VIREO/ECNU @ TRECVID 2013: A Video Dance of Detection, Recounting and Search with Motion Relativity and Concept Learning from Wild

Chong-Wah Ngo†, Feng Wang‡, Wei Zhang†, Chun-Chet Tan†, Zhanhu Sun‡,
Shi-Ai Zhu†, Ting Yao†
† *Video Retrieval Group (VIREO), City University of Hong Kong*
‡ *School of Information Science and Technology, East China Normal University*
http://vireo.cs.cityu.edu.hk

Abstract

The VIREO group participated in four tasks: *instance search*, *multimedia event recounting*, *multimedia event detection*, and *semantic indexing*. In this paper, we will present our approaches and discuss the evaluation results.

Instance Search (INS):

We submitted four runs in total, experimenting three search paradigms for particular objects retrieval: (1) an elastic spatial consistency checking method; (2) a background context weighting strategy; and (3) a re-ranking step based on objects mining. The first two approaches are similar as last year [1], while the last one is our new exploration. Our submissions are all based on BoW model and tailored for the INS task. In particular, we use Delaunay Triangulation (DT) to address the complex spatial transformations for non-planar and non-rigid queries; the lack of information for small query objects is tackled with context modeling; and object mining augments the results by exploring frequent instances in TV series.

- F_X_NO_vireo_dt_2: BoW method + elastic spatial checking via DT. This run corresponds to our paradigm (1), which models elastic spatial structures as deformable graphs.
- F_X_NO_vireo_dtc_1: vireo_dt + context modeling. This run corresponds to our paradigm (2) by weighting the importance of different features in the query.
- F_X_NO_vireo_dtm_4: The mining result is fused with the results by vireo_dt via random walk (paradigms (1) + (3)). Links established by our mining algorithm serves as the cues for re-ranking.
- F_X_NO_vireo_dtcm_3: The mining result is fused with vireo_dtc through random walk (paradigms (2) + (3)). This run uses the ranking list from vireo_dtc.

Multimedia Event Detection (MED):

In this year's MED task, we submitted two runs to evaluate our visual and full systems respectively.

- FullSys_PROGAll_PS_100Ex_1: Detectors trained by combining visual and audio features.
- VisualSys_PROGAll_PS_100Ex_1: Visual features including SIFT, ColorSIFT, Motion relativity, and STIP are used for event detection.

Multimedia Event Recounting (MER):

We submitted the recounting for the positive videos based on the evidences from the audio-visual concepts. The visual evidences are built upon a graphical network and recounting is generated by exploiting the network's ontology. In particular, we implemented object/scene, action and non-speech audio detectors for evidence collection. Besides that, important keywords are mined from the ASR and OCR output as the supplementary evidences.

Semantic Indexing (SIN):

This year, we focused on a new feature representation extracted using deep neural networks (DNN). In the semantic indexing system, we adopted DNN feature, local and global features to train SVM models for each concept. Then we evaluated the contributions of different features using several fusion strategies. In addition, we submitted two runs for "no annotation" using Web images crawled from Flickr as training examples. These two runs are based on the model developed in [2]. In total, we submitted five runs as summarized below:

- 13_M_A_vireo.Baseline+DNN_1: Fusing the detection scores of classifiers using two global features, three local features and the new DNN feature.
- 13_M_A_vireo.DNN_2: Concept detectors are learned using DNN feature.
- 13_M_A_vireo.Baseline_3: Same with the baseline of our TRECVID 2012 systems, where global and local features are used.
- 13_M_F_vireo.SP_4: Concept detectors are learned on the training set sampled from Web images using Semantic Pooling (SP) method [2]. Both local and global visual features are used.
- 13_M_F_vireo.SP_KW_5: Training set is same with the run "13_M_F_vireo.SP_4", but only local features are employed.

1 Instance Search

This year, the dataset is composed with 243 episodes of TV series from "BBC EastEnders", where we extracted a total number of 640k keyframes (FPS = 1/4) from the 470k shots. SIFT features [3] and BoW model is adopted for all runs with a 250k visual vocabulary. Hamming Embedding [4], Multiple Assignments [5] are used to further enhance the model.

Similar to last year, we use DT and Context Modeling to address the problem of small query objects. Since this year's dataset is composed with TV series, which has strong dependency between shots and episodes. We explore this relationship as a cue to further enhance our searching system.

1.1 Methods

1.1.1 Elastic spatial checking via Delaunay Triangulation (DT)

For the run "DT", we used an elastic spatial topology checking technique based on Delaunay Triangulation (DT) [6, 7]. Most of previous spatial checking methods rely on a strict linear transformation. While in the query topics of INS, there exist lots of non-planar structures (e.g., 3D objects) and non-rigid objects (e.g., persons), which do not follow the commonly used Planar Homography or Epipolar assumption. In this case, we turned to elastically model the spatial topology with DT. The matched feature points on each image are first triangulated to approximate the spatial proximity using a mesh graph. Then the consistency of topological layouts is measured by the similarity of the graphs accordingly. This method emphasizes the topological consistency rather than a strict linear transformation, and the graph encodes the topology for matched points. This gives better tolerance to true responses in INS by accumulating evidences from local regularities of the instance.

1.1.2 Background context modeling by "stare"

We built the run "DTC" on "DT" by adding a practical background context modeling method [6], which simulates the "stare" behavior of human eyes. Instances often occupy a small area on the query image, and the background context is often different. Generally, the ROI is important, and the information outside the ROI may enrich the limited information and provide more clues for the instance. In our method, background context is modeled into the query with the "stare" model by weight contributions from the ROI and background.

1.1.3 Re-ranking based on Instance Mining

The rest of our runs "DTM" and "DTCM" were built upon "DT" and "DTC", respectively. The only difference is introduced by adding a re-ranking step based on a frequent objects mining method. Since this years' INS dataset is a TV series, there are numerous frequent "instances" (locations, objects, characters) appearing from time to time in different shots. We first design an object mining algorithm to mine frequent occurring objects from the dataset offline, then the mined links are used to re-rank the results by "DT" and "DTC", using a standard random walk. This run is by far the first try to tackle the search problem with the knowledge, specifically common visual instances, obtained through data mining.

Our re-ranking operates on the search results directly. Each node in the random walk process corresponds to a shot in the ranking list (up to $n = 1000$ shots). The transition matrix $\mathbf{P} = [p_{ij}]_{n\times n} = [p(j|i)]_{n\times n}$ encodes the transition probability between all pairs of shots. Let $\mathbf{x}_{(k)}$ be the state probability at time k. We follow the iteration method and solve the stationary probability $\hat{\mathbf{x}}$ as k goes to infinity. The initial state vector $\mathbf{x}_{(0)}$ is given as the scores of the top-1000 shots produced by DTM/DTCM. The rest of this section will focus on the construction of $\mathbf{P}$, which is given by the following algorithm:

1. Threads Extraction. Our mining algorithm adopts a bottom-up approach, building up visual instances from the elementary components: Thread of Features (ToF). A ToF corresponds to a set of consistent local patches across multiple shots. It only keeps reliable links among images sharing consistent local patches, and discards most of the unstable patches. To extract ToFs, we first organize the dataset as inverted file, and then extract threads from the posting list of each visual word. To speed up processing, a small binary code (Hamming signature) is attached to each local feature, and only features with similar binary codes are compared.

2. ToF Hashing. ToFs that links similar set of shots are extracted using Min-Hash. Multiple hash tables are used and the collisions in each table correspond to correlated ToF clusters. Note the number of ToFs found in each collision indicates the size of the instance. Larger instances (e.g., locations) lead to larger collisions, and vice versa. In our experiment, we discard both very small collisions (mostly noises) and very large collisions (mostly near duplicates). Only median size collisions are collected for further processing.

3. Seed Generation. For each cluster of ToFs, all the linked images are considered as potential holders of an instance. To further reduce noises, only the candidate images appearing in 80% ToFs are kept. We define a *seed* as a pair of images sharing the same pattern, and extract all image pairs in the cluster as the seeds.

As a result, our mining algorithm ends up with a pool of seeds $\mathcal{S}$. Each entry p_{ij} in $\mathbf{P}$ is given as the number of seeds for shot $(i,\, j)$ in $\mathcal{S}$. Finally, each row of $\mathbf{P}$ is l_1-normalized to fit as a Markov matrix,

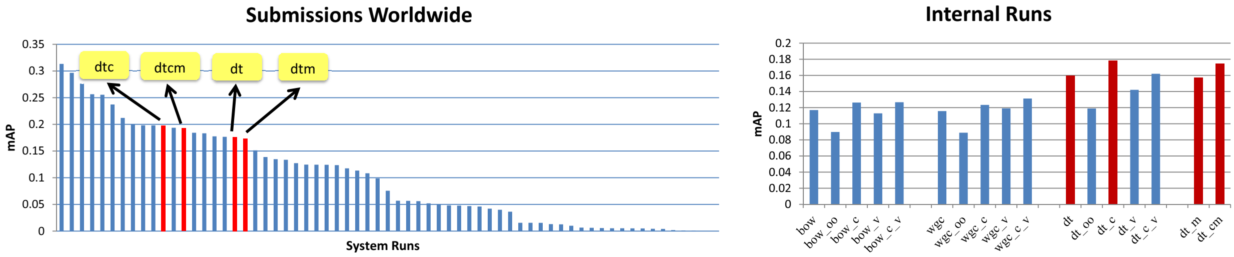


Figure 1: Left: Mean Average Precision of all 65 INS automatic runs in TRECVID 2013. Right: Internal runs for INS 2013. Naming convention: wgc (weak geometric consistency), dt (delaunay triangulation), oo (retrieve with ROI only), c (context modeling with "stare"), v (video level fusion), m (re-ranking based on mining results).

whose largest eigenvalue equals to 1. Standard random walk iteration is applied to get the stationary probability, which is then converted to the final ranking list.

1.2 INS Result Analysis

Figure 1 (left) shows the performance of all submission runs for this year's INS task, where our runs are marked with red color. Despite our low sampling rate (640k/470k keyframes per shot), our system is still comparable to most submissions. The best run was given by DT plus Context Modeling.

The whole picture is detailed in Figure 1 (right), where additional results are given by combining different techniques. A raw BoW method (bow) only has a mAP of 0.11, while our elastic topology checking (dt) brings the performance to 0.16. DT is beneficial for ranking properly by modeling various deformations introduced by different viewpoints for non-planar/non-rigid objects/scenes (e.g., 9097: checkerboard spheres, 9070: small red obelisk, 9086: these scales, 9089: this pendant). It is worth noting that: though some of the queries themselves (e.g., 9089: this pendant, 9079: this CD stand) may not be non-planar or non-rigid, regions outside the ROI is often non-planar or non-rigid (e.g., the women wearing the pendant; the moving crowds in front of the CD stand). In contrast, WGC does not improve the performance this year, since it works best for near-duplicates with large number of clean and consistent orientations/scales. Most of this year's queries are small in size, and large portion of background confuses the dominant transformation. Another reason is that the estimation of orientation/scale is sensitive against viewpoint change and non-rigid motion.

As observed in Figure 1 (right), background context did contain useful information, since the method with object alone (runs with "oo") always gives worse result. Searching with context modeling (runs with "c") weights features properly and always gave better mAP than runs using full image. The "stare" seems to be a reasonable model for INS. Video level fusion (runs with "v": see [1] for detail) does not work as expected. This might due to our sparse keyframe sampling for each shot.

Unfortunately, re-ranking with mining (runs with "m") only improves a few topics, and does not improve the overall performance. As observed, most of the results are near duplicate images, which introduce little new cue to a mature image search system. These links does not affect the ranking list too much, since most of the near duplicates have already been retrieved. Figure 2 shows some less-near-duplicate clusters mined from the dataset. Although our system is capable of extracting numerous frequent instances (e.g., paintings on the wall, clothes, logos, boards) from the TV archive, only a small portion is related with the 25 topics under evaluation, and is potentially useful to instance search system. Other instances mined from the dataset might show negative impact, since the re-ranking is based on links

Figure 2: Examples of the mining result.

other than the querying object. Our mining system only improves quite a few topics, which are directly linked by the instance itself or by their highly correlated background context. Overall, the improvement is unstable and more studies (e.g., considering the relationship of mined instances and the query) are needed to avoid negative linking for re-ranking. Although our first try does not work as expected, it did show some merits and could improve the searching system when used properly.

2 Multimedia Event Detection

2.1 Visual System (Visual_PROGAll_PS_100Ex_1)

In this run, we experimented our approaches with only visual features including static features (SIFT, ColorSIFT) extracted from keyframes, motion and spatio-temporal features extracted along frame sequences.

For static features, one keyframe is sampled in every 5 seconds along each video sequence. Features are then extracted in sampled frames. Local interest points are detected using DoG (Difference of Gaussian) [3] and Hessian Affine [8] detectors. 128-dimension SIFT descriptors [3] are employed to describe the local image patches. ColorSIFT [9] is further used to employ the missing color information in SIFT. For SIFT and ColorSIFT, Bag-of-Words approach is employed in feature representation. The feature spaces are first quantized into 2000 and 4000 words respectively. Soft weighting is used for word assignment and each descriptor is mapped to the 3 nearest words.

For motion features, we employ motion relativity proposed in our previous work [10]. A video is first segmented into a number of 5-second video clips. Keypoints are detected and tracked along the video sequence. A histogram is then computed by accumulating the relative motion between each pair of visual words in a given video volume. This results in a sequence of histograms to capture the relative motion information between different objects/scenes in the video. EMD (Earth Mover's Distance) is used to measure the similarity between two videos and integrated into SVM kernel [10] for classification.

Compared with the original approach in [10], to avoid the false alarms in keypoint tracking by KLT algorithm, we remove the trajectories between two frames which are more than 2 times longer than the previous and next ones. Given a trajectory $t = (q_1, q_2, \cdots, q_n)$ where q_i is the location of the keypoint at i-th frame, the motion between q_i and q_{i+1} is ignored if $dist(q_i, q_{i+1}) > 2 * dist(q_{i-1}, q_i)$ and

$dist(q_i, q_{i+1}) > 2 * dist(q_{i+1}, q_{i+2})$ where $dist(.)$ calculates the motion between two neighboring frames. In our experiments on development set, this improves the MAP by about 6%.

For spatio-temporal features, STIP are extracted. Laptev's algorithm is adopted. It captures a space-time volume in which video pixel values have large variations in both space and time. Histogram of Oriented Gradients (HOG; 72 dimensions) and Histogram of Optical Flow (HOF; 90 dimensions) are computed as the descriptors.

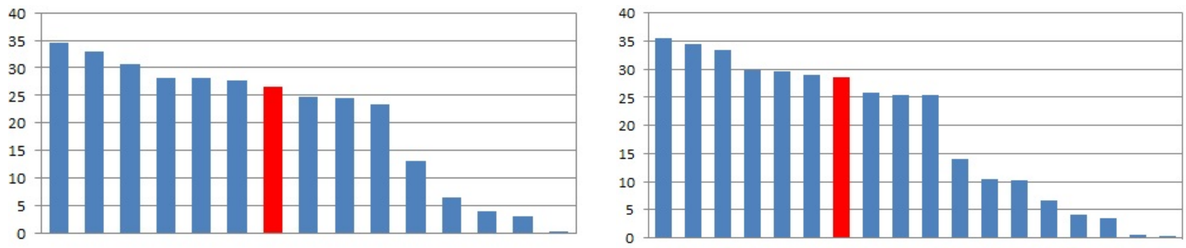
For classifier learning, LIBSVM [11] is employed. We consider two kinds of approaches: $\chi^2 - RBF$ kernel and EMD (Earth Mover's Distance) based temporal matching [12]. $\chi^2 - RBF$ kernel is employed for SIFT, ColorSIFT, and STIP, while temporal matching for motion relativity. The results of all classifiers are combined with linearly-weighted fusion.

2.2 Full System (Visual_PROGAll_PS_100Ex_1)

In the full system, audio features are fused with the visual system for event detection. MFCC coefficients are extracted in every audio frame of 50ms, where each frame overlaps with neighboring ones by 25ms. Bag-of-Words representation is used for MFCC feature and a vocabulary of 8000 words is constructed.

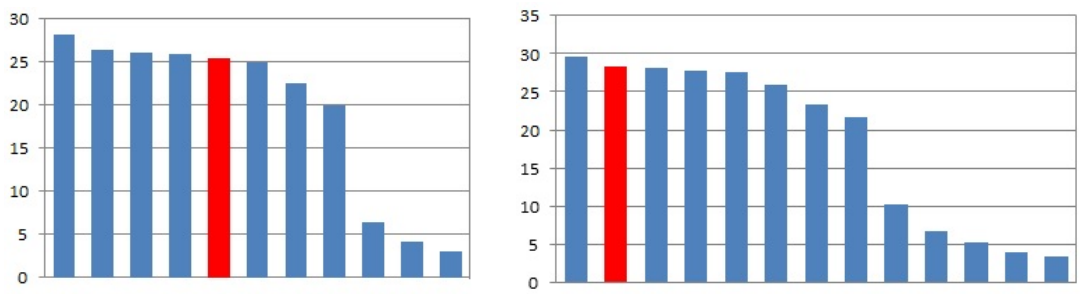
We found that MFCC is insensitive to a certain audio spectrum. This has inspired us to investigate the other audio features. Eventually, a combination of audio features is adopted, including line spectral frequency (LSF), octave band signal intensity (OBSI), linear predictor coefficients (LPC), MFCC and their first and second derivatives.

In our system, linear fusion is used to combine audio and visual features, where the weights are estimated on the development set.



(a) PROGAll dataset (b) PROGSub subset

Figure 3: Performance of our full system (red bars) in MED task among all submitted full systems.



(a) PROGAll dataset (b) PROGSub subset

Figure 4: Performance of our visual system (red bars) in MED task among all submitted visual systems.

2.3 Evaluation Results

Figure 3 and Figure 4 present the evaluation results of our full system and visual system in MED task respectively. Our performance is close to the median of all submitted systems, and the visual features contribute the most to the full system. Compared to the visual system, the MAP (Mean Average Precision) is improved by around 4% after combining audio features. For the detection of some events such as *Birthday_party* and *Marriage_Proposal*, OCR and ASR are good hints, and should be investigated and incorporated into our system. Furthermore, the fusion of different features should be carefully studied to effectively combine different modalities for event detection.

3 Multimedia Event Recounting

The objective of this task is for every clip that the MED system deems to be positive for an event, textual descriptions have to be generated to recount the important evidence. Ideally, the recounted evidence should be the evidence used to detect the event in the MED system. We consider the problem by constructing an event network to represent each of the event. There are two-fold reason behind. First of all, it gives an ontology representation of the relevant concepts and the pertaining relationships in an event. More importantly, it allows us to exploit the relationships for creating more comprehensive sentence structure. In the following we explain how we generate the event network in the evidence collection and describe the process of how the recounting sentences are constructed.

3.1 Evidence Collection

The evidence of an event is mainly composed of those visual concepts and non-speech audio concepts. Besides of that, transcripts from both automatic speech recognition (ASR) and optical character recognition (OCR) are also important. We first focus on the evidence collection based on the visual and audio concepts, later we elaborate the details of ASR and OCR implementations.

3.1.1 Visual Concepts from ConceptNet

Each event kit explicitly defines an event. Given these event kits, we extract the important keyphrases from the event kits using text mining technique. Based on the keyphrases, we infer the relevant concepts of an event from the ConceptNet [13]. ConceptNet is a semantic network generated automatically from the 700,000 sentences. The knowledge of ConceptNet encompasses the spatial, physical, social, temporal and psychological aspects of everyday life. It is encoded with common sense knowledge by linking two concepts with their appropriate relationship. It focuses on diverse relational ontology, and its emphasis on conceptual relationship allows us to make practical and context-oriented concept inference over an event. The inference is similar to that of principal component analysis (PCA). More relevant concepts pertaining to the event are acquired.

Based on all these obtained concepts, a subgraph, namely event network, is extracted from the huge structure of the ConceptNet to represent an event. To build a single graph without any isolated vertex, we run the shortest path algorithm to link those isolated vertices or subgraphs with intermediate vertices. Normally the concepts are not far apart as the concepts are having related meaning to an event. Post-processing is carried out to get rid of the abstract concepts and the subgraph is then packed to fewer concepts by removing most of the noisy concepts with less connections at the boundary. After that, the remaining concepts are categorized into a few categories, namely *none*, *object*, *scene*, *event* and *action*.

The process is done automatically based on the reasoning using the relationships of the edges in the event network and the word type from the WordNet. Since the event network is of multiple directed edges, we consider the category of a concept in a hierarchy nature, in which *object* and *action* are at the lowest and highest ranks respectively. For example, the concept *kitchen* is first assigned to the category of *object* when the object relationship, such as *UsedFor*, is observed. However, it is *promoted* to *scene* when the corresponding relationship, such as *AtLocation*, is observed. There are 26 types of relationships in the ConceptNet and we use most of them for this purpose. The concept will never be *demoted*. Thus there are very few action concepts. WordNet is used for checking the word type of a concept in assistance with the relationship for category classification. This process is crucial as it helps in building the sentence structure in the latter part.

For the *object* and *scene* concepts, we crawl Flickr images for training the concept detectors, while for *action* concepts, we use the training data, annotated with action labels for every 5-second interval, for the training of the concept detectors. SIFT is used in the object and scene detectors while motion relativity is employed [14] in the action detectors. There 122 object and scene concepts, and 8 action concepts determined in our system. All of the detectors are learned using chi-square SVMs. During testing, keyframes are extracted from test videos and the object and scene detectors are used to detect the existence of the objects and scenes in the keyframes. For action detection, we split the test videos into 5-second intervals and examine the occurrence of the particular action using action detectors.

3.1.2 Non-speech Audio Concepts

In additional to visual concepts, 14 non-speech audio concepts, including *cheering*, *clapping*, *hammering*, etc. are determined manually in our system. The audio concepts are important in certain events. For instance, it is easier to detect the *hammering* sound than detecting the hammer itself in the event of *working on a metal crafts project.* We found that MFCC is not sensible to certain audio spectrum, and thus we train the audio detectors using a combination of audio features, namely line spectral frequency (LSF), octave band signal intensity (OBSI), linear predictor coefficients (LPC), MFCC and their first and second derivatives. The training data, annotated similar to that of the action concepts, is used for training the audio concepts. Identical training and test methods are adopted for the audio detectors, similar to that of the action detectors.

3.1.3 Automatic Speech Recognition and Optical Character Recognition

Apart from the audio-visual concepts, important keywords are extracted from the speeches and optical characters as the evidence for an event. In this work, we use the Sphinx-III [15] speech recognition engine to transcribe the videos. Instead of outputting the transcripts, we mine the transcripts for important keywords that are related to an event. It could be difficult to detect the small objects in the event of "making a sandwich" and "working on a sewing project", but it would be easy if the keywords, such as *sandwich* and *fabric*, are detected in the speeches. The keywords being mined are the concepts determined in the event networks. Along the lines of that ASR analysis, the texts appear in the video keyframes are also a useful information for event recounting, we use the tesseract-ocr [16] engine to recognize the texts. The same method is applied for mining the important keywords similar to that of ASR.

3.2 Sentence Generation

Since the action and audio concepts are detected every five seconds, we recount the test videos in this basis. Although the detections of object and scene are performed every second on the keyframes, we

Table 1: Performance summary.

	MER-FullSys		
	Accuracy	ObsTextScore	PRRT
Mean	59.58%	1.60	80.12%
VIREO	36.91%	2.06	22.93%

consider the appearance of the object and scenes in a five-second basis too. In other words, we recount the evidence of all the concepts every five seconds. We form the sentence using a parser tree by setting the detected concepts to the appropriate part of speech (POS) tags. Thus different combination of sentences can be formed using the POS tags. The generation of a sentence is action centric. It means the sentence is built upon an action. If an action is detected, we first change the action (verb) to gerund. Next, determine whether an object concept is associated with the action from the event network. For example, climbing *mountain*. Afterward, check if there is any scene concept in the event network. If there is, set the scene concept to proper POS tag, so that it will appear at the end of the verb phrase. If there is a gerund, we set the subject to "Someone is", else the subject is set to "There is" or "The background is" accordingly. Eventually, the sentence is formed using the concepts available according to the POS tags. For example, with the relationship of *AtLocation* in the edge between the concepts of *plate* and *kitchen*, sentence "There is a plate in the kitchen." is generated if both the concepts are detected. The sentence generated is still far from perfect. For example, in this sentence "Someone is climbing in the rock wall.", the preposition is inappropriate and should be void in this case.

Although the sentence generation is action centric and not all of the event networks have an action concept, it does not affect the recounting performance. In fact, the action concepts in ConceptNet are basic actions, i.e. *run*, *jump*, *climb* etc. These basic actions are easier to be detected by motion classifiers. Those complex actions, such as *making sandwiches*, which composed of multiple different basic actions are difficult to be defined and detected.

For non-speech audio, we manually create a short sentence, like "Engine sound is heard.", "Someone is laughing.", etc. to represent each of the audio concept detected. For the important keywords mined in the ASR and OCR, snippets of the corresponding keywords in speech and text are extracted and shown as the evidence.

The evidence is presented according to the detection scores. The audio-visual evidence is presented before the ASR and OCR evidence as the performance of the audio-visual evidence is observed to perform better in our case. The concept detection scores are normalized to cumulative distribution function values for fair comparison. If there are multiple recountings with the same descriptions, the latter ones will be skipped if there are more than 10 unique evidences to be presented. This will preserve the variety of evidence to be presented at the top ranked list, especially in the lengthy clips.

3.3 MER Results and Analysis

For a test clip deemed as a positive video by the MED system, evidence collection is carried out. The recounting output consisting of the evidence describing the object, scene, action and audio concepts of the particular event. Note, this constraint causes the false positive videos to have no observation in general, in turn jeopardizes the accuracy in the performance evaluation, as depicted in Table 1.

Tables 2 and 3 show the numbers of video detected to contain such a concept corresponding to an events for the top 200 test videos in each event. From the tables, it is noticed that there are more right action concepts happen in the right events. There are also fewer false positive concepts happen in other

Table 2: Performance of the action detectors. The values show the numbers of video detected to contain such a particular concept corresponding to an event. The highlighted cells represent the right concepts in the right events.

	E006	E007	E008	E009	E010	E011	E012	E013	E014	E015	E021	E022	E023	E024	E025	E026	E027	E028	E029	E030
bike trick	0	3	0	1	3	4	2	4	3	4	201	3	2	6	7	9	3	7	5	2
climb	1	4	6	11	6	6	5	11	19	18	10	4	16	9	5	7	135	1	11	12
dance	0	3	161	2	2	0	7	1	2	1	2	1	2	2	1	0	1	2	7	1
jump	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1
march	5	6	4	5	2	3	126	4	6	1	5	5	9	5	11	5	4	10	2	6
run	1	5	2	8	3	1	6	7	8	2	7	4	10	10	3	4	4	6	141	4
somersault	0	0	0	0	0	0	0	14	0	0	0	0	1	0	0	1	0	1	0	0
swim	0	0	1	3	3	0	1	4	6	2	3	3	2	8	4	4	2	0	20	1

Table 3: Performance of the audio detectors. The values show the numbers of video detected to contain such a particular concept corresponding to an event. The highlighted cells represent the right concepts in the right events.

	E006	E007	E008	E009	E010	E011	E012	E013	E014	E015	E021	E022	E023	E024	E025	E026	E027	E028	E029	E030
band	5	15	24	16	6	7	30	6	5	7	19	8	4	12	10	10	11	19	14	18
birthday song	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
cheering	24	12	76	9	16	9	52	21	13	7	23	21	39	10	23	10	24	46	75	22
clapping	17	5	36	7	9	6	38	9	7	2	13	18	26	4	15	12	5	57	43	17
dog barking	36	16	53	23	27	17	43	22	28	15	27	44	51	19	19	16	17	58	51	35
engine sound	0	2	10	10	3	1	5	1	2	2	3	3	3	4	1	3	2	4	4	3
hammering	11	31	22	19	47	37	25	19	46	26	34	51	19	18	9	28	32	21	24	66
laughing	16	15	30	12	16	6	7	8	11	8	17	23	14	14	10	9	8	28	25	20
power tool	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
sewing machine	0	0	2	0	0	0	1	0	0	1	0	4	2	0	0	1	0	1	0	2
talking via mic	6	10	6	7	13	15	5	1	44	16	8	16	6	14	7	8	10	38	3	20
tool sound	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 4: Performance of the object detectors. The values show the numbers of video detected to contain such a particular concept corresponding to an event. The highlighted cells represent the right concepts in the right events.

	E006	E007	E008	E009	E010	E011	E012	E013	E014	E015	E021	E022	E023	E024	E025	E026	E027	E028	E029	E030
automobile	0	15	3	8	5	8	5	8	11	3	13	6	6	8	7	3	0	2	6	7
bicycle	0	9	4	0	0	3	2	1	6	3	19	3	0	0	0	0	0	1	3	1
cake	33	58	29	37	45	81	41	20	81	73	37	44	38	30	26	33	62	16	26	74
car	0	1	0	1	0	1	1	1	0	1	0	0	0	0	0	0	0	0	0	2
crowd	18	27	114	8	16	11	87	25	19	20	31	7	49	37	39	11	47	47	32	21
dancer	6	16	50	5	5	4	45	4	8	14	14	2	22	13	17	4	21	5	9	9
dog	0	1	0	1	4	0	0	2	0	0	3	0	2	0	1	0	2	1	2	0
fabric	1	3	1	0	3	2	3	4	6	8	0	4	2	7	2	7	12	0	3	7
kitchen	9	20	21	3	11	43	23	23	25	22	9	44	8	19	17	46	10	24	9	18
mountain	7	36	26	52	46	46	20	64	56	36	90	34	35	60	26	24	75	20	49	52
parade	14	35	128	14	15	18	101	11	11	14	40	8	67	49	31	7	29	18	41	14
park	0	4	12	14	5	3	10	25	2	1	36	2	11	17	4	2	11	2	18	6

events for the action detectors. The action detectors basically perform better compared to the audio counterparts. Besides the discriminative power of the descriptors, it could be mainly contributed by the high dimensionality, namely 100,000, of visual words used in the action detectors compared to that, namely 4,000, of the audio detectors.

Due to the length limit, only partial analysis of object/scene detectors corresponding to events is depicted in Table 4. It can be seen that the object/scene detectors perform badly in overall. There are a few factors behind. It is observed that the main problem in detecting the object/scene is caused by the difference of scale and the position/location of the object. The classifiers for audio concepts do not suffer from this spatial problem. While the classifiers of the motion concepts are trained using the videos from the training data set, the scales of the objects corresponding to a motion in the training and test videos are of the similar ratios. Most of the successful cases of the object/scene detection happen in the scene concepts, e.g. "crowd", "parade", "kitchen" etc. in which the visual appearance is of the same pattern over the whole image. Normally the objects we try to detect are not occupying the whole image. As the SIFT features are quantized into visual words and pooled into a vector to represent an image, the discriminative power for detecting an object is *diluted* by the background features, and this is the major drawback of our system.

As mentioned earlier, the sentences for recounting are generated automatically using our parser tree mechanism based on the concepts detected in the event network. The relationship of the edges are used in the recounting process, a more complicated sentence can be generated as the result. A good performance is observed in the score obtained in the evaluation of precision of the observation text, namely 2.06 compared to the mean score 1.60, as shown in Table 1. Although it is difficult to conclude the recounting mechanism performs well, in some way it shows the effectiveness of the automatic recounting mechanism.

In the other evaluation of percent recounting review time (PRRT), the score of our result (22.93%) is much lower than the mean (80.12%). It is due to the limited number of evidence to be evaluated as only the related concepts in an event network are recounted, in additional to the non-speech audio, ASR and OCR evidence. Furthermore, the false positive videos without any observation speed up the evaluation process too. However, low value of PRRT, on the other hand, could also mean that our system is precise in recounting the most relevant concepts for an event, and thus save up the time of evaluation.

4 Semantic Indexing

In previous years, we have provided valuable tries to improve the semantic concept detection in video domain by using free-sampled image or video data. However, due to the large domain gap, the Web data seems less effective when sufficient training instances are available in TRECVID. In addition, according to our recent findings [17, 18], the performance is affected by other two factors: 1) concept category and 2)degree of data distribution mismatch. This year, there are no new training instances, but the evaluated concepts are different from previous year. Thus, to further confirm our observations, we reuse the models learned with Web images in our TRECVID 2012 system. Furthermore, our focus will be on evaluating new visual representations. In specific, we try to employ the visual features extracted from deep neural networks (DNN), which has been successfully applied in image classification.

4.1 Visual Features

Same with our TRECVID 2012 system [1], we considered Bag-of-visual-words (BoW) representation derived from local keypoint features, and two global features grid-based color moments (CM) and grid-based wavelet texture (WT). Specifically, SIFT feature are computed for each local keypoint which is detected using DoG and Hessian Affine. In addition, spatial information is considered by using 2×2 and 3×1 partitions.

Recently, deep neural networks (DNN) has demonstrated its effectiveness in learning image representation and classifier simultaneously with a large number of training instances. The learnt image representation from DNN is close to semantics, and ever exceeds current estimate of Inferior Temporal (IT) representation performance in macaque's visual cortex [19]. Inspired by the success of DNN, we use it to generate visual representation as another visual feature in our system, which is a 1024-dimensional feature vector. Similar to [20], the used DNN architecture is denoted as $Image - C64 - P - N - C128 - P - N - C192 - C192 - C128 - P - F4096 - F1024 - F1000$, which contains five convolutional layers (denoted by C following the number of filters) while the last three are fully-connected layers (denoted by F following the number of neurons); the max-pooling layers (denoted by P) follow the first, second and fifth convolutional layers; local contrast normalization layers (denoted by N) follow the first and second max-pooling layers. The weights of DNN are learnt on ILSVRC-2010, which is a subset of ImageNet dataset with 1.26 million training images from 1,000 categories. For each keyframe, its representation is the neuronal responses of the layer $F1024$ by input the keyframe into the learnt DNN.

As a result, we extract six kinds of visual features which are further used for learning SVM models respectively. Given a testing keyframe, the SVM classifiers are applied on the corresponding feature representation and the raw outputs of SVM are converted to posterior probabilities.

4.2 Feature Fusion

The extracted visual features represent the instances from different view points, such as color, textual and semantics. The complementary nature of multiple features is likely to improve the performance even further. Thus we evaluate the usefulness of features using late fusion to combine the posterior probabilities of classifiers learnt with different features. Except the classifiers using DNN feature, other models are learnt in our TRECVID 2012 systems. The evaluated fusion strategies are summarized as follows:

- Concept detectors are learned on TRECVID training set.

  * *Baseline*: Late fusion using the two global features, three local features.

  * *DNN*: Using DNN feature only.
  * *Baseline+DNN*: Late fusion using all six features.

- Classifiers are learned on Web images sampled using Semantic Pooling approach [2].

  * *SP*: Both local and global visual features are used.
  * *SP_KW*: Only local features are employed.

4.3 SIN Results and Analysis

Figure 5 shows the mean average precision (MAP) performance of all 98 full version submitted system runs where our five runs are marked in red. Similar to the observations in our TRECVID 2012 system, there is a larger performance gap between the classifiers learnt using TRECVID training data and free-sampled Web images. We can see that *SP* performs similar to *SP_KW*. This may indicate that the gap is less likely to be narrowed by using the features which may be not able to model the common visual aspects of instances from two domains. For the new visual representation, we observe that *Baseline+DNN* with $MAP = 0.154$ improves the *Baseline* with $MAP = 0.127$ by 21.2%. This verifies the effectiveness of DNN feature in semantic-level similarity measurement. This can also be observed in Figure 6, which further details the average precision (AP) of our five submissions. For some concepts such as "Hand" and "Chair", the *DNN* even performs best. In addition, while overall result of *Baseline*, which adopts multiple visual features, is similar to that of *DNN*, the *DNN* is much more efficient as the dimension of feature vector is much less than that of *Baseline*. Both the computational costs in training and testing will be saved significantly.

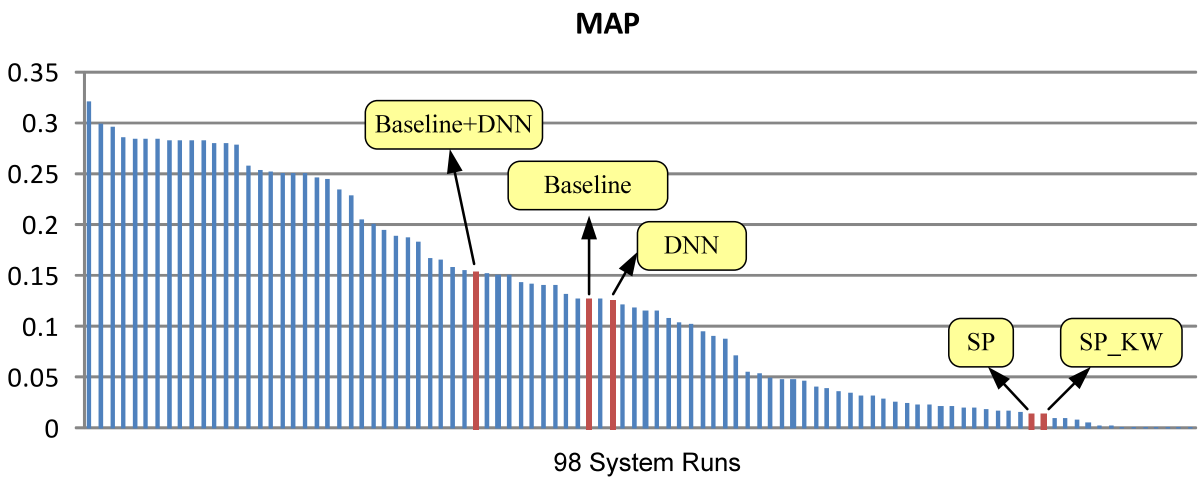


Figure 5: Mean average precision of all 68 SIN full version runs submitted to TRECVID 2011. Our submissions are marked in red.

5 Summary

For INS, we experimented three searching paradigms for object retrieval: elastic spatial topology checking (DT), background context modeling (stare), and object mining based re-ranking. We gain substantial performance improvement by using the first two techniques, especially DT. Our elastic topology matching has again been proven a suitable model for instance search, since it encodes the spatial topology rather than hard locations, and bypass the geometry estimation using noisy orientation/scale of local features. Context modeling gives a good tradeoff in finding near duplicates of the query and novel results with

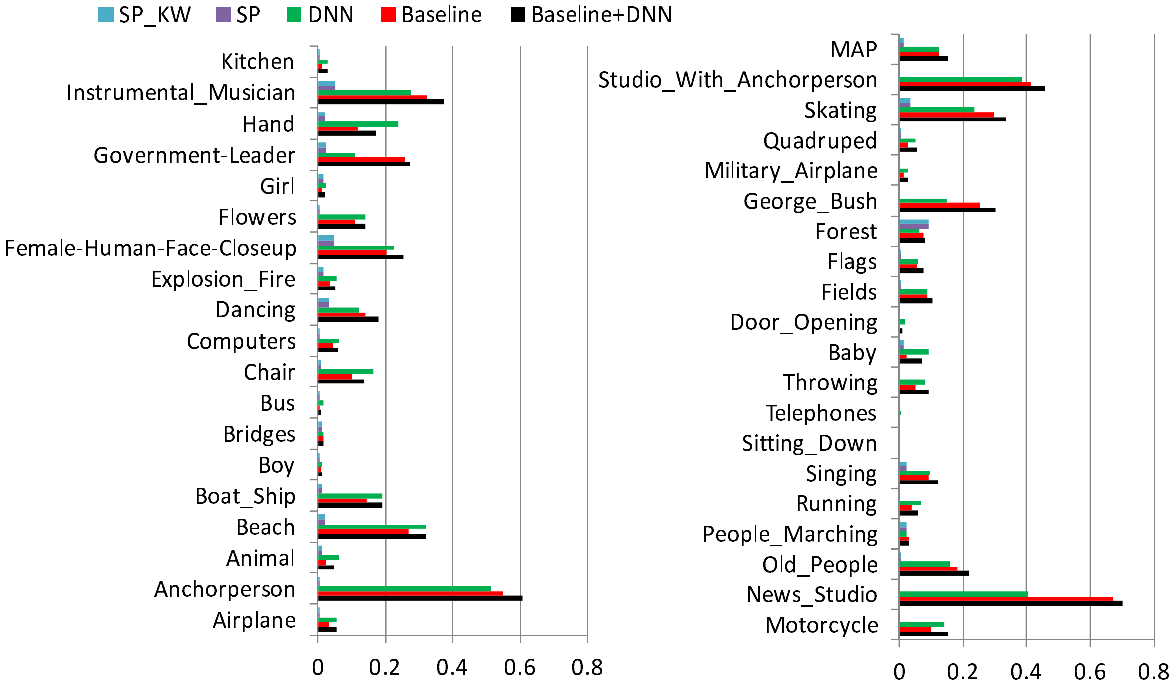


Figure 6: Per-concept performance of our submitted systems.

diverse backgrounds. Though our attempt of object mining hurts the overall performance, we did find some links involving the query objects. How to leverage instance mining for instance searching remains our future work.

For MED, we mainly focused on visual features especially motion information which is good at describing motion-intensive events. The low-level audio features can help detect some events and slightly improve the performance. Semantic information such as ASR and OCR can be employed in future work. Furthermore, the fusion methods of different modalities should be carefully studied.

For MER, basically our model is suffering from the accuracy of detection. The problem is in fact in lines with the lack of ability to locate the position of objects detected. Methods for object detection and localization play an inevitable role in multimedia recounting. Much efforts have been devoted to solve the problems by using different methods, among others but not limited to, template-based and graph-based matching methods. The deformable part models have proved to be efficient and have achieved state-of-the-art performance on benchmarks, such as the PASCAL dataset. It is of interest to study and implement the method in our framework to improve the detection accuracy. Besides that, we are also interested to implement the classification models for weakly labeled data. This is crucial for handling the training data from the Internet, which are with noise and without the bounding boxes of the objects or concepts we are looking for. Finally, more sophisticated forms of sentence should be explored by further exploiting the ontology structure of the event networks.

For SIN, we tried the feature extracted using a DNN trained on a large amount of instances for 1,000 concepts. Compared to the models learned in our TRECVID 2012 system, DNN, which is much more efficient, performs similar to the fusion result of local and global visual features. The performance is further improved by combining all the visual features. In addition, current features are still less effective for narrowing the domain gap. Further directions include a more distinctive visual representation for addressing the problem of domain gap, and incorporating features representing other aspects of visual instances into our system, such as attribute features.

Acknowledgment

The work described in this paper was fully supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (CityU 118812).

References

[1] W. Zhang, C.-C. Tan, S.-A. Zhu, T. Yao, L. Pang, and C.-W. Ngo, "Vireo @ trecvid 2012: Searching with topology, recounting will small concepts, learning with free examples," in *NIST TRECVID Workshop*, 2012.

[2] S. Zhu, C.-W. Ngo, and Y.-g. Jiang, "Sampling and ontologically pooling web images for visual concept learning," in *IEEE Trans. on Multimedia*, 2012.

[3] D. Lowe, "Distinctive image features from scale-invariant keypoints," *IJCV*, vol. 60, no. 2, pp. 91–110, 2004.

[4] H. Jégou, M. Douze, and C. Schmid, "Hamming embedding and weak geometric consistency for large scale image search," in *ECCV*, 2008.

[5] J. Philbin, M. Isard, J. Sivic, and A. Zisserman, "Lost in quantization: Improving particular object retrieval in large scale image databases," in *CVPR*, 2008.

[6] W. Zhang and C.-W. Ngo, "Searching visual instances with topology checking and context modeling," in *International conference on multimedia retrieval*, 2013, pp. 57–64.

[7] W. Zhang, L. Pang, and C.-W. Ngo, "Snap-and-ask: Answering multimodal question by naming visual instance," in *ACM international conference on Multimedia*, 2012.

[8] K. Mikolajczyk and C. Schmid, "Scale and affine invariant interest point detectors," *International Journal of Computer Vision*, vol. 60, no. 1, pp. 63–86, 2004.

[9] K. van de Sande, T. Gevers, and C. Snoek, "Evaluating color descriptors for object and scene recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 9, pp. 1582–1596, 2010.

[10] F. Wang, Y.-G. Jiang, and C.-W. Ngo, "Video event detection using motion relativity and visual relatedness," in *ACM international conference on Multimedia*, 2008, pp. 239–248.

[11] "Libsvm," http://www.csie.ntu.edu.tw/˜cjlin/libsvm/.

[12] D. Xu and S.-F. Chang, "Video event recognition using kernel methods with multilevel temporal alignment," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 11, pp. 1985–1997, 2008.

[13] H. Liu and P. Singh, "Conceptnet – a practical commonsense reasoning tool-kit," *BT Technology Journal*, vol. 22, no. 4, 2004.

[14] F. Wang, Y.-G. Jiang, and C.-W. Ngo, "Video event detection using motion relativity and visual relatedness," in *Proceedings of ACM International Conference on Multimedia*, 2008.

[15] S. C. Doh, K. Seymore, S. Chen, S. Doh, M. Eskenazi, E. Gouva, B. Raj, M. Ravishankar, R. Rosenfeld, M. Siegler, R. Stern, and E. Thayer, in *DARPA Speech Recognition Workshop*, 1998, pp. 55–59.

[16] A. Kay, "Tesseract: an open-source optical character recognition engine," *Linux J.*, 2007.

[17] T. Yao, C.-W. Ngo, and S. Zhu, "Predicting domain adaptivity: redo or recycle?" in *ACM international conference on Multimedia*, 2012.

[18] S. Zhu, T. Yao, and C.-W. Ngo, "Video concept detection by learning from web images: A case study on cross domain learning," in *ICME Workshop on Media Fragment Creation and reMIXing (MMIX'13)*, 2013.

[19] C. F. Cadieu, H. Hong, D. Yamins, N. Pinto, N. J. Majaj, and J. J. DiCarlo, "The neural representation benchmark and its evaluation on brain and machine," in *ICLR*, 2013.

[20] A. Krizhevsky, I. Sutskever, and G. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, 2012.