SENSOR FUSION OF RANGE AND REFLECTANCE DATA FOR OUTDOOR SCENE ANALYSIS *

In So Kweon Martial Hebert Takeo Kanade

The Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213

Abstract

In recognizing objects in an outdoor scene, range and reflectance (or color) data provide complementary information. This paper presents the results of experiments in recognizing outdoor scenes containing roads, trees, and cars from the CMU Navlab (Navigation Laboratory) project. The recognition program uses range and reflectance data obtained by a scanning laser range finder, as well as color data from a color TV camera. After segmentation of each image into primitive regions, models of objects are matched using various properties.

1 Introduction

In order to be able to handle a variety of environments and tasks, a mobile robot must be able to extract features from its visual sensors that enable it to correctly interpret its environment. Many sensing strategies are now possible, from standard video cameras to sophisticated ranging systems. A single sensing modality cannot provide enough information to interpret outdoor scenes however. Geometrical data from ranging systems is necessary for describing the shapes of the observed objects, but some type of reflectance data is also necessary to properly analyze their physical properties, such as terrain type or surface markings. Combining different types of sensor data is not an easy task however. It involves the combination of data sets that are measured by sensors with different characteristics of field of view, range, and accuracy. It also involves the combination of sets of informations of different nature that have been extracted using very different algorithms, such as the combination of surface patches and color edges, for example.

In this paper, we investigate ways to combine geometrical informations from a laser range finder with physical informations from a color camera and an active reflectance sensor (Actually the reflectance images are also provided by the range sensor). We demonstrate the approaches to the combination of those sensors in two examples. The first one concerns the analysis of outdoor for the recognition of natural objects, such as trees, for which only weak models exist. The second one is the recognition of landmarks for which an accurate geometric model is available. In both cases, the combination of shape and reflectance information provides a better, more reliable, interpretation of the sensor data. We have implemented all the techniques described in this paper on the CMU Navlab (Navigation Laboratory) which is a self-contained mobile robot designed for navigation in outdoor terrain [12].

*This research was sponsored in part by the Defense Advanced Research Projects Agency, DoD, through ARPA Order 5351, monitored by the US Army Engineer Topographic Laboratories under contract DACA76-85-C-0003, by the National Science Foundation contract DCR-8604199, by the Digital Equipment Corporation External Research Program, and by NASA grant NAGW-1175. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the funding agencies.

2 Description of the sensors

In this Section, we describe the geometry and the outputs of the two sensors that we use: a laser range finder, and a color camera. Even though some of the characteristics are fairly specific to the particular sensor, the geometries and noise models of the sensors are representative of a wide range of existing visual sensors.

2.1 The range and active reflectance sensors

The basic principle of active sensing techniques is to observe the reflection of a reference signal (sonar, laser, radar..etc.) produced by an object in the environment in order to compute the distance between the sensor and that object. In addition to the distance, the sensor may report the intensity of the reflected signal which is related to physical surface properties of the object. In accordance with tradition, we will refer to this type of intensity data as "reflectance" data even though the quantity measured is not the actual reflectance coefficient of the surface.

Active sensors are attractive to mobile robots researchers for two main reasons: first, they provide range data without the computation overhead associated with conventional passive techniques such as stereo vision, which is important in time critical applications such as obstacle detection. Second, it is largely insensitive to outside illumination conditions, simplifying considerably the image analysis problem. This is especially important for images of outdoor scenes in which illumination cannot be controlled or predicted. For example, the active reflectance images of outside scenes do not contain any shadows from the sun. In

addition, active range finding technology has developed to the extent [1] that makes it realistic to consider it as part of practical mobile robot implementations in the short term.

The range sensor is a time-of-flight laser range finder developed by the Environmental Research Institute of Michigan (ERIM). The basic principle of the sensor is to measure the difference of phase between a laser beam and its reflection from the scene [7]. A two-mirror scanning system allows the beam to be directed anywhere within a $30° \times 80°$ field of view. The data produced by the ERIM sensor is a 64×256 range image, the range is coded on eight bits from zero to 64 feet, which corresponds to a range resolution of three inches. In addition to range images, the sensor also produces active reflectance images of the same format ($64 \times 256 \times 8$ bits), the reflectance at each pixel encodes the energy of the reflected laser beam at each point. Figure 1 shows a pair of range and reflectance images of an outdoor scene.

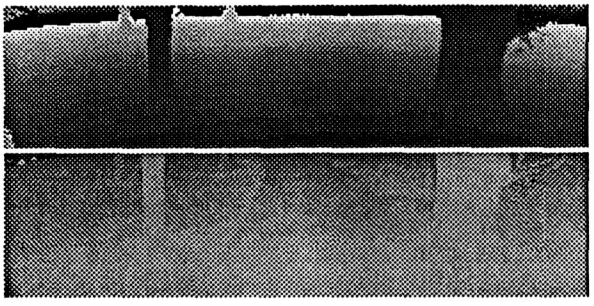

Figure 1: Range and reflectance images

The position of a point in a given coordinate system can be derived from the measured range and the direction of the beam at that point. We usually use the Cartesian coordinate system shown in Figure 2, in which case the coordinates of a point measured by the range sensor are given by the equations:

$$x = D \sin\phi cos\theta \quad (1)$$

$$y = D \cos\phi \cos\theta \quad (2)$$

$$z = D \sin\theta \quad (3)$$

where ϕ and θ are the vertical and horizontal angular angles of the beam direction. The two angles are derived from the row and column position in the range image (r, c), by the equations:

$$\theta = \theta_0 + c \times \Delta\theta$$
$$\phi = \phi_0 + r \times \Delta\phi \quad (4)$$

where θ_0 (resp. ϕ_0) is the starting horizontal (resp. vertical) scanning angles, and $\Delta\theta$ (resp. $\Delta\phi$) is the angular step between to consecutive columns (resp. rows). Figure 3 shows an overhead view of the scene of Figure 1, the coordinates of the points are computed using Equ. (4).

As is the case with any sensor, the range sensor returns values that are measured with a limited resolution which are corrupted by measurement noise. In the case of the ERIM sensor, the main source of noise is due to the fact that the laser beam is not a

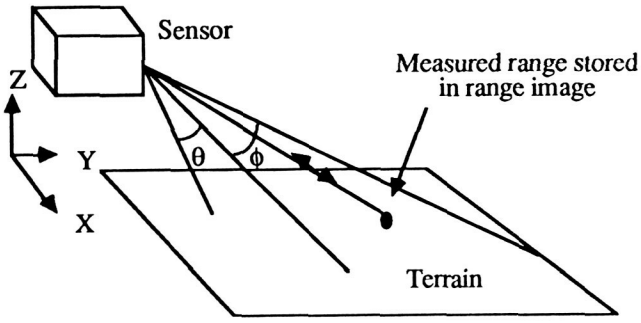

Figure 2: Geometry of the range sensor

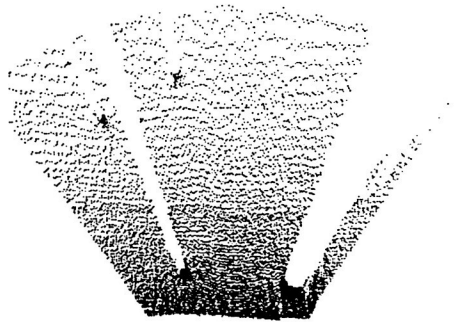

Figure 3: Overhead view

line in space but rather a cone whose opening is a 0.5° solid angle (the instantaneous field of view). The value returned at each pixel is actually the average of the range of values over a 2-D area, the *footprint*, which is the intersection of the cone with the target surface. Simple geometry shows that the area of the footprint is proportional to the square of the range at its center. As a result, the accuracy of the sensor degrades rapidly as the measured points are further away from the sensor which makes the feature extraction a difficult task. The footprint affects all pixels in the image.

There are other effects that produce distortions only at specific locations in the image. The main effect is known as the "mixed point" problem in which the laser footprint crosses the edge between two objects that are far from each other. In that case, the returned range value is some combination of the range of the two objects but does not have any physical meaning. This problem makes the accurate detection of occluding edges more difficult. Another effect is due to the reflectance properties of the observed surface; if the surface is highly specular then no laser reflection can be observed. In that case the ERIM sensor returns a value of 255. This effect is most noticeable on man-made objects that contain a lot of polished metallic surfaces.

2.2 The video camera

The video camera is a standard color vidicon camera equipped with wide-angle lenses. The color images are 480 rows by 512 columns, each band is coded on eight bits. The wide-angle lens induces a significant geometric distortion, that is, the relation between a point in space and its projection on the image plane does not obey the laws of the standard perspective transformation. We alleviate this problem by first transforming the actual image into an "ideal" image: if (R, C) is the position in the real image, then the position (r, c) in the ideal image is given by:

$$r = f_r(R, C), c = f_c(R, C) \quad (5)$$

where f_r and f_c are third order polynomials. This correction is cheap since the right-hand side of (5) can be put in lookup tables. The actual computation of the polynomial is described in [9] The geometry of the ideal image obeys the laws of the perspective projection in that if $P = [x, y, z]'$ is a point in space, and (r, c) is its projection in the ideal image plane, then:

$$r = fx/z, c = fy/z \quad (6)$$

where f is the focal length. In the rest of the paper, row and column positions will always refer to the positions in the ideal image, so that perspective geometry is always assumed.

3 Merging range and video images

In order to merge range and color data, we have to store pixels from the two sensors into a common representation, the "*colored-range*" image [10]. Each pixel of a colored-range image contains a color value (*red, green, blue*) from the video camera as well as the position (x, y, z) of the measured point in space as derived from the geometry of the range sensor. By "*image*", we do not necessarily mean an array that stores those 6-dimensional pixels, but rather a set of functions that provide access to the range and color data at any point as described in Section 3.1.4. Our first task for building a colored-range image is to express the points in video and range image in a common reference frame, that is to solve the registration problem.

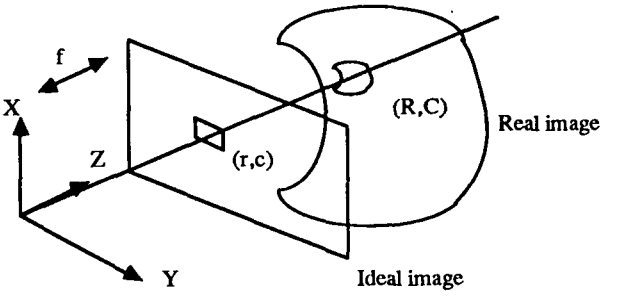

Figure 4: Geometry of the video camera

3.1 The registration problem

Range sensor and video cameras have different fields of view, orientations, and positions. In order to be able to merge data from both sensors, we first have to estimate their relative positions, this is known as the calibration, or registration problem (Figure 5). We approach the problem as a minimization problem in which pairs of pixels are selected in the range and video images. The pairs are selected so that each pair is the image of a single point in space as viewed from the two sensors. The problem is then to find the best calibration parameters given these pairs of points. The problem is further divided into two steps: we first use a simple linear least-squares approach to find a rough initial estimate of the parameters, and then apply a non-linear minimization algorithm to compute an optimal estimate of the parameters.

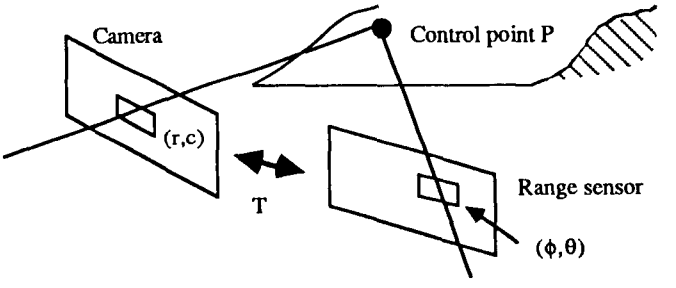

Figure 5: Geometry of the calibration problem

3.1.1 The calibration problem as a minimization problem

Let P_i be a point in space, with coordinates P_i^s with respect to the range sensor, and coordinates P_i^c with respect to the video camera. The relationship between the two coordinates is:

$$P_i^c = RP_i^s - T \quad (7)$$

where R is a rotation matrix, and T is a translation vector. R is a non-linear function of the orientation angles of the camera: pan (α), tilt (β), and rotation (γ). P_i^s can be computed from a pixel location in the range image. P_i^c is not completely known, it is related to the pixel position in the video image by the perspective transformation:

$$z_i^c r_i = f x_i^c \tag{8}$$

$$z_i^c c_i = f y_i^c \tag{9}$$

where f is the focal length. Substituting (7) into (8) and (9) we get:

$$R_x P_i^s r_i - T_x r_i - f R_x P_i^s + T_x' = 0 \tag{10}$$

$$R_x P_i^s c_i - T_x c_i - f R_y P_i^s + T_y' = 0 \tag{11}$$

where R_x, R_y, and R_z are the row vectors of the rotation matrix R, and $T_y' = fT_y$, $T_x' = fT_x$.

We are now ready to reduce the calibration problem to a least-squares minimization problem. Given n points P_i, we want to find the transformation (R, T) that minimizes the left-hand sides of equations (10) and (11). We first estimate T by a linear least-squares algorithm, and then compute the optimal estimate of all the parameters.

Assuming that we have an estimate of the orientation R, we first want to estimate the corresponding T. The initial value of R can be obtained by physical measurements using inclinometers. Under these conditions, the criterion to be minimized is:

$$\sum_{i=1}^{n} (A_i - T_z B_i - f C_i + T_x')^2 + (D_i - T_z E_i - f F_i + T_y')^2 \tag{12}$$

where $A_i = R_x P_i^s r_i$, $B_i = r_i$, $C_i = R_x P_i^s$, $D_i = R_z P_i^s c_i$, $E_i = c_i$, and $F_i = R_y P_i^s$ are known and T_z, T_x', T_y', f are the unknowns.

Equation (12) can be put in matrix form:

$$C = \|U - AV\|^2 + \|W - BV\|^2 \tag{13}$$

where $V = [T_x', T_y', T_z, f]^t$ is the vector of unknowns, and A, U, W, B are matrices that are functions of the known quantities only. The minimum for the criterion of Equation (13) is attained at the parameter vector:

$$V = (A^t A + B^t B)^{-1} (A^t U + B^t W) \tag{14}$$

Once we have computed the initial estimate of V, we have to compute a more accurate estimate of (R, T). Since R is a function of (α, β, γ), we can transform the criterion from equation (12) into the form:

$$C = \sum_{i=1}^{n} \|I_i - \Pi_i(S)\|^2 \tag{15}$$

where I_i is the 2-vector representing the pixel position in the video image, $I_i = [r_i, c_i]^t$, and S is the full vector of parameters, $S = [T_x', T_y', T_z, f, \alpha, \beta, \gamma]^t$. We cannot directly compute C_{min} since the functions Π_i are non-linear, instead we linearize C by using the first order approximation of Π_i [8] thus reducing the problem to a linear least-squares minimization that can be solved directly. The procedure is iterated until S cannot be improved any further.

3.1.2 Implementation and performance

The implementation of the calibration procedure follows the steps described above. Pairs of corresponding points are selected in a sequence of video and range images. We typically use twenty pairs of points carefully selected at interesting locations in the image (*e.g.* corners). An initial estimate of the camera orientation is $(0, \beta, 0)$, where β is physically measured using an inclinometer. The final estimate of S is usually obtained after less than ten iterations. This calibration procedure has to be applied only once, as long as the sensors are not displaced.

Once we have computed the calibration parameters, we can merge range and video images into a colored-range image. Instead of having one single fusion program, we implemented this as a library of fusion functions that can be divided in two categories:

1. Range → video: This set of functions takes a pixel or a set of pixels (r^s, c^s) in the range image and computes the location (r^c, c^c) in the video image. This is implemented by directly applying Equations (10) and (11).
2. Video → range: This set of functions takes a pixel or a set of pixels (r^c, c^c) in the video image and computes the location (r^s, c^s) in the range image. The computed location can be used in turn to compute the location of a intensity pixel in 3-D space by directly applying Equation (4). The algorithm for this second set of functions is more involved because a pixel in the video image corresponds to a line in space (Figure 4) so that Equations (10) and (11) cannot be applied directly. More precisely, a pixel (r^c, c^c) corresponds, after transformation by (R, T), to a curve C in the range image. C intersects the image at locations (r^s, c^s), the algorithm reports the location (r^s, c^s) that is the minimum among all the range image pixels that lie on C of the distance between (r^c, c^c) and the projection of (r^s, c^s) in the video image (using the first set of functions). The algorithm is summarized on Figure 6.

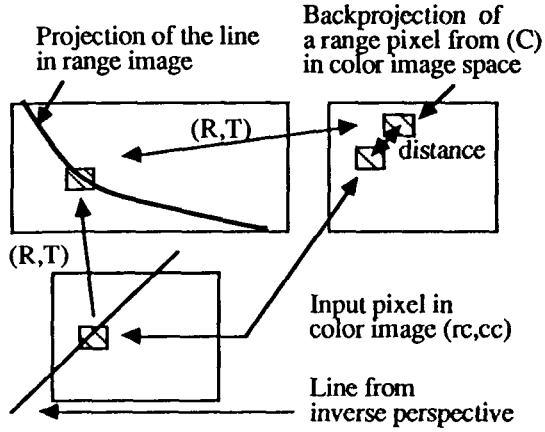

Figure 6: Geometry of the "video → range" transformation

Figure 7 shows the colored-range image of a scene of stairs and sidewalks, the image is obtained by mapping the intensity

values from the color image onto the range image. Figure 8 shows a perspective view of the colored-range image. In this example [10], we first compute the location of each range pixel (r^e, c^e) in the video image, and then assign the color value to the 64×256 colored-range image. The final display is obtained by rotating the range pixels, the coordinates of which are computed using Equation (4).

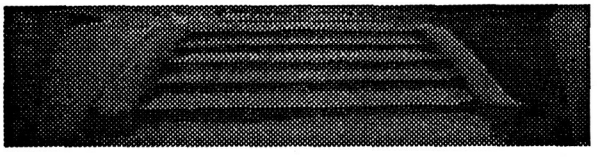

Figure 7: Colored-range image of stairs

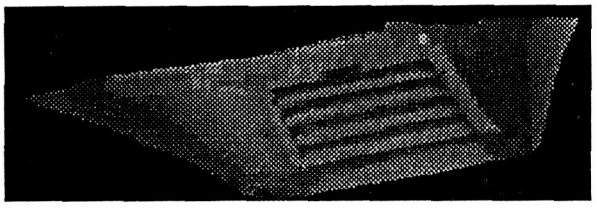

Figure 8: Perspective view of registered range and color images

3.2 Application to outdoor scene analysis

An example of the use of the fusion of range and video images is outdoor scene analysis [6,10] in which we want to identify the main components of an outdoor scene, such as trees, roads, grass, etc. The colored-range image concept makes the scene analysis problem easier by providing data pertinent to both geometric information (*e.g.* the shape of the trees) and physical information (*e.g.* the color of the road).

3.2.1 Feature extraction from a colored-range image

The features that we extract from a colored-range image must be related to two types of information: the shapes and the physical properties of the observed surfaces.

The geometric features are used to describe the shape of the objects in the scene. We propose to use two types of features: regions that correspond to smooth patches of surface, and edges that correspond either to transitions between regions, or to transitions between objects (occluding edges). Furthermore, we must be able to describe the features in a compact way. One common approach is to describe the regions as quadric patches, and the edges as sets of tri-dimensional line segments. More sophisticated descriptions are possible [4], such as bicubic patches or curvature descriptors. We use simpler descriptors since the range data is relatively low resolution, and we do not have the type of accurate geometric model that is suited for using higher order geometric descriptors. The descriptors attached to each geometric feature are:

- The parameters describing the shape of the surface patches. That is the parameters of the quadric surface that approximate each surface patch.
- The shape parameters of the surface patches such as center, area, and elongations.
- The 3-D polygonal description of the edges.
- The 3-D edge types: convex, concave, or occluding.

The surface patches are extracted by fitting a quadric of equation $X^tAX + B^tX + C = 0$ to the observed surfaces, where X is the Cartesian coordinate vector computed from a pixel in the range image. The fitting error,

$$E(A,B,C) = \sum_{X_i \in patch} [X_i^tAX_i + B^tX_i + C]^2 \qquad (16)$$

is used to control the growing of regions over the observed surfaces. The parameters A, B, C are computed by minimizing $E(A,B,C)$ as in [3].

The features related to physical properties are regions of homogeneous color in the video image, that is regions within which the color values vary smoothly. The choice of these features is motivated by the fact that an homogeneous region is presumably part of a single scene component, although the converse is not true as in the case of the shadows cast by an object on an homogeneous patch on the ground. The color homogeneity criterion we use is the distance $(X - m)^t\Sigma^{-1}(X - m)$ where m is the average mean value on the region, Σ is the covariance matrix of the color distribution over the region, and X is the color value of the current pixel in (*red, green, blue*) space. This is a standard approach to color image segmentation and pattern recognition. The descriptive parameters that are retained for each region are:

- The color statistics (m, Σ).
- The polygonal representation of the region border.
- Shape parameters such as center or moments.

The range and color features may overlap or disagree. For example, the shadow cast by an object on a flat patch of ground would divide one surface patch into two color regions. It is therefore necessary to have a cross-referencing mechanism between the two groups of features. This mechanism provides a two-way direct access to the geometric features that intersect color features. Extracting the relations between geometric and physical features is straightforward since all the features are registered in the colored-range image.

An additional piece of knowledge that is important for scene interpretation is the spatial relationships between features. For example, the fact that a vertical object is connected to a large flat plane through a concave edge may add evidence to the hypothesis that this object is a tree. As in this example, we use three types of relational data:

- The list of features connected to each geometric or color feature.
- The type of connection between two features (convex/concave/occluding) extracted from the range data.
- The length and strength of the connection. This last item is added to avoid situations in which two very close regions become accidentally connected along a small edge.

3.2.2 Scene interpretation from the colored-range image

Interpreting a scene requires the recognition of the main components of the scene such as trees or roads. Since we are dealing with natural scenes, we cannot use the type of geometric matching that is used in the context of industrial parts recognition [4]. For example, we cannot assume that a given object has specific quadric parameters. Instead, we have to rely on "fuzzier" evidence such as the verticality of some objects or the flatness of others. We therefore implemented the object models as sets of properties that translate into constraints on the surfaces, edges, and regions found in the image. For example, the description encodes four such properties:

- $P1$: The color of the trunk lies within a specific range $\Longrightarrow$ constraint on the statistics (m, Σ) of a color region.
- $P2$: The shape of the trunk is roughly cyclindrical $\Longrightarrow$ constraint on the distribution of the principal values of the matrix A of the quadric approximation.
- $P3$: The trunk is connected to a flat region by a concave edge $\Longrightarrow$ constraint on the neighbors of the surface, and the type of the connecting edge.
- $P4$: The tree has two parallel vertical occluding edges $\Longrightarrow$ constraint on the 3-D edges description.

Other objects such as roads or grass areas have similar descriptions. The properties P_{ij} of the known object models M_j are evaluated on all the features F_k extracted from the colored-range image. The result of the evaluation is a score S_{ijk} for each pair (P_{ij}, F_k). We cannot rely on individual scores since some may not be satisfied because of other objects, or because of segmentation problems. In the tree trunk example, one of the lateral occluding edges may itself be occluded by some other object, in which case the score for $P4$ would be low while the score for the other properties would still be high. In order to circumvent this problem, we first sort the possible interpretations M_j for a given feature F_k according to all the scores $(S_{ij})_i$. In doing this, we ensure that all the properties contribute to the final interpretation and that no interpretations are discarded at this stage while identifying the most plausible interpretations.

We have so far extracted plausible interpretations only for individual scene features F_k. The final stage in the scene interpretation is to find the interpretations (M_{j_k}, F_k) that are globally consistent. For example, property $P3$ for the tree implies a constraint on a neighboring region, namely that this has to be a flat ground region. Formally, a set of consistency constraints C_{mn} is associated with each pair of objects (M_m, M_n). The C_{mn} constraints are propagated through the individual interpretations (M_{j_k}, F_k) by using the connectivity information stored in the colored-range feature description. The propagation is simple considering the small number of features remaining at this stage.

The final result is a consistent set of interpretations of the scene features, and a grouping of the features into sets that correspond to the same object. The last result is a by-product of the consistency check and the use of connectivity data. Figure 9 shows the color and range images of a scene which contains a road, a couple of trees, and a garbage can. Figure 10 shows a display of the corresponding colored-range image in which the white pixels are the points in the range image that have been mapped into the video image. This set of points is actually sparse because of the difference in resolutions between the two sensors, and some interpolation was performed to produce the dense regions of Figure 10.

Only a portion of the image is registered due to the difference in field of view between the two sensors (60° for the camera versus 30° in the vertical direction for the range sensor). Figure 12 shows a portion of the image in which the edge points from the range image are projected on the color image. The edges are interpreted as the side edges of the tree and the connection between the ground and the tree. Figure 11 shows the final scene interpretation. The white dots are the main edges found in the range image. The power of the colored-range image approach is demonstrated by the way the road is extracted. The road in this image is separated into many pieces by strong shadows. Even though the shadows do not satisfy the color constraint on road region, they do perform well on the shape criterion (flatness), and on the consistency criteria (both with the other road regions, and with the trees). The shadows are therefore interpreted as road regions and merge with the other regions into one road region. This type of reasoning is in general difficult to apply when only video data is used unless one uses stronger models of the objects such as an explicit model of a shadowed road region. Using the colored-range image also makes the consistency propagation a much easier task than in purely color-based scene interpretation programs [11].

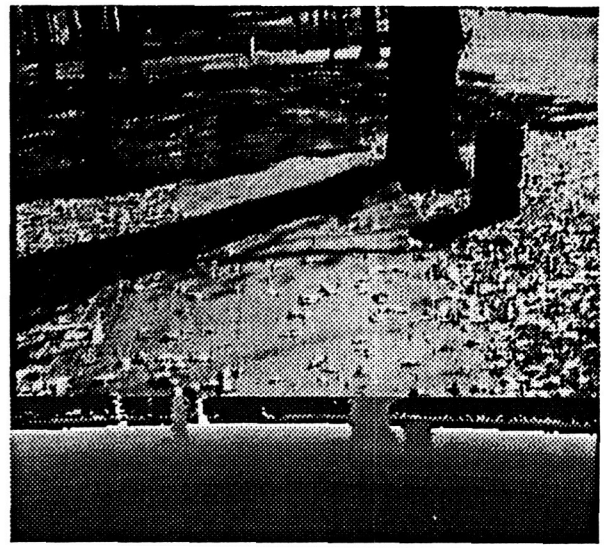

Figure 9: Color and range images of an outdoor scene

4 Merging range and active reflectance images

In the previous section we discussed the fusion of data from a video camera and a range sensor. We now discuss the fusion

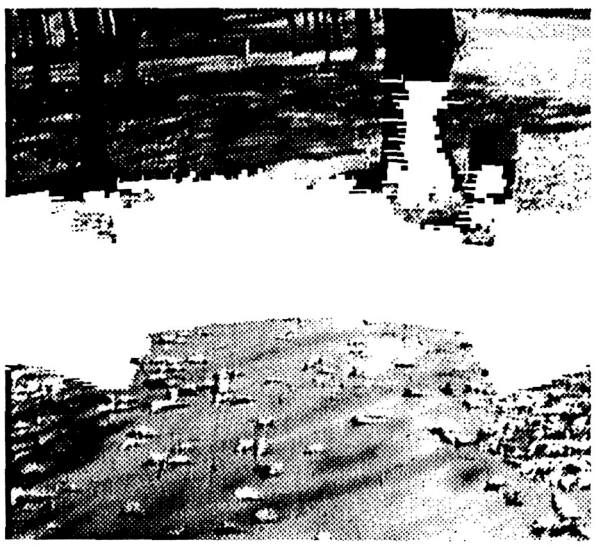

Figure 10: A view of the corresponding colored-range image

Figure 11: Final scene interpretation

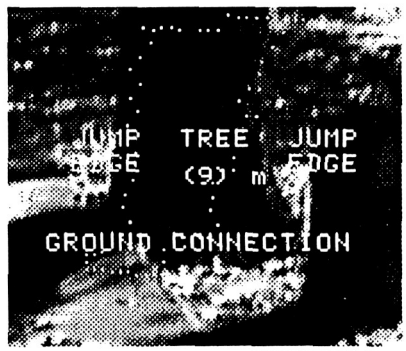

Figure 12: Edge features from the colored-range image

of two types of data from the ERIM sensor: range and active reflectance. This problem is somewhat simpler since the two images are already registered by the sensor itself. We will therefore focus our attention on the analysis of the active reflectance images, and the application of simultaneous range and reflectance processing to object recognition.

4.1 Correction of active reflectance images

A reflectance image from the ERIM sensor is an image of the energy reflected by the reflection of a laser beam. Unlike conventional intensity images, this data provide us information which is to a large extent independent of the environmental illumination. In particular, the reflectance images contain no shadows from outside illumination. The measured energy depends also on the shape of the surface, and its distance to the sensor. We correct the image so that the pixel values are functions of the material reflectance only. The measured energy, P_{return}, depends on the specific material reflectance, ρ, the range, D, and the angle of incidence, γ:

$$P_{return} = \frac{K\rho\cos\gamma}{D^2} \tag{17}$$

Due to the wide range of P_{return}, the value actually reported in the reflectance image is compressed by using a log transform. That is, the digitized value, P_{image} is of the form [14]:

$$P_{image} = A\log(\rho\cos\gamma) + B\log D \tag{18}$$

where A and B are constants that depend only on the characteristics of the laser, the circuitry used for the digitization, and the physical properties of the ambiant atmosphere. Since A and B cannot be computed directly, we use a calibration procedure in which a homogeneous flat region is selected in a training image, we then use the pixels in this region to estimate A and B by least-squares fitting Equ. (18) to the actual reflectance/range data. Given A and B, we correct subsequent images by:

$$P_{new-image} = (P_{image} - B\log D)/A \tag{19}$$

The value $P_{new-image}$ depends only on the material reflectance and the angle of incidence. This a sufficient approximation for our purposes since for smooth surfaces, such as smooth terrain, the

$cos\gamma$ factor does not vary widely. For efficiency purposes, the right-hand side of 19 is precomputed for all possible combinations (P_{image}, D) and stored in a lookup table. Figure 1 shows an example of ERIM image and Figure 13 shows the resulting corrected image.

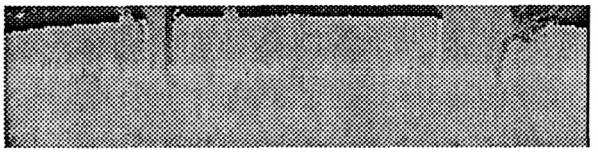

Figure 13: Corrected reflectance image

4.2 Application to 3-D feature extraction for object recognition

We now tackle the problem of fusing range and reflectance data for recognizing objects for landmark-based robot navigation [5]. The problem is different from the previous scene description problem in several respects. First of all, we assume that we have a geometric model of the landmark. Furthermore, we want to not only identify the object in the scene, but also to compute its position and attitude. It is critical to extract accurate geometric features from the images in order to relate the observed scene to the stored models. The fusion of range and reflectance data is used to improve the quality of the surface description extracted from the image data.

The 3-D features that are needed for object recognition are connected surface patches. Each patch corresponds to a smooth portion of the surface and is approximated by a parameterized surface. In addition to the parameters and the neighbors, each region has two uncertainty factors: σ_a, and σ_d. σ_a is the variance of the angle between the measured surface normal and the surface normal of the approximating surface at each point. σ_d is the variance of the distance between the measured points and the approximating surface. Those two attributes are used in the object recognition algorithm.

Several range image segmentation techniques have been proposed in previous works [4]. These techniques are based either on clustering in some parameter space, or region growing using the smoothness of the surface. We chose to combine both approaches into a single segmentation algorithm. The algorithm first attempts to find groups of points that belong to the same surface, and then uses these groups as seeds for region growing, so that each group is expanded into a smooth connected surface patch. The smoothness of a patch is evaluated by fitting a surface, plane or quadric, in the least-squares sense.

The strategy for expanding a region is to merge the best point at the boundary of each region at each step. This strategy guarantees a near optimal segmentation. It has, however, two major drawbacks: it may be computationally expensive, and it may lead to errors due to sensor errors on isolated points, such as mixed points. To alleviate these problems, we use a multi-resolution approach. We first apply the segmentation to a reduced image in which each pixel corresponds to a $n \times n$ window in the original image, n being the reduction factor. This first, low-resolution, step produces a conservative description of the image. The low-resolution regions are then expanded using the full-resolution image. No new regions are created at full resolution.

The region segmentation algorithm should produce a reliable description of a scene from a range image. The range measurements are corrupted by sensor noise (Section 2.1) which may produce gross errors in the segmentation. The first source of error is the sensor accuracy which degrades rapidly as the measurements are taken further away from the sensor. Due to the limited sensor accuracy, it is difficult to separate regions whose differences in orientation are of the order of the sensor noise. The second source of error is the presence of mixed points at the occluding edges of objects. This problem my lead to erroneous segmentation of the regions that border an object, as well as errors in the estimation of the parameters of those regions.

Merging informations from the reflectance images with the pure range image segmentation removes both types of segmentation errors. Specifically, we are interested in using the edges from the reflectance image. The edges correspond either to occluding edges or to edges on the surface of the object. In the first case, the reflectance edges indicate the possible locations of mixed points, which can therefore be removed from the range image prior to segmentation. In the second case, the reflectance edges may correspond to boundaries between surface patches that may not be distinguishable in the range image due to sensor noise. In the low-resolution segmentation step, pixels that correspond to a window that contains at least one edge pixel are removed so that mixed points at the occluding edges are removed. In the full-resolution step, regions are expanded so that they do not cross an edge. As a side effect, edge pixels are all part of the regions boundaries.

As an example, Figure 15 shows the edges extracted from the reflectance image of Figure 14. The edges were extracted by using a 10×10 Canny edge detector [2]. Figure 16 shows the corresponding low resolution segmentation for a reduction factor of $n = 2$. Each region is displayed with a different gray level. Figure 17 shows the final segmentation obtained at full resolution.

Figure 14: Range and reflectance images

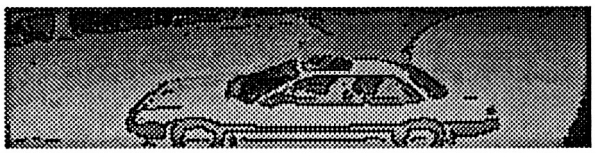

Figure 15: Edges from reflectance image.

Figure 16: Low-resolution segmentation ($n = 2$)

4.3 Object recognition from range and reflectance images

The 3-D features extracted from the range and reflectance images are matched against stored models in order to recognized known objects in the scene. The models are described by a set of surface patches and constraints between them. The constraints encapsulated geometrical properties of the object such as "these two patches are roughly orthogonal", for example. The constraints are implemented as numerical tests on the parameters of the regions extracted from the images. Instead of using strict constraints, such as "the normals v_1 and v_2 of those two regions are exactly orthogonal", we use intervals of confidence within which a given constraint is satisfied, such as "the angle between v_1 and v_2 must be within the interval $[\alpha_1, \alpha_2]$". Using intervals allows us to take into account the imprecision on the parameters of the features, and the fact that the stored model may not correspond exactly to the observed object.

The matching between scene and model features first generates hypothesis for each scene feature, and then explores the set of hypothesis in order to find matchings that satisfy the constraints stored in the model. The final product of the matching algorithm is a set of interpretations, that is a set of possible positions of the object in the scene. The interpretations are weighted by comparing the projection of the model onto the range image at the computed location, and the actual observed scene. The interpretation with the largest correlation is retained as the final interpretation (See [5] for a complete description of the recognition algorithm). Figures 18 and 19 show two examples of an object recognized in a range image (in this case a car). The top image is the reflectance image of the scene, the middle image shows the computed location of the car in the range image, and the bottom part of the Figures shows an overhead view of the scene with the object superimposed at the computed location. These results show that the combination of range and reflectance images provides the necessary features to accurately recognize and locate 3-D objects in outdoor scenes.

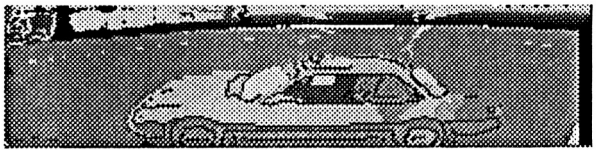

Figure 17: Final segmentation

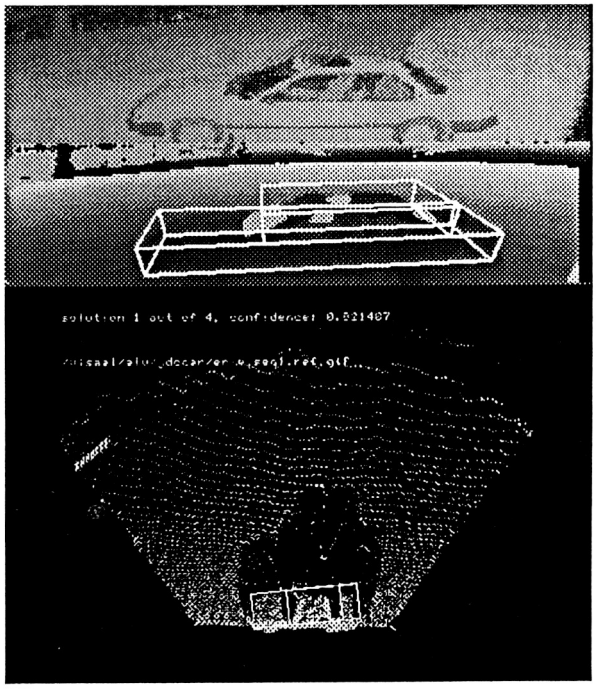

Figure 18: Result of object recognition from range and reflectance images

5 Conclusion

We have developed techniques for the fusion of data from multiple sensors. We have demonstrated the relevance of the resulting merged sensor data in the context of object recognition and scene interpretation for autonomous mobile robots. The experiments with real images showed conclusively that sensor data fusion provides useful additional information for scene interpretation. In order to represent the merged data, we have proposed the concept of a colored-range image in which pixels contain data from different sensors. One obstacle to building colored-range images is the geometric registration between sensors that may take images from different vantage points and with different fields of view. We have found that a simple sensor calibration scheme provides the parameters necessary to perform the registration. Even though we applied the data fusion approach to only three types of data, video, range, and active reflectance images, it is clear that the concept of colored-range image should be extended to other sensors such as sonars, active multiband reflectance, or multiple cameras. The sensor fusion techniques have been successfully integrated into the large autonomous mobile robot systems developed at CMU [10,12], and provide the

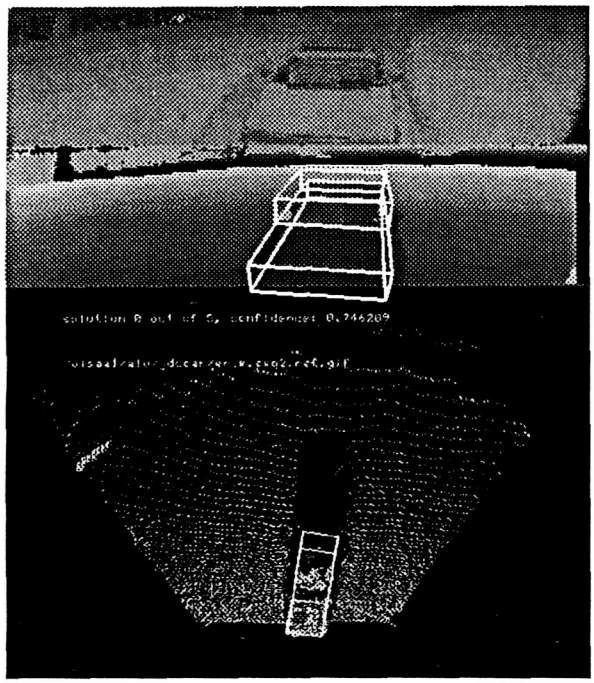

Figure 19: Result of object recognition from range and reflectance images

basis for the development of perception systems for Planetary exploration robots [13].

References

[1] P. Besl. *Range imaging sensors.* Technical Report GMR-6090, General Motors Research Lab, Warren, MI, March 1988.

[2] J. Canny. *Finding Edges and Lines in Images.* Master's thesis, Massachussets Institute of Technology, June 1983.

[3] O.D. Faugeras M. Hebert. The representation, recognition, and locating of 3-d objects. *International Journal of Robotics Research*, 5(3), 1986.

[4] P. J. Besl R. C. Jain. Three-dimensional object recognition. *ACM Comp. Surveys*, 17(1), march 1985.

[5] M. Hebert T. Kanade. 3-d vision for outdoor navigation by an autonomous vehicle. In *Proc. Image Understanding Workshop*, Cambridge, 1988.

[6] M. Hebert T. Kanade. First results on outdoor scene analysis. In *Proc. IEEE Robotics and Automation*, San Francisco, 1985.

[7] D. Zuk F. Pont R. Franklin V. Larrowe. *A system for autonomous land navigation.* Technical Report IR-85-540, Environmental Research Institute of Michigan, Ann Arbor MI, 1985.

[8] D.G. Lowe. Solving for the parameters of object models from image descriptions. In *ARPA Image Understanding Workshop*, 1980.

[9] H.P. Moravec. *Obstacle avoidance and navigation in the real world by a seeing robot rover.* Technical Report CMU-RI-TR-3, Carnegie-Mellon University, 1980.

[10] Y. Goto K. Matsuzaki I. Kweon T. Obatake. Cmu sidewalk navigation system: a blackboard-based outdoor navigation system using sensor fusion with colored-range images. In *Proc. First Joint Computer Conference*, Dallas, 1986.

[11] Y. Ohta. *Knowledge-based Interpretation of Outdoor Natural Color Scenes.* Pittman Publishing, Inc., 1984.

[12] C.E. Thorpe M. Hebert T. Kanade S.A. Shafer. Vision and navigation for the carnegie-mellon navlab. *PAMI*, 10(3), 1988.

[13] J. Bares W. Whittaker. Configuration of an autonomous robot for mars expolration. In *Proc. World Conference on Robotics*, Pittsburgh, PA, 1988.

[14] R. Watts F. Pont D. Zuk. *Characterization of the ERIM/ALV sensor - range and reflectance.* Technical Report , Environmental Research Institute of Michigan, Ann Arbor, MI, 1987.