The NTU Toolkit and Framework for High-Level Feature Detection at TRECVID 2007 *

Ming-Fang Weng[1], Chun-Kang Chen[1], Yi-Hsuan Yang[2], Rong-En Fan[1], Yu-Ting Hsieh[1], Yung-Yu Chunag[1], Winston H. Hsu[1,3], and Chih-Jen Lin[1,3]

[1] Department of Computer Science and Information Engineering,
[2] Graduate Institute of Communication Engineering,
[3] Graduate Institute of Networking and Multimedia,
National Taiwan University,
No. 1, Sec. 4, Roosevelt Road, Taipei, 10617, Taiwan

Abstract. In TRECVID 2007 high-level feature (HLF) detection, we extend the well-known LIBSVM and develop a toolkit specifically for HLF detection. The package shortens the learning time and provides a framework for researchers to easily conduct experiments. We efficiently and effectively aggregate detectors of training past data to achieve better performances. We propose post-processing techniques, concept reranking and temporal filtering, to exploit inter-concept contextual relationship and inter-shot temporal dependency. The overall improvement is 46% over that by our baseline in terms of *infMAP*. We briefly summarize our six submitted runs in this abstract. The run (runid: *A_nt20Giants_6*) adopts multiple low-levels features (all visual features), SVM models, ensemble bagging classifier, and multi-modal fusion. We take this setting as our baseline. We then experiment with post-processing methods and the leverage of classifiers using past data. The proposed post-processing framework is firstly applied to the baseline to obtain a new run (runid: *A_ntMonster_4*). in terms of *infMAP*, this new run improves 16.7% over the baseline The runs, *A_ntTank05_1* and *A_ntTransformer_5*, aggregate classifiers of using past data by averaging and weighted averaging their results, respectively. The results of these two runs, *A_ntTank05_1* and *A_ntTransformer_5*, are respectively 17.3% and 25.0% higher than that of *A_ntMonster_4*. Based the observation of our experimental results, we conclude that post-processing and using past data are helpful to improve HLE detection.

Table 1. Description of each submitted run

HLF Run	infMAP	Description
A_nt20Giants_6	0.0599	BASELINE: 20 bagging classifiers, multi-modal average fusion.
A_ntMonster_4	0.0699	bagging classifiers, weighted fusion, post-processing.
A_ntTank05_1	0.0820	bagging classifiers, weighted fusion, average aggregation, post-processing.
A_ntTransformer_5	0.0874	bagging classifiers, weighted fusion, weighted aggregation, post-processing.
A_ntReranking_3	0.0756	bagging classifiers, weighted fusion, average aggregation, reranking.
A_ntFiltering_2	0.0787	bagging classifiers, weighted fusion, average aggregation, filtering after reranking.

* This work is primarily supported by the National Science Council of Taiwan, R.O.C., under contracts NSC95-2622-E-002-018 and NSC96-2622-E-002-002.

1 Introduction

Due to the popularity of content sharing platforms such as YouTube, an increasing number of interesting videos are easily accessible in our daily life. Despite being able to enjoy various kinds of videos, people usually find that it is difficult to efficiently retrieve a specific video. Recently, much research has been devoted to addressing the issue of video indexing and retrieval [1, 2]. Semantic concept detection leads to more effective results because it can bridge the gap between low-level features and high-level human interpretations [3, 4]. In TRECVID 2007, we make several contributions described below.

Almost all public HLF detection baselines use LIBSVM [5] to train classifiers. However, as a general machine learning solver, LIBSVM suffers from long training time for HLF detection. We extend LIBSVM in three aspects specifically tailored for HLF detection. The total time is reduced from 14 days to about 3 days. The shorter training time thus allows us to experiment with various settings. Moreover, most public baselines only provide data (low-level features) and models (SVM parameters), but not tools, preventing them from being applied to new datasets. To address this problem, we will release our toolkit including the tailored LIBSVM and low-level feature extractors, which should facilitate experiments for large-scale concept detection.

Past dataset with annotations and trained classifiers are precious since they either require a tremendous manual effort or a huge learning time. In addition, they are potentially useful for a new dataset. To exploit the existent HLF detectors from prior data domains (e.g. , TV05-06), we average their output scores with those from newly trained classifiers. We observe that the video contents from the two sources are quite diverse. Therefore, concept-dependent aggregative weights are learned from a held-out set. Evaluation on TRECVID 2007 shows that the weighted combination with prior models learned from TV05 development set help TV07 detectors averagely improve 33% in *infMAP*.

To exploit contextual relationship among concepts and temporal dependency among shots, we propose post-processing approaches – concept reranking and temporal filtering. Motivated by observation that concepts usually co-occur in a shot. We learn occurrences by exploring contextual relationships among hundreds of concepts from Columbia374 [6] to improve accuracy. We investigated reranking methods to automatically model the contextual relationships from the initial detection results and rerank them accordingly. Because video is temporally correlated and a concept usually spans multiple shots, detection results from neighboring shots are helpful for the current shot. We proposed temporal filtering to explore this. The proposed post-processing approaches require neither separate training data nor additional learning processes; moreover, they are universally applicable to any detection results. Salient improvements are observed in contextually and temporally related concepts, e.g. Charts(45%), Boat_Ship(29%) by reranking and Weather(37%), Animal(17%) by filtering. Furthermore, both methods compensate each other and their parallel combination gains 10.1% improvements in *infMAP*.

2 A Toolkit for Semantic Concept Detection

There are several public HLF detection baselines for TRECVID benchmark [7], such as the MediaMill Challenge Problem [8], Columbia374 [6], and VIREO-374 [9]. All use LIBSVM [5] to train SVM classifiers. However, LIBSVM is designed for generic classification problems and thus suffers from long training time for HLF detection. To improve the efficiency, we extend LIBSVM in three aspects specifically tailored for HLF detection.

First, the main computational bottleneck of LIBSVM is on the inner products between sparse feature vectors. Since feature vectors for HLF detection are usually dense, we modify LIBSVM to calculate dense inner products. The new code reduces the computational time to around half.

Second, past research results indicate that SVM is sometimes sensitive to parameters. Using cross-validation, LIBSVM provides a tool to search parameters in a grid space. This code can be easily parallelized as independent SVMs (i.e. , SVMs with various parameters) can be run on different computers. However, sometimes jobs of certain parameters are longer than others. Hence, the speedup of parallelization is poor. Now for the HLF detection, assume the the system attempts to detect M concepts. If N feature-type are considered, the system requires $M \times N$ SVM models. Hence, $M \times N$ parameter-selection procedures are needed. As all of them are independent, we can combine all tasks together to obtain better parallelism. The LIBSVM parameter selection tool is modified to achieve this goal. Our CPU utilization is roughly increased from 75% to more than 90%.

Finally, the default search range of LIBSVM parameters is too large. Checking a smaller range already gives a reasonable SVM model. Hence, we narrow down our search space from the default 110 points to 42 points.

Overall, these adaptations cut the training time approximately from 14 days to about 3 days. With such improvements, it becomes more feasible to try variant experiments.

3 Semantic Concept Detection

3.1 Low-level Features

We adopt the six visual features of video descriptors used in *IBM Research TRECVID-2005 Video Retrieval System* [10]. These low-level features, including Color Histogram (HSV space, 166 dimensions), Color Correlogram (HSV space, 166 dimensions), Color Moments (Lab space, 225 dimensions), Co-occurrence Texture (96 dimensions), Wavelet Texture Grid (108 dimensions), and Edge Histogram Layout (320 dimensions), are extracted from each keyframe as the visual representations.

To avoid that some large feature values dominate the classification, we adjust the values of each single feature to about the same range[6]. This is done by normalizing each feature to have zero-valued mean and unit standard deviation.

3.2 Concept Modeling via Training SVMs

As described in Section 2, we develop a toolkit based on LIBSVM for fast learning HLF detectors. For most concepts in TRECVID benchmark data sets, the resulting classification problems are highly unbalanced. That is, there are a lot more negative samples than positive ones. To avoid that the SVM model predicts everything as negative, it is important to consider only a subset of negative samples. We follow *Columbia374* [6] to use all positive samples and around 20% negative samples. We further refine this sampling procedure by similar settings in [6]. The computational time is also reduced due to a smaller training set.

Since SVM parameters may influence the performance of classifiers, we mentioned in Section 2 that the grid-selection tool of LIBSVM is used. This tool is designed to obtain the best classification accuracy, but for HLF detection, the average precision (AP) [10] is used as the performance metric. We thus modify the parameter selection tool so that parameters achieving higher cross validation AP are chosen.

3.3 Ensemble Bagging Classifiers

Bagging is a common machine learning technique for combining classifiers [11]. It may help to obtain more stable results. Moreover, since we now select only a subset of negative samples for training, it is possible to train models with different subsets and then use them together. In our procedure, the output of 20 classifiers are averaged to give final predictions. Preliminary experiments indicate that this bagging procedure slightly improves the performance.

3.4 Multi-Modal Fusion

Many early TRECVID experiments report better results using multi-modal fusion. That is, results of using different feature types are combined. Basically there are two types of fusion: early and late fusions [12]. Early fusion is the concatenation of multiple features into a higher dimensional feature vector. One than trains a new SVM model. On the other hand, late fusion is the combination of separate output scores. In Tseng *et al.* [13], late fusion is shown to work better than early fusion. We thus experimente with two approaches for late fusion:

Average Fusion. For each feature type, we average SVM decision values from the multiple classifiers of the bagging procedure. We then use a standard Sigmoid function [6] to convert the score into the range of $[0, 1]$. Then six values from six feature types are averaged with equal weights for the final prediction.

Weighted Fusion. The six visual features may not be equally important for all concepts. Therefore, when averaging the six transformed values for the final prediction, we impose different weights to combine them. Their weights are proportional to the best cross-validation AP from the parameter-search stage.

3.5 Aggregation of Past Detectors

Past data sets with annotations are potentially useful for a new classification task. There are two possible methods to exploit these previous resources. One is the early aggregation in which all examples from different datasets are combined to train a new classifier. This may causes considerable training time due to the large data size. The other one is late aggregation, where classifiers trained on past data are directly used to predict new data. Thus, the prediction scores are combined with those from new classifiers. Similar to multi-modal fusion, we try two ways to aggregate the results of past classifiers.

Equally Average Aggregation. We simply average scores of past and newly trained classifiers.

Concept-dependent Weighted Aggregation. We observe that the video contents from the two sources are quite distinct. Therefore, old and new classifiers may not be equally useful for the same concept. Thus, concept-dependent aggregative weights are found via a hold-out validation. Different weigts are then applied to aggregate classifiers.

4 Post-processing Framework

4.1 Overview

We exploit post-processing techniques to incorporate context knowledge (both inter-concept and inter-shot) to further improve the accuracy of semantic concept detection in video. Figure 1 shows our post-processing framework for HLF detection using concept reranking and temporal filtering. During the training phase, concept reranking captures the inter-concept relationships in concept lexicon while temporal filtering models the temporal dependency among multiple neighboring shots. The inter-concept contextual relationships are

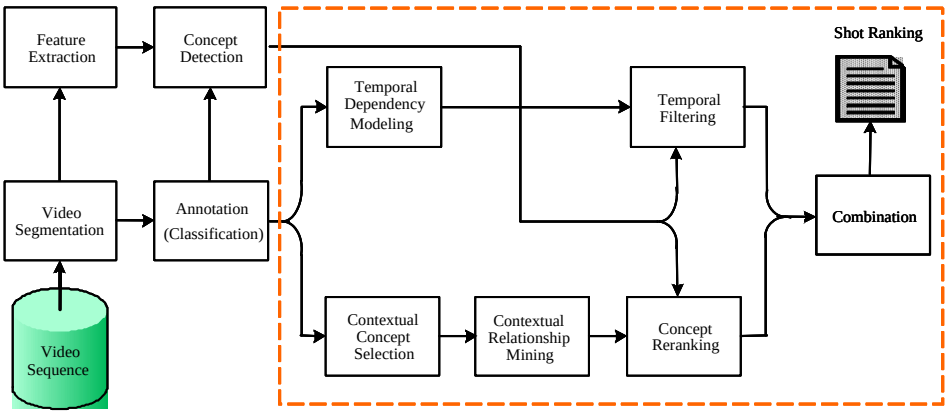

Fig. 1. Post-processing framework for HLF detection in video using concept reranking and temporal filtering.

mined in existent classification results, e.g. *Columbia374* [6], and the temporal dependencies are discovered from manual shot annotations without any extra training data. At the detection stage, given only the detection results for shots, our ad-hoc post-processing framework uses the learned contextual relationships and temporal dependencies to rerank and filter the test shots. After that, the scores from concept reranking and temporal filtering are combined to give the final ranking for shots.

4.2 Concept reranking

To improve the concept detection results, we exploit contextual information among hundreds of pre-trained concept detectors in a reranking framework, where we investigate variants of reranking methods to automatically learn the contextual relationships from the initial concept detection results and then further rerank them. We adopted discriminative reranking (e.g. , ListNet, RankSVM, etc.) since we found them more effective than similarity- or frequency-based ranking methods in mining the ordinal relationship. Besides, such reranking methods are ease of the ad-hoc thresholding for noisy binary labels and require no extra off-line learning processes or training data. We also propose several efficient feature selection methods to select effective contextual concepts for reranking each concept.

4.3 Temporal filtering

Videos exhibit temporal continuity in both visual content and semantics. Since the temporal dependency varies a lot among concepts, the temporal neighborhood distance of concepts and the temporal dependency of shots at different temporal distances should be considered. Firstly, as suggested by Liu *et al.* [14], the *chi-square test* with confidence level at 99.9% is used to determine the temporal distance (the size of dependency window) and we also set a maximal temporal distance at 20. Secondly, we try four different statistical measurements, including *chi-square test*, *likelihood ratio*, *mutual information* and *pointwise mutual information* [14] to determine a set of distance-dependent weighting coefficients. After another cross-validation on the TRECVID 2007 development data set, we decide to adopt *chi-square test* to measure the temporal dependency at each temporal distance and take its values as weighting coefficients.

To exploit temporal coherence, a temporal filter is designed to "smooth" the prediction of a shot with respect to a concept by a weighted combination of the *inference values* [14] of the shots within dependency window. The *inference value* infers the prediction value for a shot by using prior probabilities and the

likelihood of its neighboring shots. Thus, the inference value of a shot s_{t-k} for the current shot s_t, where s_{t-k} is k shots apart from s_t, can be defined as follows:

$$P(l_t=1|\mathbf{x}_{t-k}) = P(l_t=1|l_{t-k}=1)P(l_{t-k}=1|\mathbf{x}_{t-k}) + P(l_t=1|l_{t-k}=0)(1-P(l_{t-k}=1|\mathbf{x}_{t-k})),$$

where $\mathbf{x}_{t-k}$ is the visual features extracted from the shot s_{t-k}, $P(l_t=1|\mathbf{x}_{t-k})$ is the probability of current shot s_t containing concept l given the features $\mathbf{x}_{t-k}$ of neighboring shot s_{t-k}, $P(l_t=1|l_{t-k}=1)$ and $P(l_t=1|l_{t-k}=0)$ are prior probabilities estimated from the annotations, and $P(l_{t-k}=1|\mathbf{x}_{t-k})$ is the prediction value given by the detector indicating how likely concept l is present in shot s_{t-k}. For more details, please refer to Liu *et al.* [14].

4.4 Combination

Since concept reranking and temporal filtering can compensate each other, we perform parallel combination of both techniques. At first, concept reranking and temporal filtering are separately applied to classifier's results. Next, the scores of both methods are normalized to have zero-valued mean and unit standard deviation. The normalized scores are then averaged to give the final score. In addition to parallel combination, another combination can be adopted by applying temporal filtering on the results of concept reranking.

5 Experimental Results

5.1 Performance of Our Baseline.

We use 20% negative samples to train a classifier when concepts' annotations are highly unbalanced. Additionally, 20 bagging classifiers are used to form an ensemble classifier so that approximately 99.9% of negative examples could be sampled at least once. Our baseline *A_nt20Giants_6* is ranked 59th among all 163 submissions for the HLF detection task of TRECVID 2007. When weighted fusion instead of average fusion is applied to this baseline, the performance is slightly raised from 0.0599 to 0.061 in terms of *infMAP*, i.e. roughly 1.84% improvement.

5.2 Past classifiers.

As described in Section 3.5, we use classifiers of past data in two ways, equally average aggregation and concept-dependent weighted aggregation. Using TRECVID 2005 classifiers, Figure 2 shows that both methods of aggregation improves the performance on TRECVID 2007 data by 21.6% and 30.2% in *infMAP*, respectively. we observe that the classifiers trained by TRECVID 2005 data greatly help the performance of some concepts, e.g. , Desert, Sports, Maps, and Computer_TV-screen.

5.3 Post-processing techniques.

In Figure 3, we observe that improvements of concept reranking and temporal filtering on *infMAP*vary from 0.5 to 5.5% and from 3.0% to 3.8%, respectively. Though the improvements seem small, the parallel combination of inter-concept concept reranking and inter-shot temporal filtering can compensate each other. This combination further boosts the improvement to around 10.1% to 14.4% for different reference settings. Generally speaking, concept reranking, temporal filtering, and especially the combination of both techniques are quite helpful for HLF detection.

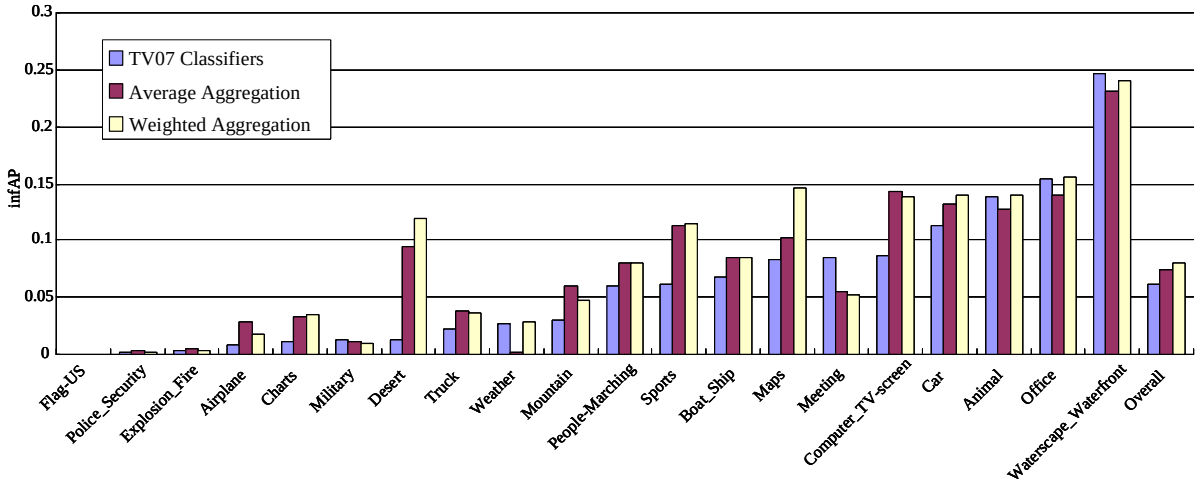

Fig. 2. Performance improvement from aggregating classifiers of past data.

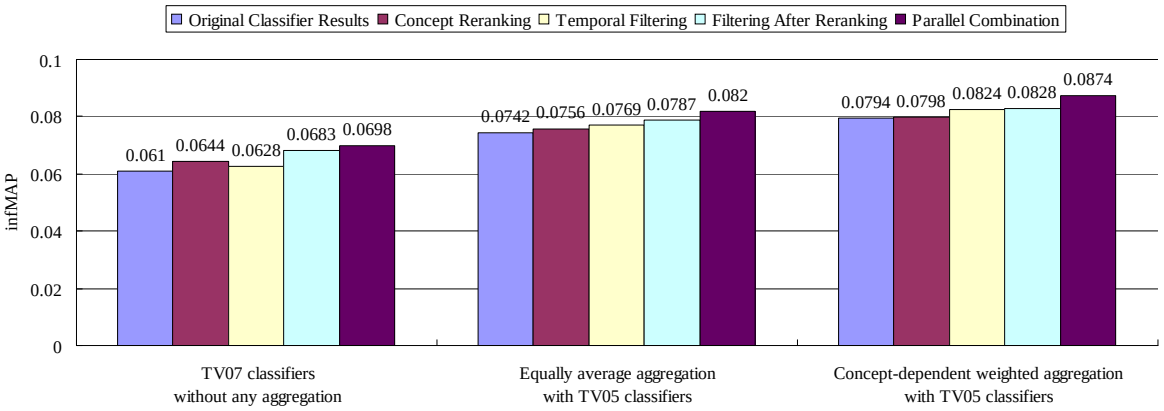

Fig. 3. Performance comparison with classier output, concept reranking, temporal filtering, filtering after reranking and parallel combination.

6 Conclusion

For HLF detection, we extend LIBSVM to improve efficiency of training SVM classifiers in three aspects. They are using dense representations of feature vector, parallelism between independent concepts, features, and SVM parameters, and narrowing down the search range of SVM parameters. Moreover, we use late aggregation to exploit the TRECVID 2005 classifiers by equally average aggregation and concept-dependent weighted aggregation. Evaluation on TRECVID 2007 shows that both of them help TV07 classifiers improve achieve much improvement. Besides, we propose post-processing approaches to enhance accuracy of semantic concept detection. Salient improvements are observed in contextually and temporally related concepts. Furthermore, the combination of concept reranking and temporal filtering gains improvements for HLF detection.

References

1. W. Adams, G. Iyengar, C.-Y. Lin, M. Naphade, C. Neti, H. Nock, and J. Smith, "Semantic indexing of multimedia content using visual, audio and text cues," *Eurasip Journal on Applied Signal Processing*, vol. 2003, no. 2, pp. 170–185, 2003.
2. C. G. M. Snoek and M. Worring, "Multimodal video indexing: A review of the state-of-the-art," *Multimedia Tools and Applications*, vol. 25, no. 1, pp. 5–35, 2005.
3. R. Datta, D. Joshi, J. Li, and J. Z. Wang, "Image retrieval: Ideas, influences, and trends of the new age," *ACM Computing Surveys*, 2007, to appear.
4. M. S. Lew, N. Sebe, C. Djeraba, and R. Jain, "Content-based multimedia information retrieval: State-of-the-art and challenges," *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 2, no. 1, pp. 1–19, 2006.
5. C.-C. Chang and C.-J. Lin, *LIBSVM: a library for support vector machines*, 2001, software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.
6. A. Yanagawa, S.-F. Chang, L. Kennedy, and W. Hsu, *Columbia University's Baseline Detectors for 374 LSCOM Semantic Visual Concepts*, March 20, 2007, Columbia University ADVENT Technical Report #222-2006-8.
7. A. F. Smeaton, P. Over, and W. Kraaij, "Evaluation campaigns and trecvid," in *MIR '06: Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval*, 2006.
8. C. G. M. Snoek, M. Worring, J. C. van Gemert, J.-M. Geusebroek, and A. W. M. Smeulders, "The challenge problem for automated detection of 101 semantic concepts in multimedia," in *Proceedings of the ACM International Conference on Multimedia*, Santa Barbara, USA, October 2006, pp. 421–430.
9. Y.-G. Jiang, C.-W. Ngo, and J. Yang, "Towards optimal bag-of-features for object categorization and semantic video retrieval," in *Proceedings of the ACM International Conference on Image and Video Retrieval (CIVR'07)*, Amsterdam, The Netherlands, 2007.
10. A. Amir *et al.*, "IBM research trecvid-2005 video retrieval system," in *TREC Video Retrieval Evaluation Online Proceedings*, 2005.
11. D. Tao, X. Tang, X. Li, and X. Wu, "Asymmetric bagging and random subspace for support vector machines-based relevance feedback in image retrieval," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 7, pp. 1088–1099, Jul. 2006.
12. C. G. M. Snoek, M. Worring, and A. W. M. Smeulders, "Early versus late fusion in semantic video analysis," in *Proceedings of the ACM International Conference on Multimedia*, Singapore, November 2005, pp. 399–402.
13. B. Tseng, C.-Y. Lin, M. Naphade, A. Natsev, and J. Smith, "Normalized classifier fusion for semantic visual concept detection," in *Proceedings of the IEEE International Conference on Image Processing*, Sept. 2003.
14. K.-H. Liu, M.-F. Weng, C.-Y. Tseng, Y.-Y. Chuang, and M.-S. Chen, "Association and temporal rule mining for post-processing of semantic concept detection in video," *IEEE Transactions on Multimedia, special issue on Multimedia Data Mining*, vol. 10, no. 2, February 2008, to appear.