

Planar Orientation from Blur Gradients in a Single Image

Scott McCloskey
Honeywell Labs
Golden Valley, MN, USA
scott.mccloskey@honeywell.com

Michael Langer
McGill University
Montreal, QC, Canada
langer@cim.mcgill.ca

Abstract

We present a focus-based method to recover the orientation of a textured planar surface patch from a single image. The method exploits the relationship between the orientation of equifocal (i.e. uniformly-blurred) contours in the image and the plane's tilt and slant angles. Compared to previous methods that determine planar orientation, we make fewer assumptions about the texture and remove the restriction that images must be acquired through a pinhole aperture. Our method estimates slant and tilt of an image patch in a single image, as compared to depth from defocus methods that require two or more input images. Experiments are performed using a large set of test images.

1. Introduction

The recovery of 3D shape information from 2D intensity images remains a primary objective of computer vision research. Numerous cues have been exploited to recover depth information, many of which mimic those used by the human visual system. Optical blur is one such cue [1, 2, 3]. Computer vision methods that exploit optical blur [4, 5, 6, 7] come in different forms, but most require multiple input images with different lens settings. Given the results of studies [2] that show the human visual system's ability to recover planar orientation from blur gradients in a single monocular presentation, it is somewhat surprising that no comparable capability has been developed in the computer vision literature. Methods that recover planar orientation from a single image, typically in the realm of Shape from Texture, assume that images have been acquired through a pinhole aperture, avoiding the issue of optical blur.

The notion of a pinhole aperture, though conceptually convenient, is problematic in practice. Even when the smallest available aperture is employed, and regardless of the perceptibility of blur in whole-image presentations, significant blur is still present in an image. As in some of our test images, with a 50mm lens at aperture $f/22$, our cam-

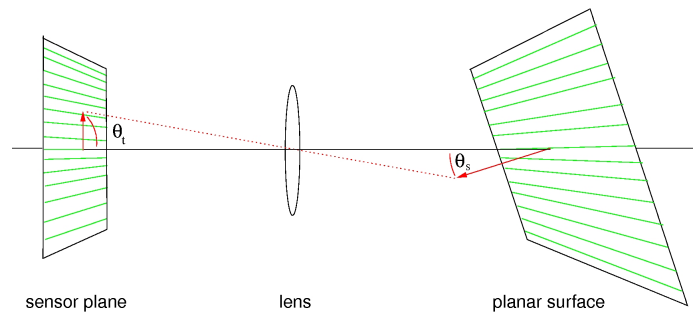


Figure 1. We recover the orientation (tilt θ_t and slope θ_s) of planar objects by finding equifocal contours in an image. Isodepth contours in the scene (green lines on the planar surface) map to equifocal contours (green lines on the sensor plane), and are orthogonal to the tilt direction. The 3D surface normal and its 2D projection on the sensor are shown as red vectors.

era will produce a blur diameter of 5 pixels for objects at 1m when focused at a distance of 0.8m. When focused at infinity, the same lens configuration will produce a blur diameter of more than 7 pixels for an object at 5 meters. Indeed, we will show that a blur cue of this magnitude is sufficient to recover planar orientation when using the smallest available aperture ($f/22$) on a high-end digital camera. Moreover, as the size of individual pixels on digital image sensors continue to shrink in accordance with Moore's law (with increasing spatial resolution), the blur magnitude will increase in pixel terms even if it is fixed in terms of visual angle. Equivalently, the same blur diameter (as measured in pixels) will be observable in smaller aperture images as pixel sizes continue to shrink.

This paper presents, for the first time, a single image method for the recovery of planar orientation using an optical blur cue. In that it makes no assumption about the camera settings - the aperture can be small or large - the method offers improved generality over current planar orientation schemes. The method first recovers planar tilt by finding the direction of maximal blur gradient, and then recovers slant by finding the angle that, when back-projected, produces equifocal contours in both principal directions (see

Figure 1). As will be demonstrated in the experiments of Section 5, our method is applicable to a wide range of textures: anisotropic textures, textures lacking dominant spatial frequencies, and those that consist of repeating patterns.

The method is tested on a large new test set for evaluating planar orientation algorithms over a range of finite apertures. It consists of 1404 camera images of 9 planar textures at 26 carefully-controlled orientations, each photographed with 6 different apertures. Our method produces an average error in estimated tilt of 2.8° over the entire test set, and below 1.5° on images captured with aperture $f/22$, which represents improved performance as compared to existing methods [8, 9] that assume a pinhole aperture. Average error in estimated slant is 5.4° , which is comparable to the state of the art in planar orientation. We also test the method against several natural images, producing results that successfully capture the 3D structure of the scene.

The method makes two assumptions about the input images. First, the texture pattern is assumed to be homogeneous. Second, the blur gradient is assumed to be linear across the image patch. This second assumption holds automatically when the surface is planar, provided that no point in the patch interior lies on the focal plane. This second assumption is common to methods that recover depth from defocus, and is necessitated by the inherent ambiguity that points on either side of the focused depth exhibit equal amounts of blur (see Figure 2). The homogeneity assumption is a common one for methods that recover planar orientation by shape from texture, though some methods make the more restrictive assumption that the texture is isotropic. The need for a homogeneity assumption arises in our problem, since changes in the contrast of the texture would be confounded with a change in blur. In addition to these assumptions on the input images we, like most depth from defocus methods, presume access to camera information embedded in a JPEG images’s EXIF header - namely the aperture, focal length, and focused distance.

2. Relation to Previous Work

2.1. Depth from Defocus

Depth from defocus (DFD) methods exploit the relationship between image blur and a 3D point’s depth relative to the focused distance. These methods require at least two images taken with different lens settings (aperture or focal position). Several older methods [4, 10] for DFD measure the change in some sharpness metric at each image location, and relate that change to depth through the known change in lens settings. More recent algorithms [5, 6] for DFD reconstruct both the scene’s radiance and 3D structure by minimizing a regularized energy function comparing the input images to a rendered appearance of the current estimate. Like our method, these DFD methods require that objects

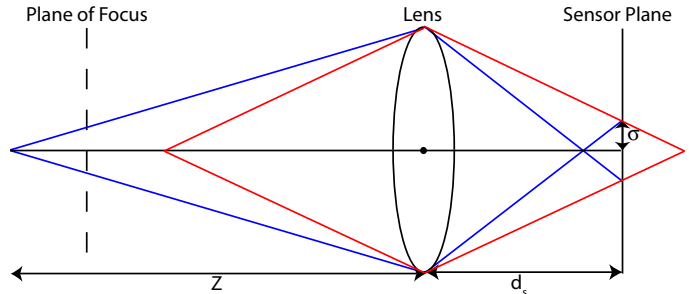


Figure 2. Points located at depths other than that of the plane of focus will appear blurred in the image. There is an ambiguity, as points on either side of the plane of focus can produce the same blur radius σ (red and blue ray-tracings). Like other defocus methods, we resolve this ambiguity by assuming that all surface points within a patch are to one side of the plane of focus.

in the scene lie either entirely in front of or behind the focused distance.

To date, the only methods that have recovered any depth information from blur in a *single* image have been presented by Pentland [4] and Subbarao and Gurumoorthy [11]. Both estimate depth at step edges between regions of uniform radiance by measuring the width of the blurred edges in the image, but are unable to recover 3D information in other regions of the image. Though different in application than the method presented in this paper, Okatani and Deguchi [12] have also recently presented a novel method for the estimation of scene scale from defocus cues in a single image. That method requires a camera with a tilt lens.

2.2. Planar Orientation by Shape from Texture

Shape from texture methods recover the 3D structure of scenes using two cues: the *distance effect*, where parts of the scene further from the camera appear smaller, and the *foreshortening effect*, where parts of the surface with high slant appear compressed in the tilt direction (e.g. the green equifocal contours appearing closer together at greater distances in Figure 1). Though shape from texture methods have been developed to recover 3D information about general scenes [13, 14], there are several that have been exclusively designed to recover either the orthographic or projective mapping of a planar surface. Methods such as Brown and Shvaytser [15] measure the deviation of the image’s texture from isotropy in order to estimate the plane’s orientation, assuming isotropic surface textures. Many other methods such as Super and Bovik’s [16, 9] assume the texture map is a regular pattern – the extreme case is that the texture is dominated by a single spatial frequency. The method of Galasso and Lasenby [17] relaxes this assumption slightly, allowing for images with a small number of local spatial frequencies. The recent method of Farid and Kosecka [8] avoids any assumption about local spatial fre-

quencies, but makes the assumption of random phase, precluding textures that have structured patterns. As mentioned in the introduction, all shape from texture methods assume a pinhole aperture.

Another related approach for a different application is presented by Aggarwal and Ahuja [18], who propose a method to estimate the orientation of a camera’s sensor by measuring defocus over a known target composed of repeating texture elements.

3. Modelling the Blur Gradient

Consider an image patch in which the visible surface is a plane of depth $Z = Z_0 + AX + BY$, where X, Y , and Z define the camera’s coordinate frame. Assume the image is formed by a thin lens of focal length f . Let d_s be the distance from the sensor plane to the lens. Dividing the equation of the plane by Z , multiplying by d_s and rearranging, we obtain

$$\frac{1}{Z} = \frac{1}{Z_0} - \frac{A}{d_s Z_0} x - \frac{B}{d_s Z_0} y \quad (1)$$

where $x = \frac{d_s X}{Z}$ and $y = \frac{d_s Y}{Z}$ are pixel positions.

The goal of planar orientation algorithms is to accurately estimate the slant and tilt of a 3D plane. As illustrated in Figure 1, the slant $\theta_s \in [0, \frac{\pi}{2}]$ is defined as the angle between the optical axis and the plane’s 3D surface normal. The tilt $\theta_t \in [0, 2\pi]$ is defined as the orientation of the 2D projection of the plane’s surface normal on the sensor. The slant and tilt are the same at all positions in the image patch, with $\tan(\theta_s) = \|\nabla Z\| = \sqrt{A^2 + B^2}$ and $(\cos \theta_t, \sin \theta_t) = (\frac{\partial Z}{\partial X}, \frac{\partial Z}{\partial Y}) = (A, B)$.

From the thin lens equation, given the camera’s aperture F (f-number), focal length f , and sensor distance d_s , the blur radius σ is a linear function of inverse depth $\frac{1}{Z}$,

$$\sigma = \frac{f}{2F} \left(d_s \left(\frac{1}{f} - \frac{1}{Z} \right) - 1 \right). \quad (2)$$

Substituting Eq. (1) into (2), we see that blur radius is a linear function of image position (x, y) . In particular, the blur gradient is constant across the image of the planar patch,

$$\left(\frac{\partial \sigma}{\partial x}, \frac{\partial \sigma}{\partial y} \right) = \frac{f}{2F Z_0} (A, B). \quad (3)$$

Our method estimates this constant blur gradient vector and, from the known¹ values of f, F , and Z_0 , we obtain the surface slant and tilt from A, B .

¹Most digital images files contain the camera intrinsics and the distance to the focal plane in the EXIF header data.

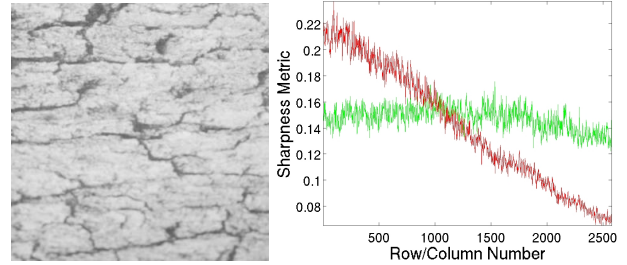


Figure 3. (Left) The bark texture at orientation $(\theta_t, \theta_s) = (\frac{3\pi}{2}, \frac{\pi}{4})$. Not that the resolution in the figure is ~ 1000 pixel per cm, so blur cannot be seen here. (Right) Sharpness measured along rows (red) and columns (green).

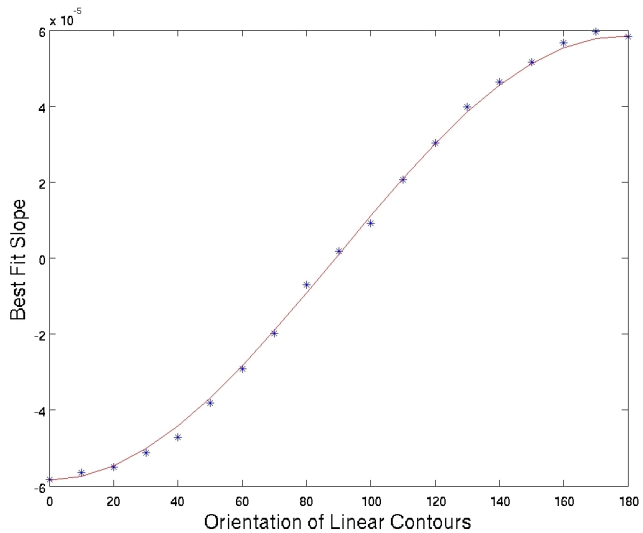


Figure 4. The average change in sharpness between adjacent, parallel linear contours of different orientations (slope of plots in Fig. 3 (right)). Blue points show measured data. The red curve is the model fit to those observations.

4. Planar Orientation Estimation Algorithm

We begin by removing the intensity non-linearities introduced by the camera’s tonescale function [19]. Next, as image blur is best observed in the middle to high spatial frequencies, we remove low frequencies from the image, namely we subtract a low-pass filtered version of the image from itself. The low pass filter is a Gaussian with standard deviation 3 pixels². As we will be comparing the blur along different lines in an image, we define a simple *sharpness measure* along a given line to be the standard deviation of the residual image [20].

4.1. Estimating Tilt

The key to understanding our algorithm is the concept of an equifocal contour - a contour along which the amount

²Experiments have shown the method to be insensitive to this value within a range [2,4].

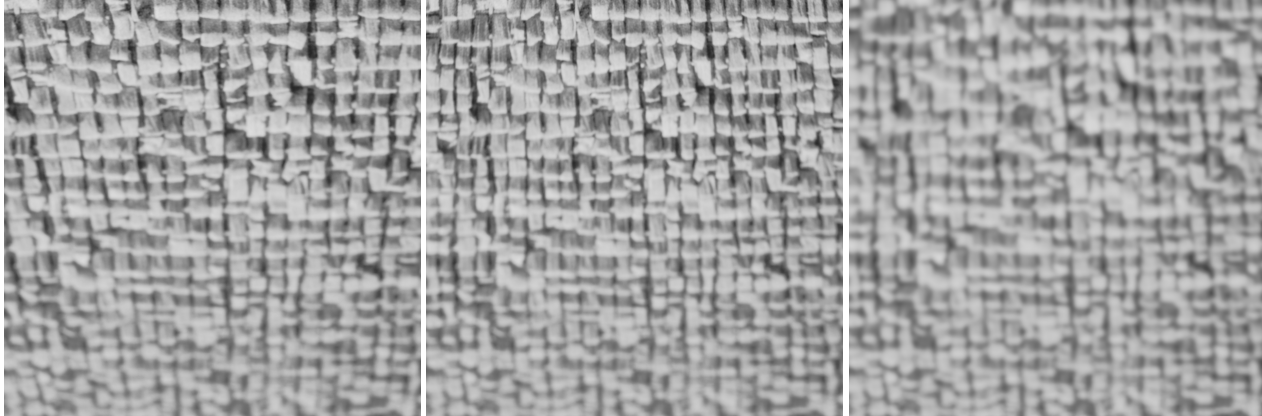


Figure 5. Steps of the slant estimation method. (Left) Image in the canonical $\theta_t = \frac{3\pi}{2}$ view. (Center) Image after accounting for changes in magnification consistent with the hypothesized slant angle $\hat{\theta}_s = \frac{\pi}{4}$. (Right) Image after accounting for both magnification and optical blur. The hypothesized angle is taken as the slant estimate because the columns of the right image are determined to be equifocal.

of optical blur remains constant. Figure 3 (left) shows an image with tilt $\theta_t = \frac{3\pi}{2}$ and slant $\theta_s = \frac{\pi}{4}$, acquired with the focused distance at the same depth as the first row in the image. Due to its tilt, rows of the image are equifocal and depict isodepth contours of the plane, whereas columns of the image cut orthogonal to the equifocal direction. Figure 3 (right) shows a plot of the sharpness measured along rows (red) and columns (green). As expected, there is a clear overall gradient in the sharpness across rows, but not across columns. Our method for finding surface tilt searches for the direction in which the sharpness gradient is maximized, as we now explain.

Let θ_m be the direction of the sharpness gradient, which is orthogonal to the orientation of parallel lines with the highest line-to-line change in sharpness. This serves as our estimate of the direction of the blur gradient, which is also the tilt. For any orientation θ , the line-to-line change in sharpness $s(\theta)$ is proportional to $\cos(\theta - \theta_m)$ (see Fig. 4). To estimate tilt, then, we perform a least squares fit of a small set of the measured $s(\theta_d)$ at orientations $\theta_d \in \Theta$ to a two parameter model

$$s(\theta) = \alpha \cos(\theta - \theta_m). \quad (4)$$

In our experiments we sample s at twelve values of θ .

4.2. Estimating Slant

Like the algorithm for finding the plane's tilt, our slant estimation method searches for an angle that best explains the blur gradient. However, whereas the tilt estimation attempted to find the direction of the sharpness gradient the slant estimation method finds the angle θ_s that removes this sharpness gradient. Slant is estimated as the angle whose back-projection (image plane to surface plane) produces the smallest gradient in the sharpness measure in the direction of former depth variation.

Before searching for the slant angle, we rotate the image into a canonical $\theta_t = \frac{3\pi}{2}$ view, as in Figure 5 (left), using the computed tilt estimate from Sec. 4.1. For each candidate slant angle $\hat{\theta}_s$, we would expect a particular blur gradient magnitude, according to Eq. 3.

The basic idea for estimating the blur gradient magnitude and its corresponding slant is to apply a blur gradient that is opposite to the inherent optical blur, such that we obtain a uniformly blurred image. (Below we refer to this as a “doubly” blurred image.) In particular, for the correct θ_s we would expect the doubly blurred image to have no sharpness gradient. This suggests that we compute a set of doubly blurred images for candidate values $\hat{\theta}_s$, and then choose the one that minimizes the sharpness gradient.

In addition to applying this blur gradient, however, one must also consider blur artifacts from perspective-induced size changes of the surface texture. If a pinhole image of a slanted plane were to be uniformly blurred, this uniformly blurred image would still have a sharpness gradient due to the changing scale of texture elements. Since texture elements at different depths would be different sizes (and hence different spatial frequency components), the sharpness measure would be non-uniform in the direction of the tilt. This would occur in the case of a checkerboard texture, for example.

To avoid this confound of focus and size (the latter being a shape from texture cue only), we first rectify the image using an appropriate homography

$$(x, y) \rightarrow \left(\frac{x}{(d_s - y \tan \theta_s)}, \frac{y}{\cos(\theta_s)(d_s - y \tan \theta_s)} \right)$$

for the candidate $\hat{\theta}_s$ value (see Fig 5 center). We then blur the rectified image according to the blur gradient that is predicted by $\hat{\theta}_s$. Each such rectified and doubly blurred image is the rendered appearance of the planar surface as if it were

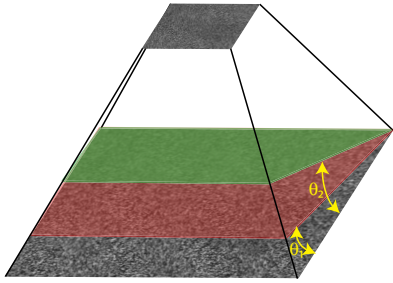


Figure 6. The rectified and doubly blurred images represent different slices through the scale space defined by the input image. The green and red planes illustrate two such slices for candidate slant angles θ_1 and θ_2 .

rotated by $\hat{\theta}_s$ about the 3D line corresponding to the last (most distant) row of the image. Equivalently, each image corresponds to a slice of the scale space of the input image, as illustrated in Fig. 6. For the correct slant $\hat{\theta}_s = \theta_s$, one obtains a uniformly blurred (doubly blurred) and geometrically rectified image (see Fig. 5 right) which now has no sharpness gradient.

As currently implemented, the method computes the doubly blurred image for ascending values of $\hat{\theta}_s$ in increments of 2° . Slant estimation is terminated when the direction of the sharpness gradient is reversed, which occurs when $\hat{\theta}_s > \theta_s$. Because the complexity of computing the doubly blurred image increases dramatically with $\hat{\theta}_s$, this has a considerable impact on overall complexity.

5. Test Data and Experimental Results

5.1. Test Image Set

To our knowledge, no existing calibrated data is available for testing our algorithm.³ We thus had to construct a large, new, public test image set⁴. The test set consists of 1404 images: all combinations of 9 textures (a noise texture, plus prints of the Brodatz textures *grass*, *raffia*, *pigskin*, *bubbles*, *bark*, *leather*, *weave*, and *sand*; see Figure 7), 6 apertures ($F = 22, 16, 11, 8, 5.6, 4$), and 26 planar orientations (see Table 1). The included textures span a wide range of characteristics. *Raffia* has a dominant spatial frequency whereas the noise texture has a uniform power spectrum. *Grass* has random phase whereas *weave* has repeating texture elements. *Sand* is roughly isotropic, whereas *bark* has a directional bias. The chosen range of apertures spans 6 stops of exposure, from a near pinhole to an aperture that

³For example, the CURET database [21] is inappropriate. CURET surface samples were 10×10 cm and viewed from a distance of $Z_0 = 200$ cm by a low resolution camera, resulting *by design* in a very low blur gradient.

⁴<http://www.cim.mcgill.ca/~scott/texturedPlanes/>

$\theta_s \setminus \theta_t$	-90°	-80°	-70°	-60°	-50°	-40°
30°	X	X	X	X	X	X
35°	X	X	X	X	X	X
40°	X	X	X	X	X	X
45°	X	X	X	X	X	
50°	X	X	X			

Table 1. Orientations included in the image test set.

offers very limited depth of field. The sampled tilt angles can span the entire range of values when coupled with image rotations of 90° , 180° , and 270° , which can be achieved without re-sampling intensity values. The chosen slant angles are sampled evenly between $30 - 50$ and degrees in 5 degree increments.

The test set consists of images acquired with a 10MP Nikon D80 DSLR camera with a 50mm focal length lens. Prints of the texture patterns were affixed to a planar stage attached to a professional tripod with 1° markings on the degrees of freedom. After carefully aligning the camera to the tripod, the planar orientations were set by hand while the camera was operated remotely to avoid altering its pose.

5.2. Orientation Estimation Results

The results of the tilt and slant estimation method are shown in Table 2, including the overall accuracy and the errors along the different testing parameters (texture, slant, and aperture). The tilt was estimated by the method described in Section 4.1, and measurements were taken over orientations in increments of 15° (i.e. only six image rotations were performed). The results indicate that the method is quite successful despite sparse sampling of orientations. Somewhat surprisingly, the tilt estimation errors are smaller for larger F-numbers.

The results of our slant estimation are also shown in Table 2. Due to the relatively longer execution time, we only search for θ_s in increments of 2° . Our tilt errors on $F = 22$ imagery average 1.3° , as compared to 3.7° in [9]. Our slant errors average 5.4° which, while worse than the 3.7° of [9] or 3° of [8], include textures on which the existing algorithms cannot work. The error of our slant estimates are about double that of the tilt estimates. This may be due in part to a cascading of the error from the tilt estimation, as the slant estimation is designed to operate on an image with a known tilt. The columns in Table 2 clarify this effect by giving the slant errors based on our estimated tilt, as well as the errors based on the use of ground truth tilt in its place. Within either column, however, we see optimum performance for mid-sized apertures ($F = 8$ or 5.6). Poorer performance results from the use of a small aperture due to the lack of a substantial blur gradient, whereas poor performance on large aperture images results from a similar prob-

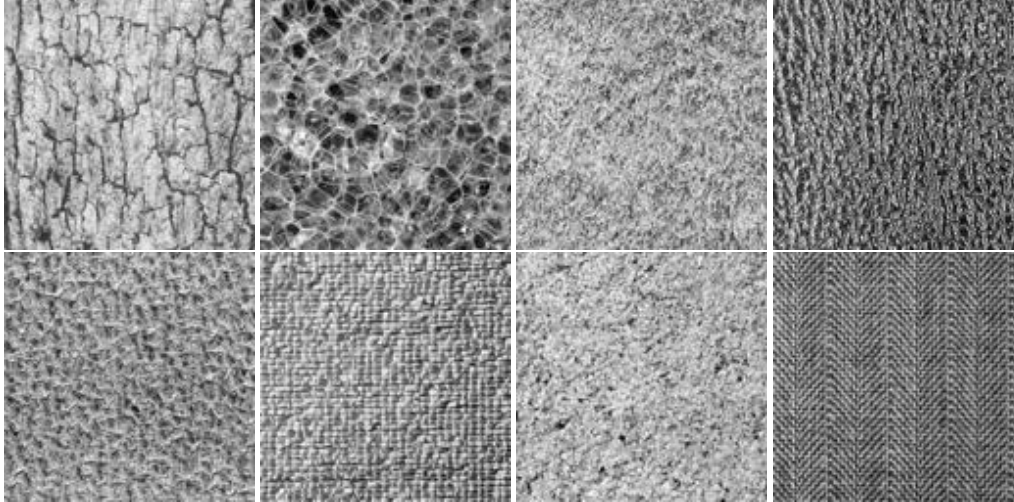


Figure 7. The Brodatz textures used in our experiments. In raster order: bark, bubbles, grass, leather, pigskin, raffia, sand, and weave.

		tilt	slant	slant from true tilt
overall		2.8	5.4	4.8
by texture	noise	2.7	6.2	6.0
	leather	3.2	6.8	3.4
	pigskin	3.7	7.5	7.4
	sand	2.8	3.9	3.9
	bubbles	1.5	4.0	3.1
	bark	3.8	4.2	4.0
	weave	2.7	4.5	4.7
	raffia	2.7	5.0	4.3
	grass	3.6	6.8	6.9
by aperture	$F = 22$	1.3	6.7	6.4
	16	2.0	5.6	5.5
	11	3.2	5.3	4.7
	8	3.7	4.2	3.2
	5.6	4.0	4.4	3.6
by slant	4	9.8	6.2	5.8
	30°	2.3	5.9	5.4
	35°	2.9	4.5	4.1
	40°	2.6	3.9	3.5
	45°	4.1	7.3	6.5
	50°	3.1	6.1	5.1

Table 2. Average absolute tilt and slant errors (in degrees). Because errors from tilt estimation cascade into slant estimation, the final column gives errors for slant estimation using the ground truth tilt.

lem due to the fact that the blurring necessary to compute the unslanted image (as in Figure 5 (Right)) is quite severe, leaving little power in the middle to high spatial frequencies used in the sharpness measure.

Though the images in our test set are captured at 10MP, the number of pixels that we actually consider in our ori-

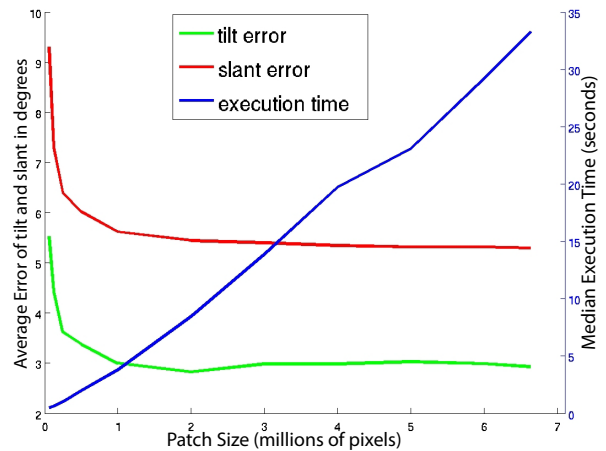


Figure 8. Slant and tilt estimation errors, along with the corresponding median execution time, as a function of patch size.

entation estimation scheme is about 6.5MP (roughly speaking, the pixels contained within the largest circle that can be inscribed in the image frame). The method is currently implemented in Matlab, and the execution time depends significantly on the parameters (particularly F and Z_0) and the plane's slant. Over our 1404 image evaluation, the median execution time was 34 seconds and 90% of the execution times were under 2 minutes. The remaining 10% had wide apertures (low F) and high slants. Because each of the doubly blurred images can be computed in parallel, an implementation that takes advantage of GPUs or multi-core processors would easily achieve significant improvements in speed.

5.3. Experiments with Image Size

Though the images captured for our test set are quite large, such high-resolution imagery is not necessary to get

good estimates of slant and tilt from our algorithm. In order to understand the algorithm’s performance on lower resolution imagery, we have conducted an experiment to measure the errors when using significantly fewer pixels. Using the same test image set, we reduce the image resolution (using Matlab’s `imresize` function) to a series of smaller values. At each test resolution, we run find the average slant and tilt errors (in degrees), as well as the corresponding execution times. The reduced-resolution imagery notionally corresponds to the imagery that one would get if the camera’s sensor were replaced by one of the same size, but with significantly fewer pixels. The results of this experiment are shown in Fig. 8, from which we see that the algorithm’s performance degrades very gradually from full resolution until a patch size of 1MP despite a significant reduction in the median execution time. Below the 1MP boundary, however, the errors grow significantly.

5.4. Experiments with Natural Images

Though the use of our new test set is useful for computing the quantitative errors produced by our algorithm, it is also interesting to consider its performance on natural images. While few natural images consist entirely of slanted, homogeneously-textured planar surfaces, such surfaces are common *components* of many natural images - particularly urban scenes with significant man-made structures. Moreover, because our test set images are of printed textures, the surface lacks depth variations that are common in more real world textures. To assess the performance of our method against such situations, we have developed a simple interface that allows a user to select planar, textured patches of images for orientation estimation. Some of the results are shown in Figure 9. User-identified planar regions are denoted by red rectangles, and the estimated orientation is illustrated by a vector indicating the direction of tilt, as well as a disc whose minor axis is foreshortened in proportion to the surface’s slant.

The staircase image (Figure 9 (row 1)) is of particular interest because its regular structure gives a rough indication of ground truth. It consists of three planar surfaces which meet at right angles to one another. As such, the left and right planar regions should have identical orientations, whereas the tilt of the middle region should be exactly opposite. The slants of the middle and left/right planes should sum to 90° , because they meet at a right angle. Indeed, our method estimates orientations that closely match these expectations. The estimated orientations (θ_t, θ_s) of these surfaces are $(359^\circ, 28^\circ)$, $(186^\circ, 66^\circ)$, and $(359^\circ, 26^\circ)$ for the left, center, and right patches, respectively. These results are achieved in spite of non-uniform illumination (in the left plane) and non-homogeneous regions of texture (the dark spots in the right plane).

The brick image (Figure 9 (row 2)) depicts a similar sit-

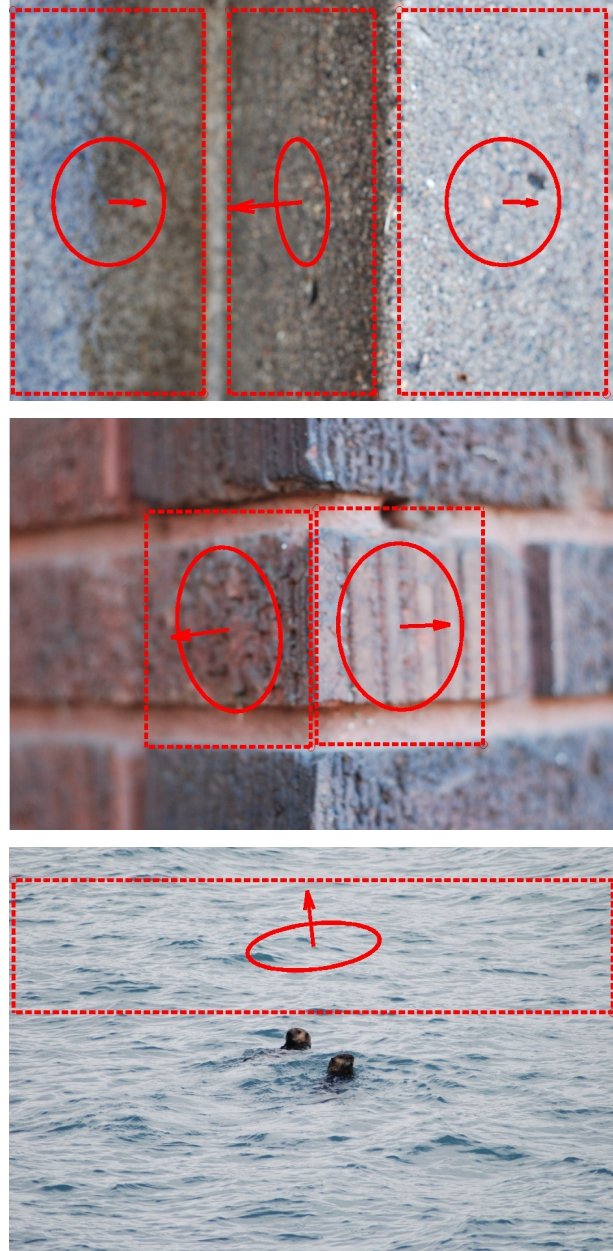


Figure 9. Orientation estimation results on natural images. User-selected bounding boxes (red dashed lines) denote planar surfaces on which we performed orientation estimation. Vectors indicate the tilt direction, and discs are foreshortened in proportion to the surface slant.

uation. In this case, the orientations are estimated from significantly smaller image regions (about 1MP) and the surface has significant depth variations. Again, the tilt estimates are nearly opposed at 182° and 2° , and the slants of 52° and 42° for the left and right planes, respectively, sum to nearly 90° .

Though the structure of the final image does not immediately suggest any way to verify the accuracy of our esti-

mates, they seem to agree with the authors' perception of scene layout.

6. Conclusion

We have presented a novel method for the recovery of planar orientation from a single image using blur cues. For the first time, we have exploited the relationship between equifocal contours in the image and a plane's slant and tilt angles. The accuracy of this method, which was tested against a large set of test images, represents an improvement in performance as compared to state of the art shape-from-texture methods despite our making minimal assumptions about the texture *and* removing the assumption that the image be acquired with a pinhole aperture. Though related to depth from defocus methods that require multiple input images, our method produces results from a single image.

It remains to be seen whether the accuracy of this method may be further improved by incorporating traditional shape from texture cues *without* making additional assumptions about the texture or aperture. With or without additional cues, we intend to explore the ability of this method to recover local orientation from smaller patches of images containing objects with greater complexity than those considered in our tests.

References

- [1] Mather, G., Smith, D.: Blur discrimination and its relation to blur-mediated depth perception. *Perception* **31** (2002) 1211–1219
- [2] Watt, S., Akeley, K., Ernst, M., Banks, M.: Focus cues affect perceived depth. *Journal of Vision* **5** (2005) 834–862
- [3] Mather, G., Smith, D.: Depth cue integration: Stereopsis and image blur. *Vision Research* **40** (2000) 3501–3506
- [4] Pentland, A.P.: A new sense for depth of field. *IEEE Trans. Pattern Analysis and Machine Intelligence* **9** (1987) 523–531
- [5] Chaudhuri, S., Rajagopalan, A.N.: *Depth from Defocus: A Real Aperture Imaging Approach*. Springer-Verlag (1998)
- [6] Favaro, P., Soatto, S.: Seeing beyond occlusions (and other marvels of a finite lens aperture). In: *IEEE Conference on Computer Vision and Pattern Recognition*. (2003) II: 579–586
- [7] McCloskey, S., Langer, M.S., Siddiqi, K.: Automated removal of partial occlusion blur. In: *Asian Conference on Computer Vision*. (2007) I: 271–281
- [8] Farid, H., Kosecka, J.: Estimating planar surface orientation using bispectral analysis. *IEEE Trans. Image Processing* **8** (2007) 2154–2160
- [9] Super, B.J., Bovik, A.C.: Planar surface orientation from texture spatial frequencies. *Pattern Recognition* **28** (1995) 729–743
- [10] Watanabe, M., Nayar, S.: Minimal operator set for passive depth from defocus. In: *IEEE Conference on Computer Vision and Pattern Recognition*. (1996) 431–438
- [11] Subbarao, M., Gurumoorthy, N.: Depth recovery from blurred edges. In: *IEEE Conference on Computer Vision and Pattern Recognition*. (1988) 498–503
- [12] Okatani, T., Deguchi, K.: Estimating scale of a scene from a single image based on defocus blur and scene geometry. In: *IEEE Conference on Computer Vision and Pattern Recognition*. (2007) 1–8
- [13] Malik, J., Rosenholtz, R.: Computing local surface orientation and shape from texture for curved surfaces. *International Journal of Computer Vision* **23** (1997) 149–168
- [14] Witkin, A.: Recovering surface shape and orientation from texture. *AI* **17** (1981) 17–45
- [15] Brown, L.G., Shvayster, H.: Surface orientation from projective foreshortening of isotropic texture autocorrelation. *IEEE Trans. Pattern Analysis and Machine Intelligence* **12** (1990) 584–588
- [16] Super, B.J., Bovik, A.C.: Shape from texture using local spectral moments. *IEEE Trans. Pattern Analysis and Machine Intelligence* **17** (1995) 333–343
- [17] Galasso, F., Lasenby, J.: Shape from texture: Fast estimation of planar surface orientation via fourier analysis. In: *British Machine Vision Conference*. (2007)
- [18] Aggarwal, M., Ahuja, N.: Estimating sensor orientation in cameras. In: *IEEE International Conference on Pattern Recognition*. Volume 1. (2000) 1896–1899
- [19] Debevec, P.E., Malik, J.: Recovering high dynamic range radiance maps from photographs. In: *SIGGRAPH*. (1997) 369–378
- [20] Krotkov, E.: Focusing. *International Journal of Computer Vision* **1** (1987) 223–237
- [21] Dana, K., Van-Ginneken, B., Nayar, S., Koenderink, J.: Reflectance and Texture of Real World Surfaces. *ACM Trans. on Graphics* **18** (1999) 1–34