Electronic Letters on Computer Vision and Image Analysis 20(2):51-68, 2021

An Efficient Bag-of-Feature Representation for Object Classification

V. Vinoharan* and A. Ramanan+

* *Computer Unit, University of Jaffna, Jaffna, Sri Lanka*
+ *Department of Computer Science, University of Jaffna, Jaffna, Sri Lanka*

Received 01 March 2021; accepted 07 October 2021

Abstract

The Bag-of-features (BoF) approach has proved to yield better performance in a patch-based object classification system owing to its simplicity. However, often the very large number of patch-based descriptors (such as scale-invariant feature transform and speeded up robust features, extracted from images to create a BoF vector) leads to huge computational cost and an increased storage requirement. This paper demonstrates a two-staged approach to creating a discriminative and compact BoF representation for object classification. In the first stage, ambiguous patch-based descriptors are eliminated using an entropy-based and one-pass feature selection approach, to retain high-quality descriptors in constructing a codebook. In the second stage, a subset of codewords which is not activated enough in images are eliminated from the initially constructed codebook based on statistical measures. Finally, each patch-based descriptor of an image is assigned to the closest codeword to create a histogram representation. One-versus-all support vector machine is applied to classify the histogram representation. The proposed methods are evaluated on benchmark image datasets. Testing results show that the proposed methods enables the codebook to be more discriminative and compact in moderate sized visual object classification tasks.

Keywords: Bag-of-Features, Compact codebook, Codeword selection, Feature selection.

1 Introduction

Visual object classification is a process of predicting the presence of a specific object in a digital image or video sequence. Visual object classification, scene classification, and image searching have posed a great challenge for computer vision. A number of factors render the problem of recognition highly challenging: changes in pose, lighting, occlusion, clutter, intra-class differences, inner-class variances, deformations, background that varies relative to the viewer, large numbers of images and several object categories.

The bag-of-features (BoF) approach [1], [2], [3], [4], [5], [6], [7], [8], [9] is a popular technique, used for more than a decade to represent the image content, and has proved to yield better performance in many computer vision tasks. The BoF approach is a multi-step process, with each step presenting many options, and

Correspondence to: vvinoharan@univ.jfn.ac.lk. The authors contributed equally.

Recommended for acceptance by Angel D. Sappa
https://doi.org/10.5565/rev/elcvia.1403
ELCVIA ISSN:1577-5097
Published by Computer Vision Center / Universitat Autònoma de Barcelona, Barcelona, Spain

its advantages have been proven in moderate sized datasets. In the BoF approach, features are usually based on the utilisation of tokenising keypoint-based features, e.g., scale-invariant feature transform (SIFT) [10] or speeded up robust features (SURF) [11], 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 approach the visual codebook is generally large which may lead to the following issues:

- The feature vectors are high-dimensional and when support vector machines (SVMs) are applied to those vectors, the complexity of computing the kernel matrix, testing a new image, or storing the support vectors is all proportional to the size of the codebook. Thus, the generation of large BoF vectors not only requires greater computational cost, but the representation of the vectors also leads to a large storage requirement,
- the feature vector is also highly sparse as there is a non-zero value in dimensions corresponding to codewords that occur in the image patch,
- it might cause overfitting and reduce generalisation of the BoF model, and
- many of the detected interest points are non-informative and/or ambiguous in object classification. Mapping those interest points into the BoF representation hinders the classification performance.

To overcome these issues, such feature vectors need to be constructed to have the discriminative power to produce better classification performance and the vectors need to be represented compactly to reduce the computational cost. This motivated us to work on the following aspects of a BoF approach:

- Choosing unambiguous patch-based descriptors prior to the construction of a codebook in order to reduce the features causing false positives in object classification, and
- Selecting the best subset of codewords from an initially constructed codebook to enhance the discriminative power of the codebook and make it more compact.

Our contribution in this work is twofold:

1. We propose techniques to select informative features using: (i) one-pass feature selection algorithm by discarding visually similar keypoints at the nearest neighbours in a fixed-radius hyperspheres and (ii) entropy-based filtering approach that measures the information gained from each dimension of a patch-based descriptor to eliminate ambiguous keypoints. In addition, the combination of those two approaches were also explored.
2. We propose techniques to enhance the discriminative power and compactness of a codebook using: (i) confidence measures based on statistical characteristics of codewords and (ii) an encoding scheme based on a sequence of visual bits to generate a compact and discriminative BoF representation.

In recent years, deep learning algorithms particularly convolutional neural networks (CNNs) have proven their popularity and power in several computer vision tasks [12], [13], [14], [15] involving large amounts of data. Though the CNN architectures are successful in dealing with images, they have the limitations due to millions of trainable parameters that requires higher computational resources and a large amount of computational time. Hence, graphics processing units (GPUs) are required to train and run large models. Obtaining a much larger training dataset is not always a viable solution in certain classification tasks such as medical analysis. In the case of smaller datasets, one may apply pre-trained CNNs and transfer learning/data augmentation to alleviate issues in the classification tasks but is still not sufficient to overcome problems. Fine-tuning the pre-trained CNN is the transfer learning approach that allows extracting more context-specific features but it is more time consuming, as it still requires running the back-propagation algorithm on the training set whereas data augmentation techniques come with a serious risk of overfitting. Therefore, CNN is sensitive to the scalability of the training data and computationally demanding.

Given the simplicity of BoF models compared to CNNs, there are many use cases for which it can be desirable to trade a bit of accuracy for better interpretability on moderate sized datasets. Thus, the focus of this paper is on relatively small scale object classification tasks that could be carried out with lower computational resources and reduced processing time, while representing BoF vectors compactly and distinctively to yield better classification performance.

After this introductory section this paper is organised as follows. Section 2 reviews various techniques that have been used to construct a codebook with discriminative power and compactness in visual object classification. Section 3 presents the required background of BoF approach. Section 4 provides details of the proposed method. Section 5 provides the experimental setup and test results. Section 6 concludes this work with possible extension.

2 Literature Review

There is an extensive body of literature on visual object classification systems using BoF or patch-based codebook models. Lots of attempts have been made to improve the performance of the traditional BoF approach in image search [16], [17], image retrieval [18], [19], hand gesture recognition [20], object classification [21], [22], [23], landmark classification [24], scene categorisation [25], [26], [27], food recognition [28], drosophila gene expression pattern annotation [29], medical imaging [2], and weed/insect management in agriculture [3]. In such BoF approaches, compact and discriminative codebooks are preferred to tackle large scale visual object classification tasks. We restrict our survey to summarising techniques that have focused on both compactness and the discriminative power of codebooks. The well-known framework in the literature uses the SIFT descriptors to describe the patches and those descriptors are clustered using the traditional K-means algorithm, in order to encode images as a histogram of visual codewords as originally proposed by [30].

In [31], the authors proposed a keypoint selection approach, that enables visualisation of the keypoints using a fast correlation based filter (FCBF). The first stage of the FCBF measures the correlation between features and class labels, and retains the top features with highest correlation. The second stage, measures the correlation among features, and discards the ones which are highly correlated. The authors represent an image with the concatenation of dense SIFT (DSIFT) features extracted from uniform grid and facial fiducial points. The dimensionality of DSIFT features is reduced using principal component analysis (PCA). Over the reduced dimensional representation, LDA subspace is learned and the cosine distance is used as a distance metric for matching a pair of images. Experiments are performed on the CASIA NIR-VIS-2.0 dataset [32] that contains predefined visible and near-infrared images of 725 subjects. The authors explored the effectiveness of concatenating the features obtained from these two kinds of keypoints. The best results obtained with their proposed keypoint selection technique are comparable to that of the feature selection, and the performance is improved in both feature and keypoint selection approaches.

In [33], the authors proposed an efficient codewords selection method, which can be applied to the pedestrian detection problem. A large number of dense-SIFT descriptors extracted from a set of training images were clustered using a K-means algorithm and characterised by a histogram. The total frequency histograms for pedestrian and non-pedestrian images were computed. The difference in the total appearance frequency of each visual word in pedestrian and non-pedestrian images was computed. If the value was positive, this visual word was effectively classified as a positive sample, and vice versa. The corresponding visual codebooks which have the positive and negative values, are sorted by descending order of absolute value. Two limit values are set to determine the size of new visual codebook and the new frequency histogram was created. The frequency histogram of each visual word forms the training data that is input to the SVM. Their experiments used the Daimler-DB dataset [34] by randomly selecting 3000 pedestrian and 3000 non-pedestrian images for the training. They set the initial visual codebook size to 500. Based on the experiment the authors conclude that 200 efficient visual codebook in the original visual codebook can result in almost the same performance as with all 500 visual words.

In [35], the authors proposed a two-step approach to mapping an initially constructed large codebook into a compact codebook while maintaining its discriminative power. SIFT descriptors from the training images were clustered by K-means algorithm to construct an initial codebook and a frequency histogram was computed. Based on the histogram vector, the training images are represented using a coding scheme that maps the importance of each visual word within an image as visual bits. The average number of descriptors that fall into each visual word is computed as the threshold for each image of the training set. These sets of visual image bits then form a sparse representation of each visual word. Therefore, the best subset is selected by eliminating inconsistent visual words within each category based on the statistics of the visual words in the initial codebook. For each dataset, the authors considered the linear OVA-SVMs in the classification. The authors evaluated the proposed framework on Xerox7 and UIUCTex image sets with 70% for training and 30% for testing, on MPEG7 silhouette [36] image set with 50%-50% for training and testing; on PASCAL VOC 2007 with the provided training set and evaluating on the testing set. The technique yields more than 50% of compactness of the initially constructed codebook for a performance loss of 0.08%.

In [37], the authors proposed an unsupervised dimensionality reduction framework for constructing histogram vectors with compact and discriminative representation of BoF or Spatial Pyramid Matching (SPM). Histograms can be viewed as points on a statistical manifold, and Hellinger distance which approximates geodesic distances defined by the Fisher information distance and not only distance measure, but also the kernel is used to build the dissimilarity matrix. By using multidimensional scaling methods it is possible to represent the high-dimensional histograms in a low-dimensional Euclidean space, enabling effective learning in the low-dimensional Euclidean space. The authors have tested their technique on three benchmark image sets: subset of PASCAL VOC 2012, subset of Caltech-101, and Scene15 [38]. Some findings based on extensive experiments were observed: (i) using an intersection kernel to build a dissimilarity matrix can achieve more accurate classification than using distance in most cases; (ii) a small degree of dimensionality of BoF or SPM is sufficient for the learning tasks without a reduction in the accuracy of classification. In combination of PASCAL VOC 2012 and Caltech-101, the accuracy of the classification obtained with the original 1000 and 2000 dimensional BoF vectors can be achieved with the proposed algorithm from dimension 30 to 100. The accuracy of the classification of the original 2100 and 4200 dimensional SPM vectors can be achieved with the proposed algorithm from dimension 50 to 100.

In addition, in Table 2 we compare the following studies relating to BoF representation with our proposed method. Oliveira *et.al.* [39] proposed a sparse spatial coding (SSC) method to improve the object recognition tasks. Lin *et.al.* [40] proposed an iterative keypoint selection (IKS) to select representative keypoints in order to generate discriminative BoF representation for image classification. Quan *et.al.* [41] proposed ensemble classifier-based dictionary learning (Easy DL), whereas Wang *et.al.* [42] proposed unidirectional representation dictionary learning (URDL) for image classification tasks. Ghalyan *et.al.* [43] proposed an enhanced stochastically robust and optimized BoW (ESRO-BoW) as an enhancement to the BoW model. Zang *et.al.* [44] proposed a fast sparse coding spatial pyramid matching (FScSPM) framework to reduce learning complexity and improve the feature's stability in image classification. Recently, Chebbout *et.al.* [1] proposed a hybrid codebook method that is applied to image descriptors to generate two variant codebooks to encode and represent an image through a patch-based codebook model and a feature-based codebook model, respectively.

3 Bag-of-Features (BoF) Approach

The bag-of-words (BoW) approach was originally used in text mining and is thereafter widely used in image scene classification, retrieval of objects from a movie, and object classification tasks [45] in computer vision. The BoW in computer vision is normally referred to as 'Bag-of-Features' (BoF) or 'Bag-of-Keypoints'. In the BoF approach, invariant-features are first extracted from local regions on images and a visual codebook is constructed by applying a clustering algorithm on a subset of the features where the cluster centres are considered "visual words" or "codewords" in the codebook. Each feature in an image is then quantised to the

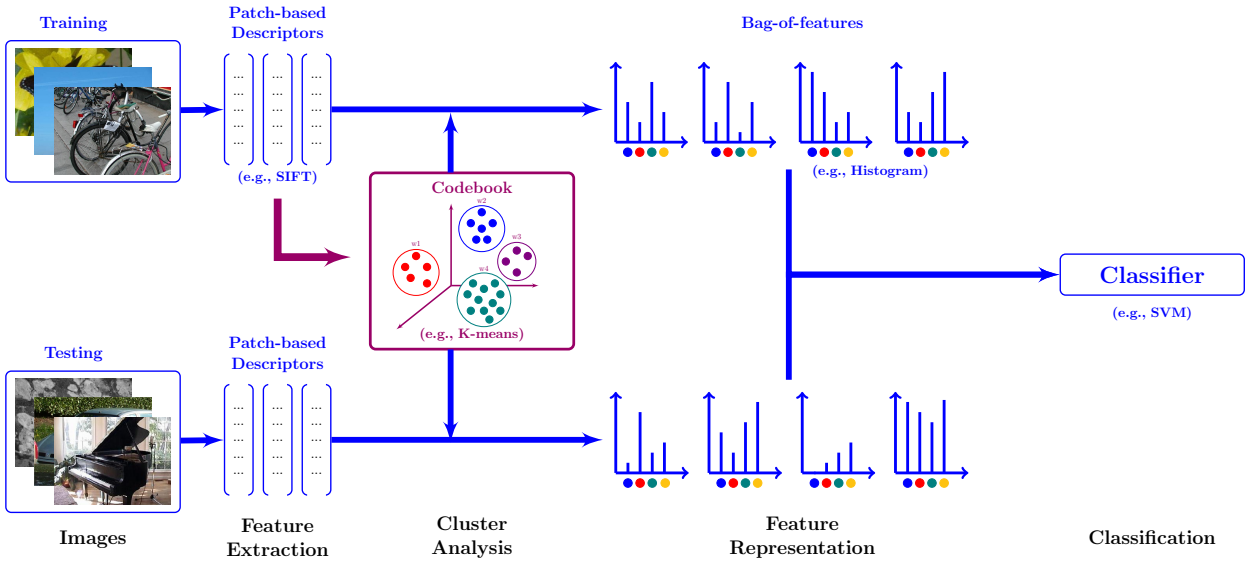


Figure 1: Generic framework of a BoF approach.

closest word in the codebook, and an entire image is represented as a global histogram counting the number of occurrences of each word in the codebook. Several patch-based visual object classification systems fit into a general framework which is depicted in Fig 1.

There are two broad categories of codebook models: Global and category-specific codebooks. A global codebook is category independent but its discriminative power may not be sufficient. On the other hand a category-specific codebook may be too sensitive to noise. Thus, the construction of a codebook plays a crucial role and affects the models' complexity. A number of different clustering techniques have been used by researchers in constructing codebooks, such as K-means [46], Resource Allocating Codebook (RAC) [47], mean-shift [48], Random Forests [49], hierarchical clustering [50], GMMs [51], etc. The study in this work uses K-means and RAC techniques. RAC is developed to overcome the problem of learning fixed size clusters that can be used at any time in the learning process, such that the learning patterns do not have to be repeated [47]. RAC carves the input space in a wider span than that which would be found by any density preserving method such as the traditional K-means algorithm. RAC is a simple and extremely fast technique for constructing visual codebooks using a one-pass setup that carves the feature space in to fixed-size hyperspheres resulting in a drastic reduction in computational needs.

4 Methodology

The BoF approach is a standard image representation scheme used in patch-based visual object recognition. In such patch-based object recognition systems, the key role of a visual codebook is to provide a way to map the low-level features into a fixed-length feature vector in a histogram space to which standard classifiers can be directly applied. A discriminative codebook can be obtained by the selection of representative keypoints and compact codebook can be obtained by elimination of indistinctive codewords that not only reduce the overall computational complexity but also increases the categorisation precision.

The central idea of the proposed algorithm in this work is to select representative keypoints and informative codewords so that the cluster structure of the image database can be best respected. The overall framework of our proposed method is depicted in Fig 2. We used SIFT descriptors in extracting the features from image sets that are reported in this work. The RAC algorithm is used as the baseline method in constructing a codebook,

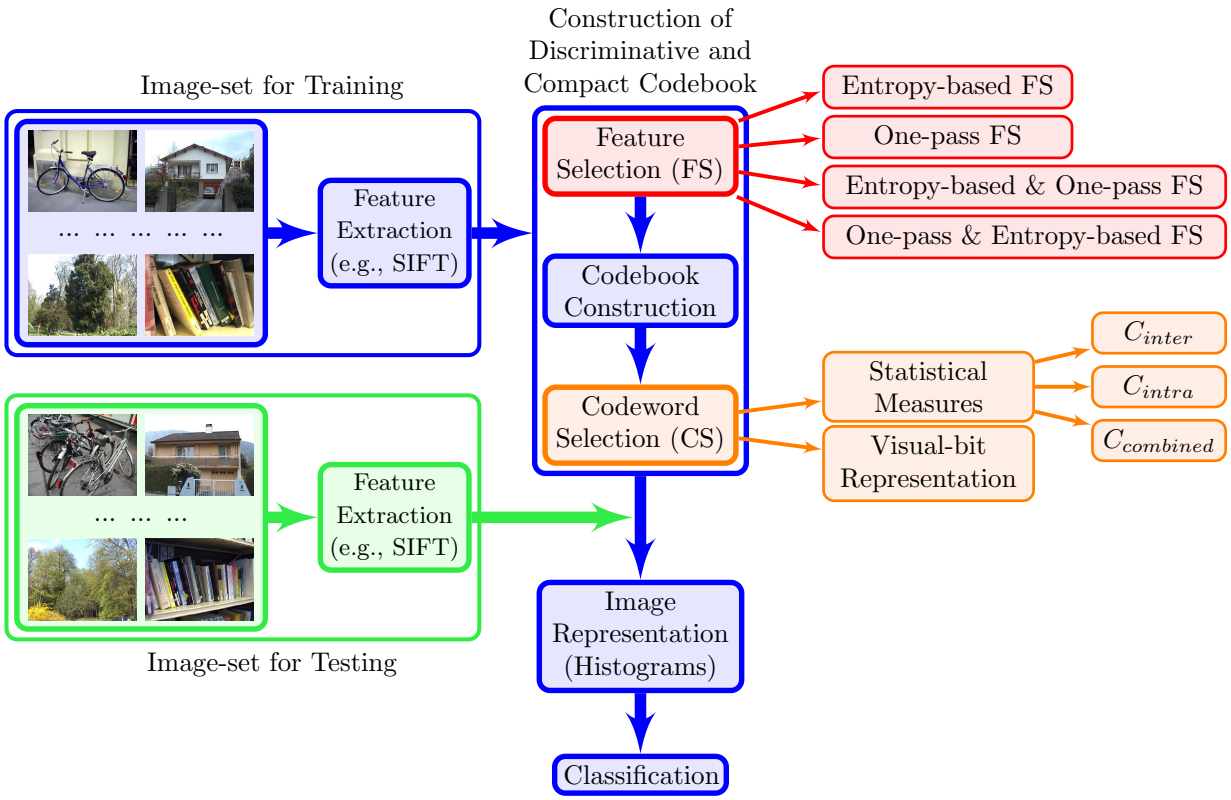


Figure 2: Overall framework of the proposed technique.

whereas an entropy-based feature selection (EBFS) approach is applied to eliminate ambiguous SIFT descriptors prior to constructing a codebook using RAC. The standard K-means algorithm is used elsewhere when constructing a codebook as the one-pass feature selection (OPFS) approach is involved during the feature selection process. The OPFS technique in principle is same as the RAC technique. Following the codebook construction, a subset of codewords are selected from the initially constructed codebook based on statistical measures C_{inter}, C_{intra} and $C_{combined}$. The C_{inter} selects discriminative codewords across categories, whereas C_{intra} selects consistent codewords within each categories. The combined measure of C_{inter} and C_{intra} is used to select an informative subset of codewords. In addition, a visual bit representation technique for codeword selection is also tested. The proposed method yields a BoF representation that enables machine learning algorithms to be trained faster, reducing the overall complexity of a model and making it easier to interpret, thus improving its accuracy.

4.1 Selecting Representative Keypoints

The proposed framework consists of techniques of selecting informative keypoints prior to constructing visual codebook using: (i) OPFS, (ii) EBFS, (iii) EBFS followed by OPFS, and (iv) OPFS followed by EBFS algorithms. It is witnessed that feature subsets give better results than complete sets of features for the same algorithm.

4.1.1 Entropy-based Feature Selection (EBFS)

Generally, entropy is a measure of how uncertain we are about the data, and can be used to measure how much information we gain from an attribute when the target attribute is revealed to us. A patch-based feature

with low entropy is detected from a homogeneous region and one with high entropy is detected from a heterogeneous region. SIFT features best suitable for object detection are those with a rich internal structure and those associated with near-empty regions are the main source of false positives: they tend to occur frequently and get easily matched against one another. This proposes an entropy-based filtering approach to eliminate ambiguous SIFT descriptors in order to retain high-quality descriptors. The proposed 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 $\{0, 1, 2, \cdots, 255\}$.

Then the entropy of *F* is computed as,

$$E(F) = -\sum_{i=0}^{255} p_i(F) log_2 p_i(F) \tag{1}$$

where,

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

The values for individual dimensions of a 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 datasets. This process is summarised in Algorithm 1.

Algorithm 1 EBFS

Input: Training images (trnImgs), #clusters (K), predefined threshold (thresh)
Output: Selected features (selFts)
for all $img_i \in$ *trnImgs* **do**
interestPts ← detectPts(img_i)
descrips{i} ← describePts(interestPts)
end for
selFts ← []
i ← 1
for all $fts_i \in$ descrips **do**
entVal ← 0
j ← 0
for $j \leq 255$ **do**
entVal ← entVal + $p_j(fts_i) \times log_2 p_j(fts_i)$,
where, $p_j(fts_i) \leftarrow \frac{|k|f_k=j|}{128}, k \leftarrow 0, 1, 2, \ldots, 255$
j = j + 1
end for
if *entVal* $\leq$ *thresh* **then**
$selFts \leftarrow \{selFts \cup fts_i\}$
end if
i ← i + 1
end for
return *selFts*

Algorithm 2 OPFS

Input: Training images (trnImgs), #clusters (K), radius of the hypersphere (r)
Output: Selected descriptors set (selDesc)
for all $img_i \in$ *trnImgs* **do**
interestPts ← detectPts(img_i)
descrips{i} ← describePts(interestPts)
end for
// Initialise selected feature set
selDesc ← descrips{1}
i ← 2
for all $desc_i \in$ descrips **do**
if *min*$\| desc_i - selDesc \|^2 > r^2$ **then**
Create a new hypersphere of *r* such that,
$selDesc \leftarrow \{selDesc \cup desc_i\}$
end if
i ← i + 1
end for
return selDesc

4.1.2 One-pass Feature Selection (OPFS)

The idea is to construct a discriminant codebook for visual object classification by means of a one-pass feature selection approach. OPFS is a simple and extremely fast way of selecting discriminative features which simultaneously achieves increased discrimination. The goal is to select features by discarding visually similar keypoints at the nearest neighbours in a fixed-radius hyperspheres.

OPFS starts by arbitrarily assigning the first data item as an entry of a discriminative feature. When a subsequent data item (i.e., descriptor) is processed, its minimum distance to all entries in the discriminative feature set is computed using an appropriate distance metric. If this distance is smaller than a predefined threshold r (radius of the hypersphere), the discriminative feature set is retained and no action is taken with respect to the processed data item. If the threshold is exceeded by the smallest distance to feature set, a new entry in the discriminative feature set is created by including the current descriptor set as the additional entry. This process is continued until all the features are seen only once. The one-pass feature selection algorithm used in this work is summarised in Algorithm 2.

4.1.3 EBFS followed by OPFS

The descriptors selected using the entropy-based filtered technique, is further filtered by an OPFS approach, that eliminates ambiguous 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 representation. Initially extracted features were first filtered using EBFS technique to reduce the false positive rate by orders of magnitude, and informative keypoints were selected using OPFS, which is especially prominent when the true positive rate is low.

4.1.4 OPFS followed by EBFS

This approach is simply the previous method in reverse order i.e., initially extracted features were first filtered using OPFS technique and informative keypoints were selected using EBFS technique in order to retain high quality descriptors.

4.2 Constructing Compact Codebook

A compact visual codebook provides a lower-dimensional representation, whereas a large-sized codebook may overfit to the distribution of codewords in an image and lead to a heavy computational load. To achieve this (i) inter-category, intra-category, and combined-category confidences were used to select the informative subset of codewords; (ii) the codebook was reformulated using a bitwise representation to generate a compact and discriminative codebook for the BoF representation.

An important issue with the visual codebook representation is its discriminative power and dimensionality. Most of the visual codebooks that are used in larger evaluations consist of 1000 to 10,000 codewords. This higher dimensionality curses the subsequent classifier in the training procedure. Thus, most of the object recognition systems expect the histogram representation of a BoF approach to be more compact while maintaining the discriminative power. The goal of codeword selection is to select the best subset of codewords from an initially constructed codebook to enhance the discriminative power and make it more compact. By eliminating indistinctive codewords, one can reduce the computational complexity and increase the categorisation precision.

4.2.1 Codeword Selection Using Statistical Approach

To achieve a compact codebook inter-category, intra-category, and combined (i.e., inter and intra) confidences are proposed to select an informative subset of codewords for the BoF representation.

i) *Selecting discriminative codewords across categories:* Visual object categories having similar histogram distribution may increase the ambiguity of the categorisation system. Intercategory 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 represented as follows:

$$C_{inter,i} = \sum_{j=1}^{N} max\left(\frac{f_{ij}}{n_i} - \frac{1}{m_i}, 0\right) \tag{2}$$

where,

- f_{ij} - number of training features in the i^{th} codeword and j^{th} category, i = 1, 2, . . . , K, and j = 1, 2, . . . , N.
- n_i - total number of features in the i^{th} codeword.
- m_j - number of object categories in the i^{th} codeword.
- N - number of object categories in classification.
- K - size of the codebook.

The inter-category confidence has the value 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 inter-category confidence has a positive value when the feature ratio of a codeword shows a single category dominating other categories in the feature domain. Since a codeword only exists in histograms of the category images, the histogram distribution differs from other categories, thus the codeword enhances the categorisation result. In this process of selecting codewords, it has been noticed that many codewords disappear from homogeneous regions. Because these homogeneous codewords are distributed in different categories by different values, their confidences are very low and they are eliminated in the codeword selection process.

In this inter-category codeword selection, we select the codeword based on the following criteria:

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

ii) *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 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 it difficult for the classifier to classify visual object categories. Thus, low variance codewords at BoF histogram domain are stable for classification. 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:

$$C_{intra,i} = \frac{1}{\sum_{j=1}^{N} var(h_{ij})} \tag{3}$$

where,

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

iii) *Selecting informative subset of 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 represented as:

$$C_{com,i} = \alpha C_{inter,i} + \beta C_{intra,i} \tag{4}$$

where, α and β are constant values, $0 \leq \alpha, \beta \leq 1$. Using the combined confidence, we select reliable codewords by weighting parameters.

4.2.2 Visual Bit Representation of Codewords

The proposed technique in [35] is used to reduce the size of a codebook by means of visual bit representations of images, and visual words which improve coding efficiency while maintaining the discriminative power of the codebook. This is achieved by following a two-step process: (i) encoding each image as 'bits', i.e., the significant presence or absence of each visual word and (ii) removing visual words, i.e., cluster centres, with 'bits' that are not activated enough in images. The technique is summarised below:

i) *Visual bit representation of images:* Training images of a specific category are used to construct an initially large codebook [CB] (e.g., $|CB| = 1000$). The patch-based descriptors of image, $\mathtt{I}$, are mapped into a feature vector by computing the frequency histogram, $\mathtt{h}$, with the initial codebook CB. The average number of descriptors that fall into each codeword C_i of CB is computed as t_0 for each image $\mathtt{I}$ of the training set. The visual bit representation of an image is then coded using the following equation.

$$h_i = \begin{cases} 1: & \text{if } C_i \geq t_0 \qquad \forall i = 1, \ldots, K \\ 0: & \text{otherwise} \end{cases} \tag{5}$$

This process is repeated to all training images of a specific category by computing t_0 corresponding to an image. In contrast, t_0 indicates the average level of significant presence of each visual word in a specific image.

ii) *Visual bit representation of codewords:* The goal of constructing codebook is to use fewer visual words that represent categories. In this regard, the best subset is selected by eliminating inconsistent visual words within each category based on the statistics of the visual words in the initial vocabulary. Categories having sparse histogram distribution may increase the ambiguity of the categorisation performance. The selected visual word subset enlarges the distribution difference. Following the visual bit representation of images in equation (5), the initial codebook CB is coded as a sparse representation by using the following equation:

$$t_1 = \frac{\lambda p_0 + p_1}{\lambda + 1} \tag{6}$$

where,

- P_0 - $min_{1 \leq i \leq K}(SB_i)$
- P_1 - $max_{1 \leq i \leq K}(SB_i)$
- SB_i - sum of visual bits associated with the i^{th} codeword.
- λ - weighting parameter for a rare informative word.

We compute the largest and the smallest coefficients associated with each of the codewords in the initial codebook. Rare low-level features are expected to be associated with the visual word having the smallest coefficient.

iii) *Reduction of codewords:* The weight or importance of each visual word of the initial codebook is learnt through the visual bit representation of visual words. Thereafter, the compression in the codebook can be performed based on the weights as expressed in the following equation:

$$Compact_{CB} = \begin{cases} \text{eleminate } C_i & : \text{if } C_i \geq t_0 \\ \text{retain } C_i & : \text{Otherwise} \end{cases} \tag{7}$$

where, t_i indicates the level of significant activation of a codeword in a codebook. If a codeword having the sum of visual bits less than the predefined threshold t_1, it is eliminated from the codebook; otherwise it is retained. Based on this concept, we select all the codewords with the visual bit value of a codeword greater than the second quartile of C_i.

5 Experimental Setup and Test Results

The proposed techniques in this paper have been tested on Xerox7 [30], UIUC Texture [52], PASCAL VOC Challenge 2007 [53] and Caltech-101 [54] benchmark datasets.

5.1 Dataset

Xerox7: consists of 1776 images from seven categories with different resolutions. The object poses are highly variable and there is a significant amount of background clutter, some of which belongs to the other categories making the classification task fairly challenging. The images contain the presence of multiple instances of the same object category, variable poses, and significant amounts of background clutter, which makes the classification much harder. This image dataset was originally used in [30].

UIUCTex: contains 25 texture classes [52] with 40 images per class with resolution of 640×480. This dataset has surfaces whose texture is mainly due to albedo variations (e.g., wood and marble), 3D shape (e.g., gravel and fur), as well as a mixture of both (e.g., carpet and brick). It also has significant viewpoint changes, uncontrolled illumination, arbitrary rotations, and scale differences within each class.

PASCAL VOC Challenge 2007: is widely used in large-scale evaluation of visual object classification task. The dataset consists a total of 9,963 images containing 24,640 annotated objects [53], split into training, validation, and test sets labelled with twenty object classes. The training and validation sets consist of images where in each image multiple objects from multiple classes may be present. The example images show the presence of multiple instances of the same object category and different object categories in the single image under various conditions that make the classification difficult.

Caltech-101: is of a total of 9,146 images [54], split between 101 different object categories, as well as an additional background/clutter category. Each object category contains between 40 and 800 images and popular categories such as faces tend to have a larger number of images than others. Most categories have about 50 images and the size of each image is roughly 300 × 200 pixels.

5.2 Experimental Setup

For the image sets: Xerox7, UIUCTex, and Caltech-101 we used 70% for training and 30% for testing from each class. The classification for PASCAL VOC 2007 was performed on each of the twenty classes by training the classifiers on the provided 'trainval' set and evaluating it on the testing set. We used SIFT descriptors in extracting the features from those image sets. The visual codebook is then constructed by clustering the descriptors that were extracted from the training images using the K-means algorithm with K = 500 or

RAC algorithm with r = 0.85 for Xerox7; r = 0.825 for UIUCTex; r = 0.845 for PASCAL VOC Challenge 2007, and r = 0.86 for Caltech-101 datasets. For each dataset, we considered the OVA-SVMs with RBF Kernel in classification and the reported classification rates are of binormal average precision (AP) [55].

5.3 Test Results

The performance comparisons of the traditional BoF approach to the proposed method in section 4.1 and section 4.2 are presented in Table 1. The baseline method indicates the performance of the traditional BoF approach when no keypoint and codeword selection is made for object classification tasks. The entropy-based and one-pass feature selection approaches in this work aim to select a subset of features by ignoring the redundant and irrelevant features that can eliminate the dimensionality of data and improve the classification performance at a later stage. A codeword could be comprised of features from different objects, either representing visual concepts/parts of objects common to those different object categories or many of them probably belonging to the same object category. In order to represent best an object category, a property that a codeword must satisfy to have a high representativeness of the object category or high generalisation over the object category. These characteristics are measured by inter-category and intra-category confidences, respectively. The above mentioned confidence measures, provide a quantitative evaluation of the representativeness and distinctiveness of the codewords in a codebook for each object category by showing the highest scores.

Table 1: Comparison of Average Precision (AP) with Number of Training Features and Codebook Size: Traditional BoF Approach and Proposed Feature Selection Method *with* and *without* Codeword Selection (CS)

Approach	Dataset	#Descriptors	Without CS		Statistical Measures with CS						Visual bit with CS	
					inter		intra		combined			
			CB	AP	CB	AP	CB	AP	CB	AP	CB	AP
Baseline	Xerox7	4,046,578	987	84.21	803	83.68	740	87.89	902	82.41	286	83.85
EBFS		2,295,071	659	93.50	546	93.76	494	94.65	598	93.41	213	93.70
OPFS		212,294	500	94.11	400	93.31	375	94.69	409	93.72	191	93.42
EBFS+OPFS		178,328	500	93.66	400	92.95	375	94.19	406	93.23	185	93.06
OPFS+EBFS		172,006	500	94.04	400	93.40	**375**	**94.79**	406	93.41	201	94.13
Baseline	UIUCTex	4,543,590	1032	82.73	835	81.94	774	86.40	842	81.53	387	90.25
EBFS		2,097,558	617	94.58	496	93.31	463	95.65	518	93.36	217	94.00
OPFS		314,724	500	93.73	400	94.56	375	95.51	401	94.27	264	92.45
EBFS+OPFS		229,244	500	95.29	399	94.71	**375**	**95.90**	404	95.26	246	94.48
OPFS+EBFS		157,094	500	94.17	400	92.95	374	94.08	404	92.88	257	93.48
Baseline	PASCAL VOC 2007	1,760,400	1049	71.78	847	72.41	787	73.71	953	71.99	421	71.69
EBFS		1,286,833	918	71.52	744	70.94	688	73.74	818	71.67	348	71.31
OPFS		245,327	500	72.93	400	73.16	375	73.47	405	73.91	262	72.88
EBFS+OPFS		233,393	500	73.76	400	74.04	375	74.35	**409**	**74.39**	273	73.81
OPFS+EBFS		181,248	500	72.58	400	72.90	375	73.64	414	72.71	252	72.20
Baseline	Caltech-101	5,659,137	925	84.72	742	82.87	694	84.80	850	82.30	336	84.32
EBFS		3,602,142	753	85.10	603	83.71	565	85.97	697	83.47	314	84.82
OPFS		393,024	500	86.01	400	85.17	375	85.97	408	85.83	289	85.48
EBFS+OPFS		351,315	500	86.41	400	86.55	**375**	**86.68**	411	86.66	256	86.36
OPFS+EBFS		286,925	500	86.02	400	85.36	375	86.34	407	85.50	249	85.35

5.3.1 OPFS with codeword selection approaches

On average about 5%, 7%, 14%, and 7% of training keypoints were selected with radius r = 0.65 for Xerox7, UIUCTex and PASCAL VOC 2007, and r = 0.70 for Caltech-101 dataset, respectively. The proposed technique, having one-pass feature selection algorithm as a preprocessing step with traditional BoF approach has shown that the filtering technique retains around 10% of the descriptors thus outperforming the later approach in all datasets. The preprocessing step with statistical measures as a post-processing step yields on average 60% of reduction and visual bit representation of codeword method yields on average 80% of reduction in the initially constructed codebook while maintaining comparable performance with the traditional approach.

5.3.2 EBFS with codeword selection approaches

On average about 57%, 46%, 73%, and 64% of training keypoints were selected with entropy value $E(F) > 4.1$, 4.4, 3.6, and 3.8 from the initially extracted descriptors set from Xerox7, UIUCTex, PASCAL VOC 2007, and Caltech-101 datasets, respectively. The proposed technique, having entropy-based feature selection algorithm as preprocessing step with traditional BoF approach has shown that the filtering technique retains around 40% of the descriptors thus outperforming the later approach in all datasets. The preprocessing step with statistical measures as post-processing step yields on average 40% of reduction and visual bit representation of codeword method yields on average 70% of reduction in the initially constructed codebook while maintaining comparable performance with the traditional approach.

5.3.3 EBFS followed by OPFS with codeword selection

On average about 4.5%, 5%, 13.25%, and 6% of training keypoints were selected with radius of OPFS r = 0.65 for Xerox7, UIUCTex and PASCAL VOC 2007, and r = 0.70 for Caltech-101 and entropy value $E(F) > 4.1$, 4.4, 3.6, and 3.8 from the initially extracted descriptors set from Xerox7, UIUCTex, PASCAL VOC 2007, and Caltech-101 datasets, respectively. The proposed technique, having EBFS followed by OPFS as a preprocessing step with traditional BoF approach has shown that the filtering technique retains around 6% of the descriptors thus outperforming the latter approach in all datasets. The preprocessing step with statistical measures as a post-processing step yields on average 60% of reduction and visual bit representation of codeword method yields on average 75% of reduction in the initially constructed codebook while maintaining comparable performance with the traditional approach.

5.3.4 OPFS followed by EBFS with codeword selection

In our experiment on average about 4%, 3%, 10%, and 5% of training keypoints were selected with radius of OPFS r = 0.65 for Xerox7, UIUCTex and PASCAL VOC 2007, and r = 0.70 for Caltech-101 and entropy value $E(F) > 4.0$ for Xerox7, PASCAL VOC 2007 and Caltech-101 and $E(F) > 3.8$ for UIUCTex dataset from the initially extracted descriptors set. The proposed technique, having OPFS followed by EBFS as a preprocessing step with traditional BoF approach has shown that the filtering technique retains around 6% of the descriptors thus outperforming traditional BoF approach in all datasets. The preprocessing step with statistical measures as post-processing step yields on average 60% of reduction, whereas the visual bit representation of codeword method yields on average 75% of reduction in the initially constructed codebook while maintaining comparable performance with the traditional approach.

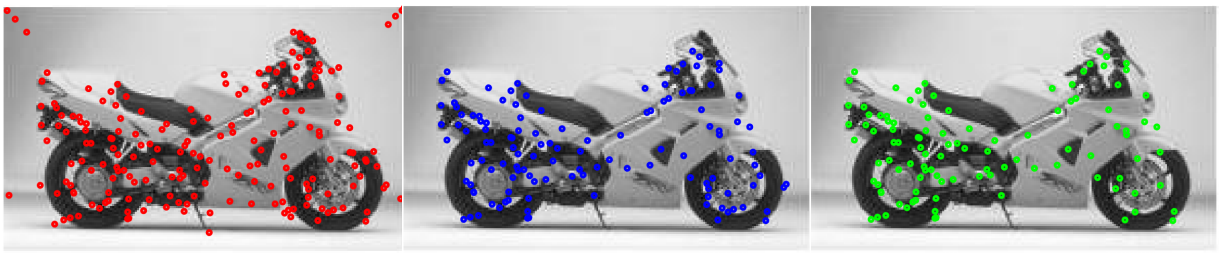
Table 2: Comparison of classification rate on Caltech-101 dataset.

Authors	Method	CB Size	Accuracy
Oliveira *et.al.* (2012) [39]	SSC	4096	69.00
Lin *et.al.* (2016) [40]	BoF+SPM	200	41.91
Quan *et.al.* (2016) [41]	Easy DL	1530	68.40
Wang *et.al.* (2018) [42]	URDL	1530	69.15
Ghalyan *et.al.* (2018) [43]	ESRO-BoW	1200	80.52
Zang *et.al.* (2019) [44]	FScSPM	—	76.30
Chebbout *et.al.* (2020) [1]	Hybrid CB	300	69.15
The current authors	FS+BoF+CS	375	**81.39**

5.3.5 Further Analysis

To compare our proposed approach to other related studies of BoF approaches in object classification reported in the literature, we followed the same experimental set up as suggested by [56] for the Caltech101 dataset with 15 training images for each object category, and used up to 50 testing images per category. We measured the performance using average accuracy over 101 classes. Table 2 shows the classification rate achieved by several state-of-the-art methods and our method (i.e., EBFS followed by OPFS for feature selection (FS), BoF representation, and C_{intra} for codeword selection (CS)) evaluated on the Caltech-101 dataset. As can be seen, our approach shows better performance on the Caltech-101 dataset.

Figure 3 shows plots of interest points detected by Difference of Gaussians (DoGs) on images of Caltech-101 (see 3(a)) and the informative keypoints selected by the proposed EBFS (see 3(b)) followed by OPFS (see 3(c)) techniques. The image plots show that keypoints eliminated by the proposed method are mostly associated with near-empty regions that are the main source of false positives. Such eliminated keypoints tend to occur frequently and get easily matched against one another. Thus, the filtering approach ensures both discriminative power and reduced computational cost in the classification step.



(a) (b) (c)

Figure 3: Plots of interest points before and after feature selection techniques. (a) Detected points by DoG, (b) Filtered points by EBFS, and (c) Filtered points by EBFS+OPFS techniques.

The computational time for clustering 1,318,254×128 SIFT descriptors using K-means with K=500 (see Table 3) and performing multiclass classification using SVMs was observed to be about 500 hours on the Caltech-101 dataset. The same task when using RAC needed 4.7 hours whereas, the proposed approach needs only 4.1 hours including the time of codebook compaction. All these computational times were obtained on a desktop computer with an Intel core i7 running at 3.4GHz and 32GB of RAM.

Table 3: Time comparison of constructing codebook and performing classification on Caltech-101 dataset.

Method	CB Size	Time (in Hrs)
K-means+SVM	500	500.42
RAC+SVM	671	4.70
K-means+CS+SVM (The current authors)	375	**4.10**

6 Discussion and Conclusion

The BoF approach is a standard image representation scheme used in patch-based visual object recognition. In such object recognition systems, the key role of a visual codebook is to provide a way to map the low-level features into a fixed-length feature vector in histogram space to which standard classifiers can be directly applied. Many of the large numbers of keypoints detected from images are actually unhelpful for recognition and the computational cost of the vector quantisation step for the generation of BoF features 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 codebook size. The high dimensional image representation could make many machine learning algorithms become inefficient and unreliable. The central idea of the proposed algorithms in this work is to select representative keypoints and informative codewords so that the cluster structure of the image database can be best respected. The proposed methods provide an effective way of reducing the BoF representation to low-dimension while maintaining the efficiency and stability of the BoF model. Furthermore, the entire feature selection process of the OPFS technique, involves use of a fixed radius as the hyper-parameter, but at a certain stage of filtering the features the radius can be updated according to a learning rate yielding small sized hyperspheres in subsequent steps of cluster analysis.

Conflict of interest

The authors declare that they have no conflict of interest.

References

[1] S. Chebbout and F. H. Merouani, “A Hybrid Codebook Model for Object Categorization Using Two-way Clustering Based Codebook Generation Method”, *International Journal of Computers and Applications*, pp.1-9, 2020.

[2] F. France and A. Jaya, “Classification and Retrieval of Thoracic Diseases Using Patch-based Visual Words: A Study on Chest X-rays”, *Biomedical Physics and Engineering Express*, 6(2):025015, 2020.

[3] W. E. Santiago, N. J. Leite, B. J. Teruel, M. Karkee, and C. A. Azania, “Evaluation of Bag-of-features (BoF) Technique for Weed Management in Sugarcane Production”, *Australian Journal of Crop Science*, 13(11):1819-1825, 2019.

[4] P. Lin, D. Li, J. Zhang, and Y. Chen, “Color-and Gradient-Related Visual Dictionary Models for Discrimination of Glycine Max (L.) Merrill Quality”, In *Proceedings of 2nd IEEE International conference on Artificial Intelligence and Big Data*, pp.400-403, 2019.

[5] V. D. Sachdeva, E. Fida, J. Baber, M. Bakhtyar, I. Dad, and M. Atif, “Better Object Recognition Using Bag of Visual Word Model With Compact Vocabulary”, In *Proceedings of 13th IEEE International conference on Emerging Technologies*, pp.1-4, 2017.

[6] A. Nasirahmadi and S-H. M. Ashtiani, “Bag-of-feature Model for Sweet and Bitter Almond Classification”, *Biosystems Engineering*, 156:51-60, 2017.

[7] G. Amato, F. Falchi, and C. Gennaro, "On Reducing the Number of Visual Words in the Bag-of-features Representation", arXiv:1604.04142, 2016. https://arxiv.org/abs/1604.04142, 2018.

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

[9] M. U. Kim and K. Yoon, "Performance Evaluation of Large-scale Object Recognition System Using Bag-of-visual Words Model", *Multimedia Tools and Applications*, 74(7):2499-2517, 2015.

[10] D. G. Lowe, "Distinctive image features from scale-invariant keypoints". *International Journal of Computer Vision*, 60(2):91-110, 2004.

[11] H. Bay, T. Tuytelaars, and Y. Van Gool, "SURF: Speeded up robust features". In *Proceedings of European Conference on Computer Vision*, pp.404-417, 2006.

[12] M. Shajini and A. Ramanan, "A Knowledge-sharing Semi-supervised Approach for Fashion Clothes Classification and Attribute Prediction", *The Visual Computer*, Springer, doi 10.1007/s00371-021-02178-3, June 2021.

[13] Z. Yang, Y. Wang, C. Liu, H. Chen, C. Xu, B. Shi, and C. Xu, "Legonet: Efficient Convolutional Neural Networks with Lego Filters", In *Proceedings of International conference on Machine Learning*, pp.7005-7014, 2019.

[14] M. Buda, A. Maki, and M. A. Mazurowski, "A systematic study of the class imbalance problem in convolutional neural networks", *Neural Networks*, 106:249-259, 2018.

[15] W. Rawat and Z. Wang, "Deep convolutional neural networks for image classification: A comprehensive review", *Neural Computation*, 29(9):2352-2449, 2017.

[16] H. Jégou, M. Hervé, and C. Schmid, "Improving Bag-of-features for Large Scale Image Search", *International Journal of Computer Vision*, 87(3):316-336, 2010.

[17] H. Jégou, M. Hervé, and C. Schmid, "Packing Bag-of-features", In *Proceedings of 12th International Conference on Computer Vision*, pp.2357-2364, 2009.

[18] M. Eitz, K. Hildebrand, T. Boubekeur, and M. Alexa M, "Sketch-based Image Retrieval: Benchmark and Bag-of-features Descriptors", *IEEE Transactions on Visualization and Computer Graphics*, 17(11):1624-1636, 2011.

[19] X. Yuan, J. Yu, Z. Qin, and T. Wan, "A SIFT-LBP Image Retrieval Model Based on Bag of Features", In *Proceedings of IEEE International Conference on Image Processing*, pp.1061-1064, 2011.

[20] N. Dardas, Q. Chen, N. D. Georganas, and E. H. Petriu, "Hand Gesture Recognition Using Bag-of-features and Multi-class Support Vector Machine", In *Proceedings of IEEE International Conference on Haptic Audio-Visual Environments and Games*, pp.1-5, 2010.

[21] Q. Chen, Z. Song, J. Dong, Z. Huang, Y. Hua, and S. Yan, "Contextualizing object detection and classification", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(1):13-27, 2014.

[22] R. Grzeszick, L. Rothacker, and G. A. Fink, "Bag-of-features Representations Using Spatial Visual Vocabularies for Object Classification", In *Proceedings of IEEE International Conference on Image Processing*, pp.2867-2871, 2013.

[23] Y. -G. Jiang, C. -W. Ngo, and J. Yang, "Towards Optimal Bag-of-features for Object Categorization and Semantic Video Retrieval", In *Proceedings of 6th ACM International Conference on Image and Video Retrieval*, pp.494-501, 2007.

[24] Y. Li, D. J. Crandall, and D. P. Huttenlocher, "Landmark Classification in Largescale Image Collections", In *Proceedings of 12th International Conference on Computer Vision*, pp.1957-1964, 2009.

[25] L. Zhou, Z. Zhou, and D. Hu, "Scene Classification Using a Multi-resolution Bag-of-features model", *Pattern Recognition*, 46(1):424-433, 2013.

[26] J. C. Van Gemert, J. -M. Geusebroek, C. J. Veenman, and A. W. Smeulders, "Kernel Codebooks for Scene Categorization", In *Proceedings of European Conference on Computer Vision*, pp.696-709, 2008.

[27] J. Yang, Y. -G. Jiang, A. G. Hauptmann, and C. -W. Ngo, "Evaluating Bag-of-visual words Representations in Scene Classification", In *Proceedings of International Conference on Multimedia Information Retrieval*, pp.197-206, 2007.

[28] M. Anthimopoulos, L. Gianola, L. Scarnato, P. Diem, and S. G. Mougiakakou, "A Food Recognition System for Diabetic Patients Based on an Optimized Bag-of-features Model", *IEEE Journal of Biomedical and Health Informatics*, 18(4):1261-1271, 2014.

[29] S. Ji, Y. -H. Li, Z. -H. Zhou, S. Kumar S, and J. Ye, "A Bag-of-words Approach for Drosophila Gene Expression Pattern Annotation", *BMC Bioinformatics*, 10(1):1-16, 2009.

[30] G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual Categorization with Bags of Keypoints". In *Proceedings of International Conference on Statistical Learning in Computer Vision*, pp.1-22, 2004.

[31] S. Ghosh, T. I. Dhamecha, R. Keshari, R. Singh, and M. Vatsa, "Feature and Keypoint Selection for Visible to Near-infrared Face Matching", In *Proceedings of 7th International Conference on Biometrics Theory, Applications and Systems*, pp.1-7, 2015.

[32] S. Li, D. Yi, Z. Lei, and S. Liao, "The CASIA NIR-VIS 2.0 Face Database", In *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition Workshops*, pp.348-353, 2013.

[33] X. Zhang, G. Chen, K. Saruta, and Y. Terata, "An Improved Visual Codebook Selection Method for Pedestrian Detection", *Digital Content Technology and its Applications*, 9(1):31-39, 2015.

[34] M. Enzweiler and D. M. Gavrila, "Monocular Pedestrian Detection: Survey and Experiments", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12:2179-2195, 2008.

[35] T. Kirishanthy and A. Ramanan, "Creating Compact and Discriminative Visual Vocabularies Using Visual Bits", In *Proceedings of IEEE International Conference on Digital Image Computing: Techniques and Applications*, pp.1-6, 2015.

[36] L. J. Latecki, R. Lakamper, and T. Eckhardt, "Shape Descriptors for Non-rigid Shapes with a Single Closed Contour", In *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp.424-429, 2000.

[37] J. Cui, M. Cui, B. Xiao, and G. Li, "Compact and Discriminative Representation of Bag-of-features", *Neurocomputing*, 169:55-67, 2015.

[38] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories". In *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp.2169-2178, 2006.

[39] G. L. Oliveira, E. R. Nascimento, A. W. Vieira, and M. F. Campos, "Sparse spatial coding: A novel approach for efficient and accurate object recognition", In *Proceedings of IEEE International Conference on Robotics and Automation*, pp.2592-2598, 2012.

[40] C. Lin Wei, F. Tsai Chih, Y. Chen Zong, and W. Ke Shih, "Keypoint Selection for Efficient Bag-of-words Feature Generation and Effective Image Classification", *Information Sciences*, 329:33-51, 2016.

[41] Y. Quan, Y. Xu, Y. Sun, Y. Huang, and H. Ji, "Sparse Coding for Classification via Discrimination Ensemble", In *Proceedings of IEEE International conference on Computer Vision and Pattern Recognition*, pp.5839-5847, 2016.

[42] X. Wang, Y. Lid, S. You, H. Li, and S. Wang, "Unidirectional representation based efficient dictionary learning", *IEEE Transactions on Circuits and Systems for Video Technology*, 30(1):59-74, 2018.

[43] I. F. J. Ghalyan, S. M. Chacko, and V. Kapila, "Simultaneous robustness against random initialization and optimal order selection in Bag-of-Words modeling", *Pattern Recognition Letters*, 116:135-142, 2018.

[44] M. Zang, D. Wen, T. Liu, H. Zou, and C. Liu, "A fast sparse coding method for image classification", *Applied Sciences*, 9(3):505, 2019.

[45] A. Ramanan and M. Niranjan, "A Review of Codebook Models in Patch-based Visual Object Recognition", *Journal of Signal Processing Systems*, 68(3):333-352, 2012.

[46] J. Winn, A. Criminisi, and T. Minka, "Object categorization by learned universal visual dictionary", In *Proceedings of Tenth IEEE International Conference on Computer Vision*, pp.1800-1807, 2005.

[47] A. Ramanan and M. Niranjan, "A One-pass Resource-allocating Codebook for Patch-based Visual Object Recognition", In *Proceedings of IEEE International Conference on Machine Learning for Signal Processing*, pp.35-40, 2010.

[48] F. Jurie and B. Triggs, "Creating Efficient Codebooks for Visual Recognition", In *Proceedings of 10th IEEE International Conference on Computer Vision*, pp.604-610, 2005.

[49] F. Moosmann, B. Trigg, and F. Jurie, "Fast Discriminative Visual Codebooks using Randomized Clustering Forests", In *Proceedings of Twentieth Annual Conference on Neural Information Processing Systems*, pp.985-992, 2006.

[50] Y. Zhao, G. Karypis, and U. Fayyad, "Hierarchical Clustering Algorithms for Document Datasets", *Data Mining and Knowledge Discovery*, 10(2):141-168, 2005.

[51] J. D. R. Farquhar, S. Szedmak, H. Meng, J. Shawe-Taylor, "Improving Bag-of-keypoints image categorisation: Generative models and PDF-kernels", In *LAVA report*, University of Southampton, 2005.

[52] S. Lazebnik, C. Schmid, and J. Ponce, "A Sparse Texture Representation Using Local Affine Regions", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8):1265-1278, 2005.

[53] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The PASCAL Visual Object Classes (VOC) Challenge", *International Journal of Computer Vision*, 88(2):303-338, 2010.

[54] L. Fei-Fei, R. Fergus, and P. Perona, "Learning Generative Visual Models From Few Training Examples: An Incremental Bayesian Approach Tested on 101 Object Categories", *Computer Vision and Image Understanding*, 106(1):59-70, 2007.

[55] K. H. Brodersen, C. S. Ong, K. E. Stephan, and J. M. Buhmann, "The Binormal Assumption on Precision-recall Curves", In *Proceedings of 20th International Conference on Pattern Recognition*, pp.4263-4266, 2010.

[56] L. Fei-Fei, R. Fergus, and P. Perona, "Learning generative visual models from few training examples: an incremental bayesian approach tested on 101 object categories", In *Proceedings of IEEE International Conference on CVPR Workshop of Generative Model Based Vision*, pp.59-70, 2004.