Proceedings of 86 IEEE International Conference on Robotics and Automation,
April 7-10, SanFrancisco. pp. 1426-1432.

Outdoor Scene Analysis Using Range Data

Martial Hebert
Robotics Institute

and

Takeo Kanade
Computer Science Department

Carnegie Mellon University
Pittsburgh PA15213

Abstract

This paper describes tecliniques tor outdoor scene analysis using range data. The purpose of these techniques is to build a 3.0 representation of the environment of an mobile robot equipped with a range sensor. Algorithms are presented for scene segmentation. object detection, map building, and object recognition.

We present results obtained in an outdoor navigation environment in which a laser range finder is mounted on a vehicle. These results have been successfully applied to the problem of path planning through obstacles.

1. Introduction

Outdoor scene analysis is a major part of an autonomous vehicle. It allows a vehicle to navigate safely through an unknown environment, to build a internal model of the environment, and to identify landmarks. Several sources of data can be used to build such a vision system: range finders, color cameras. sonars. While a reliable system would probably use all of these Sensors, our work focused on the use of range data for outdoor vision. The advantages of using range data for an autonomous vehicle fall in two categories: first, the measurements are less sensitive to environmental conditions, such as lighting. thus alleviating shadow or highlight problems. Second, a geometrical description of the observed scene is more easily derived from range data. This property is important in the area of autonomous vehicles in which the output of vision programs must be converted into usable space coordinates for navigation. For our work, we use the ERIM laser range sensor which provides reasonable accuracy, a field of view large enough for outdoor applications, and high acquisition speed.

Following the description of the ERIM range sensor in the next Section, we present preprocessing techniques for removing sensor-dependent defaults. In Section 4, we present the 3-D features extraction algorithms which are designed to produce relevant features for outdoor vehicle navigation and object recognition. 3-D map reconstruction from range data is described in Section 5. In Section 6, we present algorithms for data fusion, that is, algorithms that use range data along With other types of visual information such as color images. In Section 7. we present one important application of range data analysis to robot vision, that is, object recognition using data fusion between range and color sensors.

2. Sensor Description

The range sensor we use for outdoor scene analysis has been designed by the Environmental Research Institute of Michigan and will be referred to as the ERIM sensor. The basic principle of the sensor is to determine the range from the sensor to the scene point for each pixel by measuring the transmit time of a modulated laser beam. The transmit time is derived by measuring the phase difference between the reference and reflected signals which corresponds to the range from the Sensor to the target. A two-mirrors scanning mechanism directs the beam onto the scene so that an image of the scene is produced. In the ERIM-ALV version, the field of view is ±40° in the horizontal plane and 30° in the vertical plane, from 15° to 45°. The resulting range image is a 64 x 256 8-bit image. The frame rate is currently two images per second. The nominal range noise is 0.4 feet at 50 feet.

Since only the phase shift is measured, the resulting values are relative instead of absolute measurements. That is, two points separated by a length equal to a complete phase shift have the same range value. This critical length is called the ambiguity interval and is equal to 64 feet.

The sensor is also capable of producing reflectance images in which the value of each pixel is the amount of light reflected by the target. This information has not been used yet.

Figure 2-1 shows a sequence of seven ERIM images taken in a park: two consecutive images are taken from positions separated by approximately five meters.

3. Preprocessing

The ERIM data introduces a periodicity problem due to the ambiguity interval. The periodicity is especially apparent in images such as the one shown in Figure 3-1, in which distant points have the same value as close points. This problem reduces the range at which the scene can be processed to the extent of the ambiguity interval, and may also create false features, such as false edges which do not correspond to any physical feature. Therefore, the first step in the ERIM image processing is to remove the periodicity. The periodicity removal algorithm has three steps:

1. Divide the image into connected components so that two points whose range difference is greater than a threshold are never connected (Figure 32). Two

This research is supported by the Defense Advanced Research Projects Agency (DOD) and monitored by the Office of Naval Research under Contract N00014-82-K-0193 and by contract DACA76-85-C-0003 issued by the U. S. Army Engineer Topographic Laboratories.

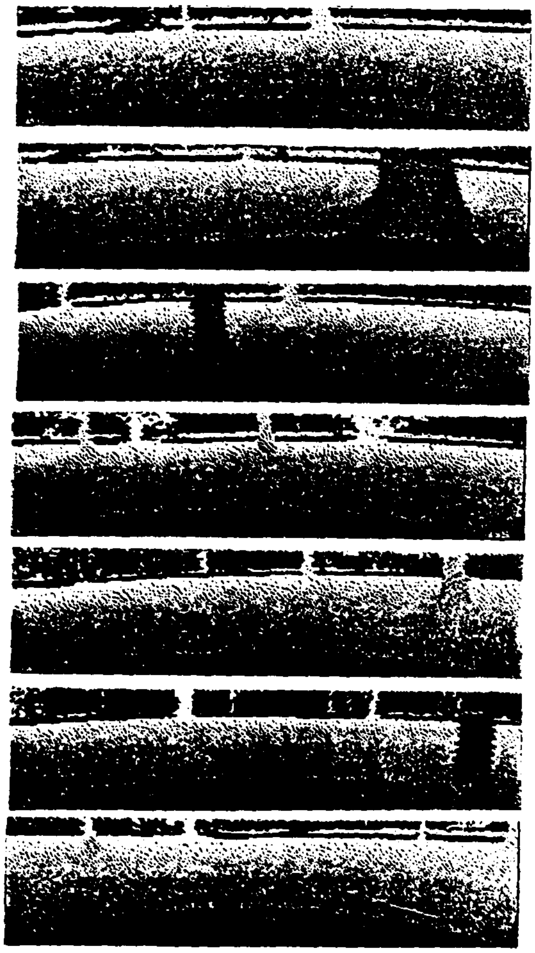

Figure 2-1: A sequence of ERIM images

such points belong to two different ambiguity intervals.

2. Remove the small regions which correspond to noise.
3. Explore the graph of components starting at the bottom of the image which is within the first ambiguity interval. During the exploration, an offset is added to all the points of the currently visited region. The offset is initially set to zero and is updated for each region by applying the following rule:

$$\text{offset}(S) = 256 + \min\{\text{offset}(S'),\ S' \text{ connected to } S\}$$

The result of the correction is shown in Figure 3.3.

We have found that this algorithm works well within the two first ambiguity intervals. Beyond that point, measurements are usually too noisy to ensure reliable results. No algorithm is guaranteed to retrieve the actual range values since it is unknown whether two regions are separated by only one or several ambiguity intervals. The algorithm assumes that only one interval separates two regions.

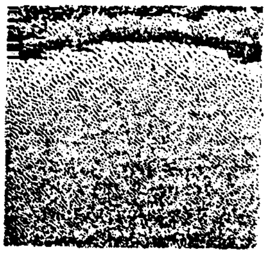

Figure 3-1: Uncorrected image

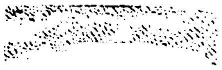

Figure 3-2: Ambiguity intervals

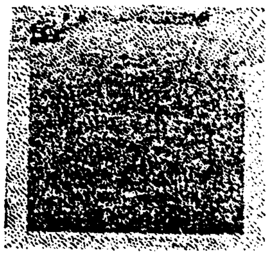

Figure 3-3: Corrected image

4. 3-D Features Extraction and Image Interpretation

In this section. we describe the feature extraction techniques Three levels are used in the feature extraction process First three basic attributes are extracted from the range image. lump edges, surface normals, and curvatures. These attributes are then used lor extracting a set of three primitive features

- 3-D edges which correspond either to discontinuities of depth such as the boundary of an obstacle, or to discontinuities of surface normals such as the boundary between a fiat region and a highly curved region.
- Smooth regions which are regions of low uniform curvature. These regions may either be part of an object, or obstacle, or part of the ground.
- Rough regions which are regions of high curvature or textured regions.

Finally, these intermediate features are merged into a scene interpretation suitable for vehicle navigation. The two main primitives of the output description are:

- Passable regions which can be safely explored by a vehicle.
- Obstacles which are either actual objects or rough ground regions.

The output structure from this feature extraction process is a connectivity graph of these features.

4.1. Surface Normals

The surface normals provide important pieces of information about the shape of the observed terrain. The best way of computing the surface normals is to approximate the neighborhood of each pixel by a plane. A straightforward method would be to minimize for each pixel:

$$\sum_{I,K,N} \alpha_{ij} \| \vec{v} \cdot \vec{p}_{ij} - D \|^2 \qquad (1)$$

where N is the size of the neighborhood, $\vec{v}$ is the surface unit normal, D is the normal distance between the origin and the plane, α_{ij} are weighting factors, and $\vec{p}_{ij}$ are the measured points. Although simple, this procedure is time-consuming. Moreover, it does not take into account the fact that the ERIM scanner delivers the radial distances instead of the Cartesian coordinates. The alternative and preferred criterion is:

$$\sum_{i,K,N} \alpha_{ij} \| \vec{V} \cdot \vec{u}_{ij} - \frac{1}{d_{ij}} \|^2 \qquad (2)$$

where $\vec{V} = \vec{v}/D$, $\vec{u}_{ij}$ is the radial vector at pixel i,j, and d_{ij} is the distance from pixel ij to the origin.

The solution of (2) is given by:

$$\vec{V} = M^{-1} \sum_{i,K,N} \alpha_{ij} \frac{\vec{u}_{ij}}{d_{ij}} \qquad (3)$$

where M is the 3 X 3 matrix defined t y

$$M = \sum \alpha_{ij} \vec{u}_{ij} \vec{u}^{\,t}_{ij}$$

Since the vectors $\vec{u}_{ij}$ depend only on the scanning parameters of the sensor, the matrix M^{-1} can be computed beforehand. Actually, the vectors and matrices depend also on the orientation of the sensor (pan and tilt angles). but their value can be updated easily: if the sensor is rotated by a rotation R, then the radial vectors $\vec{u}$ and the resulting normal $\vec{V}$ are changed to $R^t\vec{u}$ and $R^t\vec{V}$, respectively.

In summary, the estimation of the surface normals proceeds as follows:

1. If the orientation of the sensor has been changed since the last image, update the vectors $\vec{u}$.

2. Correlate the inverse distances $1/d_{ij}$ with $\vec{u}$ with weights α_{ij}.

3. Multiply the resulting vector image by M^{-1}.

4. Normalize the resulting vector to obtain the unit surface normal.

Figure 4-1 shows a range image and the three components of the surface normal estimated by using a 5×5 window.

4.2. Curvatures

4.2.1. Using Principal Curvatures

Several authors have shown that differential geometry, namely the theory of principal curvatures. can be used to recover properties of a surface observed by a range sensor [6, 5]. These properties range from the extraction of roof edges to the extraction of cylindrical surfaces. However, these techniques do not work well for outdoor imagery. The perequisites of these techniques are that an accurate estimation of second order differential attributes is possible and that the surfaces are mathematically well defined. Outdoor imagery has a limited accuracy, and the observed surfaces usually do not have a well-defined mathematical representation since in a natural environment, most surfaces, such as a grassy terrain and tree foliage, are highly textured and irregular. Therefore, we limit ourselves to the computation of curvatures for the purpose of roughly segmenting the scene into separate regions, each of which is a region of low and uniform maximum curvature.

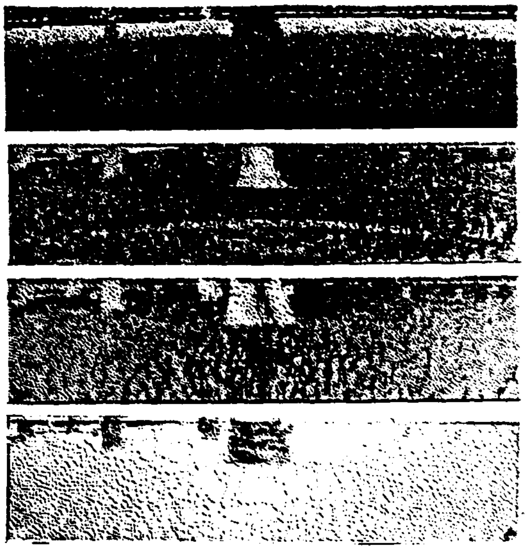

Figure 4-1: Original range image and surface normals

4.2.2. Computing Curvatures

The computation of principal curvatures can be reduced to the computation of first and second derivatives of the image (see [2] for a complete definition). The standard way of computing the curvature is to define the range image as a function $Z=f(X,Y)$, where the two coordinates X and Y are assumed to be uniformly distributed dong the image rows and columns. This assumption is not true when using the ERIM sensor because of the width of the field of view. The solution is to use the spherical representation $\vec{p} = d\vec{u}(\varphi,\theta)$ where $\vec{p}$ is a measured point on the surface, d is the range given by the ERIM sensor, and $\vec{u}(\varphi,\theta)$ is the radial vector of angles φ and 8. A surface is considered as a parametric surface $\vec{p} = F(\varphi,\theta)$. The curvatures can be computed by using the first and second derivatives of F with respect to the two angles:

$$\frac{\partial \vec{F}}{\partial \varphi} = \frac{\partial d}{\partial \varphi} \vec{u}(\varphi,\theta) + d \frac{\partial \vec{u}(\varphi,\theta)}{\partial \varphi} \qquad (4)$$

The radial vectors $\vec{u}(\varphi,\theta)$, and hence their derivatives, depend only on the characteristics of the sensor and its orientation. These vectors can therefore be computed beforehand, thus minimizing the amount of computation at run time. The Curvatures cornputation is thus reduced to the computation of the first and second derivatives of the iniage d with respect to the angles φ and 8. This computation is done by first applying a Gaussian smoothing, and then computing the derivatives by convolving with 3x3 masks. These masks are derived from a second order approximation of $\vec{p} = F(\varphi,\theta)$. The curvature computation algorithm proceeds as follows.

1. Apply a Gaussian smoothing on the range image.

2. Compute the derivatives of the range image with respect to the two spherical angles by applying 3 x 3 operators.

3. Derive the derivatives of the three Cartesian coordinates by using equations (4).

4. Derive the curvatures by applying the fundamental forms equations (see[2]).

Figure 4-2 shows a range image and the corresponding maximum curvature image estimated by using the above algorithm.

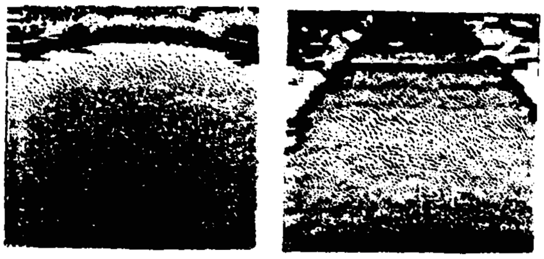

Figure 4-2: Original range image and curvature image

4.3. Edges

Edges are computed by detecting depth jumps. This is done by first detecting the zero-crossings of a difference of Gaussian masks in the depth image. One difficulty is that if two pixels are far from the sensor. they may appear as neighbors in the image wen though they may be far apart in space and hence do not correspond lo edges. This problem is overcome by considering only points of the image with sufficiently high curvature as potential edge points. This algorithm detects mainly the jump edges corresponding to lhe boundaries of vertical obstacles. A more elaborate edge finder for detecting discontinuities of the surface normals is based on region segmentation.

4.4. Segmentation Algorithm

The segmentation algorithm proceeds by combining partial segmentations obtained from attributes, such as edges, surface normals and curvatures. into a consistent scene segmentation. The advantage of this approach is that it takes into account all the available information while dividing the whole segmentation problem into smaller ones.

The initial segmentations are produced for each of the surface normal components and curvatures by a three step region growing algorithm:

1. Find clusters in the attribute space, such as clusters in me surface normals space.

2. Identify the regions corresponding to those clusters in the original image.

3. Use these regions as starting regions for a region growing algorithm. The edges are used as region boundaries in the region growing,

The segmentations obtained from individual attributes are then merged together. This merging step compares each region of one segmentation to the corresponding region in the image of the other segmentations. If the segmentations agree, the region is reported, otherwise it is split into connected regions consistent with the other segmentations. Figure 4-3 shows the segmentation of a short sequence of ERIM images. The regions are labeled as smooth, i.e., accessible portions of the terrain, and *obstacle*, i.e., objects in the scene. The first principal direction of a region is attached to the region in the display.

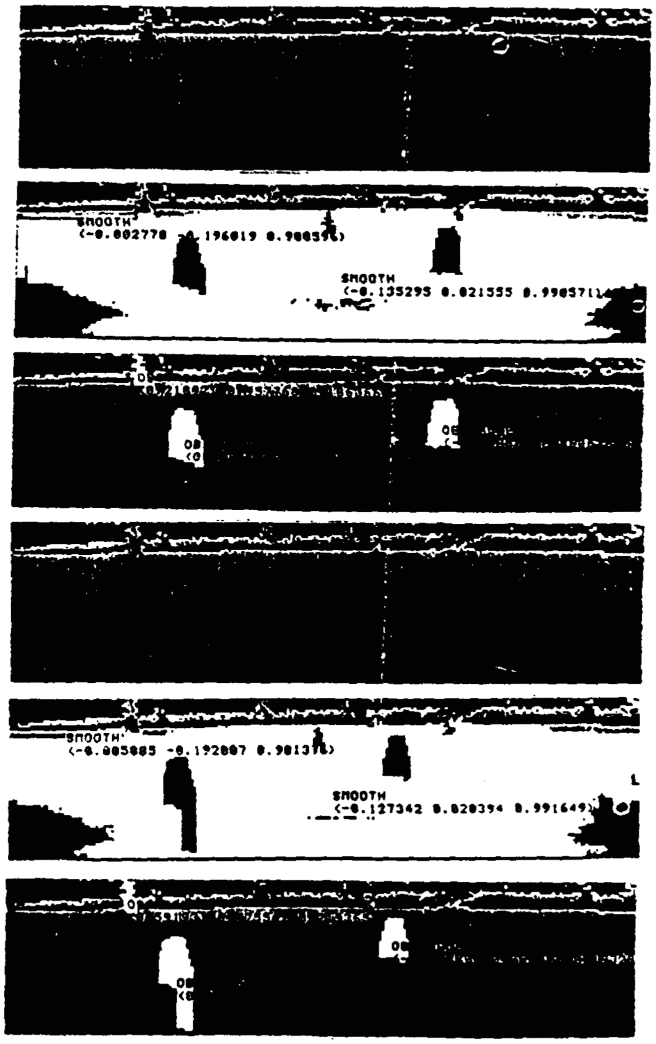

Figure 4-3: Segmented sequence

5. Surface Reconstruction and 3-D Map Building

In the previous section we presented tecliniques for extracting information for ERIM images. In this section. we present techniques for storing this information as a local 3-D *map* of the environment. A map is a structure describing the geometry of the environment from which one can derive information such as the type of terrain at a given space location *x,y,z*.

5.1. Snapshot Map

One important characteristic of a range image is that it permits the production of a three-dimensional map of the current local environment. Such a map, called a snapshot map, is derived from only one image and can be viewed as the local state of the environment. This map can in turn be used to predict the appearance of the scene from another viewpoint. and to plan a safe path for a vehicle while taking into account the 3-D shape of the traversed terrain. The path planning is especially important in cross-country navigation where the "flat ground" assumption does not usually hold.

Figure 5-1 shows an example of such a map and the corresponding ERIM image. The map is displayed as a 3-D mesh of measured points with the surface normal at each point.

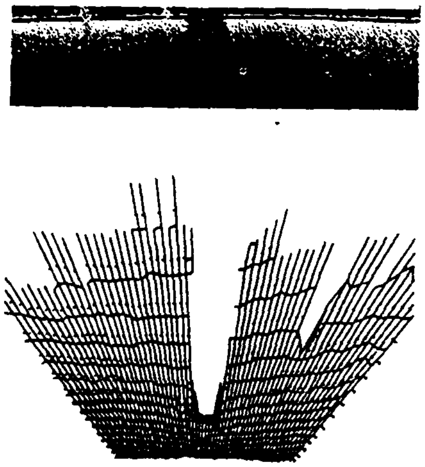

Figure 5-1: Snapshot map derived from an erim image

5.2. Local Map

All the techniques described so far proceed by independently processing one image at a time. In an outdoor navigation system, consecutive images are related to each other to develop a local map. That is, the robot grabs an image every one to ten meters as in the sequence shown in Figure 2-1. Then each image is registered with respect to the previous ones. In other words, we merge the snapshot maps produced by each individual image into a local map describing the environment explored so far. Such a local map can be used for two main purposes:

- *Incremental description refinement:* A single image provides only partial information about the identified objects. For example, only the front part of the tree is visible in figure 5-1, and, as a result, no information is available in the cone-shaped unknown region behind it. Putting together Snapshot maps obtained from several different viewpoints would refine the objects' descriptions by reducing the size of the unknown regions.

- *Incremental construction of a reference map:* One application of outdoor vision is the exploration scenario. A vehicle equipped with sensors discovers an unknown environment and stores information in a reference map usable during later missions.

Local map building is considered as a matching process between consecutive images:

1. Assume that frame 1 is part of the local map and registered in some global coordinate system.

2. Extract features from frame 2. These features are planar regions and edges approximated by polygonal chains.

3. Match the features of images 1 and 2 by using a tree search procedure guided by the transformation estimation. This procedure is similar to the ones described in [1] and [3].

4. The matching provides an estimate of the transformation between images 1 and 2, which in turn provides an estimate of the position of image 2 with respect to the local map.

5. Include the Snapshot map derived from image 2 into the local map. That involves the identification of overlapping regions and the updating of objects' descriptions.

We have done experiments on steps 1 through 4 using ERIM data. Figures 5 2 and 5-3 show two consecutive frames. The matching of tho two images leads to a transformation estimate which is applied to the second snapshot map. Figure 5-4 shows the two registered snapshot maps. Finally. overlapping regions are identified. thus leading to the updated map of figure 5-5. In this example, the map updating has been local at the pixel level and does not include the updating of the symbolic description

Matching the features is efficient since the number of features is usually small. and the transformation between images is partially known beforehand. That is, bounds on the displacement between two consecutive frames are available.

Figure 5-2: First image

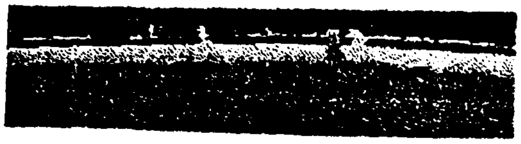

Figure 5-3: Second image

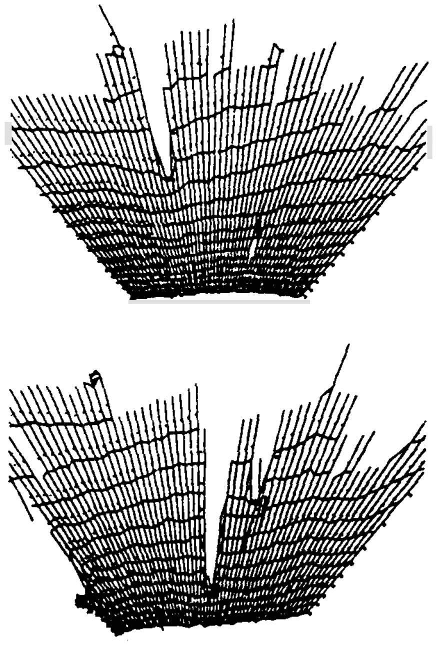

Figure 5-4: Registered snapshot maps

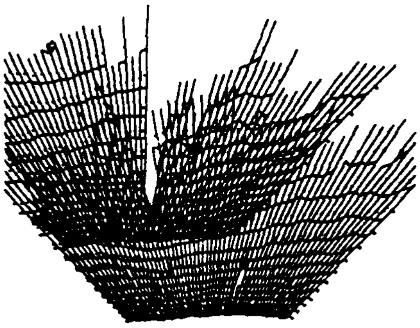

Figure 5-5: Local map

6. Data Fusion

Range data provides important clues on the geometry of an observed scene. However, it does not provide any information about the physical properties of the scene objects, such as color or reflectance. On the other hand, it is extremely difficult to extract spatial information from reflectance information. Therefore. a fundamental feature of a powerful outdoor vision system is the ability of reasoning on both range data and color data. This feature is refered to as “data fusion”. Data fusion can be first reduced to a registration process between range data and other images. That is. one must be able to convert a spatial feature extracted from range data, such an edge or a surface patch, to a feature in the corresponding color images. Conversely. one must be able to compute the spatial position of a feature from the color image by computing the corresponding feature in the range image. The registration is illustrated in Figures 6 1 to 63 Figure 6-1 shows a range image and the corresponding segmentation into three surface patches, Figure 6-2 shows a color image of the same scene (only the blue band is displayed), and Figure 6.3 shows the three surfaces patches registered with the color image.

The registration algorithm works in two steps:

1. Compute the transformation between color images and range images.
2. Use this transformation to transform features from one image to the other.

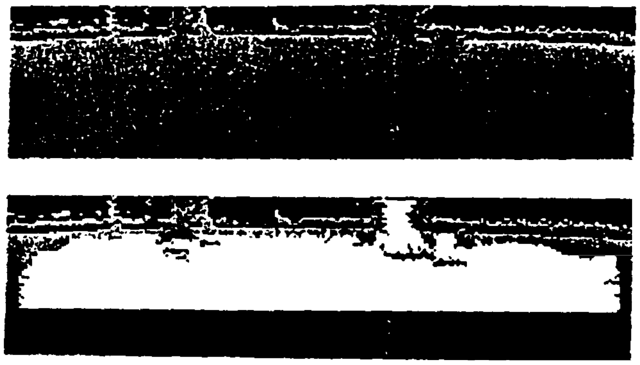

Figure 6-1: Segmented range image

The first step is a calibration step that is performed only once for each set of images. A pair of range/color images is first measured, then a set of points $P_i = (x_i, y_i, z_i)$ is selected in the range image along with the corresponding set of pixels (I_i, J_i). The translormation T, which includes the tilt angle of the camera, the position of the camera relative to the range scanner, and its focal length. is computed by solving a least-squares criterion.

Once the camera/scanner transformation is computed, the 3-D segmentations can be registered with the color image. This is done by applying the transformation to each point (x,y,z) of each surface patch in the Erim image, thus transforming each patch into a set of scattered pixels (I,J) in the color image. These pixels

Figure 6-2: Color image

Region 1
Quadric :
Diameters : (0 02 001 0.00)
Global normal (0.98 -0.1 -0.16)
Long axis : (0.2 0.7 0.96)
Otype Ellipsoidal cylinder

Edges :
type : jump,jump, and internal
connrciivity : region 0
Color : tree

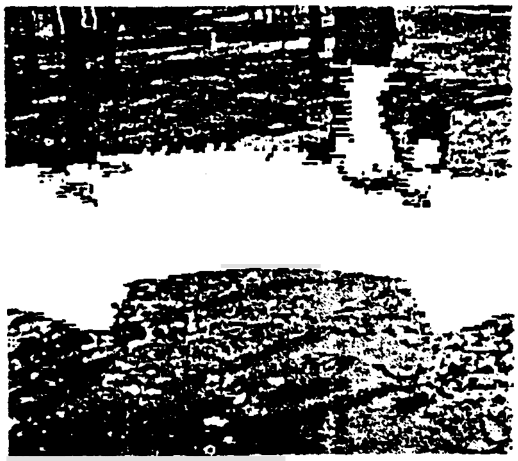

Figure 6-3: Registration of surfaces and color image

into a polygonal mesh. Each face of the mesh is then filled to produce a compact connected region in the color image as in Figure 6-3.

7. Object Recognition

Object recognition, that is the identification of specific objects in the scene by matching extracted features and stored models, is of central importance in the development of an autonomous vehicle. For example, object recognition capabilities can be used in a landmark-drivennavigation scenario. Our approach is to use both range and color data for object recognition. This choice is motivated by the fact that these two types of data are complementary: range data provides information on the geometry and the position of objects, while color data provides information on the physical structure of the objects. Moreover, the techniques presented in Section 6 make possible to relate features extracted from both sources.

7.1. Scene Description

The current object recognition system takes the range image segmentation as the primary information. The segmentation is described by a graph of edges and surface patches approximated by quadric patches. The surface patches are then converted into regions in a color image using the techniques of Section 6. The only color processing currently supported is the segmentation of each region in the color image using a color classifier. The resulting range/color image description is a description file which contains the graph of 3-D edges and surface patches along with the segmentation of each patch into subpatches according to the color characteristics.

7.2. Object Description

An object to be recognized is stored as a set of conditions that its Shape and color must satisfy to be present in the scene. As a simple example, a tree trunk should satisfy condilions such as: cylindrical shape. connected to the ground, parallell vertical edges, color in a given region of the color space. The reason why we are using heuristic models instead of more accurate object models is that we are dealing with natural scenes instead of well-defined man-made environments. Moreover. we are interested at this point in recognizing classes of objects rather than specific instances.

7.3. Object Recognition

The object recognition proceeds by evaluating the rules describing the sought object on each object found in the scene. This is done by extracting the relevant parameters, such as the quadric parameters or the color type, from the description file and evaluating the current condition. An instance of the sought object is found if an object in the scene satisfies the conditions. Figure 6-3 shows the description of one region extracted from the image of Figure 6-2. The region is identified as a tree since it satisfies the four rules describing that object (Section 7.2).

The approach currently used in the object recognition system is entirely bottom-up since the scene description is independent of the object model and cannot be modified. A more involved version will contain model-driven feature extraction algorithms.

8. Conclusion

The techniques presented in this paper have been experimented in a realistic outdoor environment by mounting the sensor on a mobile robot [4]. The scenes descriptions produced by these techniques have been successfully used as an input to a path planning and obstacle avoidance module. In addition, we have shown that data fusion between range and color data can provide an higher level scene description. The results indicate that active range data processing is suitable for the navigation through an unknown environment. Future work includes the combination of range data with other sources of visual data, such as reflectance, and the integration of the scene analysis techniques in a general navigation system which includes landmark recognition, obstacle detection, and map management.

Acknowledgments

Mike Blackwell installed the hardware and software environment of the ERIM sensor. The CMU Civil Engineering Lab provided the testbed vehicle. Takayoshi Obatake implemented part of the 3 D map builder. InSo Kwron implemented the data fusion algorithms. The CMU Image Understanding group provided helpful discussions and support

References

[1] Faugeras O.D., Hebert. M
A 3D recognition and positioning algorithm using geometrical constraints between primitive surfaces.
In *Proc. 8th Int Joint Coril On Artificial Intelligence*. pages 996 1002 Karlsruhe, August. 1983.

[2] Faux. I.D., Pratt. M.J.
Computational Geometry lor Design and Manufacture
Ellis Horwood. 1979.

[3] Hebert. M.. Kanade, T.
The 3-D Profile Method for Object Recognition.
In *Proc. CVPR'85*, pages 458-464. San Francisco. June. 1985

[4] Kanade. T., Thorpe, C.
CMU ALV *Project Report: 1984 to 1985*.
Technical Report. Carnegie Mellon University. 1986

[5] Medioni, G.. Nevatia, *R.*
Description of 3-D Surfaces Using Curvature Properties
In *Proc. Image Understanding Workshop.* DARPA, Science Applications, McLean, Va., 1984.

[6] Ponce, J., Brady M.
Toward a Surface Primal Sketch.
In *Proc. Intern. Conf. on Robotics and Automation*. St.Louis. March, 1985