BUPT&ORANGELABS(OrangeBJ) AT TRECVID 2014: INSTANCE SEARCH

Yanchao Feng[†], *Yuan Dong*[†‡], *Yue Wu*[†], *Hongliang Bai*[‡], *Shusheng Cen*[†]
Bo Liu[†], *Kai Wang*[†], *Yuxuan Liu*[†]

[†]Beijing University of Posts and Telecommunications,100876, P.R.China
[‡]Orange Labs International Center Beijing, 100013, P.R.China
{yuandong,censhusheng}@bupt.edu.cn

ABSTRACT

The framework of TRECVID Instance Search (INS) 2014 task is introduced by BUPT & Orange Lab Beijing. Two automatic and one interactive programs have been presented with 2014 TRECVID INS dataset, which has 464 hours of the BBC EastEnders programs, available in MPEG-4 format (Programme material copyrighted by BBC). Some novel methods have been tried. There are total eleven submissions namely:

OrangeBJ.F_X_NO_OrangeBJ_1_A: Color-SIFT feature, BOW, Lucene-based search, and face verification with SET-A.

OrangeBJ.F_X_NO_OrangeBJ_1_B: Color-SIFT feature, BOW, Lucene-based search, and face verification with SET-B.

OrangeBJ.F_X_NO_OrangeBJ_1_C: Color-SIFT feature, BOW, Lucene-based search, and face verification with SET-C.

OrangeBJ.F_X_NO_OrangeBJ_1_D: Color-SIFT feature, BOW, Lucene-based search, and face verification with SET-D.

OrangeBJ.F_X_NO_OrangeBJ_1_E: Color-SIFT feature, BOW, Lucene-based search, object tracking and face verification with SET-E.

OrangeBJ.F_X_NO_OrangeBJ_2_A: Color-SIFT feature, BOW, Lucene-based search with SET-A.

OrangeBJ.F_X_NO_OrangeBJ_2_B: Color-SIFT feature, BOW, Lucene-based search with SET-B.

OrangeBJ.F_X_NO_OrangeBJ_2_C: Color-SIFT feature, BOW, Lucene-based search with SET-C.

OrangeBJ.F_X_NO_OrangeBJ_2_D: Color-SIFT feature, BOW, Lucene-based search with SET-D.

OrangeBJ.F_X_NO_OrangeBJ_2_E: Color-SIFT feature, BOW, Lucene-based search and object tracking with SET-E.

OrangeBJ.I_X_NO_OrangeBJ_3_D: Color-SIFT feature, BOW, Lucene-based search and relative feedback rerank method with SET-D.

After experiments, the best mAP performances of each program above are **run1 :** 0.161(on Set E) ,**run2 :** 0.167(on Set D) and **run3 :** 0.249(on Set D) respectively. More image examples seem make the result better and the interactive run is better than the automatic runs. It is consistent with our experience.

Keywords

TRECVID, Instance Search, Face Verification, Object Tracking, Rerank.

1. INTRODUCTION

The number of videos on the web or somewhere is always increasing. A lot of new video clips occur on YouTube every day. Video explosion is just really happening. The requirements for searching special videos clips are more and more strong because the clips are delighting, instructive or useful. The search topics are related with objects, places, persons and so on.

Many state-of-art methods and algorithms have been proposed to meet with the above requirements in the recent years. They mainly include feature extraction, feature encoding, video search and rerank. In the feature extraction stage, the local feature is most frequently used. Its extraction basically has two steps: one is feature detectors, such as Harris detector, Harris Laplace detector, Hessian Laplace, Harris/Hessian Affine

detector, and the other is feature descriptors, such as Scale Invariant Feature Transformation (SIFT)[1], Shape Context, Gradient Location and Orientation Histogram, Speeded Up Robust Features(SURF), DAISY. In many applications, integration of features often has been proved to show better properties[2]. The reranking is usually regarded as machine learning problems, such as Rank SVM , IR SVM, AdaRank. The learning-based algorithm integrates both the initial ranking and visual consistency between images. Bag-Of-Visual-Word and inverted table framework are widely used in feature encoding and searching. Particularly, Apache Lucene is a high-performance, full-featured text search engine library, and we use it in image search successfully.

Instance search task can be regard as a visual search task. The performance is heavily dependent on the choice of data representation (or features) which they are applied. Hand-crafted features used to focus on visual invariance such as Light intensity change, Light color change and Scale[2]which making them suitable for the task.

Object tracking technology is widely used in video analyzing. INS-2014 gives video clips where the image examples come from, having contextual information of those image-queries. After tracking we generate more image-examples with different viewing angles of the topic objects for the runs on SET-E.

As we used to get the scores-result of each query image first, it is easy for us to merge the results for echo set conveniently, reserve the highest score shot and generate the results. For those queries whose ROI containing human face, we adopted face verification method: detecting the faces in the query image then compare with the faces detected in the database images, images with no faces detected will be ignored.

Interactive instance search is based on automatic runs. It can be regard as a rerank procedure and can make up the low performance of automatic search. Unlike last year, we tried a re-search-simple-merge strategy.

The rest of paper will introduce our work of implementing the INS task. Section 2 and 3 are the pipeline of our Automatic& Interactive runs and adding methods including: object tracking & face verification. The experiments and discussion are in the section 4. Finally, we will propose the future work to improve current performance.

2. AUTOMATIC&INTERACTIVE INSTANCE SEARCH

2.1. System Pipeline

The basic structure of our instance search consists of two stages: off-line and online stages.

In the off-line stage, we extract frames form videos at the rate of one image per second, then resize it to 75% compared to its original size, as a big image size can cause computational problems. Extracting features from images is essential and Color-Sift (CSIFT) has been implemented. With the CSIFT library, we randomly sample some descriptors, which are used to build a codebook. Then, we mapped the image descriptors (CSIFT) to words (by using codebook), mapping images to docs, and Lucene was used to index them ultimately.

In the online stage, given a query image and mask, we first extract its CSIFT descriptors and refine the descriptors by considering the information of its mask, then project them into the codebook tree.With running a Lucene search, ranking list can generated by measuring the similarity between the query and items in the reference database we built off-line. Finally, we did an interactive feedback which improves the search performance. Face verification was used to specific topics.

2.2 Feature Extraction and Refining

Local feature based on scale-invariant key point, has already been shown to be effective in multiple computer vision tasks. We use hessian-affine detector to detect key points and extract CSIFT descriptors to represent the local geometry of these key points and the dimension of resulting feature is 192. As we extract features from the whole image, it is inevitable to suffer

from noise. Only features raised from region-of-interest (ROI given by mask) of a query image should be reserved for searching and the features raised from background are considered as noise seem reasonable. However, in some topics, the targets always appear in a certain scene and the features raised from background are helpful to identify the scene. Especially when the target is a tiny object, which means few local features can be extracted from ROI, background becomes crucial cue to find the image containing the target. Finally, we made a trade off on above situations: we over-sample the ROI's features by duplicating them for 3 times, which has the same effect of putting large weight on them, this scheme combines information from the target object and background.

2.3 Codebook Training

Visual codebook plays an important role in mapping images to docs. The larger the codebook is, the more sensitive the retrieval system will be, which means it is close to a local copy retrieval system. Our system trained a 1M codebook using approximate k-means (AKM) from nearly 0.1m images' CSIFT descriptions.

In our implementation, the open-source library FLANN [1] [3] is used for fast approximate nearest neighbors search.

2.4 Indexing Using Lucene

Apache Lucene [2] is an outstanding text search software. It is open-source and has been deployed in many large web-sites as search engine. In a standard bag-of-word implementation, Vector Space Model (VSM) is used to compare the similarity between two images and the cosine distance of vectors after tf-idf weighting is computed as similar score. The similar scoring function used by Lucene is:

$$Score(q,d) = coord(q,d) * queryNorm(q) * \sum_{t\ in\ q} tf(t) * idf(t) * norm(d)) \quad (1)$$

where *queryNorm(q)* is a normalizing factor used to make scores between queries comparable, *coord(q, d)* is a score factor based on how many of the query terms are found in the specified document, and *norm(d)* is a normalizing factor to balance the inequality caused by non-uniform length of the indexed documents. With the trained codebook, we can encode each 192- dimensional CSIFT descriptor into one of the codes, completing the mapping procedure, so that images can be regarded as docs, and Lucene works.

2.5 Interactive Search Feedback

Interactive run is based on the automatic runs, and it can be regard as a rerank strategy. Given the lucene's result as an initial list, we label each shot-image as "relevant" or "non-relevant". Thanks to the good performance of the automatic run's speed, the top 1200 candidates on the list were reviewed during the limited time. In general the number of picked shots were less than 1000, using some typical images (picked out above) do an automatic search then merge the results, completing the list is what we did.

3. OBJECT TRACKING & FACE VERIFICATION

3.1 Object Tracking

In our system,Tracking-Learning-Detection (TLD) [4] is used for object tracking. First, we estimate a bounding box using the given outline of the object. Then, the track of this object is extracted from the video. And for face tracking, we use the method proposed in [5]. After tracking the object or the face, we obtain the related samples in all frames and utilize these samples to retrieve this object.

Figure 1: original examples and tracked examples

[1] http://www.cs.ubc.ca/research/flann/
[2] http://lucene.apache.org/

3.2 Face Verification

Viola-Jones classifier is used to detect faces in the given images. Then, twenty seven face landmarks are located by exploiting the method proposed in [6]. These landmarks are separated for different components (e.g. eyebrows, eyes, mouth, nose and the whole face). SIFT feature in different scales is extracted based on these landmarks and is concatenated to represent the components. At last, simile classifiers based on reference datasets are trained to build the high-level face descriptor**.**

The basic idea of our face description method is to use attribute classifiers trained to recognize the presence or absence of describable aspects of visual appearance (e.g., gender, race, and age). And to removes the manual labeling required for attribute classification and instead learns the similarity of faces, or regions of faces, to specific reference person. The similarities should be insensitive to environment variations. To train simile classifiers, we first have created a dataset of 244 reference persons. The dataset consists of 54415 images. The largest person has 750 images while the smallest one has 89 images. Some images of the dataset are from PubFig dataset (49 persons), while the remainders are crawled from Baidu and Google Images (195 persons).

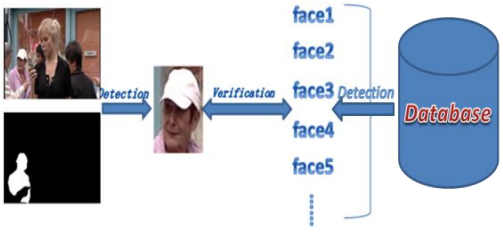


Figure 2: Face Detection and Verification

In each component image, SIFT feature in two different scales can be obtained. The feature is fed into 1000 best simile classifiers which are selected in the training process. The output value of each classifier is concatenated to construct 1000D high-level descriptor. Instead of building the targeted person classifier which is used last year, we try to use the face verification technology to retrieve the targeted person. The face verification answers the question whether two faces are the same person. We use the training samples provided by LFW [7] to train the face verification classifier. And the classifier is RBF support vector machine. All test face images and the targeted face are fed into this classifier and the output values server as the similarity between test faces and the targeted person's face.

4. EXPERIMENTS

4.1. Database Description

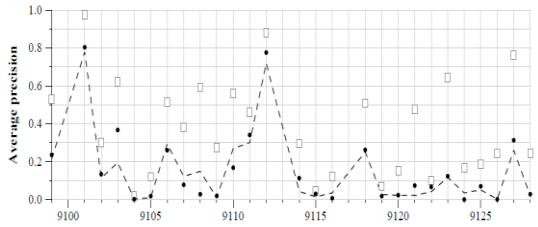
TRECVID is a laboratory-style evaluation that attempts to model real world situations or significant component tasks involved in such situations [8]. The INS-Task's main purpose is finding more video segments of a certain specific person, object, or place, given a visual example (a pair of images: one whole image and one mask image <point out the ROI>). INS2014 created 30 topics (only 27 topics have been final considered) and each topic contains four pair of images (each pair have a video clip form where it was extracted) [9].

Figure 3: Query Image & Mask examples

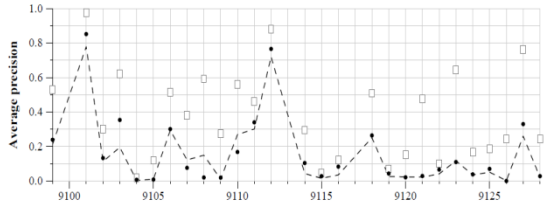
4.2 Automatic INS Performance

The automatic runs have two programs and have run on all the SETS (A to E). The best results are shown Fig.4. The result gets better and better with the number of query examples getting larger and larger, this can be seen in comparing the results of set A, B, C, and D. In program-1, the MAP is: 0.135(SET-A),0.139 (SET-B), 0.158(SET-C), 0.161(SET-D), 0.161(SET-E). And we finally see result-E is not better than result-D in program-2, the main reason I think is that the masks of

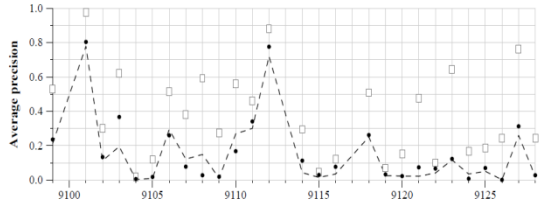
the tracked queries is rectangular which don't fit the object well enough. With the positive impact getting smaller and smaller as the images describe the same thing, the negative impact of noise covered the positive impact of adding query-images. Poor resolution of the tracked object images also has bad impact on final results. The face verification algorithm take much more time and seems didn't make a difference, maybe the queries' face is just not in good condition for verification. Among the results, MAP of topic 9109(a Mercedes star logo), 9126(a Peugeot logo) and 9128(this F pendant) are closed to zero because of few features.



(a) OrangeBJ.F_X_NO_OrangeBJ_1_E(SET-E)



(b) OrangeBJ.F_X_NO_OrangeBJ_2_D(SET-D)



(c) OrangeBJ.F_X_NO_OrangeBJ_2_E(SET-E)

Figure 4: Three results of Automatic runs

4.3 Interactive INS Performance

The interactive run result is shown in Fig. 5, the MAP of this run is 0.249 and won the second place. As it regards the automatic run's result as an initial list and a basic procedure, in addition to more robust rerank algorithms, a quicker and preciser automatic search engine will also make the interactive run better.

5. CONCLUSIONS

The small or less-texture objects are still hard to deal with as few visual descriptions can be generated from them. With the good time performance of automatic run, many different rerank strategies can be carried out. Strategy integration is also a promising way to improve performance such as using different methods to handle small and big topic objects, dog and vase.

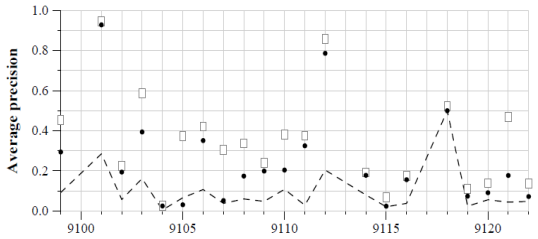


Figure5: OrangeBJ.I_X_NO_OrangeBJ_3_D(SET-D)

6. REFERENCES

[1] D. Lowe,"Distinctive Image Features from Scalein Variant Keypoints" IJCV, vol. 60, no. 2, pp. 91–110,2004.

[2] Sande.K., Gevers.T, Snoek.C.G.M,"Evaluating color descriptors for object and scene recognition" IEEE Trans. on PAMI 32, 1582–1596 (2010).

[3] Marius Muja and David G. Lowe,"Fast approximate nearest neighbors with automatic algorithm configuration" VISAPP , 331--340, pp.331-340,2009.

[4] Kalal, Z., Mikolajczyk, K., & Matas, J. (2012). Tracking-learning-detection.Pattern Analysis and Machine Intelligence, IEEE Transactions on, 34(7), 1409-1422.

[5] Ozerov, A., Vigouroux, J. R., Chevallier, L., & Pérez, P. (2013, September). On evaluating face tracks in movies. In IEEE International Conference on Image Processing (ICIP 2013).

[6] Belhumeur, P. N., Jacobs, D. W., Kriegman, D., & Kumar, N. (2011, June). Localizing parts of faces using a consensus of exemplars. in Computer

Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on (pp. 545-552). IEEE.

[7] Huang, G. B., Mattar, M., Berg, T., & Learned-Miller, E. (2008). Labeled faces in the wild: A database forstudying face recognition in unconstrained environments. In Workshop on Faces in'Real-Life'Images: Detection, Alignment, and Recognition.

[8] Alan F. Smeaton, Paul Over, andWessel Kraaij, "Evaluation campaigns and trecvid," in MIR '06: Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval, New York, NY, USA, 2006, pp. 321–330, ACM Press.

[9] Paul Over, George Awad, Martial Michel, Jonathan Fiscus, Greg Sanders, Wessel Kraaij, Alan F. Smeaton, and Georges Quenot, "TRECVID 2014 – An Overview of the Goals, Tasks, Data, Evaluation Mechanisms and Metrics" in Proceedings of TRECVID 2014. NIST, USA, 2014.

[10] Sivic and Zisserman, "Video Google: A Text Retrieval Approach to Object Matching in Video", ICCV, 2003.

The Orange Labs International Center China (FTRDBJ) Video Semantic Indexing Systems – TRECVID 2014 Notebook Paper

Kun Tao[1], Yuan Dong [2], Yunlong Bian[2], Hongliang Bai[1], Yinan Liu[1]

[1]Orange Labs International Center China, Beijing, 100013, P.R.China
[2]Beijing University of Posts and Telecommunications, Beijing, 100876, P.R.China
taokun81@hotmail.com yuandong@bupt.edu.cn

ABSTRACT

In this paper, we introduce our Convolutional Neural Networks (CNN) systems in TRECVID2014 Semantic Indexing task. This year, we submitted 4 main task runs: a baseline run using traditional composite-kernel SVM, 2 CNN based runs and an ensemble of them. Our CNN models are pre-trained on ImageNet dataset. Then it can be used as a feature extractor or be transformed to SIN concept detector by fine-tuning. The experiments show that using CNN feature and SVM classifier performs better than our previous methods. Finally, the ensemble run reached our best MAP 0.232.

1. INTRODUCTION

In the past several years, the deep neural network technology has brought noticeable improvement to many classic pattern recognition tasks including audio recognition, image recognition, face recognition etc. It is especially noteworthy that the convolutional neural networks (CNN) are widely implemented in ImageNet Large Scale Visual Recognition Challenge and show amazing strong performance [1,2]. Orange Labs China also built a CNN based visual recognition engine of 1000 object classes using ImageNet dataset and reached an acceptable performance. It's naturally that we wish to transform this engine and use it to detect appointed semantic labels in TRECVID SIN evaluation.

The 2014 SIN task kept using 60 concepts and IACC.1 training set [3,4], the small concept corpus and unbalanced positive sample numbers will cause over-fitting in CNN. So we shouldn't used SIN training data to training a CNN model directly. Instead, our CNN model was pre-trained on ImageNet dataset. Then it could be directly used as a feature extractor or transformed to 60 SIN concept detector by a fine-tuning phase [5]. To use CNN as a feature extractor, we tried the 1000-D final softmax output and 4096-D second fully-connected layer output. The 4096-D feature show better performance. For fine-tuning, we modified the fully-connected layers and trained new 60-way classification layer to replace the original 1000-way layer. But after all it didn't reach an expected performance.

Finally we submitted 4 main task runs. The basic information of submitted runs is shown below:

- 2B_M_A_OrangeBJ.14_1: Using composite-kernel SVM with 9 features. MAP = 0.174.
- 2B_M_D_OrangeBJ.14_2: CNN feature + SVM. MAP = 205.
- 2B_M_D_OrangeBJ.14_3: CNN fine-tuning. MAP = 0.117.
- 2B_M_D_OrangeBJ.14_4: ensemble of above 3 runs. MAP = 0.232.

2. CNN BASED SEMANTIC INDEXING

2.1 Network Architecture

The architecture of our CNN model is shown in Fig. 1, which includes 5 convolutional layers and 3 fully-connected layers. Response-normalization layers + 3×3 max-pooling layers are used in the first, second and last convolutional layers. The ReLU non-linearity is applied to all convolutional and fully-connected layers. The first convolutional layer filters 224×224×3 input images with 96 7×7×3 kernels with a stride 2. The following 4 convolutional layers has 256 5×5×96 kernels, 384 3×3×256 kernels, 384 3×3×384 kernels and 256 3×3×384 kernels respectively.

For pre-training on ImageNet dataset, the last softmax classification layer is 1000-way and in fine-tuning run it's replace by 60-way classifier. The neuron number of the first and second fully-connected layers in pre-training is 4096, but in fine-tuning it's reduced to 2048 for preventing over-fitting. In both pre-training and fine-tuning a multinomial logistic regression objective function is used. We use 50% dropout in the first and second fully-connected layers during pre-training.

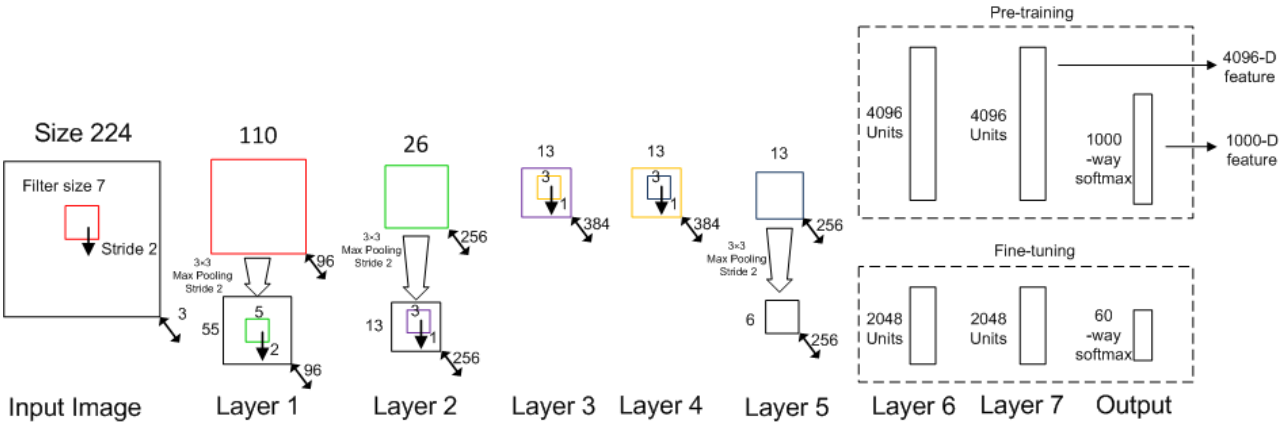


Fig.1 CNN Model Architecture

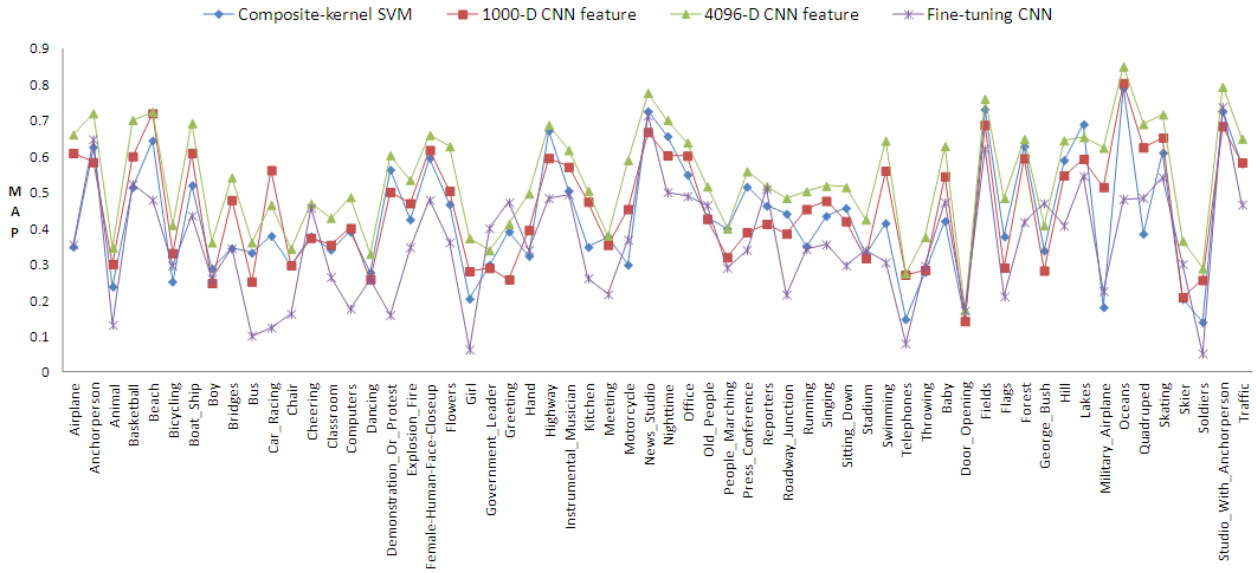


Fig. 2 Evaluation MAP of 4 Methods

2.2 CNN Feature Extractor

Intuitively, we can use the pre-trained CNN model to analyze SIN key-frames. The output of 1000-way classifier can be regarded as semantic histogram features of the key-frames and be used to train a mapping model to the 60 SIN concepts. As a further consideration the 4096D output of the second fully-connected layer can be regarded as a kind of mid-level semantic feature. In our experiments, above two kinds of features were used to train SVM classifiers for SIN evaluation. We found that χ^2 kernel works best for these features and 4096D feature outperforms than 1000D feature.

2.3 Transform Model by Fine-tuning

Replacing the 1000-way softmax layer with a 60-way layers corresponding to 60 SIN concepts, we can train a fine-tuned model for SIN task. To prevent overfitting, the neuron number of the first and second fully-connected layer is reduced to 2048. As most of the classification capacity of the CNN model comes from the 5 convolutional layers, the fine-tuned model can inherit the powerful features pre-trained on ImageNet dataset.

The new fully-connected layers were randomly initialized. Then only positive samples of 60 SIN concepts were used for fine-tuning training. To avoid over-fitting, a small learning rate (0.0005) was used in this step.

3. EXPERIMENT RESULTS

To evaluate our systems, 70% IACC.1 labeled key-frames were selected as training set and 30% for evaluation. As the CNN models need a fixed input size of 224×224, an adaptive scale strategy is used to cut the two sides of some frames in unconventional aspect ratio. A comparison result of our composited-kernel SVM baseline [6], 1000-D/4096-D CNN feature models, and fine-tuning CNN

models is show in Fig. 2. The evaluation MAP of above 4 methods are 0.429, 0.455, 0.534 and 0.360 respectively. It's obvious that the 4096-D CNN feature outperforms than the other 3.

As the I-frames of IACC.2.B were provided by NIST, a clustering based key-frame extractor [7] was used to extractor 1 to 3 RKF and NRKFs for testing video shots. 126963 key-frames were extractor for 89170 shots with at least one I-frame.

The run1 ~ run3 submitted by us are based on composite-kernel SVM, 4096D CNN feature and fine-tuning CNN respectively. Then the 4th run is an ensemble of them. A weighted average is used with the weights 0.3/0.5/0.2 correspondingly. The testing results provide by NIST provided that the 4096D CNN feature is the best solo run and the ensemble run can even reach a better MAP of 0.232.

Reviewing above experiment results, the most regrettable thing is the fine-tuning model didn't reach an expected performance. It should be caused by two reasons:

First, only positive samples were used in fine-tuning. Current training labels can't cover all iacc.1 key-frames. As the negative samples are much more than positive samples, it becomes frequent that a positive sample of one concept acts as the negative samples of several other concepts. To avoid such confusion, we only use positive samples to fine-tune the 60-way softmax model. The small concept corpus with unbalanced sample number increased the probability of over-fitting.

Second, most image samples in ImageNet dataset are chosen carefully, which are clean and typical. The target objects occupy the main part of the pictures. But for some SIN samples, the target objects are no significant enough and even too small to help the training. Directly follow the architecture used in ImageNet can't reach an expected result on difficult SIN task.

In the future, we wish to found a better architecture and training strategy for fine-tuning on SIN dataset. If we can overcome above two problems, an obvious improvement can be expected.

4. CONCLUSION

This year, the CNN based methods were introduced to our SIN systems for the first time. It brought remarkable improvement. Using CNN as a feature extractor is proved to be powerful because it's flexible and can combine the advantage of CNN and SVM. But how to use fine-tuning to transform a CNN model is still a challenging problem. Anyway, we wish to use this new tool to boost our system revolution in many stagnating tasks in the future.

5. REFERENCES

[1] O. Russakovsky, J. Deng, etc. "ImageNet Large Scale Visual Recognition Challenge", ILSVRC2014

[2] A. Krizhevsky, I. Sutskever, G. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks", NIPS2012

[3] P. Over, G. Awad, etc. "TRECVID 2014 -- An Overview of the Goals, Tasks, Data, Evaluation Mechanisms and Metrics ", http://www-nlpir.nist.gov/projects/tvpubs/tv14.papers/tv14overview.pdf, 2014

[4] S. Ayache and G. Quénot, "Video Corpus Annotation using Active Learning", 30th European Conference on Information Retrieval (ECIR'08), 2008

[5] Y. Bian, Y. Yuan, etc. "Reducing structure of deep Convolutional Neural Networks for Huawei Accurate and Fast Mobile Video Annotation Challenge", ICMEW2014

[6] K. Tao, etc. "The France Telecom Orange Labs (Beijing) Video Semantic Indexing Systems – TRECVID 2012 Notebook Paper," http://www-nlpir.nist.gov/projects/tvpubs/tv.pubs.12.org.html, 2012.

[7] K. Tao, Y. Dong, etc. "The Orange Labs International Center China (FTRDBJ) Video Semantic Indexing Systems – TRECVID 2013 Notebook Paper", http://www-nlpir.nist.gov/projects/tvpubs/tv. pubs.13.org.html, 2013.