University of Pennsylvania
ScholarlyCommons

Technical Reports (CIS)

Department of Computer & Information Science

April 1991

Occlusions as a Guide for Planning the Next View

Jasna Maver
University of Pennsylvania

Ruzena Bajcsy
University of Pennsylvania

Follow this and additional works at: https://repository.upenn.edu/cis_reports

Recommended Citation

Jasna Maver and Ruzena Bajcsy, "Occlusions as a Guide for Planning the Next View", . April 1991.

University of Pennsylvania Department of Computer and Information Science Technical Report No. MS-CIS-91-27.

This paper is posted at ScholarlyCommons. https://repository.upenn.edu/cis_reports/375
For more information, please contact repository@pobox.upenn.edu.

Occlusions as a Guide for Planning the Next View

Abstract

The task of constructing a volumetric description of a scene from a single image is an underdetermined problem, whether it is a range or an intensity image. To resolve the ambiguities that are caused by occlusions in images, we need to take sensor measurements from several different views. We have limited ourselves to range images obtained by a laser scanning system. It is an active system which can encounter two types of occlusions. 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 3-D data acquisition is divided into two subproblems: to acquire the depth information from one scanning plane and to select the proper scanning planes from which the direct laser light illuminates the entire scene. The first kind of occlusions (range shadows) are easily detected and can be used in designing an efficient algorithm. We develop a strategy to determine the sequence of different views using the information in a narrow zone around the occluded regions. Occluded regions are approximated by polygons. Based on the height information of the border of the occluded regions and geometry of the edges of the polygonal approximation, the next views in the same scanning plane the directions of the next scanning planes for further data acquisition are computed.

Comments

University of Pennsylvania Department of Computer and Information Science Technical Report No. MS-CIS-91-27.

**Occlusions As A Guide For
Planning The Next View**

**MS-CIS-91-27
GRASP LAB 257**

**Jasna Maver
Ruzena Bajcsy**

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

April 1991

Acknowledgements:

**This research was supported in part by : AFOSR
Grants 88 0244, AFOSR 88-0296; Army/DAAL
03-89-C-0031PRI; NSF Grants CISE/CDA 88-22719,
IRI 89-06770; ARPA Grant N0014-88-K-0630 and
Dupont Corporation.**

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

The task of constructing a volumetric description of a scene from a single image is an underdetermined problem, whether it is a range or an intensity image. To resolve the ambiguities that are caused by occlusions in images, we need to take sensor measurements from several different views. We have limited ourselves to range images obtained by a laser scanning system. It is an active system which can encounter two types of occlusions. 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 3-D data acquisition is divided into two subproblems: to acquire the depth information from one scanning plane and to select the proper scanning planes from which the direct laser light illuminates the entire scene. The first kind of occlusions (range shadows) are easily detected and can be used in designing an efficient algorithm. We develop a strategy to determine the sequence of different views using the information in a narrow zone around the occluded regions. Occluded regions are approximated by polygons. Based on the height information of the border of the occluded regions and geometry of the edges of the polygonal approximation, the next views in the same scanning plane are determined. From the acquired information in the first scanning plane the directions of the next scanning planes for further data acquisition are computed.

1 Introduction

The task of constructing a 3-D description of a scene from a single image is an underdetermined problem, whether it is a range or an intensity image. To resolve the ambiguities that are caused by occlusions in images, we need to take sensor measurements from several different views. The intelligent sensor system should be able to select the proper views by itself and to build the spatial description of the scene. The 3-D scene can be described in several different ways: voxels, octrees, general cylinders, and superquadrics are the techniques most commonly employed.

The task addressed in this paper deals primarily with the strategy of acquiring 3-D information. How this information can be exploited in building a variety of 3-D models is not a current issue. The problem of data acquisition can be classified according to different criteria like, for example, sensor detectability [6], type of environment (indoor or outdoor), and task. Depending on how much a priori geometrical information about the scene is available, the following classification can be done:

1. If we know the complete scene geometry, which are the “best” viewing directions or locations for data acquisition?
2. If we have only partial spatial information of the previous views but have a predefined finite set of all possible models that can occur in the scene, which are the best viewing directions or the best locations for further data acquisition?
3. If we have only partial spatial information of the previous views and no predefined models, which are the best viewing directions or the best locations for further data acquisition?

The first kind of problem is a familiar question in computational geometry. An example in two dimensions is the well-known “Art Gallery Problem” [4, 7, 12]: suppose that 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. Since we already have the complete spatial information of the scene, the problem is not really data acquisition but computation of the “best” locations and directions for the sensors in the scene where the shape, the location, and the orientation of the target object and the obstacle object are known in advance. The term “best” can mean the minimal number of observation locations, as in the “Art Gallery Problem”. It can also mean to compute the observation locations and directions for sensor placement from which some region or feature can be observed best. Cowan et al. [3] 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 where to place the sensors such that the working place can be observed best 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 attentions 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 unique object identification. In order to recognize the object another sensing operation is 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 — computation of a predefined optimization criterion or strategy which yields the next sensing operation.

An instance of such a problem is addressed in [5]. After taking the measurement, the 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 its 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 equivalence 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. The identification becomes difficult not only because the sets of features change but also because it is not clear which features belong to which object.

The third form of the data acquisition problem arises when a sensor system meets an unknown environment which it has to investigate with its sensors: it is the task of autonomous exploration. In the literature [2, 1] several different strategies for controlling the sensing operations are proposed. The set of sensor actions can be predefined, like 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. Always thirteen views must 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 [2] 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 on it 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 necessary 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, 2] the algorithms are tested only on simulated data.

Another way is to interpret the current data with volumetric models, for example superellipsoids. This approach is described in the work of P. Whaite and F. P. Ferrie [15]. After the surface segmentation the task is to find superquadric parameters that describe each segmented part best. The solutions obtained by using the Levenburg-Marquardt method show an inherent lack of uniqueness. The next viewing directions are computed to disambiguate the solutions of the current interpretation.

In our work we separate data acquisition from model building. The computation of viewing directions is based on the geometry of the sensors and on the extracted information of the scene geometry near the occlusions. The range images are obtained by a laser scanning system. It is an active binocular system which can encounter two types of occlusions. 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 3-D data acquisition is done in two steps. After taking the depth image of a scene we first extract the regions of missing data due to the first kind of occlusions. These data are acquired by rotating the sensor system in the scanning plane which is defined by the first scan. After a complete $2\frac{1}{2}$ -D image of the scene from that scanning plane has been built, the missing data which are due to the second kind of occlusions are located. Then the directions of the next scanning planes from which the unknown data can be acquired are computed.

The remainder of the paper is organized as follows. In section 2 we describe our sensor system and in section 3 we outline two-step sensing strategy. The problem definition of acquiring the data from the first scanning plane and a proposal to its solution is 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 the active range scanner system [14], consisting of a laser, a CCD camera, and a turntable which supports the scanned scene. The laser and the camera are coupled. The laser produces a beam which is spread into an illuminating plane. The illuminating plane intersects with the object surface, forming a planar curve (laser-stripe). Each point on the

Figure 1: Sensing.

curve is mapped onto a single point in the camera image plane. The distance of the point on the curve to the camera center is determined by the intersection of the illuminating plane and the line which goes through the camera center and the point on the image plane. The range image is obtained by scanning the scene in a series of parallel illuminating planes. This is achieved by moving the scene support with constant velocity perpendicular to the illuminating plane. The y -axis on the image corresponds to the laser-stripe and the x -axis corresponds to the shift of the support. During the scanning, the laser sweeps out a plane, called the *scanning plane* (see fig. 1). The distance measurements are transformed into the perpendicular distance between the scanning plane and the surface of the objects and are stored as intensity values in the final range image. Higher values correspond to smaller distances, lower to greater distances.

3 Two step strategy of data acquisition process

The described sensor system can encounter two types of occlusions. An occlusion arises either when the reflected laser light does not reach the camera or when the direct laser light does not reach the surface. The task of 3-D data acquisition is done in two steps.

First step: The system is calibrated so that the depth of the support plane has the lowest value, which is greater than zero. This enables us to detect the first kind of *occluded region*—range shadows whose intensity value is zero. Because of the occluded regions the range image contains incomplete $2\frac{1}{2}$ -D data of the scene. We can get a complete $2\frac{1}{2}$ -D data of the scene by scanning from different directions in the same scanning plane. In each scan (view) the illuminating plane illuminates the same part of the scene surface, but the camera sees its different parts. For every coordinate pair (x, y) in the scanning plane, we get the distance of the closest point on the surface to the scanning plane (fig. 2),

$$h(x, y) = \max(h_{on\ surface}(x, y)). \quad (1)$$

Figure 2: Height at the point (x, y) is defined by the distance of the closest point on the surface to the scanning plane.

Second step: From the complete $2\frac{1}{2}$ -D data of the first scanning plane we compute the orientations of the next scanning planes for possible further 3-D data acquisition.

4 Getting the complete $2\frac{1}{2}$ -D data of the scene

Coordinate planes: Two coordinate planes are defined.

- **Image Plane (IP)** refers to the center of the range image. The x -axis is aligned with the rows and the y -axis with the columns of the image. The range image is obtained by moving the laser-stripe and the camera from left to right, with the camera to the right of the laser-stripe.
- **Image of the Scanning Plane (ISP)** is the coordinate plane in which the images are combined and refers to the center of the scanning plane. The transformation between the IP and ISP is determined by the scanning direction.

Scanning direction φ is the angle of counterclockwise rotation of the camera-laser configuration about the origin of the scanning plane. However, in our experiments we rotated the support in a

clockwise direction instead.

The point $(x, y)_{IP}$ in the IP is transformed into the point $(x, y)_{ISP}$ in the ISP by the following transformation:

$$\begin{bmatrix} x \\ y \end{bmatrix}_{ISP} = \begin{bmatrix} \cos(\varphi) & -\sin(\varphi) \\ \sin(\varphi) & \cos(\varphi) \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}_{IP} \quad (2)$$

4.1 Problem definition

In all occluded regions the information about the height of the surface is unknown. The question is how to determine the next scanning directions to acquire the missing information.

Figure 3: The distance l_{ik} between points $p_i(x, y)$ and $p_k(x, y)$ and the length of occlusion l'_{ik}

Problem 1: For a given point $p_i(x, y)$ in the scanning plane, we want to find all the directions from which $p_i(x, y)$ is visible, assuming orthographic projection and a scene where all the points on the surface are illuminated during the scanning.

In order to determine whether a given point $p_i(x, y)$ is visible from the direction φ_j , we must check if any of the points in the scene along this direction can occlude the point $p_i(x, y)$ (fig. 3). We consider the relation between the distance l_{ik} of two points in the scanning plane and the length of the occlusion l'_{ik} which would be produced by the step change in height $h(p_k) - h(p_i)$ at $p_k(x, y)$ in

4.1 Problem definition

direction φ_j and is defined as:

$$l'_{ik} = \frac{h(p_k) - h(p_i)}{\tan(\alpha(h(p_k), l_{ik}))}, \quad (3)$$

where $h(p_i)$ and $h(p_k)$ represent the height of the scene at points $p_i(x, y)$ and $p_k(x, y)$, respectively.

The angle α changes by the following equation (fig. 4):

$$\alpha(h(p_k), l_{ik}) = \arctan\left(\frac{H - h(p_k)}{L - l_{ik}}\right), \quad (4)$$

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

Figure 4: The angle α changes with the height (h_1, h_2, h_3) of an obstacle and with the distance l_{ik} between the light and the obstacle.

If for all k the distance l_{ik} between $p_i(x, y)$ and $p_k(x, y)$ is greater than the length of occlusion

$$l_{ik} > l'_{ik} \text{ for all } k, \quad (5)$$

then $p_i(x, y)$ is visible from direction φ_j .

In the same scanning plane the point $p_i(x, y)$ can be seen from several different directions $\varphi_j, \varphi_{j+1}, \dots$. To compute the set of all directions from which $p_i(x, y)$ is visible, we have to know the height at each point in the scanning plane. As indicated previously, after the first scan the height at the points that belong to the occluded regions is unknown. For the height at the point $p_i(x, y)$ which was occluded in the first scan by the obstacle at the point $p_o(x, y)$ (fig. 5), we can derive the inequality

$$h(p_i) < h(p_o) - \tan(\alpha(h(p_o), l_{io})) \cdot l_{io}, \quad (6)$$

Figure 5: The sketch of the inequality (6).

where l_{io} represents the distance between points $p_i(x, y)$ and $p_o(x, y)$. The equation (3) gives for the predicted length of occlusion l'_{ik} the inequality

$$l'_{ik} > \frac{h(p_k) - h(p_o) + \tan(\alpha(h(p_o), l_{io})) \cdot l_{io}}{\tan(\alpha(h(p_k), l_{ik}))} \quad (7)$$

From (7) it follows that condition (5) cannot be tested unless we make some assumptions about the height at the point $p_i(x, y)$. The question is: *How restrictive an assumption about the height at the point in the occluded region can be made?* If the assumption is too restrictive then a solution may not be found at all, while if the assumption is too weak it may result in a direction from which the point is not visible.

Problem 2: Having determined for each point in the occluded regions the set of all directions from which the point is visible, which are the best directions for further data acquisition?

4.2 Representation of occluded regions

The occluded regions do not have any depth information but they do contain same information in their shape. We approximate the occluded regions by polygons. The visibility directions for pixels in occluded regions are computed from the analysis of these polygons.

The contours¹ of the occluded areas are represented as contour descriptors $x(s)$ and $y(s)$ for each area separately. Each contour must be segmented into a series of straight lines. The breakpoints on the contour are points of maximum curvature.

4.2.1 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

¹The contours of the occluded regions in the image are found by Pavlidis' algorithm TRACER [9].

tangent angle function as

$$\phi(s) = \arctan\left(\frac{dy/ds}{dx/ds}\right). \quad (8)$$

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

$$\kappa(s) = \frac{d\phi(s)}{ds}. \quad (9)$$

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

4.2.2 Border height

The height of the contour $h(s)$ is found by searching in a narrow zone around the occluded area. For each pixel on the contour the search is done for k pixels in three directions. The directions of the search are determined by the change in the x - and y -coordinates of the new pixel on the contour (fig. 6). The highest value found in the search is chosen as the pixel height.

The height of the border between two vertices is approximated by a constant. It is defined as the median of all height values on the contour between the two vertices.

Figure 6: Searching directions for the pixel height.

4.2.3 Polygon representation

The points of maximum curvature are the vertices $v_i(x, y)$ of a polygon $P(v_1, v_2, \dots, v_n)$ which approximates the occluded region. The vertices are ordered clockwise so that, when moving from vertex to vertex in direction of order, the occluded region is on the right side of the edge. For each edge $e_i(v_i, v_{i+1})$ in polygon P its angle (fig. 7) with respect to the IP is defined by the next equation:

$$\theta_{IP}(e_i) = \arctan\left(\frac{y_i - y_{i+1}}{x_i - x_{i+1}}\right), \quad (10)$$

where (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 IP. The angle of the same edge $e_i(v_i, v_{i+1})$ in the ISP can be computed by adding the angle of the scanning direction φ to $\theta_{IP}(e_i)$

$$\theta_{ISP}(e_i) = \theta_{IP}(e_i) + \varphi. \quad (11)$$

Figure 7: Angle θ of the edge e_i with respect to the IP and ISP.

4.3 Analysis of occluded regions

Two properties are added to each edge, according to its angle $\theta(e_i)_{IP}$ in the image plane and its height $h(e_i)$. The outline of the occluded region (range shadow) depends on

- the position of the light and the observer (if the direction of the light coincides with the direction of the line of sight, then there is no range shadows),
- the shape of the obstacle scene (also that part that is not illuminated during the scanning),
- the shape of the scene on which the range shadow is cast.

In our case the configuration of the light and the observer is fixed and known. The orthogonal projection of the line of sight into the scanning plane is perpendicular to the direction of the light and parallel to the direction of the scanning motion.

For each polygon P , presenting the occluded region, we would like to determine which part of the border has cast the range shadow and which part can occlude the current occluded region looking at the scene from any other direction. Assuming that all the points on the surface in the scene are illuminated during the scanning, the following properties of an edge can be defined:

1. **Occlusion:** An edge $e_i(v_i, v_{i+1})$ with angle $\theta(e_i)_{IP} \in (0^\circ, 180^\circ)_{ccw}$ is called an *occluding edge* e_i^{occ} and belongs to the set of occluding edges, $\mathcal{O}$. Occluding edges have caused the occluded region.

If in polygon P we draw lines through the end points of the occluding edges parallel to the x -axis (the direction of motion during the scanning), we cut the polygon P into areas A_i , as shown in figure 8. To each edge e_i^{occ} in the polygon P belongs such an area A_i . Polygon P can then be defined in the following way:

$$P(v_1, v_2, \dots, v_n) = \bigcup_{\{i; e_i^{occ} \in \mathcal{O}\}} A_i, \quad (12)$$

Note that only occluding edges create areas A_i .

Figure 8: Areas A belonging to occluding edges.

2. **Activity:** This property is derived from the height of the border. 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 . We compare the length of occlusion l_{ji} on the range image caused by the occluding edge e_i^{occ} with the length of occlusion caused by the height difference $(h(e_i^{occ}) - h(e_j))$ for $\alpha(h(e_i^{occ}), l_{ji})$. For each area A_i , the edges which limit the area are tested as follows:

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

$$e_j = \begin{cases} \text{inactive} & \text{if } l_{ji} \leq \frac{h(e_i^{occ}) - h(e_j)}{\tan(\alpha(h(e_i^{occ}), l_{ji}))} \\ \text{active} & \text{if } l_{ji} > \frac{h(e_i^{occ}) - h(e_j)}{\tan(\alpha(h(e_i^{occ}), l_{ji}))}, \end{cases} \quad (13)$$

where $j = 1, \dots, n$ and l_{ji} is measured at the midpoint of the edge e_j . The active edges form the set $\mathcal{A}$. An occluding edge is also an active edge, so $\mathcal{O} \subseteq \mathcal{A}$. See figure 9 for an example.

Given these properties, we say that the pixels in an occluded region can be occluded only by active edges. This statement implicitly rests on two assumptions:

1. Changes in surface height inside the occluded regions are so small that they cannot cause occlusions.

Figure 9: Definition of edges.

2. Changes in surface height outside the border of the occluded regions are so small that they cannot cause occlusions.

From the properties defined above also the assumption about the height of the occluded pixels will be derived in the next subsection. If the scene is not as we have assumed — i.e. in the scene there also exist points on the surface which are not illuminated during the scanning — then the non-illuminated parts of the scene can shape the occluded regions in such a way that some edges will 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.

4.4 Viewing angles computation

For each pixel p_k in a polygon P , we define the *viewing angle* $\Phi(p_k)$ as the sector containing all directions from which the pixel p_k is visible, assuming, as stated above, that only active edges can cause occlusions. As mentioned in subsection 4.1, an assumption about the pixel height must be made in order to compute the visibility directions for each pixel.

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 (6)). In this case the computed viewing angle $\Phi(p_k)$ can be the same as or greater than the true one. If from the new selected scanning direction the pixel p_k is still occluded, the new active edges are located, which narrow the viewing angles of the first computation.

The weakest assumption seems to be inefficient because it 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 the entirety of the occluded pixels. Suppose that for a set of pixels there exists a direction from which they can be seen together. Too restrictive an assumption can lose this common viewing direction for 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 generally are concave, the areas A_i have a constrained shape that makes it easier to find their pixels. Since each area A_i has only 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:

$$h(p_k) = \begin{cases} h(e_j) & \text{if } e_j \notin \mathcal{A} \\ h_{support} & \text{if } e_j \in \mathcal{A} . \end{cases} \quad (14)$$

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 $\{e_i; e_i \in \mathcal{A}\}$ an *occluding angle* $\Phi_{occ}(p_k, e_i)$ is computed as follows. Knowing the height of the occluded pixel $h(p_k)$ and the height of the obstacle edge $h(e_i)$, the distance r from which the obstacle occludes the pixel can be derived from the equations (3) and (4):

$$r = L \frac{h(e_i) - h(p_k)}{H - h(p_k)} . \quad (15)$$

If the line on which e_i lies intersects the circle around pixel p_k with radius r (fig. 10), then $\Phi_{occ}(p_k, e_i)$ 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 with which the image was obtained then the occluding angles are transformed from IP to ISP by rotating them by φ . The viewing angle $\Phi(p_k)$ is computed by subtracting all the occluding angels $\{(\Phi_{occ}(p_k, e_i))_{ISP}; e_i \in \mathcal{A}\}$ from the unit circle.

The viewing angles are the finite sets of closed intervals (*viewing intervals*)

$$\Phi(p_k) = \{[\varphi_{j_l}, \varphi_{j_u}], [\varphi_{j+1_l}, \varphi_{j+1_u}], \dots\}, \quad (16)$$

Figure 10: Occluding angle computation.

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

4.5 Determination of the next scanning direction

After having defined the viewing angles, we must determine the next scanning directions. If there exists a direction from which the whole occluded region 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 otherwise. The problem can be formulated in the following way: *Having finite sets of closed intervals*

$$\Phi(p_k) = \{[\varphi_{1_l}, \varphi_{1_u}], [\varphi_{2_l}, \varphi_{2_u}], \dots\} \text{ for each pixel } p_k, k = 1, \dots, r,$$

find intervals $\Omega_i = [\omega_{i_l}, \omega_{i_u}]$, with $i = 1, \dots, n_\Omega$, such that by selecting any value from each interval Ω_i at least one value from each set $\Phi(p_k)$ is selected, and their number n_Ω is minimal.

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

- $\forall \mathcal{I}_i \in \mathcal{V} : \mathcal{I}_i \subset [0, 2\pi)$, since each pixel in the occluded regions is occluded at least from one direction,
- Any pair of intervals $\mathcal{I}_i, \mathcal{I}_j \in \Phi(p_k)$ has no elements 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:

$$\xi_i(\varphi) = \begin{cases} 1 & \text{if } \varphi \in [\varphi_{i_l}, \varphi_{i_u}] \\ 0 & \text{otherwise} \end{cases} \quad (17)$$

The interval $[0, 2\pi)$ can be represented by the circumference of a unit circle. All intervals are defined counterclockwise. The symbols $\prec, \succ$ will be used to denote the relations *counterclockwise before* and *counterclockwise after*, respectively.

4.5.1 Selection by histogram

To select the scanning directions from the best partial intersections of the viewing intervals, a histogram is build. 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 $[0, 2\pi)$:

$$\mathcal{H}(\varphi) = \sum_{\{i; \mathcal{I}_i \in \mathcal{V}\}} \xi_i(\varphi). \quad (18)$$

With φ_j and φ_k the lower and 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 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 . $\square$

Lemma 1: *The histogram $\mathcal{H}(\varphi)$ has a constant value on an interval (φ_j, φ_k) if no lower or upper bound of any interval $\mathcal{I}_i \in \mathcal{V}$ falls into (φ_j, φ_k) .*

Proof: The functions $\xi_i(\varphi); \mathcal{I}_i \in \mathcal{V}$ are piecewise constant. They have discontinuities at the interval bounds φ_{i_l} and φ_{i_u} . Since no bound falls on the interval (φ_j, φ_k) , the functions $\xi_i(\varphi)$ have the value 0 or 1 on the whole interval (φ_j, φ_k) . The sum of constants on some interval is again a constant. $\square$

Lemma 2: *If among the intervals $\mathcal{I}_i \in \mathcal{V}$ none of their lower or upper bounds falls into the interval (φ_j, φ_k) then for all $\varphi_m \in (\varphi_j, \varphi_k)$ the following relations hold:*

$$\mathcal{H}(\varphi_j) \geq \mathcal{H}(\varphi_m),$$

$$\mathcal{H}(\varphi_k) \geq \mathcal{H}(\varphi_m).$$

Proof: Because of Lemma 1 the histogram has a constant value on (φ_j, φ_k) , so it is sufficient to prove Lemma 2 only for one $\varphi_m \in (\varphi_j, \varphi_k)$. Suppose that $\mathcal{H}(\varphi_m) > \mathcal{H}(\varphi_j)$. The histogram value can be greater at φ_m than at φ_j only if there exists at least one function $\xi_i(\varphi)$ which is greater

at φ_m than at φ_j ; $\xi_t(\varphi_m) > \xi_t(\varphi_j)$. Because $\varphi_j \prec \varphi_m$, $\xi_t(\varphi)$ must change its value from 0 to 1 somewhere on the interval $(\varphi_j, \varphi_m]$, therefore the lower bound φ_{i_l} of the interval $\mathcal{I}_t$ must lie in the interval $(\varphi_j, \varphi_m]$. Since $(\varphi_j, \varphi_m] \subset (\varphi_j, \varphi_k)$, this contradicts the previous assumption that no lower or upper bound of any interval $\mathcal{I}_i$ falls into (φ_j, φ_k) . The second relation is proven analogously. $\square$

Theorem 1: *In each viewing interval $\mathcal{I}_i(\varphi) = [\varphi_{i_l}, \varphi_{i_u}]$ at least one local maximum can be found in the histogram $\mathcal{H}(\varphi)$.*

Proof: Suppose that $\xi_i(\varphi)$ is the last function that is added to the histogram, and that φ_j is the closest bound which is counterclockwise before the lower bound φ_{i_l} of interval $\mathcal{I}_i$. According to Observation 1 there exists a nonempty interval $(\varphi_j, \varphi_{i_l})$. Since φ_j is the closest bound counterclockwise before φ_{i_l} , there is no other bound on the interval $(\varphi_j, \varphi_{i_l})$, thus the histogram value is constant on the interval $(\varphi_j, \varphi_{i_l})$ (Lemma 1). The histogram value at φ_{i_l} can be the same as or greater than on the interval $(\varphi_j, \varphi_{i_l})$ (Lemma 2).

A similar observation can be made for the upper bound φ_{i_u} of the interval $\mathcal{I}_i$. Suppose that φ_k is the closest bound which is counterclockwise after φ_{i_u} . Again, the histogram value on the interval $(\varphi_{i_u}, \varphi_k)$ is constant. The histogram value at φ_{i_u} can be the same as or greater than on the interval $(\varphi_{i_u}, \varphi_k)$.

After adding the function $\xi_i(\varphi)$ to the histogram, the histogram values increase by one for all $\varphi \in [\varphi_{i_l}, \varphi_{i_u}]$. Therefore the histogram value at φ_{i_l} is greater than on the interval $(\varphi_j, \varphi_{i_l})$, and the histogram value at φ_{i_u} is greater than on the interval $(\varphi_{i_u}, \varphi_k)$:

$$\mathcal{H}((\varphi_j, \varphi_{i_l})) < \mathcal{H}(\varphi_{i_l}),$$

$$\mathcal{H}(\varphi_{i_u}) > \mathcal{H}((\varphi_{i_u}, \varphi_k)).$$

Thus the interval $\mathcal{I}_i = [\varphi_{i_l}, \varphi_{i_u}]$ must contain at least one local maximum of the histogram. Since addition is a commutative operation, the order in which the functions are added does not change the resulting histogram. Therefore each of the functions $\xi_i(\varphi)$ could be added as the last one to the histogram, and the above reasoning applies to all of them. $\square$

With Theorem 1 we proved that by taking one point from each histogram maximum, at least one point is selected from each viewing interval and therefore from each viewing angle.

Definition 1: Suppose that there exists a nonempty intersection of the subset $\mathcal{C}$ of intervals $\mathcal{I}_i, \mathcal{I}_j, \dots, \mathcal{I}_t \subseteq \mathcal{V}$,

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

If the intersection of intervals in $\mathcal{C}$ with any other interval $\mathcal{I}_v \in \mathcal{V} \setminus \mathcal{C}$ is empty, -

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

4.5 Determination of the next scanning direction

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

Observation 2: *If there exists a nonempty intersection of closed intervals $\mathcal{I}_i, \mathcal{I}_j, \dots, \mathcal{I}_t$,*

$$\mathcal{I}_i \cap \mathcal{I}_j \cap \dots \cap \mathcal{I}_t \neq \emptyset,$$

then this intersection is a closed interval $[\varphi_l, \varphi_u]$, with φ_l the lower bound of that interval from $\mathcal{I}_i, \mathcal{I}_j, \dots, \mathcal{I}_t$ whose lower bound lies in all intervals $\mathcal{I}_i, \mathcal{I}_j, \dots, \mathcal{I}_t$ and φ_u the upper bound of that interval from $\mathcal{I}_i, \mathcal{I}_j, \dots, \mathcal{I}_t$ whose upper bound lies in all intervals $\mathcal{I}_i, \mathcal{I}_j, \dots, \mathcal{I}_t$.

Proof: If the intersection of intervals $\mathcal{I}_i, \mathcal{I}_j, \dots, \mathcal{I}_t$ is non-empty, then there exists at least one value φ_k that lies in all viewing intervals $\mathcal{I}_i, \mathcal{I}_j, \dots, \mathcal{I}_t$, i.e.

$$\varphi_{i_l} \preceq \varphi_k \preceq \varphi_{i_u}$$

$$\varphi_{j_l} \preceq \varphi_k \preceq \varphi_{j_u}$$

$$\vdots$$

$$\varphi_{t_l} \preceq \varphi_k \preceq \varphi_{t_u}.$$

It is obvious that φ_k is limited by the most restrictive conditions:

$$\varphi_l \succeq \varphi_{i_l} \text{ and } \varphi_u \preceq \varphi_{i_u}$$

$$\varphi_l \succeq \varphi_{j_l} \text{ and } \varphi_u \preceq \varphi_{j_u}$$

$$\vdots$$

$$\varphi_l \succeq \varphi_{t_l} \text{ and } \varphi_u \preceq \varphi_{t_u}. \square$$

If φ_l is equal to φ_u then the intersection has exactly one point $\varphi_k = \varphi_l = \varphi_u$.

Since the intervals $\mathcal{I}_i, \mathcal{I}_j, \dots, \mathcal{I}_t$ can be the same as or wider than 180° it can happen that their intersection consists of two intervals (i.e. two lower bounds and two upper bounds can be found that lie in all viewing intervals).

Theorem 2: *The complete subsets $\mathcal{C} = \{\mathcal{I}_i, \mathcal{I}_j, \dots, \mathcal{I}_t\}$ form the maxima in the histogram.*

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 $\xi_i(\varphi), \xi_j(\varphi), \dots, \xi_t(\varphi)$ 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 (otherwise they would form an intersection with $\mathcal{C}$). 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 $(\varphi_{j_u}, \varphi_{j_l})$ while $\xi_k(\varphi)$ is 0 on $(\varphi_{k_u}, \varphi_{k_l})$; therefore the sum of $\xi_i(\varphi), \xi_j(\varphi), \dots, \xi_t(\varphi)$ is on interval (φ_u, φ_l) at least 1 less than on the interval $[\varphi_l, \varphi_u]$. None of the other functions $\xi_v(\varphi); \mathcal{I}_v$ cannot completely compensate for

$\xi_j(\varphi)$ and $\xi_k(\varphi)$ on (φ_u, φ_l) , because between φ_u and φ_{v_l} we can always find 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 . $\square$

By selecting the points which lie in the intersections of the complete subsets we take the largest sets of intervals from $\mathcal{V}$ that can be selected together.

The histogram maxima are both the intervals from which each of the viewing angle can be selected and the intervals from which the complete subsets can be selected. We must find the smallest combination of histogram maxima such that all viewing angles are selected. This can be achieved by histogram decomposition.

4.5.2 Histogram decomposition

For each viewing angle we must find the number of maxima it contains. If only one histogram maximum exists on the interval of the viewing angle $\Phi(p_k)$ then $\Phi(p_k)$ belongs to only one complete subset $\mathcal{C}$ and can be selected only by selecting the scanning direction from 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 in the histogram which contains only one maximum then the histogram decomposition must be done for each local maximum. The solution with the minimum number of necessary maxima is selected. The obtained set of the necessary histograms maxima is a solution to our problem.

4.5.3 Making the selection robust

In general the viewing angles can have several viewing intervals, each with at least one maximum (see Theorem 1). Adding up the viewing angles in the histogram can therefore make the maxima very thin (cf. experimental results). Since we are working with a polygonal approximation of the occluded region, thin maxima are undesirable because they might not yield precise enough directions. This drawback can be overcome by first computing the viewing angles for the areas A_i separately. For each area A_i a histogram is then build from these viewing angles. From the necessary maxima of these histograms we build the final histogram from which the next scanning directions are then selected.

4.6 Completion of the analysis of one plane

To compute the new scanning directions from the first view we assume that the changes in surface height in the occluded regions are too small to cause occlusions. The same is assumed for the regions outside the border of the occluded regions. If this is not the case and after combining the

4.6 Completion of the analysis of one plane

Figure 11: Angle β_i .

images of the computed scanning directions in the ISP there still exist occluded regions, then they should be explored from new directions. Since we use only the height information of the borders of occluded regions 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.

For each scanning direction that was taken occluded regions 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 $P(v_1, v_2, \dots, v_n)$ to which the pixel belongs are determined by the following procedure (fig. 11): the angles β_i , with $i = 1, 2, \dots, n$, formed by the lines intersecting in the pixel p_k and going through the neighboring vertices v_{i+1} and v_i are computed

$$\beta_i(v_i, v_{i+1}) = \arctan\left(\frac{y_i - y_k}{x_i - x_k}\right) - \arctan\left(\frac{y_{i+1} - y_k}{x_{i+1} - x_k}\right), \quad (19)$$

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.

$$\sum_{i=1}^n \beta_i = 2\pi, \quad (20)$$

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 thin maxima, first the histograms for pixels belonging to the same set of polygons P are constructed. 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 one plane to get the complete $2\frac{1}{2}$ -D data of the scene. In each scan the illuminating plane illuminates the same part of

Figure 12: Occluding borders, and their projections

the scene surface but the camera sees different parts of it. The points on the surface to which the illumination is tangential (fig. 12) form borders which separate illuminated from non-illuminated surfaces. 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 the $2\frac{1}{2}$ -D image of the first scanning plane. Additional scanning planes must be computed for that purpose. We will look for such scanning planes only among those perpendicular to the first one.

Assume that the scanning plane from which we constructed the $2\frac{1}{2}$ -D data is the top of a cylinder (fig. 13). If the new scanning plane shall be perpendicular to the first one, it can be defined by an angle δ , which is the angle of rotation of the illuminating plane about the center of the top of the cylinder. The direction of the scanning plane normal $\vec{n}$ is equivalent to the direction of the light that illuminates the scene during the scanning; while the first scanning plane presented us with a top view, we are now looking from the sides.

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

5.1 Occluding borders

5.1 Occluding borders

The only connection between the $2\frac{1}{2}$ -D data of the first scanning plane and the non-illuminated surfaces are the occluding borders. Occluding borders are defined by the jumps in height in the $2\frac{1}{2}$ -D image of the first scanning plane. To compute the next scanning planes we will use an approximation of the occluding borders. With an edge operator that is sensitive to C_0 discontinuities we locate the occluding borders. They are then approximated by piecewise linear segments $l_i(p_i, p_{i+1})$, $i = 1, \dots, m$, 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 [1980]. 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 of 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 figure 6, where as 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

$$\psi(l_i) = \arctan\left(\frac{y_{i+1} - y_i}{x_{i+1} - x_i}\right). \quad (21)$$

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 $[\psi(l_i), \psi(l_i) + 180^\circ]_{ccw}$.

5.2 Illuminating angles

Figure 14: Illuminating angle computation

To compute the illuminating angle $\Psi(F_i)$ for each face F_i , we must locate the areas (“islands”) I in the $2\frac{1}{2}$ -D image which are higher than the height of projection $h_p(l_i)$ of the occluding line segment l_i . We find the contours for each l_i of such islands I with the same TRACER algorithm as we found the contours of occluded regions 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 [1983] 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 the following: the direction of movability d is in our case the direction of the light, the obstacles Q are the islands I , 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 $[\psi(l_i), \psi(l_i) + 180^\circ]_{ccw}$. If there is an obstacle island then each line lc_j on face F_i , connecting a point p_j on the occluding line segment l_i with its projection p_p , (fig. 14a), can be illuminated from a different subset of directions from the range $[\psi(l_i), \psi(l_i) + 180^\circ]_{ccw}$. The illuminating angle of the line lc_j for K obstacle islands is:

$$\Psi(lc_j) = [\psi(l_i), \psi(l_i) + 180^\circ]_{ccw} - (\cup_{k=1}^K \alpha_j(I_k)). \quad (22)$$

The angle $\alpha_j(I_k)$ includes all directions from which the island I_k occludes the line lc_j . If we insist on illuminating the whole face F_i at the same time, then the rotation angle δ must be selected from the illuminating angle

$$\Psi(F_i) = [\psi(l_i), \psi(l_i) + 180^\circ]_{ccw} - (\cup_{k=1}^K \gamma_{i,i+1}(I_k)). \quad (23)$$

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

$$\gamma_{i,i+1}(I_k) = (d_i(I_k), d_{i+1}(I_k)) = \bigcup_j \alpha_j(I_k). \quad (24)$$

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 figure 14b.

5.3 Computation of the next scanning plane

The illuminating angles $\Psi(F_i)$ are as viewing angles finite sets of closed intervals — illuminating intervals. The angles δ of the next scanning planes are determined from the histogram in the same way as we computed the new scanning directions in the previous section. 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 structured lighting laser-scanner with 1 mm/pixel spatial resolution and 1.2 mm depth resolution.

Scene 1: The scene contains three polyhedra with different heights and sizes and one half of a cylinder (fig. 15). The first image is taken from the scanning direction 0° (fig. 16 left). From this direction the tallest polyhedron completely occludes the shortest one and the curved surface of the half cylinder is only partially seen.

The occluded regions are approximated by polygons. By searching around the contour in a belt 7 pixels wide, the border height is obtained. The occluding, active, and inactive edges are computed. In figure 16 right the active edges (dark lines) of the polygonal approximations are presented together with the occluded regions (grey areas). For all pixels in the occluded polygons the viewing angles are computed. For each area A the histogram of viewing angles is build (fig. 17, 18 left). From the necessary maxima of these histograms the final histogram is constructed (fig. 18 right). It has two maxima $[148^\circ, 179^\circ]$ and $[271^\circ, 275^\circ]$, both of which are necessary. From each maximum 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. Figure 19 shows the images of the two selected viewing directions. The pictures of all three views are merged in the ISP (fig. 20), yielding the complete $2\frac{1}{2}$ -D image of the scene.

To compute the orientations δ of the next scanning planes, we first locate the edges in the $2\frac{1}{2}$ -D image (fig. 21 left). They represent the occluding borders and are approximated by piecewise linear segments (fig. 21 right). Each of the linear segment approximates one part of the occluding border and its projection. The heights of both are defined by searching 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. 22 left). During the histogram decomposition (fig. 22 right, 23 left) three necessary maxima are found: $[52^\circ, 64^\circ]$, $[151^\circ, 159^\circ]$, and $[278^\circ, 300^\circ]$. 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 figures 23 right, 24 left, and 24 right.

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. 25 left). Occluded regions are approximated by polygons, and the active edges are computed. In fig. 25 right the active edges (dark lines) are presented together with the occluded regions (gray areas). The constructed histogram (fig. 26 left) has four maxima. During the histogram decomposition the necessary maxima among them, $[179^\circ, 187^\circ]$ and $[226^\circ, 230^\circ]$ are selected. Additional images are taken from the directions 183° and 228° (fig. 26 right and fig. 27 left). The images of all three views are merged in the ISP (fig. 27 right).

Figure 15: The scene

Figure 16: Left: image of the first view, taken from the scanning direction 0° ; right: occluded regions (grey areas) and active edges of their polygonal approximation.

Figure 17: Histograms of areas A

Figure 18: Left: Further histograms of areas A; right: the final histogram

Figure 19: Left: the second view; right: the third view

Figure 20: Left: merged images of the first and the second view in the ISP; right: merged images of all three views in the ISP.

Figure 21: Left: edges; right: their linear approximation.

Figure 22: Left: histogram of illuminating angles; right; histogram decomposition.

Figure 23: Left: histogram decomposition; right: light simulation from direction 58°.

Figure 24: Left: light simulation from direction 155°; right: light simulation from direction 289°.

Figure 25: Left; image of the first view taken from direction 0° ; right: occluded regions (grey areas) and active edges of their polygonal approximation.

Figure 26: Left: histogram of viewing angles; right: image of the second view taken from direction 183° .

Figure 27: Left: image of the third view taken from direction 228°; right: images of first, second, and third view in ISP.

Figure 28: Left: second view - occluded region and active edges; right: third view - occluded regions and active edges.

Figure 29: Left: histogram; right: image of the fourth view taken from direction 272°.

Figure 30: Left: image of the fifth view taken from direction 45°; right: images of all five views in ISP.

Figure 31: Left: edges; right: their linear approximation.

Figure 32: Left: histogram of illuminating angles; right: histogram decomposition.

At the place where the polyhedron touches the half cylinder, still small occluded parts 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° (fig. 28), 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. 29 left). Two additional views are taken from the direction 272° (fig. 29 right) and 45° (fig. 30 left). All five views together yield the complete $2\frac{1}{2}$ -D image of the first scanning plane (fig. 30 right).

For the lateral scanning planes the edges are located in the ISP (fig. 31 left) and approximated by linear segments (fig. 31 right). 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 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. 32): $[1^\circ, 2^\circ]$, $[37^\circ, 45^\circ]$, $[183^\circ, 190^\circ]$, $[240^\circ, 240^\circ]$, $[6^\circ, 12^\circ]$, $[282^\circ, 288^\circ]$, $[304^\circ, 308^\circ]$.

7 Conclusions

The camera-laser configuration of the light stripe range finder allows to measure the distance only of those portions of 3-D scene which are simultaneously illuminated by the light and visible to the camera. Two types of occlusions can appear in a range image: parts which are illuminated during the scanning but not visible to the camera cause missing data in the initial $2\frac{1}{2}$ -D range image, while parts which are not illuminated during the scanning are missing data which will show up in the the 3-D spatial map of the scene. We have developed an algorithm for selecting the proper views to produce the complete $2\frac{1}{2}$ -D image of the scene. From the $2\frac{1}{2}$ -D data acquired in the first scanning plane we can then compute new scanning planes for complete 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. Even if we would allow arbitrary scanning planes, the exploration of such regions would be limited because the camera-laser configuration is fixed and sensor geometry can not reach deeply hidden structures. Perhaps in this case the easier solution is to use different sensors, for example tactile sensors, which are better suited to examine the shape of such regions.

The next step of an autonomous exploration is to build the scene representation. In our work we separate data acquisition from model building. To make further decisions we explore the knowledge of the sensor geometry. The basic question is: can we build a model of the scene during the data acquisition process which can help us to make better decisions for the next data gathering? The

major problem is to find an expressive model for shape description which can be easily 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 situation at least a coarse model constructed from the current data is desirable. Imagine an exploration system where different sensors can gather the data. If not all geometric data can 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.

In our future work we will interpret the scene geometry by volumetric models — superellipsoids. Based on the current interpretation of the scene we will try to find rules for further sensing action.

Acknowledgments

The authors wish to thank Ulf Cahn von Seelen for comments on a draft of this paper.

This research was supported in part by AFOSR Grants 88 0244, AFOSR 88-0296; Army/DAAL 03-89-C-0031PRI; NSF Grants CISE/CDA 88-22719, IRI 89-06770; ARPA Grant N0014-88-K-0630 and Dupont Corporation.

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] C.I.Connolly, "The Determination of Next Best Views," in *Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 432–435, 1985.
- [3] 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.
- [4] S.Fisk, "Short proof of Chvatal's Watchman Theorem," *J. Combinatorial Theory*, ser. b, vol. 24, p. 374, 1978.
- [5] 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.
- [6] K.Ikeuchi and J.C.Robert, "Modeling Sensor Detectability with Vantage Geometric/Sensor Modeler," *Image Understanding Workshop*, May 1989, pp. 721–746.

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

R.Nevatia and K.R.Babu, "Linear Feature Extraction and Description," *Computer Graphics and Image Processing*, Vol.13, 1980, pp. 257-269.
- [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, 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] 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 18-19, 1983.
- [14] C.J.Tsikos, *Laser Range Imaging System User's Guide*, 1989.
- [15] P.Whaite and F.P.Ferrie, "From Uncertainty to Visual Exploration," *ICCV Conference 1990*, pp. 690-697.