Improved Sub-pixel Stereo Correspondences through Symmetric Refinement

Diego Nehab[1] Szymon Rusinkiewicz[1] James Davis[2]
[1]Princeton University [2]University of California at Santa Cruz

Abstract

Most dense stereo correspondence algorithms start by establishing discrete pixel matches and later refine these matches to sub-pixel precision. Traditional sub-pixel refinement methods attempt to determine the precise location of points, in the secondary image, that correspond to discrete positions in the reference image. We show that this strategy can lead to a systematic bias associated with the violation of the general symmetry of matching cost functions. This bias produces random or coherent noise in the final reconstruction, but can be avoided by refining both image coordinates simultaneously, in a symmetric way. We demonstrate that the symmetric sub-pixel refinement strategy results in more accurate correspondences by avoiding bias while preserving detail.

1. Introduction

The computation of precise sub-pixel stereo correspondences is vital to areas such as 3D scanning and image based modeling and rendering. Most dense stereo correspondence algorithms start by determining discrete pixel matches and later refine these matches to sub-pixel precision [11]. The initial set of correspondences is usually computed by minimization of a *matching cost function* that has been laid out as a *disparity space image* (DSI) [2, 14].

Sub-pixel refinement of correspondences can be performed over a finely sampled or continuously reconstructed neighborhood of the DSI around the initial integer match. The continuous reconstruction strategy has the advantage of being simple and efficient. On the other hand, although computationally more expensive, the supersampling alternative tends to be more accurate. Efforts have been made both to improve the quality of reconstruction-based refinement [12, 13] and to improve the efficiency of supersampling [6].

In this paper, we identify a new source of bias for reconstruction-based sub-pixel refinement strategies (section 2). It can be observed when one image is considered as reference and the refinement is performed on the corresponding coordinate in the matching image. It arises from the sensitivity of this "traditional" approach to the varying confidence of the matching cost function when evaluated at neighboring pixels. In the final reconstruction, the bias can be experienced as random or coherent noise, as the "texture embossing" addressed by Curless and Levoy [4], or as the "striping effect" addressed by Zhang et al. [16].

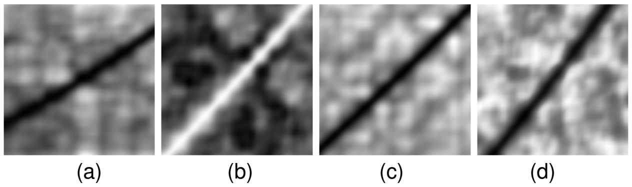


Figure 1: Examples of matching cost functions. (a) Sum of squared differences. (b) Normalized cross-correlation. (c) Birchfield and Tomasi [1]. (d) Sum of absolute differences. Note the matching ridge and how the functions are symmetric with regard to it.

To avoid bias, our symmetric sub-pixel refinement strategy refines both the reference and the matching image coordinates simultaneously, in a symmetric way, by looking for the minimum of the matching cost function along a direction that is insensitive to its confidence variations (section 3). We present results on both synthetic data and real scans obtained using active stereo (section 4), which show that this new method significantly reduces bias in high-variance situations. Additionally, we demonstrate that one of its variants avoids the "pixel locking" effect addressed by Shimizu and Okutomi [12].

2. The Symmetry of Matching Cost

Consider two rectified cameras C_1 and C_2, producing images I_1 and I_2 of an object, such that the scan-lines in each image are corresponding epipolar lines [7]. In this setup, Yang et al. [14] reduced the problem of stereo matching to that of finding a surface in the disparity-space image $\Xi(x_1, y, d)$, which measures the cost of matching points (x_1, y) in I_1 and $(x_1 + d, y)$ in I_2. The matching cost is defined by a metric M that compares neighborhoods of pixel values, so that $\Xi(x_1, y, d) \equiv M(I_1(x_1, y), I_2(x_1 + d, y))$. Note that for a given scan-line y, the problem simplifies even further to that of finding a *matching ridge*, which is the extremum curve in $\Xi_y(x_1, d)$.

Instead of working in disparity space, we prefer to work directly with image coordinates. The concept of disparity implies taking one camera as reference and, as we shall see, this is a source of bias. The direct parameterization $F_y(x_1, x_2) \equiv \Xi_y(x_1, x_2 - x_1)$ is more symmetric and simplifies our analysis. Figure 1 shows examples of popular

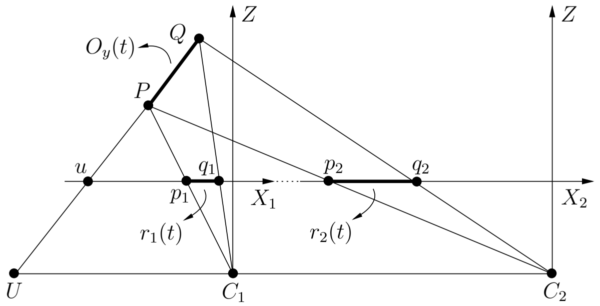


Figure 2: The slope of the matching ridge. The geometry of the setup yields an expression for the slope $\mathrm{d}r_2/\mathrm{d}r_1$, as given by equation 6. Each intersection U between the object tangent and the baseline of the cameras produces a different slope.

matching cost functions under this direct parametrization. In each case, the matching ridge is clearly visible. We also notice a certain symmetry of the matching cost, which we explain below.

Consider the intersection between the object being imaged and a given epipolar plane, as shown in figure 2. It defines a curve $O_y(t)$ that is projected into I_1 and I_2. If $r_1(t)$ and $r_2(t)$ are the corresponding parametrizations for these projections, the matching ridge is simply the curve defined by $R_y(t) = (r_1(t), r_2(t))$. Given a perfect matching pair (x_1, x_2), it is clear that R_y goes through (x_1, x_2) for some t. If r_1 and r_2 are continuous and smooth at t, then $(x_1 + \mathrm{d}r_1(t), x_2 + \mathrm{d}r_2(t))$ is a first order approximation for R_y. It follows that $(x_1 \pm \mathrm{d}r_1, x_2 \pm \mathrm{d}r_2)$ are also on the matching ridge and therefore are also matching pairs.

Comparing the values of $F_y(x_1 + \mathrm{d}r_1, x_2 - \mathrm{d}r_2)$ and $F_y(x_1 - \mathrm{d}r_1, x_2 + \mathrm{d}r_2)$, we notice that they must be similar:

$$\begin{aligned}
&F_y(x_1 + \mathrm{d}r_1, x_2 - \mathrm{d}r_2) \equiv \\
&\quad\equiv\ M(I_1(x_1 + \mathrm{d}r_1, y), I_2(x_2 - \mathrm{d}r_2, y)) && (1)\\
&\quad=\ M(I_2(x_2 - \mathrm{d}r_2, y), I_1(x_1 + \mathrm{d}r_1, y)) && (2)\\
&\quad\approx\ M(I_1(x_1 - \mathrm{d}r_1, y), I_1(x_1 + \mathrm{d}r_1, y)) && (3)\\
&\quad\approx\ M(I_1(x_1 - \mathrm{d}r_1, y), I_2(x_2 + \mathrm{d}r_2, y)) && (4)\\
&\quad\equiv\ F_y(x_1 - \mathrm{d}r_1, x_2 + \mathrm{d}r_2) && (5)
\end{aligned}$$

Steps (1) and (5) are by definition. Step (2) follows from the symmetry of M. Steps (3) and (4) come first from the fact that, since $x_1 \pm \mathrm{d}r_1$ matches $x_2 \pm \mathrm{d}r_2$, $I_1(x_1 \pm \mathrm{d}r_1, y)$ must be similar to $I_2(x_2 \pm \mathrm{d}r_2, y)$. The continuity of M then leads to the approximations.

We have thus shown that F_y is locally skew-symmetric about R_y. The symmetry is such that, if a segment of the matching ridge is the diagonal of a rectangle, then symmetric pairs can be found along *symmetric lines* parallel to the other diagonal. These may or may not be perpendicular to the matching ridge (see figure 3).

The slope of the matching ridge (which is also the symmetry axis) is given by $\frac{\mathrm{d}r2}{\mathrm{d}r1}$. This ratio can can be written as a function of the baseline distance between the cameras and the tangent to the object at the point being imaged (equation 6). For a geometric derivation, assume a linear object segment PQ as in figure 2. The slope $\frac{\mathrm{d}r_2}{\mathrm{d}r_1}$ is equal to the ratio between the lengths of segments $\frac{p_2q_2}{p_1q_1}$. From triangles UPC_1 and UPC_2, we have the relation $\frac{up_1}{UC_1} = \frac{up_2}{UC_2}$. The same relation holds for triangles UQC_1 and UQC_2, from which $\frac{uq_1}{UC_1} = \frac{uq_2}{UC_2}$. Subtracting both relations, we obtain:

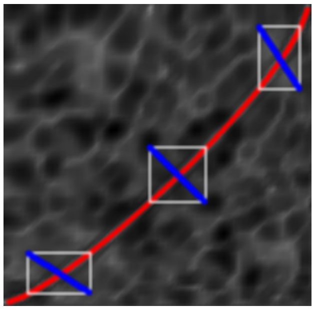


Figure 3: The skew-symmetry of matching cost. If a segment of the matching ridge (shown in red) is the diagonal of a rectangle (shown in white), symmetric pairs can be found along symmetric lines parallel to the other diagonal (shown in blue).

$$\frac{\mathrm{d}r2}{\mathrm{d}r1} = \frac{UC_2}{UC_1} = \frac{Z - X_2\frac{\mathrm{d}Z}{\mathrm{d}X_2}}{Z - X_1\frac{\mathrm{d}Z}{\mathrm{d}X_1}} \tag{6}$$

The slope is $\frac{\pi}{4}$ if the two cameras coincide or if the object tangent is parallel to the baseline. When the tangent goes through either center of projection, the ridge is perpendicular to the corresponding axis. Interestingly, any object tangent going through a given point U in the baseline produces the same matching ridge slope. When U falls between C_1 and C_2, only one of the cameras can see the object, because the other camera sees it from behind. It follows that the slope of the matching ridge is always positive.

With these observations in mind, we proceed to an analysis of the traditional approach to sub-pixel refinement.

2.1. Traditional Sub-pixel Refinement

Assume (i_1, i_2) is a pair of integer best matches. Considering C_1 as the reference camera, the traditional approach is to determine a sub-pixel precision correspondence $i_2 + \bar{t}_2$ for each i_1. To determine the optimal value of the displacement $\bar{t}_2$, a continuous constant-i_1 cut is reconstructed from the matching cost function values neighboring (i_1, i_2) and $\bar{t}_2$ is chosen to bring the reconstruction to its extremum. It is common to fit a parabola or other curve to the values $F_y(i_1, i_2 - 1)$, $F_y(i_1, i_2)$ and $F_y(i_1, i_2 + 1)$.

Unfortunately, a closer look at the matching ridges of figure 1 reveals that their "widths" vary considerably. This variation reflects the changing confidence of matching cost functions when applied to different pairs of epipolar points. For example, uneven surface texture and illumination might cause the matching function to be highly discriminating in one part of the surface, leading to a higher and narrower

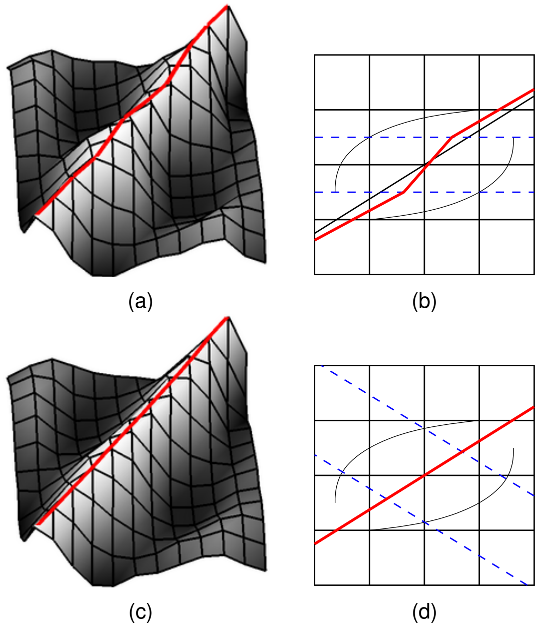


Figure 4: Uncertainty bumps. (a) Sliding an axis-aligned cut across uncertainty bumps causes bias. (c) On the other hand, cuts aligned to the symmetric lines of the matching cost function are insensitive to the bumps. Figures (b, d) show schematic views of the real data shown in figures (a, c). Curved lines show a hypothetical uncertainty bump and dashed lines show the cut directions.

ridge. Elsewhere, the matches might be more ambiguous, leading to a lower and flatter ridge.

As we compute the sub-pixel matches for each i_1, we slide the constant-i_1 cut past several of these *uncertainty bumps* in the matching ridge. As each bump goes by, the fit is biased first to one side, then to the other. Figure 4(a) shows the phenomenon in real data, and figure 4(b) explains why it happens schematically. This bias is responsible for most of the noise seen in the "traditional" reconstructions of figures 7 and 8.

As suggested by figures 4(c) and 4(d), we can avoid this problem if we look for the extrema along the symmetric lines of the matching cost function. Neither camera is considered as reference, and the refined matches will have sub-pixel precision in the coordinates of *both* images. This is the fundamental idea behind our symmetric sub-pixel refinement method.

3. Symmetric Sub-pixel Refinement

Guided by the desire to capture the symmetry of the matching cost function, we consider a 2D neighborhood of matching cost values around (i_1, i_2), and reconstruct a continuous *surface* $\mathcal{S}(t_1, t_2)$ from it. We then define $\mathcal{C}(t) = \mathcal{S}(s_1 t, s_2 t)$, a cut through the reconstruction in the $[s_1\ s_2]^T$ direction. The symmetric sub-pixel refined match is given by the pair $(i_1 + s_1\bar{t}, i_2 + s_2\bar{t})$, where $[s_1\ s_2]^T$ follows the lines of symmetry of matching cost, and $\bar{t}$ is chosen to bring the cut to its extremum.

All that is left to do is choose the surface reconstruction method and find the direction of the cut. Below we investigate some options.

3.1. Quadric Interpolation

One candidate for reconstruction is a quadric that interpolates all 9 values in the 3×3 neighborhood N_3 around (i_1, i_2). This quadric is uniquely defined by the following formulas:

$$\mathcal{S}_q(t_1, t_2) = \mathbf{q}^T(t_2)\, N_3\, \mathbf{q}(t_1) \tag{7}$$

$$\mathbf{q}(t) = \begin{bmatrix} \frac{1}{2}t(t+1) \\ 1-t^2 \\ \frac{1}{2}t(t-1) \end{bmatrix} \tag{8}$$

Note that, under this reconstruction, the traditional approach of fitting a parabola to the constant-i_1 cut reduces to finding the extremum in the $[0\ 1]^T$ direction.

Since a cut through a quadric is at most a degree 4 polynomial, there is a closed form expression for the position of its extremum. Furthermore, since the initial bracket is trivial and the target precision is modest, it can be easily and efficiently determined with a golden section search [9].

3.2. Uniform B-Spline Approximation

Moving away from interpolation, we can consider a larger neighborhood and use a B-Spline approximation for the matching cost function. Consider a 5×5 neighborhood around (i_1, i_2). It is composed of four overlapping 4×4 neighborhoods N_4^j. We can define cubic patches for each of these and use their union as the B-Spline approximation:

$$\mathcal{S}_b(t_1, t_2) = \mathcal{S}_b^j(t_1 - o_1^j, t_2 - o_2^j) \tag{9}$$

$$\mathcal{S}_b^j(t_1, t_2) = \mathbf{b}^T(t_2)\, N_4^j\, \mathbf{b}(t_1) \tag{10}$$

$$\mathbf{b}(t) = \frac{1}{6}\begin{bmatrix} t^3 \\ -3t^3 + 3t^2 + 3t^2 + 1 \\ 3t^3 - 6t^2 + 4 \\ -t^3 + 3t^2 - 3t + 1 \end{bmatrix} \tag{11}$$

The offsets o_1^j and o_2^j simply adjust (t_1, t_2) to local patch coordinates. Note that S_b is C^2 continuous everywhere and only the 3×3 neighborhood around (i_1, i_2) influences the surface at the center of the parametrization.

3.3. Gaussian Cylinder Approximation

Since the matching cost function neighborhoods we are interested in are part of the matching ridge, we can design a surface with meaningful degrees of freedom. To this end, the following surface represents a Gaussian Cylinder generated by a straight line:

$$\mathcal{S}_g(t_1, t_2) = G(D(t_1, t_2)) \tag{12}$$

$$G(d) = a\mathrm{e}^{-d^2} + b \tag{13}$$

$$D(t_1, t_2) = s_1 t_1 + s_2\, t_2 - p \tag{14}$$

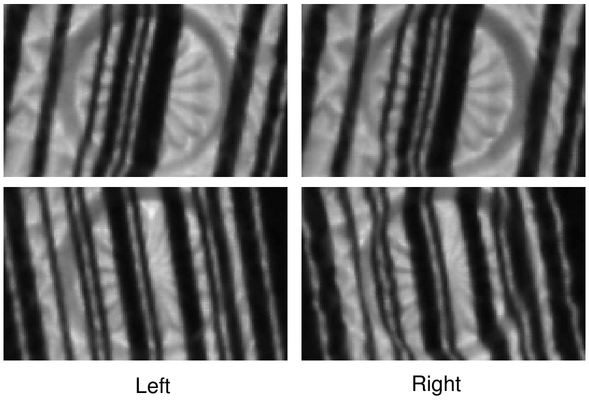


Figure 5: Examples of input images. A close-up is shown from two of the image pairs used in the reconstruction of the object shown on the left of image 8. Patterns of varying orientation are required to ensure that no ambiguities arise when the projector is placed away from the baseline of the cameras.

This surface enforces a ridge-like shape for the reconstruction. The parameters a, b, s_1, s_2, and p can be determined by non-linear least squares minimization on the 3×3 neighborhood around (i_1, i_2). The line $D(t_1, t_2) = 0$ then gives the local approximation for the matching ridge, from which the sub-pixel estimate can be easily found. Usually, a few iterations of the Levenberg-Marquardt method, as implemented by Lourakis [8], are enough for a good fit.

3.4. Choice of Cut Direction

As suggested by figure 4, the direction $[s_1\ s_2]^T$ that follows the symmetric lines of the matching cost function is the right choice for a cut through $\mathcal{S}$. Besides respecting the symmetry of matching cost, this direction will in general be more stable than axis aligned directions. Unfortunately, since formula 6 requires previous knowledge about the scene, we can not directly use it to determine the cut direction.

We notice, however, that the direction of highest curvature of $\mathcal{S}$ at $(0, 0)$ provides a good estimate for $[s_2\ s_1]^T$. This is because the highest curvature happens for cuts almost perpendicular to the matching ridge. From that, $[-s_1\ s_2]^T$ is an approximation for the matching ridge direction and $[s_1\ s_2]^T$ is therefore a good estimate for the direction we are looking for.

In practice, this is how we obtain the cut direction for the quadric interpolation and the B-Spline approximation. For the Gaussian cylinder, the estimate (which is directly available from the surface formulation) is not required.

4. Results

To evaluate our method, we tested it with real and synthetic data, using a temporal stereo triangulation scanner setup [5, 15]. In this active scanning technique, random stripe patterns are projected onto the scene while two cameras capture synchronized images. Since each point in the visible surface receives a unique light profile through time, it is possible to establish correspondences in a fashion similar to the area-based matching of standard stereo, but using windows that extend only through time (i.e., with spatial width and height of just one pixel). This strategy has the advantage of producing perfect correspondences and of being unaffected by depth discontinuities. It provides us with a way to isolate the sub-pixel refinement evaluation from other sources of error that could mask the effects we want to analyze.

Our real scanner is composed of two Sony DFW-X700 1024×768 firewire cameras and a Toshiba TLP511 projector with the same resolution. The cameras are calibrated using the toolbox by Bouguet [3] and synchronized by an external trigger. Our virtual scanner uses similar camera parameters, but produces image pairs from a 3D model, simulating the stripe patterns with projective textures. Both scanners have an estimated resolution of 0.2mm and a working volume 2000 times as large. Our tests are performed with static scenes, using sequences of 32 images, and with the normalized-cross-correlation metric. Fewer images would be sufficient, but the additional information improves the quality of the matching cost function. Figure 5 shows examples of input images to our system. The close-ups shown correspond to two of the pairs used to produce the object reconstruction on the left of figure 8.

We use two synthetic reference models: a sphere, for its wide range of smooth depth and orientation variation, and a parametric surface $Z(r, \theta)$, for its sharp features and arbitrarily small details:

$$Z(r, \theta) = -\frac{1}{10} r |\sin 16\theta| \tag{15}$$

Figures 7 and 8 show renderings of data recovered by the virtual and real scanners respectively, using the traditional parabolic fit, the method by Shimizu and Okutomi [12], and our symmetric method employing each of the proposed reconstruction alternatives. The figures show that our method eliminates most of the visible noise equally well for each reconstruction alternative. In particular, note how the “striping effect” has been eliminated from the Greek panel in figure 8.

Figures 9 and 10 show depth profiles for the reconstructed synthetic sphere and parametric surface. The profile for the 5mm radius sphere shows a considerable variation in object tangent direction (about 115 degrees). This in turn produces large variations in the matching ridge slope. The spherical profiles show that our method performs well across such variations. In the parametric surface, sharp details are progressively smaller closer to the center. The profiles show that our method recovers details up to the same resolution as the traditional approach. Therefore, it is not simply eliminating noise at the expense of detail.

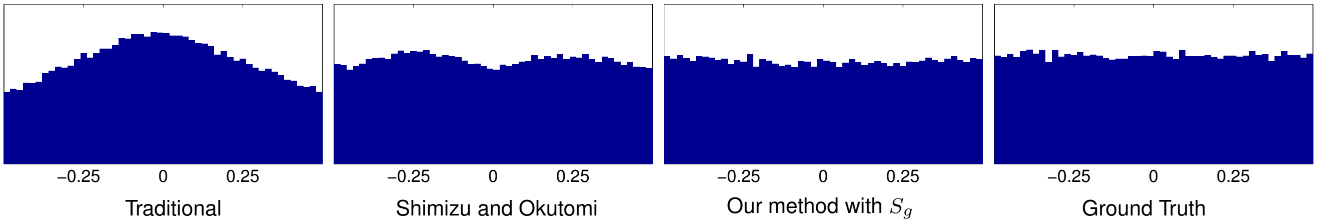


Figure 6: Histograms of sub-pixel deviation from the integer match for the spherical model. The traditional method is biased towards the center, producing a "pixel locking" effect. The method by Shimizu and Okutomi [12] performs better, but is still biased towards ± 0.25. In contrast, the histogram for the Gaussian cylinder reconstruction is almost flat, as is the ground truth.

The Gaussian cylinder reconstruction also reduces the "pixel locking" effect addressed by Shimizu and Okutomi [12]. This is no surprise, since a similar result holds for the traditional sub-pixel refinement with Gaussian fit [10]. Using the spherical model, we computed histograms of the estimated sub-pixel displacement from the integer match. Results are shown in figure 6.

Figure 11 shows the "texture embossing" effect on the depth profile of a real planar object whose reflectance varies sinusoidally. The varying reflectance causes the confidence of the matching cost function to vary wildly along the matching ridge. Accordingly, severe uncertainty bump errors disrupt the traditional sub-pixel refinement strategy. In contrast, the noise levels observed in the symmetric reconstructions are within the expected scanner precision.

5. Conclusion

In this paper, we identified a new source of bias in the sub-pixel refinement of stereo correspondences. In reconstructed scenes, it manifests itself as random or coherent noise. To avoid this bias, we present a novel technique that exploits the inherent symmetry of matching cost functions and refines matching coordinates in both images simultaneously. Results show that our approach performs better than previous techniques.

Acknowledgements

We would like to thank Hector Gonzalez-Banos for fruitful discussions during the early stages of this project. We also thank the National Science Foundation for funding that partially supports this project (Szymon Rusinkiewicz: CCF-0347427, "Practical 3D Model Acquisition").

References

[1] Stan Birchfield and Carlo Tomasi. A pixel dissimilarity measure that is insensitive to image sampling. *PAMI*, 20(4):401–406, April 1998.

[2] Aaron F. Bobick and Stephen S. Intille. Large occlusion stereo. *IJCV*, 33(3):181–200, 1999.

[3] Jean-Yves Bouguet. *Camera Calibration Toolbox for Matlab*, October 2004. URL http://www.vision.caltech.edu/bouguetj/calib_doc.

[4] B. Curless and M. Levoy. Better optical triangulation through spacetime analysis. In *ICCV*, pages 987–994, 1995.

[5] James Davis, Diego Nehab, Ravi Ramamoorthi, and Szymon Rusinkiewicz. Spacetime stereo: A unifying framework for depth from triangulation. *PAMI*, 27(2): 296–302, February 2005.

[6] R. W. Frischholz and K. P. Spinnler. Class of algorithms for real-time subpixel registration. In Donald W. Braggins, editor, *Proceedings of SPIE, Computer Vision for Industry*, volume 1989, pages 50–59, December 1993.

[7] C. Loop and Zhengyou Zhang. Computing rectifying homographies for stereo vision. In *CVPR*, pages 125–131, 1999.

[8] M.I.A. Lourakis. levmar: Levenberg-Marquardt nonlinear least squares algorithms in C/C++, 2004. URL http://www.ics.forth.gr/~lourakis/levmar.

[9] William H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery. *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, 1992.

[10] T. Roesgen. Optimal subpixel interpolation in particle image velocimetry. *Experiments in Fluids*, 35:252–256, 2003.

[11] D. Scharstein, R. Szeliski, and R. Zabih. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. In *SMBV*, pages 131–140, 2001.

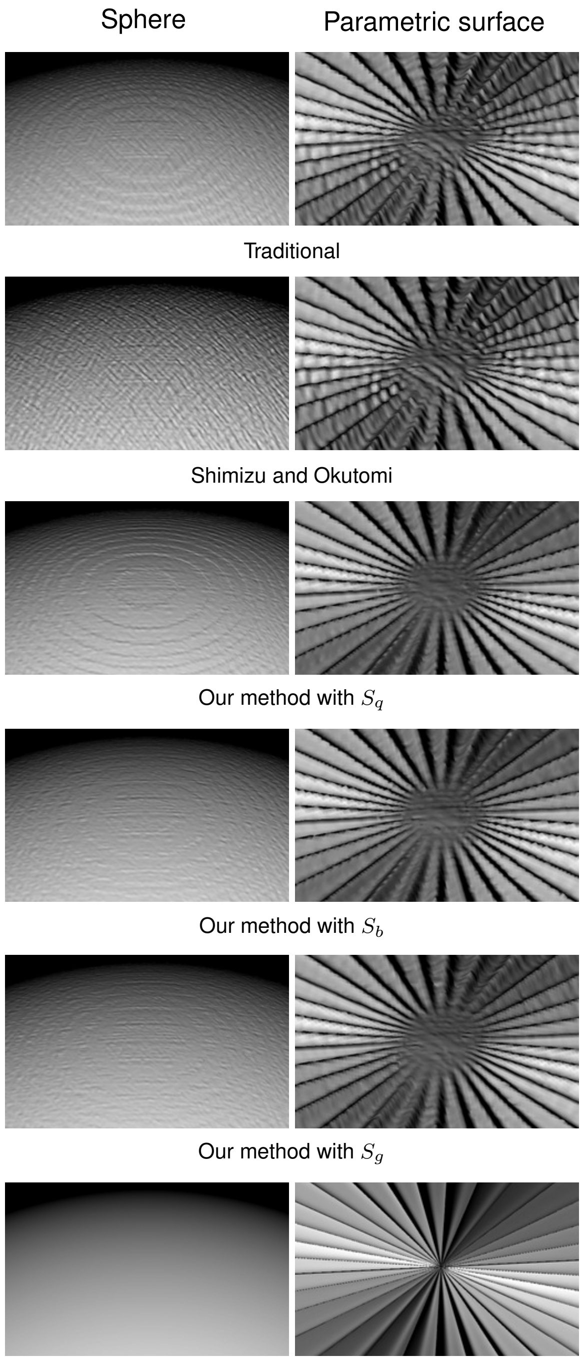
[12] M. Shimizu and M. Okutomi. Precise sub-pixel estimation on area-based matching. In *ICCV*, pages 90–97, 2001.

[13] Richard Szeliski and Daniel Scharstein. Sampling the disparity space image. *PAMI*, 25(3):419–425, March 2004.

[14] Y. Yang, A. Yuille, and J. Lu. Local, global, and multilevel stereo matching. In *CVPR*, pages 274–279, June 1993.

[15] Li Zhang, Brian Curless, and Steven M. Seitz. Spacetime stereo: Shape recovery for dynamic scenes. In *CVPR*, pages 367—374, June 2003.

[16] Li Zhang, Noah Snavely, Brian Curless, and Steven M. Seitz. Spacetime faces: High-resolution capture for modeling and animation. In *ACM Transactions on Graphics*, pages 548–558, August 2004.



Sphere

Parametric surface

Traditional

Shimizu and Okutomi

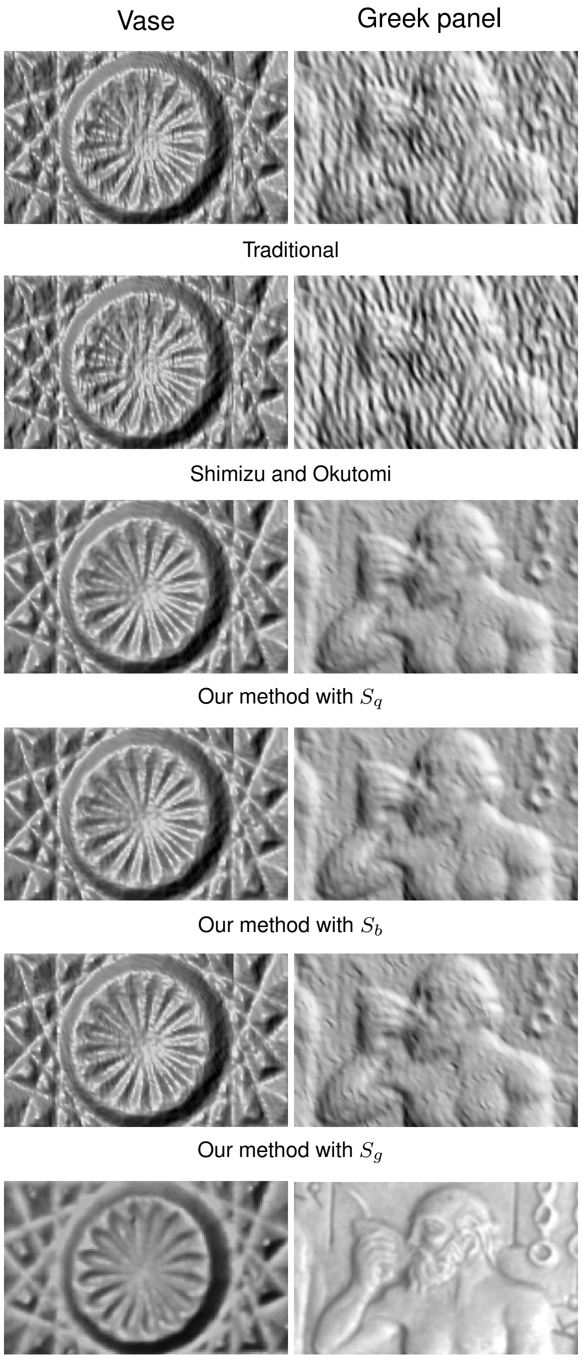
Our method with S_q

Our method with S_b

Our method with S_g

Ground truth

Figure 7: Renderings from reconstructed geometry for the virtual scanner. From the spherical model renderings, note how S_g reduces the “pixel locking” effect. From the parametric surface, note how detail is preserved while noise is eliminated.



Vase

Greek panel

Traditional

Shimizu and Okutomi

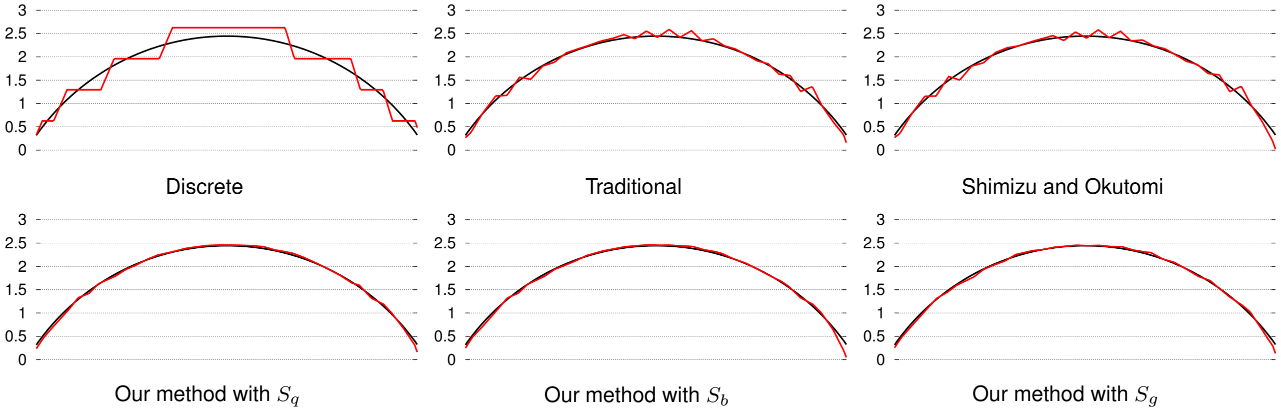
Our method with S_q

Our method with S_b

Our method with S_g

Photograph

Figure 8: Renderings from reconstructed geometry for the real scanner. Note how the noise level is reduced by our method. In addition, note how the “striping effect” was eliminated from the (replica) Greek panel scan.



Discrete

Traditional

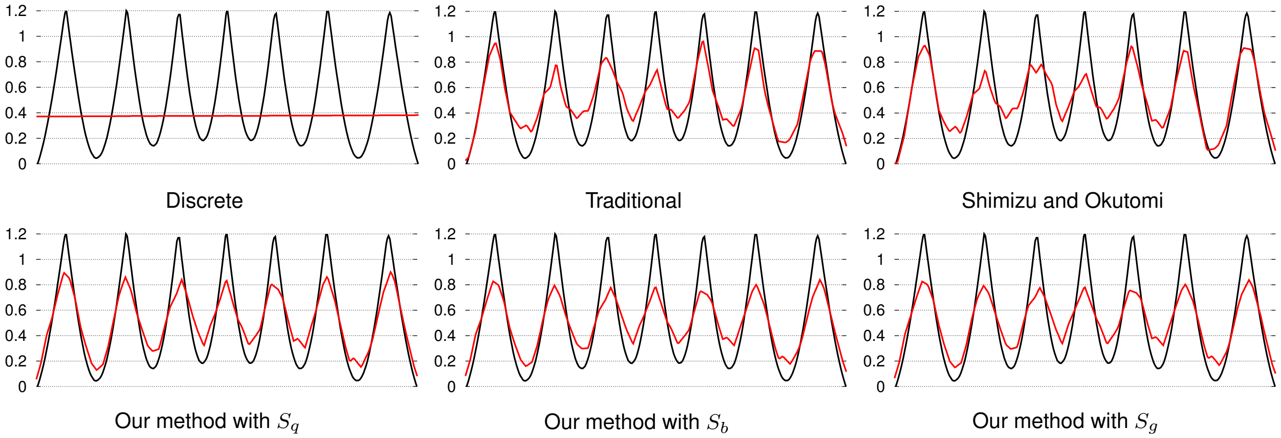
Shimizu and Okutomi

Our method with S_q

Our method with S_b

Our method with S_g

Figure 9: Depth profiles for a synthetic spherical model (5mm radius). For each plot, ground truth is shown in black. Note the reduced noise level for a variety of matching ridge slopes (produced by the varying object tangent).



Discrete

Traditional

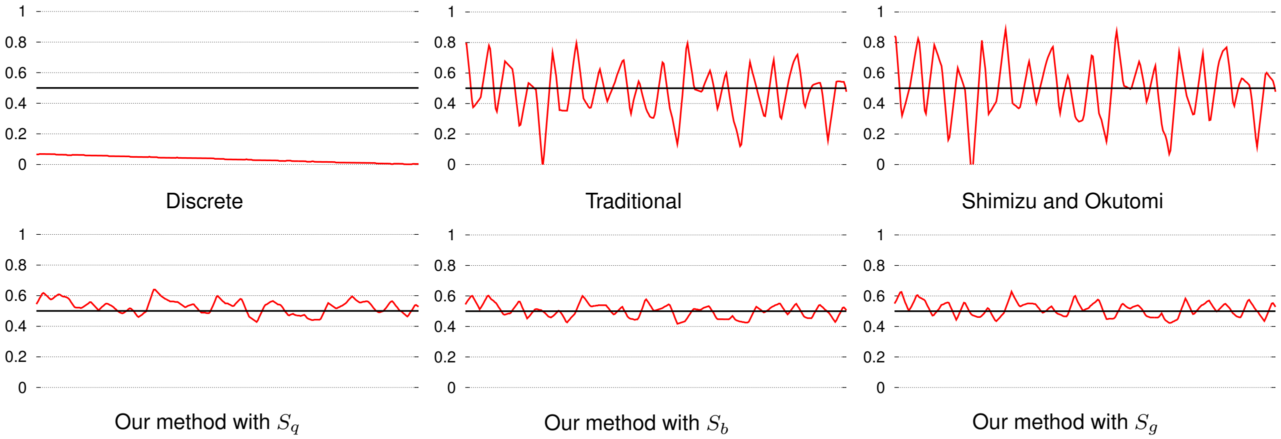
Shimizu and Okutomi

Our method with S_q

Our method with S_b

Our method with S_g

Figure 10: Depth profiles for the synthetic parametric surface. For each plot, ground truth is shown in black. The profiles show that our method does not simply eliminate detail along with noise.



Discrete

Traditional

Shimizu and Okutomi

Our method with S_q

Our method with S_b

Our method with S_g

Figure 11: Depth profiles for a *real* planar object whose reflectance varies sinusoidally. The least-squares fit plane is shown in black. The varying albedo generates severe systematic biases in the traditional sub-pixel estimation. On the other hand, the noise observed in the symmetric reconstructions is within the scanner precision.