Image Classification by Spatial Pooling of Surf

Kalpana. S[1] Sivanandham. N[2]
[1]PG Scholar [2]Associate Professor
[1]Department of Communication and Networking [2]Department of Electronics and Communication
[1,2]P.B College of Engineering, Sriperumbuthur, India

Abstract— Today, there are many multimedia applications based on image understanding and processing, such as image retrieval, image classification, scene understanding, and so on. In image classification tasks, one of the most successful algorithms is the Bag-of-Features(BOF) model. Although the BOF model has many advantages, such as simplicity, generality, and scalability, it still suffers from several drawbacks, including the limited semantic description of local descriptors, lack of robust structures upon single visual words, and missing of efficient spatial weighting. To address these problems, the traditional BOF model is expanded on three aspects. First, a new scheme for combining texture and edge-based local features together at the descriptor extraction level. Next, to build geometric visual phrases to model spatial context upon complementary features for midlevel image representation. Finally, based on a smoothed edge map, a simple and effective spatial weighting scheme is performed to count occurrences of gradient orientation in localized portions of an image. Then test the framework by comparing the image with Fused Descriptors, which has the combination of SIFT, Edge-SIFT and Histogram of Oriented Gradient (HOG) descriptors for image classification and retrieval.

Key words: Pattern Classification, Retrieval Feature Extraction, Spatial Weighting, Performance Analysis, Fused Descriptors, Robustness Evaluation

I. INTRODUCTION

Image classification methods have been significantly developed in the last decade. Most methods stem from bag-of-features (BOF) approach and it is recently extended to a vector aggregation model, such as using Fisher kernels. Although the BOF model has many advantages, such as simplicity, generality, and scalability, it still suffers from several drawbacks, including the limited semantic description of local descriptors, lack of robust structures upon single visual words, and missing of efficient spatial weighting. To overcome these shortcomings, various techniques have been proposed, such as extracting multiple descriptors, spatial context modelling, and interest region detection. Though they have been proven to improve the BOF model to some extent, there still lacks a coherent scheme to integrate each individual module together. In this paper, a framework of feature extraction method for image classification is used. By following the Hybrid BOF(SIFT, Edge-SIFT and HOG) approach, a plenty of local descriptors are first extracted in an image and the proposed method is built upon the probability density function(p.d.f) formed by those descriptors. Since the p.d.f essentially represents the image, we extract the features from the image by means of the gradients on the p.d.f. The gradients, especially their orientations, effectively characterize the shape of the p.d.f from the geometrical viewpoint. The features of the histogram of oriented gradients are extracted via orientation coding followed by aggregation of the orientation codes. Integrating all the above coherently, a very powerful model that outperforms the combination of algorithms on image classifications, retrieval and understanding applications is obtained.

Containing these regions or the geometric relationships between them. There are global approaches which treat the image as a whole object and use all the information included in it. Many methods have been proposed that include the use of Eigen values, discrete cosine transform, and Gabor Wavelets These methods suffer from the size of the feature vector provided to the classifier. For this reason, many linear and nonlinear methods for vector size reduction are applied (PCA, LDA, ICA,). Hybrid approaches: The principle of these approaches is to imitate the human visual system, which uses both local and global features to recognize persons. The combination of these two methods has only one interest: to take advantage of the combined benefits of both approaches. Despite the number of researchers and the proposed methods, several factors can significantly affect the recognition performances, such as the pose, the presence/absence of structural components, occlusion, and illumination variations. The main aim of this project is to exhibit superior performances compared to the other existing methods of object recognition, scene classification and image retrieval using various datasets.

II. RELATED WORK

Here we review previous work, highlighting the concepts that will be exploited in our framework.

A. Spatial Bag-of-Features:

A taxonomy to capture the invariance of object translation, rotation, and scaling. Then the most representative features are selected based on a boosting-like method to generate a new bag-of-features-like vector representation of an image. The proposed retrieval framework works well in image retrieval task owing to the following three properties: 1) the encoding of geometric information of objects for capturing objects' spatial transformation, 2) the supervised feature selection and combination strategy for enhancing the discriminative power, and 3) the representation of bag-of-features for effective image matching and indexing for large scale image retrieval.

In this method the only the texture and colour features of the images are extracted and compared. Hence there may be chance of retrieving images which are not required or unmatched images. Poor performance for un-rigid object.

B. Feature Normalization For Part-Based Image Classification:

Classical performance evaluation methods, has introduced an efficient part-based Bag-of-Feature based on solid normalization parameters(power and coefficient), and two straight forward part-based properties, i.e., the independent assumption and the hierarchical-contribution assumption, to scale the feature super-vectors separately. Finally, algorithm is tested on challenging image sets, for general and fine-grained classification, to show its efficiency, scalability and adaptability in both scenarios. if the classifier is retrained [8], [10]].3 In both cases, during operation, testing data may follow a different distribution than that of training data. Therefore, robustness evaluation can not be carried out according to the classical paradigm of performance evaluation.

There is no any discussion on specialized normalization algorithms for part-based BOF models. May fail to work in severe distortion.

C. Hierarchical Matching Pursuit for Image Classification: Architecture and Fast Algorithms:

The framework gives a comprehensive analysis of Extracting good representations from images is essential for many computer vision tasks. In this paper, we propose hierarchical matching pursuit (HMP), which builds a feature hierarchy layer-by-layer using an efficient matching pursuit encoder. It includes three modules: batch orthogonal matching pursuit, spatial pyramid max pooling, and contrast normalization. We investigate the architecture of HMP, and show that all three components are critical for good performance. To speed up the orthogonal matching pursuit, we propose a batch tree orthogonal matching pursuit that is particularly suitable to encode a large number of observations that share the same large dictionary. HMP is scalable and can efficiently handle full-size images. In addition, HMP enables linear support vector machines (SVM) to match the performance of nonlinear SVM while being scalable to large datasets. We compare HMP with many state-of-the-art algorithms including convolutional deep belief networks, SIFT based single layer sparse coding, and kernel based feature learning. HMP consistently yields superior accuracy on three types of image classification problems: object recognition (Caltech-101), scene recognition (MIT-Scene), and static event recognition.

This system propose the matching pursuit encoder, and investigate its architecture and fast algorithms to compute sparse codes Builds a feature hierarchy layer-by-layer using an efficient matching pursuit encoder. The system has to improve the potential to become a new standard representation in image classification.

III. SYSTEM ANALYSIS

A. Existing System:

In this system, a novel framework based on the traditional BOF model for image classification in introduced. Three new modules are added into the BOF model to enhance its description power. First, to extract SIFT and Edge-SIFT descriptors from the original image and the corresponding edge map respectively, and then fuse them directly into a large set of descriptors. Experiments have revealed good compensation property of SIFT and Edge-SIFT descriptors. This system express a feature extraction method for image classification following the extended Bag-of-Feature approach. The local descriptors are first extracted in an image and the classification method is built upon the probability formed by those descriptors. The framework is tested by comparing the image with Fused Descriptors, which has the combination of SIFT and Edge descriptors. Moreover, fusing them at very early stage gives more opportunities for spatial context modeling. This method extends the traditional BOF model, by combining texture and edge-based local features together. Scale and orientation of local descriptors are not accurate, hence image matching is not efficient.

The information contained in (BoF) is not completely exploited. No better description of object's heterogeneous features such as texture and orientation.

B. Proposed System:

In our proposed system, to address the problems in extended BOF model , a new scheme for combining features such as texture, edge-based local features and localization of geometrical gradients together by using hybrid descriptor(BOF, SIFT, edge-SIFT, HOG). The Geometric Phrase Pooling(GPP) is used to build geometric visual phrases to model spatial context of features for image representation. In essential, the BOF model is a statistics based model aiming at providing better representation for images. For this purpose, local descriptors such as SIFT[4] are extracted from images, and a codebook is built upon all descriptors, depressing noises and forming a compact visual vocabulary for the dataset. Finally, descriptors are quantized onto the codebook, and visual words are pooled as a statistical histogram for image representation. The output of the BoF model could be applied for various tasks, such as image classification and image retrieval. Desired property in invariance in changes of scale, rotation, illumination.

Highly distinctive and descriptive in local patch. Especially effective in rigid object representation. Good compensation property. Good description for texture feature. Normalize the histogram. Pattern recognition and pattern classifier based spam filtering in the email service turn this proposed system more thriving and thus overcomes the drawbacks of existing system.

IV. A FRAMEWORK FOR EMPIRICAL EVALUATION OF BAG-OF-FEATURE

We propose here a framework for the empirical evaluation of Bag-of-Features (BoF) model is one of the most popular pipelines for image classification. In this section, we build a mathematical notation system for this model.

A. Local Descriptor Extraction:

We start with an original image I which is a W ×H matrix:

$$I = (a_{i\,j})_{W\times H} \qquad (1)$$

Where $(a_{i\,j})$ is the pixel on position (i, j).

Due to the limited semantic meaning of raw image pixels, we extract local descriptors from small patches on the image plane. There are many works on describing local patches.

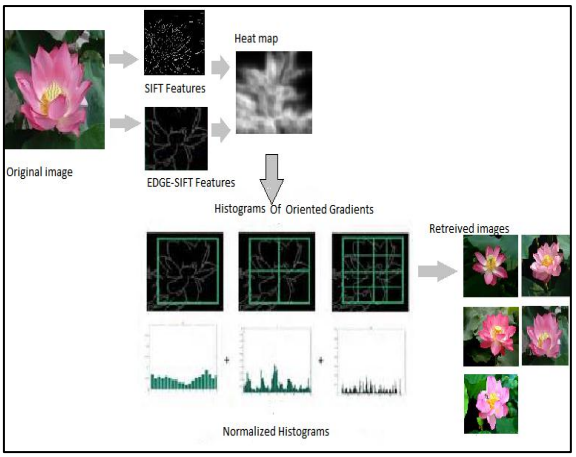

Fig. 1: Architecture Diagram

Among those, SIFT [5] and HOG [17] are probably the most widely used ones. They are both gradient-based histograms extracted on interest points of images. Detecting interest points is also a challenging problem. Since many detectors such as DoG [5] or MSER [18] sometimes fail to find semantic and discriminative patches, we use an alternative method by performing dense sampling of local patches, leading to the Dense-SIFT or Dense-HOG algorithm [19]. When the color information is useful for image understanding, it is also reasonable to calculate color SIFT descriptors, such as RGB-SIFT (calculating SIFT on red, green and blue channels individually), OpponentSIFT, HSV-SIFT, C-SIFT and so on. Among them, the OpponentSIFT descriptor is verified to outperform other ones in most cases [20]. After descriptor extraction, the image I could be represented as a set of local descriptors, M:

$$M = \{(d1, l1), (d2, l2), \ldots, (dM, lM)\} \quad (2)$$

Where dM and lM denote the D-dimensional description vector and the geometric location of the m-th descriptor, respectively. *M* is the total number of dense descriptors, which could be hundreds or even thousands under dense sampling. It is verified that SIFT and HOG descriptors are only good at describing texture features. To capture other important properties such as shape and color, it is reasonable to extract other kinds of descriptors. Systems with multiple types of descriptors [9] have been proposed, showing a much better performance over those using single type of descriptors.

B. Feature Pooling:

After all the local descriptors are quantized as visual words, we shall aggregate them for global image representation. We call this step feature pooling, for we are putting visual words into a pool for statistics. For this purpose, two major pooling strategies are often used. The max-pooling strategy calculates the maximal response on each codeword:

$$w = \max_{1 \le m \le M} V_m \quad (3)$$

Where the notation $\max_m$ denotes the element-wise maximization.

Differently, the average-pooling strategy calculates the average response:

$$w = 1/M \text{ sum of } W_m \quad (4)$$

Here w, a K-dimensional vector, is named representation vector or feature vector of the image.

Some researchers have discussed the choice of max-pooling versus average-pooling [10], showing that max-pooling gives more discriminative representation under soft quantization strategies, while average-pooling fits hard quantization better. Recently, various methods have been proposed to integrate both pooling methods to improve their effectiveness. For example, the Geometric p-norm Pooling algorithm [12] proposes a generalized p-norm pooling strategy and uses a complex optimization to find the best p for each image.

C. A Model of the Image Classification:

We consider the standard setting for classifier design E. Classification Models

Before sending the feature vectors into the SVM, feature normalization is considered a crucial data pre-processing step. One of the most popular feature normalization methods is the normalization, in which we divide each feature vector with its length in the p space so that all the vectors become lp-unit-length. In [16], the authors claim that lp-norm produces much lower classification accuracy than l2-norm, whereas it is verified in [13] that with a large enough normalization coefficient, the 1p-norm formula would give comparable performance with the l2-norm. In [13], the authors also discuss several advanced normalization methods for the part-based classification models.

Image classification tasks are usually configured with very long feature vectors and relatively smaller number of images. Therefore, the Support Vector Machine (SVM) is often taken as the default choice of classifier. It is verified that different choices of kernels would severely impact the classification accuracy, and the non-linear kernels such as ÷2 often gives higher performance than the linear inner-product kernel. However the latter one is proved more efficient and scalable [14], therefore is widely adopted in the classification tasks with large number of categories. In some cases, we can also use the Hellinger's kernel, or Bhattacharya's kernel, to produce feature vectors with less values falling in the close neighbourhood of 0 [35]. Although it is non-linear, we can still fit it in the linear model with a simple square-root transformation.

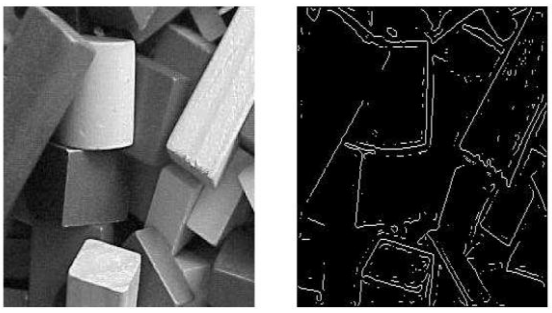

Fig. 2: Edge Map Extraction

D. Complementary Descriptors:

In this section, we propose a novel idea using complementary descriptors for image classification. First, we introduce a new kind of image descriptors named Edge-SIFT, which are extracted on the edgemap (boundary image) of the original image. Though SIFT and Edge-SIFT descriptors are calculated on different images, they share the same physical meaning, *i.e.*, histogram of gradients, in their corresponding dimension, therefore we could simply mix them on the image space to train the BoF model. We test our

model and provide detailed analysis on the effect of using multiple types of descriptors, and also make a short comment on the early fusion strategy to reveal its advantages. Finally, we discuss on the limitations of the proposed descriptor fusion strategy.validation or bootstrapping.

A. SIFT and Edge-SIFT Descriptors For a W×H image I, we extract dense SIFT [5] descriptors from the image. Denote the set of SIFT descriptors as MS:

$$M_S = \{(d_{S1}, l_{S1}) , (d_{S2}, l_{S2}) , \ldots , (d_{SMS} , l_{SMS})\} \quad (5)$$

where the subscript S stands for SIFT, and MS is the number of SIFT patches on the image plane. As we know, SIFT descriptors are effective on describing texture features, but less effective to capture the shape information. To overcome, we can introduce shape descriptors to help understanding the semantics. Following [9], we apply a boundary detector on image I, producing another W × H grayscale image IE:

$$IE = (e_{i\,j})_{W\times H} \quad (6)$$

Where $e_{i\ j}$, a floating value in [0, 1], is the significance quantity of pixel (i, j) located on an edge.We call IE the corresponding edgemap (boundary image) for the original image I. We use Compass Operator [13] for boundary detection. Some detected edgemaps are shown in Figs. On the edgemaps, texture details of the objects are filtered and the shape features become more clear. Therefore, it is reasonable to extract another set of SIFT descriptors on the edgemap for shape description. We call them Edge-SIFT descriptors to differ from the original SIFT descriptors. Denote the set of Edge-SIFT descriptors as ME:

$$M_E = \{(d_{E1}, l_{E1}) , (d_{E2}, l_{E2}) , \ldots , (d_{EME} , l_{EME})\} \quad (7)$$

Similarly, the subscript E stands for Edge-SIFT, and ME is the number of descriptors, which could be different from MS due to the different spatial strides and window sizes used in dense sampling. It is worth noting that both SIFT and Edge-SIFT descriptors are histograms of gradients, therefore they share the same physical meaning on the corresponding dimensions (histogram bins). We could naturally combine them together to capture both texture and shape features on the images.

E. Block Normalization:

Dalal and Triggs explore four different methods for block normalization. Let $v$be the non-normalized vector containing all histograms in a given block, $\|v\|_k$be its k-norm for $k = 1, 2$and $e$be some small constant (the exact value, hopefully, is unimportant). Then the normalization factor can be one of the following:

L2-norm:

$$f = \frac{v}{\sqrt{\|v\|_2^2 + e^2}} \quad (8)$$

L2-hys: L2-norm followed by clipping (limiting the maximum values of v to 0.2) and renormalizing, as in [8].

L1-norm:

$$f = \frac{v}{(\|v\|_1 + e)} \quad (9)$$

L1-sqrt:

$$f = \sqrt{\frac{v}{(\|v\|_1 + e)}} \quad (10)$$

In addition, the scheme L2-Hys can be computed by first taking the L2-norm, clipping the result, and then renormalizing. In their experiments, Dalal and Triggs found the L2-Hys, L2-norm, and L1-sqrt schemes provide similar performance, while the L1-norm provides slightly less reliable performance; however, all four methods showed very significant improvement over the non-normalized data.

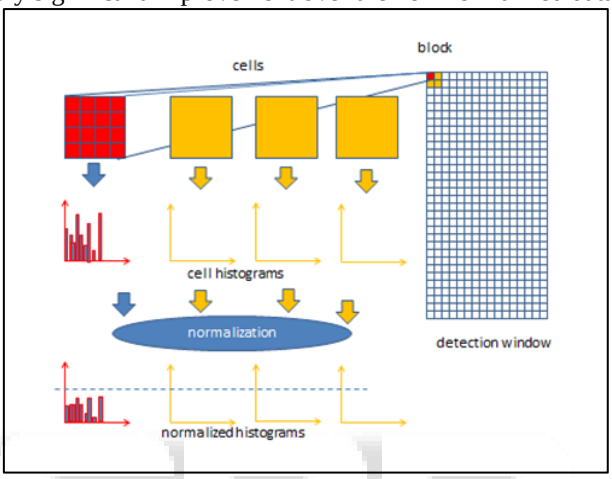

Fig. 3: HOG Implementation

F. Fusing Descriptors:

Following the basic idea, we simply unite the set of SIFT and Edge-SIFT descriptors on the image plane:

$$M = M_S \cup M_E \quad (11)$$

The difference between our model and [9] (best viewed in color PDF). The arrows and texts highlight the fusion step in each framework. Here, M is the union descriptor set for image representation. The number of descriptors in M is denoted as M, which satisfies M = M_S + M_E.

We illustrate the fusion operation on two sets of descriptors in Fig. Here we shall emphasize that the fusion process preserves both the description vector and location of original descriptors. It is shown later that this strategy gives us natural benefits by extracting geometric visual phrases consisting of both texture and shape visual words. Of course, except for the original image and boundary image (edgemap), one can extract more kinds of SIFT descriptors on other images such as saliency map or contour map. To fuse them, we only need to guarantee that all the descriptors share the same physical meaning on the corresponding dimensions, for we are comparing and combining the descriptors dimension-wise at the clustering and quantization steps. Throughout this paper, we only consider two set of SIFT descriptors extracted on the original and boundary images.

Finally, we shall make a short comment to compare our work with [9]. In [9], descriptors are also extracted from original and boundary images respectively. However, two

types of descriptors are individually processed through the BoF model, until fusion operation is performed on the pooled representation of both images to form a concatenated supervector. As we could see the late fusion limits the flexibility of the model, and makes it difficult to construct midlevel structures consisting of both kinds of descriptors. On the contrary, we finish the fusion step much earlier, leaving plenty of room for mid-level structures.

V. SPATIAL WEIGHTING

Traditional BoF model uses all the extracted local descriptors for image representation, however some of them might not fall on the objects we really want to recognize. Since such descriptors often introduce noises into the model, it is reasonable to filter them for better image understanding. This is equivalent to learning a spatial weighting, a saliency map, or simply a heatmap on the image plane. In this paper, we follow the observation that higher contrast regions provide stronger stimuli to vision, and propose a simple spatial weighting strategy through a Gaussian blur process on the boundary images. First of all, we calculate an edgemap (boundary image) for the original image I of size W×H. Following 12), the edgemap IE is another W×H matrix in which the elements represent the intensity of edge responses. We thereafter calculate a $W \times H$ weighting matrix W:

$$W = (w_{ij})_{W \times H} \qquad (12)$$

Here, w_{ij} is the spatial weight at position (i, j), which is accumulated from the decayed edge responses. The heatmaps generated by GPP and edgemap look similar. The difference lies in that, GPP enhances those local regions with co occurrence of similar features, whereas edgemap gives higher weight onto the regions with strong edge response. Both of them provide useful information for image classification.

Original $e = 0.01$ $e = 0.03$ $e = 0.10$ $e = 0.20$

Fig. 4: Heat Maps

Where *e* is smoothimg parameter.

It is worth noting that detecting the saliency regions on the image is itself an open problem in computer vision. Certainly, our algorithm could not completely solve the problem, but we provide a simple and efficient algorithm which provides useful information for image classification. To illustrate this, we calculate the classification accuracies by category, and list the most increased and decreased ones in Fig. 16. We can see that our algorithm works well on the situations with relatively simple background clutters, but could also harm the classification accuracy in the categories with poor saliency detection results. Since the proposed algorithm produces larger accuracy gains than drops, the averaged classification accuracy is boosted. Finally let us consider the time complexity of the proposed spatial weighting algorithm. It is easy to note that requires a complete enumeration on every pairs of pixels, therefore is very computational expensive, i.e., takes more than 30 seconds on a single-core CPU for a 300×300 image. To accelerate, we adopt an approximation by skipping the accumulation of the pairs with Euclidean distances larger than 50 pixels. Under the best smoothing parameter. With the reasonable approximation, our algorithm only requires less than 0.5 second on a single image, which is very efficient in practise considering the spatial weighting is computed only once.

VI. IMAGE RETRIEVAL

Content-based" means that the search analyzes the contents of the image rather than the metadata such as keywords, tags, or descriptions associated with the image. The term "content" in this context might refer to colours, shapes, textures, or any other information that can be derived from the image itself. CBIR is desirable because searches that rely purely on metadata are dependent on annotation quality and completeness. Having humans manually annotate images by entering keywords or metadata in a large database can be time consuming and may not capture the keywords desired to describe the image.

The evaluation of the effectiveness of keyword image search is subjective and has not been well-defined. In the same regard, CBIR systems have similar challenges in defining success.

A. Content Comparison Using Image Distance Measures:

The most common method for comparing two images in content-based image retrieval (typically an example image and an image from the database) is using an image distance measure. An image distance measure compares the similarity of two images in various dimensions such as colour, texture, shape, and others. For example a distance of 0 signifies an exact match with the query, with respect to the dimensions that were considered. As one may intuitively gather, a value greater than 0 indicates various degrees of similarities between the images. Search results then can be sorted based on their distance to the queried image. Many measures of image distance (Similarity Models) have been developed.

Content-based image retrieval is the task of searching images in databases by analyzing the image contents. In this demo, a simple image retrieval method is presented, based on the colour distribution of the images. The user simply provides an "example" image and the search is based upon that example (query by image example). For this first version of the demo no relevance feedback is used.

The 3D (HSV) histogram of the query image is computed. Then, the number of bins in each direction (i.e., HSV space)is duplicated by means of interpolation. For each image i in the database: Load its histogram Hist(i). Use interpolation for duplicating the number of bins in each direction. For each 3-D histogram bin, compute the distance (D) between the hist of the query image and the i-th database image. Keep only distances (D2) for which, the respective histogram bins of the query image are larger than a predefined threshold T (let L2 the number of these distances). Use a 2nd threshold: find the distance (D3) values which are smaller than T2, and let L3 be the number of such values. The similarity measure is defined as: S(i) = L2 * average(D3) / (L3^2). Sort the similarity vector and prompt the user with the images that have the M smaller S values.

VII. EXPERIMENTAL RESULTS

In this section, we show the experimental results on several publicly available image classification datasets. To compare our method with other works, we inherit the same settings from the state-of-the-art algorithms, and adopt descriptor fusion, GPP, and spatial weighting with the best settings learned from the previous sections.

1) Boundary detection. We use the Compass Operator [13] for boundary detection. The radius parameter σ is fixed as 4 as proposed in the same literature.
2) Image descriptors. We use the VLFeat [42] library to extract dense SIFT descriptors. The spatial stride and window size are discussed individually for each dataset.
3) Codebook construction. We use K-Means for clustering. The codebook size is 4096 for Caltech256, 8192 for Pascal VOC 2007, and 2048 for others. The number of descriptors for clustering does not exceed 2 million.
4) Coding and phrase pooling. We use LLC [16] for local feature coding and apply GPP with the best parameters.
5) Spatial weighting. We take $\sigma e = 0.05$ for the edge-based spatial weighting scheme.
6) Spatial Pyramid and normalization. We apply a 3-layer $(1 \times 1 + 2 \times 2 + 4 \times 4)$ SPM for enhancing the global spatial context. After that, an $l2$-norm normalization is performed to produce comparable feature vectors.
7) SVM for classification. We use LibLINEAR [14], a recent scalable SVM implementation for training and testing. For the Pascal VOC retrieval task, we rank the testing images according to their confident scores.
8) Accuracy evaluation. For the Pascal VOC 2007 Challenge, we use the standard benchmark [?]. On other datasets, we select fixed numbers of images for training the classification model, and test it on the remaining images to calculate the average classification accuracy over all the categories. We repeat the random selection 10 times and report the averaged results.

VIII. CONTRIBUTIONS, LIMITATIONS AND OPEN ISSUES

In this paper we focused on empirical method based on multiple criteria thus provides better approximation of the image when modelling image features. SURF (Speeded Up Robust Features) is a robust local feature detector that can be used in computer vision tasks like object recognition or 3D reconstruction. It is partly inspired by the SIFT descriptor. The standard version of SURF is several times faster than SIFT and to be more robust against different image transformations than SIFT. SURF is based on sums of three descriptor responses and makes an efficient use of integral images. It uses an integer approximation to the feature detector, which can be computed extremely quickly with an integral image (3 integer operations). For features, it uses the sum of the colour, texture, contrast, edge-map, gradient of localization response around the point of interest. Again, these can be computed with the aid of the integral image. A simple and effective spatial weighting scheme to detect regions-of-interest is proposed. Integrating all the above features coherently, a very powerful model that outperforms the state-of-the-art algorithms on various image classifications tasks is obtained.

Despite the excellent accuracy gain we have obtained, there are still some open problems in our framework. It is verified that objects are better described by complementary features such as texture and shape. However, calculating SIFT descriptors on the boundary images directly is not the best way for shape description. Hope sto investigate these problems in the future towards a more powerful classification model. Neural network concept can also be introduced to extract more number of features and provide perfect shape and texture for the image. By introducing the above method, we obtain a very powerful model of image classifications and retrieval which gives superior performances compared to the other existing methods.

REFERENCES

[1] L. Bo, X. Ren, and D. Fox, "Hierarchical matching pursuit for image classification: Architecture and fast algorithms," Neural Inf. Process. Syst., vol. 1, no. 2, pp. 1–6, 2011.
[2] A. Bosch, A. Zisserman, and X. Muoz, "Image classification using random forests and ferns," in Proc. 11th Comput. Vis. Pattern Recognit., 2007, pp. 1–8.
[3] E. Nowak, F. Jurie, and B. Triggs, "Sampling strategies for bag-offeatures image classification," in Proc. Comput. Vis. Pattern Recognit., 2006, pp. 490–503.
[4] Y. Cao, C. Wang, Z. Li, L. Zhang, and L. Zhang, "Spatial-bag-offeatures," in Proc. Comput. Vis. Pattern Recognit., 2010, pp. 3352–3359.
[5] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," Int. J. Comput. Vis., vol. 60, no. 2, pp. 91–110, 2004.
[6] D. Nister and H. Stewenius, "Scalable recognition with a vocabulary tree," in Proc. Comput. Vis. Pattern Recognit., 2006, pp. 2161–2168.
[7] Y. Lu, L. Zhang, J. Liu, and Q. Tian, "Constructing lexica of high-level concepts with small semantic gap," IEEE Trans. Multimedia, vol. 12, no. 4, pp. 288–299, Jul. 2010.
[8] L. Xie, Q. Tian, and B. Zhang, "Spatial pooling of heterogeneous features for image applications," in Proc. 20th ACM Multimedia, 2012, pp. 539–548.
[9] M. Marszalek and C. Schmid, "Spatial weighting for bag-of-features," in Proc. Comput. Vis. Pattern Recognit., 2006, pp. 2118–2125.
[10] Y.-L. Boureau, F. Bach, Y. LeCun, and J. Ponce, "Learning mid-level features for recognition," in Proc. Comput. Vis. Pattern Recognit., 2010,pp. 2559–2566.
[11] Y. Zhang, Z. Jia, and T. Chen, "Image retrieval with geometry-preserving visual phrases," in Proc. Comput. Vis. Pattern Recognit., 2011, pp. 809–816.
[12] J. Feng, B. Ni, Q. Tian, and S. Yan, "Geometric p-norm feature pooling for image classification," in Proc. Comput. Vis. Pattern Recognit., 2011,pp. 2609–2704.
[13] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing

natural scene categories," in Proc. Comput. Vis. Pattern Recognit., 2006, pp. 2169–2178.
[14] J. Canny, "A computational approach to edge detection," IEEE Trans. Pattern Anal. Mach. Intell., vol. 8, no. 6, pp. 679–698, Jun. 1986.
[15] M. Ruzon and C. Tomasi, "Color edge detection with the compass operator," in Proc. Comput. Vis. Pattern Recognit., 1999, pp. 160–166.
[16] J. Yuan, Y. Wu, and M. Yang, "Discovery of collocation patterns: From visual words to visual phrases," in Proc. Comput. Vis. Pattern Recognit., 2007, pp. 1–8.
[17] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in Proc. Comput. Vis. Pattern Recognit., 2005, pp. 886–893.
[18] J. Matas, O. Chum, M. Urban, and T. Pajdla, "Robust wide-baseline stereo from maximally stable extremal regions," Image Vis. Comput., vol. 22, no. 10, pp. 761–767, 2004.
[19] A. Bosch, A. Zisserman, and X. Munoz, "Scene classification via pLSA," in Proc. Int. Conf. Comput. Vis., 2006, pp. 517–530.
[20] K. Van De Sande, T. Gevers, and C. Snoek, "Evaluating color descriptors for object and scene recognition," IEEE Tran. Pattern Anal. Mach. Intell., vol. 32, no. 9, pp. 1582–1596, Sep. 2010.
[21] D. Li, L. Yang, X. Hua, and H. Zhang, "Large-scale robust visual codebook construction," in Proc. ACM Multimedia, 2010, pp. 1183–1186.
[22] Q. Tian, S. Zhang, W. Zhou, R. Ji, B. Ni, and N. Sebe, "Building descriptive and discriminative visual codebook for large-scale image applications," Multimedia Tools Appl., vol. 51, no. 2, pp. 441–477, 2011.
[23] S. Zhang, Q. Huang, G. Hua, S. Jiang, W. Gao, and Q. Tian, "Building contextual visual vocabulary for large-scale image applications," in Proc. Int. Conf. Multimedia, 2010, pp. 501–510.
[24] H. Lee, A. Battle, R. Raina, and A. Ng, "Efficient sparse coding algorithms," in Proc. Neural Inf. Process. Syst., 2007, pp. 801–805.
[25] J. Yang, K. Yu, Y. Gong, and T. Huang, "Linear spatial pyramid matching using sparse coding for image classification," in Proc. Comput.Vis. Pattern Recognit., 2009, pp. 1794–1801.
[26] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong, "Localityconstrained linear coding for image classification," in Proc. Comput. Vis. Pattern Recognit., 2010, pp. 3360–3367.
[27] W. Zhou, Y. Lu, H. Li, Y. Song, and Q. Tian, "Spatial coding for large scale partial-duplicate web image search," in Proc. Int. Conf. Multimedia, 2010, pp. 511–520.
[28] L. Xie, J. Wang, B. Zhang, and Q. Tian, "Orientational pyramid matching for recognizing indoor scenes," in Proc. Comput. Vis. Pattern Recognit., 2014, pp. 1–4.
[29] 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," Comput. Vis. Image Understand., vol. 106, no. 1, pp. 59–70, 2007.
[30] D. Liu, G. Hua, P. Viola, and T. Chen, "Integrated feature selection and higher-order spatial feature extraction for object categorization," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2008, pp. 1–8.
[31] S. Zhang, Q. Tian, G. Hua, Q. Huang, and S. Li, "Descriptive visual words and visual phrases for image applications," in Proc. 17th ACM Int. Conf. Multimedia, 2009, pp. 75–84.
[32] J. Yuan, M. Yang, and Y. Wu, "Mining discriminative co-occurrence patterns for visual recognition," in Proc. Comput. Vis. Pattern Recognit., Jun. 2011, pp. 2777–2784.in Intrusion Detection, ser. LNCS. Springer, 2007, pp. 42–62.
[33] L. Xie, Q. Tian, and B. Zhang, "Feature normalization for part-based image classification," in Proc. Int. Conf. Image Process., 2013, pp. 1–3.
[34] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "LIBLINEAR: A library for large linear classification," J. Mach. Learn. Res., 2008, pp. 1817–1874.
[35] F. Perronnin, J. Sánchez, and T. Mensink, "Improving the fisher kernel for large-scale image classification," in Proc. Eur. Conf. Comput. Vis. 2010, pp. 143–156.
[36] M. Nilsback and A. Zisserman, "A visual vocabulary for flower classification," in Proc. Comput. Vis. Pattern Recognit., 2006, pp. 1447–1454.
[37] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie, "The caltech-UCSD birds-200-2011 dataset," California Inst. Technol., Pasadena, CA, USA, Tech. Rep. CNS-TR-2011-001, 2011.
[38] L. Xie, Q. Tian, R. Hong, S. Yan, and B. Zhang, "Hierarchical part matching for fine-grained visual categorization," in Proc. Int. Conf. Comput. Vis., 2013, pp. 1–8.
[39] Y. Chai, V. Lempitsky, and A. Zisserman, "Symbiotic segmentation and part localization for fine-grained categorization," in Proc. Int. Conf Comput. Vis., 2013, pp. 1–10.
[40] E. Gavves, B. Fernando, C. Snoek, A. Smeulders, and T. Tuytelaars, "Fine-grained categorization by alignments," in Proc. Int. Conf. Comput. Vis., 2013, pp. 1–10.
[41] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," Pattern Anal. Mach. Intell., vol. 20, no. 11, pp. 1254–1259, 1998.
[42] A. Vedaldi and B. Fulkerson, "VLFeat: An open and portable library of computer vision algorithms," in Proc. Multimedia, 2010, pp. 1469–1472.
[43] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL visual object classes challenge 2007 (VOC2007) results," 2011.
[44] G. Griffin, A. Holub, and P. Perona, "Caltech-256 object category dataset," California Inst. Technol., Pasadena, CA, USA, Tech. Rep. CNSTR- 2007-001, 2007.
[45] S. Gao, I. Tsang, and L. Chia, "Kernel sparse representation for image classification and face recognition," in Proc. Eur. Conf. Comput. Vis., 2010, pp. 1–14.

[46] S. Lazebnik, C. Schmid, and J. Ponce, "Semi-local affine parts for object recognition," in Proc. Brit. Mach. Vis. Conf., 2004, pp. 959–968.
[47] D. Larlus and F. Jurie, "Latent mixture vocabularies for object categorization and segmentation," Image Vis. Comput., vol. 27, no. 5,pp. 523–534, 2009.
[48] P. Gehler and S. Nowozin, "On feature combination for multiclass object classification," in Proc. 12th Int. Conf. Comput. Vis., 2009, pp. 221–228.
[49] L. Li, H. Su, E. Xing, and L. Fei-Fei, "Object bank: A high-level image representation for scene classification and semantic feature sparsification," Neuron Inf. Process. Syst., 2010, pp. 1–3.
[50] A. Quattoni and A. Torralba, "Recognizing indoor scenes," in Proc. Comput. Vis. Pattern Recognit., 2009, pp. 413–420.

