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
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 $\mathcal{D} = \{\{I_{c,t}\}_{t=1}^{T^2}\}_{c=1}^{C}$ denote the whole texture dataset, with $C$ distinct texture classes and each class having $T$ texture samples. Suppose $T_i$ samples are selected as training samples, with the remaining $T_2 = T - T_1$ samples for testing. Let $\mathcal{Y}_{c,t} = \{y_{c,t,i}\}_1^{T_1}$ denote the random feature vector set extracted from the corresponding texture sample $I_{c,t}$, and let $\mathcal{Y}_c = \{y_{c,t,i}\}_1^{T_1}$ 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
sorted random projections in (b) offer superior class compactness and separability. 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 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 \(x_{i,j}^a\) represent the pixel in the \(j\)th position in the \(i\)th square ring about the neighborhood origin, as illustrated in Fig. 1 (b); similarly define \(x_{i,j}^c\) to represent the pixel in the \(j\)th position in the \(i\)th circular ring, illustrated in Fig. 1 (c). Since the native image pixels lie on a square grid, necessarily the circular pixel values \(x_{i,j}^c\) 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:

\[
\begin{align*}
\mathbf{y} &= \Phi \mathbf{x}^\text{Glob} \\
\mathbf{x}^\text{Glob} &= [x_{0,0}, sort([x_{1,0}^a, \ldots, x_{a,p_a-1}^a])]^T
\end{align*}
\]

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 \( \mu_{c,t} \) of compressed textons is learned for each particular training sample \( I_{c,t} \) 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 \( H_c = \{ \mu_{c,t} \}_t \) 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 \( S_{c,t} = \{(p_{c,t,i}, u_{c,t,i})\}_i^K \), which is learned for each training sample \( I_{c,t} \) by clustering only \( Y_{c,t} = \{ y_{c,t,i} \}_i^K \), where \( K \) is the number of clusters, \( u_{c,t,i} \) 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 \( u_{c,t,i} \).

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

\[
D(h_1, h_2) = \frac{1}{2} \sum_k \frac{[h_1(k) - h_2(k)]^2}{h_1(k) + h_2(k)}
\]  
(3)

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 \( S_1 = \{(p_1, u_i)\}_{i=1}^{K_1} \) and \( S_2 = \{(q_j, v_j)\}_{j=1}^{K_2} \) is defined as follows:

\[
D(S_1, S_2) = \sum_{i=1}^{K_1} \sum_{j=1}^{K_2} f_{ij} d(u_i, v_j)
\]  
(4)

where \( d(u_i, v_j) \) is the so-called ground distance between cluster centers \( u_i \) and \( v_j \) (Euclidean distance is used in this work), and \( f_{ij} \) 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

\[
K(h_1, h_2) = \exp(-\gamma \|h_1 - h_2\|^2)
\]  
(5)

and the \( \chi^2 \) kernel

\[
K(h_1, h_2) = \exp(-\gamma D(h_1, h_2))
\]  
(6)

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

\[
K(S_i, S_j) = \exp(-\gamma D(S_i, S_j))
\]  
(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
Table 2. Summary of texture datasets used in our experiments.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Notation</th>
<th>Image Size</th>
<th>Rotation</th>
<th>Illumination</th>
<th>Scale Variation</th>
<th>Viewpoint Variation</th>
<th>Texture Class</th>
<th>Sample Size</th>
<th>Samples per Class</th>
<th>Total Samples</th>
</tr>
</thead>
<tbody>
<tr>
<td>UMD</td>
<td>D^{15x15}</td>
<td>1024 x 1024</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>15</td>
<td>200</td>
<td>20</td>
<td>3600</td>
</tr>
<tr>
<td>UIUC</td>
<td>D^{15x15}</td>
<td>1024 x 1024</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>15</td>
<td>200</td>
<td>20</td>
<td>3600</td>
</tr>
<tr>
<td>CUReT</td>
<td>D^{15x15}</td>
<td>1024 x 1024</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>15</td>
<td>200</td>
<td>20</td>
<td>3600</td>
</tr>
<tr>
<td>KTH-TIPS</td>
<td>D^{15x15}</td>
<td>1024 x 1024</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>15</td>
<td>200</td>
<td>20</td>
<td>3600</td>
</tr>
</tbody>
</table>

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 SVM training.

To combine multiple SRP features, we use the kernel $K(h_i, h_j) = \exp(-\gamma \chi^2(h_i, h_j))$. 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 $K'(h_i, h_j) = \prod_{i=1}^{F} K_i(h_i, h_j)$.

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 $D^C$ 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 $D^C$.

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 $D^{UIUC}$.

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 $D^{UIUC}$, where three samples are randomly selected as training and the remaining six as testing and except for those clearly stated. The kernel parameter $\gamma$ 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 $\gamma$ are searched exponentially in the ranges of $[2^{-5}, 2^{18}]$ and $[2^{-15}, 2^8]$, 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 $D^{K}$, based on distance measure $\chi^2$. 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: CURêT, KTH-TIPS, UIUC and UMD, comparing the proposed SVMs and NNC. The number of textons $K$ per class is 10 for CURêT, 20 for KTH-TIPS, and 40 for UIUC and UMD. The $\chi^2$ 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 $D^C$, $D^{C/R}$, $D^{HOGC}$, $D^{EMD}$, $D^B$ and $D^{KT}$. 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 images per class are 46, 46, 20, 20, 3 and 41 respectively.

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-Diff 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 $D^{HOGC}$ and $D^{EMD}$, we can readily observe the relative complexity of the two datasets, as even simple NNC performs well for the simpler dataset $D^{EMD}$.

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 $\chi^2$-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 $D^{KT}$. For the EMD kernel in SOLC on $D^B$, 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
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 $D^C$ and the results of Lazebnik on $D^{\text{LMF}}$ are quoted from the recent comparative study of Zhang et al. [2], and the results of Lazebnik with SVMs on $D^{\text{UMD}}$ 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 gives a comprehensive summary of the results for our proposed approach with 12 recent state-of-the-art methods.

![Figure 5. Classification rate vs. number of training samples, comparing single and combined SRP results. The patch sizes used are $13 \times 13$, $13 \times 13$, and $9 \times 9$ for $D^C$, $D^{\text{LMF}}$, and $D^{\text{UMD}}$ respectively. The number of textons $K$ used per class is 10, 40, and 40 respectively. The product kernel SVMs with $\chi^2$ distance is used.](image1)

![Figure 6. Classification rate vs. number of training samples, comparing the proposed combining SRP approach with various state-of-the-art methods. Implementation parameters involved are the same as those in Fig. 5.](image2)
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 challenging 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, 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

1Due 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.