IDIAP RESEARCH REPORT

Classifying Materials in the Real World

Barbara Caputo [a] Eric Hayman [b]
Mario Fritz [c] Jan-Olof Ekluhnd [d]

IDIAP–RR 07-69

December 2007

submitted for publication

[a] IDIAP Research Institute, Martigny, Switzerland -bcaputo@idiap.ch
[b] Tracab, Stockholm, Sweden -eric.hayman@tracab.com
[c] Department of Computer Science, TU-Darmstadt, Germany -fritz@mis.tu-darmstadt.de
[d] CVAP, Dept. of Numerical Analysis and Computer Science, KTH, Stockholm, Sweden -joe@nada.kth.se

IDIAP Research Institute **www.idiap.ch**
Av. des Prés–Beudin 20 P.O. Box 592 1920 Martigny – Switzerland
Tel: +41 27 721 77 11 Fax: +41 27 721 77 12 Email: info@idiap.ch

IDIAP Research Report 07-69

Classifying Materials in the Real World

Barbara Caputo Eric Hayman Mario Fritz Jan-Olof Ekluhnd

December 2007

submitted for publication

Abstract. Classifying materials from their appearance is challenging. Impressive results have been obtained under varying illumination and pose conditions. Still, the effect of scale variations and the possibility to generalize across different material samples are still largely unexplored. This paper [1] addresses these issues, proposing a pure learning approach based on support vector machines. We study the effect of scale variations first on the artificially scaled CUReT database, showing how performance depends on the amount of scale information available during training. Since the CUReT database contains little scale variation and only one sample per material, we introduce a new database containing ten CUReT materials at different distances, pose and illumination. This database provides scale variations, while allowing to evaluate generalization capabilities: does training on the CUReT database enable recognition of another piece of sandpaper? Our results demonstrate that this is not yet possible, and that material classification is far from being solved in scenarios of practical interest.

[1] A preliminary version of this work was presented in [1]: E. Hayman, B. Caputo, M. Fritz, J.-O. Eklundh. On the significance of real world conditions for material classification. In Proceedings of ECCV04.

Figure 1: Three images of white bread taken from the CUReT database demonstrating the variation of appearance of a 3D texture as the pose and illumination conditions change.

1 Introduction

Recognising materials from their visual textures is a challenging task, which can be useful in several applications. For instance, it may facilitate object recognition and image retrieval, as texture is the most distinctive feature for many objects. Knowledge of the material can also be useful in robotic manipulation tasks, as it helps in adopting an appropriate grasping strategy. These real-world applications call for robust recognition algorithms. Previous work has mostly concentrated on the ability to recognise materials from a variety of poses and with different illumination conditions. This task is particularly challenging when the material has considerable 3-dimensional structure. In this case, cast shadows and highlights can cause the visual appearance to change radically with varying view and illumination conditions. Figure 1 shows an example from the CUReT database [2] which highlights these variations in appearance on the 'white bread' sample. This database has been considered as the benchmark for this research. A popular approach so far has been to design robust feature descriptors and determine the progress in classification performance on the CUReT database. Impressive classification results, up to 99%, have been obtained [3, 4, 5, 6] (we refer the reader to section 2 for a thorough review of the relevant literature).

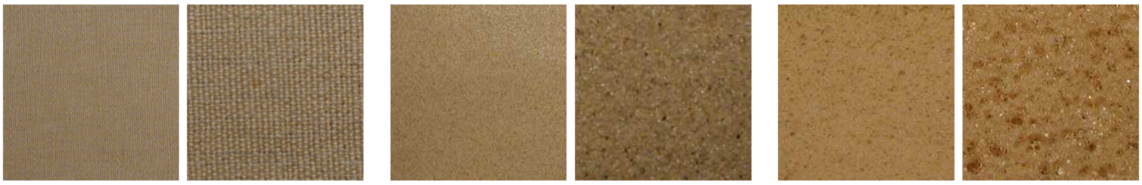
However, other key issues that are essential to recognise materials in real-world conditions have been largely unaddressed:

1. *Robustness to scale changes:* variations in scale affect heavily the visual appearance of a material. It is likely that fine level details will become visible at a closer distance, while at a coarser scale they might not be seen because of the finite resolution of the imaging device. Figure 2 shows how 'cotton', 'sandpaper' and 'sponge', from the KTH-TIPS database, can vary dramatically in their appearance when one consider samples at different scales.

2. *Generalisation across material samples:* in realistic settings it is essential to be able to generalise with respect to different samples of the same material. For instance, the goal could be to recognise *any* wooden table. On the other hand, only recognizing a single, particular piece of wood, supplied during training, is pointless in most practical applications.

In spite of their significance, these two points are still widely ignored.

The overall goal of this work is to bring material recognition algorithms closer to the stage where they will be useful in real-world applications. Our first major objective is to provide robustness to variations in *scale*. We show experimentally that even state-of-the-art algorithms are very sensitive to scaling effects, and classification accuracy rapidly degrades. Our solution is a pure-learning approach, based on the Support Vector Machines (SVM) algorithm [7, 8]. The key features of SVMs are the use of kernels, the absence of local minima, the sparseness of the solution and the capacity control obtained by optimizing the margin [7]. These properties make SVMs an effective choice for material classification. This statement is supported by an extensive experimental evaluation of their performance on the CUReT database, using several kernel types and comparing results with the Varma



(a) Cotton (b) Sandpaper (c) Sponge

Figure 2: The appearance of materials can change dramatically with distance to the camera.

Zisserman algorithm (section 3). Our results [1] are in agreement with those reported in [26], and underline the success of SVMs for this application. We then accommodate variations in scale in the training samples. This is similar to how differing illumination and pose are currently modeled. This approach proves successful when sufficient information on the scale variation is given during training. We compare SVMs with the nearest-neighbour classification scheme adopted by Varma and Zisserman [5]. Experiments show clearly the superiority of SVMs on the Varma-Zissermann approach.

s already alluded to, experiments are conducted on the CUReT image database [2] which captures variations in illumination and pose for 61 different materials, many of which contain significant 3D structure. This database does not, however, contain many scaling effects. It is possible to artificially scale the images, or modify the scales of the filters in the filter bank. In this way it is possible to get some indication of the performance under varying scale. However, we also investigate classification results on pictures of materials present in the CUReT database, imaged in our laboratory at different scales. This constitutes a more realistic test. The goals of these experiments are two-fold. First, they permit a systematic study of scale effects while still providing some variations in pose and illumination. Second, we investigate whether it is possible to recognise materials in this new database given models trained on the CUReT database. This indeed proves a stern test, since both the sample of material, the camera and lighting conditions are different to those used during training. We conduct an extensive set of experiments to assess the new database. We also experimentally study the generalisation capability across different material samples of SVMs and the Varma-Zissermann approach. Our results emphasise the difficulty of the task. They show unequivocally that scale variations and generalisation across samples are today two of the most important obstacles for the use of material classification algorithms in realistic settings.

In summary, the contributions of this paper are:

1. We address the issue of robustness to scale variation, and we propose a learning-based approach, as opposed to the more researched feature-based one. We show that this strategy is effective as long as enough information on the scale changes is provided during training. Experiments comparing our strategy with a similar extension of the Varma-Zisserman algorithm indicate that the SVM strategy is more robust.

2. We address the issue of generalisation across different samples of the same material. To do so, we constructed a new database, designed to complement the CUReT database with scale variations. This database, called KTH-TIPS (Textures under varying Illumination Pose and Scale) is freely available to other researchers via the web [10]. An extensive set of experiments show that the generalisation across different material samples is still an open problem, where neither feature-based nor learning-based approaches are able to obtain satisfactory recognition performances.

The remainder of the paper is organised as follows. Section 2 reviews previous literature in the field of material classification. Particular emphasis is placed on the algorithm of Varma and Zisserman

[1]Our results were state-of-the-art when first presented in [1], but Broadhurst [6] has since demonstrated higer classification rates of 99.2% on the CUReT database.

[5] on which we ourselves build to a large extent. Section 3 discusses the application of Support Vector Machines to this problem, and also presents experiments which demonstrate their superior performance relative to the original approach of [5]. Then, section 4 discusses issues with scale, presents a pure learning approach for tackling the problem, and conducts experiments on the CUReT database. Section 5 introduces the database designed to supplement the CUReT database for experiments with scale. It presents a series of experiments assessing the new corpus and studying how SVM and the Varma-Zisserman algorithm behave when trained and tested on different samples of the same material. Conclusions are drawn and potential avenues for future research outlined in section 6.

2 Previous work on material classification

n this section we review existing literature on material classification. Relevant work on support vector machines for visual pattern recognition will be reviewed in section 3.

Using texture for recognising fairly broad classes is important within remote sensing. The aim is to distinguish between water, forest, crops and urban areas, or subclasses thereof. For instance, Gabor filters were used in [11], and [12] performed a comparative study of several texture features. Texture analysis has also been used for medical diagnosis. For instance [13] used conventional images for classifying skin lesions.

The vast majority of work on texture recognition [14, 15, 16] has dealt with planar image patches sampled, for instance, from the Brodatz collection [17]. There training and test was typically performed on non-overlapping patches taken from the same images. During the last years researchers have started to address the problems associated with recognising materials in spite of varying pose and illumination. Leung and Malik [3] modeled 3D materials in terms of *texton histograms*. While the notion of textons is familiar from the work of Julesz [18], it was only recently defined for greyscale images as a cluster center in a feature space formed by the output of a filter bank. Then, given a vocabulary of textons, the filter output of each pixel is assigned to its nearest texton, and a histogram of textons is formed over an extended image patch. This procedure was described for 2D textures in [19] and for 3D textures in [3] by stacking geometrically registered images from the training set on top of each other. Recognition is achieved by gathering multiple images of the material from the same viewpoints and illuminations and performing the geometric registration. Then, the texton histograms are computed. Classification is achieved using a nearest-neighbour scheme based on the χ^2 distance between model and query histograms.

Cula and Dana [4] adapted the method of Leung and Malik to form a faster, simpler and more accurate classifier. They realised that the 3D registration was not necessary. Thus they described a material by multiple histograms of 2D textons, where each histogram is obtained from a single image in the training set. This also implies that recognition is possible from a single query image.

Varma and Zisserman [5] argued strongly for a rotationally invariant filter bank. First, two images of the same material differing only by an image-plane rotation should be equivalent. Second, removing the orientation information in the filter bank considerably reduced the size of the feature vector. Third, it led to a more compact texton vocabulary since it was no longer necessary for one texton to be a rotated version of another. Rotational invariance was achieved by storing only the maximum response over orientation of a given type of filter at a given scale. As Fig. 3 indicates, the filter bank contains 38 filters, (one Gaussian, one Laplacian, and first and second Gaussian derivatives each at three scales and six orientations), but only 8 responses are stored, yielding the so-called MR8 (Maximum Response 8) descriptor. This reduced storage requirements and also gave a huge speed-up since clustering was performed in a space of much lower dimension. The use of this descriptor reduced storage requirements and computation times. It also led to an improvement in recognition rate. In their experiments [5] they use 92 of the 205 images in the CUReT database, removing samples at severely slanted poses. Splitting these 92 images of each material equally into 46 images for training and 46 images for the test set, they obtain an impressive classification accuracy of up to 97.43% [9]. This is therefore the system that we will be using as a reference in our own experiments.

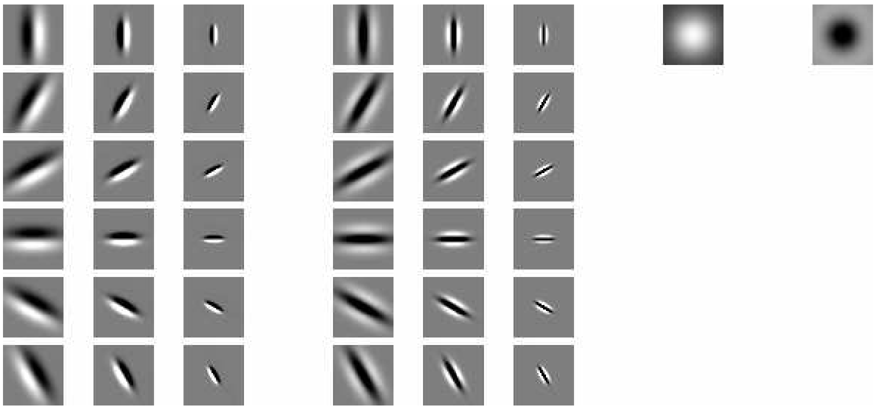


Figure 3: Following [5] we use a filter bank consisting of edge and bar filters (first and second Gaussian derivatives) at 3 scales and 6 orientations, and also a Gaussian and Laplacian. Only the maximum response is stored for each orientation, yielding the 8-dimensional MR8 descriptor.

Many different descriptors have been proposed for texture discrimination. Filter banks are indeed very popular [14, 3, 4, 5, 20], and comparative studies [14] have shown them to perform very well. Furthermore there is evidence that biological systems process visual stimuli using filters resembling those in Fig. 3. However, non-filter descriptors have recently been regaining popularity [21, 9, 22, 23]. [9] presents state-of-the-art results on the CUReT database using a Markov Random Field (MRF) model. [22] and [23] are noteworthy in that they tackle texture recognition in the presence of scale variations. Määnpää and Pietikäinen [22] extend the Local Binary Pattern approach [15] to multiple image resolutions and obtain near-perfect results on a test set from the Outex database. However, this database does not contain any variations in pose or illumination, and the variation in scale is rather small (100dpi images in the training set and 120dpi images in the test set). Lazebnik *et al.* [23] considers simultaneous segmentation and classification of textures under varying scale. Interest points are detected, normalised for scale [24], skew and orientation, and intensity domain spin images computed as descriptors. Each interest point is assigned to a texture class before a relaxation scheme is used to smooth the response. It remains to be seen, however, whether this scheme can handle large variations in illumination, and the number of classes in their experiments is rather small. Scale-invariant recognition using Gabor filters on Brodatz textures was considered by Manthalkar *et al.* [25]. [6] proposed a parametric approach for estimating the likelihood of homogeneously textured images. This paper reported the state-of-the-art recognition rate on the CUReT database.

A thorough comparison of descriptors and classifiers was given in [26] for both texture and object recognition. One contribution of that paper was to demonstrate the suitability of local detectors (scale, rotation, and affine invariant Harris and Laplace) and descriptors (SIFT [27], spin images [28] and RIFT [29]) for these tasks. It is interesting to note the very high performance of our method in their evaluation. On the CUReT database (Fig. 15 in [26]) our method was best by a significant margin, giving an error rate roughly half of all other techniques, including those based on local descriptors. With KTH-TIPS (Fig. 14 in [26]) our approach performed only marginally poorer than the method of [26]. It is important to note that the authors do not address the robustness to scale issue. Also, the generalization issue is not addresses, as all the experiments are performed on separated databases [26] and not also across, as we do.

3 Using Support Vector Machines for material classification

upport Vector Machines are state-of-the-art large margin classifiers. They have become increasingly popular for classification of visual patterns, during the last years. This section shows the effectiveness of SVMs for material classification. We show with a set of experiments (section 3.3) that using SVMs leads to a gain in performance compared to the method proposed by Varma and Zisserman [5, 9]. In a first series of experiments, we compared SVMs with the RBF Gaussian kernel (a widely popular choice [30, 31, 16]) with the original VZ algorithm. We also compared SVMs with an extension of the VZ method, obtained substituting several k-nearest neighbour algorithms to the nearest neighbour which was proposed by the authors.

A key component for the performance of SVMs is the choice of the kernel function. The kernel determines the Hilbert space where the classification is performed. Thus, it corresponds to the choice of a family of similarity measures. There is awareness of the importance of the kernel function for SVMs' application to visual tasks. This has led to the introduction of several new kernels, especially designed for specific visual applications and/or visual features [32, 33, 34]. This gives users the possibility to tailor the algorithm for a specific task. At the same time, it might lead to wrong choices for the kernel function. This raises the question of the robustness of SVMs with respect to this parameter. Thus, the second contribution of this section is a series of experiments where we evaluate how SVM's performance is affected by the designer's choice of the kernel type (section 3.4).

In the rest of the section we review existing literature on SVMs for visual recognition (section 3.1). We put a particular emphasis on previous work on texture and material classification. We then provide a brief review of the theory behind this type of algorithm (section 3.2). The experimental sections 3.3 and 3.4 finally show the improvements that can be achieved with SVMs.

3.1 Previous work on SVM-based visual recognition

Support Vector Machines and kernel methods are widely used approaches for visual pattern classification, particularly for object recognition and categorisation. Pontil and Verri [35] demonstrated the robustness of SVMs to noise, bias in the registration and moderate amounts of occlusion. Roobaert *et al.* [36] examined their generalisation capabilities when trained on only a few views per object. Barla *et al.* [32] proposed a new class of kernel inspired by similarity measures successful in vision applications. Other notable work includes [37, 38, 31]. Wallraven et al [33] introduced a kernel able to compute similarity measures with sets of features. It averages over the similarities of the best matching feature, found for each feature member within the other set. Grauman and Darrel [34] proposed to map unordered feature sets to multi-resolution histograms and computed a weighted histogram intersection in this space. This idea was then extended to spatial pyramid matching for recognition of natural scene categories [39].

Several authors explored the effectiveness of SVMs for texture classification. Kim et al [16] proposed a purely learning approach, where the feature extraction step was incorporated within the classifier's architecture. This approach leads to learning features from the training data. They are shown to correspond to several conventional feature extraction methods, commonly used for feature classification. The authors apply binary SVMs to multi texture classification, and a neural network is used for the final decision step from all the SVMs' outputs. The approach was tested on the Brodatz album and on the MIT Vision Texture images. This method was then applied to text detection in images, combined with a continuously adaptive mean shift algorithm [40]. Li et al [41] proposed instead to use translation-invariant features generated from the discrete wavelet frame transform. Classification is performed through an SVM-based voting scheme. For a fixed kernel type (RGB in this case), several values of the kernel parameters are chosen, and the corresponding classifiers are trained. The final decision is taken via a voting algorithm. The method is tested on the Brodatz texture album and benchmarked against a Bayes classifier and a learning vector quantisation algorithm. Note that all those SVM-based approaches have been used on planar textures. A common limitation of SVM and kernel methods proposed so far, is the heuristic in the choice of the kernel function, and in the choice of the kernel parameters. The performance of the algorithm depends heavily on these choices.

3.2 Support Vector Machines: a review

This section reviews the theory behind the SVM algorithm (we refer the reader to [7, 8] for a more detailed treatment). We start from two-class, linear SVM (section 3.2.1) and then generalise the algorithm to the non-linear case (section 3.2.2). Section 3.2.3 describes the kernel functions used in the paper, and finally section 3.2.4 discusses the possible extensions of the two-class SVM to the multi-class case.

3.2.1 Linear SVM

Consider the problem of separating a set of training data $(\vec{x}_1, y_1), (\vec{x}_2, y_2), \ldots, (\vec{x}_m, y_m)$, where $\vec{x}_i \in \Re^N$ is a feature vector and $y_i \in \{-1, +1\}$ its class label. Assume that the two classes can be separated by a hyperplane $\vec{w} \cdot \vec{x} + b = 0$, and that we have no prior knowledge about the data distribution. Then the optimal hyperplane (the one with the lowest bound on the expected generalisation error) is that which has maximum distance to the closest points in the training set. The optimal values for $\vec{w}$ and b can be found by solving the following constrained minimisation problem:

$$\underset{\vec{w},b}{\text{minimise}} \frac{1}{2}\|\vec{w}\|^2 \quad \text{subject to} \quad y_i(\vec{w} \cdot \vec{x}_i + b) \geq 1, \forall i = 1, \ldots m \tag{1}$$

Introducing Lagrange multipliers $\alpha_i (i = 1, \ldots, m)$ results in a classification function

$$f(x) = \text{sign}\left(\sum_{i=1}^{m} \alpha_i y_i \vec{w} \cdot \vec{x} + b\right), \tag{2}$$

where α_i and b are found by Sequential Minimal Optimization (SMO, [7, 8]). Most of the α_i take the value of zero. Those $\vec{x}_i$ with non-zero α_i are the "support vectors". In cases where the two classes are non-separable, an additional penalty term is introduced that results in Lagrange multipliers $0 \leq \alpha_i \leq C, i = 1, \ldots, m$, where C determines the trade-off between margin maximisation and training error minimisation.

3.2.2 Non-linear SVM

To obtain a non-linear classifier, one maps the data from the input space $\Re^N$ to a high dimensional feature space $\mathcal{H}$ by $\vec{x} \to \Phi(\vec{x}) \in \mathcal{H}$, such that the mapped data points of the two classes are linearly separable in the feature space. Assuming there exists a kernel function K such that $K(\vec{x}, \vec{y}) = \Phi(\vec{x}) \cdot \Phi(\vec{y})$, a non-linear SVM can be constructed by replacing the inner product $\vec{w} \cdot \vec{x}$ by the kernel function $K(\vec{x}, \vec{y})$ in eqn. (2). This corresponds to constructing an optimal separating hyperplane in the feature space. The kernel $K(\vec{x}, \vec{y})$ can be seen as a non-linear generalisation of the Euclidean scalar product. Thus, choosing a kernel type corresponds to the choice of a similarity function for the classifier. Note that the embedding given by the feature mapping $\{x_1, \ldots, x_n\} \to \{\Phi_1(\vec{x}), \ldots \Phi_N(\vec{x})\}$ is non-linear. Hence, in general it defines a possibly contorted manifold, whose dimension is at most that of the input space, where the mapped data lie. The properties of this manifold are related to the properties of the similarity measure induced by the kernel $K(\vec{x}, \vec{y})$. It is possible to study them using tools of differential geometry (we refer the interested reader to [42] for an exhaustive discussion on this topic).

3.2.3 Kernel functions

The impact of the choice of a kernel type on SVMs' performance has been clear since the introduction of the kernel trick for the non-linearization of the algorithm. Between the kernel types which were proposed at first, the Gaussian *Radial Basis Function (RBF) kernel*

$$K(\vec{x}, \vec{y}) = \exp\{-\gamma ||\vec{x} - \vec{y}||^2\}; \tag{3}$$

and the *polynomial kernel*

$$K(\vec{x}, \vec{y}) = (\vec{x} \cdot \vec{y} + c)^d; \tag{4}$$

have been widely used in computer vision. The polynomial kernel has been used mostly for object recognition [35, 36, 32], while the Gaussian RBF kernel has been used also for face detection [37], tracking [31] and many other applications. This kernel can be considered as the most used for non--linear SVM. Since SVMs have started to be used for visual recognition, several researchers have proposed new kernel types and have studied their performances. In [30], Chapelle et al proposed two new types of exponential kernels: the *generalised Gaussian RBF kernel*

$$K(\vec{x}, \vec{y}) = \exp\{-\gamma ||\vec{x}^a - \vec{y}^a||^b\}, a \in \Re+, 0 < b \leq 2 \tag{5}$$

and the χ^2 *kernel*

$$K(\vec{x}, \vec{y}) = \exp\{-\gamma \chi^2(\vec{x}, \vec{y})\}, \qquad \chi^2 = \sum_i \frac{||x_i - y_i||^2}{||x_i + y_i||}. \tag{6}$$

This last kernel has been proved later to be a Mercer kernel [43]. More recently, Barla et al [32] introduced the *Intersection kernel*

$$K(\vec{x}, \vec{y}) = \sum_i min(x_i, y_i) \tag{7}$$

and showed its usefulness for object detection. All these kernels have been proposed and tested for visual recognition tasks using global features. Another lively direction of research for visual kernels addresses the issue of how to use local features as input of an SVM classifier [34, 33]. The vast majority of methods proposed in the literature for material classification use global features. For this reason, in the rest of this paper we will consider only kernel types suitable for global descriptors.

3.2.4 Multi-class SVM

As we are addressing multi-class classification problems, we have to choose a multi-class strategy, that extends the SVM from 2-class to M-class problems. Theoretically sound ways of extending large-margin classifiers to more than 2 classes in a practical manner is still a topic of ongoing research. Thus, we are left with the following two basic and widely used strategies: In a *one-vs-others* approach, M SVMs are trained, each separating a single class from all remaining classes. Although the most popular scheme for extending to multi-class problems (see for instance [7, 38, 30]), there is no bound on its generalisation error, and the training time of the standard method scales linearly with M [7]. In the second strategy, the *pairwise approach*, $M(M-1)/2$ two-class machines are trained. The pairwise classifiers are arranged in trees, where each tree node represents an SVM. Decisions can be made using a bottom-up tree similar to the elimination tree used in tennis tournaments [7], or a Directed Acyclic Graph (DAG, [44]).

3.3 Results: SVM vs VZ

Based on the analysis of the generalisation error for DAG of Platt and others [44], we decided for a pairwise approach with DAG, using the *LibSVM* library [45]. Their study indicates that DAGs in a high dimensional feature space can yield good generalisation performance in the context of large margin classification. The SVM parameter C was fixed at 100 whereas the kernel parameters were obtained during training by cross-validation. More specifically, we divided the training set in 5 random and disjoint train/test splits. The training set contained 4/5 of the original training set, the test set 1/5. γ varied between $[10^{-3}, \ldots, 10^3]$. Once the optimal scale range was identified, we proceeded with a search on finer scale. We considered as the best parameters those giving the best average score on the 5 splits. These parameters were then used to train the SVM on the whole training set. The obtained model was used for classification. This is the strategy we adopted for all the experiments reported in this paper. The histogram features that were used in all our experiments were normalised to unit length.

We compared the SVM classifier with our own implementation of the algorithm of Varma and Zisserman [5], which from now on will be denoted the VZ algorithm. To ensure comparability we use the same 200×200 pixels grey scale image patches as they do. The patches are selected such that only foreground is present. For this comparison, we used the RBF kernel.

Although quantisation of feature domains by clustering has found its way into many computer vision applications, maximum performance is often attained for a large cluster number. Therefore we decided to use a very large texton vocabulary in our first experiment. In this way we can figure out the maximum performance that can be achieved on the CUReT database by our approach. 40 textons were found from each of the 61 materials, giving a total dictionary of $40 \times 61 = 2440$ textons. The 92 images per sample were split equally into training and test sets. Varma and Zisserman [9] previously reported a 97.43% success rate, while our own implementation of their algorithm gave an average of 97.66% with a standard deviation of 0.11% over 10 runs[2]. In contrast, the SVM classifier gave $98.36 \pm 0.10\%$ using an RBF kernel and $98.46 \pm 0.09\%$ using the χ^2-kernel $K = \exp\{-\gamma\chi^2\}$. We implemented this Mercer kernel [43] within *LibSVM*. This performs better even than the very best result obtained in [9] using an MRF model (98.03%).

Another natural extension to the Varma and Zisserman algorithm is to replace the Nearest Neighbour classifier with a k-Nearest Neighbour scheme. The basic idea is to find the class which has most training patterns, out of the k nearest neighbours to the query pattern. This scheme reduces the influence of "uncharacteristic" training points which are distant from all other members within that class. k is typically chosen to be odd. In practice *conflicts* may arise when two or more classes $\{\omega_i\}$ share the maximum number of nearest neighbours. Since it may not be convenient to report multiple solutions, a decision may be forced via a number of techniques:

1. Make a random choice amongst those classes $\{\omega_i\}$ with most nearest neighbours.

2. Increment k until the conflict is resolved. In other words, consider next the $k+1$ nearest neighbour problem. If that too ends in a conflict, attempt the $k+2$ nearest neighbour classifier, and so forth. We note that this method can potentially yield a class $\omega_j \notin \{\omega_i\}$ if e.g. a large number of the next nearest neighbours belong to ω_j

3. Decrement k until the conflict is resolved. In other words, consider next the $k-1$ nearest neighbour problem. If that too ends in a conflict, attempt the $k-2$ nearest neighbour classifier, and so forth. Unlike Method # 2, this method is guaranteed to output a class $\omega_j \in \{\omega_i\}$ which was one of the classes involved in the conflict

4. Similar to Method # 2 above, but now restrict the procedure to only permit a solution $\omega_j \in \{\omega_i\}$ by ignoring the $k+1$ nearest neighbour if its class $\omega_{j'} \notin \{\omega_i\}$.

5. Model-1 NN rule from [47]: For each class ω_i involved in the conflict, compute the average distance $\bar{d}_i$ between the query point and test patterns from that class among the k nearest neighbours. The class with lowest $\bar{d}_i$ is selected.

6. Model-2 NN rule from [47]: For *all* classes represented among the k nearest neighbours, compute the average distance $\bar{d}_i$ between the query point and test patterns from that class among the k nearest neighbours. The class with lowest $\bar{d}_i$ is selected. This approach implies that the method is not merely applied when a query generates a conflict, but for each and every query.

[2]The variability within experiments is due to slightly different texton vocabularies; images are selected at random when generating the dictionary with K-means clustering. The difference of 0.23% between our results and the figure of 97.43% reported in [9] is caused by our use of more truncated filter kernels (41×41 compared to 49×49 [46]) although the scales used to compute the kernels were identical. For a texton to be assigned to a pixel, the entire support region of the filter kernel is required to lie inside the 200×200 image patch. Thus the texton histograms contain more entries when a smaller filter kernel is used. It may be noted that the MRF algorithm of [9] computes descriptors from significantly smaller regions, for instance 7×7.

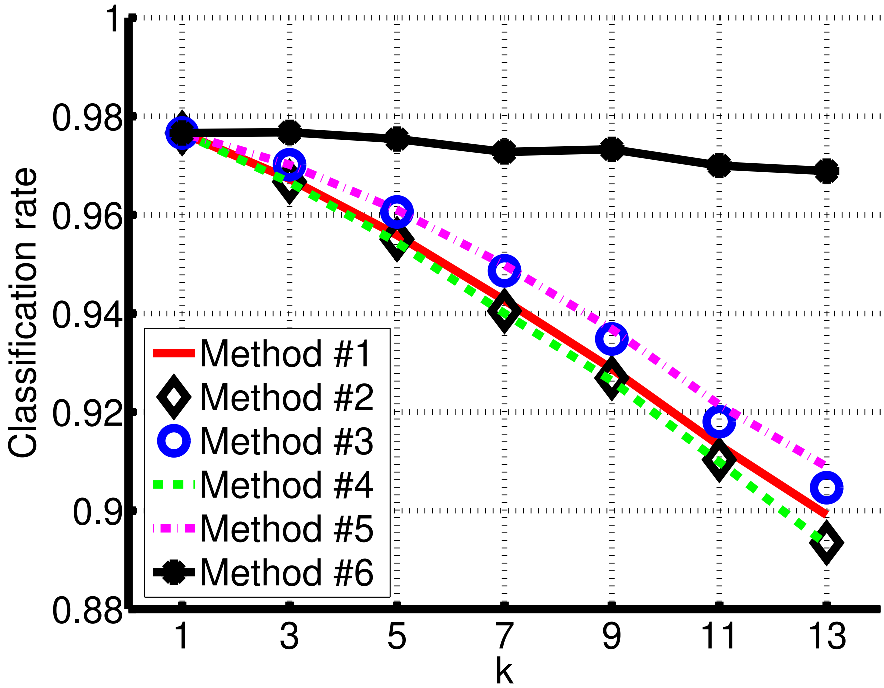


Figure 4: Experiments using k-Nearest Neighbours for various k on the CUReT database. A number of schemes were tested for resolving *conflicts* which arise when two or more classes share the highest number of training patterns among the k Nearest Neighbours. None of the methods achieve better performance for $k > 1$ for this database.

Figure 5: Experiments comparing our SVM scheme with the VZ [5] approach. (a) plots the reliance on the number of views in the training set, (b) the dependency on the size of the texton vocabulary, and (c) the size of the stored model. In (c) the model reduction schemes of [5, 9] were not implemented.

As Fig 4 shows, Method # 6 (Model-2 NN rule from [47]) proved best in our scenario, but no variant yielded an improved recognition rate for any choice of $k > 1$. This is probably due to a relatively sparse sampling of the pose and illumination conditions in the training set.

In addition we examine the dependency on the size of the training set (Fig. 5a) and the texton vocabulary (Fig. 5b). Both plots clearly demonstrate that the SVM classifier reduces the error rate by $30 - 50\%$ in comparison with the method of [5]. In both experiments, textons were found from the 20 materials specified in [3] rather than all 61 materials. In Fig. 5a, 10 textons per material are used, giving a dictionary of $20 \times 10 = 200$ textons. In Fig. 5b, the training set consists of 23 images per material, and the remaining 69 images per material are placed in the test set.

We want to stress that training and test times are modest for the proposed approach. For the SVM training, the computation time varies from about 20 seconds (with a vocabulary of 100 textons, 12 views per material in the training set) up to roughly 50 minutes (for 2440 textons, 46 views per material). Finding γ by cross-validation, if required, typically incurs a further cost of 3–7 times the figures reported above. Recognition time, for a single image, is always below 0.5 seconds.

The size of the model is another factor that determines the applicability of the approach. This is shown in Fig. 5c. SVM reduces the size of the model by 10–20%, and storing the coefficients α_i causes a little overhead. This is significantly less than the reduction by almost 80% obtained using the greedy algorithms described in [5] and [9]. However, the scheme in [5] used the test set for validating the model, which is unreasonable in a recognition task. Meanwhile the method in [9] was extremely

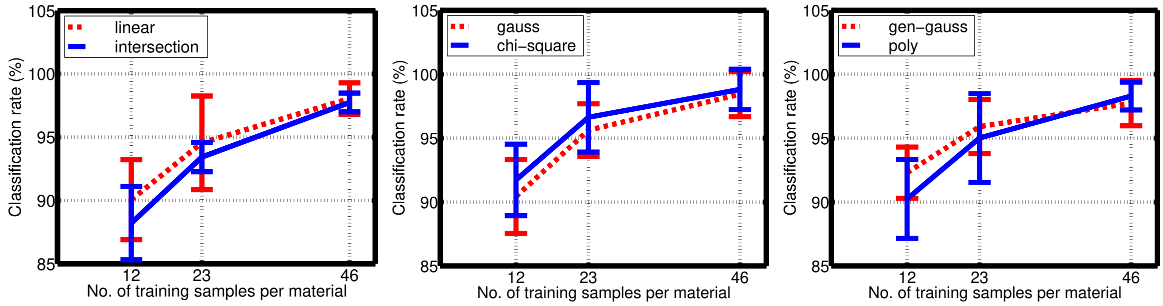


Figure 6: Classification results using different kernel functions, for 12, 23 and 46 training samples per material. Experiments were repeated for 10 different partitions of the training and test set; we report here the average results with standard deviation. Fig 5 (a) shows results for the linear and intersection kernels; fig 5 (b) shows results for the Gaussian and the χ^2 kernels; fig 5 (c) shows results for the generalised Gaussian and the polynomial kernels. All results are statistically equivalent, showing the robustness of our approach.

expensive in training, in fact by a few orders of magnitude [46] in comparison with the more expensive times listed for SVM above. Moreover, their procedure for selecting a validation set from the training set is largely heuristic and at a high risk of over-fitting, in which case the performance on the test set would drop very significantly [46].

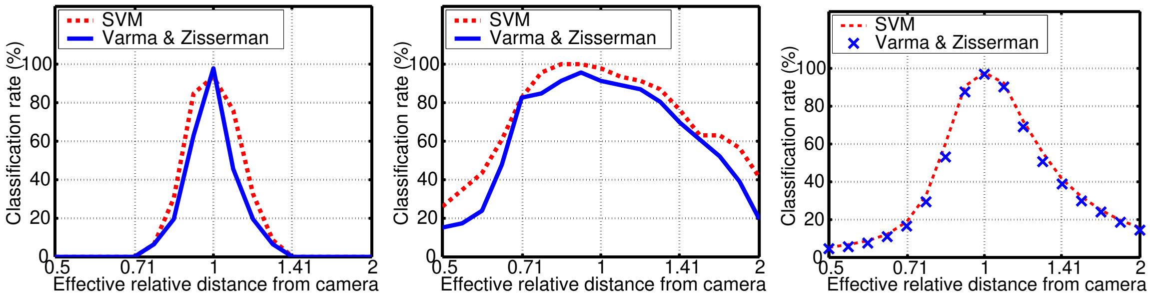
3.4 Results: SVM with Different Kernels

A key ingredient for the success of SVM is the kernel function, that determines the space where data are mapped and classified. As the kernel type is chosen by the user, it represents an element of heuristic in our approach. In order to test the robustness of SVM for material classification with respect to the kernel function, we ran an extensive set of experiments benchmarking 6 different kernel types: linear, polynomial, intersection, Gaussian, generalised Gaussian and χ^2. Experiments were performed with 12, 23 and 46 images per material in the training set, and the remaining images in the test set as described previously (Fig 4 (a)). We used a texton vocabulary of 200 textons, and we repeated each run for 10 different partitions of training and test set, generated randomly. Fig 5 shows the obtained results, averaged and with standard deviation. Considering only the averaged results, we see that the χ^2 kernel obtains the best performance. These results are in agreement with those reported in [26]. Note that in that work the authors compared only two kernel types. The overlaps between the standard deviation bars show that there is not a very significant difference between classification results obtained using different kernels. Thus, we can conclude that SVM's performance is robust to the choice of the kernel function for this application. Motivated by this result, in the rest of the paper we will use the χ^2 kernel.

4 Material classification under variations in scale

Up to this point, we assumed a constant scale in our experiments on the CUReT database [3]. However, for realistic applications this assumption is clearly violated. Scale plays an important role that has to be taken into account. Reviewing the approaches described so far, it seems unlikely that they will perform well in this extended setting. First, the individual filters are tuned to certain frequencies.

[3]Four samples are zoomed in images of other materials. In the experiments reported in this paper, classifying one material as the zoomed in version of that same material is labeled an incorrect match. In practice such confusions are fairly common for those four materials, but this does not have a very large effect on classification rates when averaged over all materials.



(a) Sandpaper (b) Sponge (c) Average

Figure 7: Variations in scale can have a disastrous effect. In this experiment the training set contains images only at the default scale whereas the test set contains images rescaled by amounts up to a factor of two both up and down. For sandpaper (a) the recognition rate drops dramatically, whereas for sponge (b) they are more stable, probably since the salient features are repeated over a wide range of scales. Results averaged over the entire CUReT database are shown in (c).

Zooming in or out on a texture changes the characteristic frequencies of its visual appearance. Second, zooming in on a texture can make visible fine-level details, which could not be recorded at coarser scales due to the finite resolution of the imaging device. Examples are given in Fig. 2. With cotton, for instance, at a coarse scale a vertical line structure is just about visible, whereas at a fine scale the woven grid can be seen clearly, including horizontal fibers.

4.1 A motivational experiment

In order to support our claim, we investigate the scale-dependence of the texton-histogram based schemes experimentally. Our approach is to supplement the CUReT database with artificially scaled versions of its samples. Rather than rescaling the images, which raises various issues with respect to smoothing and aliasing, the *filters* were rescaled. For instance, reducing the size of the image (zooming out) by a factor of two is equivalent to doubling the standard deviations in the filters. We compute such altered samples at eight logarithmically spaced scales for each octave. As we investigate scale variations of two octaves (one octave below and one octave above) we end up with the following scales: $2^{-1}, 2^{-0.875}, 2^{-0.75}, \ldots, 2^{0.75}, 2^{0.875}, 2^{1}$. This results in $2 \times 8 = 16$ scaled images in addition to the unscaled original, giving a total of 17 images. Only the unscaled images are placed in the training set, whereas recognition is attempted at all 17 scales [4]. The 92 images per sample are split evenly into training and test sets, and a texton vocabulary of 400 textons was used. For SVMs, the kernel parameter γ is determined via cross validation, as described in section 3.3.

Fig. 7 illustrates this dependency on scale for two materials. Sandpaper (Fig. 7a), shows almost no robustness to changes in scale, whereas sponge (Fig. 7b) is much more resilient. These effects can be attributed to two main factors. The first concerns *intra-class* properties: materials with a highly regular pattern have a clear characteristic scale, whereas others, such as sponge, exhibit similar features over a range of scales. The feature vector for the former material could be severely mutated, whereas we expect the descriptor of the latter to be more robust to changes in scale. The second factor depends on the *inter-class* variation in the database: the recognition rate depends on the degree of distraction caused by other materials. It is feasible that a material imaged at a certain scale closely resembles another material at the default scale. Fig. 7c shows corresponding plots for an average over all 61 materials in the CUReT database.

[4] We acknowledge that this method is no true replacement for real images since (i) it is not possible to increase the resolution while artificially zooming in, and (ii) the information content is reduced somewhat when artificially zooming out since the size of the 200×200 pixels patch is effectively reduced.

No of Scales	**SVM (%)**			**VZ (%)**		
	46	23	12	46	23	12
9	98.00	94.96	92.12	92.14	89.23	83.65
5	96.37	92.94	90.15	81.19	77.91	71.95
3	83.05	78.82	77.32	58.00	55.69	51.57
1	37.96	36.32	34.18	34.47	33.16	30.90

Table 1: The recognition rate (in %) on the artificially rescaled CUReT database as the richness of the model is varied both with respect to the sampling density in the scale direction and in how many of the original 92 images are incorporated in the training set (per scale). With 3 scales present, the training set includes the original image and also samples at scales one octave up and one octave down. With five scales, half-octave positions are made available during training, and with 9 scales, quarter-octave positions are also used.

4.2 Robustness to scale variations: a pure learning approach

As demonstrated by the previous experiment, robustness to changes in image scale can be crucial for material recognition in the real world. An option to cope with this challenge is to take a machine learning perspective. We can incorporate the additional variation into the training set in order to make it more representative. Thus the training set is extended to cover not just variations in *pose* and *illumination* conditions, but also *scale*. An alternative, left unexplored here, would be to include only images at one scale during training, but then artificially rescale the query image to a number of candidate scales by rescaling the filter bank.

An open question is how densely it is necessary to sample in the scale direction, particularly since the size of the training set has obvious implications for algorithm speed and memory requirements. Clearly there will be some dependence on the bandwidth of the filters, but the amount of inter-class variation will also be of consequence.

This dependence on sampling in the scale dimension was ascertained empirically on the rescaled CUReT database. Our findings are summarised in Tables 1a-b for the SVM and VZ classifiers, with a vocabulary of 400 textons. We see that impoverishing the model in the *scale* dimension has a more severe effect than reducing the size of the training set, with respect to the proportion of the original 92 images. Both SVM and the VZ schemes exhibit such behaviour. A further point worth emphasising is that SVM systematically outperforms the VZ classifier, as was also seen in Section 3. Again, we attempted replacing the Nearest Neighbour classifier in the Varma and Zisserman approach with k-Nearest Neighbour schemes, but without observing any improvement for $k > 1$.

5 The KTH-TIPS database of materials under varying scale

Results presented in the previous section give some indication on how performance is affected by changes in scale. Still, the artificial rescaling cannot be a perfect replacement for real images. In order to address this issue, we created a new database, supplementing the CUReT, that provides variations in *scale* in addition to pose and illumination. A second objective with the database is to evaluate whether models trained on the CUReT database can be used to recognise materials from pictures taken in other settings. This could indeed prove challenging since not only the camera, poses and illuminant differ, but also the actual samples: can *another* sponge be recognised using the CUReT sponge?

In the rest of the section we introduce the database (section 5.1), and we describe a series of experiments assessing the new database, and testing the out-of-database generalisation capabilities of SVMs and the VZ method (section 5.2).

Figure 8: Exemplar images of the materials within the KTH-TIPS database. From left to right, top row: sandpaper, aluminium foil, styrofoam, sponge, corduroy. From left to right, bottom row: linen, cotton, brown bread, orange peel, cracker B.

5.1 The KTH-TIPS database

The KTH-TIPS (Textures under varying Illumination Pose and Scale) database contains ten materials also present in the CUReT database. The objectives with this database are to supplement the CUReT image database in two directions, both of which concern extending material classification algorithms to function in real-world conditions. Thus, the aim of the KTH-TIPS are:

- to provide variations in *scale* as well as variations in pose and illumination. This allows a systematic study of how important unknown viewing distance is to material classification. It also provides data for evaluating algorithms intended to be robust to such variations.

- to provide images of *other samples* of a subset of the CUReT materials, taken under different settings. We wanted to see whether it would be possible to actually classify materials in the real-world, as opposed to recognising exemplars of materials within a single database.

While the CUReT database images 61 materials, the KTH-TIPS database contains images of 10 of those materials: sandpaper (material 06), crumpled aluminium foil (material 15), styrofoam (material 20), sponge (material 21), corduroy (material 42), linen (material 44), cotton (material 46), brown bread (material 48), orange peel (material 55) and cracker B (material 60). Exemplar views for each material are shown in Fig 8. Each of the samples is planar. The orange peel was flattened by placing it inside a CD case. The images were taken with an Olympus C-3030ZOOM digital camera at a resolution of 1280×960 pixels. We used a single light source (a standard desk lamp with a 60W tungsten light bulb).

Materials were imaged at nine different scales spanning two octaves, as illustrated in Fig. 10 for the cracker. The central scale was selected, by visual inspection, to correspond roughly to the scale used in the CUReT database.

At each distance nine images were captured using a combination of three poses (frontal, 22.5° turned left, 22.5° turned right) and three illumination conditions (front, side at roughly 45° and top at roughly 45°), giving a total of $3 \times 3 = 9$ images per scale, and $9 \times 9 = 81$ images per material. The 9 images for a fixed scale are shown in Fig. 9 for the cracker. For each image we selected a 200×200 pixels region to remove the background. The database is freely available on the web [10].

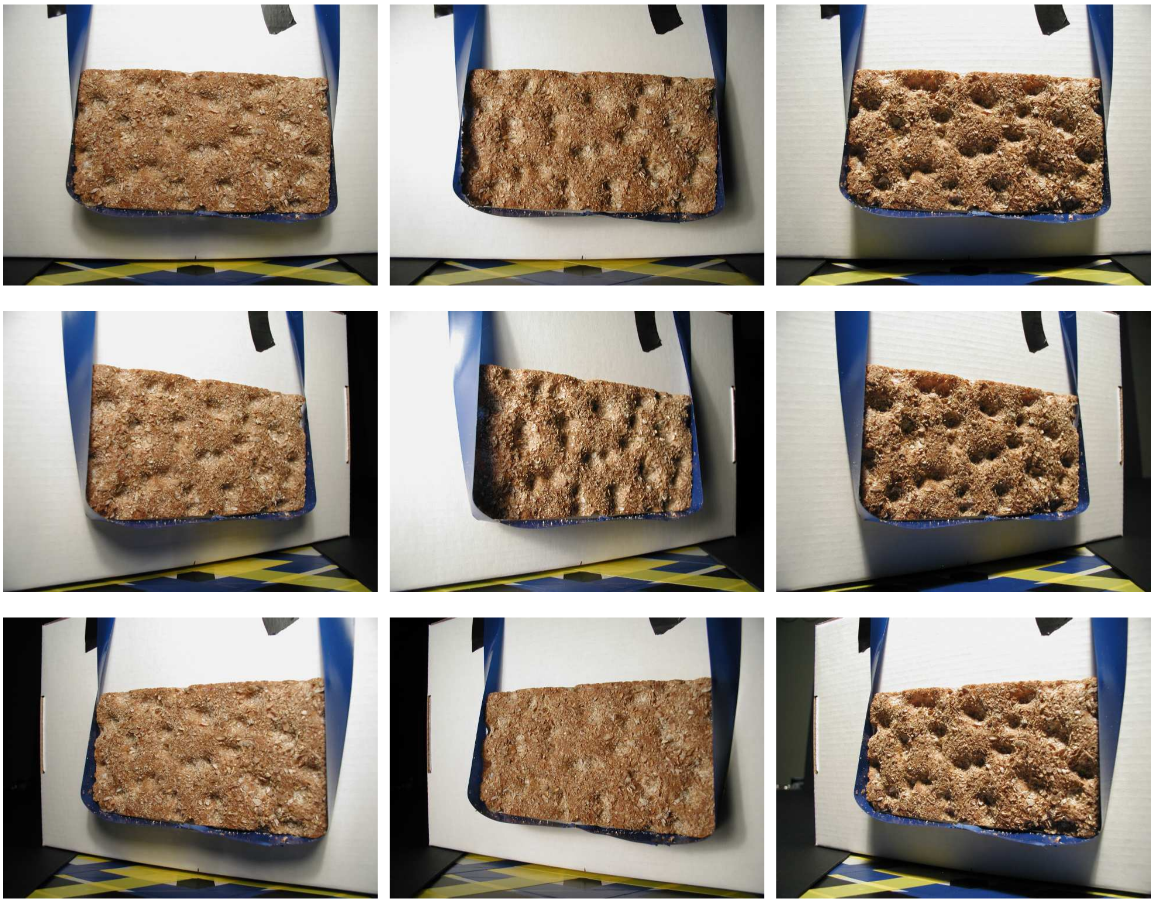


Figure 9: The variations in pose and illumination contained in the new KTH-TIPS (Textures under varying Illumination Pose and Scale) database. Prior to use, images were cropped so only foreground was present.

5.2 Experiments on the new database

We now present three sets of experiments on the KTH-TIPS database, differing in how the model was obtained. For all three sets, we used both the SVM and VZ algorithms. The goal of the first set of experiments was to assess the new database. Thus, we used the KTH-TIPS for training and test. We performed experiments so to study how performance varies when the model is built on views taken at an increasing number of scales. These experiments are reported in section 5.2.1. In the second set of experiments we combined together the CUReT and KTH-TIPS databases, both for training and test. We varied the number of materials in the training model. We analysed how well the two recognition algorithms perform, when they are asked to generalise over different instances of the same materials, acquired under similar, but not identical conditions. We describe these results in section 5.2.2. In the third and last set of experiments we instead kept the two databases separated, and we used them alternatively for training and test. With these experiments we explored the capability to recognise materials across databases, where material instances differ from training to test. These results are reported in section 5.2.3.

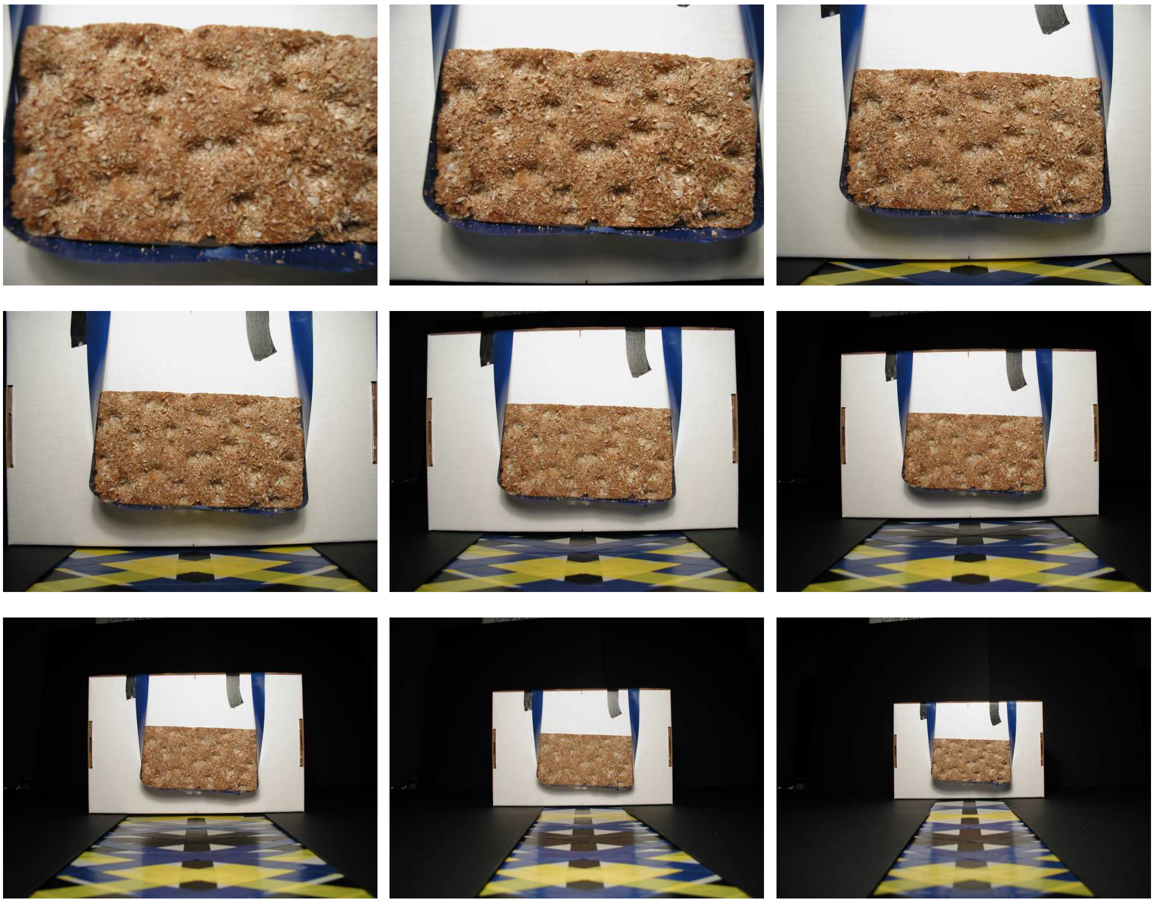


Figure 10: The variations in scale contained in the new KTH-TIPS (Textures under varying Illumination Pose and Scale) database.

5.2.1 First set of experiments: assessing the KTH-TIPS database

We performed a first set of experiments on the KTH-TIPS database, varying the number of images per material in the training set and comparing the SVM and VZ approach. Specifically, we chose three different experimental settings:

- **training on 1 scale:** the training set consisted of 3 views per material, imaged at different poses, taken at the central scale (figure 10, diagonal images). The test set consisted of all views, with different poses and illumination, taken at the remaining scales;
- **training on 3 scales:** the training set was built similarly to what described for the 1 scale experiments, but using instead 3 equally spaced scales. Thus, the test set consisted of all views taken at the 6 remaining scales;
- **training on 5 scales:** the training set for these experiments was built as described above but using 5 equally spaced scales. Consequently, the test set contained all the views from the remaining 4 scales.

We used MR8 features, as for all the previous experiments. 40 textons were found from all the 10 materials, giving a total dictionary of $40 \times 10 = 400$ textons. Table 2 shows the overall recognition rates obtained for these experiments, for both methods, as well as the classification results obtained

Material	**1 scale**		**3 scales**		**5 scales**	
	SVM (%)	VZ (%)	SVM (%)	VZ (%)	SVM (%)	VZ (%)
Sandpaper	75.00	73.61	74.07	68.52	83.33	77.78
Al. Foil	88.89	88.89	96.29	94.44	100	100
Styrofoam	79.17	77.78	96.29	94.44	100	94.44
Sponge	77.78	77.78	92.59	92.59	100	94.44
Corduroy	61.11	54.16	74.07	70.37	86.11	83.33
Linen	41.67	33.33	70.37	61.11	91.67	88.89
Cotton	34.72	31.94	55.56	48.15	77.78	66.67
Brown Bread	54.17	54.17	74.07	64.81	91.67	88.89
Orange Peel	41.67	36.11	57.40	42.59	80.55	69.44
Cracker B.	79.17	76.39	92.59	88.89	94.44	91.67
AVERAGE	**63.33**	**60.42**	**78.33**	**72.59**	**90.56**	**85.56**

Table 2: Recognition results for the KTH-TIPS database. Experiments were conducted using for training (a) the central scale and 3 views per material (1 scale); (b) 3 equally spaced scales and 3 views per scale per material (3 scales); (c) 5 equally spaced scales and 3 views per scale per material (5 scales). The test set consisted of all the views of the remaining scales.

for each material. In all experiments, SVM achieves a better performance than VZ. We see quite predictably that, for both approaches, performance increases as the number of scales in the training set grows. This is in agreement with what observed on the artificially scaled CUReT database (sec 4, Table 1) and confirms the effectiveness of the purely learning approach for robustness to scale changes. The easiest materials to classify are aluminium foil and styrofoam. The most difficult are cotton, linen and orange peel.

5.2.2 Second set of experiments: generalisation within databases

In the second set of experiments we combined together the CUReT and KTH-TIPS databases for training and test. Training was done using 5 equidistant scales in the log-scale dimension, spanning two octaves. For KTH-TIPS materials, at each of these 5 scales, 3 out of 9 images in the KTH-TIPS database were used for training. To preserve the overall size of the training set relative to previous experiments, 43 images from the CUReT database were also inserted in the training set, giving a total of 46 images per scale. As the number of materials differ for the two databases (61 and 10 respectively), we performed two experiments:

- **CUReT10+KTH-TIPS** In this experiment we used for training the 10 materials imaged in the KTH-TIPS, with samples coming from both databases. Training was also performed on samples of those 10 materials from the CUReT and KTH-TIPS databases.
- **CUReT61+KTH-TIPS** In this experiment we used for training the same set described above, plus the remaining 51 materials of the CUReT that act as distractors for the classification task. For those additional 51 materials, we used 46 training images per scale

The texton vocabulary for the MR8 representation here was built as follows: we found 20 textons from 20 of the 61 materials, chosen randomly. Note that the 20 materials could in principle contain views from both the KTH-TIPS and the CUReT database. We thus obtained a texton dictionary of $20 \times 20 = 400$ textons. Results for these experiments using both classification algorithms are summarised in Table 3. A first comment is that generalising within two different databases proved harder than recognising different views of the same materials. By comparing the results reported in Table 3 with those in Table 1, we see that both recognition methods performed better on the CUReT only experiment. This is probably due to two main factors: *(i)* the different scale and

Material	**CUReT10+ TIPS**		**CUReT61+ TIPS**	
	SVM (%)	VZ (%)	SVM (%)	VZ (%)
Sandpaper	91.69	77.78	77.78	66.67
Al. Foil	100	91.69	91.67	88.89
Styrofoam	100	92.59	100	91.67
Sponge	100	100	100	100
Corduroy	91.69	83.33	80.56	80.56
Linen	80.56	48.15	61.11	41.67
Cotton	77.78	61.11	61.11	47.22
Brown Bread	91.69	91.69	77.78	80.56
Orange Peel	100	66.67	100	63.89
Cracker B.	98.91	83.33	91.67	80.56
AVERAGE	**93.23**	**84.44**	**84.17**	**74.17**

Table 3: Recognition results for the second set of experiments. The left column shows results obtained using both databases for training and test, on 10 material classes. The right column shows results obtained using also the remaining 51 materials of the CUReT database as distractors in the training model.

illumination conditions at which the two databases were acquired, and *(ii)* the fact that the images for each material were of two different physical samples. A second comment is that the presence of distractors in the trained model affected the recognition performances for both approaches, leading to a decrease in performance of 10% and more for both algorithms. A possible interpretation is that, as the generalisation task became harder due to different material instances for 10 classes, the between-material vs within-material recognition task became more difficult to solve. The similarity between images from the same materials decreased in some cases, while at the same time it increased the similarity between images from different materials. This made the generalisation task harder for both methods. As a last remark, we notice once again that SVM performs better than VZ for both experiments.

5.2.3 Third set of experiments: generalisation across databases

The third and last set of experiments attempted to recognise the material instances imaged in one database using a model trained on the material instances of the other one. This is a challenging task, requiring high generalisation capabilities. A successful outcome for one or both of the recognition methods under examination would be a strong indicator of the applicability of the approach in the real world. Specifically, we performed three experiments:

- **CUReT61 → TIPS** In this experiment we trained on all the 61 materials of the CUReT database and we tested on KTH-TIPS. 46 out of 92 images per material were placed in the training set. To cope with variations in scale, we used the procedure described in Section 4.2. The model was acquired at multiple scales by adapting the Gaussian derivative filters. For this experiment the training set contained data from 9 scales, equidistantly spaced along the log-scale dimension over two octaves. Test was performed on all the images from the KTH-TIPS database. We used as features MR8 with a texton vocabulary of 400 textons, obtained by computing 20 textons from 20 materials chosen randomly from the original 61.

- **CUReT10 → TIPS** In this experiment we trained on the 10 materials of the CUReT database corresponding to the 10 materials imaged in the KTH-TIPS. In this way the trained model does not contain distractor materials. Training images per material were chosen as described in the experiment above. Test was also performed on the whole KTH-TIPS database. Features for these experiments were MR8, with 40 textons found from the 10 seleced materials. This gave a texton dictionary of 400 textons.

- **TIPS → CUReT** In this experiment we trained on all the KTH-TIPS data and we tested on all the images of the corresponding 10 materials in the CUReT database. In this experiment, we did not use artificially scaled images for testing. The MR8 features consisted of a texton vocabulary of 400 textons, computed from the 10 KTH-TIPS material (40 textons per material).

The recognition rates for all experiments and for all 10 materials are provided in Table 4. While results are, on the whole, well above chance (i.e. 10%), the best results are scarcely above 50% (TIPS → CUReT experiment, both for the SVMs and VZ classifiers), while in the worst case are just above 20% (CUReT61 → TIPS experiments, 20.5% recognition rate for SVM and 23.83 % recognition rate for VZ). Performance is heavily affected by the presence of distracting materials (Table 4, right), showing that the generalisation task is indeed very hard. These results clearly demonstrate that material recognition cannot be performed reliably in the real world merely using only one sample instance to form the model.

Tables 5-8 show the confusion matrices for the TIPS → CUReT and for the CUReT10 → TIPS experiments, for both algorithms. For space reasons we indicated the materials using their CUReT label. We see that linen and cotton (M 44 and 46 respectively) are frequently confused in both experiments, a reasonable behaviour. We note also that aluminium foil (M 15) is overall the easiest material to recognise. Apart from those two common trends, recognition performance and confusions between materials tend to vary between the two sets of experiments. They tend instead to be similar for the two classifiers, for each experimental setting. For instance, orange peel (M 55) acts as a strong distractor for cracker B (M 60) in the TIPS → CUReT experiment (Tables 5-6). This does not happen for the CUReT → TIPS experiment (Tables 7-8), where instead orange peel is a strong distractor for brown bread (M 48). The reasons for these different behaviours might be several. To begin with, despite our efforts, the sample chosen for the materials and the different imaging conditions might have been too different from those used in the CUReT database, thus the generalisation task has proved too hard. Second, the artificial scaling of the CUReT10 training data might be responsible for the asymmetric results of the two sets of experiments. To investigate this point further, we report in Figure 11 results obtained for the CUReT61 → TIPS experiment, at every scale, for the materials sandpaper (Figure 11a), sponge (Figure 11b) and corduroy (Figure 11c). We see that performance on sandpaper is very poor. This could be due to differences between our sample of sandpaper and the CUReT sample of sandpaper, despite our efforts to provide similar samples. It must be stressed anyway that sandpaper was a very difficult material to recognise also in experiments using the CUReT database as the test set.

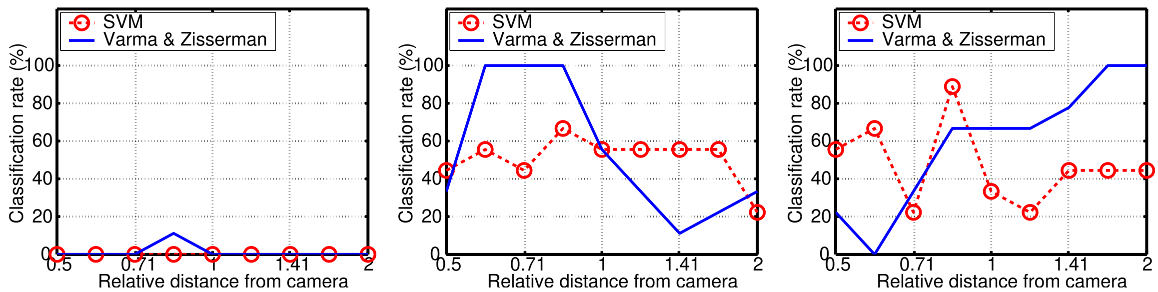
Results are much better for sponge and corduroy, where we achieve recognition results of around 50% . Interestingly, the VZ classifier outperformed SVM in these experiments. The success rate of the VZ approach varies considerably with scale, which might indicate that there is not perfect overlap between the two octaves in scale in the two datasets. Another explanation for a drop-off in performance at fine scales is that the rescaling of the CUReT database cannot improve the resolution. Rescaling the filters does not permit sub-pixel structure to appear. A third reason is that the images closest to the camera were poorly focused in some cases. The SVM classifier provided much more consistent results over varying scales, as could perhaps be expected from the experiment reported in Table 1. However, the recognition rate was consistently fairly low over *all* scales. By supplying a test set too different to the samples provided during training, we are asking the SVM to perform a task for which it was not optimised.

6 Discussion and Conclusions

The goal of this paper was to bring material classification a step closer to realistic scenarios. To achieve this, we addressed two important but often neglected issues: robustness to scale variations, and the

Material	TIPS → CUReT		CUReT10 → TIPS		CUReT61 → TIPS	
	SVM (%)	VZ (%)	SVM (%)	VZ (%)	SVM (%)	VZ (%)
Sandpaper	43.48	44.56	32.09	33.33	0.00	1.23
Al. Foil	100	100	74.07	76.54	11.35	12.35
Styrofoam	59.78	59.78	61.72	64.19	34.72	38.27
Sponge	63.04	65.22	67.90	71.60	50.62	54.32
Corduroy	83.69	84.78	65.43	65.43	46.91	59.26
Linen	27.17	28.26	37.04	41.97	30.41	25.93
Cotton	44.56	45.65	23.46	23.46	11.11	20.99
Brown Bread	0	2.17	37.04	39.51	5.11	7.41
Orange Peel	8.69	9.78	30.86	33.33	11.11	11.11
Cracker B.	98.91	97.83	43.21	45.68	3.70	7.41
AVERAGE	**52.93**	**53.80**	**47.28**	**49.51**	**20.50**	**23.83**

Table 4: Recognition results for the third set of experiments. The left column shows results obtained by training on the KTH-TIPS database and testing on the corresponding 10 materials in the CUReT; the middle column shows results obtained training on 10 materials from the CUReT database, artificially scaled, and testing on the KTH-TIPS database. The right column reports results obtained training on all the CUReT database, with images artificially scaled, and testing on the KTH-TIPS.



(a) Sandpaper (b) Sponge (c) Corduroy

Figure 11: Experiments attempting to recognise images from the new KTH-TIPS database using a model trained on all 61 materials of the CUReT database. The recognition rate is plotted against scale for three materials.

capability to generalise across different instances of the same materials. We showed experimentally that scale plays a crucial role in material classification, and must thus be modeled in some way. We proposed a scale-robust classifier that incorporates scale changes directly into the training set, similar to how varying pose and illumination are usually tackled. We conducted experiments on an artificially rescaled version of the CUReT database, and on a new database designed to supplement the CUReT database by imaging a subset (currently 10 out of 61) of the materials at a range of distances, while still maintaining some variation in pose and illumination. This database represents the second contribution of this paper, and is available to other researchers via the web [10].

However, a more sobering conclusion, and the most important message from this paper, is that such success on the CUReT database does *not* necessarily imply that it is possible to recognise those materials in the real world, even modeling scale. Indeed, experiments performed training on one database and testing on another clearly showed that the generalization capability of both the VZ approach and our SVM approach is not sufficient to achieve reasonable performance, and thus work in real-world conditions.

This problem might be attacked in many ways. A straightforward solution could be to include

	M06	M15	M20	M21	M42	M44	M46	M48	M55	M60
M06	40	0	0	0	0	50	2	0	0	0
M15	0	92	0	0	0	0	0	0	0	0
M20	30	0	55	0	0	6	1	0	0	0
M21	15	12	0	58	0	0	0	6	0	1
M42	0	0	0	1	77	3	11	0	0	0
M44	35	0	8	0	0	25	24	0	0	0
M46	45	0	5	0	1	0	41	0	0	0
M48	32	50	1	0	1	0	0	0	0	8
M55	0	1	0	0	1	0	0	1	8	81
M60	0	1	0	0	0	0	0	0	0	91

Table 5: Confusion matrix for the TIPS → CUReT experiment, SVM classifier. The materials are sandpaper (M06), aluminium foil (M15), styrofoam (M20), sponge (M21), corduroy (M42), linen (M44), cotton (M46), brown bread (M48), orange peel (M55) and cracker B. (M60).

	M06	M15	M20	M21	M42	M44	M46	M48	M55	M60
M06	41	0	0	0	0	49	2	0	0	0
M15	0	92	0	0	0	0	0	0	0	0
M20	31	0	55	0	0	6	0	0	0	0
M21	15	10	0	60	0	0	0	6	0	1
M42	0	0	0	0	78	3	11	0	0	0
M44	35	0	8	0	0	26	23	0	0	0
M46	45	0	5	0	0	0	42	0	0	0
M48	32	50	0	0	0	0	0	2	0	8
M55	0	1	0	0	0	0	0	1	9	81
M60	0	1	0	0	0	1	0	0	0	90

Table 6: Confusion matrix for the TIPS → CUReT experiment, VZ classifier. The materials are sandpaper (M06), aluminium foil (M15), styrofoam (M20), sponge (M21), corduroy (M42), linen (M44), cotton (M46), brown bread (M48), orange peel (M55) and cracker B. (M60).

multiple samples of the same material in a database, but with increased intra-class variability, the risk of inter-class confusion might increase too. This will probably call for sophisticated machine learning techniques, and for class-specific feature selection. Furthermore, the risk of inter-class confusion depends on the number of classes in the database. Thus, keeping this number low (e.g. in production line applications) should make it feasible to separate the classes, but with a large number it might only be possible to classify into broader *groups* of materials, that can then be arranged in hierarchical structures. The performance will again depend on scale since most materials appear more homogeneous with increased imaging distance.

While the pure-learning approach to scale variation proved effective for recognising views of the same material instance, imaged under different conditions, this solution might become computationally unfeasible when several material classes, and several samples per material, are considered. A possible solution might be to select scale as a preprocessing step; interesting work in that direction has been proposed in [24]. Although it might still be necessary to store models at multiple characteristic scales, this number should still be smaller than with the pure-learning approach. This would reduce storage requirements, and also the recognition time.

	M06	M15	M20	M21	M42	M44	M46	M48	M55	M60
M06	26	3	0	0	0	19	18	14	1	0
M15	2	60	0	3	0	0	0	10	6	0
M20	1	0	50	0	0	11	10	4	4	1
M21	4	1	1	55	1	0	0	13	5	1
M42	0	2	0	3	53	1	4	10	8	0
M44	25	3	3	3	0	30	10	5	2	0
M46	3	3	0	4	23	23	19	2	1	3
M48	1	5	0	30	0	0	2	30	12	1
M55	1	14	0	13	0	2	2	20	25	4
M60	0	2	0	25	0	0	0	6	13	35

Table 7: Confusion matrix for the CUReT10 $\rightarrow$ TIPS experiment, SVM classifier. The materials are sandpaper (M06), aluminium foil (M15), styrofoam (M20), sponge (M21), corduroy (M42), linen (M44), cotton (M46), brown bread (M48), orange peel (M55) and cracker B. (M60).

	M06	M15	M20	M21	M42	M44	M46	M48	M55	M60
M06	27	2	0	0	0	19	18	14	1	0
M15	0	62	0	0	0	0	0	13	6	0
M20	1	0	52	0	0	9	12	2	4	1
M21	4	0	0	58	0	0	0	13	6	0
M42	1	1	0	0	53	1	7	8	10	0
M44	31	0	3	0	0	34	9	4	0	0
M46	3	0	0	1	26	26	19	1	2	3
M48	0	6	0	30	0	0	0	32	13	0
M55	0	15	0	13	0	0	2	22	27	2
M60	0	0	0	25	0	0	0	6	13	37

Table 8: Confusion matrix for the CUReT10 $\rightarrow$ TIPS experiment, VZ classifier. The materials are sandpaper (M06), aluminium foil (M15), styrofoam (M20), sponge (M21), corduroy (M42), linen (M44), cotton (M46), brown bread (M48), orange peel (M55) and cracker B. (M60).

Acknowledgments

The authors are very grateful to Manik Varma and Andrew Zisserman for discussions regarding their algorithms, and for providing appropriately cropped images from the CUReT database. BC, EH and MF conducted most of this work while at CVAP, NADA, KTH.

This work was funded by the EU-IST projects IST-2000-29688 *Insight2+* (EH, MF), IST-2000-29375 *CogVis* and IST-FP6-0027787 *DIRAC* (BC), IST-FP6-004250-IP *CoSy* (MF) and the *VISCOS* project funded by the Swedish Foundation for Strategic Research (EH).

References

[1] E. Hayman, B. Caputo, M. J. Fritz, J.-O. Eklundh, On the significance of real world conditions for material classification, in Proc ECCV, Lecture Notes in Computer Science, vol. 4 , Springer, Prague, 2004, 253-266.

[2] K.J. Dana, B. van Ginneken, S.K. Nayar, and J.J. Koenderink, Reflectance and texture of real-world surfaces, ACM Transactions on Graphics, 18(1), 1999, 1–34.

[3] T. Leung and J. Malik, Representing and recognizing the visual appearance of materials using three-dimensional textons, International Journal of Ccmputer Vision, 43(1), 2001, 29–44.

[4] O.G. Cula and K.J. Dana, Compact representation of bidirectional texture functions, in Proc. CVPR, Kauai, Hawaii, vol I, 1041–1047, 2001.

[5] M. Varma and A. Zisserman, Classifying images of materials: Achieving viewpoint and illumination independence, in Proc. ECCV, Copenhagen, vol III, 255 ff, 2002.

[6] R. E. Broadhurst, Statistical estiation of histogram variation for texture classification, In Texture 2005: Proceedings of the 4th International Workshop on Texture Analysis and Synthesis, pp 25-30, 2005.

[7] N. Cristianini and J. S. Taylor. *An introduction to support vector machines and other kernel-based learning methods*. Cambridge University Press, 2000.

[8] V. Vapnik. *Statistical learning theory*. Wiley and Son, New York, 1998.

[9] M. Varma and A. Zisserman. Texture classification: are filter banks necessary? in Proc. CVPR, Madison, Wisconsin, vol II: 691–698, 2003.

[10] M. Fritz, E. Hayman, B. Caputo, and J.-O. Eklundh. The KTH-TIPS database. Available at `http://www.nada.kth.se/cvap/databases/kth-tips`.

[11] M. Shi and G. Healey, Hyperspectral texture recognition using a multiscale opponent representation, IEEE Trans. Geoscience and Remote Sensing, 41(5), 2003, 1090-1095.

[12] L. Ruiz, A. Fdez-Sarra, and J. Recio, Texture feature extraction for classification of remote sensing data using wavelet decomposition: A comparative study, 20th ISPRS Congress, 2004.

[13] H. Ganster, P. Pinz, R. Rohrer, E. Wildling, M. Binder, H. Kittler, Automated melanoma recognition, IEEE Trans on Medical Imaging, 20(3), 2001, 233-239.

[14] B.S. Manjunath and W.Y. Ma, Texture features for browsing and retrieval of image data, IEEE Trans on Pattern Aanalysis and Machine Intelligence, 18(8), 1996, 837–842.

[15] T. Ojala, M. Pietikäinen, and D. Harwood, A comparative study of texture measures with classification based on feature distributions, Pattern Recognition, 29(1), 1996, 51–59.

[16] K.I. Kim, K. Jung, S.H. Park, and H.J. Kim, Support vector machines for texture classification, IEEE Trans on Pattern Aanalysis and Machine Intelligence, 24(11), 2002, 1542–1550.

[17] P. Brodatz. *Textures*. Dover, 1966.

[18] B. Julesz and R. Bergen, Textons, the elements of texture perception, and their interactions, Nature, 290(1981), 91–97.

[19] J. Malik, S. Belongie, J. Shi, and T. Leung, Textons, contours and regions: Cue integration in image segmentation, in Proc. ICCV, Kerkyra, Greece, pages 918–925, 1999.

[20] A. Penirschke, M.J. Chantler, and M. Petrou. Illuminant rotation invariant classification of 3D surface textures using Lissajous's ellipses. In Texture Workshop, pages 103–108, 2002.

[21] A. Efros and T. Leung. Texture synthesis by non-parametric sampling. in Proc. ICCV, Kerkyra, Greece, pages 1033–1038, 1999.

[22] T. Mäenpää and M. Pietikäinen. Multi-scale binary patterns for texture analysis. in Proc. SCIA, Gothenberg, Sweden, pages 885–892, 2003.

[23] S. Lazebnik, C. Schmid, and J. Ponce. Affine-invariant local descriptors and neighbourhood statistics for texture recognition. in Proc. ICCV, Nice, pages 649–655, 2003.

[24] T. Lindeberg, Feature detection with automatic scale selection, International Journal on Ccomputer Vision, 30(2), 1998, 79–116.

[25] R. Manthalkar, P.K. Biswas, and B.N. Chatterji. Rotation and scale invariant texture classification using gabor wavelets, in Texture Workshop, pages 87–90, 2002.

[26] J. Zhang, M. Marszalek, S. Lazebnik, C. Schmid, Local features and kernels for classification of texture and object categories: a comprehensive study, International Journal on Computer Vision, Vol 73(2), 2007, 213-238.

[27] D. Lowe, Distinctive image features from scale-invariant keypoints, International Journal of Computer Vision, 60 (2), 2004, 91-110.

[28] A. Johnson and M. Hebert, Using spin images for efficient object recognition in cluttered 3D scenes. IEEE Transactions on Pattern Analysis and Machine Intelligence, 21(5), 1999, 433-449.

[29] S. Lazebnik, C. Schmid, and J. Ponce, A sparse texture representation using local affine regions, IEEE Transactions on Pattern Analysis and Machine Intelligence, 27(8), 2005, 1265-1278.

[30] O. Chapelle, P. Haffner, and V. Vapnik, SVMs for histogram-based image classification. IEEE Trans. on Neural Networks, 10(5), 1999, 1055-1064.

[31] S. Avidan. Support vector tracking, in Proc. CVPR, Kauai, Hawaii, vol I:184–191, 2001.

[32] A. Barla, F. Odone, and A. Verri. Hausdorff kernel for 3D object acquisition and detection, in Proc. ECCV, Copenhagen, vol IV: 20 ff., 2002.

[33] C. Wallraven, B. Caputo, A. Graf, Recognition with local features: the kernel recipe, in Proc ICCV, Nice, France, 257-264, 2003.

[34] K. Grauman, T. Darrell, Pyramid match kernels: discriminative classification with sets of image features, in Proc ICCV, Beijing, China, vol. 2, 1458-1465, 2005.

[35] M. Pontil and A. Verri, Support vector machines for 3D object recognition, IEEE Trans on Pattern Aanalysis and Machine Intelligence, 20(6), 1998, 637–646.

[36] D. Roobaert, M. Zillich, and J.O. Eklundh, A pure learning approach to background-invariant object recognition using pedagogical support vector learning, in Proc. CVPR, Kauai, Hawaii, vol II,351–357, 2001.

[37] S.Z. Li, Q.D. Fu, L. Gu, B. Scholkopf, Y. Cheng, and H.J. Zhang, Kernel machine based learning for multi-view face detection and pose estimation, in Proc. ICCV, Vancouver, vol II, 674–679, 2001.

[38] B. Caputo and Gy Dorko, How to combine color and shape information for 3D object recognition: kernels do the trick, in Proc. NIPS, Vancouver, 2002.

[39] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: spatial pyramid matching for recognizing natural scene categories, in Proc CVPR, New York, 2006.

[40] K. I. Kim, K. Jung, J. H. Kim, Texture-based approach for text detection in images using support vector machines and continuously adaptive mean shift algorithm, IEEE Trans on Pattern Aanalysis and Machine Intelligence, 25(12), 2003, 1631-1639.

[41] S. Li, J. T. Kwok, H. Zhu, Y. Wang, Texture classification using the support vector machine, Pattern Recognition, 36(12), 2003, 2883-2893.

[42] C. Burges. Geometry and invariance in kernel based methods. *Advances in Kernel Methods - Support Vector Learning*, 1998.

[43] S. Belongie, C. Fowlkes, F. Chung, and J. Malik. Spectral partitioning with indefinite kernels using the Nyström extension, in Proc. ECCV, Copenhagen, vol III, 531 ff., 2002.

[44] J. C. Platt, N. Cristianini, and J. Shawe-Taylor. Large margin DAGs for multiclass classification, in Proc. NIPS 2000, Denver, Colorado, 2000.

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

[46] M. Varma. Private communication, 2003.

[47] S. Singh, J. Haddon, and M. Markou, Nearest-neighbour classifiers in natural scene analysis, Pattern Recognition, 34(8), 2001, 1601–1612.