Information Sciences

Automated Recognition of 3D Features in GPIR Images

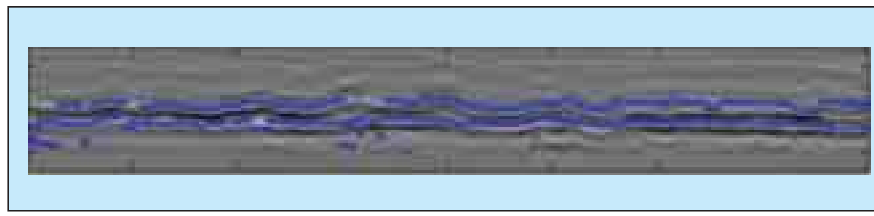
Enhanced images emphasizing features of interest are prepared for scrutiny by human analysts.

NASA's Jet Propulsion Laboratory, Pasadena, California

A method of automated recognition of three-dimensional (3D) features in images generated by ground-penetrating imaging radar (GPIR) is undergoing development. GPIR 3D images can be analyzed to detect and identify such subsurface features as pipes and other utility conduits. Until now, much of the analysis of GPIR images has been performed manually by expert operators who must visually identify and track each feature. The present method is intended to satisfy a need for more efficient and accurate analysis by means of algorithms that can automatically identify and track subsurface features, with minimal supervision by human operators.

In this method, data from multiple sources (for example, data on different features extracted by different algorithms) are fused together for identifying subsurface objects. The algorithms of this method can be classified in several different ways. In one classification, the algorithms fall into three classes: (1) image-processing algorithms, (2) feature-extraction algorithms, and (3) a multiaxis data-fusion/pattern-recognition algorithm that includes a combination of machine-learning, pattern-recognition, and object-linking algorithms.

The image-processing class includes preprocessing algorithms for reducing noise and enhancing target features for pattern recognition. The feature-extraction algorithms operate on preprocessed data to extract such specific features in images as two-dimensional (2D) slices of a pipe. Then the multiaxis data-fusion/pattern-recognition algorithm identifies, classifies, and reconstructs 3D objects from the extracted features. In this process, multiple 2D features extracted by use of different algorithms and representing views along different directions are used to identify and reconstruct 3D objects. In object linking, which is an essential part of this process, features identified in successive 2D slices and located within a threshold radius of identical features in adjacent slices are linked in a directed-graph data structure. Relative to past approaches, this multiaxis approach offers the advantages of more reliable detections, better discrimination of objects, and provision of redundant information, which can be helpful in filling gaps in feature recognition by one of the component algorithms.



Features Representing Two Pipes were generated by applying a feature-extraction algorithm to data from a GPIR scan of 110th Street in New York City. This synthetic image contains the detection marks overlaid on GPIR data from a mid-depth horizontal slice viewed from overhead. The gaps and undulations are minimized in subsequent processing by a multiaxis data-fusion/pattern-recognition algorithm.

The image-processing class also includes postprocessing algorithms that enhance identified features to prepare them for further scrutiny by human analysts (see figure). Enhancement of images as a postprocessing step is a significant departure from traditional practice, in which enhancement of images is a preprocessing step.

This work was done by Han Park, Timothy Stough, and Amir Fijany of Caltech for NASA's Jet Propulsion Laboratory.

The software used in this innovation is available for commercial licensing. Please contact Karina Edmonds of the California Institute of Technology at (626) 395-2322. Refer to NPO-40698.

Algorithm Plans Collision-Free Path for Robotic Manipulator

This algorithm is designed to make minimal demands upon computational resources.

NASA's Jet Propulsion Laboratory, Pasadena, California

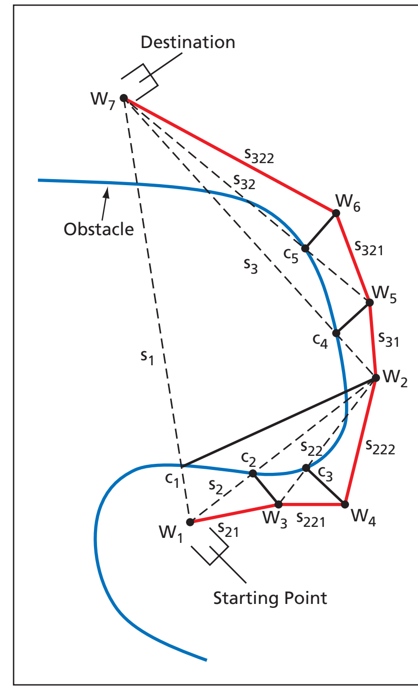
An algorithm has been developed to enable a computer aboard a robot to autonomously plan the path of the manipulator arm of the robot to avoid collisions between the arm and any obstacle, which could be another part of the robot or an external object in the vicinity of the robot. In simplified terms, the algorithm generates trial path segments and tests each segment for potential collisions in an iterative process that ends when a sequence of collision-free segments reaches from the starting point to the destination. The main advantage of this algorithm, relative to prior such algorithms, is computational efficiency: the algorithm is designed to make minimal demands upon the limited computational resources available aboard a robot.

This path-planning algorithm utilizes a modified version of the collision-detection method described in "Improved Collision-Detection Method for Robotic Manipulator" (NPO-30356), *NASA Tech Briefs*, Vol. 27, No. 3 (June 2003), page 72. The method involves utilization of mathematical models of the robot constructed prior to operation and similar models of external objects constructed automatically from sensory data acquired during operation. This method incorporates a previously developed method, known in the art as the method of oriented bounding boxes (OBBs), in which an object is represented approximately, for computational purposes, by a box that encloses its outer boundary. Because many parts of a robotic manipulator are cylindrical, the OBB method has been extended in this method to enable the approximate representation of cylindrical parts by use of octagonal or other multiple-OBB assemblies denoted oriented bounding prisms (OBPs).

A multiresolution OBB/OBP representation of the robot and its manipulator arm and a multiresolution OBB representation of external objects (including terrain) are constructed and used in a process in which collisions at successively finer resolutions are detected through computational detection of overlaps between the corresponding OBB and OBP models. For computational efficiency, the process is started at the coarsest resolution and stopped as soon as possible, preferably before reaching the finest resolution. At the coarsest resolution, there is a single OBB enclosing all relevant external objects and a single OBB enclosing the entire robot. At the next finer level of resolution, the coarsest-resolution OBB is divided into two OBBs, and so forth. If no collision is detected at the coarsest resolution, then there is no need for further computation to detect collisions. If a collision is detected at the coarsest resolution, then tests for collisions are performed at the next finer level of resolution. This process is continued to successively finer resolutions until either no more collisions are detected or the finest resolution is reached.

The path-planning algorithm operates on a representation of the robot arm and obstacles in a Cartesian coordinate system. The figure schematically depicts a simplified example of the geometric effects of the algorithm. In this example, the robot arm has been commanded to move from a starting point to a destination. The problem to be solved by the algorithm is to choose waypoints (W_1, W_2, ...) and straight-line path segments connecting the waypoints (including the starting point and destination as waypoints) so that there is no collision along any segment. The algorithm can be summarized as follows:

1. Generate a straight-line path (s_1) from the starting point (W_1) to the destination.
2. Using the collision-detection method described above, test for collisions along this path.
3. If there is a collision (denoted by collision point c_1), then by use of a geometry-based subalgorithm too complex to be described within the space available for this article, generate two new sub-paths (s_2 and s_3) that connect a new waypoint (W_2) with the ends of s_1.
4. Test each new sub-path for collisions.
5. If a collision is detected on either subpath (e.g., at collision point c_2 on s_2), then in the manner of step 3, generate new sub-sub paths (s_{21} and s_{22}) that connect new way point W_3 with W_1 and W_2.
6. Test for collisions and generate new path segments in the manner described above until the starting and destination points are connected by collision-free path segments. In this example, the result is a total of seven waypoints connected by six path segments.



A **Multi-Segment Path** is generated in an iterative process of generating candidate segments and testing them for collisions.

This work was done by Paul Backes and Antonio Diaz-Calderon of Caltech for NASA's Jet Propulsion Laboratory.

The software used in this innovation is available for commercial licensing. Please contact Karina Edmonds of the California Institute of Technology at (626) 395-2322. Refer to NPO-41697

Hybrid Automated Diagnosis of Discrete/Continuous Systems

Integration of complementary tools offers new approach to hybrid diagnosis.

NASA's Jet Propulsion Laboratory, Pasadena, California

A recently conceived method of automated diagnosis of a complex electromechanical system affords a complete set of capabilities for hybrid diagnosis in the case in which the state of the electromechanical system is characterized by both continuous and discrete values (as represented by analog and digital signals, respectively). The method is an integration of two complementary diagnostic systems: (1) beacon-based exception analysis for multimissions (BEAM), which is primarily useful in the continuous domain and easily performs diagnoses in the presence of transients; and (2) Livingstone, which is primarily useful in the discrete domain and is typically restricted to quasi-steady conditions. BEAM has been described in several prior *NASA Tech Briefs* articles: "Software for Autonomous Diagnosis of Complex Systems" (NPO-20803), Vol. 26, No. 3 (March 2002), page 33; "Beacon-Based Exception Analysis for Multimissions" (NPO-20827), Vol. 26, No. 9 (September 2002), page 32; "Wavelet-Based Real-Time Diagnosis of Complex Systems" (NPO-20830), Vol. 27, No. 1 (January 2003), page 67; and "Integrated Formulation of Beacon-Based Exception Analysis for Multimissions" (NPO-21126), Vol. 27, No. 3 (March 2003), page 74.

Briefly, BEAM is a complete data-analysis method, implemented in software, for real-time or off-line detection and characterization of faults. The basic premise of BEAM is to characterize a system from all available observations and train the characterization with respect to normal phases of operation. The observations are primarily continuous in nature. BEAM isolates anomalies by analyzing the deviations from nominal for each phase of operation. Livingstone is a