Shape from Single Stripe Pattern Illumination

Conference Paper · September 2002

DOI: 10.1007/3-540-45783-6_29 · Source: DBLP

Simon Winkelbach
Technische Universität Braunschweig

Published in
Luc Van Gool (Editor): Pattern Recognition, Lecture Notes in Computer Science 2449, Springer 2002, page 240-247.

Shape from Single Stripe Pattern Illumination

S. Winkelbach and F. M. Wahl

Institute for Robotics and Process Control,
Technical University of Braunschweig
Mühlenpfordtstr. 23, D-38106 Braunschweig, Germany
{S. Winkelbach, F. Wahl}@tu-bs.de

Abstract. This paper presents a strategy for rapid reconstruction of surfaces in 3d which only uses a *single* camera shot of an object illuminated with a simple stripe pattern. With this respect, it is a meaningful extension of our 'shape from 2d edge gradient' method introduced earlier. The reconstruction is based on determining stripe directions and stripe widths in the camera image in order to estimate surface orientation. I.e., this method does not use triangulation for range data acquisition, but rather computes surface normals. These normals can be 2d integrated and thus yield the surface coordinates; in addition they can be used to compute robust 3d features of free-form surfaces for object recognition, pose estimation, etc. The method is straightforward and very efficient by processing only one image and using only simple image processing operations.

1 Introduction

3d shape recovery is an important field of research since many years. Many publications deal with shape reconstruction techniques, such as stereo vision [e.g.1] structured light [e.g.2,3,4], coded light [e.g.5] and one-shot systems projecting a local identification code [e.g.6,7]. All these systems are based on triangulation to compute range data. An other class of techniques estimate surface normals instead of absolute range data, such as shape from texture [e.g.8,9], shape from shading [e.g.10], shape from specularity [e.g.11],etc., as well as our 'shape from 2d edge gradients' approach published earlier in [12]. An advantage of the second class of systems is, that no correspondence problem has to be solved and, if surface properties are needed, they directly generate surface normals without the necessity to derive them subsequently from noisy range data. Surface normals are an important basis of robust 3d feature determination, as for example relative surface orientations, curvatures, local maxima of free-form surfaces. Moreover, they can be used to reconstruct the surface itself, or they can be utilized as basis for 3d segmentation, object recognition, pose estimation, calculation of an illumination independent model and recalculation of lighting from any desired direction, etc. The technique described here directly generates surface normals by acquiring only one shot of a single stripe pattern illumination. Thus, it even is able to retrieve the shape of moving objects! As instrumentation it requires one camera and one static light stripe projector (Figure 1). The reconstruction technique is based on the fact, that

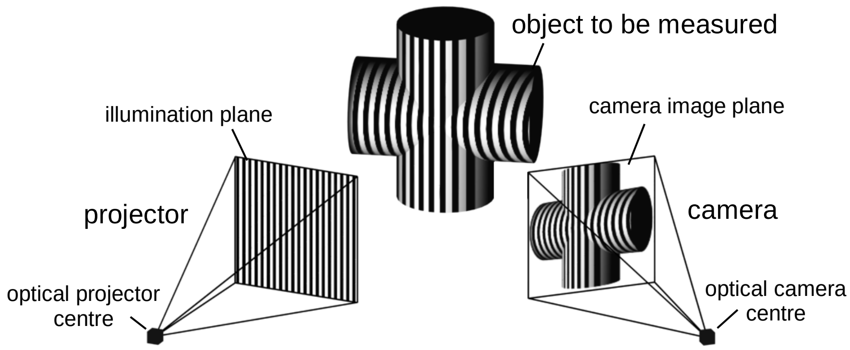

Fig. 1. Measurement setup with a static stripe pattern projector and a horizontally displaced camera

directions *and* widths of projected stripes in the captured 2d image depend on the local orientation of the object's surface in 3d. Surface normals are obtained by analysing the local stripe edge directions and stripe widths of the deformed 2d stripe images. This approach is an alternative technique of our 'shape from 2d edge gradients' approach published earlier in [12], which employs local edge directions of two! stripe projections rotated relative to each other.

During the review process of this paper we became aware of two short papers of Asada, Ichikawa and Tsuji [13,14] which already described the possibility to use direction and width of stripe illumination for surface normal measurement. It appears to us, that these two papers have been forgotten by the computer vision community. In their work the authors describe only very briefly the essential image processing and calculation methods for surface normal calculation and then focus on usage of normals for segmentation of planar surface parts. The discussed results seem to be qualitative and quantitative poor; many advantages and applications of the 'one shot stripe illumination strategy' remain unconsidered. Maybe this is the reason why their work has been fallen into oblivion. In this paper we will give a more detailed discussion of the approach and answer some open questions of [13] and [14].

At a first glance, an almost similar method using a square grid has been presented in [15]; but in contrast to the approach discussed here it requires complex computation, e.g. for detection of lines and line crossings and for checking grid connectivity. Moreover, it achieves a lower density of measurement points yielding a lower lateral resolution.

The surface reconstruction described here can be subdivided into several functional steps. First, we take a grey level image of the scene illuminated with a stripe pattern. Subsequently, local edge directions and widths of stripes are measured (Section 2). Missing measure values between the stripes are augmented by simple linear 1d interpolation. On the basis of the local stripe directions and stripe widths we calculate the local surface normals (Section 3). Finally we present some experimental results and discuss the accuracy of this technique (Section 4).

2 Local Stripe Direction and Stripe Width Determination

If the object surface has an approximate homogeneous reflection characteristic, it is sufficient to analyse the stripe pattern of one camera image, for example by using a simple threshold to extract the illumination pattern from object colour. Object textures and other varying surface reflections may inhibit reliable detection and analysis of the projected illumination pattern in the camera image. For this case, we proposed a pre-processing procedure (see [12]), which separates the illumination pattern from the surface reflection characteristics of the object. By regarding real grey level edges rather than binary ones subsequent measurement of stripe widths and stripe directions becomes more precise and reaches sub-pixel precision.

Determination of local stripe directions already has been explained in detail in [12]. For reason of completeness we will give a short synopsis: Gradient directions can be calculated by well-known operators like Sobel, Canny, etc. Noisy angles arise mainly in homogenous areas where gradient magnitudes are low. Thus high gradient magnitudes can be used to mask reliable edge angles, i.e. eliminating erroneous ones. The masking of valid angles can be realized by simply using a threshold to separate high and low gradient magnitudes; but we even get better results by using only angles at local maxima of the gradient magnitudes for each 1d scan line. After elimination of erroneous angle values, they will be replaced by 1d interpolation.

Determination of local stripe widths should be performed on sub-pixel level in order to achieve high accuracy. Each scan line can be processed independently; thus only 1d signal processing operations are required. Our technique achieves slightly better results than the 'Linear Mixing Model' suggested in [13,14], but a detailed verification of these results needs more time and space and is a topic of further publications. To estimate the mean width of a stripe, we analyse a small region around each edge (see Fig. 2 left side), which can be found by using local maxima of edge gradient magnitudes.

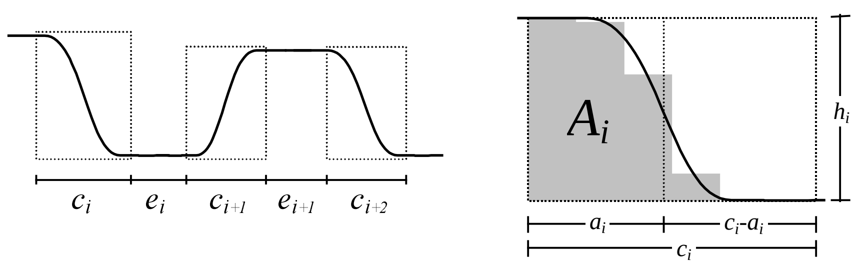

Fig. 2. Left side: Partitioning of a grey level profile into small regions around each edge
Rigth side: Model of a stripe edge for sub-pixel estimation of the mean stripe width

The right side of Figure 2 shows a model of such a region around a stripe edge. The mean distance a_i between the left region margin and the edge can be calculated by the edge height h_i, the region width c_i and the area A_i below the sampled edge by using the width-area proportion

$$\frac{a_i}{c_i} = \frac{A_i}{c_i \cdot h_i} \quad \Leftrightarrow \quad a_i = \frac{A_i}{h_i} \tag{1}$$

After processing this calculation for every stripe edge, it is easy to compute the stripe widths d_i' of all stripes

$$d_i' = (c_i - a_i) + e_i + a_{i+1} \tag{2}$$

The mean width of two neighboured stripes can be stored into the 'width image' at the pixel location of the interior edge. By this method, we prevent errors if bright stripes are wider than dark stripes, which can result from camera white clipping or blooming. In a final step, missing values in the 'width image' are 1d interpolated. Figure 3 shows the result of this method applied to a normalized image of a stripe pattern illuminated surface.

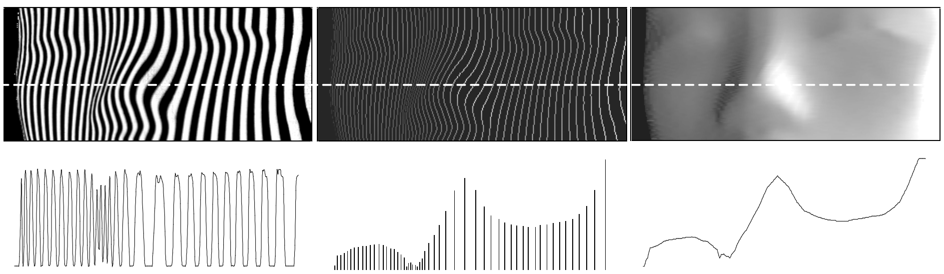

Fig. 3. Left: Normalized image of a stripe pattern illuminated surface and a profile line; Middle: Determined local stripe widths and corresponding profile line; Right: Linear interpolated stripe widths ('widths image') and corresponding profile line

3 Surface Normal Computation

After estimation of local stripe directions and stripe widths in 2d, the local surface normals can be computed. The model of Figure 4 illustrates the mathematical relation of surface slope and estimated local stripe direction (adapted from [12]). All vectors in the following are given with respect to the camera coordinate frame. Each image pixel defines a 'view vector' $\vec{s}$ which points from the camera coordinate origin of frame *C* (optical centre) to the pixel coordinate in the image plane. The stripe angle value ω at this image coordinate specifies a stripe direction vector $\vec{v}\,'$ lying in the viewing plane as well as in the image plane. The real tangential vector $\vec{v}_1$ of the projected stripe on the object's surface in 3d is a linear combination of $\vec{s}$ and $\vec{v}\,'$. Thus $\vec{v}_1$ is perpendicular to the normal $\vec{c} = \vec{v}\,' \times \vec{s}$. $\vec{v}_1$ is also perpendicular to the normal $\vec{p}$ of the stripe projection plane. Assuming that the projection planes of all stripes are parallel, all stripes have the same projection normal. This justifies the usage of the cross product $\vec{v}_1 = \vec{c} \times \vec{p}$ to calculate a tangential vector $\vec{v}_1$ of the local surface, which determines one degree of freedom of the corresponding surface normal in 3d.

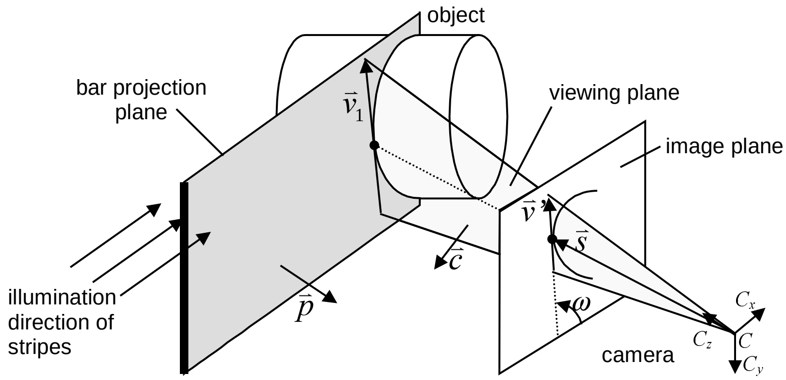

Fig. 4. Projection of a stripe edge onto an object's surface and its mapping into the image plane (adapted from [12])

This holds without using any correspondence information between stripes in the image and the illumination stripes of the projector.

In [12] we proposed to acquire a second image of the scene with a rotated stripe illumination relative to the first one, in order to get a second tangential vector $\vec{v}_2$; the surface normal in this case can be computed as $\vec{n} = \vec{v}_1 \times \vec{v}_2$. The approach described in this paper only uses one stripe illumination with one pattern orientation. In contrast to [12] we utilize the stripe widths to determine the remaining degree of freedom of the surface normals. This holds under the assumption of parallel projecting systems (camera and light projector). In practice we use telecentric lenses or systems with long focal lengths and small apex angles.

For calculation of surface gradients in 3d on the basis of stripe widths, we can use a simple 2d model. Figure 5 illustrates the 2d mathematical relation between the stripe widths in the image and the surface orientations. The stripe width d of the stripe pattern illumination and the constant angle γ between central stripe illumination direction and camera viewing direction

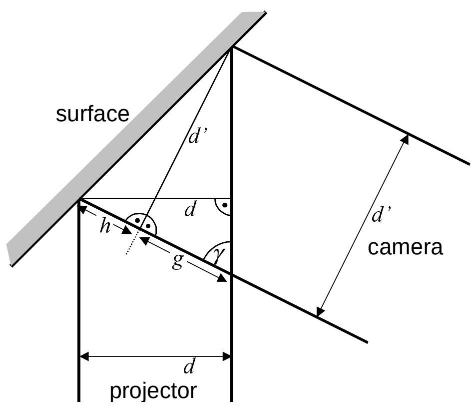

Fig. 5. Calculation of surface gradients on the basis of stripe widths

can be derived from a simple preceding calibration. By using the stripe width *d'* of the stripe in the camera image we can calculate the surface gradient *p* as follows:
Regarding the triangles of the model in Figure 5, we get the simple trigonometrical equations

$$h + g = \frac{d}{\sin\gamma} \quad ; \quad g = \frac{d'}{\tan\gamma} \tag{3}$$

According to this, it follows that

$$h = \frac{d}{\sin\gamma} - \frac{d'}{\tan\gamma} = \frac{d - d' \cdot \cos\gamma}{\sin\gamma} \tag{4}$$

yielding the surface gradient *p*

$$p = \frac{h}{d'} = \frac{d/d' - \cos\gamma}{\sin\gamma} \tag{5}$$

The second tangential surface vector now is given by $\vec{v}_2 = (1 \quad 0 \quad -p)^{\mathrm{T}}$
To calculate the surface normal, we simply use the cross product of the two tangential surface vectors

$$\vec{n} = (n_x \quad n_y \quad n_z)^{\mathrm{T}} = \vec{v}_1 \times \vec{v}_2 \tag{6}$$

Instead of storing three values of the surface normals, we store the surface gradients *p* (in x-direction) and *q* (in y-direction). *p* is given by equation (5); for calculation of *q* we compute the y-z-ratio of the surface normal $q = -n_y / n_z$.

4 Experimental Results and Conclusion

For experimental evaluation we used an off-the-shelf CCD-camera and a conventional video beamer. For projection of only one simple stripe pattern it is also possible to use an ordinary slide projector or a low-cost static pattern projector. The method described above is totally unaffected by shifted patterns and only little affected by defocused pattern illumination or blurred camera images. In order to evaluate the new technique we conducted many experiments with varying setups and different test objects. Due to the limited space of this paper we only can show a few reconstruction results (Figure 6). Regarding accuracy we should mention, that the following experiments only use the angle γ between central stripe illumination direction and camera viewing direction as well as the stripe width *d* of the stripe pattern illumination from a simplified preceding calibration. Therefore inaccuracies also result from neglecting consideration of intrinsic camera and projector parameters. (a1) of Figure 6 shows a spherical surface, illuminated with a stripe pattern, which was the sole basis of our surface normal computation. (a2) shows a small fraction of the reconstructed surface normals (we compute normals at every pixel coordinate covered with a stripe edge and interpolate intermediate areas). Comparing the computed non-interpolated normals

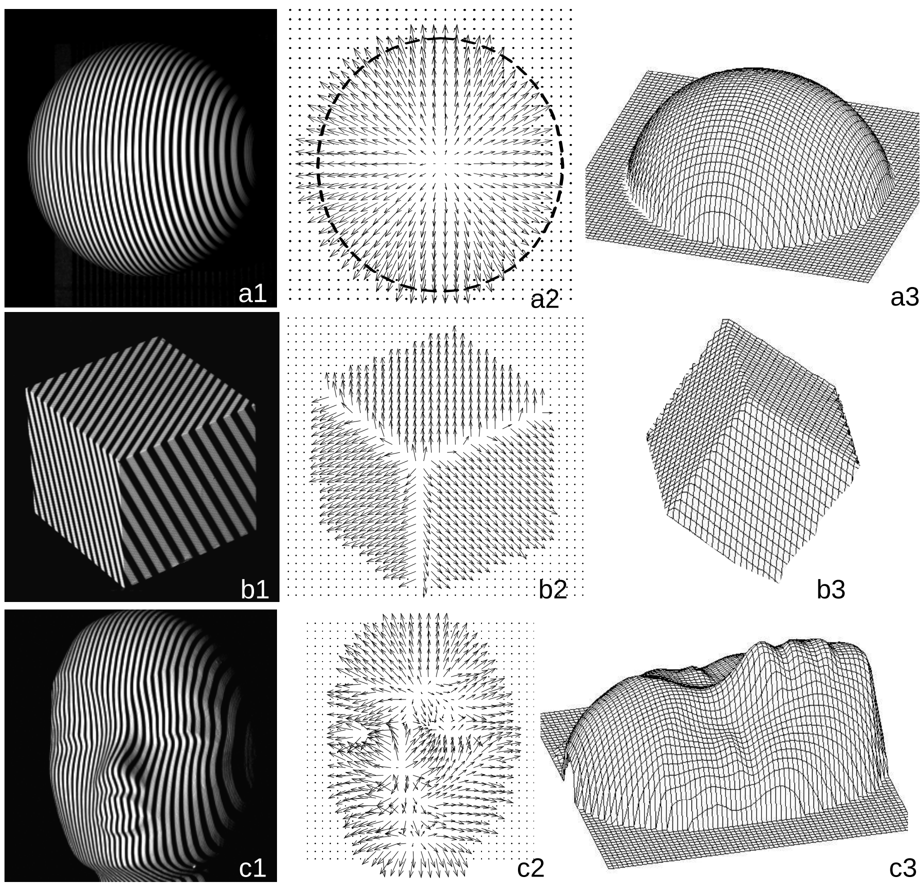

Fig. 6. Reconstruction results: (a1) stripe pattern illuminated spherical surface; (a2) determined surface normals; (a3) 3d plot of the reconstructed surface; (b1-b3) the same for a cube; (c1-c3) the same for a styrofoam head.

with the mathematical exact sphere normals we obtain a mean angular error between exact and reconstructed surface normals of 1.79 degrees with a standard deviation of 1.22 degrees. To compute range data from surface gradients, we applied the 2d integration method proposed by Frankot/Chellappa [16]. As a matter of fact, this integration technique only works for continuous surface parts and is only able to calculate relative range (shape) and not absolute distance information. Especially the discontinuous boundaries between object and background can cause problems and demand an adapted range estimation, which uses only valid gradients. For this purpose we develop a special iterative extension, which will be topic of a further publication. (a3) shows a 3d plot of the computed range map of the sphere. The next three images (b1-b3) show the same sequence for a solid cube. Comparing the computed cube normals with the mathematical exact ones we get an accuracy of 1.14 degree mean angular error with 0.58 degree standard deviation for the top cube face. The most inaccurate normals with 1.95 degree mean angular error and 2.12 degrees standard deviation are obtained for the lower left cube face. In this case the lower left surface has a high tilt angle and the stripes get very close together; thus measurement of stripe orientations and widths becomes more inaccurate. The last three images (c1-c3) show the same sequence with a styrofoam head as test object. As can be seen, shape acquisition with reasonable precision (sufficient for many applications) can be obtained with low cost equipment using just single camera shots. This makes the proposed technique attractive even in case of moving objects. The whole process (image acquisition, surface normals and range map reconstruction of a 512x512 image) only takes about two seconds on a 500 Mhz Pentium III. By acquiring several stripe illuminated scene images with same illumination direction but with phase shifted stripe patterns, the number of gradients with high magnitude can be increased, thus reducing the need for replacing erroneous stripe angles and stripe widths by interpolation.

References

1. D. C. Marr, T. Poggio: A computational theory of human stereo vision, Proc. Roy. Soc. London 204, 1979
2. T. Ueda, M. Matsuki: Time Sequential Coding for Three-Dimensional Measurement and Its Implementation, Denshi-Tsushin-Gakkai-Ronbunshi, 1981
3. M. Oshima, Y. Shirai: Object recognition using three dimensional information, IEEE Transact. on PAMI, vol. 5, July 1983
4. K. L. Boyer and A. C. Kak: Color-Encoded Structured Light for Rapid Active Ranging, IEEE Transact. on PAMI, vol. 9, no. 1, Januar 1987
5. F. M. Wahl: A Coded Light Approach for 3-Dimensional (3D) Vision, IBM Research Report RZ 1452, 1984
6. M. Maruyama, S. Abe: Range Sensing by Projection Multiple Slits with Random Cuts, IEEE PAMI 15(6), pp. 647-650, 1993.
7. P. Vuylsteke, A. Oosterlinck, Range Image Acquisition with a Single Binary-Encoded Light Pattern, IEEE Transact. on PAMI, vol. 12, no. 2, 1990.
8. J.J. Gibson, The Perception of the Visual World, MA: Reverside Press, Cambridge, 1950
9. J.R. Kender, Shape from texture, Proc. DARPA IU Workshop, November 1978
10. B. K. P. Horn and M. J. Brooks: Shape from Shading, M.I.T., Cambridge, 1989
11. G. Healey, T.O. Binford: Local Shape from Specularity, Proc. ICCV, London, June 1987
12. S. Winkelbach, F.M. Wahl: Shape from 2D Edge Gradients, Pattern Recognition, Lecture Notes in Computer Science 2191, Springer, 2001
13. M. Asada, H. Ichikawa, S. Tjuji: Determining of Surface Properties by Projecting a Stripe Pattern, IEEE Proc. of ICPR'86, 1986
14. M. Asada, H. Ichikawa, S. Tsuji: Determining Surface Orientation by Projecting a Stripe Pattern, IEEE Transact. on PAMI, vol. 10, no. 5, September 1988
15. M. Proesmans, L. Van Gool and A. Oosterlinck: One-Shot Active Shape Acquisition, IEEE Proc. of ICPR'96, 1996
16. Robert T. Frankot, R. Chellappa: A Method for Enforcing Integrability in Shape from Shading Algorithms, IEEE Transact. on PAMI, vol. 10, no. 4, July 1988