Sorted Random Projections for Robust Texture Classification

Li Liu* Paul Fieguth‡ Gangyao Kuang* Hongbin Zha§
*National University of Defense Technology, Changsha, Hunan, China 410043
‡System Design Engineering Dept., University of Waterloo, Waterloo, Ontario, Canada N2L 3G1
§Machine Intelligence Dept., Peking University, Beijing, China 100871
{dreamliu2010@gmail.com;pfieguth@uwaterloo.ca;kuangyeats@vip.sina.com;zha@cis.pku.edu.cn}

Abstract

This paper presents a simple and highly effective system for robust texture classification, based on (1) random local features, (2) a simple global Bag-of-Words (BoW) representation, and (3) Support Vector Machines (SVMs) based classification. The key contribution in this work is to apply a sorting strategy to a universal yet information-preserving random projection (RP) technique, then comparing two different texture image representations (histograms and signatures) with various kernels in the SVMs.

*We have tested our texture classification system on six popular and challenging texture databases for exemplar based texture classification, comparing with **12** recent state-of-the-art methods. Experimental results show that our texture classification system yields the best classification rates of which we are aware of 99.37% for CUReT, 97.16% for Brodatz, 99.30% for UMD and 99.29% for KTH-TIPS. Moreover, combining random features significantly outperforms the state-of-the-art descriptors in material categorization.*

1. Introduction

Texture classification is a branch of computer vision and pattern recognition which has received considerable attention. Yet despite almost 50 years of research and development, designing a high-accuracy and robust texture classification system for real-world applications remains a challenge for at least three reasons: the wide range of various natural texture types; the presence of large intra-class variations in texture images, such as brightness, contrast, rotation, skew, occlusion, scale and even non-rigid surface deformation, caused by arbitrary viewing and illumination conditions; and the demands of low computational complexity and a desire to limit algorithm tuning.

The basic building elements that constitute a reliable texture classification system are (i) local texture descriptors, (ii) non-local statistical descriptors, (iii) the design of a distance/similarity measure, and (iv) the choice of classifier.

Recently, there has been significant interest in using a Bag-of-Words (BoW) model for texture classification, to represent a texture non-locally by the distribution of local textons [1, 2, 6, 7, 8, 11]. Given a BoW framework, the key remaining challenge, and the focus of our present work, is on points (i) and (iv), the development of effective local texture descriptors and a corresponding classifier.

It is well known that the true, inherent dimensionality of a textured image is far less than the dimensionality of the neighborhood over which the texture is defined. In order to reduce the number of features, trimming the number of required dimensions for texture representation, filter bank-based techniques are conventionally used. The motivating principal underlying filter bank methods is that the local appearance of a texture can be summarized by the distribution of the responses of a family of filters. However, the design of an optimal filter bank is nontrivial and normally application dependent.

Notably, the supremacy and dominant role of filter bank-based methods for texture analysis have been recently challenged: Varma and Zisserman [6] argue that raw image patch descriptors achieve better performance than the popular filter banks they compared against. Clearly the size of the patch must be large enough to encompass the dominant texture variations, however the increase in the dimension of the feature space with the size of the neighborhood and the sensitivity to image rotation limits the applicability of the patch descriptor. Similarly, in [7] Ojala *et al*. advocated the use of the Local Binary Pattern (LBP) textons. However, the size of the LBP textons increases drastically with an increase in sampling radius and number of sampling points, leading to only uniform textons being kept which are insufficiently descriptive.

In contrast, in [3, 4] Liu *et al*. introduced an important innovation by using Random Projections (RP), a universal, information-preserving, dimensionality-reduction technique to project from the patch vector space to a compressed

2011 IEEE International Conference on Computer Vision
978-1-4577-1102-2/11/$26.00

patch space without loss of salient information, and claim that the performance achieved by random features can outperform patch features, LBP and various filter bank-based methods.

However the results in [3, 4] were obtained with a BoW approach using a nearest-neighbor classifier, whereas SVMs classifiers have shown substantial promise for texture classification tasks, and the development of kernels suitable for use with local features has emerged as a hot research topic [2, 12, 22, 23].

The above two paragraphs serve as the motivation for the research of this paper: we seek to build an effective and robust texture classification system, taking advantage of the universality of random features, but further generalizing them for robustness to rotation, together with the local-global representation of the BoW approach and with a kernel-based learning method. This paper conducts a comprehensive assessment of the proposed method, trying both a Histograms-Of-Global-Codebook (HOGC) and Signatures-Of-Local-Codebook (SOLC) using an SVMs classifier. Furthermore, motivated by the excellent classification results obtained by using single sorted random projection (SRP) feature, this paper also seeks to combine multiple SRP features using multiple kernel SVMs. Combining descriptors has been explored in [1, 2] in texture classification and texture material categorization.

The rest of this paper is organized as follows. Section 2 gives a very brief review on the background and some related work. Details of our descriptors, the building of our texture classification system, as well as the method for combining multiple SRP features are elaborated in Section 3. Experimental results are given in Section 4.

2. Background and Related Work

A BoW approach represents an image as a collection of regions described by some local descriptors, spatially possibly sparse [1, 2] or dense [3, 5, 6, 7, 8, 23]. An interesting alternative, the so-called MFS-based approach, was proposed by Xu *et al*. [9, 10, 11] where, as opposed to sparse and dense approaches, the MFS approach characterizes the marginal histogram bins of the extracted features using fractal geometry, and this characterization encodes the spatial distribution of the image pixels in the bin.

Rather than a specialized feature extractor, tuned to a particular texture database, random projection [15, 16] refers to the technique of projecting a set of points from a high-dimensional space to a randomly chosen low-dimensional subspace. The technique has been used for combinatorial optimization, information retrieval, face recognition [17] and machine learning. Random features represent a computationally simple and efficient means of preserving texture structure without introducing significant distortion.

The information-preserving and dimensionality reduction power of RP is firmly demonstrated by the theory of compressed sensing (CS) [13, 14], which states that for sparse and compressible signals, a small number of non-adaptive linear measurements in the form of random projections can capture most of the salient information in the signal. Moreover, RP also provides a feasible solution to the well-known Johnson-Lindenstrauss (JL) lemma [16, 15], which states that a point set in a high-dimensional Euclidean space can be mapped down onto a space of dimension logarithmic in the number of points with the distances between the points approximately preserved. RP plays an important role in both JL embedding and CS [18].

3. The Proposed SRP Classifier

Let $\mathcal{D} = \{\{\mathbf{I}_{c,t}\}_{t=1}^{T}\}_{c=1}^{C}$ denote the whole texture dataset, with C distinct texture classes and each class having T texture samples. Suppose T_1 samples are selected as training samples, with the remaining $T_2 = T - T_1$ samples for testing. Let $\mathcal{Y}_{c,t} = \{\underline{\boldsymbol{y}}_{c,t,i}\}_i$ denote the random feature vector set extracted from the corresponding texture sample $\mathbf{I}_{c,t}$, and let $\mathcal{Y}_c = \{\{\underline{\boldsymbol{y}}_{c,t,i}\}_i\}_{t=1}^{T_1}$ denote the random feature vector set extracted from all the training samples available for class c.

The RP classifier uses random measurements of local image patches to perform texture classification, however the fact that the image patch features are not rotationally invariant can be a serious limitation. Existing general methodologies to achieve rotation invariance in the patch vector representation include three main approaches.

1. Add randomly rotated versions of the training samples to the training set when learning textons. This results in clusters having many more points and a much greater spread (Fig. 2 (a)), clearly posing storage and processing challenges, and also creating challenges in clustering the texton space, since the required number of cluster centers k increases with cluster spread. 2. Estimate the dominant gradient orientation of the local patch and align the patch with respect to it [1, 2, 5, 6]. The dominant orientation estimates tend to be unreliable, especially for blob regions which lack strong edges, and for corner regions which have more than one dominant orientation. 3. Marginalize the intensities weighted by the orientation distribution over angle, or compute multilevel histograms at fixed distances from the center of a patch (e.g. the SPIN descriptor adopted in [1, 2]).

Motivated by the striking classification results by Liu *et al*. [3, 4], we would like to further capitalize on the RP approach by proposing a robust variant. An ensemble of patches extracted from a texture produces a cluster of points in some feature space. However, as is illustrated in Fig. 2, rotating the texture patch causes the cluster to be spread along some curve, where the greater class spread and

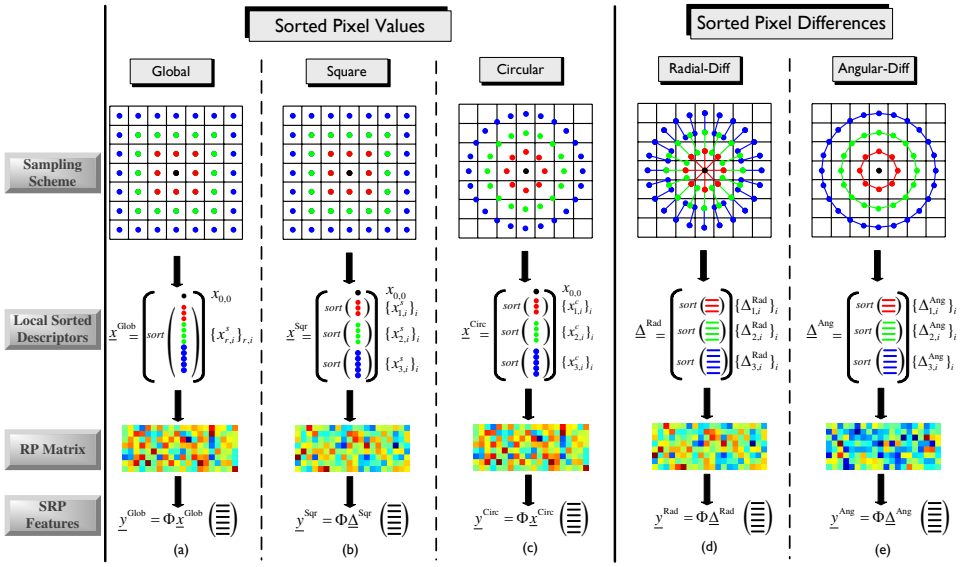

Figure 1. Extracting SRP features on an example local image patch of size 7×7: sorting pixels (a, b, c) or sorting pixel differences (d, e). The pixels may be taken natively on a square grid (a, b) or interpolated to lie on rings of constant radius (c, d, e).

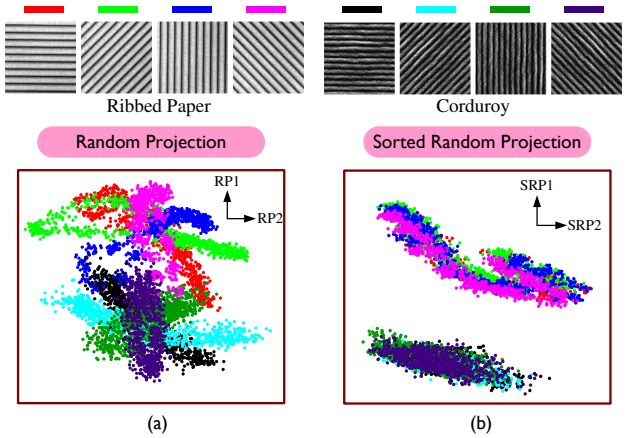

Figure 2. Consider random projections of two different textures at varying orientations. The scatter plots in the bottom row show the random projections for a large number of extracted texture patches. Relative to random projections (a), it is clear that the sorted random projections in (b) offer superior class compactness and separability.

complexity leads to a greater difficulty in the corresponding clustering. As is seen in the figure, the random projections can lose class separability when rotated, and because the clustering requires many more codevectors (i.e., cluster centers or textons) to cover the more complex, rotated cluster shapes, the method has increased computational requirements.

Our goal, then, is to re-localize the cluster, to make it more compact, with the intent of improving separability and reducing the number of textons. To this end, we propose to replace a local texture patch vector with a sorted one: $\underline{\boldsymbol{y}} = \boldsymbol{\Phi}\ sort(\underline{\boldsymbol{x}})$, where we sort over all (or parts) of $\underline{\boldsymbol{x}}$. Fig. 2 (b) shows the potential of this idea: In striking contrast to the scatter plots of the RP features in Fig. 2 (a), the sorted features offer better localization, better separability, and simpler cluster shapes.

3.1. Gray-Scale and Rotation Invariance

Since textures often appear on undulating real-world surfaces, invariances to contrast and rotation must necessarily be local rather than global [8]. Whereas in filter bank based methods a local texture patch is convolved with a set of filters, usually with large support, in [3, 4] random projections in a BoW context provided excellent classification results based on modestly-sized patches. Therefore the RP approach would appear to be well suited for generalization to the rotation-invariant context.

To begin, consider a $(2a+1)\times(2a+1)$ square neighborhood. Let $x^s_{i,j}$ represent the pixel in the jth position in the ith square ring about the neighborhood origin, as illustrated in Fig. 1 (b); similarly define $x^c_{i,j}$ to represent the pixel in the jth position in the ith circular ring, illustrated in Fig. 1 (c). Since the native image pixels lie on a square grid, necessarily the circular pixel values $x^c_{i,j}$ represent a bilinear interpolation of nearby native pixels.

We can modify the RP classifier by replacing the RP measurements of a local patch vector with the RP measurements of globally sorted pixels in a patch:

$$\underline{\boldsymbol{y}} = \boldsymbol{\Phi}\ \underline{\boldsymbol{x}}^{\text{Glob}} \quad (1)$$

$$\underline{\boldsymbol{x}}^{\text{Glob}} = [x_{0,0}, sort([x^s_{1,0}, ..., x^s_{a,p_a-1}])]^T \quad (2)$$

illustrated in Fig. 1 (a).

Clearly, global sorting provides a poor discriminative ability, since crudely sorting the whole patch (the center pixel excluded) leads to an ambiguity of the relationship among pixels from different scales. A natural extension of global sorting is to sort pixels of the same scale, where our schemes follow a strategy similar to some recently developed descriptors like SIFT, SPIN and RIFT [1, 2]. By definition, sorting provides stability against rotation, since no notion of angle or orientation is preserved, however sorting at each scale individually preserves some spatial information. In this way, a compromise is achieved between the conflicting requirements of greater geometric invariance on the one hand and greater discriminative power on the other.

Sorting each ring of pixels loses any sense of spatial coupling, whereas textures clearly possess a great many spatial relationships. Therefore, as the next step in sophistication we propose sorting radial or angular differences, illustrated in Fig. 1 (d) and (e). Of specific interest are the radial differences, which encode the inter-ring structure, thus sorted radial differences will achieve rotation invariance while preserving the relationship between pixels of different rings, a concept which has not been explored by many rotation invariant methods such as LBP. Furthermore, by preserving only pixel *differences*, the method is inherently invariant to additive shifts in texture intensity.

3.2. Texture Description

Given a set of extracted random vectors (local features), many such vectors, extracted for an ensemble of patches, need to be characterized in order to learn the non-local behavior of the texture. This characterization is undertaken using a histogram, for which there are two basic approaches: (1) **HOGC**. A global texton codebook learning stage is needed. A histogram $\underline{\boldsymbol{h}}_{c,t}$ of compressed textons is learned for each particular training sample $\mathbf{I}_{c,t}$ by labeling each of the random feature vectors extracted at its pixels with the closest texton. Each texture class then is represented by a set of normalized histogram models $\mathcal{H}_c = \{\underline{\boldsymbol{h}}_{c,t}\}_t$ corresponding to the training samples of that class.

(2) **SOLC**. There is no global texton codebook learning stage here. Instead of representing each texture image as a histogram of global texton codebook, it represents each image as a signature $S_{c,t} = \{(p_{c,t,i}, \underline{\boldsymbol{u}}_{c,t,i})\}_{i=1}^{K}$, which is learned for each training sample $\mathbf{I}_{c,t}$ by clustering only $\mathcal{Y}_{c,t} = \{\underline{\boldsymbol{y}}_{c,t,i}\}_i$, where K is the number of clusters, $\underline{\boldsymbol{u}}_{c,t,i}$ is the center of the ith cluster, and $p_{c,t,i}$ is the cluster frequencies by counting how many of the pixels belong to cluster $\underline{\boldsymbol{u}}_{c,t,i}$.

A classifier needs only to be able to assess the degree of dissimilarity between two histograms, measured using a χ^2 statistic:

$$D(\underline{\boldsymbol{h}}_1, \underline{\boldsymbol{h}}_2) = \frac{1}{2}\sum_k \frac{[\underline{\boldsymbol{h}}_1(k) - \underline{\boldsymbol{h}}_2(k)]^2}{\underline{\boldsymbol{h}}_1(k) + \underline{\boldsymbol{h}}_2(k)} \tag{3}$$

Table 1. Random projection dimensionality used for the corresponding patch sizes in our experimental evaluation. Theoretical and technical developments for deciding the dimensionality can be found in [4].

Patch Size	5×5	7×7	9×9	11×11	13×13	15×15	17×17	19×19
RP Dim	10	20	30	40	50	60	70	80

The Earth Mover's Distance (EMD) is used to measure the dissimilarity between signatures that are compact representations of distributions. The EMD distance between two signatures $S_1 = \{(p_i, \underline{\boldsymbol{u}}_i)\}_{i=1}^{K_1}$ and $S_2 = \{(q_j, \underline{\boldsymbol{v}}_j)\}_{j=1}^{K_2}$ is defined as follows:

$$D(S_1, S_2) = \frac{\sum_{i=1}^{K_1}\sum_{j=1}^{K_2} \hat{f}_{ij} d(\underline{\boldsymbol{u}}_i, \underline{\boldsymbol{v}}_j)}{\sum_{i=1}^{K_1}\sum_{j=1}^{K_2} \hat{f}_{ij}} \tag{4}$$

where $d(\underline{\boldsymbol{u}}_i, \underline{\boldsymbol{v}}_j)$ is the so-called *ground distance* between cluster centers $\underline{\boldsymbol{u}}_i$ and $\underline{\boldsymbol{v}}_j$ (Euclidean distance is used in this work), and $\hat{f}_{ij}$ is the optimal *flow* which can be determined by solving a linear programming problem. While the EMD works very well on signatures it should not, in general, be applied to histograms. Small histograms invalidate the ground distance as the bin centers are rather far, while computing the EMD on large histograms can be very slow.

3.3. Classification

The benefits of the SVMs for histogram-based classification is clearly demonstrated in [2, 23, 22]; here we use the non-linear SVMs of [20].

Kernels commonly used include polynomials and Gaussian Radial Basis Function (RBF). The Gaussian RBF has been found to perform better for histogram-like features. Incorporating distance functions (*e.g.* the distance measures defined in (3) and (4)) into kernel functions is a well-known method to create problem specific SVMs [2, 20]. For the histogram-based representation, we test the performance with two types of kernels: a Gaussian Radial Basis Function (RBF) kernel

$$K(\underline{\boldsymbol{h}}_i, \underline{\boldsymbol{h}}_j) = \exp(-\gamma\|\underline{\boldsymbol{h}}_i - \underline{\boldsymbol{h}}_j\|^2) \tag{5}$$

and the χ^2 kernel

$$K(\underline{\boldsymbol{h}}_i, \underline{\boldsymbol{h}}_j) = \exp(-\gamma D(\underline{\boldsymbol{h}}_i, \underline{\boldsymbol{h}}_j) \tag{6}$$

for D from (3)). For the signature-based representation, only the EMD kernel is used:

$$K(S_i, S_j) = \exp(-\gamma D(S_i, S_j) \tag{7}$$

for D in (4).

SVMs were originally designed for binary classification. In the case of texture classification we are clearly dealing with a multi-class problem. How to effectively extend the two-class problem for multi-class classification is still an on-going research issue [20]. Of the two basic strategies

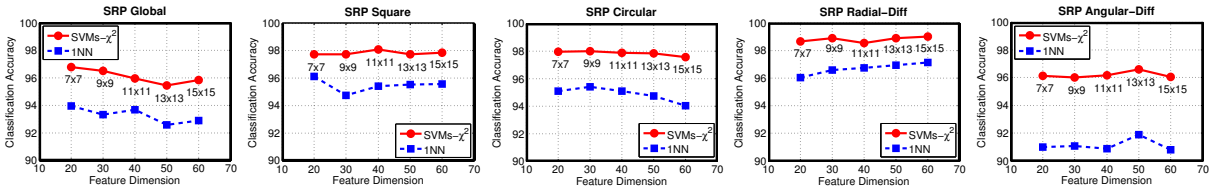

Figure 3. Classification accuracy for all the five descriptors on the KTHTIPS database, comparing the proposed SVMs and NNC. The number of training images per class is 41. The number of textons K per class is 20. The χ^2 distance is used.

Table 2. Summary of texture datasets used in our experiments.

Texture Dataset	Dataset Notation	Image Rotation	Controlled Illumination	Scale Variation	Significant Viewpoint	Texture Classes	Sample Size	Samples per class	Samples in Total
CUReT	$\mathcal{D}^{\mathrm{C}}$	√	√			61	200×200	92	5612
CUReTRot	$\mathcal{D}^{\mathrm{CRot}}$	√	√			61	140×140	92	5612
UIUC	$\mathcal{D}^{\mathrm{UIUC}}$	√		√	√	25	640×480	40	1000
UMD	$\mathcal{D}^{\mathrm{UMD}}$	√		√	√	25	320×240	40	1000
Brodatz	$\mathcal{D}^{\mathrm{B}}$					111	215×215	9	999
KTH-TIPS	$\mathcal{D}^{\mathrm{KT}}$		√	√		10	200×200	81	810

for extension — *one-against-one* and *one-against-other* — it has been shown that *one-against-one* is more suitable for practical use, so in this paper we propose the *one-against-one* technique, which trains a classifier for each possible pair of classes.

Since the descriptors in this paper (especially SRP Rad-Diff) are, on their own, already very discriminative, there may be limitations to applying Multiple Kernel Learning (MKL); furthermore, simple kernel combination methods are capable of reaching the same classification accuracy as MKL. Therefore, we propose to combine kernels in a predefined deterministic way and subsequently use the resulting kernel for SVMs training.

To combine multiple SRP features, we use the kernel $\mathbf{K}(\underline{\boldsymbol{h}}_i, \underline{\boldsymbol{h}}_j) = \exp(-\gamma\chi^2(\underline{\boldsymbol{h}}_i, \underline{\boldsymbol{h}}_j)$. When multiple descriptor types are used, we represent each texture sample using F Bag-of-Words histograms derived from F feature descriptors. The multiple kernel method we consider is to combine several kernels by multiplication. Richer representations can be achieved in such case, since taking products of kernels corresponds to taking a tensor product of their feature spaces, leading to a much higher dimensional feature representation and corresponding SVMs kernel $\mathbf{K}^*(\underline{\boldsymbol{h}}_i, \underline{\boldsymbol{h}}_j) = \prod_{l=1}^{F} \mathbf{K}_l(\underline{\boldsymbol{h}}_i, \underline{\boldsymbol{h}}_j)$.

4. Experimental Evaluation

4.1. Datasets and Experimental Setup

To demonstrate the effectiveness of the proposed approach for robust texture classification we have performed extensive testing in a comprehensive set of *six* commonly-used texture datasets:

For **CUReT**, we use the same subset of images as Varma and Zisserman [5, 6]. The texture appearances vary significantly from one to the next due to being captured under different illuminations and viewing directions. The **CRot** dataset is generated from $\mathcal{D}^{\mathrm{C}}$ by rotating each sample according to a randomly generated angle, uniformly between 0 and 360 degrees. Randomly rotating the texture samples can help validate the rotation invariance, since there is not significant rotation of each texture in $\mathcal{D}^{\mathrm{C}}$.

For **Brodatz** we used the same dataset as [3, 1, 2].

The **UIUC** dataset [1] has been designed to require local invariance. Textures are acquired under significant scale and viewpoint changes, arbitrary rotations, and uncontrolled illumination conditions, even including textures with nonrigid deformation. The **UMD** dataset [11] has been designed in a similar way as $\mathcal{D}^{\mathrm{UIUC}}$.

The **KTH-TIPS** dataset [23] extends CUReT by imaging new samples of ten of the CUReT textures at a subset of the viewing and lighting angles used in CUReT but also over a range of scales. Although KTH-TIPS is designed to be combined with CUReT in testing, we follow Zhang *et al*. [2] in treating it as a stand-alone dataset.

Implementation details. To make the comparisons as meaningful as possible, we use the same experimental settings as in [3] and [6], and the reader is referred to those papers for additional details. The RP dimensions used in our experiments are summarized in Table 1. Each sample is normalized to be zero mean and unit standard deviation, and the extracted SRP vector is normalized via Weber's law. All results are reported over 50 random partitions of training and testing sets. Half of the samples per class are randomly selected for training and the remaining half for testing, except for $\mathcal{D}^{\mathrm{B}}$, where three samples are randomly selected as training and the remaining six as testing and except for those clearly stated. The kernel parameter γ is found by cross-validation within the training set. The values of the parameters and of SVMs are specified using a grid search scheme. In this work, the publicly available *LibSVM* library [21] is employed. The parameters C and γ are searched exponentially in the ranges of $\left[2^{-5}, 2^{18}\right]$ and $\left[2^{-15}, 2^{8}\right]$, respectively, with a step size of 2^1 to probe the highest classification rate.

4.2. Implementation Evaluation

Evaluation of Sorted Descriptors: Fig. 3 plots the classification performance for all five proposed descriptors on dataset $\mathcal{D}^{\mathrm{KT}}$, based on distance measure χ^2. It is clear that the classification rates for SVMs are consistently higher

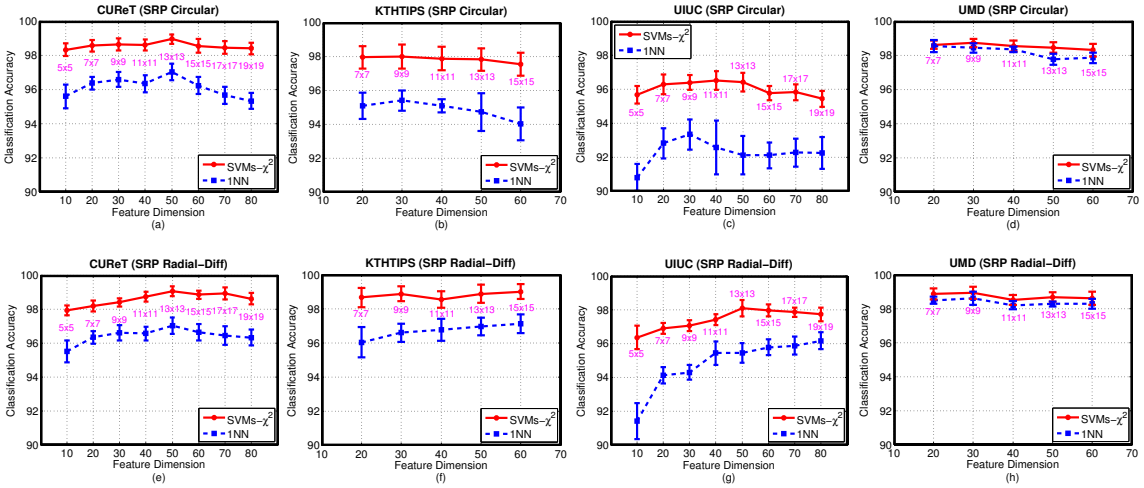

Figure 4. Classification accuracy for SRP Circular (top row) and SRP Radial-Diff (bottom row) on four texture databases: CUReT, KTHTIPS, UIUC and UMD, comparing the proposed SVMs and NNC. The number of training images per class is 46. The number of textons K per class is 10 for CUReT, 20 for KTHTIPS, and 40 for UIUC and UMD. The χ^2 distance is used. A comparison of the accuracies achieved by NNC and SVMs classifiers as a function of the neighborhood size. We see that NNC accuracies are consistently lower than those of the SVMs methods.

Table 3. Classification accuracy (%) of different kernels for SRP Radial-Diff on six texture datasets $\mathcal{D}^{\mathrm{C}}$, $\mathcal{D}^{\mathrm{CRot}}$, $\mathcal{D}^{\mathrm{UIUC}}$, $\mathcal{D}^{\mathrm{UMD}}$, $\mathcal{D}^{\mathrm{B}}$ and $\mathcal{D}^{\mathrm{KT}}$. For HOGC, the number of textons K used per class are 10, 10, 40, 40, 10 and 40 respectively, while for SOLC, the number of textons used per class are all 40. The number of training samples per class are 46, 46, 20, 20, 3 and 41 respectively.

Paradigm	HOGC									SOLC					
Classifier	NNC			SVMs						NNC			SVMs		
Metric	χ^2			RBF	χ^2		RBF	χ^2		EMD			EMD		
Patch size	5×5	9×9	13×13	5×5			9×9			13×13					
$\mathcal{D}^{\mathrm{C}}$	95.51	96.61	96.52	95.53	97.92		96.83	98.39		97.72	**99.05**				
Patch size	9×9	15×15	19×19	9×9			15×15			19×19					
$\mathcal{D}^{\mathrm{CRot}}$	94.55	94.76	95.01	94.69	96.95		95.76	97.05		96.18	**97.45**				
Patch size	5×5	9×9	13×13	5×5			9×9			13×13			5×5	9×9	13×13
$\mathcal{D}^{\mathrm{UIUC}}$	91.40	94.28	95.43	95.66	96.35		96.40	97.06		97.18	**98.08**		78.49	84.58	88.14	88.77	92.40	93.28
Patch size	7×7	9×9	13×13	7×7			9×9			13×13			7×7	9×9	13×13	7×7	9×9	13×13
$\mathcal{D}^{\mathrm{UMD}}$	98.48	98.60	98.26	**98.92**	98.86		98.59	**98.92**		98.53	98.67		90.37	91.37	92.97	94.92	95.16	96.08
Patch size	5×5	9×9	13×13	5×5			9×9			13×13			5×5	9×9	13×13	5×5	9×9	13×13
$\mathcal{D}^{\mathrm{B}}$	93.13	94.74	94.73	93.07	94.44		94.29	95.77		94.24	**96.04**		84.18	89.30	91.38	87.78	90.72	92.67
Patch size	9×9	13×13	15×15	9×9			13×13			15×15			9×9	13×13	15×15	9×9	13×13	15×15
$\mathcal{D}^{\mathrm{KT}}$	97.16	97.35	97.71	98.78	98.95		98.72	99.02		98.65	**99.11**		93.06	95.28	95.27	94.63	95.78	95.20

than those of NNC by roughly 2%-5%, with the best SVMs results achieved by the Radial-Difference method, surpassing an accuracy of 99%, with the high performance of Radial-Difference supporting the large body of work using gradient histograms for recognition. Since all results are averaged over 50 runs with randomly chosen disjoint sets of training and testing samples, Fig. 3 implies that there is a statistically significant difference in performance between SVMs and NNC.

Comparison of SVMs and NNC over datasets: We now compare the performance of SVMs and NNC on four benchmark datasets, with the results shown in Fig. 4. Due to space limitation, and motivated by the results of Fig. 3, we only show the results for the SRP Circular and SRP Radial-Difference descriptors. Clearly, for both proposed descriptors and across all datasets, the SVMs approach significantly and consistently outperforms NNC, consistent with recent work [22, 2, 23] that favors the use of SVMs for texture classification. From the results of $\mathcal{D}^{\mathrm{UIUC}}$ and $\mathcal{D}^{\mathrm{UMD}}$, we can readily observe the relative complexity of the two datasets, as even simple NNC performs well for the simpler dataset $\mathcal{D}^{\mathrm{UMD}}$.

Kernel Evaluation: For SVMs under the HOGC representation, we followed the methodology of [22]; for SOLC we use 40 clusters per sample, consistent with [2, 1]. As a baseline comparison, we also implemented EMD with a nearest neighbor classifier. Table 3 shows the classification results, again using the preferred Radial-Difference. We can see that the HOGC representation performs consistently and significantly better than SOLC, with the χ^2-SVMs in HOGC being the clear winner. Note that both EMD-NNC and EMD-SVMs outperform any single descriptor evaluated in the work of Zhang *et al*. [2] on $\mathcal{D}^{\mathrm{KT}}$. For the EMD kernel in SOLC on $\mathcal{D}^{\mathrm{B}}$, the result of 92.67% is better than the results of single descriptor SPIN and RIFT in [2].

Combining SRP Features: Fig. 5 shows results for three datasets, comparing the combined descriptors with the best single one (SRP Radial-Diff). What is clear from both the table and the figure is that, uniformly across all datasets and across all degrees of training data, the combined classifiers

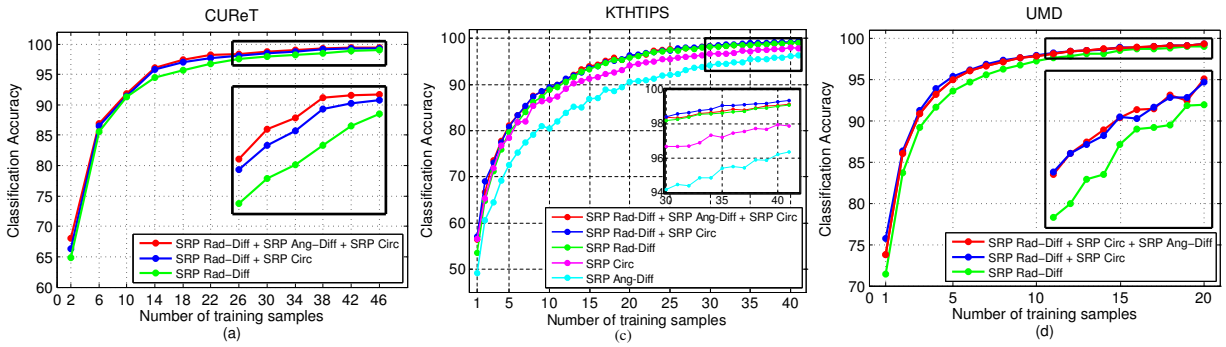

Figure 5. Classification rate vs. number of training samples, comparing single and combined SRP results. The patch sizes used are 13×13, 13×13, and 9×9 for $\mathcal{D}^{\mathrm{C}}$, $\mathcal{D}^{\mathrm{KT}}$, and $\mathcal{D}^{\mathrm{UMD}}$ respectively. The number of textons K used per class is 10, 40, and 40 respectively. The product kernel SVMs with χ^2 distance is used.

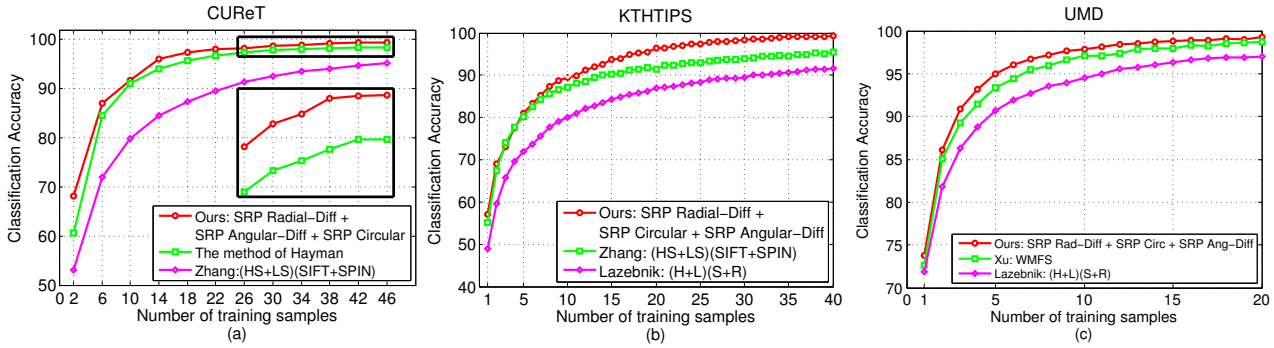

Figure 6. Classification rate vs. number of training samples, comparing the proposed combining SRP approach with various state-of-the-arts methods. Implementation parameters involved are the same as those in Fig. 5.

outperform the single one.

4.3. Comparative Evaluation

Our specific experimental goal is to compare the proposed SRP approach with a comprehensive suite of five recent state-of-the-art methods. (1) **Random Projections** [3]: Each patch is compressed using random projection; both training and testing are performed in the compressed domain. (2) **Patch** [6]: Each local patch of size $\sqrt{n} \times \sqrt{n}$ is reordered into an n-dimensional patch vector. Both training and testing are performed in the patch domain. (3) **The method of Lazebnik *et al*.** [1]: First characterize the texture using Harris-affine corners and Laplacian-affine blobs, with two descriptors (SPIN and RIFT) used for feature extraction, and the NNC is used for classification. (4) **The method of Zhang *et al*.** [2]: Based on the method of Lazebnik *et al*. [1], using three types of descriptors (SPIN, RIFT and SIFT) and a kernel SVMs classifier. (5) **WMFS**[11]: Based on a combination of wavelet transform and multifractal analysis.

All results are taken directly from the original publications, except that the results of Caputo on $\mathcal{D}^{\mathrm{C}}$ and the results of Lazebnik on $\mathcal{D}^{\mathrm{KT}}$ are quoted from the recent comparative study of Zhang *et al*. [2], and the results of Lazebnik with SVMs on $\mathcal{D}^{\mathrm{UMD}}$ from the work of Xu *et al*. [11]. Fig. 6 compares our approach with the state-of-the-art of Zhang *et al*. [2] and Lazebnik *et al*. [1] on three datasets, who have attempted to combine local RIFT, SIFT and SPIN descriptors. Our method consistently improves significantly outperforms competing methods.

Table 4. Comparing the best classification scores achieved by our approach with those achieved by 13 state-of-the-art methods on five datasets. Scores are as originally reported, except for those marked $(*)$ which are taken from the comparative study in Zhang *et al*. [2] The bracketed numbers are the number of training samples per class used for the corresponding databases.

	$\mathcal{D}^{\mathrm{C}}$ (46)	$\mathcal{D}^{\mathrm{B}}$ (3)	$\mathcal{D}^{\mathrm{KT}}$ (41)	$\mathcal{D}^{\mathrm{UIUC}}$ (20)	$\mathcal{D}^{\mathrm{UMD}}$ (20)
1. Our Results	**99.37**%	**97.16**%	**99.29**%	**98.56**%	**99.30**%
SRP Radial-Diff	√	RP Rad-Diff	√	√	√
SRP Circular	√	RP Ang-Diff	√	√	√
SRP Angular-Diff	√			√	√
2. VZ-MR8 [5]	97.43%				
3. VZ-Patch [6]	98.03%	92.9%$(*)$	92.4%$(*)$	97.83%	
4. Caputo *et al*. [23]	98.46%	95.0%$(*)$	94.8%$(*)$	92.0%$(*)$	
5. Lazebnik *et al*. [1]	72.5%$(*)$	88.15%	91.3%$(*)$	96.03%	
6. Mellor *et al*. [19]		89.71%			
7. Zhang *et al*. [2]	95.3%	95.9%	96.1%	**98.7**%	
8. Varma and Ray [12]				**98.76**%	
9. Crosier and Griffin [8]	98.6%		98.5%	**98.8**%	
10. Xu-MFS *et al*. [9]				92.74%	93.93%
11. Xu-OTF *et al*. [10]				97.40%	98.49%
12. Xu-WMFS *et al*. [11]				**98.60**%	98.68%
13. Liu *et al*. [3]	98.52%	**96.34**%	97.71%	96.27%	**99.13**%

Table 4 gives a comprehensive summary of the results for our proposed approach with **12** recent state-of-the-art

results. We can observe that our approach scores very well across all five commonly used datasets, producing what we believe to be the best reported result on the CUReT, Brodatz, KTH-TIPS, and UMD datasets, and for UIUC database our classification rate 98.56% is very close to the best reported results (98.9%). It needs to be emphasized that our method is universal and achieved this state-of-the-art performance without any database-specific parameter tuning.

Finally, before concluding this work, we present some results on the very chalenging Flickr Material Database (FMD) used in the work of Liu *et al*. [24]. FMD is designated for material recognition, which is closely related to, but different from, texture recognition. Our single SRP Radial-Diff descriptor gives 48.2% classification accuracy[1], which significantly outperformed the rate 35.2% of the best single feature (*i.e*. the texture feature SIFT) reported by Liu *et al*. [24] with their aLDA system. Our preliminary result 48.2% is even better than the best reported rate of 44.6% by Liu *et al*. [24], who have combined color, texture, shape and edge features with aLDA.

5. Conclusions

This paper has introduced a simple, robust, but remarkably capable texture classification system, combining sorted random projections with a bag-of-words approach and a SVMs classifier. The experiments reveal that (1) SVMs outperforms NNC; (2) dense HOGC outperforms SOLC; (3) the Radial-Difference sorting approach outperforms the other proposed sorting methods; and (4) Combing SRP features is found to produce consistently better classification performance than a single SRP feature. We compared the proposed approach with **12** recent state-of-the-art methods, and the proposed approach outperformed the state-of-the-art in three different standard benchmark texture databases.

References

[1] S. Lazebnik, C. Schmid, and J. Ponce. A sparse texture representation using local affine regions. *TPAMI*, 27(8): 1265–1278, 2005.

[2] J. Zhang, M. Marszalek, S. Lazebnik, and C. Schmid. Local features and kernels for classification of texture and object categories: a comprehensive study. *IJCV*, 73(2): 213–238, 2007.

[3] L. Liu, P. Fieguth, and G. Kuang. Compressed sensing for robust texture classification. In *ACCV*, pages 378–391, 2010.

[4] L. Liu, P. Fieguth. Texture classification from random features. *TPAMI*, To appear.

[1] Due to the lack of space, only an elementary experimental result is provided. We used 100 textons per class, patch size 7×7, SVMs classification. There are 10 material categories in the FMD, with each category containing 100 images, out of which 50 images per category as training and the left 50 as testing, classification accuracy was averaged over 200 random partitions of training and testing sets.

[5] M. Varma, and A. Zisserman. A statistical approach to texture classification from single images. *IJCV*, 62(1-2): 61–81, 2005.

[6] M. Varma, and A. Zisserman. A statistical approach to material classification using image patches. *TPAMI*, 31(11): 2032–2047, 2009.

[7] T. Ojala, M. Pietikäinen, and T. Mäenpää. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *TPAMI*, 24(7): 971–987, 2002.

[8] M. Crosier, and L. D. Griffin. Using basic image features for texture classification. *IJCV*, 88(3): 447–460, 2010.

[9] Y. Xu, H. Ji, C. Fermuller. View point texture description using fractal analysis. *IJCV*, 83(1): 85–100, 2009.

[10] Y. Xu, S. Huang, H. Ji, and C. Fermüller. Combining powerful local and global statistics for texture description. In *CVPR*, 2009.

[11] Y. Xu, X. Yang, H. Ling, and H. Ji. A new texture descriptor using multifractal analysis in multi-orientation wavelet pyradmid. In *CVPR*, 2010.

[12] M. Varma, and D. Ray. Learning The Discriminative Power-Invariance Trade-Off. In *ICCV* , 2007.

[13] E. Candès, and T. Tao. Near-optimal signal recovery from random projections: universal encoding stratigies? *IEEE Trans. Information Theory*, 52(12): 5406–5425, 2006.

[14] D. Donoho, Compressed sensing. *IEEE Trans. Inform. Theory*, 52(4): 1289–1306, 2006.

[15] S. Dasgupta, and A. Gupta. An elementary proof of a theorem of Johnson and Lindenstrauss. *Random Structures and Algorithms*, 22(1): 60–65, 2003.

[16] D. Achlioptas. Database-friendly random projections. In *Proceedings of the Twentieth ACM Symposium on Principles of Database Systems*, pages 274–281, 2001.

[17] J. Wright, A. Yang, A. Ganesh, S. S. Sastry, and Y. Ma. Robust face recognition via sparse representation. *TPAMI*, 31(2): 210–217, 2009.

[18] R. G. Baraniuk, M. Davenport, R. A. DeVore, and M. Wakin. A simple proof of the restricted isometry property for random matrices. *Constructive Approximation*, 28(3): 253–263, 2008.

[19] M. Mellor, B.-W. Hong, and M. Brady. Locally rotation, contrast, and scale invariant descriptors for texture analysis. *TPAMI*, 30(1): 52–61, 2008.

[20] B. Schölkopf, and A. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization and Beyond*, MIT Press, Cambridge, MA, 2002.

[21] C.-C. Chang, and C.-J. Lin. *LIBSVM: a library for support vector machines*. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`, 2001.

[22] B. Caputo, E. Hayman, and P. Mallikarjuna. Class-specific material categorization. In *ICCV*, pages 1597–1604, 2005.

[23] B. Caputo, E. Hayman, M. Fritz, and J.-O. Eklundh. Classifying materials in the real world. *Image and Vision Computing*, 28: 150–163, 2010.

[24] C. Liu, L. Sharan, E. Adelson, R. Rosenholtz. Exploring features in a bayesian framework for material recognition. In *CVPR*, 2010.