Compressed Sensing for Robust Texture Classification

Li Liu[1], Paul Fieguth[2] and Gangyao Kuang[1]

[1]School of Electronic Science and Engineering, National University of Defense Technology, Changsha, Hunan, China 410043
dreamliu2010@gmail.com; kuangyeats@vip.sina.com
[2]Department of System Design Engineering, University of Waterloo, Waterloo, Ontario, Canada N2L 3G1
pfieguth@uwaterloo.ca

Abstract. This paper presents a simple, novel, yet very powerful approach for texture classification based on compressed sensing. At the feature extraction stage, a small set of random features is extracted from local image patches. The random features are embedded into a bag-of-words model to perform texture classification, thus learning and classification are carried out in the compressed domain. The proposed unconventional random feature extraction is simple, yet by leveraging the sparse nature of texture images, our approach outperforms traditional feature extraction methods which involve careful design and complex steps. We report extensive experiments comparing the proposed method to the state-of-the-art in texture classification on four databases: CUReT, Brodatz, UIUC and KTH-TIPS. Our approach leads to significant improvements in classification accuracy and reductions in feature dimensionality, exceeding the best reported results on CUReT, Brodatz and KTH-TIPS.

1 Introduction

The classification of texture is a key problem in computer vision and pattern recognition, especially for real-world texture images with great intra-class variability due to illumination variations, rotations, viewpoint changes and nonrigid deformations.

By extracting features from a local patch, most feature extraction methods focus on local texture information, characterized by the gray level patterns surrounding a given pixel; however texture is also characterized by its global appearance, representing the repetition of and the relationship among local patterns. Recently, a bag-of-words (BoW) model, borrowed from the text literature, has opened up new prospects for texture classification [1][2][3][4][5][6]. The BoW model encodes both the local texture information, by using features from local patches to form textons, and the global texture appearance, by statistically computing an orderless histogram.

Very popular is the use of large support filter banks to extract texture features at multiple scales and orientations [1][2][3]. However, more recently, in [4] the

Table 1. Summary of texture datasets used in the experiments. Example textures are provided in the supplemental material.

Texture Dataset	Dataset Notation	Image Rotation	Controlled Illumination	Scale Variation	Texture Classes	Sample Size	Samples per Class	Samples in Total
CUReT	$\mathcal{D}^{\text{CUReT}}$	$\surd$	$\surd$		61	200×200	92	5612
UIUC	$\mathcal{D}^{\text{UIUC}}$	$\surd$		$\surd$	25	640×480	40	1000
BrodatzFull	$\mathcal{D}^{\text{BFull}}$				111	215×215	9	999
Brodatz90	$\mathcal{D}^{B90}$				90	128×128	25	2250
KTH-TIPS	$\mathcal{D}^{\text{KT}}$		$\surd$	$\surd$	10	200×200	81	810

authors challenge the dominant role that filter banks have been playing in texture classification, claiming that classification based on textons directly learned from the raw image patches outperforms textons based on filter bank responses.

The key parameter in patch-based classification is patch size. Small sizes cannot capture large-scale structures that may be the dominant features of some textures, and are highly sensitive to noise and other variations, whereas large patch sizes lead to high storage and computational complexity. Therefore, it is natural to ask whether high-dimensional patch vectors can be projected into a lower dimensional subspace without suffering great information loss.

The compressed sensing (CS) approach [7][8], which has been the motivation for this research, is therefore appealing because of its surprising result that high-dimensional sparse data can be accurately reconstructed from just a few nonadaptive linear random projections. When applying CS to texture classification, the key question is therefore how much information about high-dimensional sparse texture signals in local image patches can be preserved by random projections, and whether this leads to any advantages in classification.

The proposed method is computationally simple, yet very powerful. Instead of performing texture classification in the original high-dimensional patch space or making efforts to figure out a suitable feature extraction method, by using random projections of local patches we perform texture classification in a much lower-dimensional compressed patch space. The theory of CS implies that the precise choice of the number of features is no longer critical: a small number of random features, above some threshold, contains enough information to preserve the underlying local texture structure.

Finally, since textures often appear on undulating real world surfaces, the invariances to illumination, rotation, viewpoint, and scale must also necessarily be local rather than global [9]. To avoid the complexity of local invariance in [5] [6], in this paper, we develop simple, yet very powerful rotation-invariant descriptors by *sorting* local patches.

Section 2 reviews the CS background. With the development of the CS approach in Section 3 and the rotation invariance descriptors in Section 4, Section 5 provides extensive experimental results for the CS and sorted CS classifiers and comparative evaluation with the current state-of-the-art on the databases listed in Table 1.

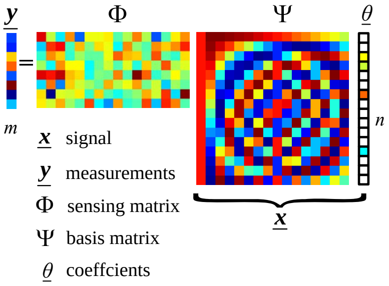

Fig. 1. Compressed Sensing measurement process

2 CS Background

The theory of compressed sensing has recently been brought to the forefront by the work of Candès and Tao [7] and Donoho [8], who have shown the advantages of random projections for capturing information about sparse or compressible signals. The premise of CS is that a small number of nonadaptive linear measurements of a compressible signal or image contain enough information for near perfect reconstruction and processing. This emerging theory has generated enormous amounts of research with applications to high-dimensional geometry [10], image reconstruction [11], and machine learning [12] etc.

CS exploits the fact that many signal classes have a low-dimensional structure compared to the high-dimensional ambient space. Therefore, a small number of nonadaptive measurements in the form of randomized projections can capture most of the salient information in a signal and can approximate the signal well and allow it to be reconstructed.

- ⧫ The key assumption in CS is that of *sparsity* or *compressibility*. Let $\underline{\boldsymbol{x}} \in \mathbb{R}^{n\times 1}$ be an unknown signal of length n and $\boldsymbol{\Psi} = [\underline{\boldsymbol{\psi}}_1 \; ... \; \underline{\boldsymbol{\psi}}_n]$ an orthonormal basis, where $\underline{\boldsymbol{\psi}}_i \in \mathbb{R}^{n\times 1}$, such that

$$\underline{\boldsymbol{x}} = \sum_{i=1}^{n} \theta_i \underline{\boldsymbol{\psi}}_i = \boldsymbol{\Psi}\underline{\boldsymbol{\theta}} \tag{1}$$

  where $\underline{\boldsymbol{\theta}} = [\theta_1 \; ... \; \theta_n]^T$ denotes the vector of coefficients that represents $\underline{\boldsymbol{x}}$ in the basis $\boldsymbol{\Psi}$, as illustrated in Fig. 1. Signal $\underline{\boldsymbol{x}}$ is said to be *sparse* or *compressible* if most of the coefficients in $\underline{\boldsymbol{\theta}}$ are zero or can be discarded without much loss of information.
- ⧫ Let $\boldsymbol{\Phi}$ be an $m \times n$ sampling matrix, with $m \ll n$, such that

$$\underline{\boldsymbol{y}} = \boldsymbol{\Phi}\underline{\boldsymbol{x}} = \boldsymbol{\Phi}\boldsymbol{\Psi}\underline{\boldsymbol{\theta}} \tag{2}$$

  where $\underline{\boldsymbol{y}}$ is an $m \times 1$ vector of measurements. The sampling matrix $\boldsymbol{\Phi}$ must allow the reconstruction of length-n signal $\underline{\boldsymbol{x}}$ from m measurements in $\underline{\boldsymbol{y}}$.

Since the transformation from $\underline{\boldsymbol{x}}$ to $\underline{\boldsymbol{y}}$ is a *dimensionality reduction*, in general there is an information loss, however the measurement matrix $\boldsymbol{\Phi}$ can be shown to preserve the information in sparse and compressible signals if it satisfies the so-called *restricted isometry property* (RIP) [13]. Intriguingly, a large class of random matrices have the RIP with high probability [7][8][13].

- Signal reconstruction takes the m measurements in $\underline{\boldsymbol{y}}$, the random measurement matrix $\boldsymbol{\Phi}$, and the basis $\boldsymbol{\Psi}$ to reconstruct $\underline{\boldsymbol{\theta}}$. A large number of approaches have been proposed to solve the reconstruction problem, however the algorithms tend to be computationally burdensome.

CS theory has been used for classification in the SRC algorithm [12] for face recognition. In contrast to the texture problem, however, the SRC algorithm is based on global features, whereas texture classification almost certainly depends on the relationship between a pixel and its neighborhood. Furthermore SRC is reconstruction based, explicitly reconstructing the sparse $\underline{\boldsymbol{\theta}}$, a computationally intensive step we wish to avoid.

3 The CS Classifier

Let us begin by formulating a basic CS classifier, with a robust extension development in Section 4. The premise underlying CS is one of signal sparsity or compressibility, and the compressibility of textures is certainly well established. Certainly most natural images are compressible, as extensive experience with the wavelet transform has demonstrated. Textures, being roughly stationary/periodic, are all the more sparse. Furthermore from the large literature on texture classification via feature extraction, the degrees of freedom underlying a texture are clearly few in number.

The local patch vector $\underline{\boldsymbol{x}} \in \mathbb{R}^{n\times 1}$ is embedded into a lower-dimensional space $\underline{\boldsymbol{y}} \in \mathbb{R}^{m\times 1}$:

$$\underline{\boldsymbol{y}} = \boldsymbol{\Phi}\underline{\boldsymbol{x}} \tag{3}$$

ideally where $m \ll n$. Clearly $\boldsymbol{\Phi} \in \mathbb{R}^{m\times n}$, $m < n$ loses information in general, since $\boldsymbol{\Phi}$ has a null space, implying the indistinguishability between $\underline{\boldsymbol{x}}$ and $\underline{\boldsymbol{x}} + \underline{\boldsymbol{z}}$, for $\underline{\boldsymbol{z}} \in \mathcal{N}(\boldsymbol{\Phi})$. The challenge in identifying an effective feature extractor $\boldsymbol{\Phi}$ is to have the null-space of $\boldsymbol{\Phi}$ orthogonal to the low-dimensional subspace of sparse signal $\underline{\boldsymbol{x}}$.

Ideally, we wish to ensure that $\boldsymbol{\Phi}$ is information-preserving, by which we mean that $\boldsymbol{\Phi}$ provides a stable embedding that approximately preserves distances between all pairs of signals, such that for any two patches, $\underline{\boldsymbol{x}}_1$ and $\underline{\boldsymbol{x}}_2$

$$1 - \epsilon \leq \frac{\|\boldsymbol{\Phi}(\underline{\boldsymbol{x}}_1 - \underline{\boldsymbol{x}}_2)\|_2}{\|\underline{\boldsymbol{x}}_1 - \underline{\boldsymbol{x}}_2\|_2} \leq 1 + \epsilon \tag{4}$$

for small $\epsilon > 0$. One of the key results in [13] from CS theory is the Restricted Isometry Property, which states that (4) is indeed satisfied by certain random matrices, including a Gaussian random matrix $\boldsymbol{\Phi}$. It is on this basis that we

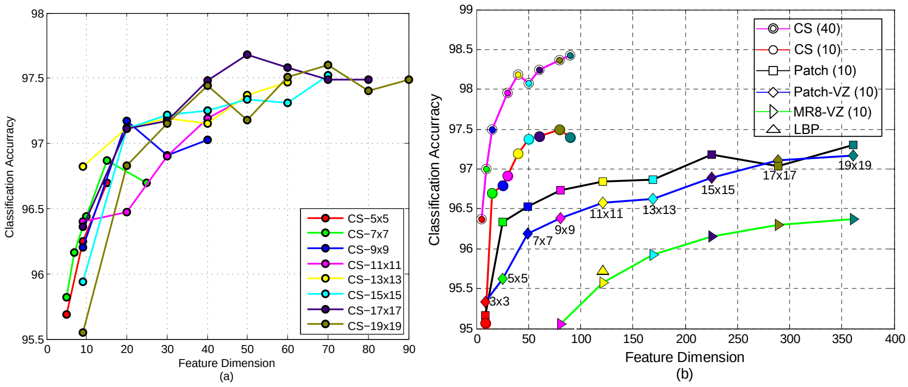

Fig. 2. Classification results on $\mathcal{D}^{\text{CUReT}}$ as a function of feature dimensionality (Except for LBP, whose results are shown as a function of patch size). The bracketed values denote the number of textons K per class. "Patch-VZ" and "MR8-VZ" results are quoted directly from the paper of Varma and Zisserman [4]. Classification rates obtained based on the same patch size are shown in the same color.

propose to use the emerging theory of compressed sensing to rethink texture classification.

We wish to preserve both local texture information, contained in a local image patch, and global texture appearance, representing the repetition of the local textures and the relationship among them. We choose a texton-based approach, an effective local-global representation [1][2][3][4], trained by adaptively partitioning the feature space into clusters using K-means. For an input data set $\mathcal{Y} = \{\underline{\boldsymbol{y}}_1, ..., \underline{\boldsymbol{y}}_{|\mathcal{Y}|}\}$, $\underline{\boldsymbol{y}}_i \in \mathbb{R}^{d\times 1}$, and an output texton set $\mathcal{W} = \{\underline{\boldsymbol{w}}_1, ..., \underline{\boldsymbol{w}}_K\}$, $\underline{\boldsymbol{w}}_i \in \mathbb{R}^{d\times 1}$, the quality of a clustering solution is measured by the average quantization error

$$Q(\mathcal{Y}, \mathcal{W}) = \frac{1}{|\mathcal{Y}|} \sum_{j=1}^{|\mathcal{Y}|} \min_{1\leq k\leq K} \|\underline{\boldsymbol{y}}_j - \underline{\boldsymbol{w}}_k\|_2^2 \qquad (5)$$

However, $Q(\mathcal{Y}, \mathcal{W})$ goes as $K^{-2/d}$ for large K [14], a problem when d is large, since K is then required to be extremely large to obtain satisfactory cluster centers, with computational and storage complexity consequences. On the other hand, Varma and Zisserman [4] have shown that image patches contain sufficient information for texture classification, arguing that the inherent loss of information in the dimensionality reduction of feature extraction leads to inferior classification performance. CS addresses the dilemma between these two perspectives very neatly. The high-dimensional texture patch space has an intrinsic dimensionality that is much lower, therefore CS is able to perform feature extraction without information loss. On the basis of the above analysis, we claim that the CS and BoW approaches are complementary, and will together lead to superior performance for texture classification.

Table 2. Experimental results for the CS classifier on $\mathcal{D}^{\mathrm{B90}}$, 13 samples per class for training and 12 for testing. Means and standard deviations have been computed over 20 runs. Ten textons used per class.

	Patch Size		
Method	3×3	5×5	7×7
Dim	5	10	25
CS	$94.3\% \pm 0.30\%$	$\mathbf{95.4\% \pm 0.39\%}$	$95.0\% \pm 0.34\%$
Dim	9	25	49
Patch	$94.0\% \pm 0.11\%$	$94.7\% \pm 0.21\%$	$94.8\% \pm 0.30\%$
Scale	1	2	3
LBP^{riu2}	$87.7\% \pm 0.63\%$	$93.6\% \pm 0.34\%$	$94.8\% \pm 0.37\%$

Table 3. Comparison of highest classification performance on $\mathcal{D}^{\mathrm{CUReT}}$ with a common experimental setup, except for Zhang [6] who used EMD/SVM classifier.

Method	LBP	MR8	Patch	Patch-MRF	Zhang et al.	**CS**
Accuracy	95.72%	97.43%	97.17%	98.03%	95.5%	**98.43%**

The actual classification algorithm is the texton-based method of [4] except that instead of using image patch vector $\underline{\boldsymbol{x}}$, the compressed sensing measurements $\underline{\boldsymbol{y}} = \boldsymbol{\Phi}\underline{\boldsymbol{x}}$ derived from $\underline{\boldsymbol{x}}$ are used as features, where we choose $\boldsymbol{\Phi}$ to be a Gaussian random matrix, i.e., the entries of $\boldsymbol{\Phi}$ are independent zero-mean, unit-variance normal.

Figure 2 plots the classification accuracy for the **CUReT dataset** $\mathcal{D}^{\mathrm{CUReT}}$ (see Table 1), using the same subset of images and the same experimental setup as Varma and Zisserman [3] [4]. Figure 2 (b), in particular, presents a comparison of CS, Patch, MR8, and LBP, with the results averaged over tens of random partitions of training and testing sets. The CS method significantly outperforms all other methods, a clear indication that the CS matrix preserves the salient information contained in the local patch (as predicted by CS theory) and that performing classification in the compressed patch space is not a disadvantage. In contrast to the Patch method, not only does CS offer higher classification accuracy, but also at a much lower-dimensional feature space, reducing storage requirements and computation time, and allowing more textons per class.

Table 2 shows the classification accuracy on **Brodatz90** dataset $\mathcal{D}^{\mathrm{B90}}$. Due to the impressive diversity and perceptual similarity of some textures in the Brodatz database, some of which essentially belong to the same class but at different scales, while others are so inhomogeneous that a human observer would arguably be unable to group their samples correctly, 90 texture classes from the Brodatz album were kept. The proposed method performs better than the Patch method, although by a relatively small margin. The example demonstrates that the proposed CS method can successfully classify 90 texture classes from the Brodatz dataset, despite the large number of classes contained in $\mathcal{D}^{B}$ which can cause a high risk of mis-classification.

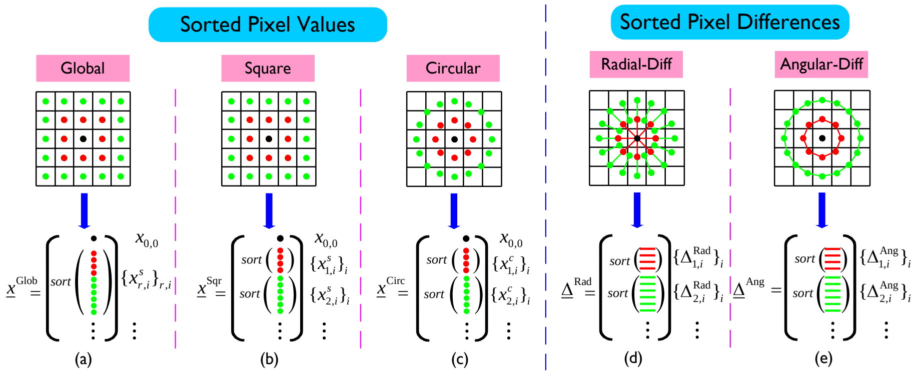

Fig. 3. Sorting schemes on an example patch of size 5 × 5-pixels: sorting pixels (a, b, c) or sorting pixel differences (d, e). The pixels may be taken natively on a square grid (a, b) or interpolated to lie on rings of constant radius (c, d, e).

Table 3 presents the overall best classification performance achieved for any parameter setting. The proposed CS method gives the highest classification accuracy of 98.43%, even higher than the best of Patch-MRF in [4], despite the fact that the model dimensionality of the Patch-MRF method is far larger than that of the CS.

4 The SCS Classifiers

Motivated by the striking classification results in Figure 2 and Table 3, we would like to further capitalize on the CS approach by proposing a robust variant. Existing schemes to achieve rotation invariance in the patch vector representation include estimating the dominant gradient orientation of the local patch [3] [4], marginalizing the intensities weighted by the orientation distribution over angle, and adding rotated patches to the training set. The dominant orientation estimates tend to be unreliable, especially for blob regions which lack strong edges at the center, and finding the dominant orientation for each local patch is computationally expensive. To avoid the ambiguity of identifying a dominant direction, and the clustering challenge with learning over all rotated patches, instead we just use

$$\underline{\boldsymbol{y}} = \boldsymbol{\Phi}\ sort(\underline{\boldsymbol{x}}) \tag{6}$$

where we sort over all (or parts) of $\underline{\boldsymbol{x}}$. Since sorting ignores the ordering of elements in $\underline{\boldsymbol{x}}$, the $sort(\underline{\boldsymbol{x}})$ is clearly rotation invariant (excepting the effects of pixellation). Classification using (6) will be referred to as Sorted CS (SCS).

4.1 Sorted Pixel Values

For example, suppose we reorder the patch vector by taking the center pixel value $x_{0,0}$ of the patch of size $(2a + 1) \times (2a + 1)$ as its first entry and simply

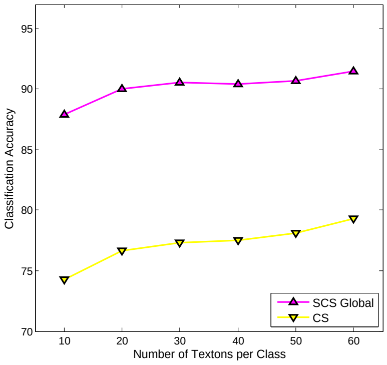

Fig. 4. Comparison of the simplest sorted CS against basic CS: classification accuracy as a function of number of textons per class on $\mathcal{D}^{\text{UIUC}}$ with 20 samples per class for training, using a patch of size 9 × 9 and a dimensionality of 30. Results are averaged over tens of random partitionings of the training and testing set.

sort the other $n-1$ pixels:

$$\underline{\boldsymbol{x}}^{\text{Glob}} = [x_{0,0}, sort([x_{1,0}, ..., x_{1,p_1-1}, ..., x_{a,0}, ..., x_{a,p_a-1}])]^T \tag{7}$$

where $x_{r,i}$, $0 \leq i < p_r$ refers to the rth concentric square of pixels (see Figure 3 (a)) and $p_r = 8r$, $1 \leq r \leq a$. Since sorting deletes all location information, clearly $\underline{\boldsymbol{x}}^{\text{Glob}}$ is invariant to rotation. The advantages of our sorting approach include simplicity and noise robustness.

Figure 4 motivates this idea, showing a jump (from below 80% to above 90%) in classification performance, compared to basic CS in classifying the challenging UIUC database. This surprising experimental result, despite the quality of the basic CS classifier in Table 3, confirms the effectiveness of the sorting strategy.

Clearly, global sorting provides a poor discriminative ability, since crude sorting over 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. We propose two kinds of sorting schemes, illustrated in Figure 3 (b) and (c). Our schemes follow a strategy similar to some recently developed descriptors like SIFT [15], SPIN and RIFT [5]: They subdivide the region of support, and instead of sorting, they adopt histogramming strategy and compute a histogram of appearance attributes (pixel values or gradient orientations) inside each subregion.

In our proposed approach, sorting provides stability against rotation, while 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. As can be seen from Figure 5, sorting over concentric squares or circular rings both offer an improvement over global sorting.

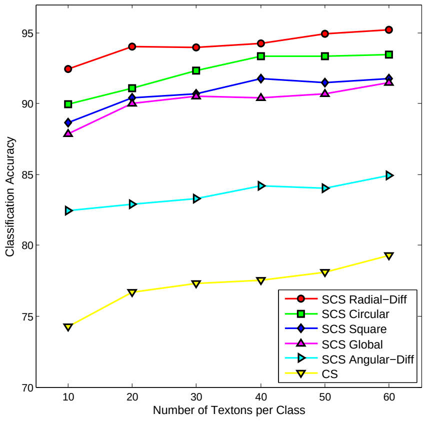

Fig. 5. Like Figure 4, but comparing all sorting schemes with basic CS.

4.2 Sorted Pixel Differences

Sorting each ring of pixels loses any sense of spatial coupling, whereas textures clearly possess a great many spatial relationships. Therefore we propose sorting radial or angular differences, illustrated in Figure 3 (d) and (e). It is worth noting that gray-level differences have been successfully used in a large number of texture analysis studies [16][17][18][19].

We propose pixel differences in radial and angular directions on a circular grid, different from the traditional pixel differences which are computed in horizontal and vertical directions on a regular grid. In particular, radial differences encode the inter-ring structure, thus sorted radial differences can achieve rotation invariance while preserving the relationship between pixels of different rings, which has not been explored by many rotation invariant methods such as LBP. The sorted radial and angular difference descriptors are computed as:

$$\begin{aligned} \underline{\Delta}^{\text{Rad}} &= [sort(\Delta^{\text{Rad}}_{1,0}, ..., \Delta^{\text{Rad}}_{1,p_1-1}),\ ...\ sort(\Delta^{\text{Rad}}_{a,0}, ..., \Delta^{\text{Rad}}_{a,p_a-1})]^T \\ \underline{\Delta}^{\text{Ang}} &= [sort(\Delta^{\text{Ang}}_{1,0}, ..., \Delta^{\text{Ang}}_{1,p_1-1}),\ ...\ sort(\Delta^{\text{Ang}}_{a,0}, ..., \Delta^{\text{Ang}}_{a,p_a-1})]^T \end{aligned} \tag{8}$$

where

$$\begin{aligned} \Delta^{\text{Rad}}_{r,i} &= x^{\delta_r}_{r,i} - x^{\delta_r}_{r-1,i}, \quad \delta_r = 2\pi/p_r \\ \Delta^{\text{Ang}}_{r,i} &= x_{r,i} - x_{r,i-1} \end{aligned} \tag{9}$$

Figure 5 plots the classification results of CS and sorted CS on $\mathcal{D}^{\text{UIUC}}$. The results show that all sorted CS classifiers perform significantly better than the CS classifier, where sorted radial differences performed the best, and the sorted interpolated-circular the best among the pixel-value methods. In general, the performance increases with an increasing number of textons used per class.

Table 4. Experimental results for $\mathcal{D}^{\text{BFull}}$ and $\mathcal{D}^{\text{UIUC}}$: all results for our proposed approach are obtained by number of textons used per class $K = 10$ for $\mathcal{D}^{\text{BFull}}$ and $K = 40$ for $\mathcal{D}^{\text{UIUC}}$, except SCS Radial-Diff (Best), which are the best obtained by varying K up to 40 for $\mathcal{D}^{\text{BFull}}$ and K up to 80 for $\mathcal{D}^{\text{UIUC}}$. Results of Lazebnik et al. and Zhang et al. are quoted directly from [5] and [6].

	$\mathcal{D}^{\text{BFull}}$				$\mathcal{D}^{\text{UIUC}}$		
	Patch Size				Patch Size		
Method	5×5	9×9	11×11	15×15	5×5	9×9	13×13
Dimensionality	10	30	40	60	10	30	50
CS	89.1%	90.8%	89.2%	88.4%	79.6%	77.5%	76.3%
SCS Global	83.6%	81.9%	82.0%	79.8%	90.3%	90.4%	89.1%
SCS Square	84.0%	87.2%	85.8%	86.8%	90.8%	91.8%	91.2%
SCS Circular	86.7%	88.4%	87.4%	85.7%	90.8%	93.3%	92.1%
SCS Radial-Diff	93.1%	94.7%	95.1%	94.7%	91.4%	94.3%	95.4%
SCS Radial-Diff (Best)	94.7%	**96.2**%	95.8%	95.5%	91.5%	95.2%	**96.3**%
SCS Angular-Diff	88.9%	89.8%	92.4%	90.5%	77.1%	84.2%	86.5%
Scale	2	3	4	5	2	3	5
LBP	87.5%	88.9%	89.7%	89.9%	75.6%	81.5%	86.1%
Lazebnik Best [5]	88.2%				96.1%		
Best from Zhang [6]	95.9%				**98.7**%		

5 Experimental Evaluation

5.1 Image Data and Experimental setup

Since near-perfect overall performance has been shown in Section 3 for the CUReT database, for our comprehensive experimental evaluation we have used another three datasets, summarized in Table 1, derived from the three most commonly used texture sources: the UIUC database [5], the Brodatz database [20] and the KTH-TIPS [21] database.

The **UIUC** dataset $\mathcal{D}^{\text{UIUC}}$ [5] has been designed to require local invariance. Textures are acquired under significant scale and viewpoint changes, and uncontrolled illumination conditions. Furthermore, the database includes textures with varying local affine deformations, and nonrigid deformation of the imaged texture surfaces. This makes the database very challenging for classification. For our classification experiments on UIUC, we replicate as closely as possible the experiments described by Lazebnik et al [5].

The Brodatz database [20] is perhaps the best known benchmark for evaluating texture classification algorithms. For the **BrodatzFull** dataset $\mathcal{D}^{\text{BFull}}$, we keep all 111 classes. To the best of our knowledge, there are relatively few publications actually reporting classification results on the whole Brodatz database. Performing classification on the entire database is challenging due to the relatively large number of texture classes, the small number of examples for each

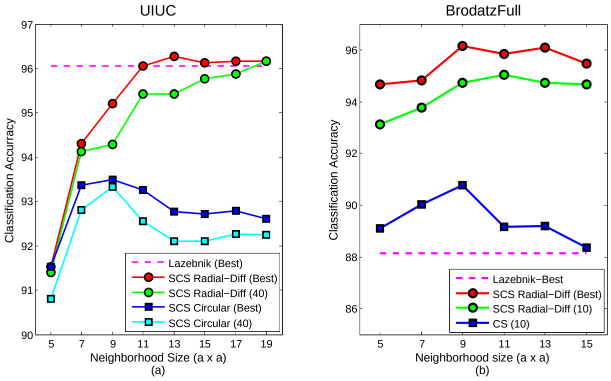

Fig. 6. Classification accuracy as a function of patch size: (a) Results on $\mathcal{D}^{\text{UIUC}}$. The SCS Radial-Diff Best curve shows the best results obtained by varying K up to 80. Similarly, SCS Circular Best curve is obtained by varying K up to 60. (b) Results on $\mathcal{D}^{\text{BFull}}$. The SCS Radial-Diff Best curve is obtained in the same way as in (a), but with K up to 40 being tried. In both (a) and (b), the results for Lazebnik Best are the highest classification accuracies, directly quoted from the original paper [5].

class, and the lack of intra-class variation. In order to obtain results comparable with Lazebnik et al.[5] and Zhang et al. [6], we used the same dataset as them, dividing each texture image into nine non-overlapping subimages.

The **KTH-TIPS** dataset $\mathcal{D}^{\text{KT}}$ [21] contains 10 texture classes with each class having 81 images, captured at 9 lighting and rotation setups and 9 different scales.

Implementation details: Each sample is intensity normalized to have zero mean and unit standard deviation. All results are reported over tens of random partitions of training and testing sets. Each extracted CS vector is normalized via Weber's law. Histograms/χ^2 and nearest neighbor (NN) classifier are used. Half of the samples per class are randomly selected for training and the remaining half for testing, except for $\mathcal{D}^{\text{BFull}}$, where three samples are randomly selected as training and the remaining six as testing.

5.2 Experimental Results and Performance Analysis

The results for datasets $\mathcal{D}^{\text{BFull}}$ and $\mathcal{D}^{\text{UIUC}}$, the same datasets used by Lazebnik et al. [5] and Zhang et al. [6], are shown in Table 4 and Figure 6. As expected, among the CS methods, SCS Radial-Diff performs best on both of these two datasets. The radial-difference method significantly outperforms LBP, outperforms the method of Lazebnik et al. [5] for both BrodatzFull and UIUC, and outperforms the method of Zhang et al. [6] on BrodatzFull. This latter result should be interpreted in light of the fact that Zhang et al. use scale invariant and affine invariant channels and a more advanced classifer (EMD/SVM), which is important for $\mathcal{D}^{\text{UIUC}}$ where some textures have significant scale changes and affine variations.

Table 5. Comparison results for $\mathcal{D}^{\mathrm{B90}}$: all results for our proposed approach are obtained by $K = 10$ except SCS Radial-Diff (Best), which are the best obtained by varying $K = 10, 20, 30, 40$.

	Patch Size			
Method	5×5	7×7	9×9	11×11
Dim	10	25	30	40
CS	95.4%	95.0%	94.5%	93.6%
SCS Radial-Diff	97.2%	**97.6**%	97.5%	97.4%
SCS Radial-Diff (Best)	97.5%	**98.2**%	98.1%	97.6%
Dim	25	49	81	121
Patch	94.7%	94.8%	94.0%	93.2%
Scale	2	3	4	5
LBP^{riu2}	93.6%	94.8%	95.1%	94.7%

Table 6. Comparisons of classification results of the Basic CS and the SCS Radial-Diff on $\mathcal{D}^{\mathrm{CUReT}}$ with $K = 10$.

Patch Size	CS	SCS Radial-Diff
7×7	96.80%	96.33%
9×9	96.91%	96.61%

Motivated by the strong performance of SCS Radial-Diff, we return to the comparison of Section 3. Table 5 shows the classification results on $\mathcal{D}^{\mathrm{B90}}$ with homogeneous or near homogeneous textures, in comparison to the state-of-the-art. We can see that near perfect performance can be achieved by the proposed SCS Radial-Diff approach. In terms of $\mathcal{D}^{\mathrm{CUReT}}$, [4] showed that the incorporating of rotation invariance is not so helpful, nevertheless the performance penalty in Table 6 for incorporating rotation invariance is very modest.

Finally, Table 7 lists the results for the KTH-TIPS database $\mathcal{D}^{\mathrm{KT}}$. Note that $\mathcal{D}^{\mathrm{KT}}$ has controlled imaging and a small number of texture classes. Textures in this dataset have no obvious rotation, though they do have controlled scale variations. From Table 7, we can see that SCS Radial-Diff again performs the best, outperforming all methods in the extensive comparative survey of Zhang et al. [6].

6 Conclusions

In this paper, we have described a classification method based on representing textures as a small set of compressed sensing measurements of local texture patches. We have shown that CS measurements of local patches can be effectively used in texture classification. The proposed method has been shown to match or surpass the state-of-the-art in texture classification, but with significant reductions in time and storage complexity. The main contributions of our paper are

Table 7. Experimental results for $\mathcal{D}^{KT}$: all results for our proposed approach are obtained by $K = 20$ except SCS Radial-Diff (Best), which are the best obtained by varying $K = 10, 20, 30, 40$. Results of Zhang et al are quoted directly from [6].

	Patch Size				
Method	7×7	9×9	11×11	13×13	15×15
CS	95.6%	95.2%	94.6%	94.1%	94.2%
SCS Global	94.0%	93.3%	93.7%	92.6%	92.0%
SCS Square	96.1%	94.7%	95.4%	95.5%	95.6%
SCS Circular	95.1%	95.4%	95.1%	94.7%	94.0%
SCS Radial-Diff	96.0%	96.6%	96.8%	96.9%	**97.1**%
SCS Radial-Diff (Best)	96.5%	97.2%	97.3%	97.4%	**97.4**%
SCS Angular-Diff	91.0%	91.1%	90.9%	N/A	N/A
Best from Zhang [6]	96.1%				

the proposed CS classifier, and the novel sorting scheme for rotation invariant texture classification. Among the sorted descriptors evaluated in this paper, the sorted radial difference descriptor is simple, yet it yields excellent performance across all databases.

The proposed CS approach outperform all known classifiers on the CUReT databases, and the proposed SCS Radial Difference approach outperforms all known classifiers on the Brodatz90, BrodatzFull, and KTH-TIPS databases.

References

1. Leung T. and Malik J.: Representing and Recognizing the Visual Appearance of Materials Using Three-Dimensional Textons. International Journal of Computer Vision 43 (2001) 29-44
2. Cula O. G. and Dana K. J.: 3D Texture Recognition Using Bidirectional Feature Histograms. International Journal of Computer Vision 59(2004) 33-60
3. Varma M. and Zisserman A.: A Statistical Approach to Texture Classification from Single Images. International Journal of Computer Vision 62 (2005) 61-81
4. Varma M. and Zisserman A.: A Statistical Approach to Material Classification Using Image Patches. IEEE Trans. Pattern Analysis and Machine Intelligence 31 (2009) 2032-2047
5. Lazebnik S., Schmid C., and Ponce J.: A Sparse Texture Representation Using Local Affine Regions. IEEE Trans. Pattern Analysis and Machine Intelligence 27 (2005) 1265-1278
6. Zhang J., Marszalek M., Lazebnik S. and Schmid C.: Local Features and Kernels for Classification of Texture and Object Categories: A Comprehensive Study. International Journal of Computer Vision 73 (2007) 213-238
7. Candès E. and Tao T.: Near-Optimal Signal Recovery from Random Projections: Universal Encoding Stratigies? IEEE Trans. Information Theory 52 (2006) 5406-5425

8. Donoho D. L.: Compressed Sensing. IEEE Trans. Information Theory 52 (2006) 1289-1306
9. Mellor M., Hong B.-W. and Brady M.: Locally Rotation, Contrast, and Scale Invariant Descriptors for Texture Analysis. IEEE Trans. Pattern Analysis and Machine Intelligence 30 (2008) 52-61
10. Donoho D. and Tanner J.: Observed Universality of Phase Transitions in High-Dimensional Geometry, with Implications for Modern Data Analysis and Signal Processing. Phil. Trans. R. Soc. A 367 (2009) 4273-4293
11. Romberg J.: Imaging via Compressive Sampling. IEEE Signal Processing Magazine 25 (2008) 14-20
12. Wright J., Yang A., Ganesh A., Sastry S. S. and Ma Y.: Robust Face Recognition via Sparse Representation. IEEE Trans. Pattern Analysis and Machine Intelligence 31 (2009) 210-217
13. Candès E. and Tao T.: Decoding by Linear Programming. IEEE Trans. Information Theory 51 (2005) 4203-4215
14. Graf S. and Luschgy H.: *Foundations of Quantization for Probability Distributions.* New York: Springer-Verlag (2000)
15. Lowe D.: Distinctive Image Features from Scale-Invariant Keypoints. International Journal of Computer Vision 60 (2004) 91-110
16. Conners R. W. and Harlow C. A.: A Theoretical Comparison of Texture Algorithms. IEEE Trans. Pattern Analysis and Machine Intelligence 2 (1980) 204-222
17. Unser M.: Sum and Difference Histograms for Texture Classification. IEEE Trans. Pattern Analysis and Machine Intelligence 8 (1986) 118-125
18. Ojala T., Valkealahti K., Oja E. and Pietikäinen M.: Texture Discrimination with Multidimensional Distributions of Signed Gray-Level Differences. Pattern Recognition 34 (2001) 727-739
19. Ojala T., Pietikäinen M., and Mäenpää T.: Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns. IEEE Trans. Pattern Analysis and Machine Intelligence 24 (2002) 971-987
20. Brodatz P.: Textures: A Photographic Album for Artists and Designers. Dover Publications. New York (1966)
21. Hayman E., Caputo B., Fritz M., and Eklundh J.-O.: On the Significance of Real-World Conditions for Material Classification. In: ECCV (2004) 253-266