Fudan University at TRECVID 2006

Xiangyang Xue, Hong Lu, Hui Yu, Shile Zhang, Bin Li, Jing Zhang,
Jie Ma, Bolan Su，Yuefei Guo
Department of Computer Science and Engineering, Fudan University, Shanghai, China

In this notebook paper we describe our participation in the NIST TRECVID 2006 evaluation. We took part in two tasks of benchmark this year including high-level feature extraction and search (manual/interactive).

For high-level feature extraction, we submitted 4 runs.

FD_SVM_BN_1: using SVM and ontology.

FD_SCM_BN_2: using GMM and ontology.

FD_SVM_MTL_3: using SVM and multi-task learning.

FD_SCM_MTL_4: using GMM and multi-task learning.

Evaluation results illustrate that there are both advantages and disadvantages exist in all methods.

For search, we submitted 3 manual runs and 1 interactive run.

M_A_2_FD_M_TEXT_1: textual retrieval

M_A_2_FD_MM_BC_3: multi-model, relation expression, BC fusion.

M_A_2_FD_M_TRAIN_TEXT_2: textual retrieval, using key words trained by develop data.

I_A_2_FD_I_LR_4: interactive run using LR regression

Evaluation results illustrate that the method only using textual information is better than other runs. We also tried to select keywords by training develop data to get better performance.

1. Introduction

Content-based video retrieval is an interesting but challenging work. It draws more and more attention to developing effective techniques for analysis, indexing, and searching of video from the database. TRECVID provides a standard dataset and evaluation criterion for comparing different algorithms and systems. This year we proposed some new algorithms in each task we took part in, i.e., high-level feature extraction and search (manual/interactive).

2. High-level Feature Extraction

For the task of high-level feature extraction, we divide the work into two parts: First, salient object detection; Second, concept (corresponding to 39 high-level features) learning based on the results of salient object detectors. We propose two methods for each step respectively, and there are four combinations. So finally, we submit four runs for this task, as Figure.1 shows below:

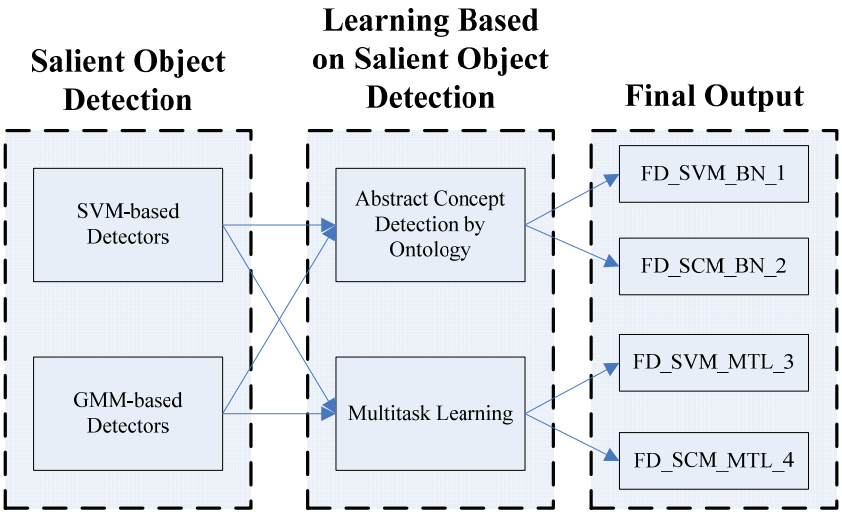


Figure.1 Overview on the framework of high-level feature extraction

2.1. Salient Object Detection

According to the task, we define 21 salient objects to detect, most of which come from the task, and the others come from the labeling work of LSCOM [1]. These salient objects are:

- Airplane
- Animal
- Boat_Ship
- Building
- Bus
- Car
- Charts
- Computer_TV-screen
- Desert
- Explosion_Fire
- Flag-US
- Flowers
- Hand
- Maps
- Mountain
- Road
- Sky
- Snow
- Truck
- Vegetation
- Waterscape_Waterfront

First, image segmentation is taken. We use JSEG [2] to segment the key frames of each shot into several regions. And then we label these regions manually. Given one salient object, if most of, or the whole of a region is part of, or contains this kind of salient object, we label this region as

positive.

After image segmentation, we can extract regional features. To lower the complexity of training so many detectors, we only extract 2 features to represent a region: one is average color and color variance on Lab color space, the other is Tamura [3] texture. The former feature is in 7-dimension, whereas the latter feature is in 15-dimension.

2.1.1. SVM-based detectors

On each salient object and each feature, we train a binary SVM classifier. The kernel we use is RBF, and grid search is taken to obtain the optimal parameter pair (C, ɣ). To merge the results from different classifiers trained on different features, an unsupervised method, Border Counter, is used. Each classifier trained on each feature is assigned the same weight.

2.1.2. GMM-based detectors

Besides the salient object detectors learned by using SVM, we also implement a series of detectors that are based on the Gaussian Mixture Models (GMM) and Bayesian theorem. We first incorporate the training samples of each salient objects (the same training set as in subsection 1.1.1) to estimate the mixture densities for 21 salient objects. Expectation Maximization (EM) algorithm [4] is applied to estimate the parameters of GMM, and we can obtain the optimal parameters for the *n*th salient object:

$$\Theta^{(n)} = \{(w_m^{(n)}, \vec{\mu}_m^{(n)}, \sigma_m^{(n)}), 1 \le m \le M\} \tag{1}$$

Where n denotes the nth task, M denotes the number of Gaussians in the mixture, while w_m, μ_m, σ_m denote the weight, the mean vector and the diagonal vector of covariance matrix of the mth Gaussian, respectively. We adopt Bayesian Information Criteria (BIC) [5] to determine the optimal number of Gaussians in the mixture.

When we have obtained the mixture models for 21 salient object detectors, we use Bayesian theorem to calculate the posterior probabilities of the nth mixture model upon a test sample X:

$$P(\Theta^{(n)} \mid X) = \frac{P(X \mid \Theta^{(n)}) P(\Theta^{(n)})}{P(X)} \tag{2}$$

We assume the prior of a sample to be a constant, and the prior of each model is proportional to the sample size of each class, then we get:

$$P(\Theta^{(n)} \mid X) = \frac{N^{(n)}}{N} P(X \mid \Theta^{(n)}) \tag{3}$$

Where N denotes the total sample size of all salient objects, and $N^{(n)}$ the sample size of the nth salient object. We use $P(\Theta^{(n)}|X)$ as the confidence of X belonging to nth class.

2.2. Learning Based on Salient Object Detection

2.2.1. Concept Detection by Ontology

We adopt ontology to complete concept detection. In our ontology, all the concepts need to be extracted are divided into two parts: concrete concepts (salient object) and concepts. The salient object can be detected by pattern classification which is suggested in 2.1.1 and 2.1.2. Then the concepts reasoning in terms of ontology and salient objects of video are applied to detect all the concepts.

The salient objects in our method are defined as: Airplane, Animal, Boat_ship, Building, Bus, Car, Charts, Computer_TV_screen, Explosion_fire, Desert, maps, Flag_US, Flowers, Hand, Mountain, Road, Sky, Snow, Truck, Vegetation, and Waterscape_waterfront. All the 39 concepts which are defined by TRECVID 2005 can be detected by ontology and the salient objects which have been detected by the methods in 2.1.1 and 2.1.2.

The two layered ontology is constructed by training data which have been annotated by all the 39 concepts.

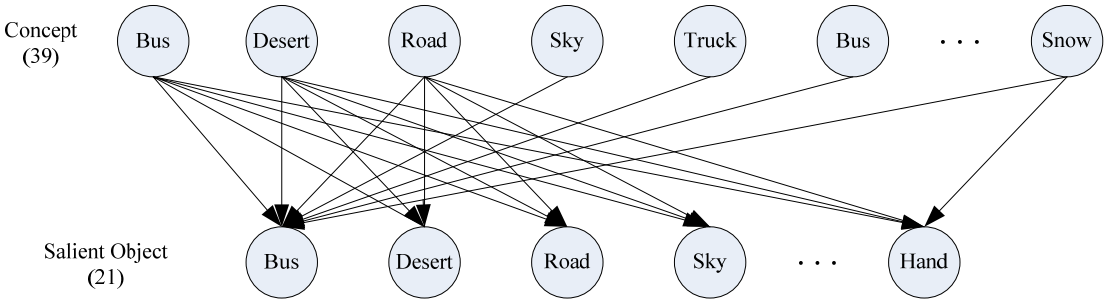


Figure.2 Ontology of concept detection

Figure 2 illustrates the ontology of concept detection, in which every two concepts in different layer has a line. It presents the probability dependence relationship (weight) between the two concepts. We compute the value of weight by conditional probability equation. After the ontology has been constructed, equation (4) can be used to compute the probability of every concept. The flowchart of concept is illustrated in Figure 3.

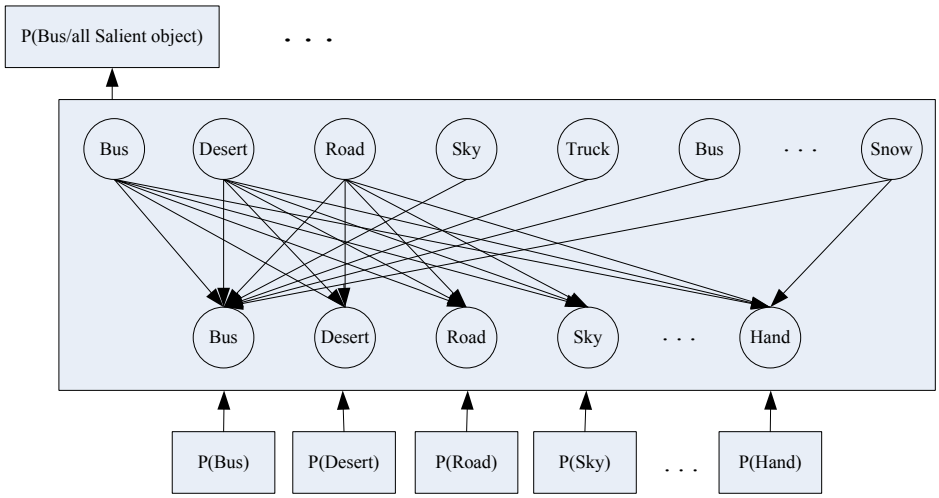


Figure.3 Flowchart of concept detection

In Figure 3, the input of our algorithm is the confidence of salient object, and output is joint probability of every high-level semantic concepts. Assume the set of salient object as $S = \{s_1, s_2, s_3, \ldots, s_{21}\}$, the set of all the high-level concepts as: $C = \{c_1, c_2, c_3, \ldots, c_{39}\}$, the weight of every c_k in C which corresponds to every salient object as $W_k = \{w_{k1}, w_{k2}, w_{k3}, \ldots w_{k21}\}$. Then the joint probability corresponding to every c_k in C is defined as follows.

$$P(c_k / S) = \sum_{i=1}^{21} (w_{ki} * P(s_i)) \tag{4}$$

where $P(s_i)$ is the confidence of the salient object which is detected by pattern classification. We judged a concept is in a test picture or not by comparing the joint probability to a threshold. Hence all the concepts can be detected.

2.2.2. Multitask Learning

In TRECVID problem setting, we are imposed with insufficient samples, especially for some high-level features, such as court, natural disaster, and prisoner. So we incorporate the multi-task learning to improve the generalization performance in the case of lack of training samples. In multi-task learning, each task seems to get extra training signals from other tasks as well as its own. Therefore, each learner in all tasks has much more training signals than learned in isolation.

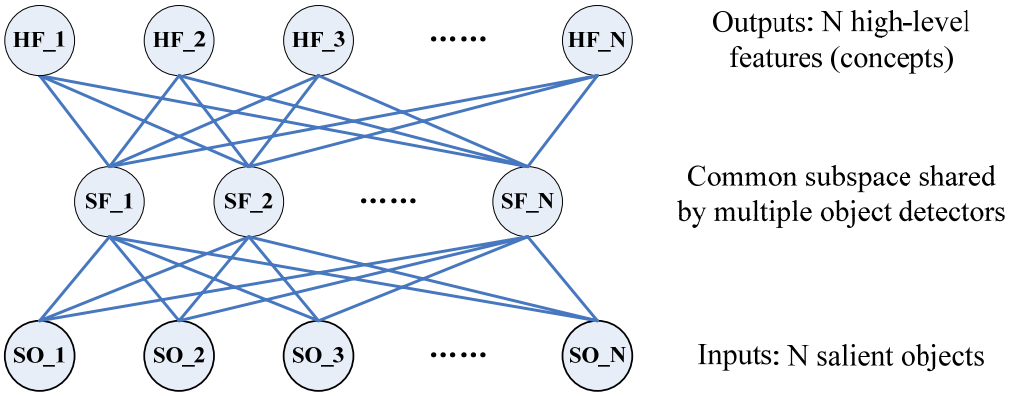


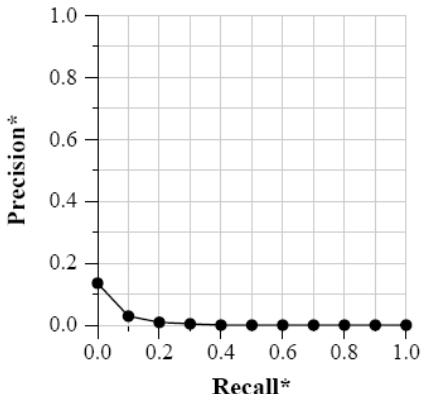
Figure.4 The architecture of multitask learning

We use the simplest three-layer network for multi-task learning. The inputs of the network for all tasks are in the same domain, i.e. 21-D vectors that are the confidences output from the salient object detectors. The second layer of the network is the common subspace shared by all the tasks, and the third layer is 39 final outputs of the high-level feature classifiers. The decision function is:

$$f^{(n)}(x) = w_n^T Bx \qquad (5)$$

Where B denotes the transformation matrix of the first-to-second layer, which projects the samples of all tasks to a common subspace, and the w_n is the weight for the nth task, i.e. nth high-level feature. We adopt the methods proposed in [6] to learn the classifiers.

Experimental results shows that run FD_SVM_BN_1 performed best in all four runs. Figure.5 shows the recall-precision and the precision at n shots of run FD_SVM_BN_1 and Figure.6 shows its results in detail.



Interpolated recall*	precision*
0.0	0.1350
0.1	0.0289
0.2	0.0090
0.3	0.0041
0.4	0.0000
0.5	0.0000
0.6	0.0000
0.7	0.0000
0.8	0.0000
0.9	0.0000
1.0	0.0000

Precision* at n shots	
5	0.0500
10	0.0500
15	0.0433
20	0.0500
30	0.0467
100	0.0570
200	0.0460
500	0.0316
1000	0.0249
2000	0.0190

Figure.5 Recall-Precision of Run FD_SVM_BN_1

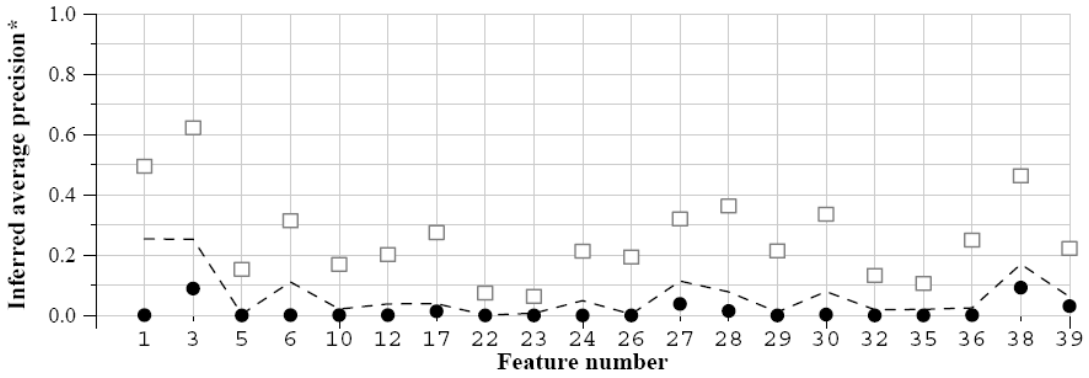


Figure.6 Run FD_SVM_BN_1 results

3. Search

For the search task of TRECVID 2006, we submitted 3 manual runs and 1 interactive run. We continued to use multi-modal information fusion which performed well in our work last year [7]. We also use some new methods such as machine learning and statistics to extract more information. In the following we will introduce these methods in detail.

3.1. Multi-Model for Video Retrieval

The aim of video retrieval is to find a set of video shots for a given query, which is formulated in multi-modalities including text description, global features, visual concepts and camera motion features. Every model only searches the video database using one kind of features which can't obtain satisfied query results, but the multi-model fusion can achieve better performance. According to last year's experience, global features and camera motion feature may bring noises into the retrieval. So we only use textual information and visual concepts this year.

3.1.1. Text Query Module:

Text query module is an IR search engine based on the match between the textual query and the portion of video transcript (includes ASR, OCR, and synchronized closed-captions) corresponding to the shot provides the evidence on the relevance of the shot. For the text query results, the TF*IDF weighting scheme is adopted to generate the text retrieval scores of shots. Considering a relevant shot does not always have keyword hit on itself, we use a related window to overcome this temporal mismatch by propagating IR score of a "hit" shot S_0 to its neighboring shots S_i in a window by an exponential decay function, i.e., $r(S_i) = r(S_0) \cdot \alpha^i$, where α is within [0, 1]. From this equation, the closer the shot is to the position of keyword hit, the larger score it gets.

3.1.2. Visual Concepts Module

Visual concepts are the high-level features which have been distilled from one of the TRECVID tasks. This year we used high-level feature results proposed by MediaMill [8]. They defined 101 concepts including people, object, event, and so on. Besides that we designed an anchor shot

detection method based on clustering, which detect the anchor shot by several specific characteristic of anchor shots, such as the position of anchor which is comparatively fixed, repeatedly appear in one video, the duration of a shot is comparatively longer, and so on. First we use the information of face detection to filter the shots which don't have faces and the face position does not fit the set threshold. Then we extract their HSV color histogram and perform clustering using the single-link clustering algorithm. The clusters which have the number of shots larger than the threshold are regarded as anchor shots candidates. Finally, we filter the shots whose durations are less than 3 seconds from the anchor shots candidates and get the anchor shots.

3.1.3. Multi-Model Fusion Based on Relation Expression

We use a search method which adopt multi-model query respectively and merge the results by relation expression. Figure.7 shows the overall framework of our multi-modal video retrieval system. We take every feature as an atom search engine and use them to search in video database respectively. Then the relation expression of every topic which we set beforehand is used to merge these query results and MC (Merge Confidence) is used to rank the final results. The method of MC is just to add all the confidence value of merge in relation expression and use the sum to rank the shots. So it is very effective and experimental results show that this method achieves promising performance.

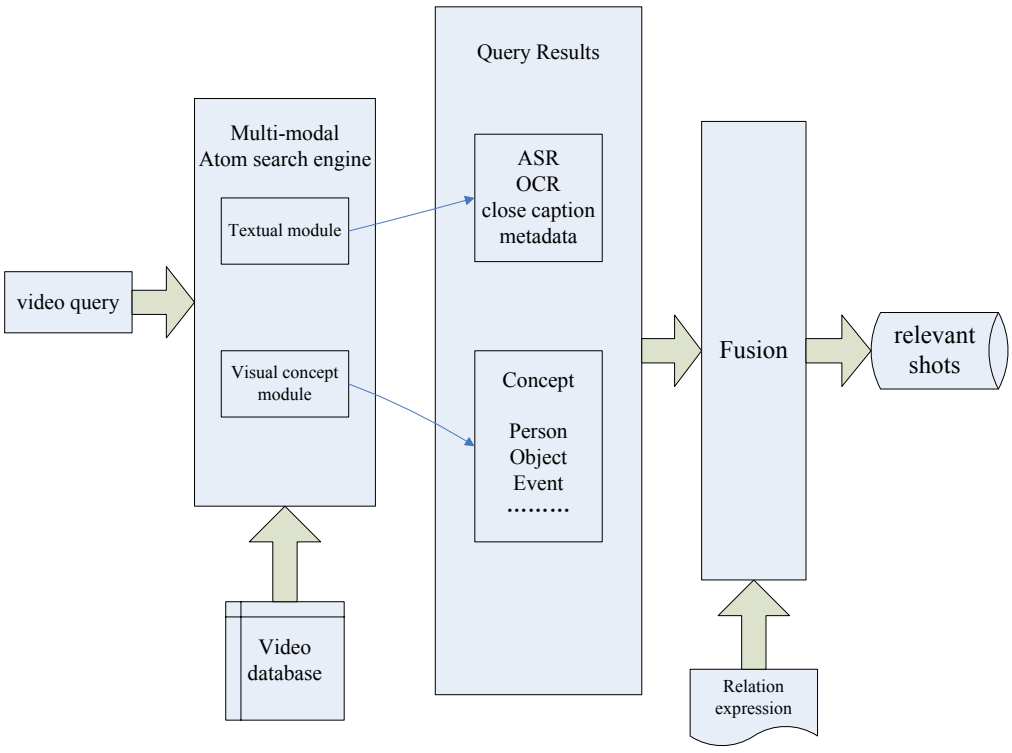


Figure.7 Framework of multi-modal retrieval model

3.2. Refining keywords used in text query

In manual runs, when we use text query module to retrieve relevant shots. We must extract keywords first. which is not an easy job. There exist two problems in it: (1) domain knowledge helps person to extract keywords, but one person cannot establish exact keywords of all topics

which influence the final retrieval results greatly; (2) some text transcript operation like ASR and OCR doesn't show good performance (precision actually about 40%), so maybe some important text information cannot be extracted. Even if we use keywords we think is proper, we still cannot get right result shots. We proposed a method: Keywords Training by Feedback (KTF) to train keywords of each topic. First, we establish basic keywords of each topic and expand them by means of WordNet [9], we use these expanded keywords to do retrieval for each topic in develop dataset (TRECVID2005 video sets). Then for each topic those words exist in right result shots are keyword candidates, we compute keyword candidates' frequency in all result right shots and rank them by their frequency. Finally we eliminate those candidates which have too high or too low frequency because it means they are too popular or too rare in text information. Figure.8 shows the framework of KTF.

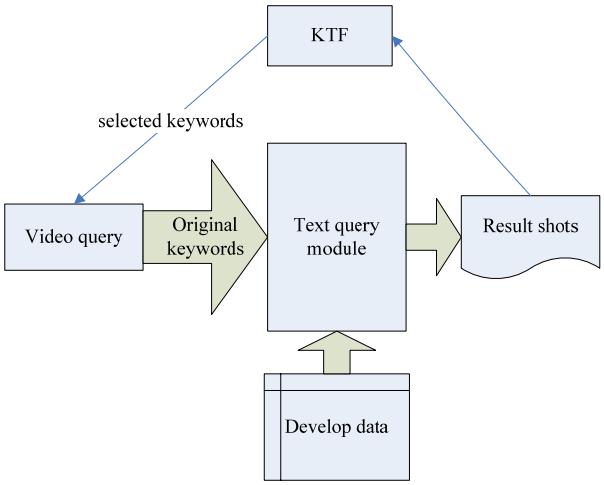


Figure.8 Framework of KTF

3.3. Interactive feedback based on Multi-Modal model

In our multi-modal video retrieval model, there exist some problems. We use expression relation (ER) to merge results generated by each atom search engine. But different weights of atom engine in ER may influence the finally results greatly. In manual run of search, we just set weights of each atom search engines manually which lacks scientific and mathematical reasons. In interactive run, we use feedback and logistic regression (LR) [10] method to train weight of each atom engine. New weights trained by LR are used to generate final results.

3.4. Evaluation

We submitted 3 manual runs and 1 interactive run to TRECVID 2006 for evaluation. They are:

Manual runs:

M_A_2_FD_M_TEXT_1: only use textual information, keywords expanded by WordNet.

M_A_2_FD_M_TRAIN_TEXT_2 : only use textual information, keywords selected by KTF.

M_A_2_FD_MM_BC_3: Use multi-modal model and fusion by relation expression, ranked by the method of MC. In text query module keywords are selected by KTF.

Interactive run:

I_A_2_FD_I_LR_4: use LR method to train weights based on multi-modal model

Experimental results of the MAPs (mean average precision) of our submissions against other submissions are shown in Figure. 9. There are 11 manual runs submitted to TRECVID 2006. We find that our run M_A_2_FD_M_TEXT_1 which only uses textual information gives the best MAP among all runs. It illustrates that textual information including ASR, OCR and close caption plays important role in video retrieval. But run M_A_2_FD_M_TRAIN_TEXT_2 didn't perform well which only ranked 9th place. We think there are two possibilities in it: (1) we didn't collect enough positive shots to get the statistical information about keywords and maybe we can collect some negative shots to join into the training course; (2) there exist some difference between develop data and test data, so sometimes selected keywords by KTF may not be reliable. Run M_A_2_FD_MM_BC_3 didn't perform well enough as expected because it used keywords trained by KTF and some concept results might bring noise which influenced the final result greatly. Our interactive run I_A_2_FD_I_LR_4 showed a disappointed performance this year. One problem is the number of positive samples we used to train is so limited, which directly led to insufficient training. We may use other machine learning methods like SVM in the next year.

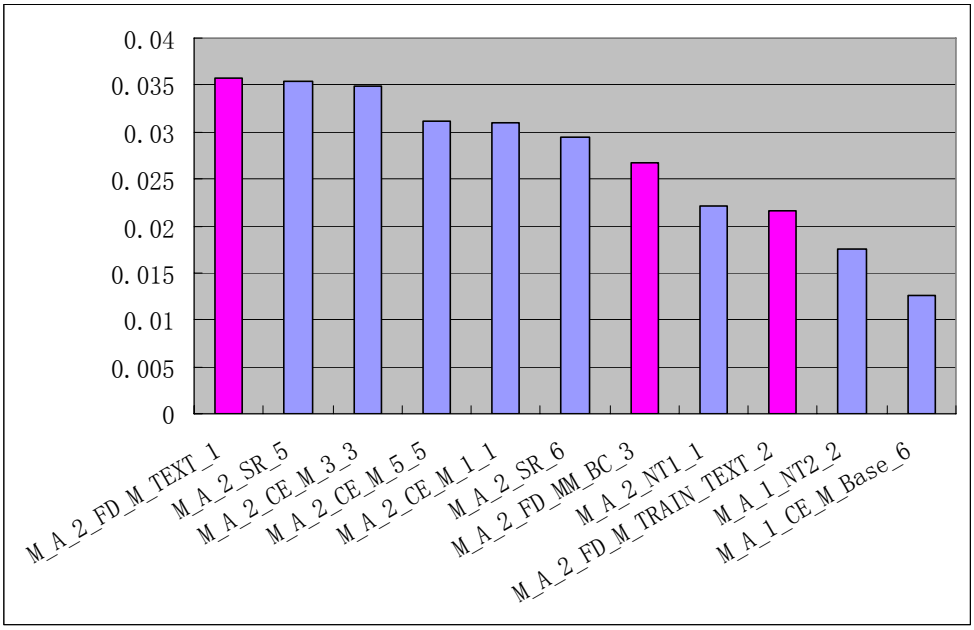
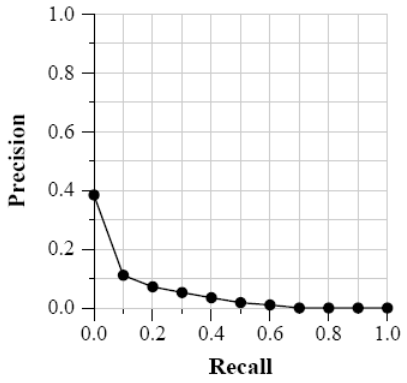


Figure.9 Performance of Fudan manual search submissions versus other manual submissions

Figure.10 gives the recall-precision and the precision at *n* shots of Run M_A_2_FD_M_TEXT_1. Figure.11 shows the result of run M_A_2_FD_M_TEXT_1, which has achieved a mean average precision (MAP) of 0.03575 which ranked 1st place in the total manual run. We have achieved best

or closed to best results for some of the queries.

Figure.12 shows the results of run I_A_2_FD_I_LR_4.



Interpolated recall	precision
0.0	0.3841
0.1	0.1110
0.2	0.0717
0.3	0.0525
0.4	0.0346
0.5	0.0177
0.6	0.0100
0.7	0.0000
0.8	0.0000
0.9	0.0000
1.0	0.0000

Precision at n shots	
5	0.1333
10	0.1167
15	0.1250
20	0.1229
30	0.1208
100	0.1083
200	0.0921
500	0.0789
1000	0.0601

Figure.10 Recall-Precision of Run M_A_2_FD_M_TEXT_1

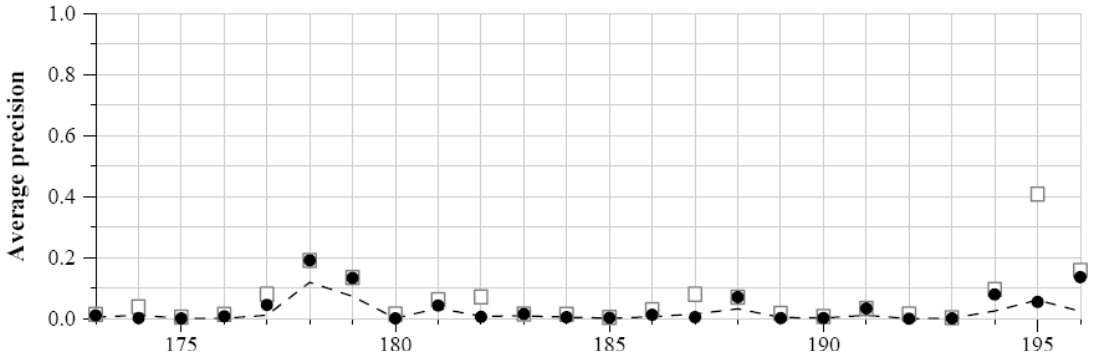


Figure.11 Run M_A_2_FD_M_TEXT_1 results

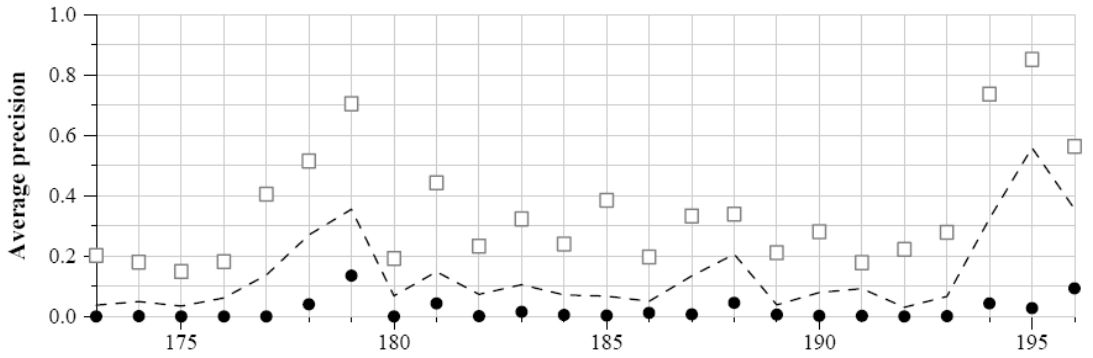


Figure.12 Run I_A_2_FD_I_LR_4 results

4. Summary

For high-level feature extraction task, we use the image segmentation method and extract the regional feature. Since only two features are used, the performance of salient object detectors

still needs much more improvement. And the fusion strategy is simply adding every classifier's confidence with the same weight. On learning the concept, we take two different ways: one is bayesian networks and the other is multi-task learning, both of which need a further research.

For search tasks, we submitted two runs which only used textual information. M_A_2_FD_M_TEXT_1 gives the best result among all submitted manual runs that illustrates textual information is very important. Another textual run whose keywords were selected by KTF didn't show good performance as expected, we may modify our algorithm in the next year. The method of multi-modal fusion still showed its effectiveness, we may modify its algorithm in establishing weights of atom search engine in the future.

Acknowledgement

This work was supported in part by Natural Science Foundation of China under contracts 60533100, 60402007 and 60373020, and Shanghai Municipal R&D Foundation under contract 05QMH1403.

Reference

[1] M. R. Naphade, L. Kennedy, J. R. Kender, S.-F. Chang, J. R. Smith, P. Over, and A. Hauptmann, "A Light Scale Concept Ontology for Multimedia Understanding for TRECVID 2005 (LSCOM-Lite)," IBM Research Technical Report, 2005.

[2] Deng, Y.; Manjunath, B.S.; Unsupervised Segmentation of Color-texture Regions in Images and Video, IEEE Trans. on Pattern Analysis and Machine Intelligence, Volume 23, Issue 8, Aug. 2001 Page(s):800 - 810

[3] H. Tamura, S. Mori and T. Yamawaki, Texture features corresponding to visual perception, IEEE Trans. on Systems, Man, and Cybernetics. 8 (6) (1978) 460–473

[4] Piater, J.H. Mixture Models and Expectation-Maximization, Lecture at ENSIMAG, May 2002, updated on Nov 15, 2004.

[5] Roberts, S.J. and Rezek, L. Bayesian Approaches to Gaussian Mixture Modeling, IEEE Trans. on Pattern Analysis and Machine Intelligence, November 1998, vol. 20, no. 11, pp. 1133-1142.

[6] R. A. Caruana, "Multitask Learning: A Knowledge-Based Source of Inductive Bias", In Proc. of ICML, 1993.

[7] Xue Xiangyang, Lu Hong, Wu Lide, Guo Yuefei, Xu Yuan, Mi Congjie, Zhang Jing, Liu Shenggui, Yao Dan, Li Bin, Zhang Shile, Yu hui, Zhang Wei, Wang Bei, "Fudan University at TRECVID 2005," TRECVID 2005.

[8] MediaMill, http://www.mediamill.nl/.

[9] WordNet, http://wordnet.princeton.edu/.

[10] Ofer Melnik, Yehuda Vardi, and Cun-Hui Zhang, "Mixed Group Ranks: Preference and Confidence in Classifier Combination", PAMI 2004.