LLC Revisit: Scene Classification with $k$-Farthest Neighbours

Katsuyuki TANAKA, Nonmember, Tetsuya TAKIGUCHI, and Yasuo ARIKI, Members

SUMMARY This paper introduces a simple but effective way to boost the performance of scene classification through a novel approach to the LLC coding process. In our proposed method, a local descriptor is encoded not only with $k$-nearest visual words but also with $k$-farthest visual words to produce more discriminative code. Since the proposed method is a simple modification of the image classification model, it can be easily integrated into various existing BoF models proposed in various areas, such as coding, pooling, to boost their scene classification performance. The results of experiments conducted with three scene datasets: 15-Scenes, MIT-Indoor67, and Sun367 show that adding $k$-farthest visual words better enhances scene classification performance than increasing the number of $k$-nearest visual words.

key words: scene classification, bag-of-words

1. Introduction

The bag-of-features (BoF) model made a success in image classification in recent years [2], [24]. Its simple mechanism and good classification performance made it a popular choice for image classification task, therefore, many BoF-based models have been introduced [1], [8], [27], [30] to date.

The typical BoF model consists of three process: i) extracting local descriptors (SIFT [15], HOG [3] etc.), ii) coding, and iii) pooling. As a result, extracted local descriptors of an image are transferred into codes with a codebook as a reference, and codes are transferred into a global descriptor, which represents an image with a single vector. Hence, the main interest in image classification is optimisation of a global descriptor construction, which is effective in producing a similar global descriptor with similar images that discriminate other images well. In the BoF model, the coding process is one of the important issues that affect the performance of image classification.

In common BoF model, a codebook consists of visual words which are representatives of similar local descriptors. Common practice of generating codebook is by grouping similar local descriptors with $K$-means and use centroid of each cluster as a visual word.

In coding process, each local descriptor of an image is encoded into a code, which is expressed by a combination of visual words in codebook. It can be considered as a summarisation of a local descriptor information, hence it is important to make as accurate summarisation as possible to create a good or discriminative code.

The simplest coding process is the hard-assignment coding method (HVQ) [2]. In HVQ, a local descriptor is encoded to the closest visual word (only one non-zero element per code). Yang [30] (SC) and Wang [27] (LLC) successfully improved the image classification performance by incorporating a sparsity constraint and locality constraint into the coding process. These methods assign a local descriptor not to one visual word but to some visual words to compensate the quantisation loss of HVQ.

Many literature reported that supervised way of generating a codebook [18], [31], incorporating spatial neighbourhoods of local descriptors [1], [16], and using multiple kind of local descriptors [7], [28] (such as RGB, texture, SIFT, HOG) could improve the performance of image classification. Moreover, utilising region of interest (ROI) or parts formation [20] are active area of research and these technique are reported to improve the performance of image classification.

Improving the coding process is beneficial to various image classification models, hence, the motivation of this research is to find a simple way of boosting the performance of image classification in coding process, so that it can be plug into various image classification methods.

In this paper, we propose a novel coding process based on LLC to improve the performance of scene classification. Scene classification is a challenging problem in computer vision since a scene not only contains various combinations of objects with frequent occlusions, but also these objects varies in size, location, viewing angle etc. even within the same scene class. Even though LLC well describes the similarities of local descriptors by using $k$-nearest visual words, these natures of scenes questioned us it may not be optimal solution to use the similarities of local descriptors only to capture their characteristics and/or concept.

In our proposed method, a local descriptor is encoded not only with $k$-nearest visual words but also with $k$-farthest visual words. We call this method a $k$-Farthest Neighbours ($kFN$) approach since it is an opposite concept with $k$-Nearest Neighbours ($kNN$) approach. We evaluate our proposed method using three scene datasets: 15-Scenes, MIT-Indoor67, and Sun367. The experimental results show
that the proposed coding approach significantly improve the performance of scene classification. A summary of the contributions of our proposed method is as follows:

- We found that adding $k$-farthest visual words is better approximation of locality-constrained [27].
- We found that adding $k$-farthest visual words provides the significant effect to boost the performance of scene classification than increasing the number of $k$-nearest visual words.
- The proposed coding approach is easy to implement and since it does not go beyond the simple BoF model, it can be easily applied to many different coding and pooling process [1], [8], [16], [21] to improve the performance by simply replacing their coding process.

The rest of the paper is organised as follows. In Sect. 2, we describe the related works with notations used in this paper. In Sect. 3, we explain how to encode local descriptors into codes with the proposed $k$FN approach. Finally in Sect. 4, we experimentally show the effectiveness of our proposed method with various scene datasets.

2. Related Works

In this section, we introduce the common flow of coding process of BoF model with notations used in this paper.

Let $V = \{v_k \in \mathbb{R}^D\}_{k=1..K}$ be a codebook with $K$ visual words and $X = \{x_m \in \mathbb{R}^D\}_{m=1..M}$ be a set of $D$-dimensional local descriptors extracted at $M$ locations from an image $I$. The purpose of coding process is to quantise $x_m$ into a $K$-dimensional code $u_m$ by finding some coefficient $u \in \mathbb{R}^K$ that approximate $x_m$ with $V$ as reference. As a results, all local descriptors in $X$ are converted into codes $U = \{u_m \in \mathbb{R}^K\}_{m=1..M}$. It is important to define the quantisation strategies that best describes $x_m$ to obtain the better classification performance.

The simplest coding is HVQ [2], which produces a code $u_m$ by assigning a local descriptor $x_m$ to the nearest visual word $v_k$. This can be formulated as Eq. (1), where $u_m \geq 0$ means all the elements of $u_m$ is non-negative and each code has only one non-zero element $\|u_m\|_0 = 1$. $\| \cdot \|_1$, $\| \cdot \|_2$ denote the $l1$, $l2$-norm respectively. However, the extreme sparse code with only one non-zero element may be ill-posed. This is because it is hardly able to express the rich information of local descriptors with such restrictive constraint, hence the quantisation loss of local descriptors is inevitable.

$$u_m = \arg\min_u \|x_m - Vu\|_2^2$$  \hspace{1cm} (1)

$$\text{s.t. } \|u_m\|_0 = 1, \|u_m\|_1 = 1, u_m \geq 0, \forall m$$

Soft-assignment coding improve the hard-assignment by representing $x_m$ by the distance between $x_m$ and multiple visual words [25] (where $\beta$ controls the softness of soft assignment).

$$u_{m,j} = \frac{\exp(-\beta\|x_m - v_k\|_2^2)}{\sum_{k=1}^{K} \exp(-\beta\|x_m - v_k\|_2^2)}$$  \hspace{1cm} (2)

SC uses $l1$-norm regularization on $u_m$ to enforce $u_m$ to have a small number of non-zero elements as formulated in Eq. (3). The so called sparsity is controlled by a parameter $\lambda$.

$$u_m = \arg\min_u \|x_m - Vu\|_2^2 + \lambda\|u\|_1$$  \hspace{1cm} (3)

LLC [32] enforces locality constraint rather than sparsity. The second term in Eq. (4) is the locality constraint, which enforces $d_m \in \mathbb{R}^K$ to $u_m$, where $\odot$ denotes the element-wise multiplication (the Hadamard product). Equation (5) is the calculation of $d_m$ and $\operatorname{dist}(x_m, v_k)$ denotes the Euclidean distance between $x_m$ and $v_k$. It penalises the visual words according to the distance from $x_m$. $\sigma$ is a parameter to control the speed of weight decay of $d_m$. Due to the locality constraint, LLC assigns similar visual words to similar local descriptors, whereas sparsity constraint of SC may choose quite different visual words for similar local descriptors. Effectively, it enables to retain the correlation between similar local descriptors in codes.

$$u_m = \arg\min_u \|x_m - Vu\|_2^2 + \lambda\|d_m \odot u\|_2^2$$  \hspace{1cm} (4)

$$\text{s.t. } 1^Tu_m = 1, \forall m$$

$$d_m = \exp\left[\frac{\sum_{k=1}^{K} \|\operatorname{dist}(x_m, v_k)\|_2^2}{\sigma}\right]$$  \hspace{1cm} (5)

3. $k$-Farthest Neighbours Approach

Since only a few of Eq. (4) has significant values, [27] suggests to solve LLC by only using $k$-nearest visual word neighbours ($k < D < K$) of $x_m$ as a set of local bases $V_{NN}$. Compacting the size of codebook and discarding the constrained of Eq. (4) speed up the algorithm and reduce the computation cost.

$$u_m = \arg\min_u \|x_m - V_{NN}u\|_2^2$$  \hspace{1cm} (6)

$$\text{s.t. } \|u_m\|_0 = k, 1^Tu_m = 1, \forall m$$

In our proposed method, we suggest a slightly different approach to encode a local descriptor $x_m$. We not only use $NN$ of $x_m$ but also use $k$-farthest visual word neighbours of $x_m$ as a set of local based $[V_{NN}, V_{FN}]$ to produce $u_m$. To avoid the confusion, we refer the size of neighbours $k$ for $NN$ and $FN$ as $knn$ and $kfn$ respectively, therefore, the number of non-zero elements of $u_m$ becomes $knn + kfn$.

$$u_m = \arg\min_u \|x_m - [V_{NN}, V_{FN}]u\|_2^2$$  \hspace{1cm} (7)

$$\text{s.t. } \|u_m\|_0 = knn + kfn, 1^Tu_m = 1, \forall m$$

We apply common pooling strategies of spatial pyramid and max pooling to summarizes $U$ to obtain a global descriptor of an image. Suppose we have $L$ regions $R_l \in \mathbb{R}^L$ of spatial pyramid, $R_l$ is represented as a single vector $I_l \in \mathbb{R}^K$ by pooling the codes located within $R_l$. As it is described in Eq. (8), each element $I_{lk}$ takes the maximum value of $u_m$.
located within $R_t$. A global descriptor is obtain by concatenating the representation of all regions $[I_1, \ldots, I_t]$.

$$I_{lk} = \max_{m \in R_t} [u_m], \quad k = K$$

(8)

In BoF model, visual words in codebook represent the general view of local descriptors and a code is a reflection of a local descriptor expressed by a combination of visual words. To achieve good classification performance, it is important to find the way to transform similar local descriptors into similar codes. Capturing the characteristics of a particular image in coding process is helpful to construct a global descriptor that discriminates other class of images well, hence it is important to define the coding process properly in order to preserve the information of a local descriptor.

While LLC succeeds to capture some information of a local descriptor by $kNN$ approach, our approach attempts to produce even more discriminative code by incorporating $k$-farthest visual words in coding. Intuitively, $kNN$ describes the similarity of a local descriptor $x_m$ while $kFN$ describes the dissimilarity. Surprisingly, adding just a few bits of additional $FN$ improve the performance of scene classification, and it turns out the produced codes have better discriminative powers than even adding more $NN$ in coding.

The effectiveness of our proposed method in scene classification can be derived from the following two properties, which effectively produce more discriminative code representation:

- Adding $kFN$ smooths the responses of $kNN$
- Having the responses of $kFN$ produces consistent patterns in code

In LLC, the distances between a local descriptor and each visual words in codebook decide the value of each element of $u_m$. We call the value of each element of $u_m$ a response of each visual word, and using different neighbors give different responses. We argue that adding $kFN$ has a good influence on the responses of $kNN$, effectively, $kFN$ method produces more discriminative code than just using $kNN$.

Also, even though the responses of $kFN$ is small, the influence of the responses of $kFN$ is not negligible in coding process. No matter what the responses, it is important to make a code which characterise the similar images well and discriminate dissimilar images. We believe that $kFN$ method produces more discriminative code representation than normal LLC, effectively producing more discriminative global feature to represent an image.

In addition, the advantage of the proposed method is simple to re-implement so that it can be easily applied to various image classification technique, such as pooling, parts model, multi-type descriptors to further improve the classification performance.

4. Experimental Results

The performance of our proposed method is evaluated by conducting scene classification with the following experimental setup.

DataSets: We use 15-Scenes [12], MIT-Indoor67 [19], and Sun397 [28].

Local Descriptors: For all experiments, we use SIFT as a local descriptor and they are densely sampled over every 8 pixels with a single scale of $16 \times 16$ patches. We use VLFeat toolbox [26] to extract these local descriptors.

Coding: We construct codebook with $K = 4096$ visual words by $K$-means for all experiments. We measure three different results of the proposed approach by increasing the size of $kfn = 5$, 10, 15 with fixed size of $knn = 5$ (so the non-zero elements of a code is 10, 15, and 20 respectively).

Pooling: To preserve spatial information, the codes are pooled with three levels of pyramids ($1 \times 1$, $2 \times 2$, $4 \times 4$) [12] and max pooling [23] is used to generate a global descriptor of an image for all experiments.

Evaluation: We compare the performance of $kFN$ approach with $kNN$ based LLC as baselines. We evaluate the performance of our proposed method by how much adding $FN$ improve over basic LLC ($knn = 5$). To evaluate the effect of different neighbourhoods, $FN$ and $NN$, in scene classification, we compare the same number of non-zero elements in $u_m$ by increasing the size of $knn$ and $kfn$ by 5 on basic LLC with $knn = 5$. Even though many literature reported the image classification results with these datasets, it is difficult to capture subtle details to re-produce their results. Hence, for fair evaluation, we compare the performance of $kFN$ approach with LLC performed by exactly the same experimental setup. We use a linear SVM [30] to train classifiers with one-vs-rest with default parameter. All experiments are run 10 times over 10 random splits of training and test data. We report the average of top-1 accuracy of each run as the classification performance for each dataset.

4.1 15-Scenes

The 15-Scenes consists with 15 classes of scenes such as kitchen and street. The number of images of each class varies from 200 to 400. In this experiment, we use 100 images for training and the rest for testing by following the standard setup. Results are shown in Fig. 1a.

The classification rate of $kFN$ approach scored 80.21% when $kfn = 5$ which improved LLC with $knn = 5$ by 0.47%. It shows that increasing the number of $FN$ increase the performance of classification, it scored 80.77% when $kfn = 15$. Increasing the number of $NN$ does not improve much when $knn = 10$, while it start showing the more improvement when adding more $NN$, yet $kFN$ approach provides a slightly better performance.

4.2 MIT-Indoor67

The MIT-Indoor67 provides a larger scene images than 15-Scenes, consist with 15620 indoor images over 67 different categories such as bookstore, florist, corridors, and museum. What makes this dataset challenging is that it is not only providing larger scene categories but also indoor scenes may be more difficult to characterises than outdoor scenes [19].
By following the experiment setup in [19], we randomly selected 80 images for training and 20 images from the remaining for testing in this experiment and report the results in Fig. 1b.

The performance differences becomes more prominent in MIT-Indoor67 dataset. The results of adding $kfn = 5, 10, 15$ improved LLC with $knn = 5$ by 0.99%, 1.49%, and 2.16%. It also outperform $kNN$ approach with the same number of non-zero elements, LLC with $knn = 10, 15, 20$, by 1.33%, 1.37%, and 1.85%. Adding more $FN$ constantly increase the performance of classification while adding more $NN$ does not have much effect in this dataset.

4.3 Sun397

One of the most challenging and large dataset on scene classification to date is the Sun397. It provides a wide range of both indoor and outdoor scenes over 397 categories with more than 100 images per category. The experiments are conducted by using the same training and testing partitions provided at the Sun database website.[8] Figure 1c shows the results of this experiment.

Again, the experimental on Sun397 dataset shows similar results to prior datasets. It scored 32.14% when $kfn = 5$ which improves LLC with $knn = 5$ by 0.8%. Comparison between $kFN$ approach and $kNN$ approach in terms of the same number of non-zero elements shows that adding $FN$ is more effective to scene classification than adding $NN$. This means that the improvement is not due to the size of non-zero non-zero elements in a code. Increasing the number of $kfn$ and $knn$ does not show as much effect as initial big jump, it scored highest 32.48% when $kfn = 15$.

5. Analysis

We conduct further experiments to investigate the effectiveness of our proposed method in different setup using MIT-Indoor67 dataset.

5.1 Effect of Different Neighbourhoods

The concept of adding more visual words to describe $x_m$ in coding is not limited by nearest and/or farthest neighbours. We explore the potential of the middle-rang visual word neighbours of $x_m$ in scene classification in this section. We define $k$-middle-rang visual words as a half distance between nearest and farthest visual words and use these visual words as a set of local bases $V_{MN}$ for local bases, that is to use a set of local based $[V_{NN}, V_{MN}]$ to produce $u_m$ in Eq. (7).

We measure three classification results with different size of $knn = 5, 10, 15$ and results are shown in Fig. 2.

The results shows that it outperform $kNN$ based model and the performance decreased with the size of $knn$. Interestingly, the results supports that the concept of adding $FN$ is more effective in scene classification.

---

[8] http://vision.princeton.edu/projects/2010/SUN/
5.2 Effect of kFN with Different Size of Codebook

We tried various sizes of codebook $K = 1024, 2048, 3072, 4096$ to investigate the effect of the codebook size. We conducted the experiment with three different neighbourhoods, $knn = 5, 10$ and $knn, kfn = 5$. As it is shown in Fig. 3, the performance increase with the size of $K$ and peaked when $K = 4096$. The kFN approach outperform others in all size of $K$ and shows a larger performance improvement as the size of $K$ increases.

5.3 Effectiveness of kFN Method

As it is shown is our experiment, incorporating kFN with $kNN$ improve the performance of scene classification. The explanation of the improvement of scene classification can be derived by the combination of the following two characteristics of our proposed method, which effectively produces more discriminative code representation:

- Adding kFN smooths the responses of $kNN$
- Having the responses of kFN produces consistent patterns in code

In LLC, the responses in $um$ are determined by the distances between a local descriptor and each visual words in codebook hence using different neighbours, $kNN$ and/or $kFN$, give different responses. We argues that adding kFN has a good influence on responses of $kNN$, effectively, kFN method produces more discriminative code than just using $kNN$.

This can be justified by investigating the performance of classification by switching off the responses of kFN. We performed the three experiments by setting the responses of kFN to zero after performing LLC with $knn$, $kfn = 5, 10, 15$. These are equivalent with LLC with $knn = 5$ but different responses in each code due to the influence of kFN. As it is shown in Fig. 4, kFN method without the responses of kFN outperform LLC with $knn = 5$ but different responses in each code due to the influence of kFN. As it is shown in Fig. 4, kFN method without the responses of kFN outperform LLC with $knn = 5$, this indicate that kFN smooth the responses of $kNN$ and producing better code representation. (As we also performed an experiment by setting the responses of farther 5 neighbors of $kNN$ to zero after performing LLC with $knn = 10$, it scored 44.74%. This result also support the effectiveness of kFN as better smoothing factors.)

Another reason why kFN is effective in scene classification is due to the influence of the responses of kFN itself in $um$. Since the responses of kFN is small, it may be suppressed by the larger responses of $kNN$ in max pooling. However, a code is quite sparse when the size of codebook is large as well as the size of $knn$ and $kfn$ is small, hence, some of the responses of kFN may survive in max pooling. These survivors aid to produce better code representation and ef-
effectively constructing more discriminative image statistic.

This can be explained by analysing the Fig. 6, which shows the distribution of the rate of local descriptors assigned over 4096 visual words (MIT-Indoor67, clothing-store, 1.jpg) using $kNN$ and $kFN$ approach. Interestingly, the behaviour is quite different between adding more $FN$ and $NN$. As it is shown in the Figs. 6 c and 6 d, the distribution is quit skewed by comparing with LLC with $knn = 5$ (Fig. 6 a), while increasing in $NN$ does not have much differences in distribution (Fig. 6 b). This implies that the responses of $kFN$ are assigned to the number of fixed visual words, therefore, it produces consistent patterns in code. This aid to produce more discriminative codes that $kNN$ could not produce by itself.

Also the results of Fig. 4 support the existence of the responses of $kFN$ influence in scene classification since the performance of $kFN$ method including the responses of $kFN$ outperforms discluding the responses of $kFN$, 47.23% and 45.98% respectively when $knn = 5$, $kfn = 15$.

As the results, no matter what the responses of codes, $kFN$ method produce more discriminative codes which better characterise the similar images well and discriminate dissimilar images than using only $kFN$. We believe that these characteristics of $kFN$ approach possibly influence to produce better code representation as well as effective way of constructing the statistic of spatial information, effectively producing more discriminative global feature to represent an image.

5.4 Effect of $kFN$ with Different Coding Framework

Improved Fisher vector (IFV) [22] and VLAD [9] models use slightly different approach to build a global descriptor. The concept of IFV model is how to fit $x_m$ by modifying the parameters of given generative model. VLAD is considered as a simplified version of IFV. IFV models consistently shows high performance than LLC based classification models [10], [22]. It would be interesting to investigate if the concept of $kFN$ approach can be applicable to these frameworks.

We use VLAD to investigate the effectiveness $kFN$. Suppose $x_m$ is assign to nearest visual word $v_k$ ($NN(x_m) = v_k$), VLAD is obtained by concatenating the sum of the differences $\mu_k$ as described in Eq. (9) for all visual words $[\mu_k]_{k=1..K}$:

$$\mu_k = \sum_{NN(x_m)=v_k} x_m - v_k$$  \hspace{1cm} (9)

The concept of $kFN$ can be integrated into VLAD by simply replacing the $NN$ strategy with $kNN$ and $kFN$. In this
since a technique to improve the performance of image classification. We state the classification performance directly refers to the mean. To see the effect of the density of local descriptors, we use SIFT sampled by 8 pixels with a single scale (Step8&Scale1) of 4 as well as 4 pixels with four scales of 4, 6, 8, 10 (Step4&Scale4). The results are shown in Fig. 5 with the best scores of our proposed kFN method with LLC (knn = 5, kfn = 15).

The experimental results showed interesting behavioural differences with the sampling rate of local descriptors. With Step8&Scale1, kFN method with LLC (47.23%) outperforms any VLAD based implementations. However, IFV and VLAD with NN significantly improved the performance with denser sampled with more scales. Both of them improved by about 8% while kFN method with LLC only improved by 1.69%. Even though IFV consists of richer information, the performance difference with kFN method with LLC was only 1.4% with Step8&Scale1, but the difference dramatically increased with Step4&Scale4 (7.49%). This may be because VLAD and IFV is a generative model which utilise the distribution of local descriptors with respect to the visual words, the qualities and quantities of local descriptors may be important factor to obtain the good performances. The results also indicated that the current way of using kFN in VLAD is not as effective as it used in LLC since SNN showed similar performance. Though a similar modification can be made with IFV, investigating the different way of using kFN in these frameworks is left as future work.

5.5 Extending Our Method

The BoF model is an active area of image classification research, so various coding and/or pooling methods have been introduced to improve the performance of image classification. We state the classification performance directly referred from literature in Table 1, just to introduce a few of the previous results with our best scores.

Regardless of the computational efficiency or the complication of the algorithm, there are some common factors reported to improve the performance of image classification.

For example, using multiple descriptor channels, such as RGB, SIFT, GIST etc. [7], [28], and/or incorporating spatial neighbours of local descriptors such as s Laplacian Sparse Coding (LSC) [6], Macrofeatures [1], and Local Pairwise codebook (LPC) [16] are reported to improve the classification performance. The drawback of these methods may be that it is sometimes not straightforward how to mix the descriptor channels and/or incorporate descriptors to gain the classification performance.

Extracting discriminative characteristic of an image by utilising region of interest (ROI) or parts based models, such as Latent Pyramid Regions (LPR) [21] and Distinctive Parts model (BoP) [10] etc. [17], [19], and/or background information [20] are active areas of research. Even though some of them require parts labels and searching for representative parts is computationally demanding, they are consistently reported to be effective technique to improve the performance of image classification.

Many different pooling methods such as Local Pooling (LP) [29] and Discriminative Spatial Pyramid (DSP) [8] are also introduced to improve the performance of image classification.

There are a lot of room left to improve the performance of kFN approach by incorporating some of the tech-

Table 1 Average classification rate of various image classification models on 15-Scenes, MIT-Indoor67, and Sun397 with the best score of kFN approach

| Method                  | 15-Scenes | MIT-Indoor67 | Sun397 |
|-------------------------|-----------|--------------|--------|
| Macrofeatures [1]       | 84.30     | -            | -      |
| LSC [6]                 | 89.75     | -            | -      |
| Local Pooling [29]      | 83.30     | -            | -      |
| DSP [8]                 | 81.81     | -            | -      |
| Liu [14]                | 82.70     | -            | -      |
| MIT-Indoor67 [19]       | -         | 26.50        | -      |
| LPC [16]                | 83.40     | 38.36        | -      |
| Object Bank [13]        | 80.90     | 37.60        | -      |
| Pandey [17]             | -         | 43.10        | -      |
| LPR [21]                | 85.81     | 44.84        | -      |
| BoP [10]                | -         | 63.10        | -      |
| LLC-BoP [10]            | -         | 56.66        | -      |
| Sun397 [28]             | 81.20     | -            | 38.0   |
| IFV [22]                | -         | -            | 47.20  |
| IFV-BoP [10]            | -         | 63.10        | -      |
| **Proposed kFN method** | **79.74** | **45.07**    | **31.44** |

**LLC (ours setup)**

**Proposed kFN method**

80.77 47.23 32.48
nique above, or even with simple encoder like LLC and a single descriptor channel, well tuned coding process could improve the performance†, and possibly outperform some of the complicated models mentioned above as it is stated in [10]. However, investigating the effect of various models with $kFN$ approach is beyond the scope of our research so that they are left as future works.

Our results are no match with some of the state-of-the-art results [11], [33], especially using deep learning could score far better classification rate. However, the purpose of this research is not to score best of the best or tuning the best combinations of various image classification techniques but to introduce the concept of $kFN$ approach which has not been considered before for scene classification. Importantly, one of the advantages of the proposed $kFN$ approach is that, since it is simply and easy to re-implement, it can be easily applicable with many different image classification models by replacing coding process with $kFN$. The performance-wise, the value of the proposed method may not have much impact on the stage of scene classification since the improvement is not strong enough. However, we believe that by spot-lighting on $kFN$ may open up the new concept on the stage of scene classification so that other researcher could give different roles for further improve. In that sense we hope the proposed method has some significance value and impact on scene classification.

6. Conclusion

In this paper, we proposed a novel approach by integrating $k$-farthest visual words into LLC coding process. Experiments on various scene datasets showed that our novel BoF model can effectively improve scene classification performance. Since the proposed method is a simple modification of the BoF model, it can be easily integrated into many different existing image classification frameworks to improve their performance.

The effectiveness of the proposed approach should be further evaluated with various different datasets such as PASCAL VOC2007 dataset [5] and large object image dataset such as ImageNet [4]. We also like to investigate how much our proposed method could improve the classification performance by applying combination of various existing coding and/or pooling process, such as multiple type of local descriptors, incorporating spatial neighbourhoods of local descriptors, as well as using ROI models.

References

[1] Y.-L. Boureau, F. Bach, Y. LeCun, and J. Ponce, Learning mid-level features for recognition, pp.2559–2566, 2010.
[2] G. Csurka, C.R. Dance, L. Fan, J. Willamowski, and C. Bray, “Visual categorization with bags of keypoints,” in ECCV’04, pp.1–22, 2004.
[3] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” In CVPR’05, pp.886–893, 2005.
[4] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “ImageNet: A large-scale hierarchical image database,” In CVPR’09, pp.248–255, 2009.
[5] M. Everingham, L.V. Gool, C.K.I. Williams, J. Winn, and A. Zisserman, “The pascal visual object classes (voc) challenge,” Int. J. Comput. Vision, vol.88, no.2, pp.303–338, 2010.
[6] S. Gao, I.W.-H. Tsang, L.-T. Chia, and P. Zhao, “Local features are not lonely - laplacian sparse coding for image classification,” In CVPR’10, pp.3555–3561, 2010.
[7] G. Griffin, A. Holub, and P. Perona, “Caltech-256 object category dataset,” Technical Report 7694, California Institute of Technology, 2007.
[8] T. Harada, Y. Ushiku, Y. Yamashita, and Y. Kuniyoshi, “Discriminative spatial pyramid,” In CVPR’11, pp.1617–1624, 2011.
[9] H. Jégou, M. Douze, C. Schmid, and P. Pérez, “Aggregating local descriptors into a compact image representation,” In CVPR’10, pp.3304–3311, 2010.
[10] M. Juneja, A. Vedaldi, C.V. Jawahar, and A. Zisserman, “Blocks that shout: Distinctive parts for scene classification,” In CVPR’13, pp.923–930, 2013.
[11] A. Krizhevsky, I. Sutskever, and G. Hinton, “Imagenet classification with deep convolutional neural networks,” In NIPS’12, 2012.
[12] S. Lazebnik, C. Schmid, and J. Ponce, “Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories,” In CVPR’06, pp.2169–2178, 2006.
[13] L.-J. Li, H. Su, E.P. Xing, and F.-F. Li, “Object bank: A high-level image representation for scene classification & semantic feature sparsification,” In NIPS’10, pp.1378–1386, 2010.
[14] L. Liu, L. Wang, and X. Liu, “In defense of soft-assignment coding,” In ICCV’11, pp.2486–2493, 2011.
[15] D.G. Lowe, “Distinctive image features from scale-invariant key-points,” Int. J. Comput. Vision, vol.60, no.2, pp.91–110, 2004.
[16] N. Morioka and S. Satoh, “Building compact local pairwise codebook with joint feature space clustering,” In ECCV’10, vol.6311, pp.692–705, 2010.
[17] M. Pandey and S. Lazebnik, “Scene recognition and weakly supervised object localization with deformable part-based models,” In ICCV’11, pp.1307–1314, 2011.
[18] F. Perronnin, C. Dance, G. Csurka, and M. Bressan, “Adapted vocabularies for generic visual categorization,” In ECCV’06, vol.3954, pp.464–475, 2006.
[19] A. Quattoni and A. Torralba, “Recognizing indoor scenes,” In CVPR’09, pp.413–420, 2009.
[20] O. Russakovsky, Y. Lin, K. Yu, and L. Fei-Fei, “Object-centric spatial pooling for image classification,” In ECCV’12, vol.7573, pp.1–15, 2012.
[21] F. Sadeghi and M.F. Tappen, “Latent pyramidal regions for recognizing scenes,” In ECCV’12, vol.7576, pp.228–241, 2012.
[22] J. Sánchez, F. Perronnin, T. Mensink, and J. Verbeek, “Image classification with the fisher vector: Theory and practice,” Int. J. Comput. Vision, vol.105, no.3, pp.222–245, 2013.
[23] T. Serre, L. Wolf, and T. Poggio, “Object recognition with features inspired by visual cortex,” In CVPR’05, pp.994–1000, 2005.
[24] J. Sivic and A. Zisserman, “Video google: A text retrieval approach to object matching in videos,” In ICCV’03, vol.2, pp.1470–1477, 2003.
[25] J.C. van Gemert, C.J. Veenman, A.W.M. Smeulders, and J.-M. Geusebroek, “Visual word ambiguity,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.32, no.7, pp.1271–1283, 2010.
[26] A. Vedaldi and B. Fulkerson, VLFeat: An open and portable library of computer vision algorithms, pp.1469–1472, 2010.
[27] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong, “Locality-constrained linear coding for image classification,” In CVPR’10, pp.3360–3367, 2010.

† As an example, more densely sampled SIFT over every 4 pixels with four scales of 4, 6, 8, 10 SIFT bins measures 48.8% while 8 pixels with a single scale of 4 measures 46.06% when $knn = 5$, $kFN = 5$. 

IEICE TRANS. INF. & SYST., VOL.E99–D, NO.5 MAY 2016
[28] J. Xiao, J. Hays, K.A. Ehinger, A. Oliva, and A. Torralba, “Sun database: Large scale scene recognition from abbey to zoo,” 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp.3485–3492, 2010.
[29] Y.-L. Boureau, N.L. Roux, F. Bach, J. Ponce, and Y. LeCun, “Ask the locals: multi-way local pooling for image recognition,” In ICCV’11, pp.2651–2658, 2011.
[30] J. Yang, K. Yu, Y. Gong, and T. Huang, “Linear spatial pyramid matching using sparse coding for image classification,” In CVPR’09, pp.1794–1801, 2009.
[31] L. Yang, R. Jin, R. Sukthankar, and F. Jurie, “Unifying discriminative visual codebook generation with classifier training for object category recognition,” In CVPR’08, pp.1–8, 2008.
[32] K. Yu, T. Zhang, and Y. Gong, “Nonlinear learning using local coordinate coding,” NIPS’09: Learning, 22(x):1–9, 2009.
[33] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva, “Learning deep features for scene recognition using places database,” In NIPS’14, pp.487–495, 2014.

Katsuyuki Tanaka  B.E. and M.E. in Computer Science and Engineering, University of New South Wales, Australia, and PhD in Kobe University. The research interests are in QA, image classification, and various information retrieval including text and image.

Tetsuya Takiguchi  received his B.S. degree in applied mathematics from Okayama University of Science, Okayama, Japan, in 1994, and his M.E. and Dr. of Engineering degrees in information science from Nara Institute of Science and Technology, Nara, Japan, in 1996 and 1999, respectively. From 1999 to 2004, he was a researcher at IBM Research, Tokyo Research Laboratory, Kanagawa, Japan. He is currently an associate professor at Kobe University. His research interests include acoustic and image signal processing and pattern recognition. He received the Awaya Award from the Acoustical Society of Japan in 2002. He is a member of IEEE, IPSJ, and ASJ.

Yauso Ariki  received his B.E., M.E., and Ph.D. degrees in information science from Kyoto University in 1974, 1976, and 1979, respectively. He was an assistant professor at Kyoto University from 1980 to 1990 and stayed at Edinburgh University as a visiting academic from 1987 to 1990. From 1990 to 1992 he was an associate professor and from 1992 to 2003 a professor at Ryukoku University. Since 2003 he has been a professor at Kobe University. He is mainly engaged in speech and image recognition and is interested in information retrieval and databases. He is a member of IEEE, IPSJ, JSAI, ITE, and IIEEJ.