An Algorithm for the Visual Perception of Laser Range Data for a Space-Based Robot

Scott A. Starks
Department of Electrical Engineering
University of Texas at El Paso
El Paso, TX 79968-0523

ABSTRACT

The Extravehicular Activity Retriever (EVAR) is a major robotic system being designed and developed at the NASA Johnson Space Center for future use in the Space Station Program. The EVAR will be an intelligent free-flying robot which will be endowed with the ability to locate, track, maneuver to, and retrieve objects which have become detached from the Space Station. Space-based robotic systems are presented with many technical problems not ordinarily found in robots designed for terrestrial settings. This report describes an approach developed for the visual perception of object surface information based upon laser range data in support of the grasping operation of the EVAR.

Introduction

It is the purpose of this paper to formulate an algorithm for the calculation of surface shape and structure from laser range data. This algorithm was developed with the intent of using it to assist in the closed loop control of the grasping operation performed by the Extravehicular Activity Retriever (EVAR). In order to perform the grasping operation a robot such as the EVAR must have knowledge of the structure of the surface of the object to be grasped. Using such information, the robot can plan the reach of its manipulator to the object and can preshape its hand in such a way so as to accommodate the object and ensure a stable grasp. The algorithm calculates information about the surface structure and is based upon concepts from differential geometry, namely the gaussian and mean curvatures of a surface. The approach to be presented is based in part upon previous works by other investigators [1], [2].

This report is based upon preliminary results from a study initiated this summer at the Johnson Space Center. In addition to the presentation of the algorithm for curvature-based surface calculation, this report also describes some of the key criteria relating to the determination of grasping strategies. It also presents a framework which allows a robot to store essential surface characterization and grasp configuration information in a computerized library. Under such a framework, a robot can make use of previously generated knowledge about objects and hand configurations to simplify the complexity of grasp planning.

Extravehicular Activity Retriever

The use of robots in hazardous environments such as space is rapidly expanding. Numerous studies and symposia have been conducted on this very important topic [3]. The EVAR is a voice-supervised, intelligent, free-flying robot which is currently being designed, developed, and demonstrated in a ground-based laboratory facility at the Johnson Space Center [4]. When operational, the EVAR will be used to retrieve objects such as construction materials, equipment, and tools which have been accidentally separated from the Space Station. The EVAR will be in a stand-by mode whenever astronauts perform extravehicular activity (EVA). If an astronaut becomes detached from the Station, the EVAR will be summoned to locate, fly to, grapple and return the astronaut to the Station. The EVAR is being designed so that it can accommodate an unconscious or uncooperative astronaut.

The EVAR consists of six integrated subsystems: the Manned Maneuvering Unit (MMU), two robotic arms with grippers, a video and tracking system, a 3D laser imaging system, an onboard computer system, and a data and control network. The MMU provides the EVAR with the ability to fly about in a manner similar to that of an astronaut during EVA. The two robotic arms are anthropomorphic in nature. One of the grippers is a dexterous hand while the other is a parallel jaw gripper. The EVAR video and tracking system employs two black and white cameras, a
tracking system, and a monitor to allow for backup supervisory control of EVAR operations by personnel at a remote location. One camera enables target acquisition and tracking, while the second can be used to assist during target grappling. The 3D laser imaging system is an Odetics 3D laser radar ranger. This subsystem, often called a 3D mapper, consists of a scan unit, an electronics unit, and a power supply. The 3D mapper provides direct digital range measurements and is capable of scanning a 60 degree horizontal by 60 degree vertical area with a 128X128 raster scan. The frame rate of the system is every .825 seconds. The onboard computer system consists of several transputers and other processors. The data and control network enables the routing of communication and control information among the various subsystems of the EVAR. A three phase ground demonstration program for the EVAR has been planned and scheduled.

Visual Perception of Range Imaging

In a broad sense, visual perception of range images can be viewed as the process of interpreting measurements made using any of a variety of range sensors. Because the success of automated devices depends critically on systems with the ability to sense and understand the environment, range image perception has received the attention of many investigators. Automatic systems have been designed to work in tightly structured environments such as assembly lines in factories. Unfortunately, much of the technology associated with systems of this type is not directly transferrable to the EVAR in that the EVAR must have the ability to cope with uncertainties in its environment.

The EVAR is equipped with a laser ranging device which is capable of determining a range image of the environment surrounding the robot. A range image is merely a large collection of distance measurements from a known reference coordinate system to surface points on objects within a scene. If the distance measurements in a range image are listed relative to three orthogonal coordinate axes, the range image is said to be in xyz form. If the distance measurements indicate range along 3D direction vectors indexed by two integers (i, j), the range image is said to be in rij form. Any image in rij form can be converted directly to xyz form, but the converse is not true [5]. The term image is used because any rij range image can be displayed on a video monitor, and it is identical in form to a digitized video image from a television camera. In addition, the term visual perception is often associated with the extraction of information from a range image for the same reason.

Range images can be a source of valuable information for an automated machine. The EVAR can use range image information to determine the location and orientation of objects in its environment. This information can be used to plan the EVAR's movements and actions. Surface fitting techniques make use of range data to produce a geometric description of objects. In this report, we are interested in determining the underlying surface structure of targets which the EVAR will be responsible for grasping. An approach to extracting the surface structure of a target is critical to the grasp operation because manipulators grasp surfaces. The following section presents an approach to surface fitting which is based upon curvature concepts taken from the field of differential geometry.

Surface Fitting

The surface fitting problem can be stated as follows: given three m x n arrays of x, y, and z coordinates, we would like to determine a surface that approximates the data in the least-squares sense and that is smooth. In a mathematical sense, smoothness is defined as the twice-differentiability of the surface at all points. Fitting a surface to an L x L window of data is equivalent to computing a surface fit for a roughly rectangular grid of data values. Intuitively, a roughly rectangular grid is one that has been obtained from a rectangular mesh that has been deformed to fit the surface. We may represent the surface in parametric form by the three equations

\[ x = f(s,t), \]
\[ y = g(s,t), \]
\[ z = h(s,t), \]

where s and t are the parameters, and the functions f, g, and h are tensor products of splines in tension. The fitting can be viewed as a mapping from 2D space to 3D space. The 2D space is characterized by the parameters s and t and the 3D space is the standard cartesian space.

Principal curvatures at a point on a surface indicate how fast the surface is pulling away from its tangent plane at that point. Principal curvatures can be computed by estimating partial derivative information for the surface. The principal curvatures will achieve a local maximum in the area surrounding a discontinuity in the surface, for example an edge in the object. By thresholding the values of the principal curvatures, we can declare the edge points in a range image.
When a plane passing through the normal to the surface at a point \( P \) is rotated about this normal, the radius of curvature changes and will be a maximum distance \( r_1 \) for a definite normal section \( s_1 \) and a minimum \( r_2 \) for another normal section \( s_2 \). The reciprocals \( k_1 = 1/r_1 \) and \( k_2 = 1/r_2 \) are called the principal curvatures; the directions of the tangents to \( s_1 \) and \( s_2 \) at \( P \) are called the principal directions of the surface at \( P \). The reciprocals \( k_1 = 1/r_1 \) and \( k_2 = 1/r_2 \) are called the principal curvatures; the directions of the tangents to \( s_1 \) and \( s_2 \) at \( P \) are called the principal directions of the surface at \( P \).

Gaussian curvature at point \( P \) is defined as the product of the two principal curvatures. It can be proven that the gaussian curvature depends only upon the coefficients of the first fundamental form of a surface and their derivatives. The coefficients of the second fundamental form share the previously stated property. The first and second fundamental coefficients also determine the surface uniquely up to a rigid body transformation. As a result, the gaussian curvature is said to be an intrinsic property of the surface. The principal curvatures at a point on a surface can be computed in terms of the parameters \( s \) and \( t \) using the following approach. Let \( X(s,t) \) represent the surface \( X(s,t) = [x(s,t), y(s,t), z(s,t)]^T \).

Here, bold face notation is used to denote a vector quantity. If we let partial differentiation be represented by subscripts, the differential element \( dX \) is a vector given by the relationship

\[
dX = X_s \, ds + X_t \, dt,
\]

where

\[
X_s = dX/ds = [x_s, y_s, z_s]^T,
\]

\[
X_t = dX/dt = [x_t, y_t, z_t]^T.
\]

If we take the scalar product of \( dX \) with itself we obtain a relationship which is known as the First Fundamental Form of the surface, \( I \), which is given below:

\[
I = dX \cdot dX = (X_s \, ds + X_t \, dt) \cdot (X_s \, ds + X_t \, dt) = E \, ds^2 + 2F \, ds \, dt + G \, dt^2,
\]

where the First Fundamental Coefficients (\( E \), \( F \), and \( G \)) are given by:

\[
E = |X_s|^2 = x_s^2 + y_s^2 + z_s^2,
\]

\[
F = X_s \cdot X_t = x_s x_t + y_s y_t + z_s z_t,
\]

\[
G = |X_t|^2 = x_t^2 + y_t^2 + z_t^2.
\]

The unit normal is then given by

\[
N = (X_s \times X_t) / |X_s \times X_t|.
\]

The differential of the unit normal is

\[
dN = N_s \, ds + N_t \, dt.
\]

If we take the negative of the scalar product of the differentials of the surface and the unit normal, we obtain the Second Fundamental Form for the Surface.

\[
II = -dX \cdot dN = - (X_s \, ds + X_t \, dt) \cdot (N_s \, ds + N_t \, dt) = L \, ds^2 + 2M \, ds \, dt + N \, dt^2,
\]

where the Second Fundamental Coefficients (\( L \), \( M \), and \( N \)) are given by:

\[
L = -X_s \cdot N_s,
\]

\[
M = -1/2 (X_s \cdot N_t + X_t \cdot N_s),
\]

\[
N = -X_t \cdot N_t.
\]

The gaussian curvature \( K \) at any point on the surface is defined as the product of the two principal curvatures \( k_1 \) and \( k_2 \) and can be expressed in terms of the First and Second Fundamental Coefficients as

\[
K = k_1 k_2 = (L N - M^2) / (E G - F^2).
\]

The mean curvature \( H \) is given by the average of the two principal curvatures. It may be expressed in terms of the First and Second Fundamental Coefficients as well,

\[
H = (L G - 2M F + N E) / (2(E G - F^2)).
\]

The principal curvatures are given by,

\[
k_1 = H - SQRT(H^2 - K),
\]

\[
k_2 = H + SQRT(H^2 - K).
\]
The above formulas can be used to compute the gaussian and mean curvatures. Let us assume that the gaussian and mean curvatures for a particular point referenced by the indices \((i, j)\) in a range image have been calculated and are stored in arrays \(K(i, j)\) and \(H(i, j)\) respectively. The points on a surface can then be classified according to the signs of these quantities: positive, negative, or zero. It turns out that there are only eight possible outcomes for surface characterization based upon the signs of the gaussian and mean curvatures. We introduce notation for two functions (modified signum function and the surface classification function). The modified signum function is a mapping of a scalar argument into one of three values \((1, 0, \text{or} -1)\). For positive \(e\), it is defined as:

\[
\text{sgn}_e(y) = \begin{cases} 
1 & \text{if } y > e \\
0 & \text{if } |y| < e \\
-1 & \text{if } y < -e.
\end{cases}
\]

We take advantage of the clarity afforded by use of the modified signum function to introduce the surface classification function

\[
T(i, j) = 1 + 3 \left(1 + \text{sgn}_{e_1} (H(i, j))\right) + \left(1 - \text{sgn}_{e_2}(K(i, j))\right).
\]

The positive scalars \(e_1\) and \(e_2\) are chosen to be close to zero. They enable the mapping to zero of a number within the limits of the numerical accuracy of the machine and the algorithm. Figure 1a gives the mapping of the values of the modified signum function applied to the gaussian and mean curvatures. Figure 1b gives a pictorial interpretation of the eight possible surface classifications afforded by the function \(T(i, j)\).

**Curvature-Based Surface Calculation Algorithm**

This section presents an algorithm to compute the object description in terms of jump boundaries, internal edges, and regions homogeneous in the sign of their gaussian and mean curvatures. The algorithm can be summarized as a sequence of the following steps:

- Divide the range image into overlapping windows.
- Detect jump boundaries and fit patches to windows of data not containing jump boundaries.
- Compute the principal curvatures and extract edge points.
- Classify each nonedge point in a patch as one of the eight possible surface classifications presented previously.
- Group all points of the same type in a patch and its neighboring patches into a region.

The details of each step in the algorithm are as follows. In the first step, the division of the input arrays of \((x, y, z)\) coordinates into \(L \times L\) windows should be overlapped to ensure that an internal edge is always contained in a patch.

In the second step, the relative degree of scatter exhibited by the data is reflected in the standard deviation of the euclidean distance between adjacent data points within the \(L \times L\) window. In the vicinity of a jump boundary of an object, the standard deviation of these euclidean distances will be relatively high. Jump boundaries can be detected by means of a threshold test. After the jump boundaries have been detected, smooth patches based upon 2D B-splines can be fitted to the data in the window not containing a jump discontinuity [6]. In the presence of a jump boundary, the window size can be modified to ensure that the window excludes the jump boundary. In order to detect fine detail in the scene it is important that the size of the objects be much larger than the size of the \(L \times L\) window.

In the third step of the algorithm, we attempt to detect all those points that belong to or fall on internal edges of the object. As was mentioned previously, in the vicinity of an edge, the values for the principal curvatures will be high and in fact achieve a local maximum. As a result, we first determine all points exhibiting principal curvatures above threshold. Due to the presence of noise in the data and an inappropriate choice of thresholds, clusters of edge points may appear in the vicinity of a true edge position. To eliminate these clusters of computed edge points, a suppression of nonmaxima is applied at every edge point. That is, we only declare an edge to be present at points where the curvature is a local maximum. Nonmaxima suppression is applied in a direction perpendicular to the edge directions which is the direction associated with the maximum absolute principal curvature.

The next step is to group object points into homogeneous regions. All points in a patch are classified into one of the types indicated in Figure 1b. All points of the same type are grouped together into a larger region. Their merging of points does not require explicit fitting of a new surface to this homogeneous region since the surfaces are chosen to be smooth. Each homogeneous region is assigned a label depicting its type. Internal edges may occur within regions. The extent of regions can be
delineated by jump boundaries, internal edges, and curvature edges, which we define as places where there is a change in curvature-based classification.

The object representation in terms of regions and curvature-based properties, as described above, have the following advantages:

- it is invariant under transformation of independent parameters of the surface.
- it is invariant to rigid body transformation and is therefore independent of viewpoint.

Summary

This paper has presented an algorithm for the calculation of curvature-based surface characteristics for objects using range data. At the present time, this algorithm is being implemented in the programming language C. Although the approach to surface description based upon curvature concepts taken from differential geometry appears promising, it would be presumptuous to leave the reader with the impression that it will provide all the information necessary for secure and stable grasping for the EVAR. It is essential to integrate this prototype software, when available, in a real-time systems level demonstration in order to fully evaluate its merits. Such a demonstration would bring to light any deficiencies which could then be addressed by future refinements.

Acknowledgements

The EVA Retriever is a joint project of the Crew and Thermal Systems Division, the Tracking and Communication Division, the Systems Development and Simulation Division, the Avionics Systems Division, and the Structures and Mechanisms Division, all of the Engineering Directorate at NASA Johnson Space Center. The author performed this work as a participant in the Summer Faculty Fellowship Program at JSC. He wishes to thank Dr. Jon D. Erickson and Dr. Kenneth R. Crouse of NASA JSC for their assistance and guidance in this project.

References


