PROCEEDINGS OF SCIENCE

Static Image Classification based on ScSPM and LBP histogram Fourier (LBP-HF) Features

Xiao Li[1], **Zhanbo Li**[2]

Information Engineering School of Zhengzhou University, Zhengzhou, Henan, 450000, China

E-mail: iexli@gs.zzu.cn, iezbli@zzu.edu.cn

Haibo Pang[3]

School of Software Technology, Zhengzhou University, Zhengzhou, Henan, 450002, China

E-mail: phb@zzu.edu.cn

In the recent digital age, support vector machines (SVMs) that use a spatial pyramid matching (SPM) kernel have been around the globe for image classification. Although this is popular, there exists many problems in its use; for examples, nonlinear SVMs have a high complexity in training and testing. Applying the algorithms to big datasets, which holds many images, greater than a thousand is a detailed and complicated task. This paper develops an extension of the linear SPM that uses linear kernel on spatial-pyramid pooling of SIFT sparse codes (ScSPM), by generalizing vector quantization to sparse coding followed by multi-scale spatial max pooling, we propose a linear SPM kernel based on LBP-HF sparse codes. This highly-innovative approach remarkably reduces complexity of the training and testing. Our image categorization experiment results regarding classification accuracy, suggests that a linear SPM, based on a sparse coding of the LBP-HF descriptors, significantly outperforms the linear SPM kernel on histograms and even better than the nonlinear SPM kernels.

ISCC 2015
18-19, December 2015
Guangzhou, China

[1] Xiao Li

PoS(ISCC2015)038

1.Introduction

The Image classification system is a significant technique that is important in digital processing and computer vision. In order to address the issue of classification, different approaches have been discussed in literature. In recent times, SVMs using SPM kernel have been widely used and achieved great success.

The SPM is an extension of the bag-of-features (BoF) model. However, both BoF and SPM need to work together with a particular type of nonlinear Mercer kernels which requires high computational and memory complexities. So it doesn't really apply to standard real-world applications. With this in mind, J. C. Yang et al. developed the ScSPM approach, and relates perfect to image classification [1].

In spite of its popularity, the ScSPM still remains an open problem. The ScSPM uses 128-dimensional normalization of an SIFT descriptor that leads to high complex computations and huge memory costs. Therefore we developed a new modified ScSPM that instead of SIFT features, we conceived an idea of using a linear SPM kernel based on LBP-HF sparse codes [2]. The flow chart of our modified ScSPM with LBP-HF descriptor is shown in Figure 1. We performed our experiments on two different datasets, namely, Caltech-101 and Caltech-256. As the results shown, our modified approach is able to obtain a recognition rate of greater than 85% which outperforms published results previously reported.

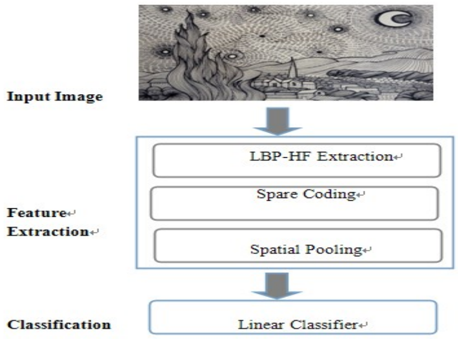

Figure 1 : The flow chart of our proposed approach.

The rest of our paper is organized as follows: Sec. 2 Related Works; Sec. 3 Proposed Approach; Sec. 4 Efficient Implementations; Sec. 5 Experiment Results; Sec. 6 Conclusion and Future Work.

2.Related work

Image feature extractions have been widely applied to image classification. The classic methods are Color Histogram, Gray Histogram, LBP Histogram, Scale-Invariant Feature Transform (SIFT) and Histogram of Gradients (HOG).

As a non-parametric operator, local binary pattern (LBP) is widely used till now. Many works have been done before to improve the original LBP model. Heikkilä et al. invented the center-symmetric LBP (CS-LBP) operator to use the LBP for region description [3]. J. D. Sun et al. introduced an improved direction LBP (ID-LBP) [4]. Recently, in the field of texture-based object classification, a rotation invariant texture descriptor played a pivotal role but reduced the classification accuracy. So Saipullah et al. proposed a completely new texture descriptor which is named sorted neighborhood differences (SND) [5]. G. Y. Zhao et al. also developed a different approach to compute "rotation-invariant features from histograms of local non-invariant patterns" and applied it to a static LBP descriptor, namely, the LBP-HF [2].

Addressing the issue of classification, different approaches have been utilized, each based on one of the following: (a) the exploitation of the classified samples; (b) pre-processing features selection and or extraction techniques aiming at transforming the original feature into another space of lower dimensionality; (c) techniques of shape description for modeling the class spectral signatures; and (d) the popular SVM classifiers.

3.The Framework of Our Proposed Approach

We should briefly review LBP methodology, because a uniform LBP pattern is base on it. And will focus later on the rotation-invariant image features for the static-texture description.

3.1LBP and The Extensions

3.1.1The Former LBP Operators

LBP is widely used for classification in computer vision. It forms labels for the image pixels by threshold of the 3×3 neighborhood of each pixel with the center values. Further extensions are considered uniform patterns. It contains a maximum of two bitwise transitions from 0 to 1 or vice versa when the bit pattern is considered circular. For instance, when using $(8,R)$ neighborhood, the 58 possible uniform patterns are shown in Figure 2. Another extension is the rotation-invariant LBP operator which is achieved by circularly rotating each bit pattern to the the minimum value.

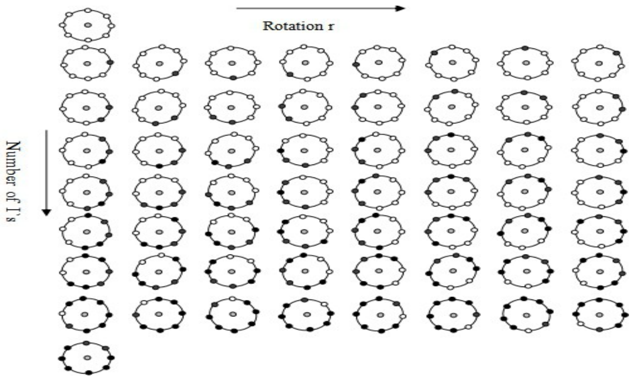

Figure 2: In $(8\ ,R)$ neighborhood, the 58 possible uniform patterns

3.1.2LBP-HF

Images may change angles for different viewing (see Figure 3). The uniform patterns correspond to the row number while the rotation of the pattern corresponds to the column. These rotation-invariant descriptors are used for static texture analysis to give understanding of what objects look like in a different view using multidimensional image resolution.

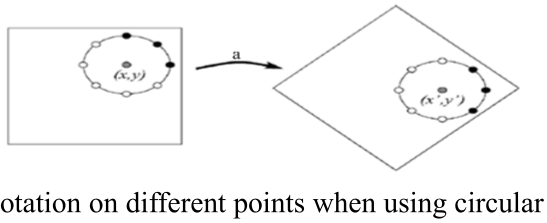

Figure 3: The effect of image rotation on different points when using circular neighborhoods.

The rotation of the pattern is unique in that the location of (x, y) coordinates are remaining the same while the object is rotated in up to 360 degrees. As the rotation occurs, there is a uniform set of data so an image may turns along during circle movement process. This will cause a cyclic shift in histograms data alone each of the rows. Histogram binary values will change based on the degree of movement. Different features can be distinguished by the input image and classified in the input histogram rows that are invariant to cyclic shifts.

$$h_{I^{a^\circ}}\left(U_P\left(n, r+A\,mod\,P\right)\right)=h_I\left(U_P\left(n, r\right)\right) \tag{3.1}$$

where $U_P(n, r)$ represents a specific uniform LBP pattern with P neighboring sampling points, n denotes the number of 1 bit (the row number in Figure 2) and r defined the rotation of the pattern (the column number in Figure 2). $h_I\left(U_P(n, r)\right)$ is the number of occurrences of uniform pattern $U_P(n, r)$ in image I , after the rotations (we limit the rotations to integer multiples of the angle between two sampling points) by a degree (see Figure 3), where $a=A(360/P), A=0,1\ldots P-1$) , it turns to be $h_{I^{a^\circ}}\left(U_P\left(n, r+A\,mod\,P\right)\right)$.

Based on the property, in 2012, G. Y. Zhao et al. proposed a feature which can be constructed from discrete Fourier transform (DFT) of the histogram row data [2]. We set $H(n, .)$ to be the DFT of the $n-th$ row of histogram $h_I\left(U_P(n, r)\right)$ (see Eq. (3.2)). An input vector can cause changes in the DFT coefficients if certain formulated data applies.

$$H(n, u)=\sum_{r=0}^{P-1} h_I\left(U_P(n, r)\right) e^{-i2\Pi ur/P} \tag{3.2}$$

While the data cyclic shifts of the rows, there is a corresponding value which is a constant to the rotations of the input image. The LBP-HF (see Eq. (3.3)) which is called Fourier magnitude spectrum is a special circumstance for these features.

$$|H(n, u)|=\sqrt{H(n, u)\overline{H(n, u)}} \tag{3.3}$$

where, we use $\overline{H(n, u)}$ to represents the complex conjugate of $H(n, u)$.

The reason we choose LBP-HF operator here is because of its detail and sensitivity to gray-scale changes, and its computational simplicity. Based on these characteristics, the LBP-HF descriptor is able to meet challenging real-time requirements.

3.2Encoding LBP-HF

In this paper, we use Sparse Coding to derive image representations for the following reasons: SC allows significantly reduce reconstruction errors; Secondly, sparsity allows the representation to be specialize which makes it possible to capture noticeable features of images. Third, related research in image statistics evidently reveals that image patches are usually sparse signals. To be precise, Sparse Coding is defined as:

$$\min_{U,V} \sum_{m=1}^{M} ||x_m, u_m V||^2+\lambda|u_m|$$

$$\text{subject to } ||V_k|| \le 1, \forall k=1, \ldots, K \tag{3..4}$$

where $X=[x_{1,} x_2, \ldots x_M]^T \in R^{M\times D}$ is the set of LBP-HF "appearance descriptors defined in D-dimensional feature space", $U=[u_{1,} u_2, \ldots u_M]^T$ is the cluster membership indicators. $V=[v_{1,}v_2,\dots v_K]^T$ are the K cluster centers, namely, codebook. $|.|$ represents the L1-norm of vectors, $||.||$ is the L2-norm of vectors and λ is a free parameter (see Sec. 5.3.2).

Spare Coding has training and coding phases. First, with respect to cluster membership indicators U and the codebook V, we can use a descriptor set X to solve Eq. (3.4). In the coding phase, as long as an image can be represented as a descriptor set X, then we can obtained the Sparse Coding codes by optimizing Eq. (3.4) with respect to U only.

3.3Linear SPM

Any image represented by a set of descriptors can compute a single featured vector based on data from statistics in the descriptors' codes. In the improvements of SPM, the image's SPM representation z represents a string of local histograms in various partitions under different scales. Through normalization it can be viewed as a histogram again. For the problem of binary image classification, the purpose of an SVM model is to study the decision function f. For any image represented by z, if $f(z)>0$ then it illustrates the image is considered positive, or else it is classified as negative.

We decide upon an approach by using linear SVMs based Sparse Coding of LBP-HF. Parameter U represents the result of applying the Sparse Coding Eq. (3.4) to a descriptor set X, supposing V to be pre-learned as well as fixed, and then use a pre-chosen pooling function $z=F(U)$ to compute the image feature. The pooling function F works on every column of U, recall that each column of U corresponds to the responses of all the local descriptors to one specific item in the codebook V. We set F to be a Max pooling function on the absolute sparse codes in our work

$$z_j=max\{|u_{1j}|,|u_{2j}|,\dots,|u_{Mj}|\} \qquad （3.5）$$

where z_j is the $j-th$ element of z, the matrix element u_{ij} locates at $i-th$ row and $j-th$ column of the cluster membership indicators U, in this region, M represents the local descriptors' number. Besides the model of SPM, we can get a similar construction by using max-pooling(see Eq. (3.5)) on spatial pyramid constructed images.

Figure 4 illustrates the whole structure of our proposed approach. We can obtain the spatial pyramid representation by string the pooled features of different regions and scales. We set up a representative z_i for image I_i. In our work, a simple linear SPM kernel has been used

$$k(z_i,z_j)=z_i^T z_j=\sum_{l=0}^{2}\sum_{s=1}^{2^l}\sum_{t=1}^{2^l}\langle z_i^l(s,t),z_j^l(s,t)\rangle \qquad (3.6)$$

where $\langle z_i,z_j\rangle=z_i^T z_j$. Under the scale of l, for the $(s,t)-th$ segments of image I_i, we can use $z_i^l(s,t)$ to represent the max pooling statistics of the descriptors' sparse codes. The decision function of the binary SVM is

$$f(z)=\left(\sum_{i=1}^{n}\alpha_i z_i\right)z+b=w^T z+b \qquad (3.7)$$

Eq. (3.7) is viewed as the primal formulation of SVMs in predecessors' researches. We found that the linear SPM kernel based on SC statistics always achieves excellent classification accuracy.

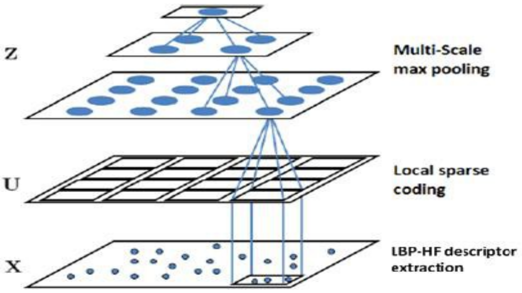

Figure 4: The illustration architecture of our proposed approach.

4.Implements

4.1spare Coding

Generally, we can deal with the problem of optimization by alternating optimizing over the parameter V or U while fixing the other one. For instance, if we fixed the parameter V , in order to solve the optimization problem, each coefficient u_m needs to be individually optimized:

$$\min_{u_m} \|x_m - u_m V\|_2^2 + \lambda |u_m| \tag{4.1}$$

Else if we fixed the parameter U , then the problem turns into a least squares problem with quadratic constraints:

$$\min V \|X - UV\|_F^2$$
$$\text{subject to } \|v_k\| \leq 1, \forall k = 1, 2, \ldots, K \tag{4.2}$$

The optimization problem can be effectively accomplished by using the Lagrange dual (as used in Eq. (4.4)).

To train the codebook V , we extract fifty thousand LBP-HF descriptors from random patches, and then through the way of iterative to carry out the steps of Eq. (4.1) and Eq. (4.2). Once we get the codebook V in the off-line training exploration, of each descriptors of an image, we can do a fast on-line sparse coding (as in Eq. (4.1)).

4.2Multi-class Linear SVM

In this part, a simple implementation of linear SVMs we used in our experiments was introduced. Let $\{(z_i, y_i)\}_{i=1}^n, y_i \in Y = \{1, \ldots, L\}$ denotes the training data, a linear SVM is designed to learn L linear functions $\{w_c^T | c \in Y\}$, such that, we can predict class label of the test datum z by using Eq. (4.3). A one-against-all strategy was used to train the L binary linear support vector machines, each of which can be used against unconstrained convex optimization:

$$\min_{w_c} \{ J(w_c) = \|w_c\|^2 + c \sum_{i=1}^{n} l(w_c; y_i^c, z_i) \} \tag{4.3}$$

where, only when we set y_i as c , then $y_i^c = 1$, otherwise $y_i^c = -1$, ; We use $l(w_c; y_i^c, z_i)$ to denote a hinge loss function. Since the standard hinge loss function is not differentiable everywhere, we adopt a differentiable quadratic hinge loss

$$l\left(w_c\,;\,y_i^c\,,\,z_i\right)=\left[max\left(0,w_c^T\,z\cdot y_i^c-1\right)\right]^2 \tag{4.4}$$

so that with the use of gradient-based optimization techniques our training can be done easily. Here we used LBFGS in our training. We trained 200,000 examples with LBP-HF features in our experiments (in Sec. 5.1 and 5.2). They were always done in three minutes.

5.Experiments and Results

After using our implementation and evaluation methods of different classes in SPM approaches on two diverse datasets: results of our experiment findings are shown below in Table 1 and Table 2 from Caltech-101 Dataset as well as Caltech-256 Dataset. The four methods we used are: KSPM, LSPM, original ScSPM and our Modified ScSPM. Besides our own implementations, we also quoted some results directly from published in a research [6].

In our implementations, we used a single descriptor type named LBP-HF descriptor, which extracted from 4×4 pixel patches which were densely sampled from each image on a grid with step-size 2 pixels. All the images need to be pre-processed into gray scale. In order for us to train the codebooks, we configured standard K-means clustering for LSPM (the nonlinear kernel SPM that uses spatial-pyramid histograms and Chi-square kernels) and KSPM (the simple linear SPM that uses linear kernel on spatial-pyramid histograms). We also configured the sparse coding scheme into the approach we developed too. In all of our testing: we fixed the codebook size as 512 for LSPM, 1024 for ScSPM and then our modified approach. We used the SVM which was introduced in Sec. 4.2 of our paper to train the linear classifiers. At that point in time, we then applied LIBSVM package to train KSPM [6].

Through the use of different random selections of images, we repeat the experiment ten times to make sure we found reliable results. We then recorded the average recognition rates of per-class for each run. Lastly, we applied the standard and means deviation of the recognition rates in order to report the final results obtained.

5.1Caltech-101 Dataset

It contains 101 categories with high shape variability in the Caltech-101 Dataset. We followed the common experiment setup for this Dataset. For every class, we selected 15 or 30 images to do the training and then testing the rest. As shown in Table 1, we found that the scheme we proposed would be superior to linear SPM, and also outperforms the nonlinear SPM.

Algorithms	15 Training	30 Training
KSPM	55.78	64.38
LSPM	52.35	57.83
ScSPM	66.04	72.97
Modifiec ScSPM	70.12	75.63

Table 1: The comparison of classification rate (%) on Caltech-101.

5.2.Caltech-256 Dataset

Compared to the Caltech-101 Dataset, the Caltech-256 Dataset has roughly 256 different types of images that are a higher intra-class variability and location variability. As we tried our method on 15 or 30 training images per class respectively; We logged the results listed in Table 2. We learned from our results that the approach we developed is superior to other methods.

Algorithms	15 Training	30 Training
KSPM	23.64	29.01
LSPM	12.97	15.76
ScSPM	28.97	33.73
Modifiec ScSPM	41.03	43.18

Table 2: The comparison of classification rate (%) on Caltech-256.

5.3.The Revisit of Our Experiments

5.3.1Patch Size and Codebook Size

Throughout the whole experimental processing, we set a single the patch size of 4×4 pixels to extract LBP-HF descriptors. As known to all, if we set the codebook size to a very small setting, then the histogram feature will probably loose the power of discriminant. Likewise if we set the codebook size setting to a very large size, then the histograms displayed image will never match. During the experimental processing of LSPM, ScSPM, and our approach, we tried three common sizes: 256, 512 and 1024. The comparison shows in Table 3.

	Codebook Size	256	512	1024
15 Traing	LSPM	51.74	53.02	51.35
	ScSPM	62.39	64.12	68.33
	Modified ScSPM	67.07	69.23	72.21
30 Traing	LSPM	57.24	58.27	58.64
	ScSPM	68.63	71.93	74.13
	Modified ScSPM	73.26	76.42	79.17

Table 3: The influence of codebook size on different approaches on Caltech-101 Dataset.

5.3.2 Sparse Coding Parameter

In Eq. (4, 8), we defined a free parameter λ that we need to specify when doing sparse coding on the feature vector. During our work, we found that keeping the sparsity around 10% is able to produce good results. In our scientific experiments, we set the value of λ from 0.3 to 0.4 and the mean number of supports to be the number of around 10.

5.3.3 Comparisons

In our comparison work, we compared three distinct pooling functions: Square root, Absolute Value and Max. Figure 5 lists the experiments on Caltech-101 of 30 training of each categories using different pooling methods. We also tried intersection kernel, Chi-square kernel and the linear classifiers in our scientific approach. In our workings, we conducted experiments on the Dataset of Caltech-101 with 15 training images per class, as shown in Figure 6.

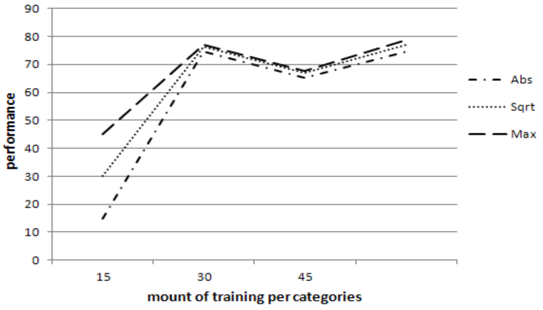

Figure 5: The comparison using different pooling

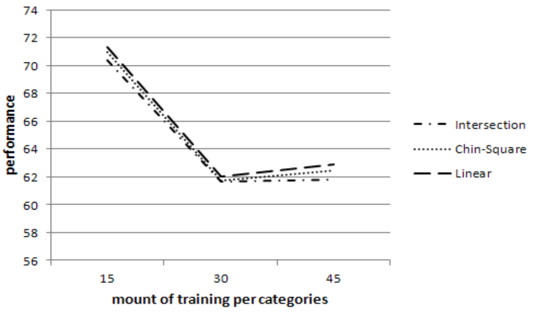

Figure 6: The comparison between different

methods: Sqrt (square root of mean squared statistics), Abs (mean of absolute values), and Max pooling.	kernels.

The comparison above strongly explains why we select Max pooling and linear classifiers in our scientific approach.

6.Conclusion and Future Work

We proposed a modified SPM approach based on LBP-HF sparse codes to deal with the issue of image classification. The image representation we obtained does indeed work extremely well together using simple linear SVMs which greatly improved the training scalability along with the testing speed and classification accuracy.

In future experiments, we can generalize the rotation-invariant image descriptor LBP-HF to embed any uniform features into the framework. Therefore this idea combines with the needs of supplementary information to obtain better description ability. This direction of research can be rewarding. A future experimenter could try the idea of a feed-foward network to improve the efficiency of encoding. Moreover, we can also increase the description ability by studying the codebook in a supervised method, as suggested by other working professionals.

References

[1] J. C. Yang, K. Yu, Y. H. Gong, T. Huang. *Linear Spatial Pyramid Matching Using Sparse Coding for Image Classification* [C]. Proceedings of the 2009 IEEE Computer Society. Conference on Computer Vision and Pattern Recognition，Miami, USA. pp, 1794-1801 (2009)

[2] G. Y. Zhao, T. Ahonen, J. Matas, M. Pietikainen. *Rotation-Invariant Image and Video Description with Local Binary Pattern Features* [J]. Image Processing. 21(4), 1465-1477 (2012)

[3] M. Heikkilä, M. Pietikäinen, C. Schmid. *Description of interest regions with local binary patterns* [J]. Pattern Recognition. 42(3), 425-436 (2009)

[4] J. D. Sun, S. S. Zhu, X. S. Wu. *Image Retrieval Based on an Improved CS-LBP Descriptor* [C]. Information Management and Engineering(ICIME). 2010 The 2nd IEEE International conference. Chengdu, CHINA. PP, 115-117 (2010).

[5] K. M. Saipullah, D.-H. Kim, S.-L. Lee. *Rotation invariant texture feature extraction based on Sorted Neighborhood* Differences [C]. Multimedia and Expo (ICME). 2011 IEEE International Conference, Barcelona, Spain. PP, 1-6 (2011).

[6] G. Griffin, P. Holub, A. D. Perona. *Caltech-256 object category dataset.* Technical Report 7694, California Institute of Technology. CaltechAUTHORS: School of Electronics and Computer Science at the University of Southampton (2007).