

# Segmentation of Perspective Textured Planes Through the Ridges of Continuous Wavelet Transform

Wen-Liang Hwang<sup>†</sup>, Chun-Shien Lu<sup>‡†</sup>, Pau-Choo Chung<sup>‡</sup>

<sup>†</sup> Institute of Information Science, Academia Sinica, Taiwan.

<sup>‡</sup> Department of Electrical Engineering, National Cheng Kung University, Taiwan.

Correspondence: Hwang, Wen-Liang

Institute of Information Science, Academia Sinica

Taiwan

whwang@iis.sinica.edu.tw

(Taiwan) 02-27883799 ext. 1609

Revised: JVC1999-0062.

## **Abstract**

A common assumption of the shape from texture problem is that a perceived image mainly contains only one type of texture with the same surface orientations. Unfortunately, a natural image is often composed of textured planes with different surface orientations. In order to deal with the shape from texture problem in a practical manner, we need to segment these surface orientations. The ridges are wavelet attributes in which information about spatial frequencies resides. In this paper, we propose a robust method for treating this problem from the ridges of continuous wavelet transform. We demonstrate the performance of our method on textured images synthesized from the Brodatz texture and several natural images.

Keyword: wavelet transform, texture, segmentation, ridge, shape from texture.

# 1 Introduction

Shape from texture methods estimate the surface orientations of a perspective projected textured surface from a monocular image. In this article, we will consider when the surface is a plane. Readers should refer to [16][15][14] for solutions of shape from texture in a curved surface. A plane surface has an orientation, described by two angles: the slant angle, which determines the degree of obliqueness of the plane, and the tilt angle, which is the direction of the slant. In the literature, almost all the shape from texture methods assume that an image is composed of only one perspective projected texture [2, 9, 11, 17]. However, in applications such as autonomous navigation, runway detection, and in many natural textured image, there are more than one perspective textured plane appears in an image [18][19]. Thus, the shape from texture methods can not be applied directly without first segmenting the textured planes. Different from conventional texture segmentation, it is the perspective geometry and, thus, the spatial relations of textured planes, rather than the intrinsic properties of textures, that are derived for segmentation. Thus, one can envision that textures will be separated once they reside in different surface planes in three dimensional space. Meanwhile, textures appearing in the same surface plane will be retained as a “coherent” textured component since they are subjected to an identical perspective projection. Fig. 2(a) shows an image composed of two inclined brick walls. Since the two brick walls are textured identically, they can only be segmented by means of perspective projections.

This segmentation problem has been neglected and has rarely been considered in previous methods. As far as we know, a close problem was first introduced by Krumm and Shafer [11, 12]. However, in Krumm and Shafer’s work, they focused on segmenting surface textures, *i.e.*, textures before perspective projection, by their local spatial frequencies. Their approach segments a textured plane into multiple components with distinct spatial frequency on each of them. In this paper, we aim at segmenting textured planes from an image obtained by perspective projection. We propose an algorithm which does not completely succeed in separating textured planes but the algorithm is quite effective in separating the planes which contain textures with dominant spatial frequencies. Thus, our results differ from Krumm and Shafer’s (1) when sur-

face textures with different spatial frequencies appear in a surface plane, and (2) when surface textures with the same spatial frequencies appear in different surface planes. In the first case, the textures are retained as a “coherent” component by Krumm and Shafter’s method but not by ours while in the second case, the textures are retained by our method but not by Krumm and Shafter’s.

In our previous work, we proposed a shape from texture method based on the ridges of continuous wavelet transform(CWT). Our method work in the case when the textured surface is flat and the surface texture can be well modeled as a superposition of periodic components. We showed that once a texture contains dominant spatial frequencies, our shape from texture method can accurately estimate the surface orientations of the texture[9]. Based on that method, we adopt an inlier selection which removes the ridge points corresponding to the textural variations that are not related to the estimation of local surface orientations. The local surface orientations are then estimated from the inlier ridge points and are then “unsupervised” clustered. Finally, the perspective textures are segmented according to the clustering results. Our method produces no blocky artifacts along the segmentation boundaries. We have verified the applicability of our method to synthesized as well as natural images. The performance of our algorithm is constrained by the feasibility of obtaining ridge points, because we use the ridge points in determining the local surface orientations.

This paper is an extended version of the conference paper presented at [13]. The rest of this paper is organized as follows. In Sec. 2, we will review the two-dimensional CWT as well as our previous shape from texture method since we will adopt that method in our algorithm. In Sec. 3, we will propose our algorithm. The detailed implementation of and the experimental results obtained using our method will be given in Sec. 4. A conclusion will be given in the last section.

## 2 Shape from Texture Using the Ridges of CWT

Several methods have been proposed in determining the surface orientations of a textured plane in a monocular image. We will assume that our image is composed of textures obtained

perspectively from the superposition of the following surface texture:

$$A \cos(\boldsymbol{\Omega}^T \mathbf{x}_s), \quad (1)$$

where  $A$  is a magnitude constant,  $\boldsymbol{\Omega}$  is the spatial frequency of the textures in the surface plane, and  $\mathbf{x}_s$  is the coordinate in the plane. In [9], we have proposed a shape from texture method which used the ridges of CWT in characterizing the surface orientations. That method is most suitable to texture given in Eq.(1), even with additive white noise in a low SNR environment. In the followings, we will review CWT, the ridges, and our previous shape from texture method.

## 2.1 2D CWT and Ridges

A complex-valued function  $\psi(\mathbf{x})$  in  $L^2(\mathcal{R}^2)$  is a wavelet if  $\int_{\mathcal{R}^2} \psi(\mathbf{x}) d\mathbf{x} = 0$ . This condition is a relaxed version of the wavelet admissibility condition given by

$$\int \frac{|\hat{\psi}(\mathbf{w})|^2}{|\mathbf{w}|^2} d\mathbf{w} < \infty,$$

where  $\mathbf{w}$  is the spatial frequency in a plane. Let  $\psi_{(\mathbf{b}, s, \theta)}(\mathbf{x})$  be obtained by means of the translation, scaling, and rotation of  $\psi(\mathbf{x})$ , and let it take the following form:  $\psi_{(\mathbf{b}, s, \theta)}(\mathbf{x}) = \frac{1}{s^2} \psi(\mathbf{r}_{-\theta} \frac{(\mathbf{x}-\mathbf{b})}{s})$ , where  $\mathbf{b} \in \mathcal{R}^2$ ,  $s > 0$ , and  $\theta \in [0, 2\pi)$  are the translation, scaling, and rotation parameters, respectively.  $\mathbf{r}_{\theta}$  is the rotation matrix of angle  $\theta$ . The 2D CWT [1] of  $f(\mathbf{x})$  is defined as its convolution product with  $\psi_{(\mathbf{b}, s, \theta)}(\mathbf{x})$ :

$$(\mathcal{W}f)(\mathbf{b}, s, \theta) = \frac{1}{s^2} \int_{\mathcal{R}^2} f(\mathbf{x}) \overline{\psi_{(\mathbf{b}, s, \theta)}(\mathbf{x}-\mathbf{b})} d\mathbf{x}, \quad (2)$$

where  $\overline{\psi(\mathbf{x})}$  is the complex conjugate of  $\psi(\mathbf{x})$ . One can interpret the squared-modulus of the wavelet coefficients  $|(\mathcal{W}f)(\mathbf{b}, s, \theta)|^2$  as the energy density of  $f(\mathbf{x})$ . The Morlet wavelet is defined as  $\psi_M(\mathbf{x}) = e^{j\mathbf{k}_0^T \mathbf{x}} e^{-|\mathbf{x}|^2/2} - e^{-|\mathbf{k}_0|^2/2} e^{-|\mathbf{x}|^2/2}$ , which is  $\hat{\psi}_M(\mathbf{w}) = e^{-|\mathbf{w}-\mathbf{k}_0|^2/2} - e^{|\mathbf{w}|^2/2} - e^{|\mathbf{k}_0|^2/2}$  in frequency domain, where  $\mathbf{k}_0 = [k_0, 0]^T$  is the center frequency of the Morlet wavelet. A simple approximation of the Morlet wavelet is known as the Gabor wavelet,  $\hat{\psi}_G(\mathbf{w}) = e^{-|\mathbf{w}-\mathbf{k}_0|^2/2}$ . Strictly speaking, the Gabor wavelet is not admissible to be a wavelet. However, it approximates the Morlet wavelet in the case for  $|\mathbf{k}_0|$  large enough. In our implementation, we use the Gabor

wavelet. Also, following the conventional approach [1], the scale parameter  $s$  takes discretized values with  $s = 2^{o+\frac{v}{n}}$ , where  $o$  is the octave,  $v$  is the voice, and  $n$  is the number of voices per octave. The Gabor wavelet optimizes both the spatial and frequency resolutions simultaneously and is, therefore, well adapted to characterizing the local spatial frequency of  $f(\mathbf{x})$ .

In many applications, the CWT coefficients are not used directly in analysis since doing so would be too redundant; instead, it is the local frequencies contained in the ridge points that are useful. For a precise extraction of ridge points, we refer to the ‘‘ridge method’’ which is based on the analysis of the phase of the CWT[5]. It was pointed out in [4] and [9] that ridge points are better preserved in the modulus of CWT in noisy environments. A good approximation of the ridge point  $\mathbf{p} = (\mathbf{b}, s_0, \theta_0)$  at a position  $\mathbf{b}$  can be obtained from the local modulus maxima of CWT throughout the neighborhood of  $\mathbf{p}$  in scale and rotation: Let  $\mathcal{N}(\ast)$  denote the neighborhood of the argument  $\ast$ , including  $\ast$ ; then,  $(\mathbf{b}, s_0, \theta_0)$  is selected as a ridge point if

$$\forall \mathcal{N}(s_0) \quad \forall \mathcal{N}(\theta_0) \quad |(\mathcal{W}f)(\mathbf{b}, s_0, \theta_0)| \geq |(\mathcal{W}f)(\mathbf{b}, \mathcal{N}(s_0), \mathcal{N}(\theta_0))|.$$

Ridge point is at  $(\mathbf{b}, s_0, \theta_0)$  where the local modulus maximum of the wavelet transform appears. It is possible for multiple ridge points to occur at the position  $\mathbf{b}$ . One can connect the neighboring ridge points as a continuous surface, called a ridge surface. This corresponds to the component in the wavelet domain that has coherent frequency variations in an image.

## 2.2 Estimation of the Surface Orientations of a Textured Plane

The ridge points are the wavelet attributes in which information about the local frequencies resides. We assume that the surface textures  $f(\mathbf{x}_s)$  that can be well approximated by the a superposition of sinusoidal components:

$$f(\mathbf{x}_s) \approx \sum_{k \in I(\mathbf{x}_s)} A_k \cos(\boldsymbol{\Omega}_k^T \mathbf{x}_s), \quad (3)$$

where  $\mathbf{x}_s = [x_s \ y_s]^T$  is the surface coordinate,  $\boldsymbol{\Omega}_k$  is the  $k$ -th spatial frequency and  $A_k$  gives the amplitude of the  $k$ -th frequency component. If  $k \in I(\mathbf{x}_s)$ , then this implies that the  $k$ th amplitude in the vector  $[A_i; i = 1, 2, \dots]^T$  and the  $k$ th frequency in the vector  $[\boldsymbol{\Omega}_i^T; i = 1, 2, \dots]^T$  appear

at  $\mathbf{x}_s$ . One can show that as long as the wavelet  $\hat{\psi}(\mathbf{w})$  is well localized in frequency such as Gabor wavelet and the different frequency components of our texture given in Eq. (3) are sufficiently separated, the energy at  $(\mathbf{b}, s, \theta)$  in the wavelet domain can be approximated as the summation of the energy of each individual component:  $|(\mathcal{W}f)(\mathbf{b}, s, \theta)|^2 \approx \sum_{k \in I(\mathbf{b})} \frac{A_k^2}{4} \left| \hat{\psi}(s\mathbf{r}_{-\theta}\mathbf{\Omega}_k) \right|^2$ . Assume that  $\hat{\psi}(\mathbf{w})$  is concentrated at the frequency  $\mathbf{k}_0$ , the  $k$ -th frequency component  $\mathbf{\Omega}_k$  of the image will be concentrated around  $\mathbf{r}_{\theta_k}\mathbf{k}_0/s_k$ , where  $\theta_k$  is the angle between  $\mathbf{k}_0$  and  $\mathbf{\Omega}_k$ , and  $s_k^{-1}$  is the magnitude multiplier in order to scale  $\|\mathbf{k}_0\|$  to  $\|\mathbf{\Omega}_k\|$ . Thus, the texture energy at  $\mathbf{b}$  is concentrated around  $|I(\mathbf{b})|$  different components centered at points

$$(\mathbf{b}, s = \frac{\|\mathbf{k}_0\|}{\|\mathbf{\Omega}_k\|}, \theta_k). \quad (4)$$

One can then read off from these points (called ridge points hereafter) important local parameters about the spatial frequency  $\mathbf{\Omega}_k$ . The ridge points at the given  $\mathbf{b}$  can be extracted by selecting the squared-modulus local maxima among  $\theta$  and  $s$  at  $\mathbf{b}$ .

We will use the perspective model that was adopted in [17]. Let the coordinate systems of the world  $(x_w, y_w, z_w)$ , of the surface plane  $(x_s, y_s, z_s)$ , and of the image  $(x_i, y_i, z_i)$  be those depicted in Fig. 3. The slant angle  $\rho$  is defined as the angle of the normals of the image plane and the surface plane. A convenient coordinate  $(x, y, z_i)$  in our shape from texture method is obtained by rotating the image coordinate such that the  $x$ -coordinate lines in the tilt direction. Let us denote  $\mathbf{x} = [x \ y]^T$ , and let the spatial frequencies of the surface texture be  $\mathbf{\Omega}_k = [u_k \ v_k]^T$ . The image texture in coordinate  $(x, y, z_i)$ , obtained perspectively from Eq. (3), will be

$$g(\mathbf{x}) = \sum_{k \in I(\mathbf{x})} A_k \cos\left(\frac{z_w}{f}(u_k x \sec \rho + y v_k)\right) = \sum_{k \in I(\mathbf{x})} A_k \cos(\phi_k(\mathbf{x})), \quad (5)$$

where

$$\frac{z_w}{f} = \frac{z_0}{\tan \rho(x_i \cos \tau + y_i \sin \tau) + f}. \quad (6)$$

The  $k$ -th spatial frequency at  $\mathbf{x}$  is the gradient of  $\phi_k(\mathbf{x})$  which is also called the instantaneous frequency at  $\mathbf{x}$ :

$$\left[ \frac{\partial \phi_k(\mathbf{x})}{\partial x} \quad \frac{\partial \phi_k(\mathbf{x})}{\partial y} \right] = \left[ \frac{z_0 u_k f \sec \rho}{(x \tan \rho + f)^2} + y \frac{\partial^2 \phi_k(\mathbf{x})}{\partial x \partial y} \quad \frac{\partial \phi_k(\mathbf{x})}{\partial y} \right]. \quad (7)$$

By assuming that the spatial frequency variation along  $y$ , the direction perpendicular to the tile, is negligible, we obtain a simple way in relating the scales  $s(\mathbf{x})$  in the ridge points of  $\mathcal{W}g(\mathbf{x})$  to the spatial frequency variation in  $x$ , the tilt direction, by Eq. (4):

$$s(\mathbf{x}) = \frac{fk_0(1 + \frac{x \tan \rho}{f})^2}{z_0 u_k \sec \rho}. \quad (8)$$

It is clear that the scale  $s(\mathbf{x})$  of the ridge surface is a parabolic function of  $x$ , independent of  $y$ . **Estimation of Tilt:** Only the  $x$ -coordinate (*i.e.* the coordinate in the tilt direction) and the scale of a ridge point are used in estimating the slant. The scale for slant estimation is a parabolic function of  $x$ . If we assume that the scales of ridge points in image coordinates,  $(x_i, y_i, z_i)$ , takes the following parabolic formula:  $s(x_i, y_i) = a_1 x_i^2 + a_2 x_i y_i + a_3 y_i^2 + a_4 x_i + a_5 y_i + a_6$ ; Then, with a rotation of exactly  $\tau$  degrees,

$$\tau = \tan^{-1} \frac{a_5}{a_4},$$

which is the tilt angle, the mixed-product term of the parabola can be eliminated.

**Estimation of Slant:** After the tilt angle is estimated, the slant angle can be calculated by using a curve fitting method according to Eq. (8).

### 3 Segmentation of the Perspective Textured Planes

We are concerned with the segmentation of textured regions in multiple planes under a perspective projection according to the spatial frequency variations of the textures. The major distinction between this and conventional texture segmentations lies in the fact that the geometrical parameters indicating the spatial relations of the textured planes and the image, rather than the intrinsic properties of the textures, are segmented. Therefore, identical textures are separated if they appear in different surface planes. Meanwhile, different textures are retained as a “coherent” component once they have been textured in the same surface plane.

We consider solutions to segmentation problems to be those which provide robust methods for partitioning images. Assume that our image is composed of  $N$  distinct perspective textured

regions:

$$\sum_{i=1}^N g^i(\mathbf{x}^i) P^i(\mathbf{x}^i), \quad (9)$$

where  $P^i(\mathbf{x}^i)$  takes value 1, indicating that  $\mathbf{x}^i$  is in the  $i$ -th perspective region; else 0. If textures are obtained from projection of the surface textures given in Eq.(3), then  $g^i(\mathbf{x}^i)$  takes the form of Eq. (5), given as

$$g^i(\mathbf{x}^i) = \sum_{k \in I^i(\mathbf{x}^i)} A_k^i \cos\left(\frac{z_w^i}{f}(u_k^i x^i \sec \rho^i + y^i v_k^i)\right) = \sum_{k \in I^i(\mathbf{x}^i)} A_k^i \cos(\phi_k^i(\mathbf{x}^i)), \quad (10)$$

where  $\mathbf{x}^i = [x^i \ y^i]^T$  is the coordinate in the  $i$ -th perspective region with the  $x^i$ -axis lined up in the tilt direction of the region. Even with such a simplified texture model, one can envision the difficulties that will be encountered if we intend to characterize all the parameters, including those of textures and of perspective projections. In our problem's concern, only the set of slant and tilt angles,  $\{[\rho^i, \tau^i]^T; i = 1, \dots, N\}$ , need to be determined for segmentation. The  $i$ -th perspective textured region is delaminated by the index function  $P^i(\mathbf{x}^i)$  which is the image region having, numerically, the same slant and tilt angles. In texture segmentation parlance, a supervised method is one in which the number of regions, which is  $N$ , to be discriminated is known as *a priori*; otherwise, the method is unsupervised.

Textured regions to be separated are characterized by their surface orientations. As mentioned above, one can calculate the surface orientations from a ridge surface. A naive approach will be as follows: Take a block in a ridge surface; then estimate the surface orientations with the ridge points restricted to the block. We obtain the surface orientations localized to the block. After all the blocks are examined and estimations are made, blocks with "similar" surface orientations are then merged. This procedure can be iterated. However, the estimations of the local surface orientations are marred by errors due to modeling noises. The variation of the local surface orientations is usually very big even if they are obtained from regions in the same perspective texture. Since there are only a small number of ridge points for estimation of local surface orientations, the existence of outliers tend to bias the estimation results dramatically. Thus, an inlier selection procedure, which removes ridge points that are not related to the estimation of surface orientations, is much needed to reduce the variation of local estimations.

We introduce a voting scheme in the two dimensional parameter space of slant and tilt  $(\rho, \tau)$  for robust clustering of the estimated results[7]. Once a local surface orientations is obtained, it is voted by adding 1 to the corresponding bin in the parameter space. This will produce a distribution of the vote in the parameter space. The statistical properties of the distribution are then investigated to recover the clusters in the parameter space. Most of the statistical clustering algorithm assumes that the distribution is composed of Gaussian mixtures, and that a Gaussian distribution corresponds to the aggregation of estimation results from the same perspective texture. The number of clusters in the parameter space is not necessarily given as *a priori*. We assume that the estimated local surface orientations  $\underline{o}$  of a perspective texture, which is a vector of the slant  $\rho$  and the tilt  $\tau$ , is a random variable, written as

$$\underline{o} = o + \underline{n},$$

where  $\underline{n}$  is the perturbation of the estimation, usually assumed to be a Gaussian distribution, and  $o$  is the mean. One should notice that  $o$  is the cluster center in the voting space, which does not necessarily represent the real surface orientations of a textured region. The robustness of our approach lies in the fact that once the local estimations of one texture produce a variation that is likely to be small and aggregated, the clustering approach can be applied. This clustering strategy is proper since the differences in the surface orientations, rather than the precise values of the surface orientations, are taken into account in separating textured regions.

## 4 Implementation and Experimental Results

### 4.1 Implementation

The following four steps are included in our algorithm: (1) multiple ridge surface detection, (2) local feature extraction, (3) coarse segmentation, and (4) fine segmentation. These steps are depicted in Fig. 4 and are discussed below.

## Multiple Ridge Surfaces Detection

Images contain more than one dominant frequency component will produce multiple ridge surfaces. In our implementation, a ridge surface can be found by applying the connected component algorithm in the wavelet domain. That is, any two ridge points differ by 1 on either their scales, angles, and positions are assigned to the same ridge surface. Multiple ridge surfaces are obtained by repeatedly applying the connected component algorithm such that any ridge point belongs to a ridge surface. Fig. 1 gives an example of multiple ridge surfaces of a building: (a) gives the building; (b), (c), and (d) show the scales of the ridge surfaces of the horizontal and the vertical structures of the building, and of the objects in front of it, respectively.

## Estimation of Local Surface Orientations

The objective is to produce small estimation variation of the local surface orientations for a perspective textured region. We use a technique which allows us to estimate locally from the inliers, where the size of the window is not fixed. Our method is based on the robust regression method: RANSAC[6]. The smoothing techniques for RANSAC is characterized by using as small an initial data set as feasible and enlarges this set with consistent data when possible.

Recall that the scales of the ridge surface related to the surface orientations is a parabolic function. The consensus condition in our case is a tolerable fitting error. Our local surface estimation algorithm is :

- Repeat the following  $N$  times.
  1. We start with a small but sufficiently large window (typical  $16 \times 16$ ), the center of which is randomly chosen from a ridge surface.
  2. Then, a set of 6 ridge points is randomly selected within the window as a seed. These ridge points are collected in set  $R$ . From 6 ridge points, we obtain our parabolic function  $s(\mathbf{x})$ .
  3. We then double the size of the window and randomly select new ridge points within the new window.

4. A ridge point  $(\mathbf{b}, s_b)$  is included as an inlier if its scale distance to the parabola  $s(\mathbf{x})$  is tolerable, that is,  $|s_b - s(\mathbf{b})| < \epsilon$ . The ridge point is included in the set  $R$ . This process will increase the number of selected ridge points in  $R$ . A new parabolic function  $s(\mathbf{x})$  is obtained by including the inlier ridge point in the parabolic estimation. Repeat this step several times.
5. Return the estimated local surface orientations and the set  $R$ .

Because inlier ridge point selection is a random process, even if we select an initial window which was selected before to start a new estimation, a different set of ridge points may have been included, resulting in numerically different surface orientations. The above procedure is performed many times. The typical number of  $N$  in our experiments was 1000 for a ridge surface approximately of size 256 by 256. This random procedure saves computational time; if the final window is 32 by 32, then we have a speed-up of about 65 times compared to window shifting on each ridge point on the ridge surface.

## Coarse Segmentation

In the literature, coarse segmentation is usually carried out by clustering methods. In our implementation, we used a voting strategy to clustering the local surface orientations. The parameter space for our voting method is the space of slant  $\rho$  and tilt  $\tau$ . We have mentioned the algorithm that we used in estimate the local surface orientations. In that algorithm, we repeat the estimations of local surface orientations  $N$  times. Each time we obtain a value of  $(\rho_i, \tau_i)$  for  $i = 1, \dots, N$ . These values are quantized and voted to the parameter space of slant and tilt: Once two local surface estimations have the same quantized slant and tilt, we add 1 to the corresponding bin of slant and tilt in the parameter space. Finally, we have a vote distribution which is a function from the parameter space to a number between 1 and  $N$ . Then, we adopt the robust unsupervised clustering method proposed in [10] to cluster the vote patterns in the parameter space. We use this method simply because it is convenient for our implementation. We believe that other unsupervised methods will come to a similar result.

However, it is the image pixels instead of the local surface orientations that we intend to

segment. To recall that in estimating local surface orientations, we have associated each run of the estimation a set  $R$ , which has all the inlier ridge points of the run. Since our local estimation is a random process, a ridge point may be assigned to one surface orientations many times and to more than one surface orientations as well, as shown in Fig. 5. An image pixel is then classified using the following simple procedure:

Let  $\mathbf{x}$  be the image pixel to be classified, let us assume that there are  $L$  clusters in the parameter space (obtained after unsupervised clustering of vote distribution), and let  $V_i$  be the collection of all the surface orientations in the cluster  $i$ . For any bin of slant and tilt  $\underline{\rho} \in V_i$ , the occurrence of the pixel  $\mathbf{x}$  within the ridge points associated with bin  $\underline{\rho}$  is recorded. We sum up this record for all the bins in  $V_i$  and obtain the total number the pixel  $\mathbf{x}$  supporting the cluster  $V_i$ . This number is denoted as  $S_i(\mathbf{x})$ . The pixel  $\mathbf{x}$  is, then, assigned to the cluster which has the largest  $S_i(\mathbf{x})$  for all  $i = 1, \dots, L$ ,

$$\mathbf{x} \rightarrow \arg_{i=1, \dots, L} \max S_i(\mathbf{x}). \quad (11)$$

The above procedure performs our image partitioning beginning with a ridge surface. However, images usually contain multiple ridge surfaces. The segmentation results from all the ridge surfaces should be integrated for final segmentation of textures. Our integration is done by using the following simple method: Let the partitioned image induced by the ridge surface  $r$  be denoted as  $P^r = \{p_i^r; i \in 1 : m(r)\}$ , where  $m(r)$  perspective regions are classified using the ridge surface  $r$ . A new partition  $P$  is generated by integrating the two partitions  $P^s$  and  $P^t$ , with respective to the ridge surfaces  $s$  and  $t$ , respectively, using the following simple strategy:

$$P = \{[p_i^s p_j^t]; i \in 1 : m(r), j \in 1 : m(s)\}.$$

The image pixels that have the same index in  $P$  are then assigned to the same partition. We summary our coarse segmentation algorithm as follows:

- Assume that there are  $M$  ridge surfaces :
  1. Estimate the local surface orientations of a ridge surface.
  2. Unsupervised clustering of the space of slant and tilt. The local surface orientations in the ridge surface are classified.

3. An image pixel is assigned to a class according to Eq.(11). Image pixels are partitioned according to the ridge surface.

- Integration of the M image partitions into a final partition.

In Fig. 4, a pictorial description of the mentioned procedure is given.

## Fine Segmentation

Fine segmentation is needed to produce a fine visual result. After coarse segmentation, there are two leftover problems that we must fix: holes and boundaries. A hole is a region surrounded by only one texture class. Image pixels within a hole can then be re-classified by simply assigning them to the class surrounding them. Boundaries are surrounded by more than one texture class. A boundary pixel is classified to its nearest surrounding class. One simple method is to center the pixel  $p$  to be classified a window. Then, the number of the classified pixels within the window for each class is counted and the pixel to be classified is assigned to the class to which most number of pixels belong.

## 4.2 Experimental Results

Experimental results will be given to show the performance of our proposed method in segmenting the perspective effects in an image. Images synthesized from Brodatz's album [3] and extracted from the MIT VisTex database [20] were used in the experiments. All the images were  $256 \times 256$  in size with 256 gray levels. The number of octaves, the number of voices per octave, and the number of rotations used in CWT were fixed to 4, 8, and 4, respectively.

Fig. 2(a) is an image composed of two inclined brick walls (D95 in Brodatz) with different surface orientations. One can realize the need for segmenting the perspective effects since the two inclined brick walls have the same texture and, except by means of the perspective effects, these two walls can not be delaminated at all. The ridge surface is shown in Fig. 2(b). The vote distribution of the estimated local surface orientations is shown in Fig. 2(c). The parameter space was classified as shown in Fig. 2(d). In Fig. 2(e), the coarse segmentation result is shown, where the black regions indicate unclassified areas. The fine segmentation

result is given in Fig. 2(f). We have superimposed the segmentation boundaries on the original image. The results of several experiments on synthesized images from Brodatz’s album using our algorithm are given: Fig. 6(a) is composed of two inclined fench canvases (D20 in Brodatz). The corresponding tilt and slant angles of the canvases are different. Like the previous brick wall example, the only feature that can separate them is the surface orientations. Fig. 6(b) is composed of inclined canes (D101 and D102 in Brodatz) with different slant angles but the same tilt angle. Fig. 6(c) is composed of inclined hexholes and D84 in Brodatz. They have the same slant angle but different tilt angles. In all the cases, the boundaries of the segmentation results were superimposed on the original images. One can see from the boundaries in (c) that our method produced no blocky effects. In autonomous navigation, we usually have to decide the geometry of corners which are composed of more than two surface planes. In Fig. 6(d) shows the segmentation result of a synthesized corner image. Again, the segmentation boundaries are superimposed on the original image. The image is formed by three inclined textures, D52, D53, and D34 in Brodatz, with different surface planes.

We further tested the performance of our method using real-world images. Figs. 7(a)~(c) show real-world images, each composed of one main textured region and a background. The segmentation results reveal that the ladders and the building were successfully extracted whereas the other parts with less pronounced perspective effects were regarded as background. It is important to note that in (a) and (b), the areas with less pronounced perspective effects are relatively fine-grained, so that CWT is less efficient for characterizing them. Finally, Fig. 7(d) shows an indoor image of a typical office environment recorded by an autonomous robot [8]. The segmentation result indicates that three different textured objects were separated. However, there are only two surface planes appeared in the image. The top plane in the figure contains two textures: the bar and the wall. Although these two textures reside in the same plane and, therefore, should have the same perspective parameters, they were still discriminated by our method. This is due to the fact that only the ridge points corresponding to the bar (and the surface orientations) were detected. The ridge points of the wall were unable to obtain (no surface orientations). Thus, our algorithm discriminated the surface orientations of the bar from no surface orientations of the wall, and was unable to discover that the two textures are

resided in the same surface plane. This example gives the constraints on the performance of our method.

Finally, we will give a comment on the computation time of our algorithm. We will discuss exclusively the complexity in time of some parts in our algorithm. The wavelet transform with orientation totally requires 16 Fourier transforms since we have used 4 octaves and 8 voices and 4 orientations. We allow at most 2 ridge surfaces a position in an image can belong to in order to reduce the computing time for finding the multiple ridge surfaces. The estimation time for the local surface orientations is significantly reduced by RANSAC, corresponding to a 65 times speed-up than that without using RANSAC. So far, our algorithm can not be implemented in real-time. Usually we have to wait for the results for performing the wavelet transform and to segmentation of a textured image.

## 5 Conclusions

A common assumption of the shape from texture problem is that a perceived image mainly contains only one type of texture with the same surface orientations. So, only one perspective parameters is to be estimated. Unfortunately, a natural image is often composed of textures with different surface orientations. Thus, we have many perspective parameters needed to be estimated simultaneously. We have proposed an approach to this problem. Our method is effective in segmenting the planes where textures in the planes contains dominant frequency components. We have shown the performance of our method for synthesized as well as natural images. Our algorithm produces no blocky effects along the segmentation boundaries. Several effects may lead to failure when our method is used, such as modeling noise from less regular textures, a small perspective region which produces an insufficient data for clustering, and too many clusters in an image. These problems require further study.

## References

- [1] J. P. Antoine, P. Carrette, R. Murenzi, and B. Piette, "Image Analysis with Two-

- Dimensional Continuous Wavelet Transform”, *Signal Processing*, Vol. 31, 1993, pp. 241-272.
- [2] D. Blostein and N. Ahuja, “Shape from texture: Integrating surface element extraction and surface estimation”, *IEEE Trans. Pattern Anal. and Machine Intell.*, Vol. 11, pp. 1233-1251, 1989.
- [3] P. Brodatz, “Textures: A Photographic Album for Artists and Designers”, New York: Dover, 1966.
- [4] R. Carmona, W.L.Hwang, B. Torr sani, “Characterization of signals by the ridges of their wavelet transform”, *IEEE Trans. on Signal Processing.*, 1997.
- [5] N. Delprat *et al.*, “Asymptotic wavelet and Gabor analysis: Extraction of instantaneous frequencies”, *IEEE Trans. Inform. Theory*, Vol. 38, No. 2, 1992, pp. 644-664.
- [6] M.A. Fischler and R. C. Bolles, “Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography”, *Communications of ACM*, Vol.24, 1981, pp.381-395.
- [7] W. L. Grimson and D. P. Huttenlocher, “On the Sensitivity of the Hough Transform for Object Recognition”, *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 12, 1990.
- [8] T. Hofmann and J. M. Buhmann, “Pairwise data clustering by deterministic annealing”, *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 19, pp. 1-14, 1997 (<http://www-dbv.cs.uni-bonn.de/image/segmentation.html>).
- [9] W. L. Hwang, C. S. Lu and P. C. Chung, “Shape From Texture: Estimation of Planar Surface Orientations Using Continuous Wavelet Transform”, *IEEE Trans. on Image Processing*, Vol. 7, No. 5, pp. 773-780, 1998.
- [10] J. M. Jolion, P. Meer, and S. Bataouche, “Robust Clustering with Applications in Computer Vision”, *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 13, No. 8, pp. 791-802, 1991.

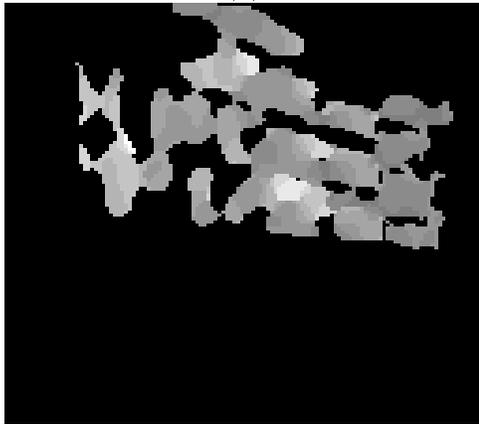
- [11] J. Krumm and S. Shafer, "Segmenting textured 3D surfaces using the space/frequency representation", *Spatial Vision*, Vol. 8, No. 2, 1994, pp. 281-308.
- [12] J. Krumm and S. Shafer, "Texture Segmentation and Shape in The Same Image", *IEEE Conf. on Computer Vision*, pp. 121-127, 1995.
- [13] C. S. Lu, W. L. Hwang, and P. C. Chung, "Segmentation of 3D Textured Image Using Continuous Wavelet Transform", *IEEE Conf. on Image Processing*, Vol. I, pp. 235-238, 1997.
- [14] M. Clerc and S. Mallat, "Shape from Texture and Shading with Wavelets", *Dynamical Systems, Control, Coding, Computer Vision*, Progress in Systems and Control Theory, vol. 25, pp. 393-417, 1999.
- [15] J. Garding, "Shape from Texture for Smooth Curved Surfaces in Perspective Projection", *Journal of Mathematical Imaging and Vision*, pp. 327-350, 1992.
- [16] J. Malik and R. Rosenholtz, "Computing Local Surface Orientation and Shape from Texture for Curved Surfaces", *International Journal of Computer Vision* 23(2), pp. 149-168, 1997.
- [17] B. Super and A. C. Bovik, "Planar surface orientation from texture spatial frequencies", *Pattern Recognition*, Vol. 28, No. 5, pp. 729-743, 1995.
- [18] F. Pla, J. M. Sanchiz, J. A. Marchant, and R. Brivot, "Building Perspective Models to Guide a Row Crop Navigation", *Image and Vision Computing*, Vol. 15, pp 465-473, 1997.
- [19] Y. L. Tang and Rangachar Kasture, "Runway Detection in an Image Sequence", *SPIE*, 1995.
- [20] R. Picard, C. Graczak, S. Mann, J. Watchman, L. Picard and L. Cawpbell, "Vision Texture 1.0(<http://www-white.media.mit.edu/vismodel/imagery/VisionTexture/vistex.html>)", 1995.



(a)



(b)

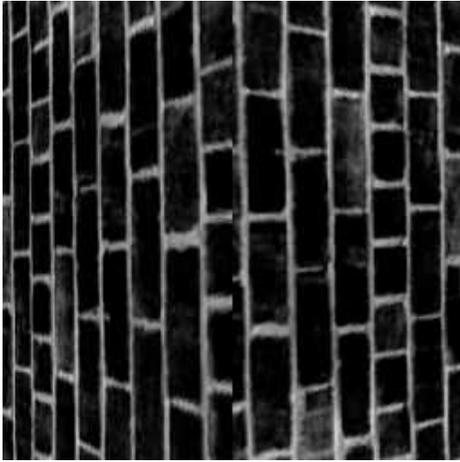


(c)

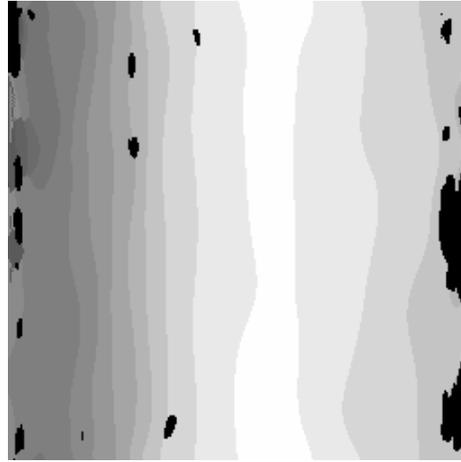


(d)

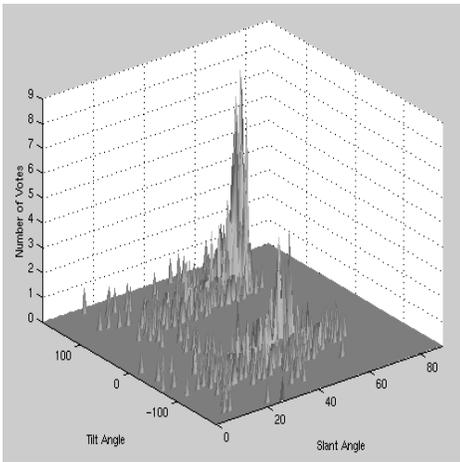
Figure 1: A scene and the scales of its ridges: (a) The scene; (b), (c), and (d) show the scales of the ridge surfaces of the horizontal and the vertical structures of the building, and of the objects in front of it, respectively. The darker gray level indicates smaller scale values. Black pixels are the places outside a ridge surface.



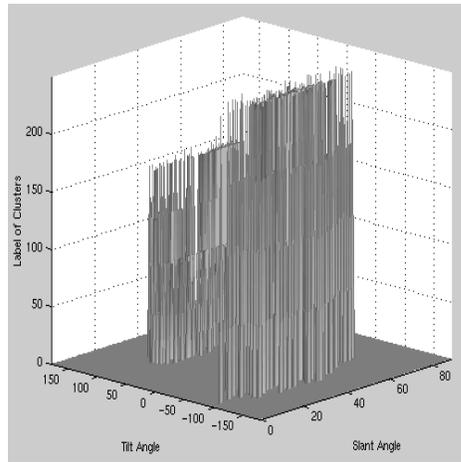
(a) Bricks (D95).



(b) Ridge Surface.



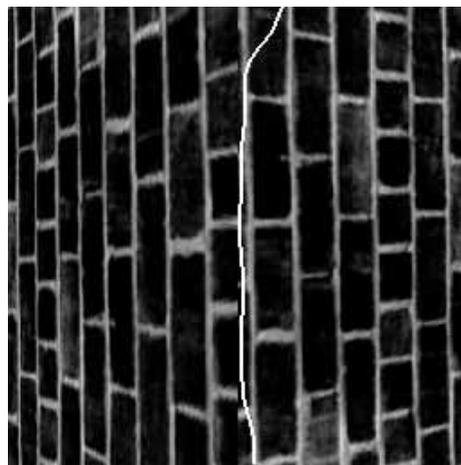
(c) Votes of local surface orientations.



(d) Clusters.



(e) Coarse segmentation result.



(f) Final segmentation result.

Figure 2: Segmentation of two inclined bricks (D95).

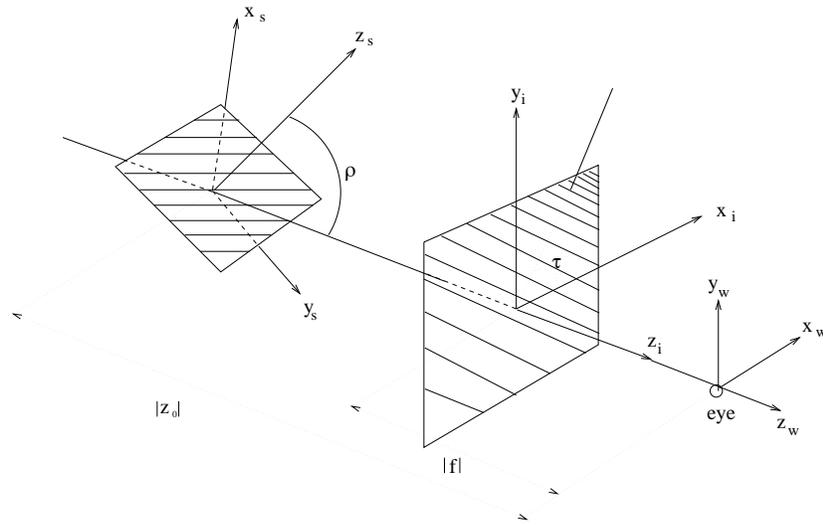


Figure 3: The coordinate relationship between the image and the surface plane.

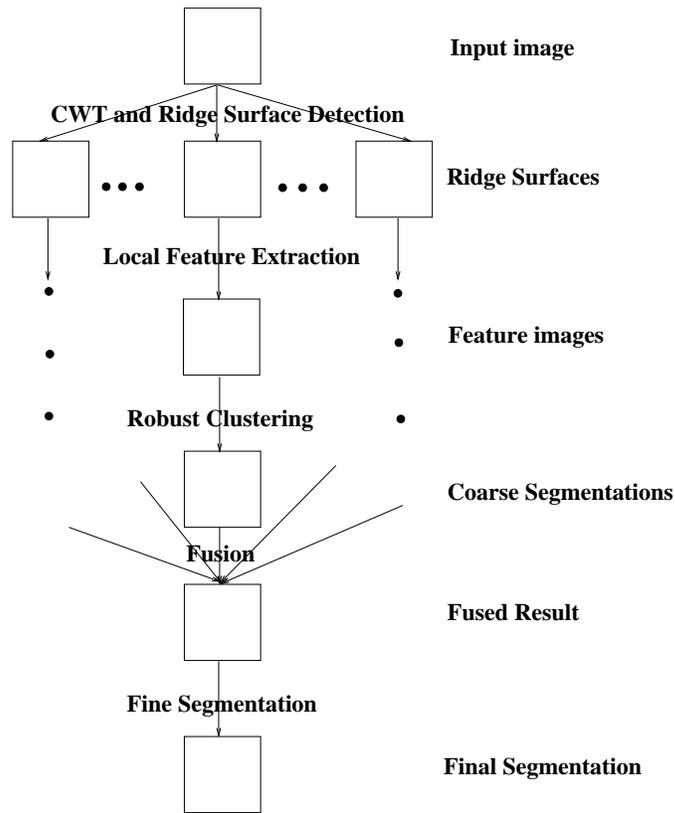


Figure 4: The steps involved in our algorithm.

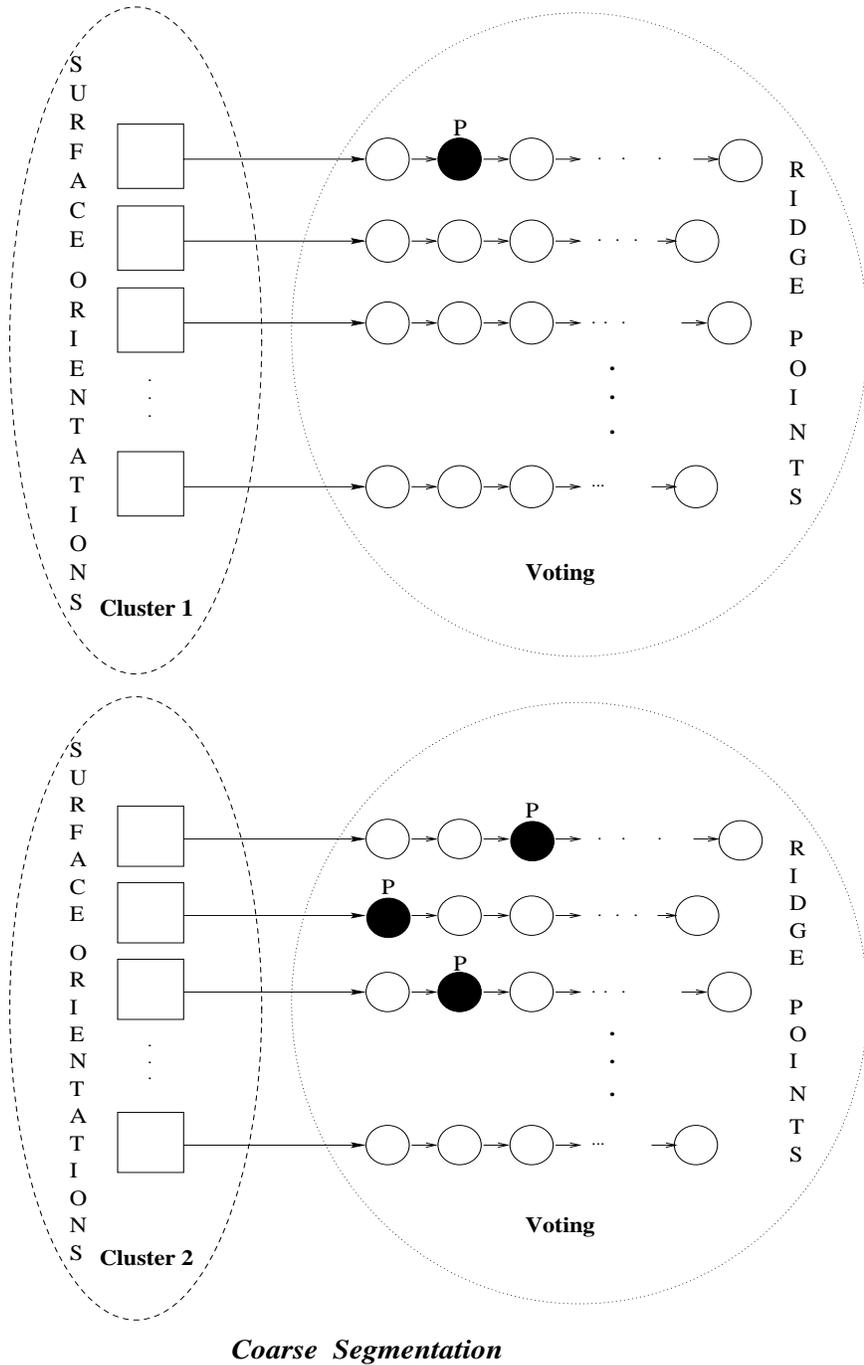
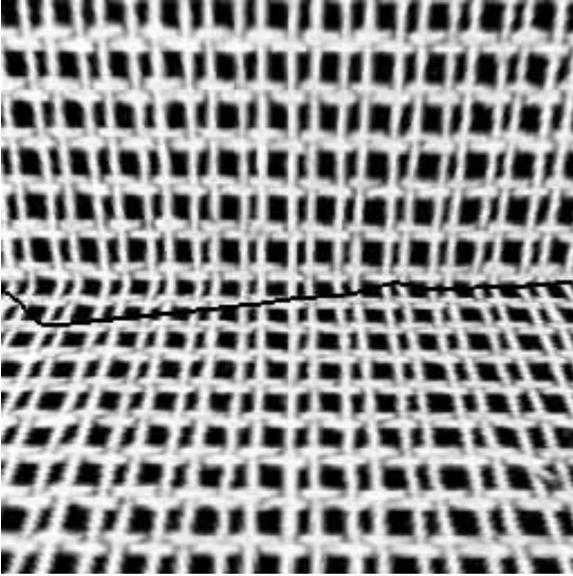
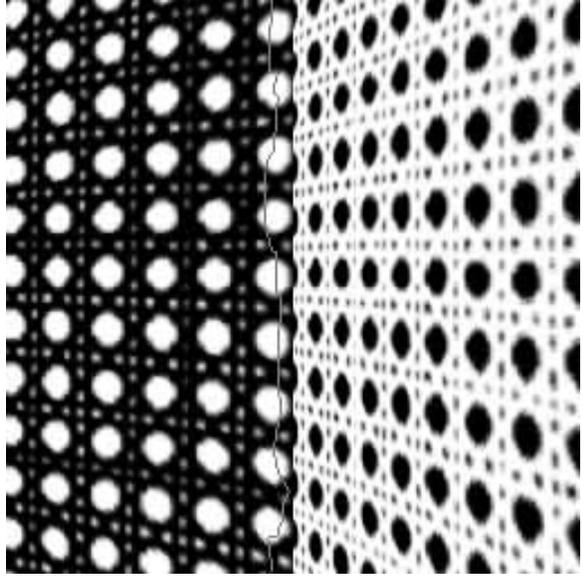


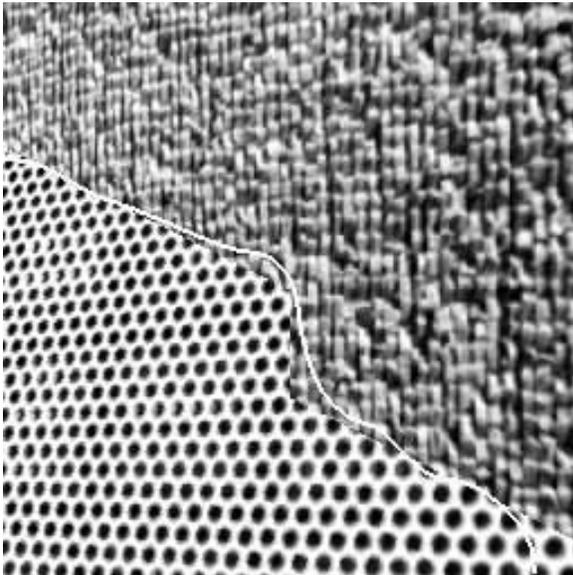
Figure 5: Robust clustering and the voting strategy with random inlier selection.



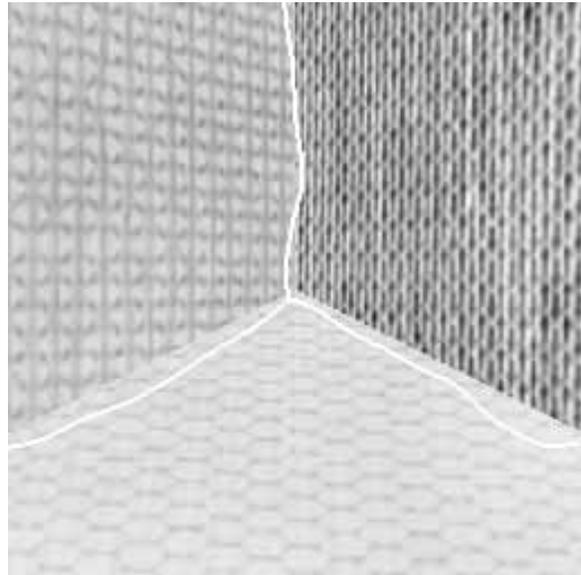
(a) Fenich canvas (D20).



(b) Two canes (D101+D102).



(c) Inclined(raffia+hexholes).



(d) A synthesized corner image (D52+D53+D34).

Figure 6: Segmentation of three inclined mixed Brodatz textures.



(a) Scene-1.



(b) Scene-2.



(c) Scene-3.



(d) Scene-4.

Figure 7: Segmentation results on natural images.