May 1992

Occlusions as a Guide for Planning the Next View

Jasna Maver
University of Pennsylvania

Ruzena Bajcsy
University of Pennsylvania

University of Pennsylvania Department of Computer and Information Science Technical Report No. MS-CIS-92-40.

Occlusions as a Guide for Planning the Next View

Abstract

To resolve the ambiguities that are caused by occlusions in images, we need to take sensor measurements from several different views. The task addressed in this paper deals with a strategy for acquiring 3-D data of an unknown scene. We must first answer the question: What knowledge is adequate to perform a specific task? Thinking in the spirit of purposive vision, to accomplish its task, a system does not need to understand the complete scene but must be able to recognize patterns and situations that are necessary for accomplishing the task.

We have limited ourselves to range images obtained by a light stripe range finder. A priori knowledge given to the system is the knowledge of the sensor geometry. The foci of attention are occluded regions, i.e., only the scene at the borders of the occlusions is modeled to compute the next move. Since the system has knowledge of the sensor geometry, it can resolve the appearance of occlusions by analyzing them.

The problem of 3-D data acquisition is divided in two subproblems due to two types of occlusions. An occlusion arises either when the reflected laser light does not reach the camera or when the directed laser light does not reach the scene surface. After taking the range image of a scene the regions of no data due to the first kind of occlusion are extracted. The missing data are acquired by rotating the sensor system in the scanning plane, which is defined by the first scan. After a complete image of the surface illuminated from the first scanning plane has been built, the regions of missing data which are due to the second kind of occlusions are located. Then the directions of the next scanning planes for further 3-D data acquisition are computed.

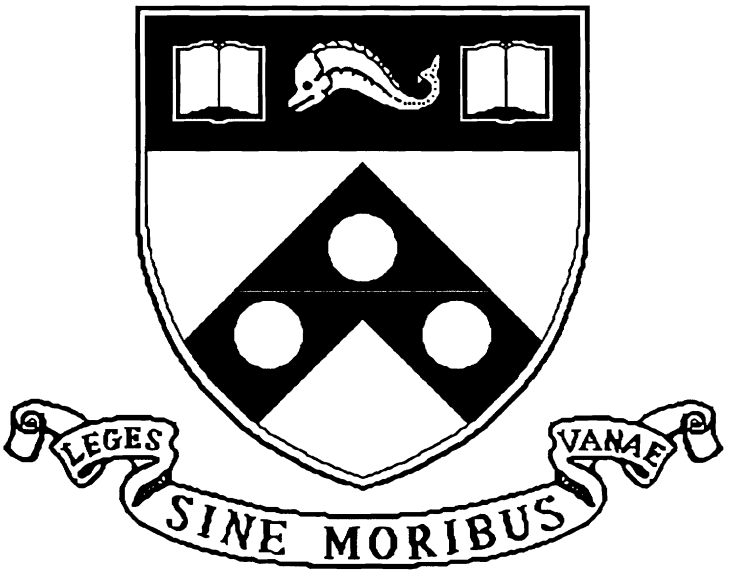
Comments

University of Pennsylvania Department of Computer and Information Science Technical Report No. MS-CIS-92-40.

Occlusions As A Guide For Planning The Next View

MS-CIS-92-40
GRASP LAB 318

Jasna Maver
Ruzena Bajcsy



University of Pennsylvania
School of Engineering and Applied Science
Computer and Information Science Department
Philadelphia, PA 19104-6389

May 1992

Occlusions as a Guide for Planning the Next View

Jasna Maver and Ruzena Bajcsy
GRASP Laboratory
Department of Computer and Information Science,
University of Pennsylvania, Philadelphia, PA 19104

Abstract

To resolve the ambiguities that are caused by occlusions in images, we need to take sensor measurements from several different views. The task addressed in this paper deals with a strategy for acquiring 3-D data of an unknown scene. We must first answer the question: What knowledge is adequate to perform a specific task? Thinking in the spirit of purposive vision, to accomplish its task, a system does not need to understand the complete scene but must be able to recognize patterns and situations that are necessary for accomplishing the task.

We have limited ourselves to range images obtained by a light stripe range finder. A priori knowledge given to the system is the knowledge of the sensor geometry. The foci of attention are occluded regions, i.e., only the scene at the borders of the occlusions is modeled to compute the next move. Since the system has knowledge of the sensor geometry, it can resolve the appearance of occlusions by analyzing them.

The problem of 3-D data acquisition is divided in two subproblems due to two types of occlusions. An occlusion arises either when the reflected laser light does not reach the camera or when the directed laser light does not reach the scene surface. After taking the range image of a scene the regions of no data due to the first kind of occlusion are extracted. The missing data are acquired by rotating the sensor system in the scanning plane, which is defined by the first scan. After a complete image of the surface illuminated from the first scanning plane has been built, the regions of missing data which are due to the second kind of occlusions are located. Then the directions of the next scanning planes for further 3-D data acquisition are computed.

1 Introduction

Existing sensing equipment acquires in one measurement only a portion of the information present in a 3-D working domain. To resolve the ambiguities that are caused by occlusions in images, we need to take sensor measurements from several different views. An intelligent sensor system should be able to automatically determine where and how to position itself in order to obtain yet unknown 3-D information of the scene.

The task addressed in this paper deals with the strategy of acquiring 3-D information of the scene of interest. The problem of data acquisition can be classified according to different criteria, for example, sensor detectability, the type of environment (indoor or outdoor), and the task. Depending on how much a priori geometrical information about the scene is available, the following classification is possible:

1. The complete geometrical information about the scene is known.

2. A set of models for objects that can occur in the scene is given.

3. No a priori geometrical information about the scene is available.

The first kind of problem is a familiar question in computational geometry. An example in two dimensions is the well-known "Art Gallery Problem" [5,7,12]. Supposing that the polygon P represents the floor plan of a building, find a bound on the number of guards such that the entire interior of P is visible. In the case when the complete geometrical information about the target object and the obstacle object is known the problem is to compute the "best" locations and directions for sensors placement [10,11,13]. The term "best" can mean the minimal number of observation locations, as in the "Art Gallery Problem". It can also mean computing the observation locations and directions for sensor placement from which some region or feature can be observed best. Cowan et al. [4] analyzed constraints on camera placement, such as resolution, focus, field-of-view, visibility and conditions for light source placement.

In [10,11] the authors deal with the problem of where to place the sensors such that the working place can be best observed for accomplishing a manipulation task. Vision planning functions in their hand-eye system are classified into global vision and local vision, similarly to a classification of gross and fine manipulation. On a geodesic dome generated around a target object model they first determine occlusion free viewpoints for a camera-in-hand in order to avoid occlusions caused by surrounding objects. For fine motion manipulation, closed-loop visual feedback is employed, restricted to some specified regions within the field-of-view where the focus of attention is defined.

The second kind of data acquisition problem arises when the system's current knowledge about the world allows different interpretations of the data. A situation like this can occur in 3-D object

recognition when the information in one view is not rich enough to extract the minimum number of features for a unique object identification. In order to recognize the object, additional sensing operations are required. This kind of process has usually three major steps: identification — grouping data into features to identify the object and to estimate its pose (position and orientation); prediction — computing the geometry of the occluded scene based on the candidate models; optimization — computing a predefined optimization criterion or strategy which yields the next sensing operation.

An instance of such a problem is addressed in [6]. After taking a measurement, hypotheses about the identity and position of an object in the scene are represented by a set of matches between sensed and model features. For each possible object, the system knows the corresponding aspect graph. An aspect is defined as a set of features which can be observed simultaneously from a single viewpoint. The aspect graph groups viewpoints with the same aspect into equivalent classes. This property of the aspect graph is exploited in the phase of object identification and in the phase of predicting the set of object features which would be observed from the selected viewpoint. The system automatically proposes a sensing operation and then determines the maximum ambiguity which might remain in the scene description if that sensing operation was applied. The sensing operation which minimizes this ambiguity is then applied. In the real world, where objects are not only self-occluded but can also be occluded by other objects, the aspect graph cannot be defined in advance. Occlusions and features of other objects in the scene change the classes of viewpoints and sets of features belonging to the object aspects. Identification becomes difficult not only because the sets of features change but also because it is not clear which features belong to the same object.

The third type of data acquisition problem arises when a sensor system meets an unknown environment which it has to investigate with its sensors—the task of autonomous exploration. In the literature [1,3] several different strategies for controlling the sensing operations are proposed. The set of sensor actions can be predefined, as in the work of N. Ahuja and J. Veenstra [1]. An octree of the scene is constructed from the orthographic projections of the scene onto a plane perpendicular to the viewing direction. Thirteen views must always be selected from any subset of directions, corresponding to the three "face" views, six "edge" views, and four "corner" views of an upright cube. In [3] the next viewing directions are computed from the partial octree model of the scene. Two different strategies are reported in the paper. The first one, named the "Planetarium Algorithm", sets up a sphere around the scene and estimates for each sample point how much of the yet unknown area can be eliminated. Unfortunately the algorithm is inefficient because the estimation is computed for each sampling point on the sphere. Also the selection of the direction from which the largest unseen volume can be observed does not necessarily lead to the minimal number of views. The second algorithm, named the "Normal Algorithm", determines the visibility

conditions from within the tree. The algorithm is much faster but does not deal with self-occluding scenes as well as the first one does. In both papers [1,3] the algorithms are tested only on simulated data.

Another way to interpret the current data is with volumetric models, for example superellipsoids. This approach is described in the work of P. Whaite and F. P. Ferrie [16]. After the data are segmented, the task is to find superquadric parameters that describe each segmented part best. The solutions obtained by using the Levenberg-Marquardt method show an inherent lack of uniqueness. The next viewing directions are computed to disambiguate the solutions of the current interpretation.

The problem described in this paper can be ranked into the third group of the above classification. We have limited ourselves to range images obtained by a laser-camera triangulation system which can measure the distance only of those portions of a 3-D scene which are simultaneously illuminated by the laser light and visible to the camera. Two types of occlusions can be encountered in a range image. An occlusion arises either when the reflected laser light does not reach the camera or when the direct laser light does not reach the scene surface.

The task of our sensor system is to acquire yet unknown 3-D information of the scene of interest. We must first answer the question: What knowledge is adequate to perform the task? Thinking in the spirit of purposive vision [2], to accomplish its task, a system does not need to understand the complete scene but must be able to recognize patterns and situations that are necessary for accomplishing the task. A priori knowledge given to our system is the knowledge of the sensor geometry. The foci of attention are occluded regions, i.e., only the scene at the borders of the occlusions is modeled to compute the next move. Since the system has the knowledge of the sensor geometry, it can resolve the appearance of occlusions by analyzing them.

The problem of 3-D data acquisition is divided into two subproblems due to two types of occlusions. After taking the range image of a scene, the regions of no data due to the first kind of occlusion are extracted. The missing data are acquired by rotating the sensor system in the scanning plane, which is defined by the first scan. After a complete image of the surface illuminated from the first scanning plane has been built, the regions of missing data which are due to the second kind of occlusions are located. Then the directions of the next scanning planes for further 3-D data acquisition are computed.

The remainder of the paper is organized as follows. In section 2 we describe the sensor system and in section 3 we outline the two-stage data acquisition process. The problem definition of acquiring the data from the first scanning plane and a proposal to its solution are described in section 4. The computation of the next scanning planes is presented in section 5. Finally, section 6 presents experimental results.

2 The sensing system

The depth of the scene is measured by a camera-laser triangulation system [15] (Fig. 1). A CCD-camera and a laser are coupled and form a fixed sensor unit. The laser beam is spread into an illuminating plane. The relative relationship between the camera image plane and the illuminating plane is fixed. The illuminating plane intersects with the object surface, forming a planar curve (laser-stripe). Each point on the curve is mapped to a single point in the camera image plane. The

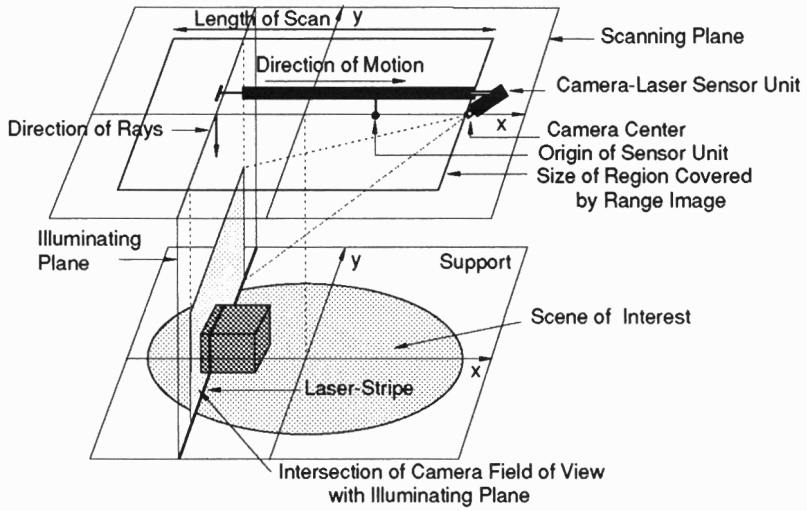


Figure 1: Sensing.

distance between the point on the curve and the camera center is determined by the intersection of the line which goes through the camera center and the point on the camera image plane with the illuminating plane. The range image is obtained by scanning the scene with a series of parallel illuminating planes. This can be achieved by moving the scene support with a constant velocity perpendicular to the illuminating plane or by moving the origin of the camera-laser sensor unit in a plane called the *scanning plane*. The scanning plane is orthogonal to the illuminating plane, i.e., the normal of the scanning plane and the direct laser rays of the illuminating plane have the same direction. The distance measurements are transformed into the distances between the illuminated points of the surface and the scanning plane, and stored in the range image. Higher/lighter values in the range image correspond to smaller distances, lower/darker to greater distances.

We assume that the range images are acquired by moving the origin of the camera-laser sensor unit in the scanning plane from left to right, with the camera to the right of the laser-stripe. The y-axis on the range image corresponds to the laser-stripe and the x-axis corresponds to the size of the shift. The center of the range image corresponds to the center of the scene of interest.

2.1 Occlusions

The described sensor system can only measure the distance of those portions of a 3-D scene which are simultaneously illuminated by the laser light and visible to the camera. Since the direction/position of the direct laser rays of the illuminating plane is different than the viewing direction/position of the camera, two types of occlusions can be encountered during the scanning:

1. A *camera occlusion* arises when a part of the illuminated surface in the scene is occluded to the camera by another part of the scene.

2. A *light occlusion* arises when the direct laser light does not reach a part of the surface in the scene because it is reflected from another part of the scene.

3 Two-stage data acquisition process

After the view-scan has been taken, the surface points of the scene of interest can be classified into three sets (Fig. 2):

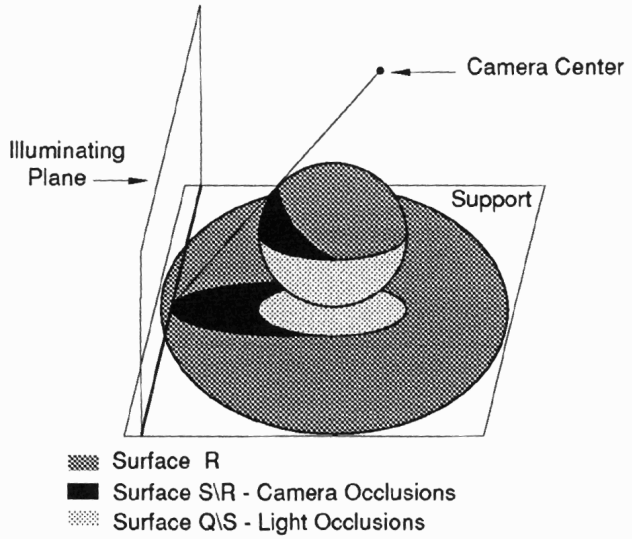


Figure 2: Surface $\mathcal{R}, \mathcal{S} \setminus \mathcal{R}$—camera occlusions, and $\mathcal{Q} \setminus \mathcal{S}$—light occlusions.

- All the surface points in the scene belong to the set $\mathcal{Q}$.

- The surface points which were illuminated during the scanning belong to the set $\mathcal{S}$, $\mathcal{S} \subseteq \mathcal{Q}$. Note that the set $\mathcal{S}$ depends on the given scanning plane. The points of the set $\mathcal{Q} \setminus \mathcal{S}$ belong to the light occlusions.

- The surface points which were illuminated by the light and seen by the camera during the scanning define the set $\mathcal{R}$, where $\mathcal{R} \subseteq \mathcal{S}$. Note: $\mathcal{R}$ depends on the orientation of the camera-laser sensor unit in the scanning plane. The set $\mathcal{S} \setminus \mathcal{R}$ represents the camera occlusions.

The task of 3-D data acquisition is decomposed into two subtasks.

The first task: The selected set of views must be such that

$$\bigcup_i \mathcal{R}_i = \mathcal{S},$$

where R_i denotes the points of $\mathcal{S}$ seen by the camera from i-th view. Such a set of views can be acquired by rotating the camera-laser unit in the fixed scanning plane. In each view the illuminating plane illuminates the same part of the scene of interest, but the camera sees different parts.

The acquired range image of i-th view represents the discrete depth map of the surface $\mathcal{R}_i$. The discrete map of $\mathcal{S}$ can be obtained by merging the range images acquired from different rotations of the camera-laser sensor unit into a common coordinate plane called the **I***mage of the* **S***canning* **P***lane*—ISP (see experimental results, Figs. 18(a), 25(a)). The center of ISP corresponds to the center of the scanning plane. The x-axis and y-axis of the ISP correspond to the x and y axis of the scanning plane, respectively. The orientation of the camera-laser sensor unit in the scanning plane is defined by a *scanning direction* φ. The scanning direction φ is the angle of counterclockwise rotation of the camera center in the scanning plane about the origin of the camera-laser sensor unit measured from the x-axis of the scanning plane (Fig. 1 shows a situation for the scanning direction $\varphi = 0°$). The range images are transformed into the ISP by rotating each of them counterclockwise about its center by an angle of the scanning direction. The point (x, y) in the range image is transformed into the point $(x, y)_{ISP}$ in the ISP by the following transformation:

$$\begin{bmatrix} x \\ y \end{bmatrix}_{ISP} = \begin{bmatrix} \cos(\varphi) & -\sin(\varphi) \\ \sin(\varphi) & \cos(\varphi) \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

The second task: From the complete depth information of the surface illuminated from the first scanning plane, the orientations of the new scanning planes must be computed so that

$$\bigcup_j \mathcal{S}_j = \mathcal{Q},$$

where $\mathcal{S}_j$ denotes the points of the surface $\mathcal{Q}$ illuminated during the scanning from the j-th scanning plane.

4 Gathering the complete depth data of the scene surface S

The first task of the sensor system is to find orientations of the camera-laser sensor unit in the fixed scanning plane from which the camera can acquire the complete depth data of the surface $\mathcal{S}$

illuminated from that scanning plane. Since no a priori geometrical information about the scene is given, the first view can be taken from an arbitrary scanning direction. After the first range image has been obtained two cases are possible:

1. The complete depth data of the surface $\mathcal{S}$ is acquired.

2. During the scanning some parts of the surface $\mathcal{S}$ were occluded to the camera and their depth not stored in the range image (see experimental results, black regions in the range images (Figs. 15(a), 17(b), 17(c)) are camera occlusions—*range shadows*).

In the second case the sensor system must acquire additional range images to obtain yet unknown depth data of the surface $\mathcal{S}$. New orientations of the sensor unit can now be computed from the data stored in the first range image.

For the sake of convenience we introduce the *support plane.* The support plane is a plane on which the set of objects that form the scene lies. This plane is parallel to the scanning plane as shown in Fig. 3. For each point (x_i, y_i) in the support plane there exists one point $s_i \in \mathcal{S}$ of the surface in the scene which is illuminated by the laser light during the scanning. The height of s_i relative to the support plane is denoted as $h(x_i, y_i)$.

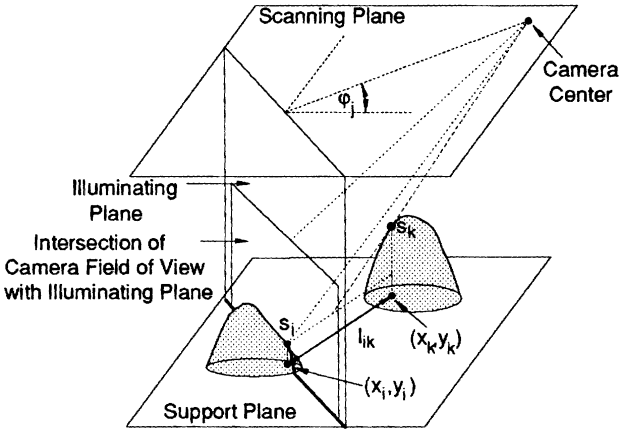


Figure 3: Can the surface point s_i be acquired from the scanning direction φ_j?

Suppose that we know the shape/height of $\mathcal{S}$, and since we do not have any other information about the scene we assume that the scene is fully occupied from the surface $\mathcal{S}$ down to the support. In order to answer the question (Fig. 3): *"Can the surface point s_i be acquired from the scanning direction φ_j?"* we must examine the scene along this direction[1]. For each surface point s_k along direction φ_j we test if it lies below the line connecting the point s_i with the camera center at the

[1]Here we assume that all the points of the laser-stripe are seen by the camera from the same direction φ_j. This can be assumed only if the radius of the scene of interest is small in comparison to the distance between the camera center and the illuminating plane.

moment when s_i is illuminated by the laser light. If the inequality (Fig. 4)

$$h(x_k, y_k) < h(x_i, y_i) + l_{ik} \tan(\alpha(h(x_i, y_i))) \tag{1}$$

is true for all k then the image of s_i can be acquired from direction φ_j. In (1) the angle α changes with the height $h(x_i, y_i)$ by the following equation:

$$\alpha(h(x_i, y_i)) = \arctan(\frac{H - h(x_i, y_i)}{L}), \tag{2}$$

where L represents the fixed distance between the camera center and the illuminating plane[2] and H is the largest depth which can be detected by the sensor.

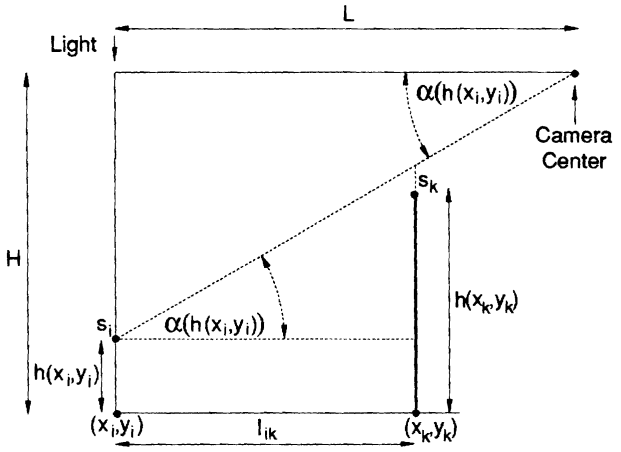


Figure 4: The angle α changes with the height $h(x_i, y_i)$.

To test the inequality (1) we have to know the height of both points, $h(x_i, y_i)$ and $h(x_k, y_k)$. If s_i was occluded in the first scan, $h(x_i, y_i)$ is not known. Suppose that in the acquired range image we can determine the obstacle $(s_o, h(x_o, y_o))$ which occluded the point s_i in the first scan. For the height of s_i (Fig. 5), we can derive the inequality

$$h(x_i, y_i) < H - L\frac{H - h(x_o, y_o)}{L - l_{io}}, \tag{3}$$

where l_{io} represents the distance between (x_i, y_i) and (x_o, y_o). Unfortunately, the information about the height $h(x_i, y_i)$ in (3) is not sufficient to be used in (1). Thus, without knowing the height $h(x_i, y_i)$ we can not give the exact answer to the above question.

To accomplish the task the sensor system should look for that information in the acquired data which is adequate to create a representation of the occluded parts. Our sensor system exploits the information of the surface height in the neighborhood of the range shadows, for the reasons that:

[2]In (2) the distance L should be different for different points of the laser-stripe; the smallest for the midpoint of the laser-stripe and the largest for the endpoints of the laser-stripe. If the radius of the scene of interest is small in comparison to L then the same L can be assumed for all the points of the laser-stripe.

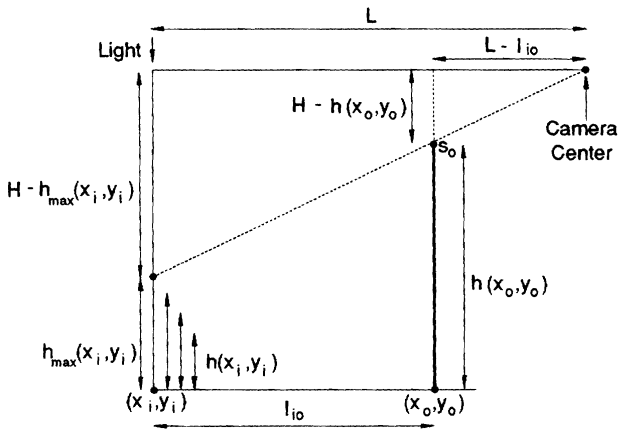


Figure 5: The sketch of the inequality (3).

- The closer the surface point s_k is to s_i the smaller its height $h(x_k, y_k)$ has to be in order not to occlude s_i (inequality (1)). Thus, for the occluded surface points in the range shadow the height of the points in the neighborhood of the range shadow must be tested the first. In other words, if some part of the scene at the border of the range shadow occludes the surface point s_i in the range shadow, then we do not need to examine the scene behind that part.

- The outline of the range shadow is a product of the sensor geometry, i.e., the position/direction of the light and the observer and the shape of the scene. The sensor geometry is known and the outline of each range shadow can be obtained. Since we do not have any a priori knowledge of the scene it is reasonable to assume that the height of the surface points in the range shadows is equal to the height of the surface points in the neighborhood of the range shadow when the sensor geometry and the shape of the outline of the range shadow do not contradict this assumption.

4.1 Detection and representation of camera occlusions

In the acquired range image we must first locate range shadows. The sensor system is calibrated so that the height of the support is greater than zero. This enables the detection of range shadows with zero height in the range image, i.e., black areas (see experimental results).

We approximate the range shadows by polygons. The contours[3] of the shadow areas are separately represented as contour descriptors $x(s)$ and $y(s)$. Each contour is segmented into a series of straight lines. The breakpoints on the contour are points of maximum curvature.

[3]The contours of the range shadows in the image are found by Pavlidis' algorithm TRACER [9].

4.1.1 Points of maximum curvature

To find the points of maximum curvature, the contour descriptors $x(s)$ and $y(s)$ are first smoothed using a Gaussian filter to filter out noise. The derivatives $\frac{dx(s)}{ds}$ and $\frac{dy(s)}{ds}$ are computed to get the tangent angle function as

$$\phi(s) = \arctan(\frac{dy/ds}{dx/ds}).$$

Special processing is required to handle phase wrapping. The breakpoints on the contour are obtained from the curvature function, which is the derivative of the tangent angle function

$$\kappa(s) = \frac{d\phi(s)}{ds}.$$

The positive maxima and negative minima of the curvature function $\kappa(s)$ represent the convex and concave vertices of the polygon.

4.1.2 Polygon representation

The vertices v_i of the polygon $P(v_1, v_2, \ldots, v_n)$ which approximates a range shadow are ordered clockwise so that, when moving from vertex v_i to vertex v_{i+1}, the range shadow is on the right side of the edge $e_i(v_i, v_{i+1})$. For each edge $e_i(v_i, v_{i+1})$ in polygon P, an angle $\theta(e_i)$ (Fig. 6) in the range image is computed by the next equation:

$$\theta(e_i) = \arctan(\frac{y_i - y_{i+1}}{x_i - x_{i+1}}).$$

(x_i, y_i) and (x_{i+1}, y_{i+1}) are the coordinates of the vertices v_i and v_{i+1} in the range image, respectively. The angle of the same edge $e_i(v_i, v_{i+1})$ in the ISP is computed by adding the angle of the scanning direction φ to $\theta(e_i)$:

$$\theta_{ISP}(e_i) = \theta(e_i) + \varphi.$$

4.1.3 Height of the border

The height of the scene in the neighborhood of the range shadow is approximated by a *constant* for each edge $e_i(v_i, v_{i+1})$. The value of the constant is defined as the median of the pixel heights belonging to the pixels in the chain between the vertices v_i and v_{i+1}. Around range shadows are narrow zones where the pixel heights are unreliable. Therefore, the pixel height for each pixel in the chain is found by searching in its close neighborhood. For each pixel the search is done for k pixels in three directions. The directions of the search are determined by the change (dx, dy) in the x-and y-coordinates of the new pixel in the chain (Fig. 7). The highest value found in the search is chosen as the pixel height.

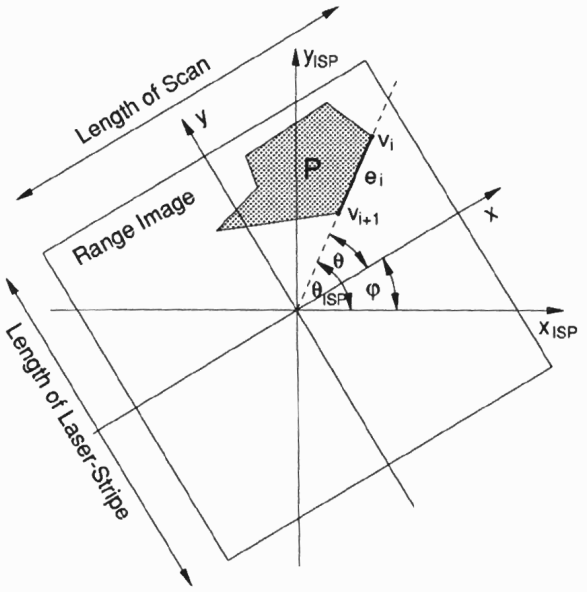


Figure 6: Angle $\theta(e_i)$ and $\theta_{ISP}(e_i)$ of the edge e_i.

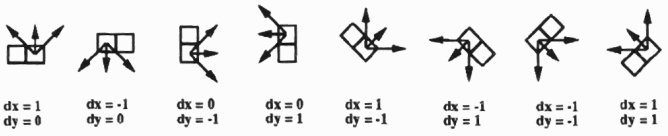


Figure 7: Searching directions for the pixel height.

4.2 Analysis of range shadows

For each range shadow, we would like to determine which part of the scene in its neighborhood caused an occlusion of the range shadow and which part could occlude the current range shadow when looking at the scene from any other direction. Taking into account the sensor geometry and assuming that the scene is fully occupied from the surface $\mathcal{S}$ down to the support, two qualitative properties, *occlusion* and *activity*, are determined for each edge of polygon P, representing the range shadow and the scene in its close neighborhood:

1. *Occlusion:* This property follows from the sensor geometry. A range image is acquired by moving the sensor unit perpendicular to the illuminating plane from left to right with the camera to the right of the illuminating plane. An edge $e_i(v_i, v_{i+1})$ with the angle $\theta(e_i) \in (0, \pi)_{ccw}$ is an *occluding edge* e_i^{occ} and belongs to the set of occluding edges, $\mathcal{O}$.

   If lines parallel to the x-axis, the direction of motion during the scanning, are drawn through the end points of the occluding edges in polygon P, the polygon is cut into areas A_i, as shown in Fig. 8. For each edge e_i^{occ} in P there is a unique A_i and vice versa. The edge e_i^{occ} causes

an occlusion of the area A_i. Polygon P can be defined in the following way:

$$P(v_1, v_2, ..., v_n) = \bigcup_{\{i; e_i^{occ} \in \mathcal{O}\}} A_i.$$

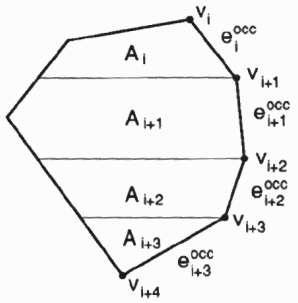


Figure 8: Areas A_i belonging to occluding edges.

2. *Activity:* The parts of the scene which can cause a partial or a complete occlusion of the polygon P, when looking at the scene from an arbitrary direction, are called active. Assume that e_i^{occ} is an occluding edge to which the area A_i is assigned, and e_j is one of the edges which limit this area A_i on the opposite side. We test if the edge e_i^{occ} is high enough to occlude all the points of the range shadow between e_i^{occ} and e_j supposing that the height of the points is $h(e_j)$. If the answer is "no" then the height of these points must be less than $h(e_j)$, since they do lie in the occlusion. The edge e_j (the scene in its close neighborhood) can occlude the points which lie close to it and whose height is less than $h(e_j)$ for scanning directions from the interval $(\theta(e_j) - \pi, \theta(e_j))_{ccw}$. Therefore, we mark the edge e_j as an *active edge*. In the algorithm we test only the midpoint of e_j. For each area A_i, the edges which limit the area are tested, using (1), as follows:

   If $e_j \neq e_i^{occ}$ and e_j limits A_i, then

$$e_j = \begin{cases} \text{inactive;} & \text{if} \quad h(e_i^{occ}) \geq h(e_j) + l_{ji} \tan(\alpha(h(e_j))) \\ \\ \text{active;} & \text{if} \quad h(e_i^{occ}) < h(e_j) + l_{ji} \tan(\alpha(h(e_j))) \end{cases}$$

   where $j = 1, ..., n$; l_{ji} and $\alpha(h(e_j))$ are measured at the midpoint of the edge e_j. The active edges form the set $\mathcal{A}$. Every occluding edge is also an active edge, so $\mathcal{O} \subseteq \mathcal{A}$. Fig. 9 shows an example.

Given these properties, we say that pixels in a range shadow can be occluded only by active edges. This statement implies that changes in surface height inside and outside the range shadows are so small that they cannot cause occlusions.

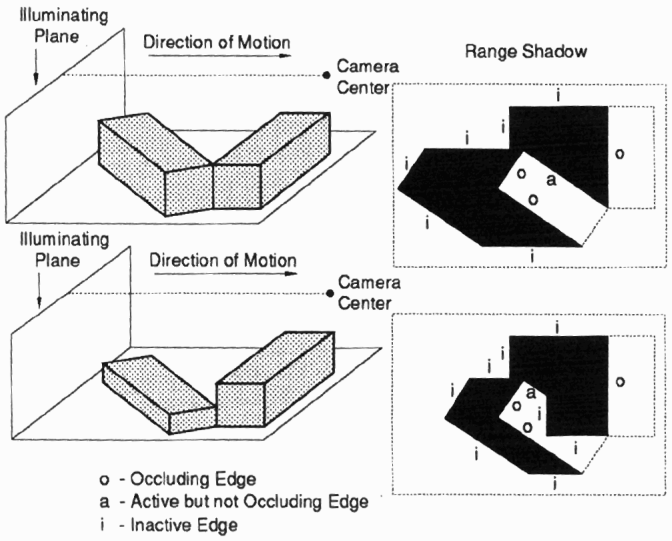


Figure 9: Definition of edges.

From the properties defined above, an assumption about the height of the occluded pixels will be made in the next section. If the scene is not as we haved assumed, i.e., the scene is not fully occupied from the surface $\mathcal{S}$ down to the support, then some edges can be wrongly classified as occluding or active. In these cases a solution is obtained which sets the height of the pixels between wrongly classified edges to the height of the support.

4.3 Viewing angles computation

For each pixel p_k in a polygon P, we define the *viewing angle* $\Phi(p_k)$ as the sectors containing all scanning directions from which the pixel p_k is visible, assuming, as stated above, that only active edges can cause occlusions. To determine in the set $\mathcal{A}$ those active edges which occlude p_k an assumption about the height of p_k has to be made.

Suppose that for the height of pixel p_k the weakest assumption is made, i.e., the highest possible value for $h(p_k)$ is assumed according to (3). In this case the computed viewing angle $\Phi(p_k)$ can be equal or greater than the true one. This assumption can lead to viewing directions from which the pixel p_k is still occluded. If, on the other hand, we make the assumption about the pixel height more restrictive, we can lose some or even all valid viewing directions, leading to a non-optimal solution for all the occluded pixels. Suppose that for a set of pixels there exists a direction from which all can be seen simultaneously. Too restrictive an assumption can lose this common viewing direction for all the pixels so that more viewing directions than really necessary will be selected.

In our algorithm we adopt a compromise, knowing that a less restrictive assumption can always be made if no solution is found for any pixel.

Although the polygons P and the areas A_i into which the polygons are split are concave in

general, the areas A_i have a constrained shape that makes it easier to find their pixels. Since each area A_i has precisely one occluding edge e_i^{occ}, all the pixels in A_i are on the right side of e_i^{occ}, going from v_i to v_{i+1}. By starting with pixels on the occluding edge and decreasing the x-coordinate until the edge on the opposite side is reached, all the pixels in area A_i are found.

For pixels p_k lying between the occluding edge e_i^{occ} on their right and the edge e_j on their left we define their height in the following way:

$$h(p_k) = \begin{cases} h(e_j); & \text{if } e_j \notin \mathcal{A} \\ \\ h_{support}; & \text{if } e_j \in \mathcal{A} \ . \end{cases}$$

To determine the viewing angle $\Phi(p_k)$ of pixel p_k, the directions from which it is occluded are computed first. For each active edge $e_i \in \mathcal{A}$ an *occluding angle* $\Phi_{occ}(p_k, e_i)$ is computed. Knowing the height of the pixel $h(p_k)$ and the height of the active edge $h(e_i)$, the distance r, from which the obstacle of the height $h(e_i)$ can occlude p_k, is derived from (1) and (2):

$$r = L \, \frac{h(e_i) - h(p_k)}{H - h(p_k)}.$$

If e_i or a part of it lies in the circle around pixel p_k with radius r (Fig. 10) then occluding angle

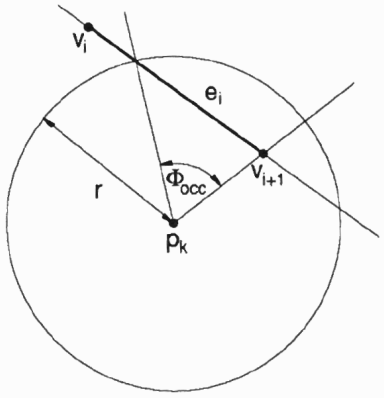


Figure 10: Occluding angle computation.

$\Phi_{occ}(p_k, e_i)$ is defined by the lines intersecting in p_k and going through the endpoints of the part of the edge e_i inside the circle. If φ is the scanning direction of image acquisition, then by adding φ to the bounds of the occluding angle $\Phi_{occ}(p_k, e_i)$ the angle $\Phi_{occ}(p_k, e_i)$ is transformed to ISP. The directions in $\Phi_{occ}(p_k, e_i)_{ISP}$ correspond to the orientations of the sensor unit in the scanning plane only if p_k lies in the center of the range image. If the radius of the scene of interest is small in comparison to the distance between the camera center and the illuminating plane L, then the directions in the occluding angle $\Phi_{occ}(p_k, e_i)_{ISP}$ can be equalized with the orientations of the sensor unit.

The viewing angle $\Phi(p_k)$ is computed by subtracting all the occluding angles $\{\Phi_{occ}(p_k, e_i)_{ISP}; e_i \in \mathcal{A}\}$ from the unit circle. The viewing angles are finite sets of closed intervals—*viewing intervals*

$$\Phi(p_k) = \{[\varphi_{j_l}, \varphi_{j_u}], [\varphi_{j+1_l}, \varphi_{j+1_u}], \ldots\},$$

where the $\varphi_{j_l}, \varphi_{j+1_l}, \ldots$ and $\varphi_{j_u}, \varphi_{j+1_u}, \ldots$ are the lower and upper bounds of the viewing intervals, respectively. The viewing angle $\Phi(p_k)$ can also be empty, which means that the pixel p_k cannot be seen from this scanning plane.

4.4 Determination of the next scanning direction

After computing the viewing angles, the next scanning directions must be determined. If there exists a direction from which the whole occluded scene can be seen at once, this direction must appear in all viewing angles. We have to intersect the viewing angles and choose a scanning direction from the global intersection if it is nonempty, or from several partial intersections if otherwise. The problem can be formulated in the following way: *Having finite sets of closed intervals*

$$\Phi(p_k) = \{[\varphi_{1_l}, \varphi_{1_u}], [\varphi_{2_l}, \varphi_{2_u}], \ldots\} \textit{ for each pixel } p_k, k = 1, \ldots, m,$$

find intervals $\Omega_i = [\omega_{i_l}, \omega_{i_u}]$, *with* $i = 1, \ldots, n_\Omega$, *such that forming a set* $\mathcal{U} = \{\varphi_i; \varphi_i \in \Omega_i, i = 1, \ldots, n_\Omega\}$, *for every* $\Phi(p_k)$ *there exist* $[\varphi_l, \varphi_u] \in \Phi(p_k)$ *and* $\varphi \in \mathcal{U}$ *such that* $\varphi \in [\varphi_l, \varphi_u]$, *and their number* n_Ω *is minimal.*

The viewing intervals $\mathcal{I}_i = [\varphi_{i_l}, \varphi_{i_u}]$, where $\mathcal{I}_i \in \bigcup_{k=1}^{m} \Phi(p_k) = \mathcal{V}$, have the following properties:

- $\forall \mathcal{I}_i \in \mathcal{V} : \mathcal{I}_i \subset [0, 2\pi)$, since each pixel in the range shadow is occluded at least from one direction, i.e., direction of image acquisition.
- Any pair of intervals $\mathcal{I}_i, \mathcal{I}_j \in \Phi(p_k)$ has no element in common:

$$[\varphi_{i_l}, \varphi_{i_u}] \cap [\varphi_{j_l}, \varphi_{j_u}] = \emptyset \quad \forall i \neq j.$$

For each $\mathcal{I}_i = [\varphi_{i_l}, \varphi_{i_u}]$, a function $\xi_i(\varphi)$ on the interval $[0, 2\pi)$ is defined:

$$\xi_i(\varphi) = \begin{cases} 1; & \text{if } \varphi \in [\varphi_{i_l}, \varphi_{i_u}] \\ 0; & \text{otherwise .} \end{cases}$$

The interval $[0, 2\pi)$ can be represented by the circumference of a unit circle. All intervals are defined counterclockwise.

4.4.1 Selection by histogram

To select the scanning directions from the best partial intersections of the viewing intervals, we build a histogram. It shows how many pixels can be seen from any direction $\varphi \in [0, 2\pi)$. The histogram $\mathcal{H}(\varphi)$ is constructed by successively adding the functions $\xi_i(\varphi)$ on the interval $[0, 2\pi)$:

$$\mathcal{H}(\varphi) = \sum_{\{i; \mathcal{I}_i \in \mathcal{V}\}} \xi_i(\varphi).$$

With φ_j and φ_k the lower or upper bound of an interval from $\mathcal{V}$, the following observation can be made on the interval $[0, 2\pi)$:

Observation 1: *If $\varphi_j \neq \varphi_k$ then on the circumference of the unit circle there can be found two nonempty intervals between φ_j and φ_k,*

$$(\varphi_j, \varphi_k) \text{ and } (\varphi_k, \varphi_j).$$

Proof: Without loss of generality, $\varphi_j < \varphi_k$. Therefore (φ_j, φ_k) exists. (φ_k, φ_j) exists because $[0, 2\pi)$ wraps around to 0 at 2π. □

Definition 1: Let $\mathcal{C} \subseteq \mathcal{V}$, $\mathcal{C} = \{\mathcal{I}_i, \mathcal{I}_j, \ldots, \mathcal{I}_t\}$ such that

$$\mathcal{I}_i \cap \mathcal{I}_j \cap \ldots \cap \mathcal{I}_t \neq \emptyset.$$

If for all intervals $\mathcal{I}_v \in \mathcal{V} \setminus \mathcal{C}$

$$(\mathcal{I}_i \cap \mathcal{I}_j \cap \ldots \cap \mathcal{I}_t) \cap \mathcal{I}_v = \emptyset,$$

then we say that $\mathcal{C}$ forms a *complete subset* of $\mathcal{V}$.

Let us analyze the intersection of the intervals belonging to the same complete subset.

Observation 2: *The intersection of closed intervals of the complete subset $\mathcal{C} = \{\mathcal{I}_i, \mathcal{I}_j, \ldots, \mathcal{I}_t\}$ of viewing intervals is a closed interval $[\varphi_l, \varphi_u]$, with φ_l the lower bound of that interval from $\mathcal{C}$ whose lower bound lies in all intervals $\mathcal{I}_i, \mathcal{I}_j, \ldots, \mathcal{I}_t$ and φ_u the upper bound of that interval from $\mathcal{C}$ whose upper bound lies in all intervals $\mathcal{I}_i, \mathcal{I}_j, \ldots, \mathcal{I}_t$.*

Proof: The topology for the circumference of the unit circle is the same as for the real line if the union of the intervals does not cover the unit circle

$$\bigcup_i \mathcal{I}_i \neq [0, 2\pi), \tag{4}$$

and in this case the intersection of a finite number of closed intervals is a closed interval.

The viewing intervals satisfy the condition (4): None of the viewing intervals contains the scanning direction of the image acquisition because all the pixels in the range shadows are occluded

from that direction. Since none of the intervals contain the direction of image acquisition the unit circumference can be cut at that point and stretched to a part of the real line.□

Theorem 1: *The histogram $\mathcal{H}(\varphi)$ has a local maximum on the intersection of intervals*

$$[\varphi_l, \varphi_u] = \mathcal{I}_i \cap \mathcal{I}_j \cap \ldots \cap \mathcal{I}_t$$

of the complete subset $\mathcal{C}$.

Proof: The value of the histogram $\mathcal{H}(\varphi)$ in the intersection of viewing intervals $[\varphi_l, \varphi_u]$ equals the number of intervals in $\mathcal{C}$, because each of the functions $\xi_i(\varphi), \xi_j(\varphi), \ldots, \xi_t(\varphi)$ is 1 on that interval while all the other functions $\xi_v(\varphi); \mathcal{I}_v \in \mathcal{V} \setminus \mathcal{C}$ are 0 there. Assume that $\mathcal{I}_j$ is the interval whose lower bound φ_{j_l} equals φ_l and that $\mathcal{I}_k$ is the interval whose upper bound φ_{k_u} equals φ_u. $\xi_j(\varphi)$ is 0 on the interval $(\varphi_{j_u}, \varphi_{j_l}) = (\varphi_{j_u}, \varphi_l)$ while $\xi_k(\varphi)$ is 0 on $(\varphi_{k_u}, \varphi_{k_l}) = (\varphi_u, \varphi_{k_l})$. Using De Morgan's law, the union of intervals $(\varphi_{j_u}, \varphi_l)$ and $(\varphi_u, \varphi_{k_l})$ can be expressed as

$$(\varphi_{j_u}, \varphi_l) \cup (\varphi_u, \varphi_{k_l}) = c[\varphi_l, \varphi_{j_u}] \cup c[\varphi_{k_l}, \varphi_u] = c(\ [\varphi_l, \varphi_{j_u}] \cap [\varphi_{k_l}, \varphi_u]\ ) = c[\varphi_l, \varphi_u] = (\varphi_u, \varphi_l).$$

Therefore, the sum of $\xi_i(\varphi), \xi_j(\varphi), \ldots, \xi_t(\varphi)$ is on the interval (φ_u, φ_l) at least 1 less than on the interval $[\varphi_l, \varphi_u]$. None of the other functions $\xi_v(\varphi)$ can completely compensate for $\xi_j(\varphi)$ and $\xi_k(\varphi)$ on (φ_u, φ_l) in the histogram $\mathcal{H}(\varphi)$, because there can always be found between φ_u and φ_{v_l} a nonempty interval $(\varphi_u, \varphi_{v_l})$ (Observation 1) where $\xi_v(\varphi)$ is 0, and similarly between φ_{v_u} and φ_l. □

Theorem 2: *In each viewing interval $\mathcal{I}_k = [\varphi_{k_l}, \varphi_{k_u}]$ at least one local maximum can be found in the histogram $\mathcal{H}(\varphi)$.*

Proof: Each viewing interval $\mathcal{I}_k \in \mathcal{V}$ belongs to at least one complete subset $\mathcal{C}$:

$$\forall\ \mathcal{I}_k\ \exists\ \mathcal{C}\ :\ \mathcal{I}_k \in \mathcal{C}.$$

According to Theorem 1 the histogram $\mathcal{H}(\varphi)$ has a local maximum on the intersection of intervals

$$[\varphi_l, \varphi_u] = \mathcal{I}_i \cap \mathcal{I}_j \cap \ldots \cap \mathcal{I}_t$$

of the complete subset $\mathcal{C}$. Therefore, the histogram maximum $\mathcal{H}([\varphi_l, \varphi_u])$ exists at the same time in all intervals in $\mathcal{C}$. □

Among intervals of histogram maxima we must find the smallest number of intervals such that by taking one value from each interval at least one value is chosen from each viewing angle. This can be achieved by histogram decomposition.

4.4.2 Histogram decomposition

For each viewing angle we must find the number of maxima it contains. If only one histogram maximum exists for the viewing angle $\Phi(p_k)$, then $\Phi(p_k)$ belongs to only one complete subset, and

can be selected only by selecting the scanning direction from the interval of that maximum. We label all such maxima as *necessary maxima* and remove from the histogram all viewing angles which contain at least one necessary maximum. From the remaining viewing angles we construct a new histogram and repeat the procedure until all viewing angles are removed. If there is no viewing angle which contains only one maximum in the histogram, then to obtain the optimal solution the histogram decomposition must be performed for each local maximum. The solution with the minimum number of necessary maxima is selected. The necessary histogram maxima determine the set of intervals from which we must choose the next scanning directions.

4.4.3 Improving the performance of the algorithm

Let us analyze the time complexity of the algorithm. One cycle of the algorithm has the following steps and corresponding time complexity for n viewing intervals:

1. *Histogram building*: We do not really need to build the histogram $\mathcal{H}(\varphi)$. It is sufficient to arrange/sort the bounds of the viewing intervals on the interval $[0, 2\pi)$. Time complexity of the sorting algorithm is $\mathcal{O}(n \log n)$.

2. *Locating the maxima of the histogram*: To find the maxima of $\mathcal{H}(\varphi)$, one search through the ordered bounds of the viewing intervals is performed. During the search we store the pairs of bounds when the lower bound is followed by the upper bound. The obtained pairs of the interval bounds are the bounds of the histogram maxima. Time complexity is $\mathcal{O}(n)$.

3. *Search for the necessary maxima*: For each viewing angle the number of maxima it contains has to be found. In the worst case, there are n maxima in the histogram. Therefore, the worst case time complexity is $\mathcal{O}(n^2)$.

4. *Search for the intervals that contain the necessary maxima*: In the worst case all the maxima are necessary, i.e., the intervals themselves are histogram maxima. The viewing intervals are tested for each necessary maximum if both the lower and the upper bound of the maximum lie in the interval. Time complexity is $\mathcal{O}(n^2)$.

In the following we estimate the number of cycles. The number of cycles depends on the number of intervals that are selected, removed from the histogram, in each cycle and on the number of maxima when the intervals are arranged in such a manner that none of the maxima in the histogram are necessary. If none of the histogram maxima are necessary, to obtain the optimal solution, the histogram must be decomposed from each local maximum. This situation requires that each of the viewing angles contains at least two maxima. By removing the histogram maximum from the

histogram at least two intervals are removed. The largest possible number of cycles is

$$n \times (n-2) \times (n-4) \times \ldots \times 2 = \Pi_{i=0}^{\frac{n}{2}-1}(n-2i).$$

If n is an even number, then $n = 2m$ and

$$\Pi_{i=0}^{\frac{n}{2}-1}(n-2i) = \Pi_{i=0}^{\frac{n}{2}-1}2(m-i) = 2^{\frac{n}{2}}(\frac{n}{2})! \ .$$

As an example, a set of viewing angles, each with two intervals, requires the above computed number of cycles if none of the intervals has an element in common with any other interval. On the average the algorithm performs much better because the viewing angles of the neighboring pixels usually have directions in common.

If in each histogram during decomposition at least one necessary maximum is found then the number of cycles drops rapidly and is at most $\frac{n}{2}$. To keep the algorithm fast we must define the new criteria for the case when none of the maxima in the histogram are necessary. A good choice is to select the global maximum. In that way the largest set of intervals is selected. If in the histogram none of the maxima are global, i.e., more maxima in the histogram have the same value, then the widest maximum between them is selected.

In general, the viewing angles can have several viewing intervals, each with at least one maximum (Theorem 2). Adding up the viewing angles in the histogram can therefore make the intervals of maxima very short (cf. experimental results) and their number very large. Searching for the solution in the large set of short intervals slows down the computation and gives as a solution the set of short intervals which are often a small subset of the truly possible solution. This drawback can mostly be overcome by first building the histograms for the areas A_i separately. For each area A_i a histogram is built and necessary maxima are found. From the intervals of necessary maxima of these histograms we build the final histogram from which the next scanning directions are then selected.

In this way we get a more reliable solution, but we can lose the optimal solution.

4.5 Completion of the analysis of the first scanning plane

To compute the new scanning directions from the first view we assume that the changes in surface height in the range shadows are too small to cause occlusions. The same is assumed for the regions outside the border of the range shadows. If this is not the case and after combining the images of the computed scanning directions in the ISP there still exist range shadows, then they should be explored from new directions. Since we use only the height information of the borders of range shadows to compute the viewing angles, we may end up with too wide viewing angles. When we compute the new scanning directions it is important to use the information about the active edges from the previous scans in order to incorporate the known constraints on viewing angles.

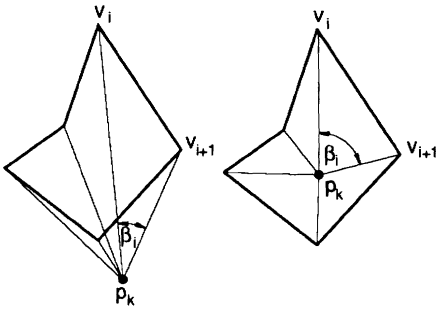


Figure 11: Angle β_i.

For each scanning direction that was taken, range shadows are extracted and approximated by polygons. Their edges are classified as active or inactive. All the polygons P are transformed into the ISP. For each pixel p_k which is still occluded after the views have been taken, the occluded polygons $P(v_1, v_2, \ldots, v_n)$ to which the pixel belongs are determined by the following procedure (Fig. 11): The angles β_i, with $i = 1, 2, \ldots, n$, formed by the lines intersecting in the pixel p_k and going through the neighboring vertices v_{i+1} and v_i are computed

$$\beta_i(v_i, v_{i+1}) = \arctan(\frac{y_i - y_k}{x_i - x_k}) - \arctan(\frac{y_{i+1} - y_k}{x_{i+1} - x_k}),$$

where $(x_i, y_i), (x_{i+1}, y_{i+1})$, and (x_k, y_k) are the coordinates of v_i, v_{i+1}, and p_k in ISP respectively. If $\sum_{i=1}^{n} \beta_i = 2\pi$, then the pixel p_k belongs to the polygon P, otherwise not. From the active edges of each polygon P to which p_k belongs, occluding angles are computed and subtracted from the unit circle. A new viewing angle is obtained. From the viewing angles the next scanning directions are determined by building the histogram. To prevent short maxima, the histograms for pixels belonging to the same set of polygons P are constructed first. From the necessary histogram maxima the final histogram is constructed from which the new scanning directions are selected.

5 Searching for the next scanning planes

In the previous section we described how to rotate the sensor system in the scanning plane to acquire the complete depth data of the illuminated surface $\mathcal{S}$. The points on the surface $\mathcal{Q}$ where the illumination plane grazes the surface $\mathcal{Q}$ (Fig. 12) form borders which separate illuminated from non-illuminated parts of the surface $\mathcal{Q}$. These borders are *occluding borders*. The information about the surfaces between the occluding borders and their projections in the direction of the illumination is not yet acquired in ISP. Additional scanning planes must be computed for that purpose. We will look for such scanning planes only among those perpendicular to the first scanning plane.

Assume that the first scanning plane is the top of a cylinder (Fig. 13). If the new scanning plane is perpendicular to the first one, it can be defined by an angle δ, which is the angle of rotation of the

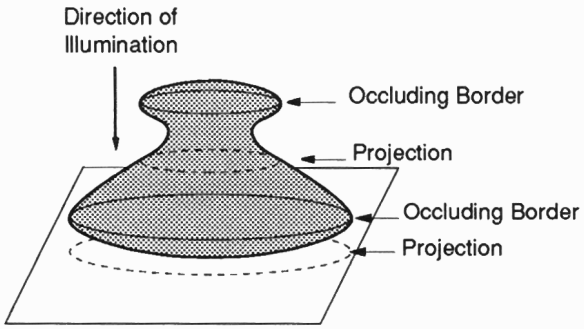


Figure 12: Occluding borders and their projections

illuminating plane about the center at the top of the cylinder. The normal direction of the scanning plane $\vec{n}$ points in the direction of the light rays that illuminate the scene during the scanning.

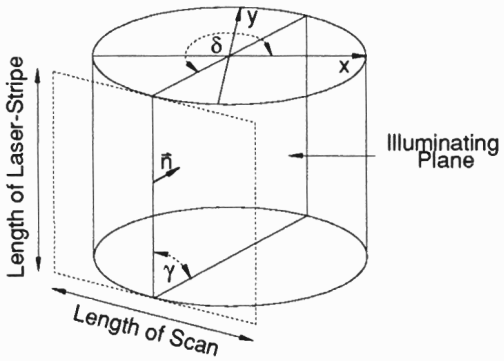


Figure 13: Orientation of the illuminating plane defined by angle δ

5.1 Occluding borders

The only connection between the acquired data of the surface $\mathcal{S}$ illuminated from the first scanning plane and the non-illuminated surfaces are the occluding borders. Occluding borders are defined by the step jumps in height of the surface $\mathcal{S}$. With an edge operator that is sensitive to C^0 discontinuities, we locate the occluding borders in the ISP. They are then approximated by piecewise linear segments $l_i(p_i, p_{i+1})$, $i = 1, \ldots, m$, where p_i, p_{i+1} are the endpoints of the linear segments. In our experiments we use the algorithm proposed by R. Nevatia and K. Ramesh Babu [8]. In their algorithm, line finding consists of the following steps: determining edge magnitude and direction by convolution of the image with a number of edge masks, thinning and thresholding these edge magnitudes, linking the edge elements based on proximity and orientation, and approximating the linked elements by piecewise linear segments.

Each line segment l_i represents, at the same time, a part of an occluding border and its projection. To compute the next scanning planes we will assume that the surface between the occluding

line segment l_i and its projection is a flat face F_i. In a similar way as we did in the previous section for occluding pixels, we define for each line segment l_i the *illuminating angle* $\Psi(F_i)$ as the set of all directions from which the face F_i can be illuminated.

5.1.1 The height of an occluding border and its projection

The height, $h_b(l_i)$, of an occluding border and the height of its projection, $h_p(l_i)$, are found by searching in a narrow zone left and right of the linked edge elements. The direction for the search is defined by the change in x-and y-coordinates of the new edge element in the link. The directions of the search on the left side are the same as those in Fig. 7, and the directions on the right side are just the opposite. For each edge element, we calculate a left and a right height, taking the median of the height values found in the search on the left and the right side, respectively. The left and right height of a line segment l_i are defined as the median of the left and right height values of the edge elements between the endpoints p_i and p_{i+1}. The higher value corresponds to the height of the occluding border, $h_b(l_i)$, and the lower value to the height of its projection, $h_p(l_i)$.

5.1.2 The angle of a line segment

The endpoints p_i, p_{i+1} of the line segment l_i are ordered such that the higher surface is always on the right side, going from p_i to p_{i+1}. The angle of the line segment l_i in the ISP is

$$\psi(l_i) = \arctan(\frac{y_{i+1} - y_i}{x_{i+1} - x_i}).$$

Since the higher surface is on the right side of the line segment l_i, the face F_i can be illuminated only from directions in the range $[\psi(l_i), \psi(l_i) + \pi]_{ccw}$.

5.2 Illuminating angles

To compute the illuminating angle $\Psi(F_i)$ for each face F_i, the areas I ("islands") which are higher than the height of projection $h_p(l_i)$ of the occluding line segment l_i must be located in the range image of ISP. The contours of such islands I are for each l_i found with the same TRACER algorithm used to obtain the contours of the range shadows in the previous section. These islands represent the obstacles for the light. We must find all directions from which the light reaches the face F_i without collisions with the islands. To find a solution to this problem we relied on the work of G. T. Toussaint and J. R. Sack [14] who have solved the problem of moving polygons in the plane: *Given a direction d, can the polygon P be translated an arbitrary distance in direction d without colliding with polygon Q?* The similarity between their problem and ours is that the direction of movability d is in our case the direction of the light, the obstacles Q are the islands I, and the occluding line segment l_i is an edge of the polygon P.

The line on which the line segment l_i lies divides ISP into two parts. If on the left side of the line there is no island I that is higher than $h_p(l_i)$, then the face F_i can be illuminated from any direction in $[\psi(l_i), \psi(l_i)+\pi]_{ccw}$. If there is an obstacle island then each line l_{c_j} on face F_i, connecting a point p_j on the occluding line segment l_i with its projection p_{p_j} (Fig. 14(a)), can be illuminated from a different subset of directions from the range $[\psi(l_i), \psi(l_i)+\pi]_{ccw}$. The illuminating angle of the line l_{c_j} for K obstacle islands is

$$\Psi(l_{c_j}) = [\psi(l_i), \psi(l_i)+\pi]_{ccw} - (\cup_{k=1}^{K} \alpha_j(I_k)).$$

The angle $\alpha_j(I_k)$ includes all directions from which the island I_k occludes the line lc_j. If we insist

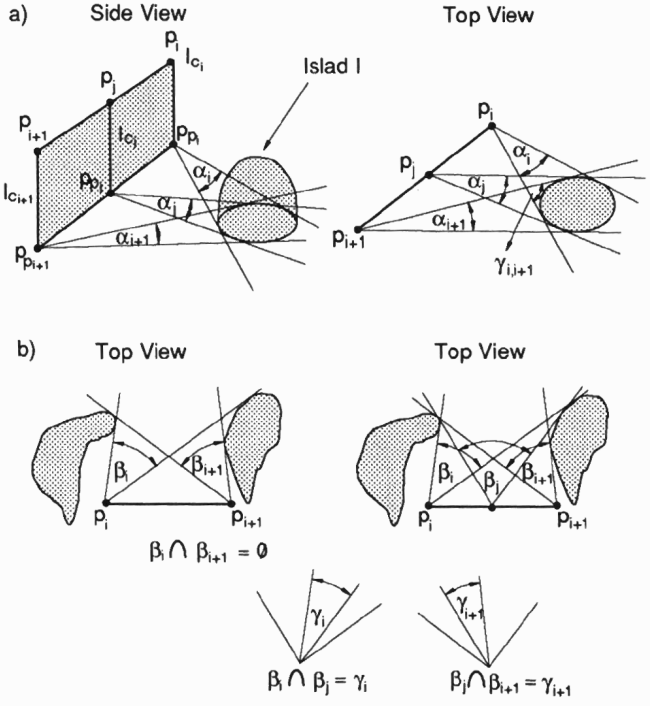


Figure 14: Illuminating angle computation.

on illuminating the whole face F_i at the same time, then the rotation angle δ must be selected from the illuminating angle

$$\Psi(F_i) = [\psi(l_i), \psi(l_i)+\pi]_{ccw} - (\cup_{k=1}^{K} \gamma_{i,i+1}(I_k)).$$

The angle $\gamma_{i,i+1}(I_k)$ is defined as the open interval $(d_i(I_k), d_{i+1}(I_k))$. $d_i(I_k)$ is the last direction from which the light reaches the endpoint p_i counterclockwise before the directions in $\alpha_i(I_k)$, and $d_{i+1}(I_k)$ is the first direction from which the light reaches the endpoint p_{i+1} counterclockwise after the directions in the angle $\alpha_{i+1}(I_k)$:

$$\gamma_{i,i+1}(I_k) = (d_i(I_k), d_{i+1}(I_k)) = \bigcup_j \alpha_j(I_k).$$

If the illuminating angle $\Psi(F_i)$ is empty, then the whole face F_i cannot be illuminated at once. A solution can be found by splitting the line segment l_i and therefore the face F_i into two parts and computing the illuminating angle for each part separately, as illustrated in Fig. 14(b).

5.3 Computation of the next scanning planes

The illuminating angles $\Psi(F_k)$ are as viewing angles finite sets of closed intervals—*illuminating intervals*

$$\Psi(F_k) = \{[\psi_{j_l}, \psi_{j_u}], [\psi_{j+1_l}, \psi_{j+1_u}], \ldots\}.$$

The illuminating intervals $\mathcal{L}_i = [\psi_{i_l}, \psi_{i_u}]$, where $\mathcal{L}_i \in \bigcup_k \Psi(F_k) = \mathcal{W}$, have the following properties:

- $\forall\ \mathcal{L}_i \in \mathcal{W}:\ \psi_{i_u} \leq \psi_{i_l} + \pi$
- Any pair of intervals $\mathcal{L}_i, \mathcal{L}_j \in \Psi(F_k)$ has no element in common:

$$[\psi_{i_l}, \psi_{i_u}] \cap [\psi_{j_l}, \psi_{j_u}] = \emptyset\ \forall\ i \neq j.$$

As for the viewing intervals, the function $\xi_i(\psi)$ is defined for each $\mathcal{L}_i$ on the interval $[0, 2\pi)$ and the histogram $\mathcal{H}(\psi)$ is constructed. The complete subsets $\mathcal{C}$ are defined in the same way as they are for the viewing intervals.

Again, let us analyze the intersection of the illuminating intervals $\mathcal{L}_j \in \mathcal{C}$. Two different cases are possible: The condition (4) is either satisfied for the intervals of the complete subset $\mathcal{C}$ or it is not.

Suppose that ψ_k is the point lying in the intersection of intervals $\mathcal{L}_j \in \mathcal{C}$. According to the properties of the illuminating intervals mentioned above, for each $\mathcal{L}_j$ two intervals $[\psi_{j_l}, \psi_k]$ and $[\psi_k, \psi_{j_u}]$ can be found such that:

$$\psi_{j_l} \geq \psi_k - \pi,\ \psi_{j_u} \leq \psi_k + \pi.$$

If either $\forall\ \mathcal{L}_j \in \mathcal{C}$: $\psi_{j_l} \neq \psi_k - \pi$ or $\forall\ \mathcal{L}_j \in \mathcal{C}$: $\psi_{j_u} \neq \psi_k + \pi$, then the union of intervals in $\mathcal{C}$ does not cover the unit circumference. In that case the Observation 2, Theorem 1 and Theorem 2 are valid also for the illuminating intervals in $\mathcal{C}$.

If in the complete subset $\mathcal{C}$ there exist two such intervals $\mathcal{L}_i, \mathcal{L}_j$ that $\psi_{i_l} = \psi_k - \pi$ and $\psi_{j_u} = \psi_k + \pi$, then these two intervals cover the unit circumference and the condition (4) is not satisfied. Since the lower and the upper bounds of the illuminating intervals are at most π apart, it follows that $\psi_k = \psi_{i_u} = \psi_{j_l}$. It is also evident that $\psi_{i_l} = \psi_{j_u}$.

Observation 3: *If the complete subset $\mathcal{C}$ of illuminating intervals has two intervals $\mathcal{L}_i, \mathcal{L}_j$ such that $\psi_{i_l} = \psi_{j_u}$ and $\psi_{i_u} = \psi_{j_l}$ then the intersection of intervals in $\mathcal{C}$ is:*

1. $[\psi_{i_l}, \psi_{i_l}] \cup [\psi_{i_u}, \psi_{i_u}] = [\psi_{j_l}, \psi_{j_l}] \cup [\psi_{j_u}, \psi_{j_u}]$, *if and only if all the intervals in* $\mathcal{C}$ *are equal either to* $\mathcal{L}_i$ *or to* $\mathcal{L}_j$,

2. *either* $[\psi_{i_l}, \psi_{i_l}] = [\psi_{j_u}, \psi_{j_u}]$ *or* $[\psi_{i_u}, \psi_{i_u}] = [\psi_{j_l}, \psi_{j_l}]$, *if and only if there exists such an interval* $\mathcal{L}_k$ *in* $\mathcal{C}$ *that* $\mathcal{L}_k \neq \mathcal{L}_i$ *and* $\mathcal{L}_k \neq \mathcal{L}_j$.

Proof:

1. If intervals in $\mathcal{C}$ are equal either to $\mathcal{L}_i$ or to $\mathcal{L}_j$ then the intersection of all intervals in $\mathcal{C}$ equals to the intersection $\mathcal{L}_i \cap \mathcal{L}_j$. The complements of intervals $\mathcal{L}_i$ and $\mathcal{L}_j$ can be expressed in two different ways:

$$c[\psi_{i_l}, \psi_{i_u}] = (\psi_{i_u}, \psi_{i_l}) = (\psi_{j_l}, \psi_{j_u}),\ c[\psi_{j_l}, \psi_{j_u}] = (\psi_{j_u}, \psi_{j_l}) = (\psi_{i_l}, \psi_{i_u}).$$

The unit circumference can be covered in the following ways:

$$[0, 2\pi) = [\psi_{i_l}, \psi_{i_l}] \cup (\psi_{i_l}, \psi_{i_u}) \cup [\psi_{i_u}, \psi_{i_u}] \cup (\psi_{i_u}, \psi_{i_l}) = [\psi_{j_l}, \psi_{j_l}] \cup (\psi_{j_l}, \psi_{j_u}) \cup [\psi_{j_u}, \psi_{j_u}] \cup (\psi_{j_u}, \psi_{j_l}).$$

Using De Morgan law the intersection of intervals $\mathcal{L}_i, \mathcal{L}_j$ is:

$$[\psi_{i_l}, \psi_{i_u}] \cap [\psi_{j_l}, \psi_{j_u}] = c(\ c[\psi_{i_l}, \psi_{i_u}] \cup c[\psi_{j_l}, \psi_{j_u}]\ ) = c(\ (\psi_{i_u}, \psi_{i_l}) \cup (\psi_{j_u}, \psi_{j_l})\ ) =$$

$$= c(\ (\psi_{i_u}, \psi_{i_l}) \cup (\psi_{i_l}, \psi_{i_u})\ ) = [\psi_{i_l}, \psi_{i_l}] \cup [\psi_{i_u}, \psi_{i_u}] =$$

$$c(\ (\psi_{j_l}, \psi_{j_u}) \cup (\psi_{j_u}, \psi_{j_l})\ ) = [\psi_{j_l}, \psi_{j_l}] \cup [\psi_{j_u}, \psi_{j_u}].$$

2. Since the interval $\mathcal{L}_k$ is different than $\mathcal{L}_i$ and $\mathcal{L}_j$ at least one of the bounds ψ_{k_l}, ψ_{k_u} must be different from ψ_{i_l}, ψ_{i_u} and different from ψ_{j_l}, ψ_{j_u}. $\psi_{k_u} \geq \psi_{k_l} + \pi$, therefore it can contain only one of the points $\psi_{i_l} = \psi_{j_u}, \psi_{i_u} = \psi_{j_l}$ which are π apart. □

Theorem 3: *If for the illuminating intervals in* $\mathcal{C}$ *the condition (4) is not satisfied, i.e., two intervals* $\mathcal{L}_i = [\psi_{i_l}, \psi_{i_u}], \mathcal{L}_j = [\psi_{j_l}, \psi_{j_u}]$ *can be found in* $\mathcal{C}$ *which cover the unit circumference, then the histogram* $\mathcal{H}(\psi)$ *has a local maximum at* $\psi_{i_l} = \psi_{j_u}$ *and at* $\psi_{i_u} = \psi_{j_l}$.

Proof: Let $\psi_{i_l} = \psi_{j_u} = \psi_1$ and $\psi_{i_u} = \psi_{j_l} = \psi_2$.

1. If $[\psi_1, \psi_1] \cup [\psi_2, \psi_2]$ is the intersection of intervals in $\mathcal{C}$: The value of histogram $\mathcal{H}(\psi_1) = \mathcal{H}(\psi_2)$ equals the number of intervals in $\mathcal{C}$, because each of the functions $\xi_k(\psi); \mathcal{I}_k \in \mathcal{C}$ is 1 at ψ_1 and at ψ_2 while all the other functions $\xi_v(\psi); \mathcal{L}_v \in \mathcal{W} \setminus \mathcal{C}$ are 0 there. $\xi_i(\psi)$ is 0 left of ψ_1 on the interval $(\psi_{i_u}, \psi_{i_l}) = (\psi_2, \psi_1)$, while $\xi_j(\psi)$ is 0 right of ψ_1 on $(\psi_{j_u}, \psi_{j_l}) = (\psi_1, \psi_2)$. Therfore the sum of functions $\xi_k(\psi)$ is at ψ_1 greater than it is left and right of ψ_1. None of the functions $\xi_v(\psi)$ cannot completely compensate for $\xi_i(\psi)$ left of ψ_1 in the histogram $\mathcal{H}(\psi)$ and none for $\xi_j(\psi)$ right of ψ_1, because between ψ_{v_u} and ψ_1 can always be found a nonempty interval (ψ_{v_u}, ψ_1) (Observation 1) where $\xi_v(\psi)$ is 0 and similar between ψ_1 and ψ_{v_l}. An analog observation can be made for ψ_2.

2. If either $[\psi_1, \psi_1]$ or $[\psi_2, \psi_2]$ is an intersection of intervals in $\mathcal{C}$: Without loss of generality let us assume that ψ_1 is the intersection of intervals in $\mathcal{C}$. The proof is for the point ψ_1 analog to the proof for the point ψ_1 above.

   Let us denote the set of all illuminating intervals which contain the point ψ_2 by $\mathcal{B}$. The histogram value equals at ψ_2 the number of intervals in $\mathcal{B}$. The function $\xi_i(\psi)$ is 0 right of ψ_2 on $(\psi_{i_u}, \psi_{i_l}) = (\psi_2, \psi_1)$ while the function $\xi_j(\psi)$ is 0 left of ψ_2 on the interval $(\psi_{j_u}, \psi_{j_l}) = (\psi_1, \psi_2)$. None of the function $\xi_t(\psi) \in \mathcal{W} \setminus \mathcal{B}$ can not completely compensate for $\xi_i(\psi)$ right of ψ_2 and none for $\xi_j(\psi)$ left of ψ_2 in the histogram $\mathcal{H}(\psi)$, because between ψ_2 and ψ_{t_l} can always be found a none empty interval (ψ_2, ψ_{t_l}) where $\xi_t(\psi)$ is 0 and similar between ψ_{t_u} and ψ_2. □

From Theorem 3 it follows that Theorem 2 is valid also for the illuminating intervals of the complete subsets when the condition (4) is not met.

The angles δ of the next scanning planes are determined during the histogram decomposition. The directions in the necessary histogram maxima are the initial candidates for the rotation angles δ of the next scanning planes.

6 Experimental Results

We tested our algorithm on a number of different scenes. Results are shown for two different scenes. The range images were scanned using a laser stripe range finder with 1 mm/pixel spatial resolution and 1.2 mm depth resolution.

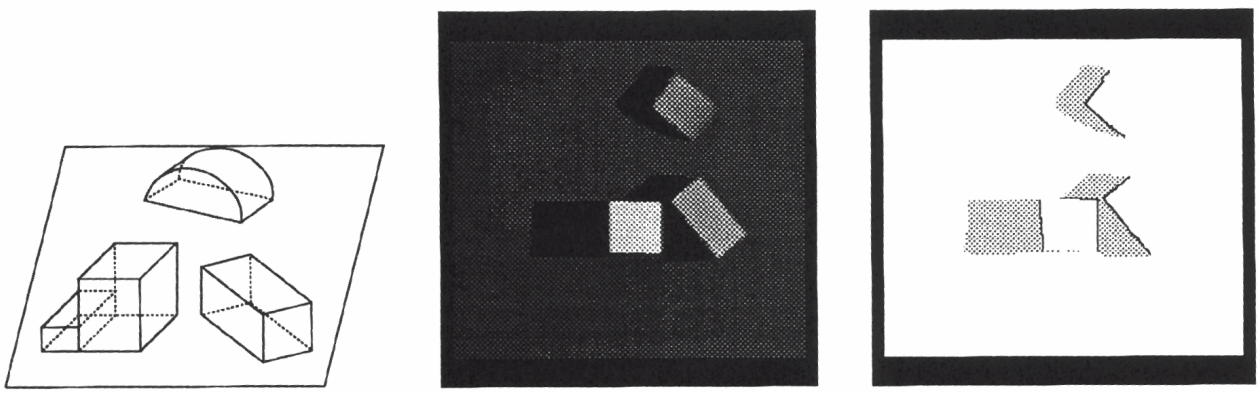
Scene 1: The scene contains three polyhedra of different heights and sizes and one half of a cylinder (Fig. 15(a)). The first image is taken from the scanning direction 0° (Fig. 15(b)). From this direction the tallest polyhedron completely occludes the shortest one and the curved surface of the half cylinder is only partially seen.

The contours of the range shadows are found in the range image and approximated by polygons. By searching around the contours in a belt 7 pixels wide, the border height is obtained for each polygon. The edges of polygons are classified as occluding, active, or inactive. In Fig. 15(c) the active edges (black lines) of the polygonal approximations are presented together with the range shadows (grey areas). The viewing angles for all pixels in the polygons are computed. For each area A_i the histogram of viewing angles is built (Fig. 16). From the necessary maxima of these histograms the final histogram is constructed (Fig. 17(a)). It has two maxima, [148°, 179°] and [271°, 275°], both of which are necessary. From both intervals a new scanning direction must be selected. We picked 165° and 272.5°, but any other pair of directions from these two intervals could have been chosen. Figs. 17(b) and 17(c) show the images of the two selected viewing directions. The pictures of all three views are merged in the ISP (Fig. 18(a)), yielding the complete depth

information of the surface $\mathcal{S}$.

To compute the orientations δ of the next scanning planes, we first locate the edges in ISP (Fig. 18(b)). They represent the occluding borders and are approximated by piecewise linear segments (Fig. 18(c)). Each of the linear segments approximates one part of the occluding border and its projection. The heights of both are defined by searching to the left and right of the edge elements belonging to the linear segment. For each linear segment the illuminating angle is computed by first locating the islands which are higher than a height of its projection. The histogram of illuminating angles is constructed (Fig. 19). During the histogram decomposition three necessary maxima are found: $[52°, 64°], [151°, 159°]$ and $[278°, 300°]$. The directions in these three intervals are the candidates for the rotation angles δ of the next scanning planes. Since our sensor system allows scanning only in one plane, we can only simulate the illumination of the scene from the side. If the selected directions are 58°, 155°, and 289°, the light would reach the surface as shown by the white stripes in Figs. 20(a), 20(b), and 20(c).

Scene 2: The scene contains a polyhedron, one half of a cylinder, a telephone receiver, and a wedge. The first image is taken from the scanning direction 0° (Fig. 21(a)). Range shadows are approximated by polygons, the height of edges is obtained and the edges are classified. In Fig. 21(b) the active edges (black lines) are presented together with the range shadows (gray areas). The constructed histogram (Fig. 22(a)) has four maxima. During the histogram decomposition, two maxima among them, $[179°, 187°]$ and $[226°, 230°]$, are selected. Additional images are taken from the directions 183° and 228° (Fig. 22(b) and Fig. 22(c), respectively). The images of all three views are merged in the ISP (Fig. 23(a)).



(a) (b) (c)

Figure 15: **Scene 1:** (a) Sketch of the scene, (b) Range image of the first view taken from the scanning direction 0°, (c) Range shadows (grey areas) and active edges (black lines) of their polygonal approximation.

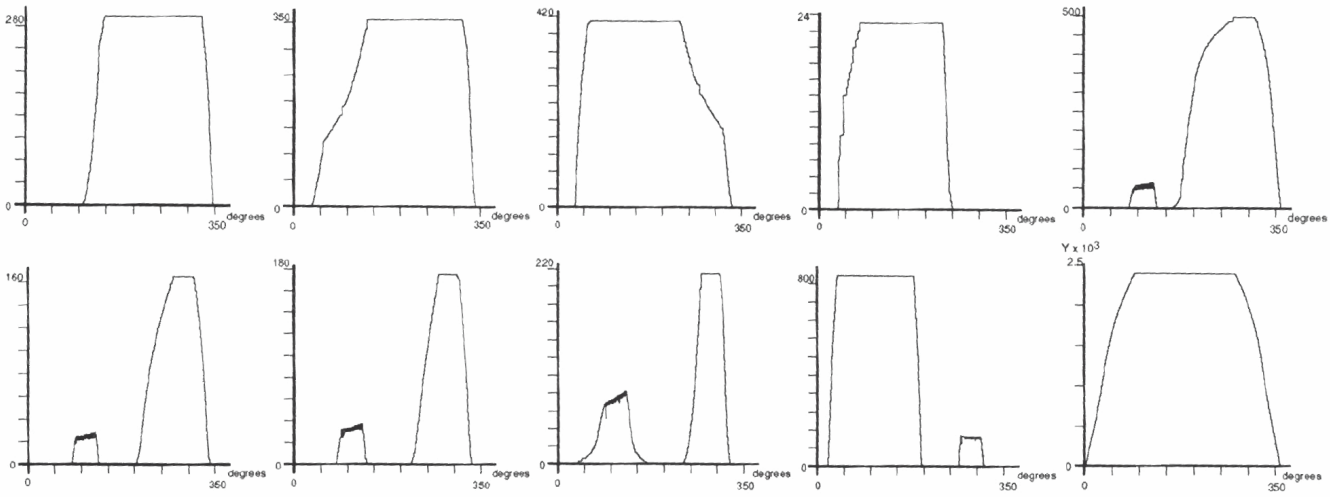
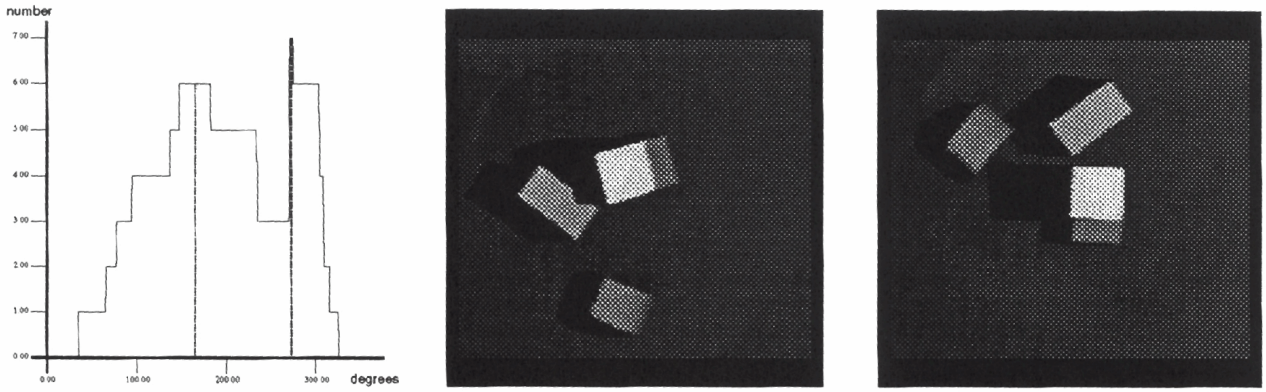


Figure 16: Histograms of areas A_i.

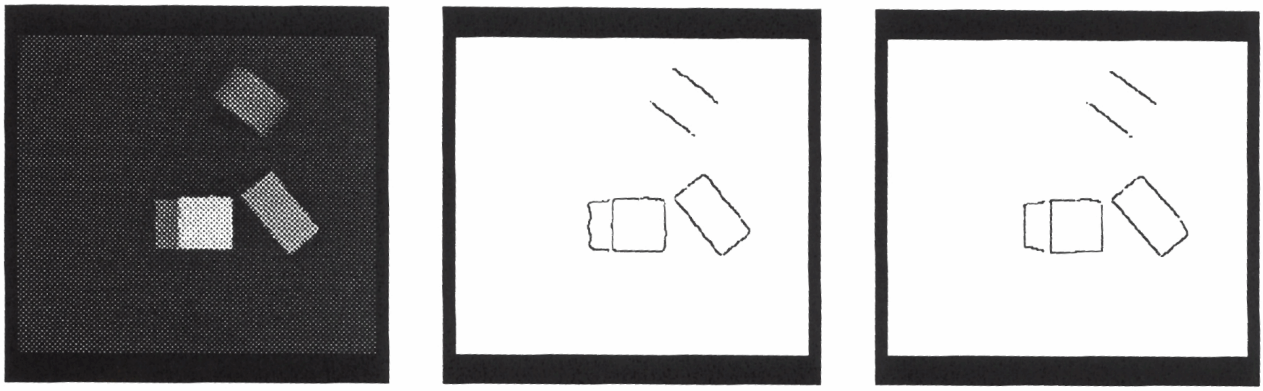


(a) (b) (c)

Figure 17: (a) Final histogram, (b) Range image taken from the scanning direction 165°, (c) Range image taken from the scanning direction 272.5°

At the place where the polyhedron touches the half cylinder, small occluded parts still exist because the image of the first view gives no information about the scene shape at that location. By locating the active edges in the images of the scanning directions 183° and 228° (Figs. 23(b) and 23(c), respectively), information about the shape at the unresolved location is obtained. New viewing angles are computed for the occluded pixels in the ISP, and the histogram is constructed (Fig. 24(a)). Two additional views are taken from the direction 272° (Fig. 24(b)) and 45° (Fig. 24(c)). All five views together yield the complete depth information of the surface $\mathcal{S}$ (Fig. 25(a)).

For the lateral scanning planes, the edges are located in the ISP (Fig. 25(b)) and approximated by linear segments (Fig. 25(c)). The histogram of illuminating angles is built. Some parts of the scene, like the inner side of the telephone receiver, the half cylinder, and the polyhedron, require a lot of directions



(a) (b) (c)

Figure 18: (a) Complete image of surface $\mathcal{S}$, (b) Edges, (c) Linear approximation of edges.

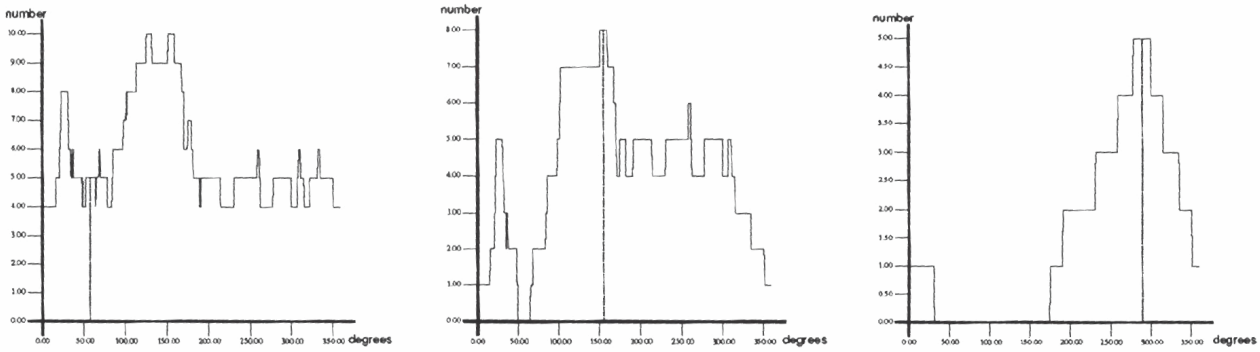
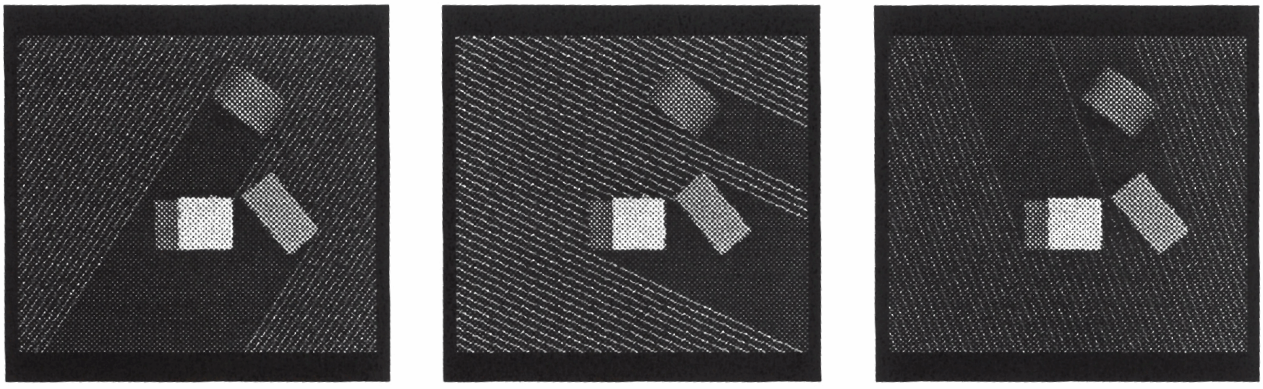
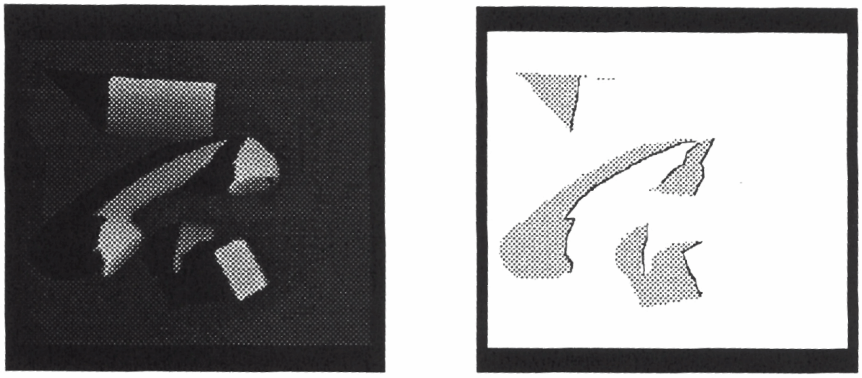


Figure 19: Histograms of illuminating angles.



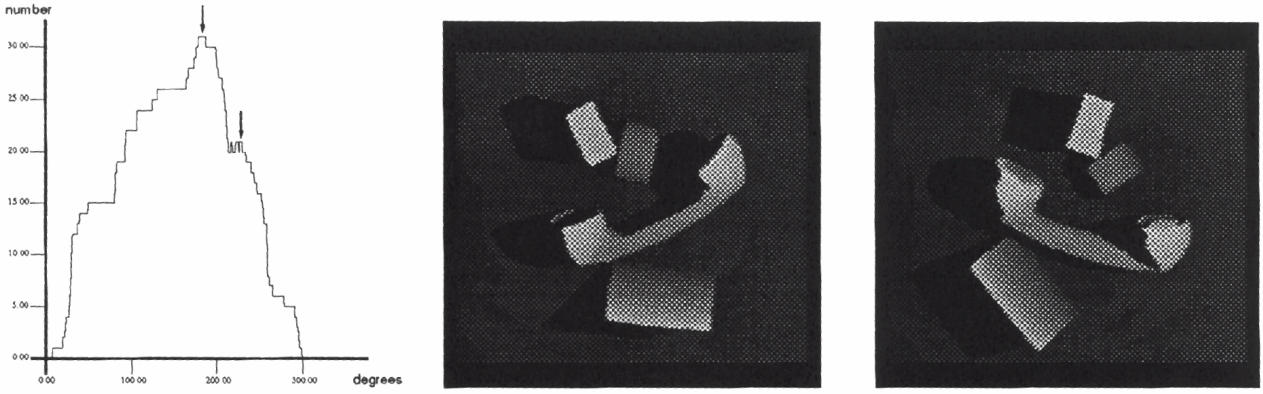
(a) (b) (c)

Figure 20: Light simulation from directions: (a) 58°, (b) 155°, (c) 289°.



(a) (b)

Figure 21: **Scene 2:** (a) Range image of the first view taken from the scanning direction 0°, (b) Range shadows (grey areas) and active edges (black lines) of their polygonal approximation.



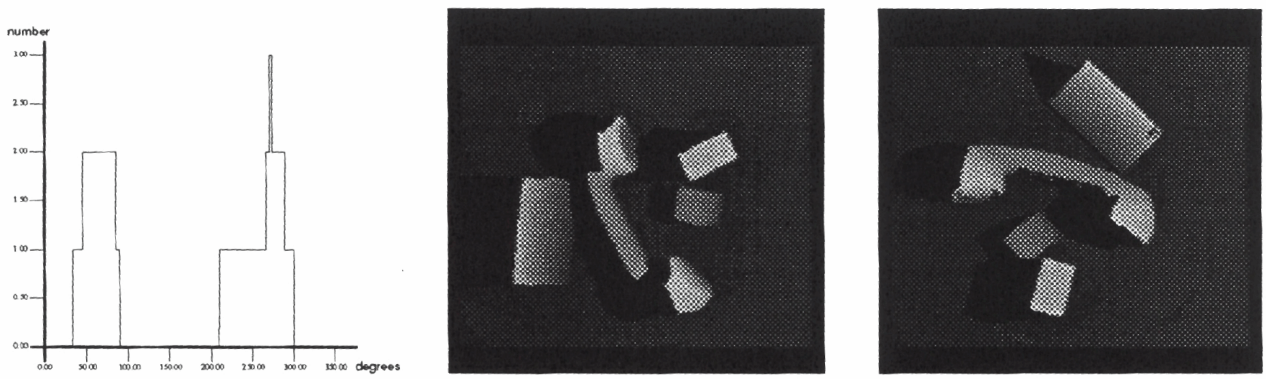
(a) (b) (c)

Figure 22: (a) Final histogram of viewing angles, (b) Range image of the second view taken from the scanning direction 183°, (c) Range image of the third view taken from the scanning direction 228°.

for complete illumination due of their shape and their arrangement in the scene. Therefore, during the histogram decomposition seven histogram maxima emerge for selection (Fig. 26): [1°, 2°], [37°, 45°], [183°, 190°], [240°, 240°], [6°, 12°], [282°, 288°], [304°, 308°].

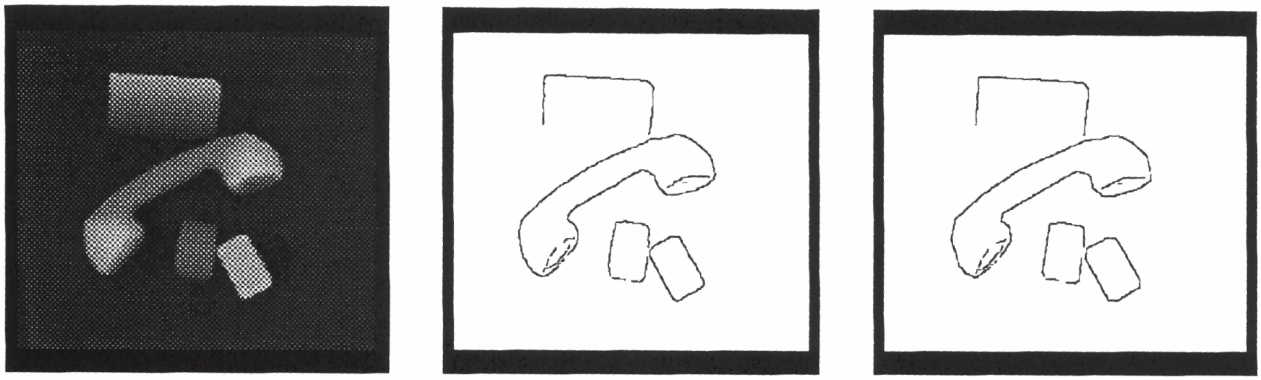
(a) (b) (c)

Figure 23: (a) Range images of the first, the second, and the third view in ISP, (b) Range shadows (gray areas) and active edges (black lines) of the second view, (c) Range shadows (gray areas) and active edges (black lines) of the third view.



(a) (b) (c)

Figure 24: (a) Final histogram, (b) Range image of the fourth view taken from the scanning direction 272°, (c) Range image of the fifth view taken from the scanning direction 45°.



(a) (b) (c)

Figure 25: (a) Complete image of surface $\mathcal{S}$, (b) Edges, (c) Linear approximation of edges.

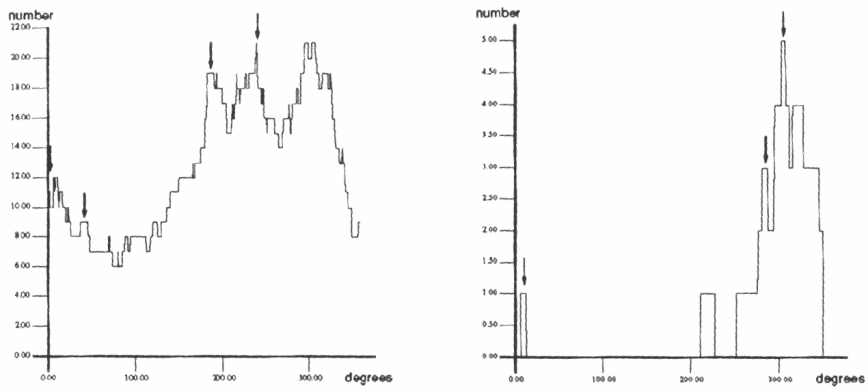


Figure 26: Histograms of illuminating angles.

7 Conclusions

The camera-laser configuration of the light stripe range finder allows only the distance of those portions of a 3-D scene which are simultaneously illuminated by the light and visible to the camera to be measured. Two types of occlusions can appear in a range image: parts which are illuminated during the scanning but not visible to the camera determine the first type of occlusions, while parts which are not illuminated during the scanning are missing data which will show up in the 3-D spatial map of the scene. We have developed an algorithm for selecting the proper views to produce the complete image of the scene illuminated from the first scanning plane. From the data acquired in the first scanning plane we can then compute new scanning planes for further 3-D data acquisition. The new scanning planes are selected only among those perpendicular to the first one. Because of this limitation, parts of the scene which are occluded to the camera from all directions in the first scanning plane, like holes, cannot be explored. The angle between the direction of the

light of the first scanning plane and the direction of the light of the new scanning planes γ, see Fig. 13, can also be smaller than $\frac{\pi}{2}$. For these scanning planes the computed set of angles δ is not necessarily the optimal solution. To obtain the optimal solution new obstacle islands I must be located in ISP, taking into account the slope of the laser light. Even if we allow arbitrary scanning planes, the exploration of such regions is sometimes limited, because the camera-laser configuration is fixed and sensor geometry can not reach deeply hidden structures. Perhaps in this case it is easier to use different kind of sensors, for example tactile sensors, which are better suited to examine the shape of holes and closed regions.

The disadvantage of our method is that even small errors in computed viewing or illuminating intervals can increase the number of viewing directions or scanning planes, respectively. For example, if two viewing or two illuminating intervals are both too long for $\triangle\varphi$ or $\triangle\psi$, respectively, and these extended parts form an intersection, then the new viewing direction or the new scanning plane is badly selected. On the other hand, if the computed intervals are too short for $\triangle\varphi$ or $\triangle\psi$ then the common directions can be lost. The algorithm can fail if the scene is closed or if the range shadows in the acquired range images are split and tiny.

To make assumptions on the surface in the occluded parts, only the scene at the borders of occlusions is modeled. The basic question is: can we build a model of the complete scene during the data acquisition process which can help us to make better decisions for the next data gathering? The major problem is finding an expressive model for shape description which can be easily and quickly modified. Any geometrical knowledge about the scene can help. If we know, for example, that the scene can be modeled by some kind of volumetric primitives, the models computed from the partial data can give us better clues about the occluded scene. In some situations, at least, a coarse model constructed from the current data is desirable. Imagine an exploration system where different sensors can gather the data. If all geometric data can not be acquired from the outside, the sensor must enter the scene to collect the data. Such situations require path planning to avoid collisions of the sensor with the scene.

Acknowledgments

The authors wish to thank Ulf Cahn von Seelen, Aleš Jaklič, Franc Solina, and Aleš Leonardis for comments on a draft of this paper.

This research was supported in part by AFOSR Grants 88-0244, AFOSR 88-0296; ARO/DAAL 03-89-C-0031PRI; NSF Grants CISE/CDA 88-22719, IRI 89-06770; DARPA Grant N0014-88-K-0630, the Dupont Corporation, and the Ministry of Science and Technology of the Republic of Slovenia.

References

[1] N.Ahuja and J.Veenstra, "Generating Octrees from Object Silhouettes in Orthographic Views," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 11, pp. 137–149, Feb. 1989.

[2] J.Aloimonos "Purposive and Qualitative Active Vision," *10 th International Conference on Pattern Recognition*, pp. 346–360, June 1990.

[3] C.I.Connolly, "The Determination of Next Best Views," in *Proc. IEEE Int. Conf. on Robotics and Automation,* pp. 432–435, 1985.

[4] C.K.Cowan and A.Bergman, "Determining the camera and light source location for a visual task," In *Proc. IEEE Int. Conf. on Robotics and Automation,* pp. 509–514, 1989.

[5] S.Fisk, "Short proof of Chvatal's Watchman Theorem," *J. Combinatorial Theory*, ser. b, vol. 24, p. 374, 1978.

[6] S.A.Hutchinson and A.C.Kak, "Planning Sensing Strategies in a Robot Work Cell with Multi-Sensor Capabilities," *IEEE Transactions on Robotics and Automation* ,vol.5, no.6, pp. 765–783, December 1989.

[7] J.Kahn, M.Klawe, and D.Kleitman, "Traditional galleries require fewer watchmen," IBM Res. Rep. RJ3021, Dec. 1980.

[8] R.Nevatia and K.R.Babu, "Linear Feature Extraction and Description," *Computer Graphics and Image Processing,* vol.13, pp. 257–269, 1980.

[9] T.Pavlidis, *Algorithms for Graphics and image processing*, Computer Science Press, 1982

[10] S.Sakane, M.Ishii, and M.Kakikura, "Occlusion avoidance of visual sensors based on a hand-eye action simulator system: HAVEN," *Advanced Robotics*, vol.2, no.2, pp. 149–165, 1987.

[11] S.Sakane, T.Sato, and M.Kakikura, "Model-based planning of visual sensors using a hand-eye action simulator system: HAVEN," *Proceedings of the 3rd International Conference on Advanced Robotics ICAR'87*, pp. 163–174, October 1987.

[12] S.Y.Shin and T.C.Woo, "An Optimal Algorithm for Finding all Visible Edges in a Simple Polygon," *IEEE Transactions on Robotics and Automation*, vol. 5, no. 2, pp. 202–207, April 1989.

[13] K.Tarabanis and R.Y.Tsai, "Viewpoint Planning: The Visibility Constraint,"*Image Understanding Workshop*, pp. 893–903, May 1989.

[14] G.T.Toussaint and J.R.Sack, "Some New Results on Moving Polygons in the Plane," in *Proc. Robotics Intelligence and Productivity Conference*, Detroit, Michigan, November 1983.

[15] C.J.Tsikos, *Laser Range Imaging System User's Guide*, 1989.

[16] P.Whaite and F.P.Ferrie, "From Uncertainty to Visual Exploration," *ICCV Conference*, pp. 690–697, 1990.