Expert Systems with Applications 42 (2015) 3991–3999

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Keybook: Unbias object recognition using keywords

Wai Lam Hoo, Chern Hong Lim, Chee Seng Chan *

Centre of Image and Signal Processing, Faculty of Computer Science & Information Technology, University of Malaya, 50603 Kuala Lumpur, Malaysia

ARTICLE INFO

Article history:
Available online 17 January 2015

Keywords:
Dataset bias
Object recognition
Codebook generation
Bag-of-Words model

ABSTRACT

The presence of bias in existing object recognition datasets is now a well-known problem in the computer vision community. In this paper, we proposed an improved codebook representation in the Bag-of-Words (BoW) approach by generating Keybook. In specific, our Keybook is composed from the keywords that significantly represent the object classes. It is extracted utilizing the concept of mutual information. The intuition is to perform feature selection by maximize the mutual information of the features between the object classes; while minimize the mutual information of the features between the domains. With this, the Keybook will not bias to any of the domain and consists of valuable keywords among the object classes. The proposed method is tested on four public datasets to evaluate the classification performance in seen and unseen datasets. Experiment results have showed the effectiveness of our proposed methods in undo the dataset bias problem.

1. Introduction

Object recognition has been an active research since decades due to the demands in many real-world applications, for example object tracking (Choi & Christensen, 2012; Serratosa, Alquézar, & Amézquita, 2012), visual surveillance (Guo, Xia, & Xiaofei, 2014; Lim, Tang, & Chan, 2014; Szpak & Tapamo, 2011), human-robot interactions (Stasse, Foissotte, & Kheddar, 2008; Rudinac, Kootstra, Kragic, & Jonker, 2012; Bielecki, Buratowski, & Śmigielski, 2013), etc. Though successful attempts, Torralba and Efros (2011) raised an important question – "*how well does a typical object detector trained on one dataset generalize when tested on a representative set of other datasets, compared with its performances on the native test set?*", or in other words, unbiased. Unfortunately, extensive experiments in Torralba and Efros (2011) found out that there is existence of various types of strong build-in bias (e.g selection bias, capture bias, and negative set bias) in popular object recognition datasets (e.g Caltech101, ImageNet, SUN09, LabelMe). These biases are embedded to these datasets unintentionally, during the data collection stage. Such situation is very critical as it causes one object representation and recognition algorithm will only work well in the specific dataset that one choose to use; and resulting in poor recognition rate when another dataset is selected. We visualize this problem in Fig. 1(a). This lead to some solutions (Duan, Xu, Tsang, & Luo, 2012; Kulis, Saenko, & Darrell, 2011; Li, Shi, Liu, Hauptmann, & Xiong, 2012; Saenko, Kulis, Fritz, & Darrell, 2010) (in this case, one dataset is treated as one domain) where the main objective in these works is to achieve good recognition rate across all the datasets as depicted in Fig. 1(b).

Khosla, Zhou, Malisiewicz, Efros, and Torralba (2012) proposed to learn a visual world SVM model to undo the bias factor. This approach learns the bias information for each dataset from a set of BoW representation in different datasets. However, there is room for improvement as the current approach is not feasible for the rapid dataset augmentation in the learning process. This is because their method needs to learn the bias weight for each of the unseen dataset to achieve better performance. That is to say, the approach will need to re-learn bias weight whenever there is a new set of images. To cope with this, our idea is to undo the bias during the codebook generation stage, and we name this as Keybook generation. From our investigation, biases exist because of the strong preference on certain information or features that cause the inclination towards a specific dataset. For an example, Fig. 2 shows a set of images of car class from two different datasets. Dataset 1 (Fig. 2(a)) has strong preference on cars with rectangle windows (representing in red boxes); while sports car with slanted front (representing in red boxes red boxes) is expected on dataset 2 (Fig. 2(b)) in order for an object to be recognized as a car. Literally, these features are not significant representation for car and such features are biased towards the corresponding training dataset. Thus, it deteriorates the recognition performance when another dataset is used in the testing stage. A better solution is to discover features for an object class that are generalized across all datasets. For example, the wheel (representing in green box) can be a choice for a car class as illustrated in Fig. 2.

* Corresponding author. Tel.: +60 3 7967 6433.
E-mail addresses: wailam88@siswa.um.edu.my (W.L. Hoo), ch_lim@siswa.um.edu.my (C.H. Lim), cs.chan@um.edu.my (C.S. Chan).

http://dx.doi.org/10.1016/j.eswa.2015.01.019
0957-4174/

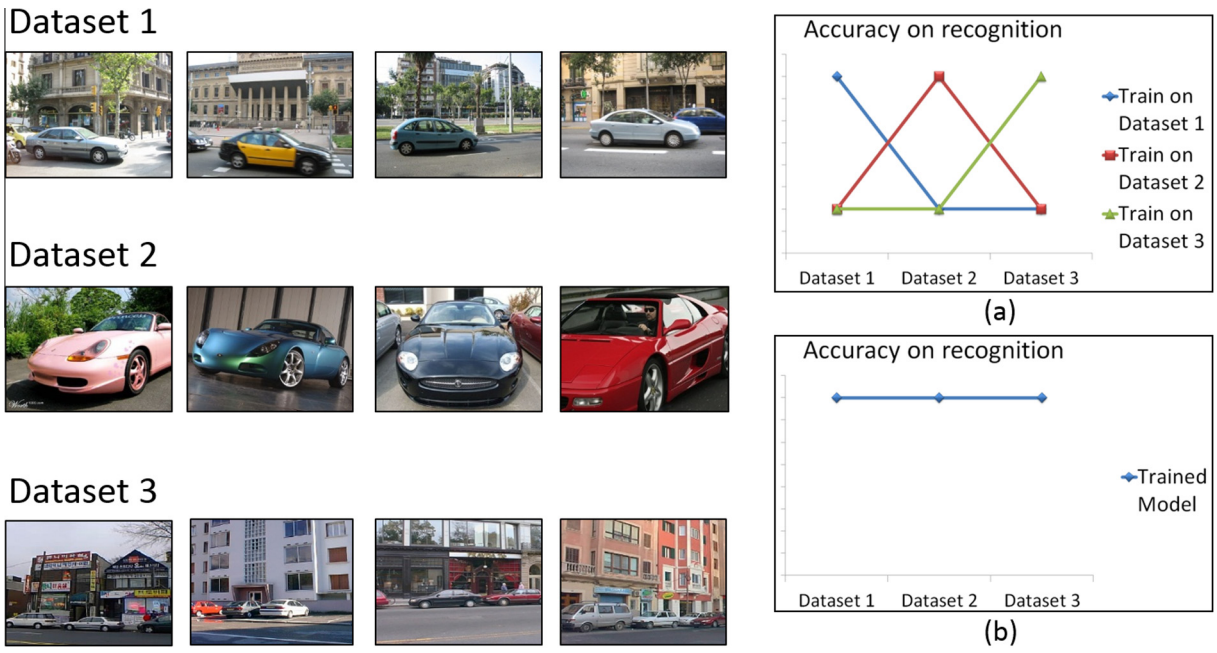

Fig. 1. (a) The recognition accuracy using a classifier that train on a specific domain of car images and then test on all other domains. Generally, it is known that a high accuracy will be obtained when the training and testing images are originated from the same domain. However, the accuracy will drop drastically when other domains are used as the testing sets because of the existence of dataset bias problem as reported in Torralba and Efros (2011). (b) The main objective of this work is to generate a trained model that are able achieve consistent accuracy when tested across all the domains.

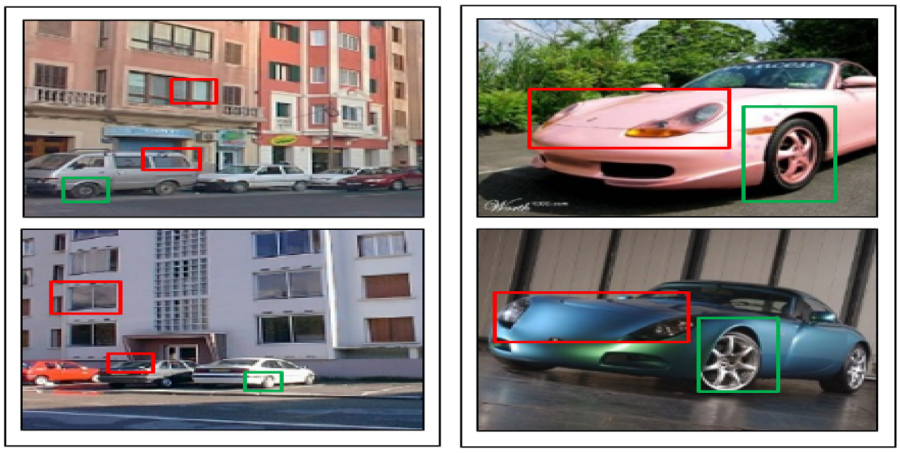

Fig. 2. Good features vs bad features. Red boxes indicate features that are bias to a particular dataset, while green boxes indicate features that share across datasets. One can notice that windows in (a) and car front in (b) are feature that biased to respective datasets. Car tyres, instead, are the common features that exist in both datasets. This image is best view in color. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

To achieve this, we extend the work in Khosla et al. (2012) by proposing Keybook generation with the aim to discover the keywords from codebook that learned from the different datasets. In general, keywords are the visual features that significantly represent the respective object classes. We utilize the mutual information (MI) to compute the semantic correlation of the visual features between the object to the classes and the datasets. According to Zitnick and Parikh (2013), MI is a good choice to discover the subset of image specific information that is semantically meaningful. In this context, the MI is a measurement between the visual features (codeword in our case) to obtain the relative degree that they are required in object classes and datasets for semantic understanding. Then, by maximizing MI between the object classes and minimize MI between the datasets, the bias towards the datasets could be undo and keywords candidate are preserved to form the Keybook. Taking example in Fig. 2 with the use of the proposed Keybook idea, the green box features that are generalized over the object class will be retained, while the red box features that are bias towards a particular dataset will be discarded.

The contribution of our work can be highlighted as follow. We propose Keybook generation to discover keywords from the conventional codebook model. The collected Keybook that is generalized over the datasets will greatly enhance the object recognition performance. For this purpose, the MI is utilized to undo the bias between the objects to the dataset. This approach is in contrast with the state-of-the-art solution (Khosla et al., 2012) where our proposed are performed in the initial training stage, which allows the framework to learn the model in an unbias manner as early as in the codebook generation stage.

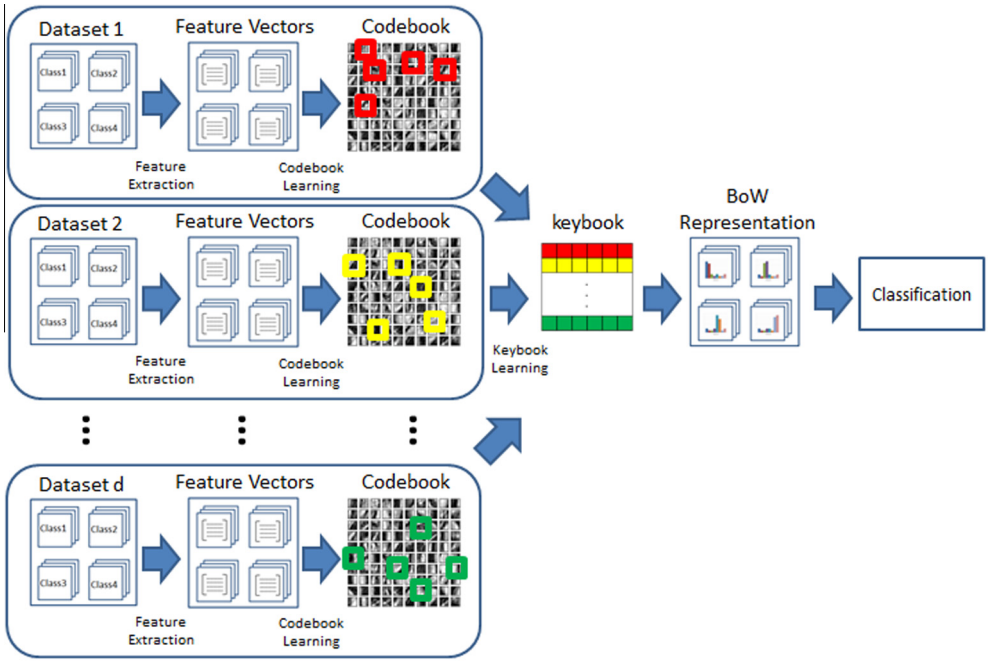

Fig. 3. Overall framework for our proposed Keybook generation. Keybook is generated via preserving the significant codewords of each object class, and remove codewords that show strong preference towards a dataset. This is make possible using the mutual information.

The rest of the paper is organized as follows. Section 2 covers the related works in domain adaptation. Our proposed framework is detailed in Section 3. Sections 4 and 5 discuss the experiments settings and results using seen and unseen dataset. Finally, we conclude our work in Section 6.

2. Related work

Detecting and recognizing objects is one of the most important research areas in computer vision community till date. However, Khosla et al. (2012) showed that conventional object recognition methods performed poorly when cross testing datasets were employed. Since then, several solutions such as transfer learning, semi-supervised learning, and unsupervised learning had been applied in this context.

To begin with, in transfer learning, the notable work was Prasath Elango, Tommasi, and Caputo (2012) whom studied learning of visual perceptual tasks such as recognizing a place or an object, taking the advantage of what the peer system had learned beforehand. However, the work is unable to handle when the system is exposed to a new environment or to deal with similar object but in a different domain, as they did not account the bias in different domains. Instead, they only transfer the discriminative model on this problem to the second system.

In semi-supervised approaches, one of the pioneer works is Daumé and Marcu (2006). The work modeled the domain adaptation problem as such there are huge labels from the source domain and scarce labels from the target domain. Then, the work formulated it in terms of a simple mixture model and applies in the context of conditional classification models and conditional linear-chain sequence labeling models. As such, inference could be efficiently solved using the conditional expectation maximization algorithm. In a more advanced work, Blitzer, Crammer, Kulesza, Pereira, and Wortman (2007) proposed to explicitly model the inherent trade-off between training on a large but inaccurate source dataset, and a small but accurate target training set.

Their approach achieve much lower target error compare to the standard empirical risk minimization method. Unfortunately in these aforesaid approaches, minimum amount of labels in target domain is still required. This is not a feasible approach in domain adaptation as the target dataset and label will not be available in the training process.

The above issue leads to unsupervised domain adaptation solutions as such approaches do not require labeled target data. Bergamo and Torresani (2010) proposed transductive SVMs while Bruzzone and Marconcini (2010) iteratively relabeling the target domain. Though successful, these works suffered from extensive computational cost as their approach depend very much on tuning parameters during the SVM training stage. A resolution for this was proposed by Gong, Shi, Sha, and Grauman (2012) which inspired from the idea of geodesic flows (Gopalan, Li, & Chellappa, 2011) to derive intermediate subspaces that interpolate between the source and target domain. Also, Gong et al. (2012) contributed in eliminating the need of tuning many parameters needed in previous works and achieve improvement in terms of computational complexity.

In what constitute closer to our work is the bias learning in different domains, for instance Khosla et al. (2012). Their idea was inspired by Torralba and Efros (2011) whom found out that there is existence of bias in popular object classification datasets. Khosla et al. (2012) exploited dataset bias during training stage, and learnt two sets of weights: (1) bias vectors associated with each individual dataset, and (2) visual world weights that are common to all datasets, which are learned by undoing the associated bias from each dataset. Their approach outperforms the SVM classification that does not account for the presence of the bias. However, the requirement of prior knowledge on the unseen domain in the their work is impractical to the real world application.

Despite the aforementioned methods achieve good accuracy, none of them considering undo the bias and collecting the significant features at the early stage (e.g during codebook generation stage). Note that, one cannot always train on augmented dataset to achieve better recognition rate. Instead, the idea of Keybook is useful to achieve better classify the unseen datasets, as well as providing a more feasibility solution for the rapid dataset augmentation. From our study, there is a surge of interest recently in understanding the semantic meaning of the image depends on the presence of objects, their features, and their relations to

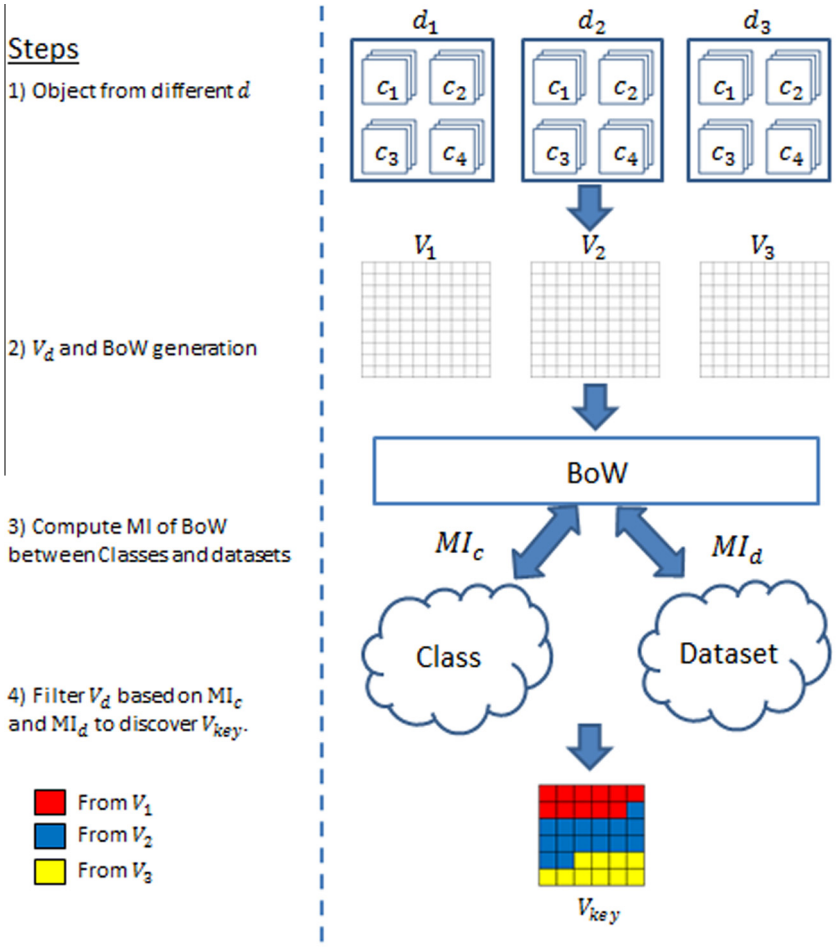

Fig. 4. Keybook generation pipeline. First, objects from different d are obtained. Then, we build V_d for each d. BoW is build based on all V_d generated. Based on the BoW, we calculate MI_c and MI_d based on class label c and dataset label d. Finally, by maximizing MI_c and minimizing MI_d, Keybook, V_{key} is generated.

the scene properties (Zitnick & Parikh, 2013). The quantitative measurement to study the semantic importance of various scene properties and features is using the mutual information (MI). Inspired from this, in our work, similar concept was employed to understand the importance of the presence of features towards class and dataset. For instance, if MI between a feature and a class is large, it indicates that the feature provides important information about an object class. On the other hand, if MI between a feature to a specific dataset is high, this show that the particular feature may be bias to that dataset. From this, one can undo the bias by maximize MI between features and object class labels; while minimize MI between features and dataset labels. With this, the significant differences between our proposed solution and those conventional solutions are firstly, our proposed solution do not require label from the target domain during the training step. This is in contrast to the semi-supervised methods. Secondly, our proposed method undo the dataset bias during the codebook generation stage. This is in contrast to the solution in Khosla et al. (2012).

3. Proposed framework

In the section, we first detail how a visual codebook is generated as illustrated in Fig. 3. Then, we elaborate our proposed method, Keybook generation that uses the MI computation. Finally, we provide the explanation on the classification method that based on Khosla et al. (2012).

3.1. Codebook generation

BoW model which is initially being employed in Natural Language Processing (NLP) has now established its contribution in the field of computer vision (Fei-Fei & Perona, 2005; Niebles, Wang, & Fei-Fei, 2008; Yang, Jin, Sukthankar, & Jurie, 2008), particularly in the area of object recognition. In general, BoW is a sparse vector of occurrence count of codewords, that is a sparse histogram over the vocabulary. In NLP, the vocabulary is a database of words such as the popular WordNet (Miller, 1995) that covered almost every words available and is online. However in computer vision, yet we had own the vocabulary of all the objects in the world, the best representation of an object is still in the discovering progress. There is still a milestone before the codebook generation which is a vital step in the BoW approach to achieve that.

To begin with the codebook generation, we represent an image $I = \{x_i, y_c, y_d\}$, where x are the extracted image features using feature extraction algorithms (for instance SIFT, DSIFT, HOG, etc.), and each x is associated with a class label y_c and domain label y_d. A codebook $V = \{w_1, \ldots, w_k\}$ is obtained by finding the cluster centers w from x_i, extracted from training images I_{train}. Conventionally, we build V_d from the training set of domain d. Then, each I is represented by the BoW model based on the obtained V via a quantization process:

$$\text{BoW}(w) = \frac{1}{n}\sum_{i=1}^{n}\begin{cases} 1, & \text{if } w = \arg\min_{w \in V}(D(w, x_i)) \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

where n is the number of features in I, and $D(w, r_a)$ is the distance between w and feature x_i.

3.2. Keybook generation

Unfortunately, conventional codebook generation induce domain bias problem, as illustrated in Fig. 2. This motivates us to propose Keybook that consists of keywords that are significant representation of the object classes. The principle of the Keybook is to discover the underlay unique features for an object class that are shared across all domains. On the other hand, we reject those features that cause bias in each specific domain. In order to achieve that, we obtain Keybook V_{key} from a set of V_d from different d by utilizing the computation of MI. In general, MI is useful in finding the correlation between two random variables A and B:

$$\text{MI}(A;B) = \sum_{a\in A}\sum_{b\in B} p(a,b)\log\left(\frac{p(a,b)}{p(a)p(b)}\right). \tag{2}$$

In our paper, A would be the BoW that is generated from a set of V_d, and B would the labels, either class label c or dataset label d.

In the proposed framework, V_{key} contains image features that share among c across d, and discard those image features that bias to d. To build V_{key}, we chose keyword $w^* \in V_{\text{key}}$ discovered from all V_d (we denote as V_{all}), by select the w that is maximizing the mutual information among the object classes, MI_c, and minimizing the mutual information of domains, MI_d:

$$w^* = \arg\max_{w\in V_{\text{all}}}(\text{MI}_c - \text{MI}_d), \tag{3}$$

where $\text{MI}_c = \text{MI}(\text{BoW}(w); y_c)$ and $\text{MI}_d = \text{MI}(\text{BoW}(w); y_d)$. Then, we build a new BoW representations based on the V_{key}, and learn a classifier from these BoW for the object recognition task. Fig. 4 shows a step-by-step process on how we obtain the Keybook.

Table 1
Experiments on seen datasets (i.e the train set of the dataset used for testing is available during training), where all four datasets are used in training and only one of the datasets is used for testing at one time. PAS, LAB, CAL, SUN refer to the four datasets, Pascal VOC2007, LabelMe, Caltech101 and SUN09, respectively. We compare both results (i.e without V_{key} and with V_{key}), where the better results are in bold. Overall, our proposed (i.e with V_{key}) outperforms in 19 out of 25 cases. The underlined results indicate that these implementations are similar to conventional classification implementation (i.e train and test on same dataset).

Train	Test	W_{PAS}	W_{LAB}	W_{Cal}	W_{SUN}	W_{vw}
Without V_{key}						
All	PAS	16.89	27.19	**33.40**	**28.19**	25.70
All	LAB	51.01	67.82	**65.68**	**69.13**	**68.24**
All	CAL	5.75	37.86	96.68	25.92	59.60
All	SUN	16.90	25.46	27.68	50.97	40.31
Average		22.63	39.68	**55.86**	43.55	48.46
With V_{key}						
All	PAS	**17.36**	**27.46**	32.67	27.96	**25.81**
All	LAB	**52.53**	**67.96**	65.16	68.72	67.64
All	CAL	**8.37**	**41.96**	**96.99**	**26.81**	**62.59**
All	SUN	**18.70**	**26.58**	**28.53**	**51.43**	**41.76**
Average		**24.24**	**40.99**	55.84	**43.73**	**49.45**

3.3. Classification

Since our proposed framework is almost similar to the conventional codebook generation method, we adopt the visual world SVM model as proposed in Khosla et al. (2012) for the classification stage. The visual world weight w_{vw} and bias vector $\triangle_d$ is taken into account and learn as below for different d:

$$\min_{w_{\text{vw}},\triangle_d,\epsilon,\phi} \frac{1}{2}\|w_{\text{vw}}\|^2 + \frac{\lambda}{2}\sum_{d=1}^{n}\|\triangle_d\|^2 + C_1\sum_{d=1}^{n}\sum_{j=1}^{s_d}\epsilon_j^d + C_2\sum_{d=1}^{n}\sum_{j=1}^{s_d}\phi_j^d \tag{4}$$

$$\text{subject to } w_d = w_{\text{vw}} + \triangle_d \tag{5}$$

$$y_j^i w_{\text{vw}}\cdot x_j^i \geqslant 1-\epsilon_j^i, \quad i=1\ldots n,\ j=1\ldots s_d \tag{6}$$

$$y_j^i w_i \cdot x_j^i \geqslant 1-\epsilon_j^i, \quad i=1\ldots n,\ j=1\ldots s_d \tag{7}$$

$$\epsilon \geqslant 0, \quad \phi_j^i \geqslant 0, \quad i=1\ldots n,\ j=1\ldots s_d \tag{8}$$

Fig. 5. Sample images of five different classes in Caltech101, LabelMe, SUN09 and Pascal VOC07.

Eqs. (5)–(8) define the constraints for the visual world SVM model, while C_1 and C_2 are the hyperparameters that employed to balance the learning terms in the objective function. This is in order to control the relative importance of the constraints. λ is used to define the weight between independent weights of different datasets, and a common weight for all datasets. ϵ represents losses across datasets when using the visual world weights w_{vw}, which need to be minimized because w_{vw} is expected to generalize across all datasets. ϕ represents the losses when test images are wrongly classified by the biased weights. Further details can be found in Khosla et al. (2012). We summarize the proposed framework in Algorithm 1.

Algorithm 1. Unbiased framework by learning Keybook (V_{key}) and visual world SVM model

Require: A set of training images with $\{x_i, y_c, y_d\}$.
Ensure: All parameters are set: total number of codewords for each dataset codebook V_d, total number of codewords for Keybook V_{key}.
1. Learn V_d for each dataset d using Eq. (1).
2. Find w^* for V_{key} using mutual information (MI) as described in Eqs. (2) and (3).
3. Learn visual world SVM model using V_{key} by optimizing Eq. (4) based on constraints in Eqs. (5)–(8).

4. Experiments

We tested our proposed framework on four different datasets that share the common object classes, but have huge variation in terms of orientation, viewpoint and environment, namely: Caltech101 (CAL) (Fei-Fei, Fergus, & Perona, 2004), LabelMe (LAB) (Russell, Torralba, Murphy, & Freeman, 2008), SUN09 (SUN) (Choi, Lim, Torralba, & Willsky, 2010), and Pascal VOC07 (VOC) (Everingham, Van Gool, Williams, Winn & Zisserman). The five object classes are 'bird', 'car', 'chair', 'dog', and 'person' and some of the example images are shown in Fig. 5.

4.1. Implementation details

We employed the experimental settings as to Khosla et al. (2012), except that we sample SIFT features on all training set using patch size = 8 and step size = 8 for simplicity and efficiency. We build V_d and V_{key} that consist of 256 codewords and keywords respectively to ensure a fair comparison. Then, we use LLC coding (Wang et al., 2010) with 3 level spatial pyramid to pool SIFT features based on V_{key}. The evaluations are tested on both seen and unseen datasets.

4.2. Testing on seen dataset

Table 1 shows the experimental results when we trained the model using training data from all four datasets, and test on one dataset at a time. Since the training data consist of images from the same dataset for the testing data, we consider this as seen dataset classification task. Table 1 (top: Without V_{key}) shows the baseline approach based on visual world SVM model (Khosla et al., 2012) without V_{key} implementation, while Table 1 (bottom: With V_{key}) shows our proposed method that combines both benefits from V_{key} and the visual world SVM model.

In general, the visual world SVM model that use V_{key} shows around 1% of mean average precision (mAP) improvement compared to the model that use conventional codebook. This shows that V_{key} enhance the model for cross-domain classification task. This is intuitively possible as Keybook attempts to find a more general representation of the objects among the same classes. Therefore it discards visual codewords that are biased to a particular dataset. Takes note that, the usage of more simpler feature space in feature representation will cause some variations to the results published in Khosla et al. (2012). However, this is a fair comparison as both methods used the same features during testing. Besides that, the underlined results indicate results using the conventional SVM model, where w_d is used to classify d, instead of w_{vw}. Comparing both results using the conventional SVM model, our proposed

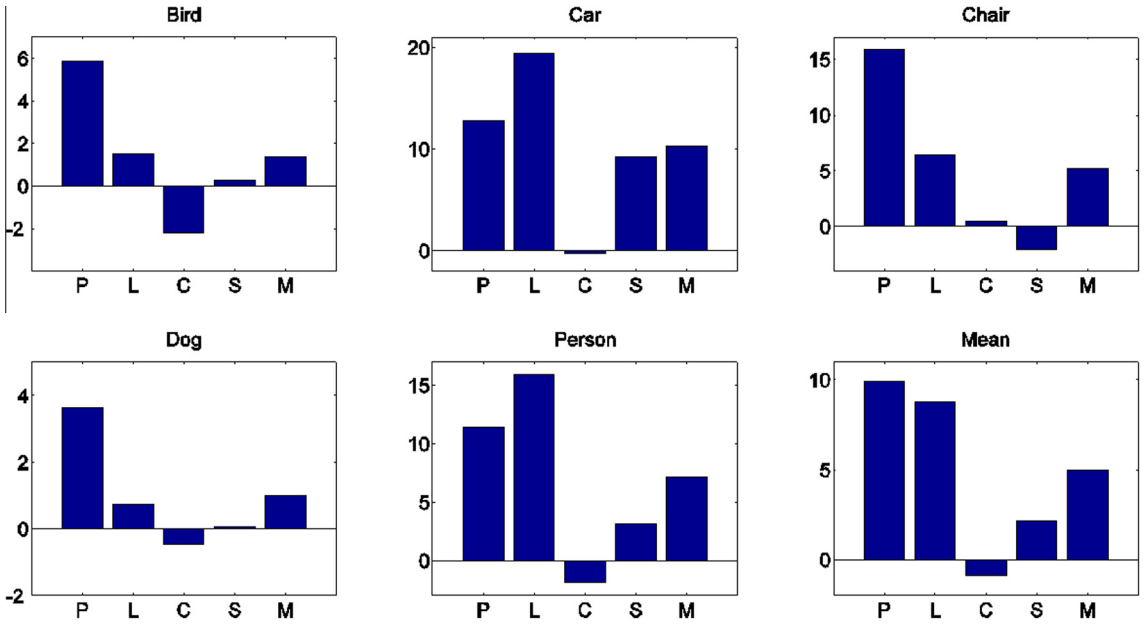

Fig. 6. The graphs show improvement in percent average precision (%AP) of classification on unseen datasets using the proposed method over the baseline approach (Khosla et al. (2012)) in 4 dataset. The labels on the x-axis 'P', 'L', 'C', and 'S' represent the datasets Pascal VOC2007, LabelMe, Caltech101 and SUN09 respectively, while 'M' represents the mean AP (mAP) increment over all datasets. Y-axis is the %AP increment over the baseline Khosla et al. (2012). Overall, our algorithm outperforms the baseline in 20 out of 25 cases, with an overall improvement of 5% mAP.

(a) No V_{key}

(b) With V_{key}

Fig. 7. Qualitative analysis on experiments that classify unseen datasets. Blue region indicates correct classification, and red means false classification. For (a) and (b), first row: Pascal VOC07, second row: SUN09, third row: LabelMe, fourth row: Caltech101. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

method achieved 0.35 mAP improvement on all four datasets compared to those methods without V_{key}.

4.3. Testing on unseen dataset

We tested our proposed method on unseen dataset, which indicates that for each testing iteration, we only select three datasets instead of four for training, and perform testing on the dataset that was leave out. For example, if we train on Caltech101, LabelMe and SUN09 datasets, then we will test on PASCAL VOC07 dataset. In here, the proposed method employs w_{vw} during the classification stage. Fig. 6 shows most of the average precision (AP) for unseen dataset is increased, except for Caltech101 dataset. This is because from Khosla et al. (2012), Caltech101 has the best AP improvement, while the Keybook generation obtained a more generalized model across the dataset. This generalized model might had slightly affected the performance in Caltech101. This also implies the bottleneck of the unbias algorithm, which further improve classification result would be hard, by comparable results could be maintained. In spite of that, we have significant improvement over 5% mAP in the overall performance.

5. Discussion

Fig. 7 shows testing images that are correctly classified and miss-classified in our proposed method for unseen dataset. From our visual inspection, one can notice that the correct classified images are those cars that the car wheels are visible. For the miss-classified images, the car wheels are either occluded or missing. This reflects our claim in the proposed method, where the car wheels could be the keywords found in the V_{key}, and we gain improve classification result in the car class when the car wheels

are visible. In contrast, our model will work poorly if the keywords (e.g. wheels) are occluded or missing. Besides, with the help of V_{key}, the proposed method could correct images that wrongly classified in the system that without V_{key}. Note that this observation is somehow not applicable for Caltech101 samples. This is maybe due to the characteristics of Caltech101 samples posses strong capture bias problem, as indicated in Torralba and Efros (2011).

6. Conclusion

Domain adaptation has risen as an important issue in object recognition research and the recent trend towards the solution has been focused on the study of the domain bias. Failing to handle the domain bias problem will deteriorate the overall object recognition system performance, especially when the trained model is used in different unseen domains. This is due to the trained model is tailored to learn for a specific domain. In this paper, we proposed a framework that undo the domain bias thru finding the most significant features of an object class which are generalized across all the domains. Specifically, we employed mutual information (MI) computation to analyse the correlation between the visual feature of the object classes and the datasets; and generate the Keybook by keyword selection. The proposed approach was tested on four publicly available datasets and achieved better performance in either seen or unseen dataset evaluation.

Although our proposed method achieves better results in the overall performance, it is still bounded with several limitations. First, the recognition is very much depends on the significant features of the object in an image, and so an occlusion on the significant feature will deteriorate the overall performance. In conjunction, deciding the number of codewords for the Keybook remains a key issue that decides the performance of the final classification.

The proposed framework will be very useful in application such as visual surveillance, ambient assisted living, and robotic vision that requires object recognition. Such systems that are employed to assist or protect human in their daily live are required to be able to perform consistently in different environments. With our proposed solution, since the significant features of the object class are retained, so it is able to cope with environments change.

There are a few potential future works to enhance the current approach. First, the discriminative characteristic (i.e hard assignment) in the current work can be relaxed with the integration the soft assignment codebook representation (Hoo, Kim, Pei, & Chan, 2014). Secondly, alternatives MI that might be more useful in Keybook during the generation of the dataset codebook could be investigated. Other possible research directions are performing the unbias attempt in the feature level (i.e when feature extraction from images occur), and dataset collection level (i.e how researcher could collect bias-free dataset). Finally, the work could be extended to space–time domain such as human motion analysis (Lim, Vats, & Chan, in press); or to fine-grained classification problem (Yao, Khosla, & Fei-Fei, 2011).

Acknowledgments

This research is supported by the High Impact Research MoE Grant UM.C/625/1/HIR/MoE/FCSIT/08, H-22001-00-B0008 from the Ministry of Education Malaysia.

References

Bergamo, A., & Torresani, L. (2010). Exploiting weakly-labeled web images to improve object classification: A domain adaptation approach. In *Advances in neural information processing systems (NIPS)* (pp. 181–189).

Bielecki, A., Buratowski, T., & Śmigielski, P. (2013). Recognition of two-dimensional representation of urban environment for autonomous flying agents. *Expert Systems with Applications, 40*, 3623–3633.

Blitzer, J., Crammer, K., Kulesza, A., Pereira, F., Wortman, J. (2007). Learning bounds for domain adaptation. In *Advances in neural information processing systems (NIPS)* (Vol. 20, pp. 129–136).

Bruzzone, L., & Marconcini, M. (2010). Domain adaptation problems: A dasvm classification technique and a circular validation strategy. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 32*, 770–787.

Choi, C., & Christensen, H. I. (2012). 3d textureless object detection and tracking: An edge-based approach. In *IEEE/RSJ international conference on intelligent robots and systems (IROS)* (pp. 3877–3884).

Choi, M. J., Lim, J. J., Torralba, A., & Willsky, A. S. (2010). Exploiting hierarchical context on a large database of object categories. In *IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 129–136).

Daumé, H., III, & Marcu, D. (2006). Domain adaptation for statistical classifiers. *Journal of Artificial Intelligence Research, 26*, 101–126.

Duan, L., Xu, D., Tsang, I. W.-H., & Luo, J. (2012). Visual event recognition in videos by learning from web data. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 34*, 1667–1680.

Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J., & Zisserman, A. The PASCAL visual object classes challenge 2007 (VOC2007) results. <http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html>.

Fei-Fei, L., & Perona, P. (2005). A bayesian hierarchical model for learning natural scene categories. In *IEEE conference on computer vision and pattern recognition (CVPR)* (Vol. 2, pp. 524–531).

Fei-Fei, L., Fergus, R., & Perona, P. (2004). Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *IEEE computer vision and pattern recognition workshop (CVPRW)* (pp. 178–178).

Gong, B., Shi, Y., Sha, F., & Grauman, K. (2012). Geodesic flow kernel for unsupervised domain adaptation. In *IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 2066–2073).

Gopalan, R., Li, R., & Chellappa, R. (2011). Domain adaptation for object recognition: An unsupervised approach. In *IEEE international conference on computer vision (ICCV)* (pp. 999–1006).

Guo, W., Xia, X., & Xiaofei, W. (2014). A remote sensing ship recognition method based on dynamic probability generative model. *Expert Systems with Applications, 41*, 6446–6458.

Hoo, W. L., Kim, T., Pei, Y., & Chan, C. S. (2014). Enhanced random forest with image/patch-level learning for image understanding. In *22nd international conference on pattern recognition, ICPR* (pp. 3434–3439).

Khosla, A., Zhou, T., Malisiewicz, T., Efros, A., & Torralba, A. (2012). Undoing the damage of dataset bias. In *European conference on computer vision (ECCV)* (pp. 158–171).

Kulis, B., Saenko, K., & Darrell, T. (2011). What you saw is not what you get: Domain adaptation using asymmetric kernel transforms. In *IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 1785–1792).

Lim, C. H., Vats, E., Chan, C. S. (in press). Fuzzy human motion analysis: A review. Pattern Recognition. http://dx.doi.org/10.1016/j.patcog.2014.11.016.

Lim, M. K., Tang, S., & Chan, C. S. (2014). iSurveillance: Intelligent framework for multiple events detection in surveillance videos. *Expert Systems with Applications, 41*, 4704–4715.

Li, H., Shi, Y., Liu, Y., Hauptmann, A. G., & Xiong, Z. (2012). Cross-domain video concept detection: A joint discriminative and generative active learning approach. *Expert Systems with Applications, 39*, 12220–12228.

Miller, G. A. (1995). Wordnet: A lexical database for english. *Communications of the ACM, 38*, 39–41.

Niebles, J. C., Wang, H., & Fei-Fei, L. (2008). Unsupervised learning of human action categories using spatial–temporal words. *International Journal of Computer Vision, 79*, 299–318.

Prasath Elango, S., Tommasi, T., & Caputo, B. (2012). Transfer learning of visual concepts across robots: A discriminative approach. Idiap-RR Idiap-RR-06-2012 Idiap.

Rudinac, M., Kootstra, G., Kragic, D., & Jonker, P. P. (2012). Learning and recognition of objects inspired by early cognition. In *IEEE/RSJ international conference on intelligent robots and systems (IROS)* (pp. 4177–4184).

Russell, B. C., Torralba, A., Murphy, K. P., & Freeman, W. T. (2008). Labelme: A database and web-based tool for image annotation. *International Journal of Computer Vision, 77*, 157–173.

Saenko, K., Kulis, B., Fritz, M., Darrell, T. (2010). Adapting visual category models to new domains. In *European conference on computer vision (ECCV)* (pp. 213–226).

Serratosa, F., Alquézar, R., & Amézquita, N. (2012). A probabilistic integrated object recognition and tracking framework. *Expert Systems with Applications, 39*, 7302–7318.

Stasse, O., Foissotte, T., & Kheddar, A. (2008). Treasure hunting for humanoids robot. In *IEEE RAS/RSJ international conference on humanoids robots, workshop on cognitive humanoid vision*.

Szpak, Z. L., & Tapamo, J. R. (2011). Maritime surveillance: Tracking ships inside a dynamic background using a fast level-set. *Expert Systems with Applications, 38*, 6669–6680.

Torralba, A., & Efros, A. A. (2011). Unbiased look at dataset bias. In *IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 1521–1528).

Wang, J., Yang, J., Yu, K., Lv, F., Huang, T., & Gong, Y. (2010). Locality-constrained linear coding for image classification. In *IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 3360–3367).

Yang, L., Jin, R., Sukthankar, R., Jurie, F. (2008). Unifying discriminative visual codebook generation with classifier training for object category recognition. In *IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 1–8).

Yao, B., Khosla, A., & Fei-Fei, L. (2011). Combining randomization and discrimination for fine-grained image categorization. In *IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 1577–1584).

Zitnick, C. L., & Parikh, D. (2013). Bringing semantics into focus using visual abstraction. In *IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 3009–3016).