Multiple Stage Residual Model for Accurate Image Classification

Song Bai, Xinggang Wang, Cong Yao, Xiang Bai

Department of Electronics and Information Engineering
Huazhong University of Science and Technology, PR China
{songbai, xgwang}@hust.edu.cn, raoconghust@gmail.com, xbai@hust.edu.cn

Abstract. Image classification is an important topic in computer vision. As a key procedure, encoding the local features to get a compact representation for image affects the final classification accuracy largely. There is no doubt that encoding procedure leads to information loss, due to the existence of quantization error. The residual vector, defined as the difference between the local image feature and its corresponding visual word, is the chief culprit that should be responsible for the quantization error. Many previous algorithms consider it as a coding issue, and focus on reducing the quantization error by reconstructing the feature with more than one visual words, or by the so-called soft-assignment strategy. In this paper, we consider the problem from a different view, and propose an effective and efficient model, which is called Multiple Stage Residual Model (MSRM), to make full use of the residual vector to generate a multiple stage code. Our proposed model is a generic framework, which can be built upon many coding algorithms and improves the image classification performance of the coding algorithms significantly. The experimental results on the image classification benchmarks, such as UIUC 8-Sport, Scene-15, Caltech-101 image dataset, confirm the validity of MSRM.

1 Introduction

Image classification is an important topic in computer vision with many applications, such as image retrieval [1, 2], video retrieval and web content analysis [3]. Given an input image, the aim of image classification is to assign one or more class labels to it, or in other words, to determine its category. The Bag-of-Features (BoF) [4, 5] model may be the most successful framework in image classification for its invariance to scale, translation and rotation.

The pipeline of a typical BoF image classification model is illustrated in Fig. 1. It consists of five basic steps: patch extraction, patch description, codebook learning, feature coding and feature pooling. With an input image in hand, the step of patch extraction is to generate lots of small patches via dense sampling, which are described by some local image descriptors in the patch description procedure. Various descriptors, such as SIFT [6] or HoG [7] can be used to describe these local patches. In the process of codebook learning, a subset of

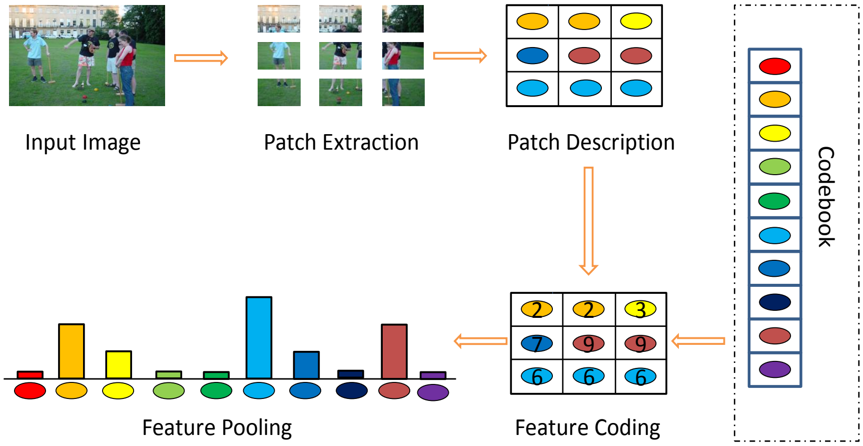


Fig. 1. The pipeline of general image classification framework.

features randomly sampled in the training images are gathered to learn the codebook by some codebook learning algorithms (*e.g.* K-means [8]). Feature coding offers the way to generate the code for each local image descriptor, and in the feature pooling step, these codes are pooled together to get the final representation of an image.

Among the aforementioned five steps, feature coding plays an important role for its great impact on the accuracy and speed of image classification. Recently, many coding algorithms [9–20] have been proposed. The representative coding method is Hard-assignment Coding (HC) [9]. HC only accounts for the nearest visual word of a local feature for coding, which makes HC sensitive to the selection of the codebook. Meanwhile due to its severe quantization error, HC cannot give a satisfactory classification result. In order to reduce the information loss in the quantization procedure, Localized Soft-assignment Coding (LSC) [11] adopts an "early cut-off" strategy, and assign the local feature to more than one visual words. The response coefficient for each visual word is determined by the distance between the local feature and the visual word. Different from voting-based algorithms like HC, Sparse Coding [18] shows its superiority gradually, but it is time-consuming. Locality-constrained Linear Coding (LLC) [10], as a typical sparse coding method, attaches more importance to locality than sparsity, and offers an efficient way to compute the approximate sparse code. Salient Coding (SC) [16] introduces the concept of "salience", and guarantees a salient representation without deviations. Low-Rank Sparse Coding (LRSC) [12] enforces sparsity in feature codes, locality in codebook construction, and low-rankness for spatial consistency by solving an optimization problem with nuclear norm and sparsity inducing $\mathcal{L}_1$ norm.

After the codes of all local features are computed, the feature pooling step is adopted to integrate these codes together to generate an equal sized feature vector for each image in the database. The common used pooling methods are sum-pooling [9] and max-pooling [18]. A comprehensive analysis can be found in [21–23]. Meanwhile, in order to include the spatial information in the pooling step, Spatial Pyramid Matching (SPM) [9] is conducted via dividing the image into increasingly finer subregions. Each subregion is pooled individually, and all pooled features are concatenated to form the final feature vector of the whole image.

Note that all the algorithms mentioned above consider reducing the quantization error, caused by the residual vector (the difference between the local feature and its corresponding visual word), as a feature coding issue. However in this paper, we propose a model called Multiple Stage Residual Model (MSRM) to make full use of the residual vector instead. We prove that our proposed model leads to less information loss compared with HC [9], and can achieve a better performance. Meanwhile, we manage separating the vector quantization process from feature coding process, and making MSRM a generic framework by introducing various state-of-the-art coding algorithms [9, 10, 16, 24] to MSRM. We also observe that the code of each stage shows some properties of complementarity, and discriminative classifiers, such as SVM, can be used to select several representative stages to get a higher classification accuracy. MSRM consists of several concatenated codebooks, and the output of each stage is the input of the next stage, which is simple and fast to compute.

The rest of this paper is organized as follows: In Section 2 we introduce some related work briefly. The introduction of MSRM is given in Section 3. In Section 4, we carry out several experiments on three benchmark datasets, and the experimental results prove the effectiveness of the proposed model. Conclusions are given in Section 5.

2 Related Work

Considering that MSRM is a generic model, in which many encoding methods can be embedded, we review some typical coding strategies recently proposed in the literature. These coding strategies act as a baseline, and a full comparison will be conducted in Section 4.

Based on the classification standard for coding methods in [25], these methods are grouped into five categories: voting-based methods, reconstruction-based methods, saliency-based methods, local tangent-based methods, and fisher coding-based methods.

Let $\mathcal{C} = \{c_1, c_2, \ldots, c_n\}$ $(1 \leq i \leq n, c_i \in \mathbf{R}^d)$ denote the codebook previously learned in the training set, and n represents the codebook size. In the case of extracting SIFT as the local image descriptor, the dimension d is usually 128. $x_i \in \mathbf{R}^d$ denotes the ith feature densely extracted in an image. Let w_i be the code of x_i, and w_{ij} be the response value of x_i with respect to c_j.

Hard-assignment Coding: Hard-assignment coding (HC) [9] is a representative of voting-based methods. For each local descriptor x_i in an image, HC assigns it to the nearest visual word in the codebook under a certain metric. It means there is only one non-zero element in w_i. w_{ij} satisfies

$$w_{ij} = \begin{cases} 1 & if \quad j = \underset{j=1,2,\ldots,n}{\arg\min} \left\|x_i - c_j\right\|_2^2 \\ 0 & otherwise \end{cases}$$

when $\mathcal{L}_2$ metric is used.

Locality-constrained Linear Coding: Locality-constrained Linear Coding (LLC) [10] is a typical example of reconstruction-based methods. Unlike traditional sparse coding methods, LLC emphasizes the importance of locality instead of sparsity, since locality must lead to sparsity but not necessary vice versa. Specifically, LLC solves the following optimization:

$$\begin{aligned} & w_i = \arg\min \left\|x_i - cw_i\right\| + \lambda \left\|d_i \odot w_i\right\| \\ & s.t. \quad 1^{\mathrm{T}} w_i = 1 \end{aligned}$$

where $d_i \in \mathbf{R}^n$ is the locality adaptor defined as

$$d_i = \exp\left(\frac{dist(x_i, \mathcal{C})}{\sigma}\right)$$

where $dist(x_i, \mathcal{C})$ is the Euclidean distance between x_i and each visual word c_j in $\mathcal{C}$.

A fast approximation of LLC is proposed in [10] to improve the computational efficiency.

Salient Coding: Salient Coding (SC) [16] is a representative method of saliency-based methods. SC deems that saliency is a fundamental property in coding, and define a "saliency" degree based on the nearest visual word c_j to x_i

$$w_{ij} = \begin{cases} \Phi(x_i, c_j) & if \quad j = \underset{j=1,2,\ldots,n}{\arg\min} \left\|x_i - c_j\right\|_2^2 \\ 0 & otherwise \end{cases}$$

where Φ is a monotonically decreasing function, usually defined as

$$\Phi(x_i, c_j) = 1 - \frac{\left\|x_i - c_j\right\|_2}{\frac{1}{K-1} \sum_{k \neq j}^{K} \left\|x_i - c_k\right\|_2}$$

Vector of Local Aggregated Descriptors: Super Vector Coding (SVC) [15] is a representative of local tangent-based methods. Super Vector Coding considers feature coding as a manifold approximation using the visual words by assuming that all features constitute a smooth manifold.

Improved Fisher Kernel (IFK) [14] is a representative of fisher coding-based methods. In IFK, the probability density distribution of the local features is described by the Gaussian mixture models.

Vector of Local Aggregated Descriptors (VLAD) [24, 26] aggregates the local features based on a locality criterion in feature space, which is defied as

$$w_{ij} = \begin{cases} x_i - c_j & if \quad j = \underset{j=1,2,\ldots,n}{\arg\min} \; \|x_i - c_j\|_2^2 \\ 0 & otherwise \end{cases}$$

Note that the code w_{ij} is a vector with the size of d.

It is known that VLAD is deemed as a simplified and non-probabilistic version of IFK, and becomes SVC if combined with BoF. Although VLAD is initially designed for large scale image retrieval, we prove it also effective in image classification as shown in Section 4. Considering the simpleness of VLAD, we adopts VLAD to get a compact representation for image throughout our experiments.

Other Related Algorithms: Spatial Pyramid Matching (SPM) [9] has been proved to be effective in the pooling procedure with spatial information included. SPM starts with dividing an image into subregions, and obtains the histogram $\mathcal{H}_l = [h_{l1}, h_{l2}, \ldots, h_{ln}]$ of each region via a pooling function

$$\mathcal{H}_l = \mathcal{F}\left(\{w_i | (\alpha_{x_i}, \beta_{x_i}) \in \mathcal{S}_l\}\right)$$

where α_{x_i} and β_{x_i} are the x-coordinate and y-coordinate of x_i in the image.

Usually, the pooling function $\mathcal{F}$ is max-pooling which selects the largest response value along each dimension of all the codes in a certain region $\mathcal{S}_l$

$$h_{lj} = max\{w_{ij} \,| (\alpha_{x_i}, \beta_{x_i}) \in \mathcal{S}_l\}$$

or sum-pooling that simply adds all the values

$$h_{lj} = \sum\{w_{ij} \,| (\alpha_{x_i}, \beta_{x_i}) \in \mathcal{S}_l\}$$

The "pyramid" means that the spatial division of image ranges from a global one, *i.e.* the entire image, to several local subregions. The final image representation is obtained by concatenating these histograms together.

Other algorithms, such as Spatial Local Coding (SLC) [27], Feature Context [28], are also widely-used for modelling the spatial information.

3 Multiple Stage Residual Model

Multiple Stage Residual Model has one codebook in each stage, and each stage will output a code with an encoder. The detail of MSRM is as follows.

3.1 Preliminary

Multiple Stage Vector Quantization (MSVQ) is a classic channel coding algorithm commonly used in Digital Voice Processing.

The theory of MSVQ is as follows: (1) Given an input signal represented by a vector x and the multiple stage codebook $\mathcal{C} = \{\mathcal{C}^1, \mathcal{C}^2, \ldots, \mathcal{C}^m\}$, where m is

Algorithm 1 Multiple Stage Codebook Learning with K-means

```
Input: The traning features $\mathcal{X}$ for codebook learning; The codebook size $n$; The number of stage $m$.
Output: The learned codebook $\mathcal{C} = \{\mathcal{C}^1, \mathcal{C}^2, \ldots, \mathcal{C}^m\}$.
for each $j \in [1, m]$ do
    divide $\mathcal{X}$ into $n$ clusters via K-means through Eq. 1 and Eq. 2;
    and output the cluster centers $\mathcal{C}^j$;
    for each $x \in \mathcal{X}$ do
        compute the residual vector $r(x) = x - q(x)$;
        $x = r(x)$;
    end for
end for
```

the number of stage. Each component $\mathcal{C}^j$ is the codebook in the jth stage of $\mathcal{C}$ with codebook size n (2) in the jth stage, the c_i^j with the minimum distortion is determined, and the subscript i, as well as the stage number j, is passed into channel (3) the input of the next stage, *i.e.* the $(j+1)$th stage, is the residual vector $r(x) = x - c_i^j$. Following the same principle, c_i^{j+1} is determined again (4) the procedure of (2)(3) is iteratively conducted, until the final stage is reached (5) in the receiving terminal, the decoder reconstructs the signal by using the subscripts and the multiple stage codebook.

In this paper, we try to propose a specifically designed model similar to MSVQ for large scale image classification. One of our goals is the generality of the model, and we want to adapt as many as state-of-the-art feature coding methods to this model.

3.2 Codebook Learning in MSRM

Codebook learning is a necessary step before encoding the local features. There are various codebook learning algorithms in an unsupervised way [8], a weakly-supervised way [29], or a supervised way [30, 31].

Among all the codebook learning algorithms, K-means may be the most widely used one for its simpleness and stableness. Given a randomly selected subset $\mathcal{X}$ of SIFT descriptors of the training set and the codebook size n, K-means seeks n vectors $\mathcal{C} = \{c_1, c_2, \ldots, c_n\}$ iteratively, and minimizes the approximation error $\mathcal{E}$ defined as

$$\mathcal{E} = \sum_{x \in \mathcal{X}} \|x - q(x)\|^2 \tag{1}$$

$$x \rightarrow q(x) = arg \min_{c \in C} \|x - c\|^2 \tag{2}$$

Our proposed model also adopts K-means to learn the multiple stage codebook $\mathcal{C} = \{\mathcal{C}^j, 1 \leq j \leq m\}$. The pseudocode is presented in Algorithm 1.

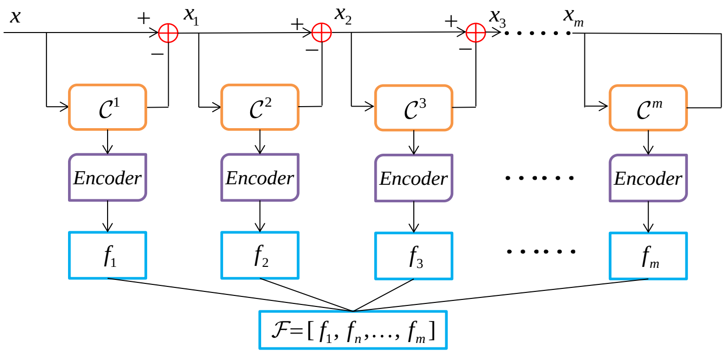


Fig. 2. The pipeline of Multiple Stage Residual Model. The input x is the local descriptor to be encoded. $\mathcal{C}^i$ $(1 \leq i \leq m)$ is the ith stage codebook. The encoder can be various. f_j $(1 \leq j \leq m)$ is the encoded feature via the encoder. The final representation for x is the concatenation of the output from all stages.

3.3 Encoder in MSRM

As is presented in Section 2, many different coding strategies were proposed. In order to embed these coding strategies into our proposed model, we separate the vector quantization procedure from the feature coding procedure. Specifically, for a given local image descriptor x, on the one hand we only consider the nearest visual word to compute the residual vector $r(x) = x - q(x)$ according to Eq. 2 and pass the residual vector to the next stage, which is the new feature vector to be encoded in the future. On the other hand, we does not restrict the way and the number of the visual words used to generate the code for x, which is usually determined by the coding algorithm. For example, if LLC [10] is chosen as the encoder of MSRM, we use k (k is usually set to 5) visual words to encode x, and use only the nearest word to generate the residual vector. Our interpretation is that the nearest visual word to x captures its main pattern, and all the local image descriptors lying in the same cluster will eliminate the information redundancy if all of them are deprived with their common pattern. The operations of computing the residual vector and encoding the features are iteratively conducted until the final stage is reached. The pipeline of Multiple Stage Residual Model is illustrated in Fig. 2.

According to the taxonomy presented in [25], we introduce some typical algorithms as the encoder to our proposed model. Specifically, Hard-assignment Coding [9] is the representative of voting-based coding methods. Local-constrained Linear Coding [10] is the representative of reconstruction-based coding methods. Salient Coding [16] is the representative of saliency-based coding methods. Vector of Aggregated Local Descriptors [24] is used to replace the role of Super Vector-coding [15] (one of local tangent-based coding methods), and Improve Fisher Kernel [14] (one of fisher coding-based methods). When the stage number m is set to 1, our proposed model degenerates into the original encoding algorithms.

As shown in Section 4, the codes from different stages are greatly complementary to each other. If the classifier cannot give the image a right label prediction with the codes from a certain stage, the prediction can be revised with the usage of the codes from other stages in most cases.

4 Experiments

In this section, we first evaluate the effectiveness of MSRM on two benchmarks particularly collected for scene classification. We will give a comparison with the original algorithms to show the extent that MSRM can improve the baseline on Scene-15 dataset [9], UIUC 8-sport dataset [32]. An extra experiment is also conducted in Caltech-101 dataset [33] to evaluate the performance of MSRM in object recognition.

4.1 Implementation Details

If not specified, we adopt the following setup for all of our experiments. Note that some results of a certain coding method offered by us may be different from the results reported in the original papers or in the survey articles [25, 34], since different settings lead to different results. For example, SC [16] conducts the experiment in Scene-15 dataset under 4096 codes, HC in [25] adopts a Hellingers kernel to boost its performance. In order to get a proper assessment of MSRM, we re-implement the experiments of HC [9], SC [16], LLC [10], VLAD [24] according the following rule. The comparisons to the original results from the original papers are also conducted.

Feature Extraction: The standard dense SIFT is extracted on a patch size 32×32, with step size fixed to 4 pixels, by using the *vl_dsift* command available in the public toolbox VLFeat [35].

Codebook Generation: Following the instruction described in Section 3.2, we learning a multiple stage codebook via k-means clustering. The codebook size depends on the coding algorithm that applied to MSRM. In particular, HC, LLC and SC adopt a relative larger codebook with a size of 1024, and 64 for VLAD respectively.

Coding and Pooling: As for the implementation of VQ, LLC and SC, we use the codes released by Huang in [25] to encode the local features. In order to include the spatial information, SPM [9] with 3 levels: 1×1 , 2×2 and 4×4 is adopted with a same weight for each level. We use *vl_vlad* command in VLFeat for VLAD coding, and no spatial information is included. The max-pooling operation is performed with LLC and SC, and HC and VLAD use the sum-pooling operation.

Database Setup: Database images are resized to no more than 300×300 in all datasets except for UIUC 8-Sport, since images in this dataset have higher resolutions. We keep the maximum image size of UIUC 8-Sport dataset 400×400.

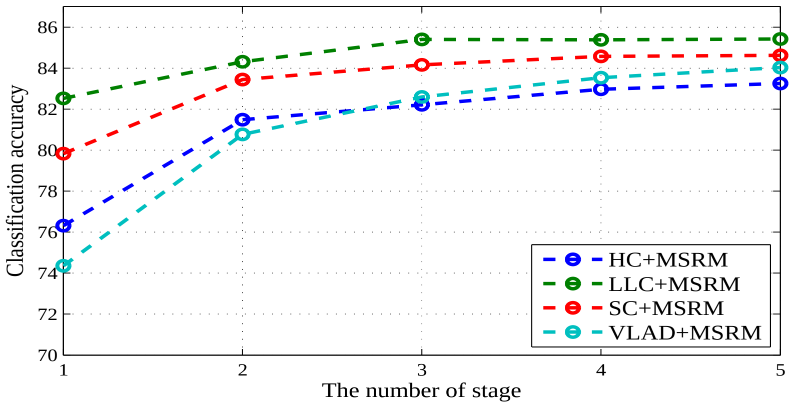


Fig. 3. The experimental results of MSRM with different encoders on Scene-15 dataset. The x-axis denotes the stage number m in MSRM, and the y-axis denotes the classification accuracy.

Table 1. The comparison of classification accuracies on Scene-15 dataset.[1]

Algorithms	Accuracies(%)
Hard-assignment Coding* [9]	78.87 ± 0.52
Locality-constrained Linear Coding* [10]	80.50 ± 0.63
Salient Coding* [16]	82.55 ± 0.41
Locality-Constrained and Spatially Regularized Coding* [13]	82.67 ± 0.57
Localized soft-assignment Coding* [11]	82.70 ± 0.39
Hard-assignment Coding [9]+MSRM	83.25 ± 0.50
Vector of Aggregated Local Descriptors [24] + MSRM	84.03 ± 0.64
Salient Coding [16]+MSRM	84.62 ± 0.61
Locality-constrained Linear Coding [10]+MSRM	**85.42** ± **0.72**

Classifier: Linear SVM, implemented by Liblinear toolbox [36], is used. We set the penalty parameter in SVM to 10.

4.2 Scene-15 Dataset

Scene-15 dataset [9] contains 15 categories and 4485 images, with 200 to 400 images per category. The categories vary from indoor scenes like bedrooms, to outdoor scenes like mountains. Based on the common experimental setting, 100 images per category are taken as training data, and the rest are used for testing.

[1] The tag "$\star$″ in the top right corner of a certain algorithm means that the classification accuracy of this algorithm is not implemented by us, but comes from the corresponding paper, or the survey articles.

Table 2. The comparison of classification accuracies on UIUC 8-sport dataset.

Algorithms	Accuracies(%)
Hard-assignment Coding* [9]	79.98 ± 1.67
Locality-constrained Linear Coding* [10]	81.77 ± 1.51
Localized soft-assignment Coding* [11]	82.29 ± 1.84
Salient Coding* [16]	85.44 ± 1.54
Locality-Constrained and Spatially Regularized Coding* [13]	87.23 ± 1.14
Low Rank Sparse Coding* [12]	88.17 ± 0.85
Hard-assignment Coding [9]+MSRM	84.35 ± 1.16
Locality-constrained Linear Coding [10]+MSRM	88.46 ± 1.13
Salient Coding [16]+MSRM	89.07 ± 1.49
Vector of Aggregated Local Descriptors [24] + MSRM	**89.09 ± 0.96**

Fig. 3 presents the performance of MSRM with different encoders. As we can draw from Fig. 3, MSRM can significantly improve the performance compared with the original state-of-the-art coding algorithms. Generally, the classification accuracy improves as the stage number increases, and it gets saturated when the stage number comes to 5. The classification accuracy is increased by 6.94% for HC [9], 2.90% for LLC [10], 4.79% for SC [16] and 9.67% for VLAD [24] when MSRM sets the stage number to 5. As we can see, our proposed model is especially suitable for HC and VLAD. The reason might be that both HC and VLAD only consider the nearest visual word in the codebook to encode a local feature, and our proposed model can boost their performances largely by making use of the computed residual vector. In comparison, both LLC and SC take into account the contributions from more than one visual words, so the improvements of MSRM with LLC and SC are not as obvious as that with HC and VLAD, but are still convincing. The performance of MSRM is slightly lower than that of IFK [14] reported in [25]. However, the classification accuracy of MSRM with VLAD can be improved to 86.90% if SPM [9] is used, which is comparable to IFK.

We also compare the performance of different coding methods in Table 1. As can be seen, the accuracy of our implementation for HC is much lower than that in [9], MSRM also enhances its discriminative ability, and even surpasses the original LLC and SC. As for SC, the original paper obtains a high performance since multi-scale dense sift and a larger codebook are used. VLAD is rarely used for scene classification, but usually applied to image retrieval. In this paper, we find VLAD also suitable to image classification. The results show that VLAD, integrated into MSRM, can achieve a superior performance to many state-of-the-art coding algorithms.

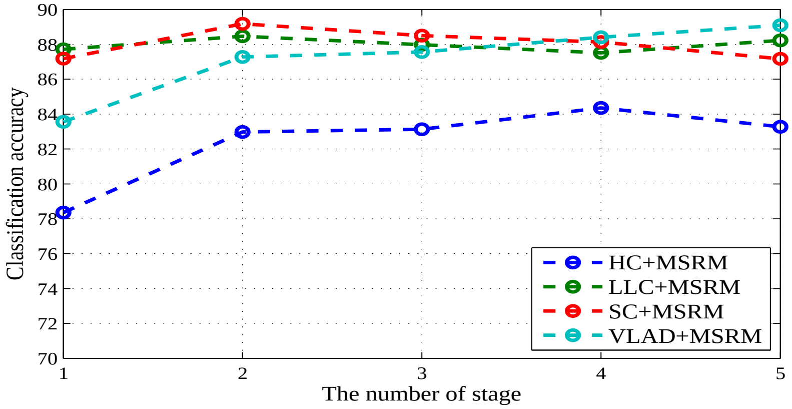


Fig. 4. The experimental results of MSRM with different encoders on UIUC 8-sport dataset. The x-axis denotes the stage number m in MSRM, and the y-axis denotes the classification accuracy.

4.3 UIUC 8-Sport Dataset

UIUC 8-Sport [32] is particularly collected for image-based event classification, and it consists of 1579 images grouped into 8 sport categories: badminton, bocce, croquet, polo, rock climbing, rowing, sailing and snow boarding. According to the standard setup for classification, we use 10 splits of the data, and random select 70 images from each category for training and 60 images for testing. The average accuracy, as well as the standard deviation, is reported.

The classification accuracies of MSRM with different encoders are illustrated in Fig. 4. We can also observe the positive effect on classification results brought by MSRM to various encoders. An exciting accuracy of $\mathbf{89.09 \pm 0.96}$ is achieved by MSRM in conjunction with VLAD when the stage number is set to 5. The baseline of VLAD in UIUC 8-sport dataset is merely 83.56±1.70, and is improved by nearly 6 percents via MSRM.

We compare our proposed MSRM with some related algorithms in Tabel 2. The performance of MSRM is better than many coding methods [11, 10, 16, 13], even outperforms Low Rank Sparse Coding [12], which achieves the state-of-the-art result recently.

4.4 Caltech-101 Dataset

We also evaluate our proposed model for object recognition in Caltech-101 dataset [33]. Caltech-101 dataset consists of 101 object categories including animals, faces, plants *etc.*, with 31 to 800 images per category. Following the standard experimental setting, we use 10 random splits of the data, while taking 30 random images per class for training and the rest for testing. Considering that Caltech-101 dataset is a relatively larger database, we extract dense sift at three

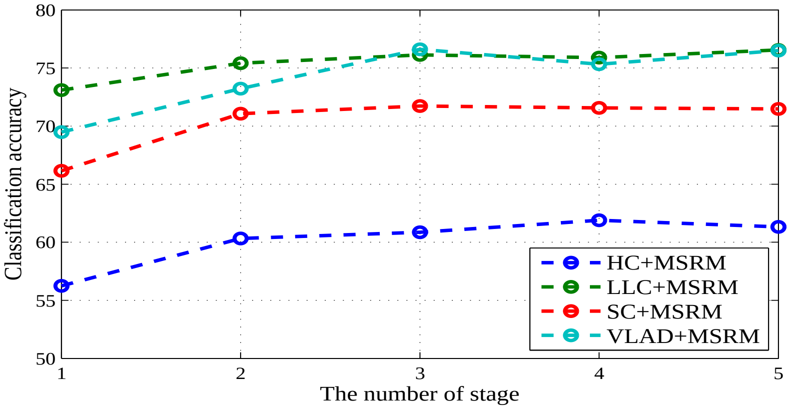


Fig. 5. The experimental results of MSRM with different encoders on Caltech-101 dataset. The x-axis denotes the stage number m in MSRM, and the y-axis denotes the classification accuracy.

scales: 16×16, 24×24, 32×32. Since the spatial layout of object in the image is important in object recognition, we apply SLC [27] to VLAD coding.

In Fig. 5, we plot the performance of MSRM with different encoders under different stage number m. Similar to our experiments in the previous dataset, M-SRM improves our selected encoders significantly, $i.e.$HC [9] by 5.07%, LLC [10] by 3.47%, SC [16] by 5.33%, VLAD [24] by 7.02%. The performance of HC is much different from the results reported in [25], due to the usage of Hellinger kernel. We conduct the experiment of HC to make clear the effect of Hellinger kernel, and find that Hellinger kernel is extremely useful to HC in this dataset. A significant improvement of 13.25% is observed.

We also list some excellent results reported by other algorithms in Tabel 3. Our proposed model also obtains competitive performance compared with many state-of-the-art algorithms.

4.5 Discussion

Information Loss: Average Quantization error is an important index in evaluating a feature coding algorithm, which is defined as

$$AQE = \frac{1}{N}\sum_{i=1}^{N} \|x_i - cw_i\|^2 \tag{3}$$

Large quantization error results in much information loss in the coding procedure, which impairs the classification accuracy heavily. The quantization error is brought in when we represent a local descriptor x by a visual word $q(x) \in \mathcal{C}$. such a behaviour is simple, but also somehow harmful due to the existence of the

Table 3. The comparison of classification accuracies on Caltech-101 dataset.

Algorithms	Accuracies(%)
Hard-assignment Coding* [9]	69.43 ± 0.52
Salient Coding* [16]	69.55 ± 0.83
Locality-constrained Linear Coding* [10]	71.67 ± 0.86
Localized soft-assignment Coding* [11]	72.58 ± 1.08
Locality-Constrained and Spatially Regularized Coding* [13]	73.23 ± 0.81
Low Rank Sparse Coding* [12]	75.02 ± 0.74
Hard-assignment Coding [9]+MSRM	61.88 ± 1.15
Salient Coding [16]+MSRM	71.73 ± 1.47
Locality-constrained Linear Coding [10]+MSRM	76.56 ± 0.90
Vector of Aggregated Local Descriptors [24] + MSRM	**76.59 ± 0.51**

Table 4. The Average Quantization Error in different datasets.

	Scene-15 dataset	UIUC 8-sport dataset	Caltech-101 dataset
HC [9]	2.16	3.02	3.02
HC+MSRM	1.01	1.43	1.73

residual vector $r(x) = x - q(x)$ between x and $q(x)$. Some previous methods [11, 10, 13] use more visual words to represent x to alleviate the problem.

In MSRM (we take Two Stage Residual Model with HC as example), the local descriptor x is represented by a tuple $(q_1(x), q_2(x - q_1(x)))$, where $q_i(x)$ is the nearest visual word of x in $\mathcal{C}^i$. The encoder applied to the tuple $(q_1(x), q_2(x - q_1(x)))$ generates the code for x.

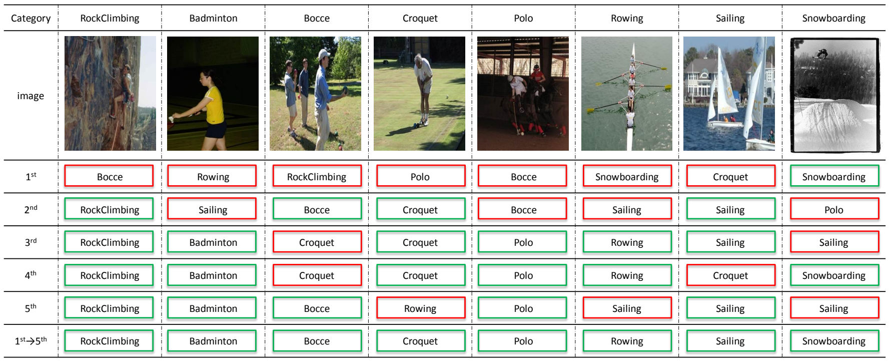
We use Hard-assignment Coding (HC) [9] to show the way that MSRM reduces the quantization error. The AQE of HC is

$$AQE_{HC} = \frac{1}{N}\sum_{i=1}^{N} \|x_i - q_1(x_i)\|^2 \tag{4}$$

while the AQE of MSRM with HC is

$$AQE_{MSRM} = \frac{1}{N}\sum_{i=1}^{N} \|x_i - q_1(x_i) - q_2(x_i - q_1(x_i))\|^2 \tag{5}$$

It is straightforward that the energy of the residual vector $x - q(x)$ is relatively smaller than x itself. We compute the AQE of MSRM with HC in Scene-15 dataset, UIUC 8-sport dataset [32] and Caltech-101 dataset [33]. The result is presented in Table 4, and find that MSRM can significantly reduce the quantization error, which strongly explains why MSRM with HC as the encoder can improve the baseline.



Category	RockClimbing	Badminton	Bocce	Croquet	Polo	Rowing	Sailing	Snowboarding
image								
1st	Bocce	Rowing	RockClimbing	Polo	Bocce	Snowboarding	Croquet	Snowboarding
2nd	RockClimbing	Sailing	Bocce	Croquet	Bocce	Sailing	Sailing	Polo
3rd	RockClimbing	Badminton	Croquet	Croquet	Polo	Rowing	Sailing	Sailing
4th	RockClimbing	Badminton	Croquet	Croquet	Polo	Rowing	Croquet	Snowboarding
5th	RockClimbing	Badminton	Bocce	Rowing	Polo	Sailing	Sailing	Sailing
1st→5th	RockClimbing	Badminton	Bocce	Croquet	Polo	Rowing	Sailing	Snowboarding

Fig. 6. Some misclassified images in UIUC 8-sport dataset. The "Category" in the first row indicates the ground truth of the image, and The third to the seventh row present the predicted label that SVM outputs based on the features of each stage in MSRM. The last row shows the result when we concatenate the features from all stages in MSRM. The red box means a false prediction, and the green box means a correct prediction.

Complementarity: We also observe that the output of each stage in MSRM is complementary to each other, and the powerful classifier SVM is able to select several distinctive stages to distinguish images from different categories.

In order to prove our conjecture, we conduct an interesting experiment shown in Fig. 6, that the features from each stage in MSRM with VLAD as the encoder form the input training set and testing set for SVM. We select one misclassified image per category, and find that although the classifier cannot give a correct judgement for the label of the image according to the features from $1st$ stage in MSRM, *i.e.* the original coding strategy, it can revise its prediction result with more complementary information from the latter stages in MSRM. For example, the image from Rowing category in the seventh column of Fig. 6, is misclassified as snowboarding in the first stage and sailing in the second stage, however, it obtains a correct labeling in the third stage and the fourth stage. When combining the features from all stages, the false prediction is corrected as shown in the last row. The last column presents an image from snowboarding is misclassified in the second, third and fifth stage, but is assigned a right label if the features from all the stages are combined together. This phenomena reveals the robustness of our proposed model.

To further confirm the complementarity between the features from each stage of MSRM, we compare our proposed MSRM with a single stage coding under the same feature dimension. Specifically, Scene-15 dataset is used, and VLAD is selected as the encoder. For MSRM, the codebook size in each layer of MSRM is 64, and the stage number m ranges from 1 to 5. Hence the feature dimension of MSRM for an image is 128*64*m. For the single stage coding, the codebook size is set to $64 * m$ $(1 \leq \; m \leq 5)$. Hence the final feature dimension for an image

Table 5. The comparison of classification accuracy between MSRM and the single stage coding under the same feature dimension in Scene-15 dataset.

m	1	2	3	4	5
VLAD	74.36%	78.15%	79.35%	80.14%	80.18%
VLAD + MSRM	74.36%	80.77%	82.59%	83.53%	84.03%

is also $128 * 64 * m$. The results are presented in Table 5, which suggest that MSRM works significantly better than only single stage coding.

5 Conclusion

In this paper, we propose a generic model called Multiple Stage Residual Model (MSRM) to make full use of the residual vector, while many coding algorithms focus on reducing it. MSRM has been proved to be effective to improve the performance of many state-of-the-art coding algorithms further.

In the future, we will study how to introduce the spatial consistency to M-SRM, and introduce more coding methods to MSRM in a proper way.

References

1. Jégou, H., Zisserman, A., et al.: Triangulation embedding and democratic aggregation for image search. In: CVPR. (2014)
2. Zheng, L., Wang, S., Liu, Z., Tian, Q.: Packing and padding: Coupled multi-index for accurate image retrieval. In: CVPR. (2014)
3. Kosala, R., Blockeel, H.: Web mining research: A survey. ACM Sigkdd Explorations Newsletter (2000)
4. Csurka, G., Dance, C., Fan, L., Willamowski, J., Bray, C.: Visual categorization with bags of keypoints. In: ECCV. (2004)
5. Fei-Fei, L., Perona, P.: A bayesian hierarchical model for learning natural scene categories. In: CVPR. (2005)
6. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. IJCV (2004)
7. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: CVPR. (2005)
8. Lloyd, S.: Least squares quantization in pcm. IEEE Trans. on Information Theory (1982)
9. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In: CVPR. (2006)
10. Wang, J., Yang, J., Yu, K., Lv, F., Huang, T., Gong, Y.: Locality-constrained linear coding for image classification. In: CVPR. (2010)
11. Liu, L., Wang, L., Liu, X.: In defense of soft-assignment coding. In: ICCV. (2011)
12. Zhang, T., Ghanem, B., Liu, S., Xu, C., Ahuja, N.: Low-rank sparse coding for image classification. In: ICCV. (2013)
13. Shabou, A., LeBorgne, H.: Locality-constrained and spatially regularized coding for scene categorization. In: CVPR. (2012)

14. Perronnin, F., Sánchez, J., Mensink, T.: Improving the fisher kernel for large-scale image classification. In: ECCV. (2010)
15. Zhou, X., Yu, K., Zhang, T., Huang, T.S.: Image classification using super-vector coding of local image descriptors. In: ECCV. (2010)
16. Huang, Y., Huang, K., Yu, Y., Tan, T.: Salient coding for image classification. In: CVPR. (2011)
17. van Gemert, J.C., Geusebroek, J.M., Veenman, C.J., Smeulders, A.W.: Kernel codebooks for scene categorization. In: ECCV. (2008)
18. Yang, J., Yu, K., Gong, Y., Huang, T.: Linear spatial pyramid matching using sparse coding for image classification. In: CVPR. (2009)
19. Shaban, A., Rabiee, H.R., Farajtabar, M., Ghazvininejad, M.: From local similarity to global coding: An application to image classification. In: CVPR. (2013)
20. Yu, K., Zhang, T., Gong, Y.: Nonlinear learning using local coordinate coding. In: Advances in neural information processing systems. (2009)
21. Boureau, Y.L., Ponce, J., LeCun, Y.: A theoretical analysis of feature pooling in visual recognition. In: ICML. (2010)
22. Boureau, Y.L., Bach, F., LeCun, Y., Ponce, J.: Learning mid-level features for recognition. In: CVPR. (2010)
23. Koniusz, P., Yan, F., Mikolajczyk, K.: Comparison of mid-level feature coding approaches and pooling strategies in visual concept detection. CVIU (2013)
24. Jégou, H., Douze, M., Schmid, C., Pérez, P.: Aggregating local descriptors into a compact image representation. In: CVPR. (2010)
25. Huang, Y., Wu, Z., Wang, L., Tan, T.: Feature coding in image classification: A comprehensive study. PAMI (2013)
26. Arandjelovic, R., Zisserman, A.: All about vlad. In: CVPR. (2013)
27. McCann, S., Lowe, D.G.: Spatially local coding for object recognition. In: ACCV. (2012)
28. Wang, X., Bai, X., Liu, W., Latecki, L.J.: Feature context for image classification and object detection. In: CVPR, IEEE (2011) 961–968
29. Wang, X., Wang, B., Bai, X., Liu, W., Tu, Z.: Max-margin multiple-instance dictionary learning. In: ICML. (2013)
30. Mairal, J., Bach, F., Ponce, J., Sapiro, G., Zisserman, A., et al.: Supervised dictionary learning. In: NIPS. (2008)
31. Yang, J., Yu, K., Huang, T.: Supervised translation-invariant sparse coding. In: CVPR. (2010)
32. Li, L.J., Fei-Fei, L.: What, where and who? classifying events by scene and object recognition. In: ICCV. (2007)
33. Fei-Fei, L., Fergus, R., Perona, P.: Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. Computer Vision and Image Understanding (2007)
34. Chatfield, K., Lempitsky, V., Vedaldi, A., Zisserman, A.: The devil is in the details: an evaluation of recent feature encoding methods. In: BMVC. (2011)
35. Vedaldi, A., Fulkerson, B.: VLFeat: An open and portable library of computer vision algorithms. http://www.vlfeat.org/ (2008)
36. Chang, C.C., Lin, C.J.: LIBSVM: A library for support vector machines. ACM Trans. on Intelligent Systems and Technology (2011) Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm.