we noted that people tended to use quite short query strings with our video library browser, and not the full 1400 concepts we

have made available. So, our MER output was also designed to describe video contents by only using the actual query strings used in MED triage and retrieval (i.e., the ECD). This resulted in MED passing to MER a very lightweight matrix of classifier scores to process. Usually a video has only between 2 and 30 rows, one per classifier, and about 75 columns, one for every two seconds of video. Such a matrix can be easily experimented with and tuned to match user expectations.

4) Video segmentation should be semantic: In prior work, we documented that people's attention span in viewing videos is limited by short term memory. This is also reflected in studies on the well-documented distributions of edited shot lengths. But we have found through experimentation that even in YouTube videos, shot-like semantic segments can be found that are defined by temporal clusters of event-specific semantics. These correspond to units of attention, rather than units of color, texture, motion, or sound. Our segmentation of videos for MER is therefore purely based on semantic coherence, which we borrowed from our previous work; it found that video memory is no longer that about 30 seconds.

5) Description should focus on accuracy: We have noted in the literature of discourse that people have a good sense about how to trade off specificity against accuracy, and how to choose the appropriate level of nouns and adjectives to do so. We also noted through experimentation and through user studies that people also tended, as a rule, to search for videos using "middle level" concepts, "tree", rather than either "plant" or "maple". Further, they tended to describe videos using accurate super-concepts ("definitely an animal") over approximate sub-concepts ("maybe a dog"). We therefore designed our algorithm to do so likewise, based on an information-theoretic measure of weighted probability, which modifies the probability of of a classifier reliability, with an approximate measure of the information gain that a term provides within the ontology tree. Thus, we can decide, in a human-tuned manner, whether to say "animal", "dog" or "boxer", even if the classifier scores are inconsistent.

6) Snippets should show discriminating concepts: We noted in the psychology literature that people tend to remember events differentially, by what makes them distinguished from other events, rather than absolutely by the content. We noted through experimentation that it is more often the second best video snippet that is more salient to a viewer, since the first best snippet tends to win on total points, but not on salient uniqueness. We also noted that, the longer the query string and the longer the video duration, the more snippets may be needed for "insurance" coverage for MER. We have tuned our snippet sorting and selection algorithm to reflect this: we prefer to display first those snippets that are a bit off-center, then follow them with more ordinary ones. 7) People like short text: In our MER experiments, we noted that people have a limited ability to absorb a long list of concepts. We found that there was a small fixed limit to the number of concept words that were helpful in each textual description; that length is about five.

8) MTV-like snippets say enough: It is clear that people are very good at understanding imagery even at tachistoscope rates. Through our experimentation, we found that video snippets rarely needed to be longer than four seconds in order to be an effective proof of event presence. Our algorithm currently finds the best internal four second subsegment of each full semantic segment, in order to serve as a representative of the full snippet. We are aware this is only a heuristic approximation; in fact, our evidence suggests that sometimes four seconds is too long.

9) Concise telegraphic textual output is best: People tend to ignore function words when they read; our user studies on our prior MER system confirmed this. So, our current MER text output generator also does. Each semantic segment with high classifier responses generates a short word list for each snippet, and uses it uses typography to make its point quickly. A typical output, for example, is: "VISUAL: Birthday\_party, Bazaar\_indoor; ACTION: Blowingcandles\_actions; AUDIO: Noisy\_audio, Birthday\_audio." We found that anything further is ignored, anyway.

In summary, these observations and studies have led to MER algorithms that greatly exceeded the state of the art in analyst time savings, while being comparable to the state of the art in description clarity. We continue to explore further improvements, including the possibility of deriving MER descriptions independently of the event description.

#### **3** Surveillance Event Detection (SED)

#### 4 Overview

We present a system for detecting events in surveillance videos and evaluate it in the SED task of TRECVID. The evaluation consists of two parts: automatic event detection (*retrospective*) and interactive event detection with human in the loop (*interactive*). The retrospective system uses a joint-segmentationdetection framework with temporal interdependencies among events considered to enhance detection. The interactive system is designed to enable simultaneous verification of multiple events with temporal relationships. This has been shown in our experiments advantageous over looking at a single event alone each time as the approach leverages the strong temporal patterns exhibited in the data. With this design, we further propose a method to present detection results to the end user effectively by risk analysis, which ranks events by combining the margin between top two candidate detections and their temporal relation ships.

# 5 Automatic Event Detection

In the retrospective task, unlike most previous work on event detection that treat video segmentation and event classification separately [10], our approach performs video segmentation and classification jointly with a temporal model. The motivation behind temporal modeling is to exploit temporal dependencies that often exist between events to enhance detection.

Our classification model is trained discriminatively using Multi-class SVM [3] with Fisher Vector (FV) encoded features [13] while the temporal relationships between events are modeled by priors estimated using ground truth. Note that we treat background clutter as a *no-event* class, which is trained together with the other 7 events of interest using [3].

We formulate the problem of joint segmentation and classification by a general quadratic integer programming framework. Given a video  $\mathbf{X}$ , let  $\{t_1, t_2, ..., t_n\}$  to be *n* segmentation points. The  $i_{th}$  segmentation of  $\mathbf{X}$  is  $S_i$ , where  $S_i = \mathbf{X}(t_i, t_{i+1})$ . We assign each segmentation  $S_i$  into a total *K* binary variables  $\zeta_i^k \in [0, 1], k = 1, 2, ..., K$ , where  $\zeta_i^k = 1$  indicates that the label of segmentation of  $S_i$  is *k* (including *no-event*). The object function  $f(\mathbf{X}, K)$  that should be maximized for an optimal segmentation can be written as follows,

$$\mu \sum_{i=1}^{n} \sum_{k=1}^{K} \zeta_{i}^{k} \varphi^{k}(S_{i}) + (1-\mu) \sum_{j=1}^{n} \sum_{j'=j+1}^{n'} \sum_{k=1}^{K} \sum_{k'=1}^{K'} p^{t}(k,k') \zeta_{j}^{k} \zeta_{j'}^{k'}$$
(2)

where  $\varphi^{k}(S_{i})$  is the SVM score and  $p^{t}(k, k')$  is the prior relation weight between event k and k'. There are some constraints for the objective function:

$$n' \leq n$$
  
$$\forall i : \sum_{k=1}^{K} \zeta_i^k \leq 1 \quad (3)$$
  
$$\forall i, \forall i', \forall k, \forall k' : \zeta_i^k + \zeta_{i'}^{k'} \leq 1 \quad \text{if} \quad S_i \bigcap S_j = 0$$

We show that the optimization can be solved efficiently by dynamic programming if only first-order dependency is considered, i.e. n' = j + 1 in the equation above. Given any video flip  $\mathbf{X}_{(0,u)}$  with length u,  $f(\mathbf{X}_{(0,u)}, u, K) = \operatorname{argmax}_{l_{\min} \leq l \leq l_{\max}} f(\mathbf{X}_{(0,u-l)}, u - l, K) + f(\mathbf{X}_{(u,u-l)}, l, K)$ .  $l_{max}$  and  $l_{min}$  are the detection length of video frames.

# **6** Interactive Event Detection

Compared to our interactive system from last year, we made two significant changes to the system this year. Our first contribution is development of a system that allows for verification of multiple events simultaneously together with temporal patterns (Fig 6). Our second contribution is development of a method to present detection results to the end user more effectively by risk ranking. Here the risk score of a detection (including nonevent) indicates the value or impact that the detection being verified, if corrected by the user, has on the system performance. In another word, a false detection with a lower risk score will attribute more to the performance after it gets corrected during the interaction. In our system, we compute both single events and pairwise events with temporal dependences and present the results to the user in an ascending order of the risk scores of all events.

Next we describe how we measure the risk of detection by maximizing the margin of top two candidate events for each detection while temporal relation of events and potential penalty weight are also considered. Given a segment  $S_i$ , let the top two detections be the *k*th and *k*'th event with scores  $\varphi^k(S_i)$  and  $\overset{k'}{\downarrow}(G)$ . The formula formula is a first second se

 $\varphi^{k^{'}}(S_{i}).$  The formulation of the risk score for a single event can be expressed as:

$$R(S_{i}) = \frac{1 - (\varphi^{k}(S_{i})p(k) - \varphi^{k'}(S_{i})p(k'))}{||S_{i}||} \cdot \begin{cases} w_{m} \\ w_{f} \end{cases}$$
(4)

where  $w_m$  is the cost of a miss detection,  $w_f$  is the cost of a false alarm and  $||S_i||$  is the length the segment  $S_i$ .  $p(k), k \in 1, 2, ..., K$  is the occurrence prior of event k learnt from the ground truth. Similarly, we can develop the formulation for a pair of events, which is omitted here due to space limit.

#### 7 Experimental Results

**Retrospective Task** The joint-segmentation-detection approach requires event classification at varied lengths of segments. In our experiments, we trained a classifier for each event using Multiclass SVM at a window size of 30, 60, 90 and 120 frames, respectively. We then performed segmentation by dynamic programming as described in section 2. It's noteworthy to mention that our approach performs event classification and segmentation simultaneously, so no thresholding is needed for the final detection. in Table 1, we show the results of our approach using 1) dynamic window sizes (d-Joint) and 2) a fixed window size (i.e. 60 frames) (f-Joint) on the evaluation SED data set, and compare the results with the Fish Vector encoding approach



Figure 15: Interactive System Design Panel.

	Table 3:	DCR of	ı Eval	Dataset of	Retrospecti	ve
--	----------	--------	--------	------------	-------------	----

Method	FV-60	f-Joint	d-Joint
CellToEar	1.0007	1.0001	0.9985
Embrace	0.8000	0.8225	0.7818
ObjectPut	1.0026	0.9618	1.0046
PeopleMeet	1.0362	1.0524	1.0267
PeopleSplitUp	0.8433	0.8837	0.8364
PeopleRuns	0.8346	0.8488	0.7887
pointing	1.0175	1.0626	1.0045

from our 2012 submissions. Clearly, our proposed approach outperforms the FV-based approach, demonstrating the effectiveness of temporal information.

**Interactive Task** We reported below the formal evaluation results provided by NIST, to exam the effectiveness of our proposed interaction methods. In table 2, we compared the actual DCRs on the evaluation set using three interaction strategies: (1) no interaction (Retro), (2) event ranking based on detection confidence scores (Confidence), (3) event ranking based on risk scores (Risk). We observe that an interactive process can improve the system performance. With risk analysis, the improvements are significant compared to the naive approach purely based on detection confidence scores.

### 8 Semantic Indexing (SIN)

For our Semantic Indexing (SIN) submission concept detection task, we submitted score results from 3 variants of our IMARS modeling framework. Much of the IMARS framework is described in Section 1.2. However, there are some variations to the system used for SIN as compared for MED. The differences are described in the following sections.

Table 4: DCR on Eval Dataset of Interactive					
Method	Retro	Confidence	Risk		
CellToEar	0.9985	0.9943	0.9956		
Embrace	0.7818	0.7403	0.7337		
ObjectPut	1.0046	1.0053	0.9928		
PeopleMeet	1.0267	0.9875	0.9584		
PeopleSplitUp	0.8346	0.8700	0.8489		
PeopleRuns	0.8346	0.7383	0.6445		
pointing	1.0045	0.9800	0.9781		

### 8.1 Data Collection

We used 1 frame per shot, those supplied with the IACC datasets. For the test set, where multiple frames per shot were given, we ran our classifiers on the first frame in the shot.

Data was organized to collect up to 5000 positive and 5000 negative examples per concept.

#### 8.2 Feature Extraction

#### 8.2.1 SIFT Descriptors

In addition to low-level features described in MED, we utilized 5000 dimensional SIFT descriptors extracted around Harris Laplace interest points. Each keypoint is described with a 128-dimensional vector containing oriented gradients. We obtain a visual words dictionary of size 1000 by running K-means clustering on a random sample of approximately 300K interest point features, we then represent each image with a histogram of visual words. We extracted two codebooks, starting from two different random samples of points. We used soft assignment following Van Gemert et al. using sigma = 90. This descriptor was extracted using the executable publicly available from the University of Amsterdam. We extracted also variations of the SIFT descriptor in different color spaces, namely rgb, hsv and opponent channels.

#### 8.2.2 High-Level Semantic Descriptors

We utilized 709 of our semantic concept detectors, trained with both linear and non-linear SVMs, as a low-level feauture for concept modeling. In internal data split experiments, the semantic feature descriptors were our top performing features, although linear performed similarly to non-linear kernels.

# 8.3 Ensemble Learning

As described in Section 1.2.7, the IMARS concept learning pipeline utilizes an ensemble learning approach from a collection of "bags," which are data subsamples across both feature and data space. Within each bag, the SVM learning problem can be constructed using a variety of methods to deal with data imbalance.

- 1. Data in the majority class is simply randomly undersampled until it is balanced with the minority class.
- 2. Data in the minority class is oversampled using a proprietary variation of SMOTE.
- 3. Data is not balanced.

### 8.4 Submitted Concept Detection Runs

- 1. For the first run, we used a data sampling bag size that balanced the data depending on how many positive examples were available for the concept. Enough balanced bags were trained to achieve 90% data coverage.
- 2. For the second run, we used data sampling rate at a maximum of 5000 positive and 5000 negative examples per bag, with balanced bags of all available data using proprietary SMOTE variant, 1 bag per feature. Since the validation split is used to only select features, models were retrained on all data after ensemble fusion.
- 3. For the third run, we used data sampling rate at a maximum of 5000 positive and 5000 negative examples per bag, with unbalanced bags of all available data, 1 bag per feature. Since the validation split is used to only select features, models were retrained on all data after ensemble fusion.
- 4. For our fourth run, we performed a rank fusion among our first 3 runs.

#### 8.5 Submitted Concept Pair Detection Runs

- 1. Our first concept pair submission comes from a combination of concept detectors developed from Run 3, using Rank Normalization for concept detector fusion.
- 2. Our second concept pair submission is similar to the first, using sigmoid normalization instead of rank normalization.

# References

 Murray Campbell, Alexander Haubold, Ming Liu, Apostol Natsev, John R. Smith, Jelena Tesic, Lexing Xie, Rong Yan, and Jun Yang. Ibm research trecvid-2007 video retrieval system. *Proc. NIST TRECVID Workshop*, 2007.

- [2] Liangliang Cao, Shih-Fu Chang, Noel Codella, Courtenay Cotton, Dan Ellis, Leiguang Gong, Matthew Hill, Gang Hua, John Kender, Michele Merler, Yadong Mu, Apostol Natsev, and John R. Smith. Ibm research and columbia university trecvid-2011 multimedia event detection system. *Proc. NIST TRECVID Workshop*, 2011.
- [3] Koby Crammer and Yoram Singer. On the algorithmic implementation of multiclass kernel-based vector machines. *J. Mach. Learn. Res.*, 2:265–292, March 2002.
- [4] Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. Liblinear: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874, 2008.
- [5] A. Haubold and A. Natsev. Web-based information content and its application to concept-based video retrieval. In ACM International Conference on Image and Video Retrieval (ACM CIVR), 2008.
- [6] Yu-Gang Jiang, Guangnan Ye, Shih-Fu Chang, Daniel P. W. Ellis, and Alexander C. Loui. Consumer video understanding: A benchmark database and an evaluation of human and machine performance. In ACM International Conference on Multimedia Retrieval (ICMR), Trento, Italy, Trento, Apr 2011.
- [7] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre. Hmdb: A large video database for human motion recognition. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 2556–2563, nov. 2011.
- [8] Keansub Lee and D. P. W. Ellis. Audio-based semantic concept classification for consumer video. *IEEE Tr. Audio*, *Speech, Lang. Proc.*, 18(6):1406–1416, Aug 2010.
- [9] R.F. Lyon, M. Rehn, S. Bengio, T.C. Walters, and G. Chechik. Sound retrieval and ranking using sparse auditory representations. *Neural Computation*, 22(9), Sept. 2010.
- [10] Minh Hoai Nguyen, Zhen-Zhong Lan, and Fernando De la Torre. Joint segmentation and classification of human actions in video. In *CVPR*, pages 3265–3272. IEEE, 2011.
- [11] Paul Over, George Awad, Martial Michel, Jonathan Fiscus, Greg Sanders, Wessel Kraaij, Alan F. Smeaton, and Georges Quenot. Trecvid 2013 – an overview of the goals, tasks, data, evaluation mechanisms and metrics. In *Proceedings of TRECVID 2013*. NIST, USA, 2013.

- [12] Genevieve Patterson and James Hays. Sun attribute database: Discovering, annotating, and recognizing scene attributes. In *Computer Vision and Pattern Recognition* (*CVPR*), 2012 IEEE Conference on, pages 2751–2758. IEEE, 2012.
- [13] Florent Perronnin, Jorge Sánchez, and Thomas Mensink. Improving the fisher kernel for large-scale image classification. In *Proceedings of the 11th European conference* on Computer vision: Part IV, ECCV'10, pages 143–156, Berlin, Heidelberg, 2010. Springer-Verlag.
- [14] Alan F. Smeaton, Paul Over, and Wessel Kraaij. High level feature detection from video in trecvid: a 5-year retrospective of achievements. *Multimedia Content Analysis*, pages 151–174, 2009.
- [15] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human action classes from videos in the wild. *CRCV-TR-12-01*, November.
- [16] Abdenour Hadid Timo Ahonen and Matti Pietikainen. Face recognition with local binary patterns. In *European Conference on Computer Vision (ECCV)*, pages 469–481, 2004.
- [17] Andrea Vedaldi and Andrew Zisserman. Efficient additive kernels via explicit feature maps. *IEEE Trans. Pattern Anal. Mach. Intell.*, 34(3):480–492, 2012.
- [18] Heng Wang, Alexander Kläser, Cordelia Schmid, and Cheng-Lin Liu. Dense trajectories and motion boundary descriptors for action recognition. *International Journal* of Computer Vision, 2013.
- [19] R. Yan, M. Fleury, M. Merler, A. Natsev, and J. R. Smith. Large-scale multimedia semantic concept modeling using robust subspace bagging and mapreduce. In ACM Multimedia Workshop on Large-Scale Multimedia Retrieval and Mining (LS-MMRM), Oct. 2009.
- [20] F. X. Yu, D. Liu, Sanjiv K., T. Jebara, and S.-F. Chang. ∝SVM for learning with label proportions. In *Proceedings* of the 30th International Conference on Machine Learning, 2013.