# Illuminant Rotation Invariant Classification of 3D Surface Textures using Lissajous's Ellipses

Andreas Penirschke, Mike Chantler and Maria Petrou

*Abstract*— **Changes in the angle of illumination incident upon a 3D surface texture can significantly alter its appearance. Such variations effect texture feature images and can dramatically increase the failure rates of texture classifiers. In a previous paper we presented theory and experimental results that showed that changes in illuminant tilt angle cause texture clusters to describe Lissajous's ellipses in feature space. In this paper we use this model to develop a classifier that can classify surface textures imaged under unknown illumination tilt angles. In experiments with 30 real textures classification rates of over 99% were achieved.**

*Keywords*— **Texture, illumination, texture features, texture classification**

## I. Introduction

CHANGES in the angle of illumination incident upon a 3D surface texture can change its appearance significantly as illustrated in Fig. 1. Such changes in image

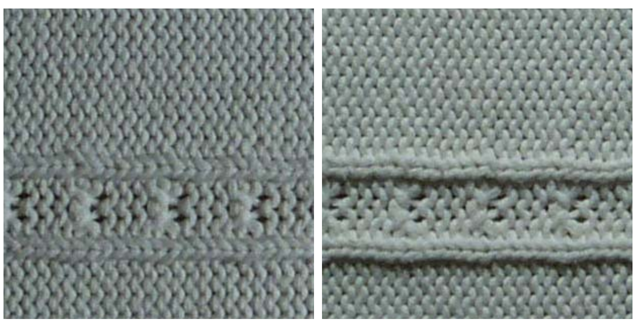

Fig. 1. Two images of the same surface texture sample captured using different illuminant tilt angles

texture can cause complete misclassification of surface textures [1]. Essentially the problem is that side-lighting, as used for instance in Brodatz's texture album [2], enhances the appearance of surface texture but produces an image which is a directionally filtered version of the surface height function.

In a previous paper [5] we presented theory and experimental results that showed that changes in illuminant tilt angle[1] cause texture clusters to describe Lissajous's ellipses in feature space. In this paper we use this model to develop a classifier that can classify surface textures imaged under unknown illumination tilt angles.

Very little work has been published on this subject. Dana, Nayer, van Ginneken and Koenderink established the Columbia-Utrecht database of real world surface textures which they used to investigate bidirectional texture functions [6]. Later they developed histogram [7], [8] and correlation models [9] of these textures. Leung and Malik developed a texture classification scheme that identifies 3D 'textons' in the Columbia-Utrecht database for the purposes of illumination and viewpoint invariant classification [10], [11].

In this paper we model a texture's behaviour in feature space as a hyper-elliptical function of illuminat tilt. We combine this with a multivariate Gaussian model of the effects of noise, shadowing etc. to provide a maximum likelihood classifier that identifies the class of the unknown texture and estimates the illuminant tilt direction.

The elliptical model of feature behaviour assumes the use of a set of 'linear texture filters'[4]. These are simply linear bandpass filters followed by energy estimation functions: such as Gabor filters, Laws masks, wedge and ring filters etc. Thus the classification scheme that we have developed is application to a wide range of classifiers.

The next section briefly presents the elliptical model of texture feature behaviour. This is followed by a description of the classifier. Finally, results using thirty real textures are presented and conclusions drawn.

## II. The output of Linear Texture Filters and their Features

### A. The behaviour of a single feature

We exploit a model of the surface to image transfer function originally due to Kube and Pentland [3]:

$$I(\omega,\theta) = \omega^2 cos^2(\theta-\tau) sin^2(\sigma) H(\omega,\theta) \quad (1)$$

where:

| | |
|---|---|
| $I(\omega,\theta)$ | is the image power spectrum; |
| $H(\omega,\theta)$ | is the surface power spectrum; |
| $\tau$ | is the illuminant tilt angle; and |
| $\sigma$ | is the illuminant slant angle. |

We define a Linear Texture Feature as a linear filter followed by a variance estimator[4]. The mean output of such a feature is therefore:

$$f(\tau) = \mathcal{VAR}(o(x,y)) \quad (2)$$

where $o(x,y)$ is the output of the linear filter.

Andreas Penirschke is with Darmstadt University of Technology, Germany; Mike Chantler is with the Texture Lab., Heriot-Watt University, Edinburgh, Scotland, (mjc@cee.hw.ac.uk, http://www.cee.hw.ac.uk/texturelab/) and Maria Petrou is with the Department of Electronic and Electrical Engineering, University of Surrey, United Kingdom (M.Petrou@ee.surrey.ac.uk, http://www.ee.surrey.ac.uk/Personal/M.Petrou)

[1] In the axis system we use, the camera axis is parallel to the $z$-axis, illuminant tilt is the angle the illuminant vector makes with the $x$-axis when it is projected into the $x, y$ plane, and illuminant slant is the angle that the illuminant vector makes with the camera axis.

If $o(x,y)$ is a zero mean filter and $O(\omega,\theta)$ is its power spectrum expressed in polar co-ordinates then:

$$f(\tau) \quad = \quad \int_0^{\infty}\int_0^{2\pi} \omega\, O(\omega,\theta)\, d\theta\, d\omega \qquad (3)$$

Using equation 1 on the preceding page we can express $O(\omega,\theta)$ as:

$$O(\omega,\theta) \quad = \quad \omega^2 cos^2(\theta-\tau) sin^2(\sigma) A(\omega,\theta) \qquad (4)$$

where:

$A(\omega,\theta)$ $= H(\omega,\theta)|\mathcal{F}(\omega,\theta)|^2$
$\mathcal{F}(\omega,\theta)$ is the transfer function of the linear filter

Using $cos^2(x) = 1/2\,(1+cos(2x))$ and $cos(x-y) = cos(x)cos(y) + sin(x)sin(y)$ gives:

$$f(\tau) \quad = \quad \int_0^{\infty} \omega^3 sin^2(\sigma) \int_0^{2\pi} 1/2\,[1+cos(2\theta)cos(2\tau) + sin(2\theta)sin(2\tau)] A(\omega,\theta)\, d\theta\, d\omega \qquad (5)$$

Hence:

$$f(\tau) \quad = \quad a + b\cos(2\tau) + c\sin(2\tau) \qquad (6)$$

The above parameters ($a$, $b$ and $c$) are all functions of illuminat slant ($\sigma$) the surface height function and the linear filter of the texture feature. None are a function of illuminant tilt ($\tau$). Thus equation 6 predicts that the output of a texture feature based on a linear filter is a sinusoidal function of illuminant tilt[2] with a period $\pi$ radians.

Figure 2 shows the behaviour of four texture features that are typical of the results that we obtained using 30 real textures. They clearly show that the features' outputs are a sinusoidal function of the illuminant's tilt angle ($\tau$).

### B. *Behaviour in a Multi-Dimensional Feature Space*

If two different features are derived from the same surface texture the results can be plotted in a two-dimensional $x, y$ feature space. From equation 6 we obtain:

$$x \quad = \quad f_1(\tau) = a_1 + b_1\cos(2\tau) + c_1\sin(2\tau) \qquad (7)$$
$$y \quad = \quad f_2(\tau) = a_2 + b_2\cos(2\tau) + c_2\sin(2\tau) \qquad (8)$$

Since the frequencies of the two cosines are the same, these two equations form two simple harmonic motion components. Therefore the trajectory in 2D feature space is a Lissajous ellipse. In the general case of two or more filters the result is an ellipse or hyper-ellipse.

Figure 3 on the next page shows the behaviour of two Gabor filters (F25A45com and F25A0com) as a function of illuminant tilt for six real textures. It clearly shows the elliptical behaviour of the cluster means.

[2]In the case of $A(\omega,\theta)$ being isotropic (for instance if both the surface and the filter are isotropic) the response will degenerate to a sinusoid of zero amplitude, i.e. it will be a constant (straight-line) function of $\tau$. However, if an isotropic filter is applied to a directional surface then $A(\omega,\theta)$ will not be isotropic and the tilt response will be a sinusoidal function of tilt.

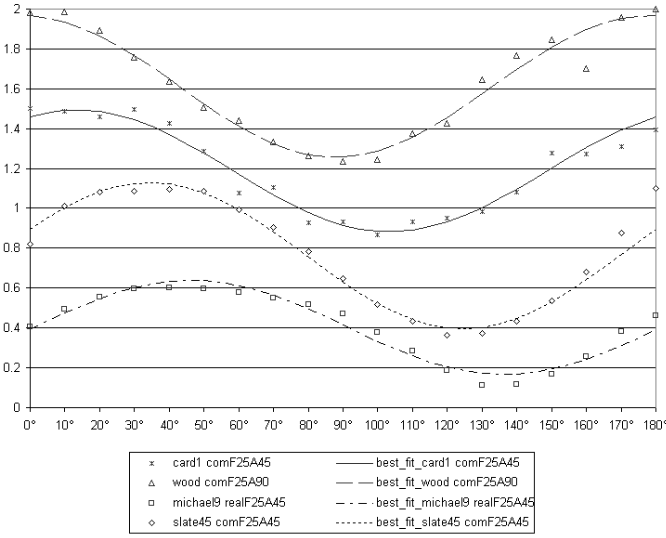


Fig. 2. Typical sinusoidal behaviour of texture features. (Each plot shows how one output of one feature varies when it is repeatedly applied to the same physical texture sample, but under varying illuminant tilt angles. Discrete points indicate measured output and the curves show the best-fit sinusoids)

## III. The Classifier

From figure 3 on the following page it is obvious that linear and higher order classifiers are likely to experience difficulty in dealing with this classification problem. We have therefore chosen to exploit the hyper-elliptical model of feature behaviour described above. We combine this with a multivariate Gaussian model of the effects of noise, shadowing etc. to develop a maximum likelihood classifier that identifies the class of the unknown texture and estimates its illuminant tilt.

### A. *Training*

The easiest way to visualise training of the classifier is with reference to the 2D case shown in figure 3 on the next page. Training requires estimation of the parameters that represent the elliptical behaviour of each texture class (*rock1, slab60* etc.). A set of training images of each texture is captured over a range of illumination tilt angles. Each image is used to calculate one feature vector value. The parameters of the ellipse are obtained by fitting:

$$f_i(\tau) = a_i + b_i cos(2\tau) + c_i sin(2\tau) \qquad (9)$$

to these points to obtain estimates of $a_i, b_i$ and $c_i$ for each feature $i$. We also estimate the variance $\sigma_i$ of the data from the best-fit sinusiod. Each texture class is therefore modelled by $4n$ parameters, where $n$ is the number of features.

### B. *Classification*

If we assume that the deviations of features from their elliptical behaviours are independent and follow a multivariate Gaussian distribution then we may express the like-

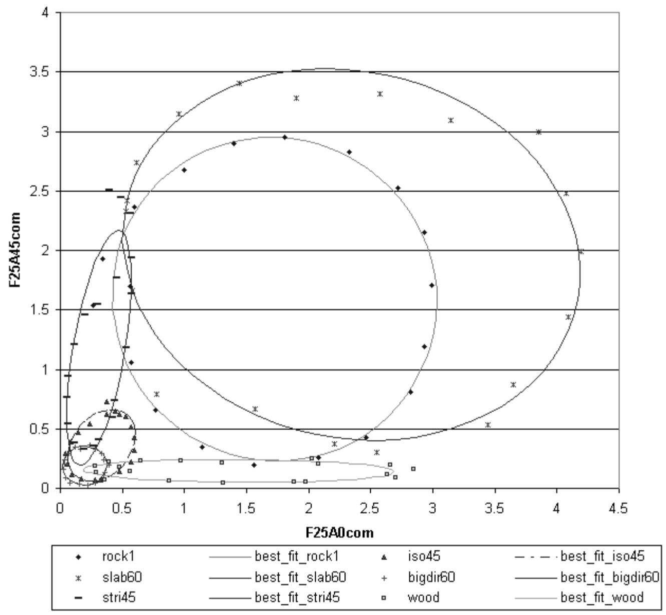


Fig. 3. The behaviour of six textures in the comF25A0/comF25A45 feature space together with the best fit ellipses (each point on an ellipse denotes a different value of illuminant tilt)

lihood that a texture belongs to a particular class as:

$$\varphi(\hat{\tau}) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}\sigma_i} e^{-\frac{(y_i - (a_i + b_i cos(2\hat{\tau}) + c_i sin(2\hat{\tau}))^2}{2\sigma_i}} \tag{10}$$

where $y_i$ is the value of the $i^{th}$ feature of the $n$-dimensional feature vector of the unknown texture, and $\hat{\tau}$ is the estimate of the illuminant tilt under which the texture was imaged.

The classification task therefore becomes one of maximising equation 10 with respect to $\hat{\tau}$ for each texture class, and assigning the unknown texture to the class with the maximum likelihood. In terms of figure 3 this approximates to finding the closest point on each ellipse weighted by the reciprocals of the variances $\sigma_i$.

To simplify the problem we take natural logs:

$$\begin{aligned} ln(\varphi(\tau)) &= \sum_{i=1}^{n} ln(\frac{1}{\sqrt{2\pi}\sigma_i}) \\ &- \sum_{i=1}^{n} \frac{(y_i - (a_i + b_i cos(2\tau) + c_i sin(2\tau))^2}{2\sigma_i} \end{aligned}$$

Substituting $(cos(2\tau))^2 = 1 - (sin(2\tau))^2$ and $x = sin(2\tau)$ reduces this function to a $4^{rd}$ order polinomial in $x$. We determine the maximum value of $\varphi(\hat{\tau})$ by finding the roots of the derivative of this polynomial.

The optimisation of equation 10 for each texture class provides both a likelihood figure for that class and an estimate of the tilt angle $\hat{\tau}$. The test sample is assigned to the class with the maximum likelihood and the associated $\hat{\tau}$ is returned as the tilt estimate.

TABLE I
Classification errors.

| Percentage of Classification errors | | | |
|---|---|---|---|
| *Number of Filters* | *six* | *five* | *four* |
| Detected errors | 0.81% | 2.02% | 3.77% |

## IV. Testing the Classifier

This section describes the texture features and image sets used to test the classifier it then presents the results that were obtained using thirty real textures.

### A. The Texture Features

Six Gabor [12] and two Laws filters [13] were used in various configurations in the classifier.

We use the notation typeFΩAΘ to denote a Gabor filter with a centre frequency of Ω cycles per image-width, a direction of Θ degrees, and of type complex or real. Five complex Gabor filters (comF25A0, comF25A45, comF25A90, comF25A135, comF50A45) together with one real Gabor filter (realF25A45) were implemented. The two Laws filters that we used were $L5E5$ and $E5L5$. Three combinations of features were used:
*set six:* four complex Gabor Filters and two Laws Filters.
*set five:* five complex Gabor Filters.
*set four:* four complex Gabor Filters.

### B. The Image-set

Thirty physical texture samples were used in our experiments. 512x512 8-bit monochrome images were obtained from each sample using illumination tilt angles ranging between 0° and 180° incremented by either 10° or 15° steps.

Every other image was selected for training. The remainder were used to test the classifier.

One sample image of each texture is shown at the end of this paper.

### C. Results

Both classification accuracy and the accuracy of illuminant tilt estimation were investigated.

Table I shows the overall misclassification rates that occurred. Table II on the next page details the misclassifications for the six and five filter feature sets. For instance it shows that using six filters, the classifier misclassified *slab45* imaged using an illuminant tilt angle of 70°, as *michael6* imaged at a tilt angle of 18°. Examining the images in the appendix explains some of the misclassifications e.g. *twins45, stri45* and *iso45* appear similar. Others look quite different from one another e.g. *radial45* and *michael3*. However, it should be noted that the distinction between these two textures blurs when *michael3* is imaged at 90° of tilt, as this filters out much of the 0° spaghetti texture.

Figures 4 and 5 on the facing page show the errors that occurred in estimating the illuminant tilt angles. Figure 4

TABLE II

Classification results.

| Misclassification | | | | | |
|---|---|---|---|---|---|
| *Input* | | *six filters* | | *four filters* | |
| texture | tilt | texture | tilt | texture | tilt |
| stones2 | 50 | | | chips1 | 23 |
| stones2 | 170 | | | michael7 | 2.2 |
| radial45 | 170 | | | michael3 | 90 |
| slab45 | 70 | michael6 | 18 | michael6 | 22 |
| twins45 | 90 | stri45 | 93 | iso45 | 90 |
| michael2 | 170 | | | michael8 | 180 |

shows the mean square error in tilt (the mean being calculated over each set of test images obtained from a single texture for a particular classifier). Figure 5 shows a histogram of the all the errors that occurred. Both of these charts show that in the majority of cases the illuminant tilt is estimated to within 5°. Only in a very small number of cases, such as *card1* and *and7*, does the error exceed 10°.

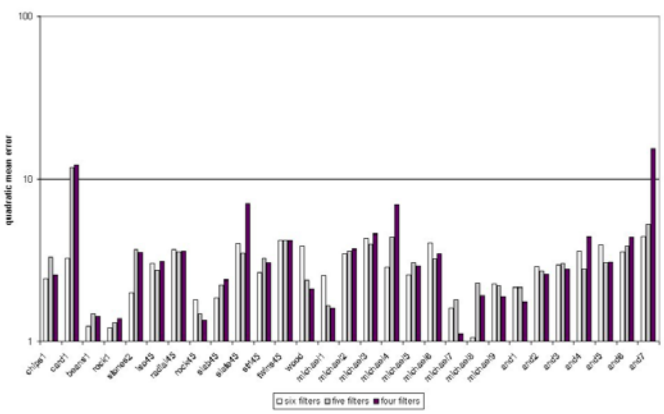


Fig. 4. Bar-chart of error metric values for the use of four, five complex Gabor Filters and a combination of four complex Gabor Filters and two Laws Filters showing the goodness of tilt deviation over the different texture samples

## V. Conclusions

We have presented a new tilt invariant 3D surface texture classifier that exploits the hyper-elliptical behaviour of texture features. It has been shown to perform well on 30 real textures: providing high classification accuracy and good illuminant tilt angle estimation when used with complete images.

The immediate questions that arise from this research are:

1. Can this approach be used for pixel by pixel classification, and hence segmentation?
2. How can we make the classifier robust to changes in illuminant slant?

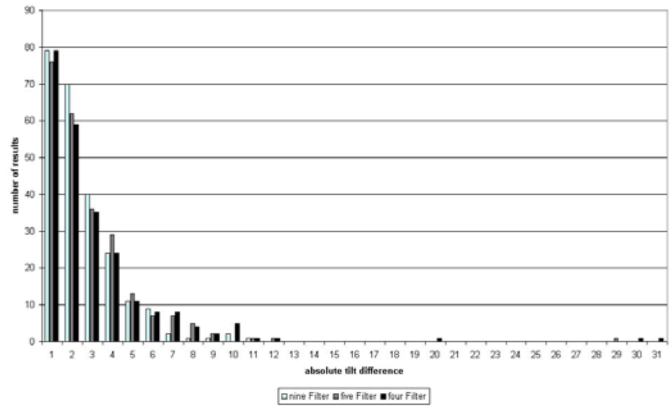


Fig. 5. Histogram of all results using four, five and six filters

## References

[1] A. Author. Title. *Journal of*, page pages, year.

[2] P. Brodatz. *Textures: a photographic album for artists and designers*. Dover, New York, 1966.

[3] P.R. Kube and A.P. Pentland. On the imaging of fractal surfaces. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 10(5):704–707, September 1988.

[4] T. Randen and J.H. Husoy. Filtering for texture classification: A comparative study. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 21(4):291–310, April 1999.

[5] A. Author. Title. In *Proceedings of*, page pages, year.

[6] K.J. Dana, S.K. Nayar, B. van Ginneken, and J.J. Koenderink. Reflectance and texture of real-world surfaces. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 151–157, 1997.

[7] K.J. Dana and S.K. Nayar. Histogram model for 3d textures. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 618–624, 1998.

[8] B. van Ginneken, J.J. Koenderink, and K.J. Dana. Texture histograms as a function of irradiation and viewing direction. *International Journal of Computer Vision*, 31(2/3):169–184, April 1999.

[9] K.J. Dana and S.K. Nayar. Correlation model for 3d texture. In *Proceedings of ICCV99: IEEE International Conference on Computer Vision*, pages 1061–1067, 1999.

[10] T. Leung and J. Malik. Recognizing surfaces using three-dimensional textons. In *Proceedings of ICCV99: IEEE International Conference on Computer Vision*, pages 1010–1017, 1999.

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

[12] A.K. Jain and F. Farrokhnia. Unsupervised texture segmentation using gabor filters. *Pattern Recognition*, 24(12):1167–1186, December 1991.

[13] K.I. Laws. *Textured Image Segmentation*. PhD thesis, Electrical Engineering, University of Southern California, 1980.

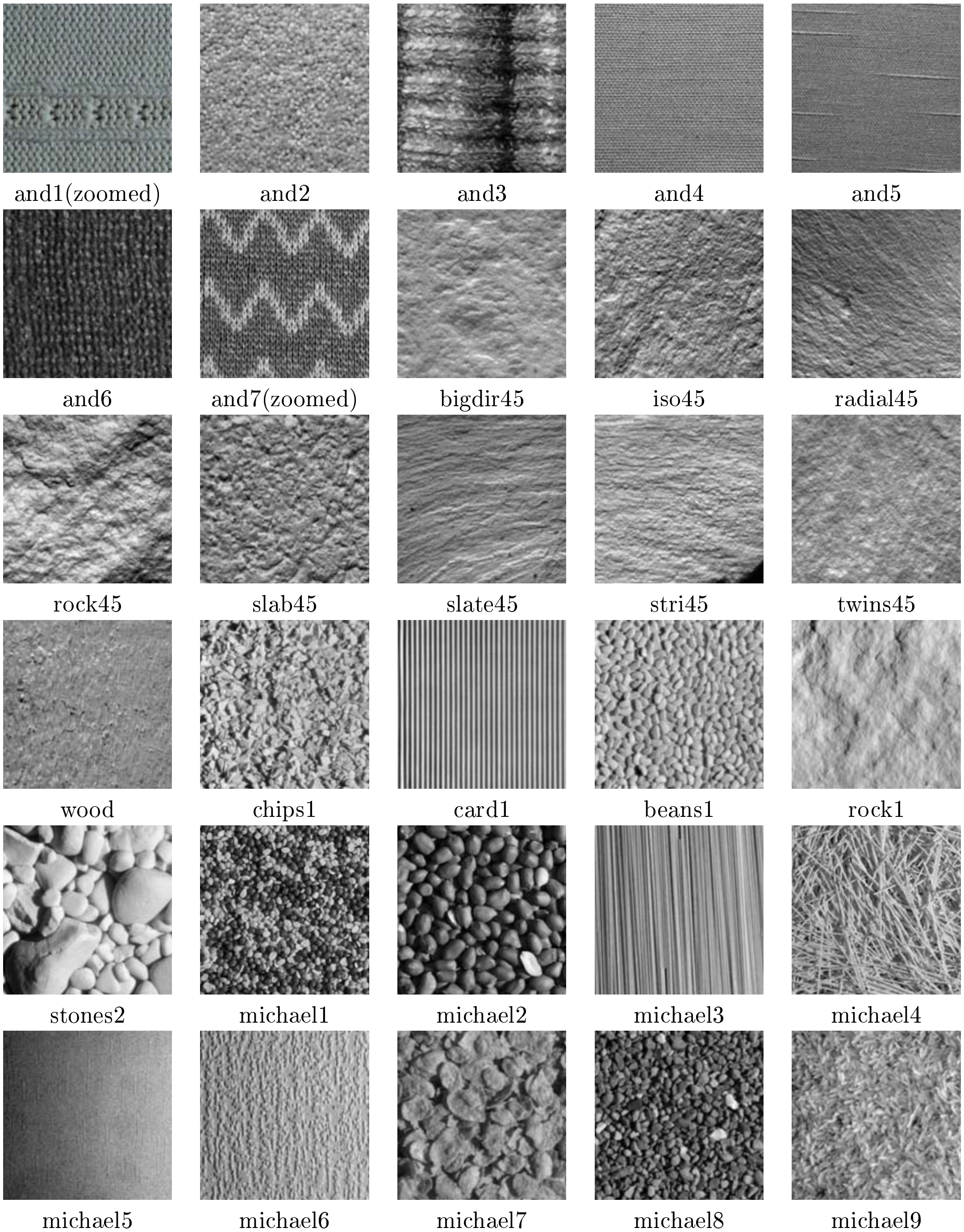


TABLE III

One image of each of the thirty sample textures