Query-Adaptive Late Fusion with Neural Network for Instance Search

Vinh-Tiep Nguyen #, Dinh-Luan Nguyen #, Minh-Triet Tran #, Duy-Dinh Le +*, Duc Anh Duong *, Shin'ichi Satoh +

# *University of Science, Vietnam National University, Ho Chi Minh city*

* *University of Information Technology, Vietnam National University, Ho Chi Minh city*

+ *National Institute of Informatics, Tokyo, Japan*

nvtiep@fit.hcmus.edu.vn, 1212223@student.hcmus.edu.vn, tmtriet@fit.hcmus.edu.vn,
ledduy@nii.ac.jp, ducda@uit.edu.vn, satoh@nii.ac.jp

Abstract—**Bag-of-Word based model is one of the state-of-the-art approaches for object retrieval or also known as instance search problem. Although this model and its extensions are good for rich-textured objects, it is still unsolved for searching on textureless ones. In this paper, we propose to combine this model with Deformable Part Models object detector using late fusion technique to improve final result. To find the optimal weights for each type of query objects, we further propose to use a neural network to learn query features including object area, number of shared visual words to get optimal weights for each model. Experimental results on TRECVID Instance Search (INS) dataset with queries in INS2013 and INS2014 show that our proposed method significantly improves 18.48% and 14.63% in mAP respectively comparing to standard BOW model and outperform other state-of-the-art methods. This method opens a new way of adaptively combining DPM, an object detector, in a hybrid model for visual instance search.**

I. Introduction

Retrieving or searching objects in video scenes is one of many challenging tasks in computer vision. Instance Search (INS), defined by TRECVID [1], is to find video segments of a certain specific object, place, or person, given visual examples from a video collection. This trend of search has various practical applications, such as surveillance, personal video organization, law enforcement, video search archive, etc. In this paper, we concentrate on instance search problem in video databases comprised hundreds of thousands of shots with millions of frames.

Bag-of-Word (BOW), introduced by J. Sivic [2], is well known for being a state-of-the-art approach in video retrieval. The key idea of this method is based on the observation that two similar images share significant number of local patches that can be matched with each other. To find interesting regions of rich-textured objects such as paintings, bunch of flowers, buildings, etc., sparse feature detectors (e.g. DoG [3], Hessian-Affine [4], MSER [5]) are proposed and widely used. Many extensions are proposed, such as spatial reranking [6], [7] and query expansion [8], to improve the accuracy of BOW.

One important and prerequisite technique is enforcing spatial consistency (e.g. topology checking [9], weak geometric consistency (WGC) with Hamming Embedding (HE) [10]). However, this method is easy to be failed since a less textured query object may leads to the confusion with object-like background. Sampling in dense grid at various scales is proposed to get over this defect. Dense feature is applied in various applications such as fine-grained classification [11], action recognition in video [12]. However, this approach increases computational cost and is not suitable for large scale datasets. Thus, using dense feature at post-processing stage is better than at early stage. Based on that idea, we use HOG sharing common characteristics with dense feature to overcome difficulties. We also use Deformable Part Model (DPM) [13], mainly based on HOG, to demonstrate our method to verify object existence in video shots because it is the state-of-the-art in object detection.

Most of INS systems which achieve highest results in TRECVID competitions [9], [14] are based on BOW. In fact, BOW is suitable for large and rich textured objects but it may perform not very well with small and fairly textured ones. Hence, current INS systems still perform not very high in mean average precision (mAP) when retrieving a wide range of different query objects, each of which has its own properties that should be exploited, such as the size and the complexity of textures in the query object.

In this paper, our objective is to propose a novel instance search system that can analyse an arbitrary query object to exploit its properties to automatically tune the re-ranking process to well-match with the query object. In our system, we combine BOW and DPM to handle various types of query objects. We use BOW model as it is suitable for big and rich textured objects. Meanwhile, DPM detector can be used as an efficient supplemental tool to verify the existence of a query object with small size or poor textures in a video shot with variant in illumination and occlusion. Figure 1 illustrates some query topics that are suitable for BOW only (a); both BOW and DPM (b); and DPM only (c), respectively.

The novel idea in our system is that we propose to use neural network to automatically estimate an adaptive weight

MMSP'15, Oct. 19 - Oct. 21, 2015, Xiamen, China.
978-1-4673-7478-1/15/$31.00 ©*2015 IEEE.*

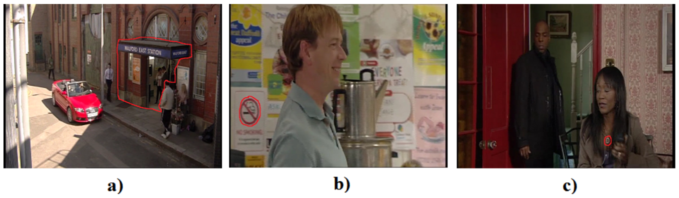


Fig. 1. Examples of query topics inside red curves in TRECVID Instance Search challenge

to combine the two scores calculated by both BOW and DPM to re-rank candidates in the final ranked list. By this way, our system can be appropriate with many kinds of query objects, some of which are more suitable with BOW while the others are more appropriate with DPM.

Main contribution. As far as we know, no previous work addresses adaptive fusion of BOW and DPM to solve a wide range of various objects for instance search in video. The soul of our method is adaptive fusion DPM and BOW model based on characteristics of query objects. The performance of our method is better than other reranking methods such as geometric consistency checking, multi-feature technique, etc.

The rest of this paper is organized as follows. Section II discusses some previous works related to our proposed method. In section III, our instance search system on video is presented. Query-adaptive method using neural network for choosing weights of DPM and BOW models is presented in section IV. Section V meticulously explains our experiments on TRECVID dataset. Finally, section VI concludes whole ideas of the paper and discuss some future works.

II. Related work

Many techniques have been proposed to boost the performance of INS systems, such as RootSIFT feature [8], large vocabulary [6], soft assignment [15], multiple detectors and feature combination at late fusion [1], query-adaptive asymmetrical dissimilarities [14], etc.

Spatial verification is usually used to verify the existence of a query object in a retrieved shot. This is one of the most effective approaches and also serves as the prerequisite step for other advanced techniques such as query expansion[8]. Spatial verification can be classified into two categories: spatial reranking and spatial ranking.

Spatial reranking checks on a short list of about 200 to 1000 results returned from BOW model for geometric consistency of visual words. It exploits the local shape of the affine covariant region for rigid affine consistency checking, which was first applied by Philbin et al. [6]. Hough Pyramid Matching approach for spatial reranking used a hierarchical structure to group matches, thus resulting in an algorithm which is only linear in the number of putative correspondences [7]. Also, an elastic spatial checking technique was proposed to emphasize topology layouts of matching points [9].

Spatial ranking incorporates the spatial information at the original ranking stage to improve the efficacy of the search system. For instance, Jegou et al. [10] used a Houghlike voting scheme in the space of similarity transformation between the query and database images, yet this is just a weak geometric consistency checking (WGC). Cao et al. proposed to use spatial-bag-of-features which capture the spatial ordering of visual words under various linear and circular projections [16]. Shen et al. proposed to transform the query ROI by the predefined scales and rotations [17]. However, this method is much more expensive in computation than other approaches such as BOW or WGC.

However, most of current methods may not perform well for objects with small size, less textures, or in confusing background. Therefore, methods based on dense features, such as Deformable Part Model [13] - one of state-of-the-art methods in object detection, can be applied to overcome such obstacles in object existence verification for INS. Crowley et al. proposed to use DPM in combination with mid-level discriminative patches (MLDPs) [18] in the reranking step for object retrieval in paintings. However, it is just an average (hard assignment) late fusion of DPM and MLDP model.

III. Proposed Instance Search system

Figure 2 illustrates our proposed system with 5 main modules. The baseline module is based on BOW to retrieve top K shots corresponding to a given query topic including one or several examples of a single query object. Two DPM modules to build DPM model and to compute DPM score are used for verifying the existence of object in each shot. We propose to use neural network to estimate a *query-adaptive weight* to combine scores calculated by BOW and DPM to form new score for final re-ranking step.

BOW for large scale video dataset retrieval. We divide the baseline process using BOW on video data into three main steps. First, from raw videos, we extract keyframes at the rate of 5 frames per second. Hessian-Affine detector and RootSIFT descriptor are used to detect and describe local features of each frame. Second, a vocabulary of 1 million codewords is constructed from previous feature collection. We use Approximate K-Means (AKM) algorithm to save training

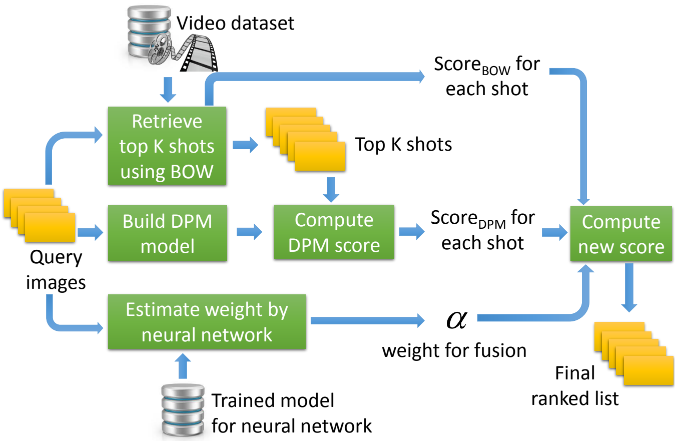

Video dataset
Score$_{BOW}$ for each shot
Retrieve top K shots using BOW
Top K shots
Query images
Build DPM model
Compute DPM score
Score$_{DPM}$ for each shot
Compute new score
Estimate weight by neural network
α
weight for fusion
Final ranked list
Trained model for neural network

Fig. 2. Proposed instance search system

time. To reduce quantization error without rising storage memory, only features of query images are soft-assigned with 3 nearest neighbors. High-dimensional histogram vector is built by aggregating all frames in each shot. We use average pooling in this aggregation step. We construct inverted index as a data structure to store retrieved shots.

To be specific, let L is the size of the codebook, n is number of keyframes of a shot, $Q_k, F_j \in R^L$ are the BOW vectors of k-th query image and j-th frame of the video shot, we get:

$$F = \frac{1}{n} \sum_{j=1}^{n} F_j \tag{1}$$

where F is a high dimensional vector represented in each shot.

Third, we apply asymmetrical metric to verify query images against shots extracted from videos. The relevance score between a query topic which contains all query images and shots is calculated by using average fusion of ranking lists returned from each query topic:

$$S_{BOW} = \frac{1}{m} \sum_{k=1}^{m} asym(Q_k, F_j) \tag{2}$$

where m is the number of query images of a query object and $asym$ is an asymmetrical similarity score [14]. K shots with highest S_{BOW} are used for the late fusion stage.

Object existence verification with DPM and Adaptive fusion of BOW and DPM scores. BOW method is created by building vocabulary of visual words so it is useful to detect objects having many textures. Furthermore, BOW is also a therapy for treating large object. In the case with small or less textured object, DPM can be used to check the existence of query object in a frame of a video shot. By thorough seeing strengths and weaknesses in each method, instead of using just only BOW or DPM in our system separately, we combine BOW and DPM in an adaptive way to take advantage of them.

We *build DPM model* from example images of a query object and their masks. After top K shots are returned using BOW, we *compute DPM score* for each of these shots. The fusion score which is meticulously described in section IV is computed by combining BOW and DPM scores in an adaptive way. To find the optimal weight for each model, we use trained neural network which adapts to each type of query object. Based on the final score, we rerank candidate list to get final shots with highest possibility of containing the query object.

IV. Adaptive Fusion between BOW and DPM

A. Object existence verification with DPM and score fusion

By combining part filters with root filters, DPM can be used as an implicit spatial checking method to verify the existence of a query object in each shot retrieved by BOW module. Because the original version of DPM is too slow to be applied in large scale of data, many researches have tried to improve not only accuracy but also speed of DPM such as [19], [20]. A sparse representation of a HOG classifier combined with inverted index provided a near state-of-the-art detection performance while maintaining instant retrieval speed [21]. Therefore, the algorithm can be applied for object detection on large scale datasets.

To build model M of a query object, we use all m query examples as positive samples and 100 random images crawled from Google with the keyword ”things” as negative samples. We only use DPM to check the certainty that a query object may exist in frames of a shot, not the exact location of that query object. Let $score(M, I_j)$ be the highest response when checking the model M against the j-th key frame of a video shot. We define the similarity score between a query topic and a shot as the maximum score for all key frames of that shot:

$$S_{DPM} = \max_{j} score(M, I_j); \tag{3}$$

where I_j is the j-th key frame of the video shot.

Late fusion between BOW and DPM scores. For each shot retrieved by BOW module, we have two scores suggested by BOW and DPM to reflect the relevance between that shot and a given query topic. We propose the two steps to determine the final score for re-ranking shots in result as follows.

First, we normalize scores of DPM (c.f. Eq. 3) and BOW (c.f. Eq. 2) by formula below:

$$S^{*} = \frac{s - \mu}{\sigma} \tag{4}$$

where, s is the score which DPM or BOW returns, S^{*} is the corresponding normalized score, μ and σ are the mean and standard deviation of returned K highest scores respectively. This normalization has an advantage that the new score is more stable than the original. Not only is it not affected by raw value of score but it can also maintain the order between raw ones.

Second, we propose a new fusion score as a linear combination on S^{*}_{BOW} and S^{*}_{DPM}.

$$S = \alpha\ S^{*}_{BOW} + (1 - \alpha)\ S^{*}_{DPM} \tag{5}$$

where fusion weight $0 \leq \alpha \leq 1$, S is the final fusion score.

This formula can be considered as the general case that can cover hard-assigned situations which rely on only BOW ($\alpha = 1$), only DPM ($\alpha = 0$), or average fusion ($\alpha = 0.5$). As each query object has different properties, weights of BOW and DPM normalize scores must be adaptive. For large query objects with rich textures, BOW score is more reliable than DPM one. Meanwhile, DPM score should has higher weight to handle query objects with small size of less textures.

B. Adaptive weight for score fusion with neural network

Instead of using a fixed value of weight for late fusion, similar to average fusion scheme by Crowley et al.[18], we take a further step to learn a model to estimate an adaptive weight for fusion in final re-ranking step. The fusion weight α should be chosen adapting to the characteristics of a query object based on its example images.

Our key idea is that from a collection of optimum fusion weights corresponding to various query objects, we can train the model to estimate a good fusion weight from the characteristics of an arbitrary query object. In this paper, we use neural network to realize this idea to estimate adaptive weight for late fusion between BOW and DPM scores.

From observation, as a BOW system usually exploits sparse features based on keypoint detectors, it may not be appropriate to deal with objects with small size or less textures. In such cases, we should rely more on DPM score to evaluate the final fusion score. Therefore, we propose 6 properties in 2 groups to learn from all examples of a query object: its size and complexity of textures (c.f. Table II). We choose the average ratio of object area to image area S_{object}/S_{image} to represent the size property of a query object. To determine whether a query is rich-textured, we use the average number of keypoints inside query mask $\mathrm{Mean}N_{key}$. As keypoints are not always give discriminative information on a query object, we propose to use the number of shared visual words between two query examples as a new property. Let N_{shared} be the number of shared words from one example to $m-1$ others. There are $m(m-1)/2$ pairs from m query examples. The average and maximum values are denoted by $\mathrm{Mean}N_{shared}$ and $\mathrm{Max}N_{shared}$. We also use the average value of ratios $\mathrm{Mean}N_{shared}/N_{key}$ and $\mathrm{Max}N_{shared}/N_{key}$ as properties of a query object.

We use a neural network with 3 layers: input, hidden, and output layer. The number of neurons in the input layer equals to the number of properties extracted from m query examples of a topic. To learn the best model to predict adaptive fusion weight, we conduct experiments with different configurations of neural networks. For the input layer, we consider different subsets of the 6 properties of a query object, each subset has at least 2 properties. The hidden layer has 3 to 10 nodes. There is only one output node to represent the adaptive fusion weight.

To train the neural network model, we manually create a dataset of good examples of fusion weights corresponding to different query objects in INS2013 and INS2014. Then we evaluate the mAP of all queries in INS2013 and INS2014 using adaptive fusion weights estimated from the model to select the best configuration of neural network and learnt model.

V. Experiments and result

We conduct two experiments to evaluate the advantages of our proposed instance search system with BOW and DPM using adaptive fusion mechanism. First, we compare the results between the two baseline instance search systems with a naive fusion method to demonstrate the superiority of BOW's and DPM's complementary characteristics. In each of the two baseline systems, shots in the final ranked list are sorted based on either BOW score or DPM score only. Second, we compare the results of our system using adaptive weighting fusion technique with other state-of-the-art methods such as Hamming Embeding and Weak Geometric Consistency [10], topology checking [9], and Multi-features fusion [22].

A. Dataset and Queries

In our experiments, we conduct experiments on the dataset used in TRECVID Instance Search competitions in 2013 and 2014. This annual competition in object detection in video, sponsored by National Institute of Standards and Technology (NIST), is challenging because of the variety of size, illumination, occlusion, etc. of query objects. The dataset consists of 244 video files extracted from BBC EastEnders program with 464 hours in duration and 300GB in storage. There are total 60 queries in the two competitions in 2013 and 2014, covering various kinds of objects. For each query object, there are $m=4$ example images, their corresponding mask images, and groundtruth. We use top $K=1000$ shots returned from BOW model and average mean precision (mAP) as a standard score for evaluation.

B. Late fusion with BOW and DPM

Figure 3 shows the mAP of different instance search methods on 30 queries in INS2013 and 30 queries in INS2014. The two baseline methods, BOW-only and DPM-only, are simply the two specific cases of our score fusion technique with $\alpha=1$ and $\alpha=0$ respectively.

The baseline BOW method achieves higher average precision than DPM method in some queries, such as query 9070 or 9101, while DPM method provides better result in other queries, such as 9073 or 9108. We can see that even our average late fusion between BOW and DPM (with $\alpha=0.5$) can provide the average precision higher or at least equal to the better precision achieved by either BOW or DPM baseline methods. For example, we get 63.60% in precision for query 9103 while BOW and DPM can only get 51.15% and 44.86% respectively. Our system gets higher result than both BOW and DPM in nearly all queries (25/30 queries in INS2013 and 23/30 queries in INS2014). Table I shows that our average fusion method has better mAP over two baseline systems.

TABLE I
Comparison between hard fusion methods

Method	mAP	
	INS2013	**INS2014**
Baseline DPM ($\alpha=0$)	19.55	21.23
Baseline BOW ($\alpha=1$)	27.91	25.01
Average fusion ($\alpha=0.5$)	32.18	28.21

C. Adaptive late fusion between BOW and DPM

Train neural network model to estimate adaptive weight. We manually prepare data to train the neural network model to estimate the adaptive weight α for linear combination of scores in late fusion. For each of the 60 query topics of INS2013 and INS2014, we find the optimum value of α by exhaustive search. Furthermore, for each query topic, we also randomly find 4 other "nearly-optimum" values of α, i.e. the value of α that can leads to the average precision of less than 5% difference with maximum average precision achieved with the optimum weight. By this way, we create a wide variation for training samples. We randomly choose 60% of query topics for training, 20% for validation, and keep 20% for independent evaluation. In each training sample, the inputs consist of properties calculated from $m=4$ query examples

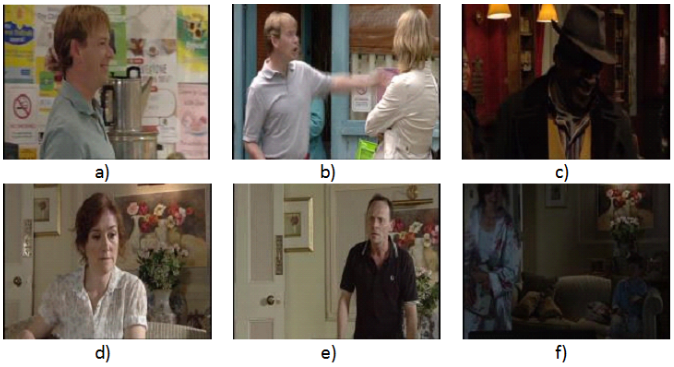


Fig. 4. Examples of combining BOW and DPM together to get better result: a - no smoking logo (Query 9069, INS2013), d - large vase (Query 9102, INS2014): query images; b, e: result scenes with occluded object; c, f: result scenes with low illuminated object

of a query topic and the target value is an optimum/nearly-optimum value of the weight.

We train different neural network models with different configurations. For each configuration, there are 2 to 6 input nodes, 3 to 10 hidden nodes, and a single output node representing the estimated weight α for late fusion. After training a neural network model, we use it to estimate the adaptive weight α for each query object in INS2013 and INS2014 and calculate the mAP. Table II illustrates the mAP on INS2013 and INS2014 using models trained with different configurations. The last row in the table is the best configuration with 3 input nodes (area ratio, $MeanN_{shared}/N_{key}$, and $MaxN_{shared}/N_{key}$) and 5 hidden nodes. With this configuration, the mAP is 33.07% and 25.80% on INS2013 and INS2014 respectively, higher than the average fusion.

Comparison with state-of-the-art methods. We compare our system with state of the art systems which have been reported on INS2013 and INS2014. The best method in TRECVID INS 2013 competition, named Multi-features, uses late fusion technique and asymmetrical similarity to combine six pairs of feature detectors and descriptors [22]. However, it only gets 31.33% in mAP while our methods get up to 33.07%. Topology checking (TC) approach uses Delaunay triangulation to improve the quality of visual matching. Hamming Embedding and Weak Geometric Consistency (HE+WCG) votes dominant scale and orientation for a fast but weak geometric checking. However, this method still does not get rid of the problem of spatial verification mentioned in this paper. As described in table III, our method is fairly evaluated over two benchmarks. Although Multi-features and other spatial reranking methods improve the performance greatly, they still do not solve the failure of spatial verification problem. Moreover, Figure 3 shows that our method is good for not only small and texture-less objects but also big and rich-textured ones.

VI. Conclusion

TABLE III
COMPARISON OF THE PROPOSED METHOD WITH STATE-OF-THE-ART ON INS2013 AND INS2014

	mAP	
Method	**INS2013**	**INS2014**
Average fusion ($\alpha = 0.5$)	32.18	28.21
Multi-features	31.33	28.77
HE+WGC	26.51	24.34
TC	20.50	N/A
Ours (Adaptive fusion)	**33.07**	**28.67**

In this paper, we address about the problem of BOW when applying on multiple types of query object dataset. A new hybrid adaptive fusion technique based on characteristics of query is introduced to make retrieval system be more stable with various kinds of object. Instead of using simple average fusion, we propose a pre-trained neural network to find the optimal weight of these models. The input of the network depends on intrinsic query characteristics such as: area, number of interest points, discriminative feature. Experimental results show the primacy of our system compared to other state-of-the-art systems. Moreover, our scheme enables us to replace DPM by any other object detectors without changing the structure of the system to get better result.

Acknowledgment

This research is funded by Vietnam National University HoChiMinh City (VNU-HCM) under grant number C2015-26-01.

References

[1] A. F. Smeaton, P. Over, and W. Kraaij, "Evaluation campaigns and trecvid," in *Proceedings of the 8th ACM international workshop on Multimedia information retrieval*. ACM, 2006, pp. 321–330.

[2] J. Sivic and A. Zisserman, "Video google: A text retrieval approach to object matching in videos," in *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*. IEEE, 2003, pp. 1470–1477.

[3] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.

[4] K. Mikolajczyk and C. Schmid, "Scale & affine invariant interest point detectors," *International journal of computer vision*, vol. 60, no. 1, pp. 63–86, 2004.

[5] M. S. Extremal, J. Matas, O. Chum, M. Urban, and T. Pajdla, "Robust wide baseline stereo from," in *In British Machine Vision Conference*. Citeseer, 2002.

[6] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Object retrieval with large vocabularies and fast spatial matching," in *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*. IEEE, 2007, pp. 1–8.

[7] G. Tolias and Y. Avrithis, "Speeded-up, relaxed spatial matching," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 1653–1660.

[8] R. Arandjelović and A. Zisserman, "Three things everyone should know to improve object retrieval," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

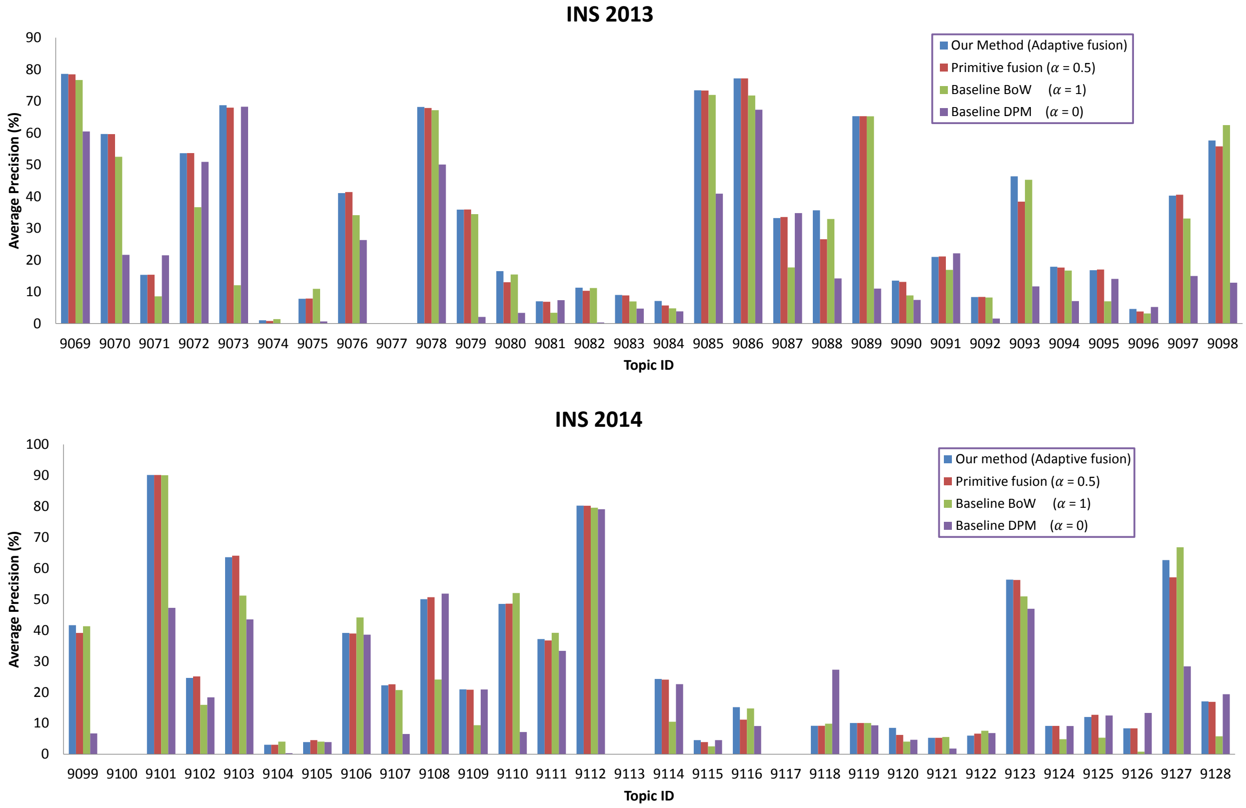
[9] W. Zhang and C.-W. Ngo, "Searching visual instances with topology checking and context modeling," in *Proceedings of the 3rd ACM conference on International conference on multimedia retrieval*. ACM, 2013, pp. 57–64.

[10] H. Jegou, M. Douze, and C. Schmid, "Hamming embedding and weak geometric consistency for large scale image search," in *Computer Vision–ECCV 2008*. Springer, 2008, pp. 304–317.

[11] P.-H. Gosselin, N. Murray, H. Jégou, and F. Perronnin, "Revisiting the fisher vector for fine-grained classification," *Pattern Recognition Letters*, vol. 49, pp. 92–98, 2014.

TABLE II
DETAILS OF INPUT UNIT IN NEURAL NETWORK

#input inits	Mean S_{object}/S_{image}	Mean N_{shared}	Max N_{shared}	Mean N_{shared}/N_{key}	Max N_{shared}/N_{key}	N_{key}	**mAP**	
							INS2013	**INS2014**
6	x	x	x	x	x	x	32.25	28.51
2		x	x				32.45	28.87
2	x	x					32.78	28.42
3	x			x	x		32.84	28.68
4	x	x	x	x			32.87	28.42
2	x		x				32.92	28.47
3	**x**	**x**	**x**				**33.07**	**28.67**


INS 2013

INS 2014

Fig. 3. Comparison between our adaptive fusion method with hard fusion, which includes baseline BOW, baseline DPM, and average fusion of BOW and DPM in INS2013 and INS2014

[12] H. Wang, A. Klaser, C. Schmid, and C.-L. Liu, "Action recognition by dense trajectories," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 2011, pp. 3169–3176.

[13] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 32, no. 9, pp. 1627–1645, 2010.

[14] C.-Z. Zhu, H. Jégou, and S. Satoh, "Query-adaptive asymmetrical dissimilarities for visual object retrieval," in *Computer Vision (ICCV), 2013 IEEE International Conference on*. IEEE, 2013, pp. 1705–1712.

[15] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Lost in quantization: Improving particular object retrieval in large scale image databases," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, 2008, pp. 1–8.

[16] Y. Cao, C. Wang, Z. Li, L. Zhang, and L. Zhang, "Spatial-bag-of-features," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2010, pp. 3352–3359.

[17] X. Shen, Z. Lin, J. Brandt, S. Avidan, and Y. Wu, "Object retrieval and localization with spatially-constrained similarity measure and k-nn re-ranking," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 3013–3020.

[18] E. J. Crowley and A. Zisserman, "The state of the art: Object retrieval in paintings using discriminative regions," in *British Machine Vision Conference*, 2014.

[19] M. A. Sadeghi and D. Forsyth, "30hz object detection with dpm v5," in *Computer Vision–ECCV 2014*. Springer, 2014, pp. 65–79.

[20] I. Kokkinos, "Bounding part scores for rapid detection with deformable part models," in *Computer Vision–ECCV 2012. Workshops and Demonstrations*. Springer, 2012, pp. 41–50.

[21] Y. Aytar and A. Zisserman, "Immediate, scalable object category detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

[22] D. D. Le, C.-Z. Zhu, L. S. Phan, S. Poullot, D. A. Duong, and S. Satoh, "National institute of informatics, japan at trecvid 2013," in *TRECVID, Orlando, Florida, USA*, 2013.