Keypoints and Codewords Selection for Efficient Bag-of-Features Representation

Veerapathirapillai Vinoharan
Computer Centre
University of Jaffna
Jaffna, Sri Lanka
Email: vvinoharan@univ.jfn.ac.lk

Amirthalingam Ramanan
Department of Computer Science
University of Jaffna
Jaffna, Sri Lanka
Email: a.ramanan@univ.jfn.ac.lk

***Abstract*—Bag-of-features (BoF) representation is one of the most popular image representations, that is used in visual object classification, owing to its simplicity and good performance. However, the BoF representation always faces the difficulty of curse of dimensionality that leads to huge computational cost and increased storage requirement. To create a discriminative and compact BoF representation, it is desired to eliminate ambiguous features before the construction of visual codebook and to select the informative codewords from the constructed codebook. In this paper, we propose a two-staged approach to create a discriminative and compact BoF representation for object recognition. In the first step, we eliminate ambiguous patch-based descriptors using an entropy-based filtering approach to retain high-quality descriptors. In the subsequent step, we select the informative codewords based on statistical measures. We have tested the proposed technique on Xerox7, UIUC texture, PASCAL VOC 2007 and Caltech101 benchmark datasets. Testing results show that more training features and/or a high-dimensional codebook do not contribute significantly to increase the performance of classification but it increases the overall model complexity and computational cost. The proposed preprocessing step of descriptor selection increases the discriminative power of a codebook, whereas the post-processing step of codeword selection maintains the codebook to be more compact. The proposed framework would help to optimise BoF representation to be efficient with steady performance.**

Keywords—keypoint selection; codebook; codeword selection; image representation; Bag-of-features

I. Introduction

The bag-of-features (BoF) approach [3], [15], [17], [18], [20] is a well known technique for representing the image content and has proved state-of-the-art performance in large scale evaluations. In the BoF approach, features are usually based on the utilisation of tokenising keypoint-based features, e.g., scale-invariant feature transform (SIFT) [2], to generate a codebook. The BoF representation of an image conveys the presence or absence of the information for each visual word in the image. In a BoF framework, the codebook plays a crucial role. An important issue of the codebook representation is its discriminative power and compactness. The size of a codebook controls the complexity of the codebook model and the discriminative power of a codebook determines the quality of the model. The number of features extracted from training images to construct a codebook and the dimensionality of a codebook causes two sets of problems: 1) the computational cost during the vector quantisation step is high and some of the detected features are not helpful for better classification; and 2) the model complexity is high that may overfit to the distribution of codewords in an image. The increase in the number of object categories, increases the computational cost and it makes the classification of histograms challenging due to its diverse range in object classes.

Most of the object recognition tasks that are reported in the literature have employed sufficiently large-sized codebook at the order of 1000 to 10000, typically resulting in hyper-dimensional and sparse histogram representations. The use of such large-sized codebook will in turn make each BoF vector to require huge storage space and the efficiency of computation in large scale datasets will yield to the well-known "curse-of-dimensionality". Therefore, the discriminative power and compactness of a codebook are important to control the complexity of the model. A straightforward way to create compact codebooks is to reduce the dimensionality, that will quickly weaken the discriminative power and degrade classification performance. Simply selecting most discriminative codewords or linearly combining the bins will not work well either [12]. In this regard we formulate and contribute the following:

- Choose unambiguous patch-based descriptors prior to the construction of a codebook in order to reduce the features causing false positives in object classification. In this regard we present an entropy-based filtering approach to eliminate ambiguous patch-based descriptors (e.g., SIFT).
- Select the best subset of codewords from an initially constructed codebook to enhance the discriminative power of the codebook and make it more compact. To achieve this we present an inter-category and intra-category confidences to select the informative codewords that generates a discriminative and compact codebook for the BoF representation.

The proposed method provides an effective way to improve the object categorisation performance when using the BoF model with very low dimensional representation.

The rest of this paper is structured as follows: Section II briefly describes the background needed for our work. Section III summarises related work that has been used to construct a discriminative and compact codebook for object recognition. Section IV explains the proposed methodology in detail. Section V describes the testing results. Finally, Section VI concludes this paper.

II. Background

A. Bag-of-Features Approach

The BoF approach is widely used in image scene classification [5] and object recognition tasks [11] in computer vision. The BoF-based object recognition systems fit into a general framework as summarised below:

1) Detecting and describing of image patches from the training and testing image sets.
2) Constructing a visual codebook by performing cluster analysis on the descriptors extracted from the training set. The codebook is the set of codewords.
3) Mapping the extracted image patches from the training and testing image sets into a feature vector (i.e., BoF) by computing the frequency histograms with the codewords.
4) Classifying the test images to predict which object category or categories to assign to the image.

There are two broad categories of codebook models: Global and object-specific codebooks. A global codebook is category independent but may suffer in its discriminative power. On the other hand a object-specific codebook may be too responsive to noise. Thus, the construction of a codebook plays a crucial role that affects the models' complexity.

B. Scale-Invariant Feature Transform

SIFT is a technique [2] that extracts distinctive features from gray-level images, by filtering images at various scales and patches of interest that have sharp intensity changes. SIFT descriptors is a 128 dimensional vector that can be used in the context of recognition and matching of the same scene or object observed under different viewing conditions. SIFT for colour image is also available [7].

C. Resource-Allocating Codebook

RAC [9] is a simple and extremely fast technique to construct visual codebooks using a one-pass setup that carves the feature space as fixed-size hyperspheres. RAC yields a better discriminative codebook with a drastic reduction in computational needs. Codebook constructed by RAC shows similar recognition performance to K-means method-based codebook with small variations [19]. RAC algorithm is summarised in the following steps:

Step 1: RAC starts by arbitrarily assigning a descriptor of an image as a first entry in the codebook to be the informative codeword.

Step 2: A subsequent descriptor is processed. The smallest distance to all entries in the present codebook is computed using Euclidean distance. If this distance exceeds the predefined hyper-parameter r of RAC:

- Then the current descriptor is recorded as an additional informative codeword that creates a new codeword in the codebook (i.e., the codebook in Step 1 is updated).
- Else no action is taken in respect of the processed descriptor.

Step 3: This process is continued until all or desired number of descriptors are processed only once.

D. Support Vector Machine

SVM [1] is a statistical learning method that has showed better performance in visual object classification problems. The objective of SVM learning is to find a hyperplane that maximises the inter-class margin of the training data. The data in the input space are projected into a high-dimensional feature space by kernel function. Multiclass classification in this paper is performed using SVMs of linear kernel trained with the one-versus-all (OVA) rule. OVA-SVMs learn to separate every category from the remaining object categories, and it allocate the category label of a test image having the highest response.

III. Related Work

In the literature of the BoF approach for object classification there exists several approaches that have focused on the discriminative power and/or compactness of codebooks.

In [14], the authors have proposed a two-step approach to map an initially constructed large codebook into a compact codebook with stable performance by maintaining its discriminative power in object recognition. Using an initial large codebook (K=1000 of K-means), training images are represented using a mapping rule that maps the importance of each codeword within an image as visual-bits. These set of visual image bits forms a sparse representation of each codewords in respect of the object-specific training sets that is used for compression. This technique reduces the size of the codebook using binary representations of images and codewords, which enhances the efficiency of the coding while showing the discriminative power of the codebook. This is accomplished by the following two-step process: 1) encoding each image as 'bits' (i.e., the significant presence or absence of each codeword) and 2) removing the codeword that are not sufficiently activated in the images. Authors have tested their technique on four benchmark image sets: 1) MPEG7 CE Shape-1 Silhouette; 2) PASCAL VOC 2007; 3) UIUC texture; and 4) Xerox7. Authors' test results indicate that the approach slightly surpasses the codebooks learned by K-means by having just half the size of the initial codebook with stable performance.

In [13], the authors have proposed an unsupervised dimensionality reduction framework for discriminative and compact BoF representation. First, they construct the dissimilarity matrix between each pair of histograms and then perform multidimensional scale technique to obtain a small Euclidean embedding of the original BoF while preserving the inherent neighbourhood structure. Authors' experimental results show that a very small dimension is sufficient for learning tasks using BoF or spatial pyramid matching without losing the precision of the classification. It has been claimed that a compact representation of BoFs can improve the accuracy of the classification in relation to the traditional BoF approach. Authors have tested their technique on three benchmark image sets: subset of PASCAL VOC 2012, subset of Caltech101, Scene15 and INRIA holidays. Testing results show that the authors method has shown promising results for the image classification and retrieval tasks with very low-dimensional representation. In Caltech101 and PASCAL VOC 2012 datasets, the classification accuracies of the original 1000-dimensional and 2000-dimensional BoF vectors are 54.23% and 54.15% respectively, whereas the proposed approach in both cases achieves better accuracy about 58% for reduced dimensional BoF vectors of 100. The classification accuracies of the original 2100-dimensional and 4200-dimensional Spatial Pyramid Matching (SPM) vectors are 55.45% and 54.78% respectively, whereas the proposed approach in both cases achieves better accuracy about 57% for reduced dimensional BoF vectors of 100. In Scene15 dataset, the standard 2000-dimensional BoF vectors classification accuracy is 70.82%, and SPM 2100-dimensional is 72.2%. Better classification rate of 75.54% for 2000-dimensional BoF vectors was obtained with 35 dimensional BoF vectors.

In [16], the authors propose an iterative keypoint selection (IKS) technique to create discriminative BoF representation by selecting most appropriate keypoints which reduces the computational cost in constructing a codebook and leads to a better discriminative codebook. There are two steps involved in each IKS: 1) Representative keypoints are randomly selected or taken from cluster centriods; and 2) the distance between identified keypoints and the selected informative keypoints are calculated and if the distance is less than a predefined threshold then those keypoints are discarded. This process iterates until no unrepresentative keypoints are found. To execute the initial stage of IKS, two particular approaches are used: 1) Identifying informative keypoints based on random selection; and 2) using K-means algorithm to select K centroids as informative keypoints. The former approach requires larger computational cost, whereas the latter approach not only reduces the computational time but also provides increased classification rate. Experiments using the Caltech101, Caltech256 and PASCAL VOC 2007 datasets demonstrate that using keypoint selection to generate both BoF and spatial pyramid matching allows the SVM classifier to produce better classification results compared to the previous techniques that use without the keypoint selection method.

IV. Methodology

A compact codebook has advantages in terms of computing efficiency and storage requirement. For example, when SVMs are used to classify feature histograms, the complexity of calculating the kernel matrix, storing the support vectors, and testing a new image are all proportional to the size of the codebook.

In this paper first, we extract local keypoints using SIFT algorithm from the training images. Before creating a relatively large original codebook, unambiguous keypoints are selected using an entropy-based filtering method to increase the discriminative power. Thereafter a codebook is constructed using RAC approach. Finally, indistinctive codewords are eliminated based on statistical measures to obtain a compact codebook. We assign each keypoint in images to the closest codeword and create a histogram representation for each image, which records how many times keypoints corresponding to the codeword occur in the image. We then apply SVM classification algorithm to these fixed-length feature vectors. The overall framework of the proposed method in this paper is illustrated in Fig. 1.

A. *Entropy-based Filtering for Feature Selection*

To create a codebook, set of keypoints that are detected in an image can be transformed to an intermediate representation. Some of the detected keypoints in the training images belonging to different object categories play a key role to make the intermediate representation more distinguishable. By selecting these important keypoints, not only the discriminative power of the codebook can be increased but also a compact codebook can be obtained.

The wide adoption of the visual codebook approach creates the impression that SIFT is a point feature. SIFT features best suitable for object detection are those with rich internal structure and associated with near-empty regions that are the main source of false positives: they tend to occur frequently and get easily matched against one another. We propose an entropy-based filtering approach to eliminate ambiguous SIFT descriptors in order to retain high-quality descriptors. This approach reduces the computational complexity of the clustering and increases the categorisation precision at the later stage of the BoF approach.

Let the SIFT descriptors $F = [f_1, f_2, \cdots, f_{128}]$ that are treated as 128 samples of discrete random variable in $\{0, 1, 2, \cdots, 255\}$.

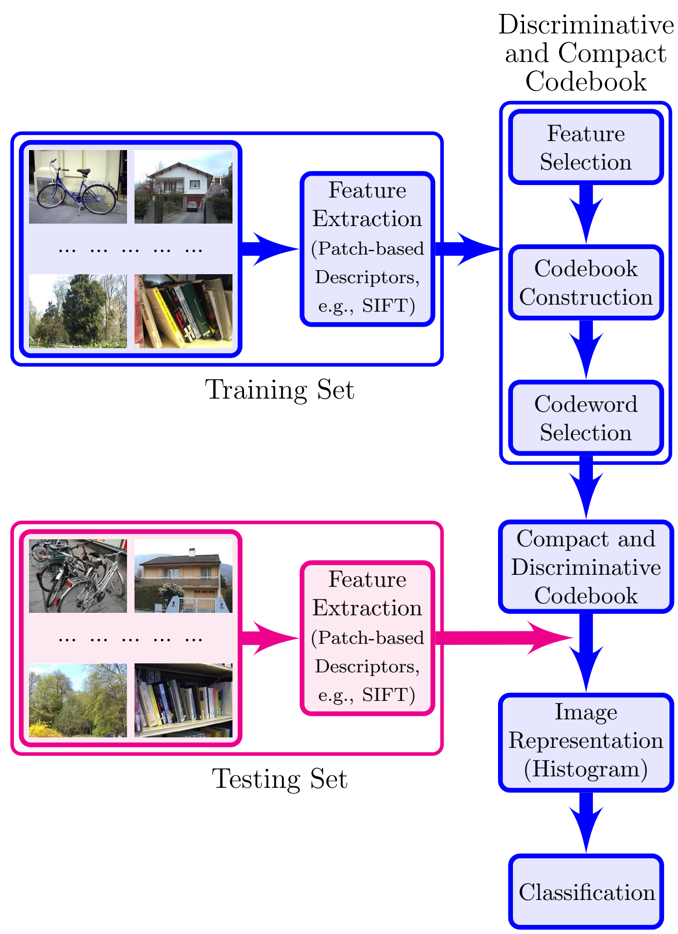

Fig. 1. Overview of the proposed framework.

Then the entropy of F is computed as,

$$E(F) = -\sum_{i=0}^{255} p_i(F) log_2 p_i(F)$$

where,

$$p_i(F) = \frac{|\{k|f_k=i\}|}{128};\ k = 0, 1, 2, \cdots, 255.$$

The values for individual dimensions of SIFT feature follow a near exponential distribution, with small values dominating the whole distribution. A SIFT value has a range of [0, 255], but almost all the values are smaller than 128 that means the range of the value is not efficiently used. Therefore the dimension of SIFT descriptors are scaled logarithmically so that the distribution will be more uniform. Note that each SIFT dimension is an 8-bit integer, so the entropy has a range of [0, 8]. In our system, we discard SIFT descriptors based on a predefined threshold which varies for different dataset.

B. Codeword Selection using Statistical Approach

A codebook is usually constructed by using a clustering algorithm. In such a codebook, the goal of codeword selection is to remove the redundancy and noise in a codebook. Elimination of indistinctive codewords not only reduces the overall computational complexity but also increases the categorisation precision.

1) Selecting Discriminative Codewords Across Categories: Object categories owing to similar histogram distribution may increase ambiguity of the classification process. Inter-category confidence is calculated by analysing category distributions of the i^{th} codeword. The inter-category confidence of the i^{th} codeword $C_{inter,i}$ is defined as follows:

$$C_{inter,i} = \sum_{j=1}^{N} max\left(\frac{f_{ij}}{n_i} - \frac{1}{m_i}, 0\right)$$

where,
K is the size of the codebook.
N is the number of object categories in classification
f_{ij} is the number of training features in the i^{th} codeword and j^{th} category, i = *1, 2,* $\cdots$, *K*., and j = *1, 2,* $\cdots$, *N*.
n_i is the total number of features in the i^{th} codeword
m_i is the number of object categories in the i^{th} codeword

The value of inter-category confidence is zero, when all the features of a codeword show a single category or equal number of features from each category in the feature domain. The value of inter-category confidence is positive when the feature ratio of a codeword shows a single category dominating other categories in the feature domain. Since the codeword only exist in histograms of the category images, the histogram distribution differs from other categories, thus the codeword enhances the categorisation result. It has been noticed by using the elimination process of indistinctive codewords many codewords tend to disappear from homogeneous regions.

In this inter-category codeword selection, the codeword is selected based on the following criteria:

- $\widehat{C}_{inter} = 0$ having a single category in the feature domain, or
- $\widehat{C}_{inter} > 20^{th} Percentile_{1 \leq i \leq K}(C_{inter,i})$

2) Selecting consistent codewords within each categories: Images of different categories may have similar histogram values of codewords that in turn will affect the classification based on the histogram. The variance of histogram value within a codeword among the same object category images is inversely proportional to the intra-category confidence.

A high variance histogram value of a codeword interrupts the classification process, i.e., it makes the classifier (e.g., SVM) difficult to classify visual object categories. Thus, low variance codewords at BoF histogram domain are stable to be classified. Based on this concept, we discard all codewords with the variance histogram value of a codeword smaller than the first quartile of C_{intra}.

The intra-category confidence of the i^{th} codeword $C_{intra,i}$ is represented as follows:

$$C_{intra,i} = \frac{1}{\sum_{j=1}^{N} var(h_{ij})}$$

where,

h_{ij} is the i^{th} codeword value of each image belonging to the j^{th} category in the BoF histogram domain,
i = *1, 2,* $\cdots$, *K*, and j = *1, 2,* $\cdots$, *N*.

3) Selecting informative codewords based on C_{inter} and C_{intra} confidences: Both confidences, C_{inter} and C_{intra}, enhance the classification process individually, and complement each other at the same time. Therefore, the combined confidence of the i^{th} codeword is shown as follows:

$$C_{com,i} = \alpha C_{inter,i} + \beta C_{intra,i}$$

where, α and β are constant values, $0 \leq \alpha, \beta \leq 1$.

Using the combined confidence, we select reliable codewords by a weighting parameter.

V. Testing Results

The proposed BoF representation scheme has been evaluated on Xerox7 [3], UIUCTex [4], PASCAL VOC 2007 [8] and Caltech101 [6] image sets that are summarised in the following:

A. Dataset

1) Xerox7: It contains 1776 images from seven categories with different resolutions. The object poses in images are highly variable and there is a significant amount of background clutter making the classification task challenging.

2) UIUCTex: It contains twenty five texture classes with 40 images per class with resolution of 640×480. The texture surfaces are of albedo variations and images have significant viewpoint changes, uncontrolled illumination, arbitrary rotations, and scale differences within each texture category.

3) PASCAL VOC 2007: Is used immensely used in large scale evaluation of object classification tasks. The dataset consists a total of 9,963 images containing 24,640 annotated objects split into training, validation, and testing sets labelled with twenty object classes.

4) Caltech101: It consists of a total of 9,146 images, split between 101 different object categories, as well as an additional background/clutter category. Each object category contains between 31 and 800 images. Common and popular categories such as faces tend to have a larger number of images than others.

B. Experimental Setup

For the image sets: Xerox7, UIUCTex, and Caltech101 we used 70% for training and 30% for testing from each class. The classification for PASCAL VOC 2007 was performed on each of the 20 classes by training the classifiers on the provided 'trainval' set and evaluating on the testing set. SIFT descriptors were extracted and global codebook was constructed by clustering the descriptors of the training images using the RAC algorithm with r = 0.85 for Xerox7, r = 0.825 for UIUCTex, r = 0.845 for PASCAL VOC 2007, and r = 0.86 for Caltech101 dataset. For each dataset, linear OVA-SVMs were employed in classification and the reported classification rates are of mean average precision (mAP) [10].

C. Results

1) Entropy-based Filtering for Feature Selection: Interestingly, On average about 57%, 46%, 73%, and 64% of training keypoints were found to have entropy value $E(F) > 4.1,\ 4.4,\ 3.6,$ and 3.8 that are selected from the initially extracted descriptor set in Xerox7, UIUCTex, PASCAL VOC 2007, and Caltech101 datasets, respectively in order to construct codebook for each classification tasks. This selection of reduced number of keypoints enhances the discriminative power of the codebook. We compare the proposed preprocessing technique with the traditional BoF approach. It has been noted that the filtering technique eliminates around 40% of the descriptors that outperforms traditional BoF approach in all datasets. The performance comparison of BoF approach prior to applying entropy-based filter vs after applying the technique is presented in Table I.

2) Codeword Selection Using Statistical Approach: On average 80% of the codewords were selected using inter-category confidence, intra-category confidence, and combined confidence C_{com} with $\alpha = 0.4$ and $\beta = 0.6$ to be the best for the initially constructed codebook. This step results in more discriminative and compact codebook. The performance of BoF approach with codeword selection using inter-category and intra-category confidences are shown in Table II.

In Table II, we compare the proposed post-processing technique (i.e., codeword selection method) with and without the use of preprocessing techniques (i.e., entropy-based filtering method). It yields on average 20% reduction in the initially constructed codebook in all datasets tested here. Finally, our proposed technique, having preprocessing and post-processing approaches, yield on average 45% of reduction in the initially constructed codebook while maintaining comparable performance with the traditional approach.

TABLE I. MEAN AVERAGE PRECISION RATE WITH CODEBOOK SIZE AND THE NUMBER OF DESCRIPTORS EXTRACTED FROM TRAINING IMAGES FOR THE ENTROPY-BASED FILTERING METHOD

BoF Approach	Dataset	#Descrips	CBSize	mAP
Traditional	Xerox7	4,046,578	987	67.64
	UIUCTex	4,543,590	1032	93.40
	PASCAL VOC 2007	1,760,400	1049	67.60
	Caltech101	5,659,137	925	77.52
Entropy-based Feature Selection	Xerox7	2,295,071	659	**69.07**
	UIUCTex	2,097,558	617	**95.04**
	PASCAL VOC 2007	1,286,833	918	**68.58**
	Caltech101	3,602,142	753	**78.36**

TABLE II. MEAN AVERAGE PRECISION RATE WITH CODEBOOK SIZE USING CATEGORICAL CONFIDENCES OBTAINED BY THE PROPOSED METHOD

Approach	Dataset	Before FS		After FS	
		CB	mAP	CB	mAP
Traditional	Xerox7	987	67.64	659	69.07
	UIUCTex	1032	93.40	617	95.04
	PASCAL VOC 2007	1049	67.60	918	68.58
	Caltech101	958	74.71	753	**78.36**
Inter-category Confidence	Xerox7	803	65.77	546	70.11
	UIUCTex	835	93.70	496	**95.84**
	PASCAL VOC 2007	847	68.02	744	67.73
	Caltech101	742	75.34	603	76.23
Intra-category Confidence	Xerox7	740	67.63	**494**	69.03
	UIUCTex	774	93.78	**463**	93.95
	PASCAL VOC 2007	787	67.80	**688**	**68.70**
	Caltech101	694	75.53	**565**	77.32
Combined Confidence	Xerox7	902	66.23	598	**70.32**
	UIUCTex	842	93.73	518	95.60
	PASCAL VOC 2007	953	67.58	818	67.30
	Caltech101	850	75.16	697	76.41

VI. CONCLUSION

BoF approach is an image representation scheme used in patch-based object categorisation. In such classification system, the major role of a codebook is to provide a way to map the low level features into a fixed length feature vector in histogram domain to which any classifiers can be directly applied.

Many of the large numbers of keypoints detected from images are actually unhelpful for recognition and the computational cost required for the vector quantisation step for the generation of BoF vectors is very high. A larger sized codebook increases the computational needs in terms of memory requirement for generating the histogram of each image which is proportional to the size of the codebook.

The high dimensional image representation could make many machine learning algorithms which become inefficient and unreliable or even a breakdown.

The central idea of the proposed algorithm in this paper is to select representative keypoints and select informative codewords so that the cluster structure of the image database can be best respected. The proposed method provides an effective way to reduce the BoF representation to low-dimension while maintaining the BoF model to be efficient with stable performance.

REFERENCES

[1] C. Cortes, V. Vapnik, *Support-Vector Networks*, Machine learning, Vol. 20, No. 3, Pages 273-297, 1995.

[2] D. Lowe, *Distinctive Image Features from Scale-invariant Keypoints*, In International Journal of Computer Vision, Vol. 60, Pages 91–110, 2004.

[3] C. Csurka, R. Dance, L. Fan, J. Willamowski, C. Bray, *Visual Categorization with Bags of Keypoints*, In Workshop on Statistical Learning in Computer Vision, ECCV'04, Pages 1-22, 2004.

[4] S. Lazebnik, C. Schmid, J. Ponce, *A Sparse Texture Representation Using Affine-invariant Regions*, In Pattern Analysis and Machine Intelligence, Vol. 27, No. 8, Pages 1265–1278, 2005.

[5] L. Fei-Fei, P, Perona, *A bayesian hierarchical model for learning natural scene categories*. In Proceedings of the IEE Conference on Computer Vision and Pattern Recognition, Vol. 2, Pages. 524-531, 2005.

[6] L. Fei-Fei, R. Fergus, R, P. Perona, *Learning Generative Visual Models from Few Training Examples: An Incremental Bayesian Approach Tested on 101 Object Categories*, In International Journal of Computer Vision and Image Understanding, Vol. 106, No. 1, Pages 59–70, 2007.

[7] K. E. A. van de Sande, T. Gevers, C. G. M. Snoek, *Color descriptors for object category recognition*. In Proceedings of the Conference on Colour in Graphics, Imaging, and Vision, Vol. 2008, No. 1, Pages 378–381, 2008.

[8] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, A. Zisserman. *The PASCAL Visual Object Classes (VOC) Challenge*, In International Journal of Computer Vision, Vol. 88, No. 2, Pages 303–338, 2010.

[9] A. Ramanan, M. Niranjan, *A One-pass Resource-Allocating Codebook for Patch-based Visual Object Recognition*, In proceedings of the IEEE International Workshop on Machine Learning for Signal Processing, Pages 35–40, 2010.

[10] K. H. Brodersen, C. S. Ong, K. E. Stephan, J. M. Buhmann, *The Binormal Assumption on Precision-Recall Curves*, In proceedings of the International Conference on Pattern Recognition, Pages 4263–4266, 2010.

[11] A. Ramanan, Niranjan. M, *A Review of Codebook Models in Patch-based Visual Object Recognition*, In Journal of Signal Processing Systems, Vol. 68, No. 3, Pages 333–352, 2012.

[12] L. Wang, L. Zhou, C. Shen, L. Liu, H. Liu, *A Hierarchical Word-merging Algorithm With Class Separability Measure*, In IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 36, No. 3, Pages. 417-435, 2014.

[13] J. Cui, M. Cui, B. Xiao, G. Li, *Compact and Discriminative Representation of Bag-of-features*, In Neurocomputing, Vol. 169, Pages 55–67, 2015.

[14] T. Kirishanthy, A. Ramanan, *Creating Compact and Discriminative Visual Vocabularies Using Visual Bits*, In Proceedings of the IEEE Digital Image Computing: Techniques and Applications, Pages 258–263, 2015.

[15] C. Wang, K. Huang, *How to use Bag-of-Words model better for image classification*. In Image and Vision Computing, Vol. 38, Pages 65–74, 2015.

[16] W. C. Lin, C. F. Tsai, Z. Y. Chen, S. W. Ke, *Keypoint Selection for Efficient Bag-of-words Feature Generation and Effective Image Classification*, In Information Sciences, Vol. 329, Pages 33–51, 2016.

[17] X. Peng, L. Wang, X. Wang, Y. Qiao, *Bag of Visual Words and Fusion Methods for Action Recognition: Comprehensive Study and Good Practice*, In Computer Vision and Image Understanding, Vol. 150, Pages 109–125, 2016.

[18] G. Amato, F. Falchi, C. Gennaro, *On Reducing the Number of Visual Words in the Bag-of-Features Representation*, In Computing Research Repository, Pages 657–662, 2016.

[19] V. Vinoharan, A. Ramanan, *Are Large Scale Training Images or Discriminative Features Important for Codebook Construction?*, In Proceedings of the 5th International Conference on Pattern Recognition Applications and Methods, Vol. 1, Pages 193–198, 2016.

[20] A. Nasirahmadi, S. H. M. Ashtiani, *Bag-of-feature Model for Sweet and Bitter Almond Classification*, In Biosystems Engineering, Vol. 156, Pages 51–60, 2017.