Space occupancy using multiple shadowimages

Michael S. Langer Gregory Dudek Steven W. Zucker
Centre for Intelligent Machines, McGill University,
3480 University Street, Montréal, H3A 2A7 Canada
{langer | dudek | zucker}@cim.mcgill.ca

Abstract

We address the problem of estimating 3-d space occupancy using video imagery in the context of mobile robotics. A stationary robot observes a cluttered scene from a single viewpoint, and a second robot illuminates the scene from a sequence of directions thus producing a sequence of grey-level images. Differences of successive images are used to compute a sequence of shadowimages. The problem is to compute free space and occupied space from these shadowimages. Solutions to this problem are known for the special case of terrain scenes. We generalize these solutions to non-terrain scenes by making two key observations. First, there is a subset constraint on the shadowimages of a non-terrain scene, which allows the visible surfaces of a non-terrain scene to be recovered by a terrain-based technique. Second, the remaining regions of the shadowimages provide a conservative estimate of the occupied space hidden by these visible surfaces.

1 Introduction

Robots navigate through free space in the world, and one of the primary uses of vision is to infer descriptions of free space. Standard computer vision techniques, such as shape-from-shading or stereo, attempt to reconstruct visible surfaces from a single viewpoint. Multiple views of these surfaces provide a more complete description of object surfaces. Free space is obtained by filling in from these object boundaries.

In this paper, we present a new approach to computing free space which is based on moving the light source rather than the camera. An analogy for the process is a mountain rescue operation, in which a flare is launched to illuminate the scene, while the rescuers remain stationary. It is the shadows and their motion of the directly visible surface that indicate the otherwise-invisible free space.

Figure 1 shows the specific situation we consider. A master mobile robot has a fixed position in a static scene, and a slave robot carrying a point light source moves in a large semicircular arc around this fixed position. For each source position, a grey-level image is obtained. Differences of successive images are used to compute a sequency of binary **shadowimages**. The question we address is how can the space occupancy be computed from this sequence of shadowimages. Related problems occur in natural outdoor environments. For example, a time-lapse sequence could be used as the sun moves across the sky, from sunrise to sunset.

This problem is known as **shape-from-darkness**. It was introduced in [5, 7] under a limiting assumption, namely that the visible surfaces define a terrain. In this paper, we extend the scope of the problem and its solution to non-terrain scenes. In contrast to most existing approaches to range sensing such as binocular stereo or laser triangulation or time-of-flight, the technique proposed here allows the inference of space occupancy in regions hidden the camera's viewpoint.

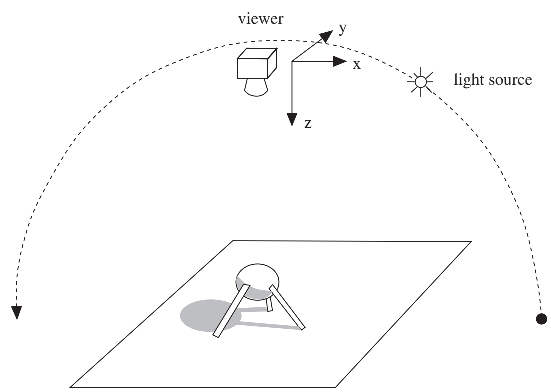

Figure 1: A scene is viewed by a distant camera, and illuminated by a distant point light source which moves in a semicircular arc. A sequence of grey-level images is obtained, and used to compute a sequence of shadowimages.

2 Experimental Setup

Our experimental setup consists of two mobile robots. A master (RWI B-12) carries a camera mounted on a pan-tilt unit, and a slave (Nomad 200) carries a light source. The master moves to a fixed vantage point and remains stationary. The slave then executes a trajectory about the master allowing the shadowimages to be computed.

The two robots must compute their relative position so that the direction of the source can be accurately determined. We propose to automate this by having the master pan the camera prior to each image acquisition to observe the slave's position. Using a pattern painted on the slave robot and knowledge of the pan angle allows its relative position to be inferred. Simply finding the relative direction of the point source is insufficient, since the distance between the two robots must also be inferred[1].

Figure 2: A master robot carries a camera mounted on a pan-tilt unit, and a slave carries a light source. The master moves to a fixed vantage point and remains stationary. The slave then executes a large trajectory about the master.

Figure 3 shows a sequence of grey-level images obtained using our robots. In order to robustly identify shadows, we need to allow for penumbrae, interreflections, and non-uniform illumination [9, 2, 4, 10] (The point source may be neither at infinity nor perfectly isotropic.) In particular, simple thresholding is unlikely to yield completely correct shadow identification.

To address these issues, we have developed a method which takes advantage of the continuum of light source directions. Image intensity variations between successive images are small when the changes are due to shading. (For the case of Lambertian reflectance, the intensity change is proportional to the sine of the angle between the light source and the surface normal.) In contrast, image intensity changes due to motion of the shadows are typically large and discontinuous (see Figure 4). Shadowimages computed using our method are shown in Figure 7.

3 Previous Work: Terrain Scenes

In the typical shape-from-darkness formulation, it is assumed that the visible surfaces define a **terrain** (see Figure 5) [5, 3, 10]. A terrain is a scene in which surfaces are defined by a single depth map, $z(x)$. That is, all points below the surfaces are occupied. The key contribution of this paper will be to extend the problem beyond this condition.

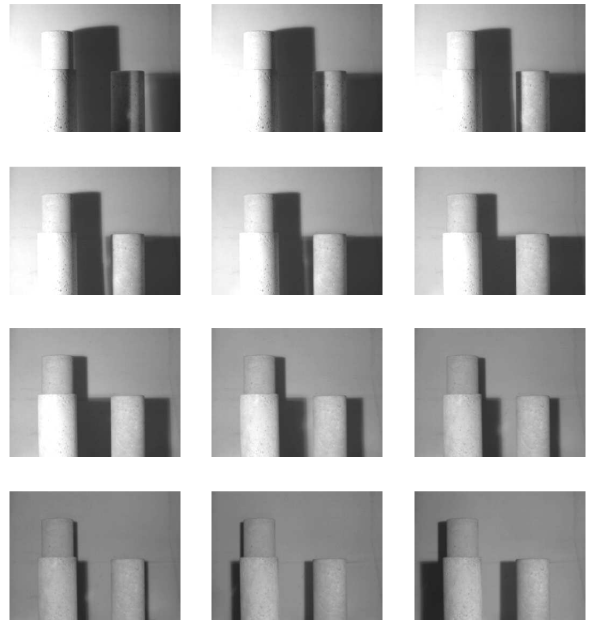

Figure 3: A sequence of grey-level images obtained using our robots. The scene consists of three cylinders in front of a wall of uniform albedo. One of the cylinders is behind another.

A second typical assumption is that the viewer is far from the visible surfaces, and that the optical axis of the viewer and the semicircle of the point source directions are coplanar[5, 7]. This reduces the computational problem by one dimension, namely, to 2-d scenes and 1-d images, or scanlines[5]. (Whether a surface point is in shadow or not depends on the occupancy of the scene within the plane defined by the point and the semicircle of light source directions.) This is the situation we assume as well.

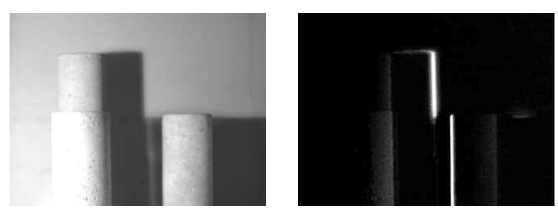

Figure 4: A grey-level image (left) and the difference between successive grey-level images (right).

Figure 5 shows a sketch of a **shadowgram**[7]. This indicates for each image coordinate, x, and for each light source direction, θ, whether or not the visible surface point, $(x, z(x))$, is in shadow. An alternative name for a shadowgram is a **suntrace**[5]. A shadowgram can be thought of as a stack of shadowimages, in which each shadowimage corresponds to a single source direction, θ. Algorithms for recovering a terrain from its shadowgram were presented in [5, 3, 7, 10].

The shadowgram of a terrain may be easily characterized. For each visible surface point $(x, z(x))$, there is a unique source direction, $\theta^-(x)$, at which the surface point goes from shadowed to unshadowed with increasing θ, and a second direction, $\theta^+(x)$, at which the point goes from unshadowed to shadowed with increasing θ. Notice that no surface points are shadowed when $\theta = \frac{\pi}{2}$ radians, since the source direction is then parallel to the optical axis (i.e. directly behind the camera). These **shadow boundaries** are marked in Figure 5.

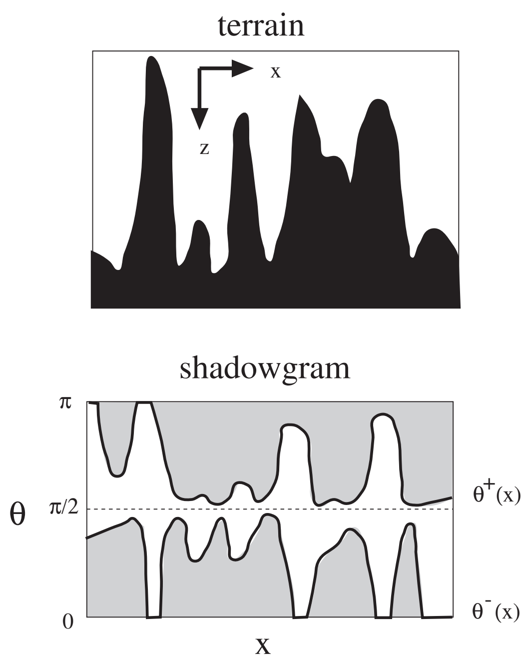

Figure 5: (upper) A 2-d terrain. By definition, all points below the surface are occupied. (lower) The shadowgram of this terrain. The grey regions represent values of (x, θ) at which a shadow occurs. The boundaries of the interval, $(\theta^-(x),\ \theta^+(x))$ are marked by a thick curve.

4 Non-terrain Scenes

When the scene is not a terrain, the shadowgram is more difficult to characterize[7]. Figure 8 shows a slice though a synthetic **non-terrain** scene, consisting of three cylinders in front of a wall, such that two of the objects are visible to the viewer and the third is occluded. Because the scene is not a terrain, light rays pass behind visible surfaces. As a result, points on the wall become unshadowed and shadowed several times as the light source moves through its semicircle. This is evident in the shadowgram. At pixel x_0's, the corresponding surface point is shadowed and unshadowed three times. A shadowgram computed from real grey level images is shown in Figure 7.

For certain non-terrains, hidden surfaces such as the back of objects may in principle be reconstructible from a shadowgram. The special case in which the back of a convex object casts its shadow onto a known reference surface was analyzed in [7], although no computational results for this special case were presented.

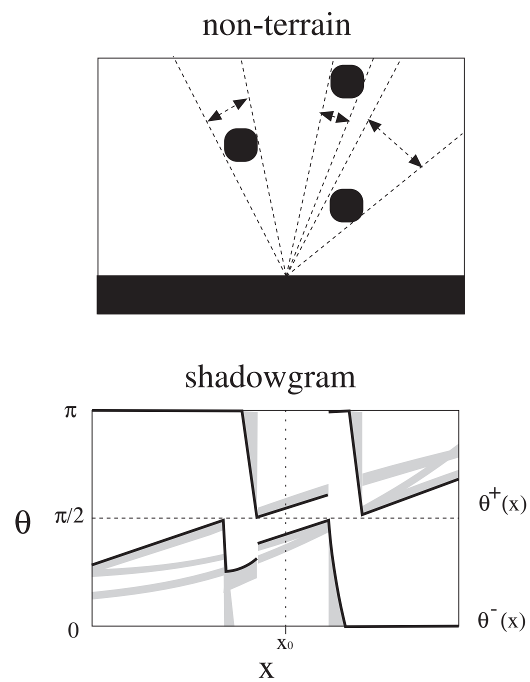

Figure 6: The shadowgram of a non-terrain is difficult to characterize since surfaces may be shadowed and unshadowed several times as the light source moves through its semicircular path. For example, point x_0 goes in and out of shadow three times.

In this paper, we extend the problem formulation to non-terrain scenes, and pose the question, "to what extent does the shadowgram of a non-terrain scene constrain the space occupancy of that scene?" We answer this question in two steps. First, we show how any previous shape-from-darkness algorithm can be used to recover the depth map of visible surfaces, even if the scene is not a terrain. Second, we provide a estimate of the occupied space hidden by the visible surfaces. This estimate is conservative in the sense that the true occupied space must be contained in the estimated occupied space.

Figure 7: A real scene consisting of three cylinders in front of a wall. Three grey-level images are shown, along with shadowgram of 45 computed shadowimages.

5 Generated Terrains

How does the shadowgram of a non-terrain constrain the depth map, $z(x)$, of visible surfaces? For any non-terrain scene, there is a unique terrain scene having the same depth map from a given viewpoint (we assume the viewing direction is $\hat{\mathbf{z}}$ throughout). This terrain scene is generated by filling in the space beyond the depth map, and is hereafter called the **generated terrain**. (see Figure 9.)

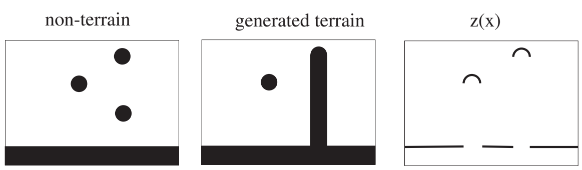

Figure 8: A non-terrain, its generated terrain, and the depth map of visible surfaces.

The shadowgram of the generated terrain is shown in Figure 9. A key observation follows which concerns the relationship between the shadowgram of a given non-terrain and the shadowgram of its generated terrain. The shadow boundaries, $\theta^-(x)$ and $\theta^+(x)$ for the generated terrain, are equivalent to those of the non-terrain. (Compare $\theta^-(x)$ and $\theta^+(x)$ in Figures 6 and 9.)

It follows that to compute the visible surfaces of a non-terrain from a given shadowgram, one need only consider the unshadowed region of the shadowgram containing $\theta = \frac{\pi}{2}$. Moreover, since a non-terrain and its generated terrain have identical visible surfaces, *any* previous shape-from-darkness algorithm may be used. This observation, that the previous techniques also apply to non-terrain scenes, is the first key result of the paper.

These ideas may be formalized as follows. Let $\mathcal{H}_{src}$ denote the semicircle of light source directions, parameterized by $\theta \in (0, \pi)$, where $\theta \in \{0, \frac{\pi}{2}, \pi\}$ correspond to "sunrise", "noon" (i.e. the viewing direction), and "sunset", respectively. For a point $\mathbf{x}$ in a scene, let $\mathcal{V}(\mathbf{x}) \subseteq \mathcal{H}_{src}$ denote the set of directions, called the **visibility field** at $\mathbf{x}$ [6], in which the source is visible from $\mathbf{x}$. In particular, let $\mathcal{V}_{surf}(x)$ denote the restriction of the visibility field to points on the visible surfaces. Observe that the surface visibility field, $\mathcal{V}_{surf}(x)$, corresponds exactly to the unshadowed region of a shadowgram.

For any point, $\mathbf{x}$, that lies on or above a visible surface, let $(\theta^-(\mathbf{x}), \theta^+(\mathbf{x}))$ denote the largest interval of $\mathcal{V}(\mathbf{x})$ containing the overhead direction, $\theta = \frac{\pi}{2}$, specifically,

$$\theta^-(\mathbf{x}) \;\equiv\; max\ \{\theta \;:\; 0 \leq \theta < \frac{\pi}{2} \;\text{ and }\; \theta \notin \mathcal{V}(\mathbf{x})\ \},$$

$$\theta^+(\mathbf{x}) \;\equiv\; min\ \{\theta \;:\; \frac{\pi}{2} < \theta \leq \pi \;\text{ and }\; \theta \notin \mathcal{V}(\mathbf{x})\ \}.$$

Similarly, $(\theta^-_{surf}(x), \theta^+_{surf}(x))$ is the largest interval of $\mathcal{V}_{surf}(x)$ containing $\theta = \frac{\pi}{2}$.

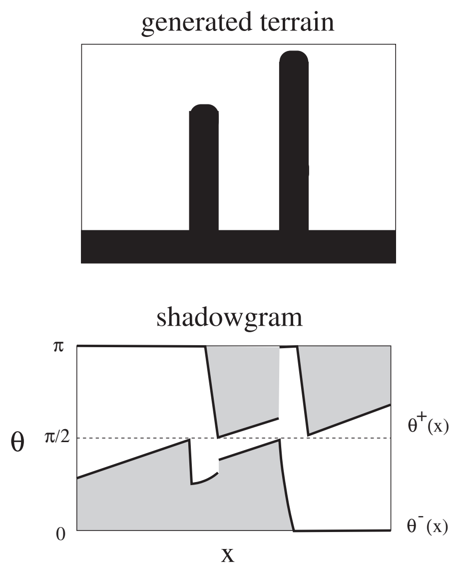

Figure 9: The generated terrain of the scene of Fig. 8. By definition, the occupied space below a non-terrain is a subset of that below its generated terrain The shadow boundaries, $\theta^-(x)$ and $\theta^+(x)$, for a non-terrain are identical to those of the generated terrain.

For a given surface point, $(x, z(x))$, the shadow boundaries, $\theta^-_{surf}(x)$ and $\theta^+_{surf}(x)$ define a cone, which is contained in free space. By definition, all visible surface points that are to the right of x lie either on the boundary of this cone or below the boundary. Those surface points that are on the boundary of the cone will be visible to the viewer. From this we have the following.

Proposition 1 *For a given scene, the shadow boundaries, $\theta^-(x)$ and $\theta^+(x)$, depend only on the depth map, $z(x)$.*

Proposition 2
The functions, $\theta^-_{surf}(x)$ and $\theta^+_{surf}(x)$, for a non-terrain scene are identical to those of the generated terrain scene.

The computational question of how to recover a depth maps from a shadowgram may thus be posed entirely in the context of terrain scenes.

6 Occupancy of Hidden Space

How does the shadowgram of a non-terrain constrain space occupancy of regions hidden by the visible surfaces? Observe that the occupied space behind the visible surfaces of a non-terrain is a subset of the occupied space behind the visible surfaces of its generated terrain. In particular, light rays from the source may pass behind surfaces of a non-terrain but not its generated terrain. It follows that the shadowed region of a non-terrain shadowgram is a subset of the shadowed region the generated terrain's shadowgram.

This observation provides the following constraint on space occupancy. If a visible surface point, $(x, z(x))$, were unshadowed for a source direction, θ, then there would have to be a light ray from the source arriving at that point in direction θ. If, moreover, $\theta \notin (\theta^-(x), \theta^+(x))$, then this ray would have to pass behind a visible surface. The set of all such source rays which pass behind visible surfaces carves out a free space behind visible surfaces.

Formally, for any x and for any $\theta \in \mathcal{V}_{surf}(x)$, the geometric ray,

$$\{\ (x, z(x)) + r(cos\theta, -sin\theta)\ \ :\ \ r \in \Re^+\},$$

must be entirely contained in free space. (Otherwise, a distant point source in direction $(cos\theta, -sin\theta)$ would not be visible from $(x, z(x))$, which is a contradiction.) The union of the trace of such geometric rays thus defines a free space $\mathcal{F} \subset \Re^3$, which is contained in the actual free space of the scene.

Proposition 3 *The set of points,*

$$\mathcal{F} \equiv \bigcup \{(x, z(x)) + r(cos\theta, -sin\theta)\ \ :\ \ \theta \in \mathcal{V}_{surf}(x),\ \ r \in \Re^+\}$$

is contained in free space.

In particular, no occupied space in the scene can intersect $\mathcal{F}$. It follows immediately that the set complement of $\mathcal{F}$ is a conservative estimate of the occupied space hidden behind the visible surfaces, in that the true occupied space must be contained in the estimated occupied space. This is the second key result of the paper.

7 Results

We represent space using an $N \times N$ square lattice, where N is the number of pixels in each scanline. Nodes in the space lattice are either OCCUPIED or FREE. Source light rays travel along linear trajectories through FREE nodes in the lattice[1]. The source moves through $2N$ directions, which correspond to the upper half of the perimeter of the space lattice.

To compute a depth map, $z(x)$, of visible surfaces from a given shadowgram, we examine all nodes at a given depth beginning with $z = 0$, and proceed incrementally. For each node, $\mathbf{x} = (x, z)$, we compute the interval $(\theta^-_z(x), \theta^+_z(x))$ by examining whether each light ray arriving at $\mathbf{x}$ has passed through, or directly below, any surfaces nodes at depths, $z' < z$. When the interval, $(\theta^-_z(x), \theta^+_z(x))$, has decreased to $(\theta^-_{surf}(x), \theta^+_{surf}(x))$, then $\mathbf{x}$ is a surface node. Otherwise, the search must continue to greater depths.

Figure 10 shows the results of this algorithm, applied to synthetic shadowgrams. The shapes of the surface in the reconstructed depth maps is quite similar to the originals (modulo an absolute depth difference). Note that the algorithm appears unstable when surface slopes are high, in particular at depth discontinuities. This instability will be discussed in detail in a future paper. For now, we suppose that the depth map can be accurately recovered, and proceed to the question of how to recover the geometry of the free space hidden by the visible surfaces. (Note that the depth map of visible surfaces could be recovered by an alternative imaging technique, such as laser.)

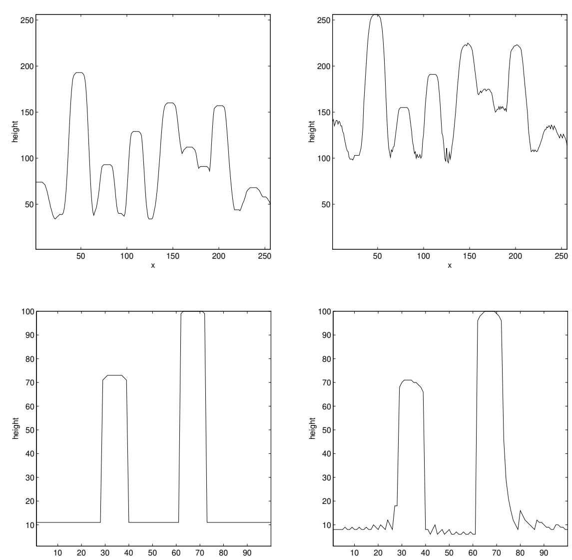

Figure 10: A set of terrain scenes is shown on the left. A shadowgram for each was computed, and depth were computed from these shadowgrams. (right). Apart from an absolute difference, the computed depth maps are almost identical to the originals. The RMS error between the actual and computed depth maps are (top) 3.0 which is 2 percent of the range of depths, and (bottom) 4.9 which is 5 percent of the range of depths. (see text for a discussion of errors).

[1]The trajectories are linear modulo integer roundoff errors.

In second step, we assume that both a shadowgram and depth map of visible surfaces are given. Recalling Proposition 3, since the source is visible in directions of $\mathcal{V}_{surf}(x)$, there cannot be any OCCUPIED nodes along rays that terminate at $(x, z(x))$ and that are in directions of $\mathcal{V}_{surf}(x)$. FREE nodes hidden by visible surfaces may be computed by tracing each source ray backwards through the space lattice, and freeing nodes along that ray. Figure 11 shows a set of non-terrains, along with the FREE nodes computed using this algorithm.

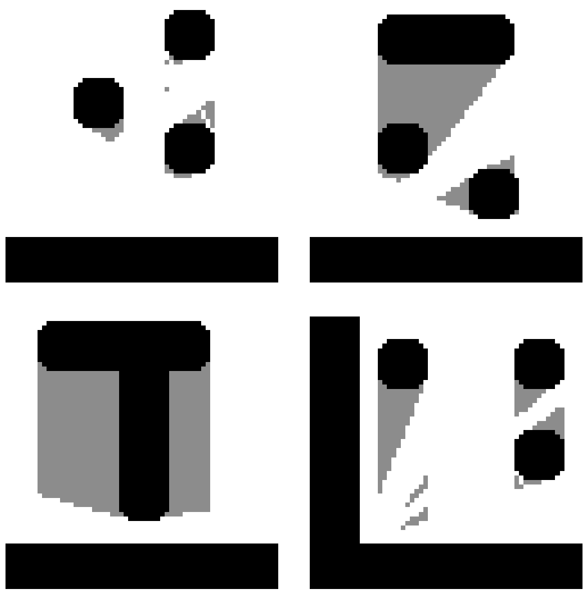

Figure 11: Four scenes are shown. The viewing direction is from above. The black regions represent actual OCCUPIED nodes. The white regions represent FREE nodes, which may be determined from the shadowgram. The light grey regions represent hidden regions whose occupancy is not determined by the shadowgram.

8 Discussion

We have proposed a two step solution to computing space occupancy from a sequence of shadow images (a shadowgram) taken at a single viewpoint. First, the depth map of visible surfaces is computed using the unshadowed region of the shadowgram containing the viewing direction. Second, a conservative estimate of the occupancy of the hidden space is recovered using the remainder of the shadowgram.

In practice, various errors have to be considered. The shadowgram is not directly available, but rather must be inferred from grey-level images. We have developed a method for robustly inferring a shadowgram from grey-level images, but it remains to verify this method for a non-point, proximal source.

Issues related to quantization of the source directions must also be addressed. Our results suggest that the recovery of the depth of visible surfaces may be unstable near occlusion boundaries. If this is indeed the case, one could compute the depth of visible surfaces using an alternative approach, such as laser range finding. The second part of the algorithm, determining occupancy of hidden space, could then be computed as we have proposed but now taking advantage of more reliable boundary data.

Acknowledgements

The authors would like to thank James Elder and Jonas August for their helpful comments, David Yang for pointing us to the work of [7, 8], and Yijen Huang for the images of Figure 7. This research was supported by grants from NSERC and AFOSR. S.W. Zucker is a Fellow of the Canadian Institute for Advanced Research.

References

[1] G. Dudek, M. Jenkin, E. Milios, and D. Wilkes. Experiments in sensing and communication for robot convoy navigation. (this proceedings).

[2] D. Forsyth and A. Zisserman. Reflections on shading. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 13:671–679, July 1991.

[3] M. Hatzitheodorou and J. Kender. An optimal algorithm for the derivation of shape from shadows. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, Ann-Arbour, Michigan, 1988.

[4] C. Jiang and M. Ward. Shadow segmentation and classification in a constrained environment. *CVGIP: Image Understanding*, 59(2):213–225, 1994.

[5] J. Kender and E. Smith. Shape from darkness: Deriving surface information from dynamic shadows. In *AIII*, pages 539–546, 1987.

[6] M. Langer and S. W. Zucker. Shape-from-shading on a cloudy day. *Journal of the Optical Society of America A*, 11(2):467–478, 1994.

[7] D. Raviv, Y. Pao, and K. Loparo. Reconstruction of three-dimensional surfaces from two-dimensional binary images. *IEEE Transactions on Robotics and Automation*, 5(5):701–710, 1989.

[8] D. Raviv, Y. Pao, and K. Loparo. Segmentation between overlapping parts: The moving shadows approach. *IEEE Transactions on Systems, Man, and Cybernetics*, 19(4):880–883, 1989.

[9] S. Shafer. *Shadows and silhouettes in computer vision.* Kluwer Academic Publishers, 1985.

[10] D. Yang and J. Kender. Shape from shadows under error. In *Image Understanding Workshop 1993*, pages 1083–1090, Washington, D.C., August 1993.