

The filter specifications were chosen to minimize the sizes of the filters, thereby placing the baseband higher in frequency than would otherwise be necessary. This is an acceptable trade-off inasmuch as (1) the consequent requisite digitizer bandwidth is still realizable by use of commercial devices and (2) the decimation performed by the digital filters eliminates excess bandwidth. The receiver band-pass filter (BPF) is placed in front of the LNA in order to limit radio-frequency interference. A programmable attenuator is included to provide adequate dynamic range in the event that the amplitude of the radar echo varies significantly. Care was taken to minimize cost by minimizing the number of parts and the number of different types of parts: in particular, the amplifiers and mixer used in the up-converter are also used in the down-converter.

The packaging of the L-band transceiver was designed in recognition that the different types of electronic devices used must be mounted and connected in different ways. The packaging approach was to place circuits that perform different functions in separate cavities in the module housing, coupling the DC signals through the walls by use of filtered connections only. This ap-

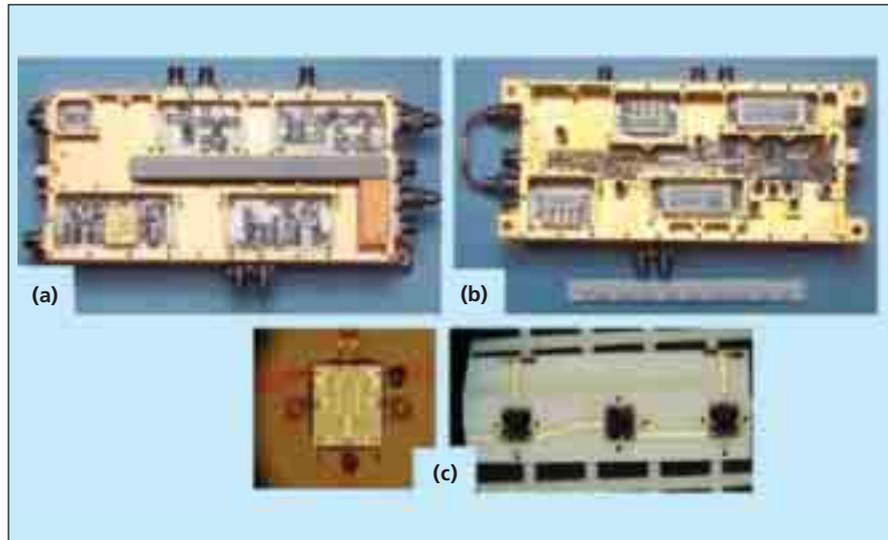


Figure 2. The Uncovered L-Band Transceiver Module shows the following: (a) top view (active components), (b) bottom view (filters and control), and (c) close-up views of packaging of individual switch die circuitry.

proach provides shielding from noise leakage. Thus, the down-converter (receiver) chain, the up-converter (transmitting) chain, and the control and power-supply circuitry are each located in separate cavities of the housing. The active RF components (e.g., amplifiers) are on one side of the module, while the passive RF components (e.g., attenuators and filters) and the control and

power circuits are on the opposite side. The RF functional blocks are further separated, according to frequency, onto individual substrates and into individual cavities.

*This work was done by Dalia McWatters, Douglas Price, and Wendy Edelstein of Caltech for NASA's Jet Propulsion Laboratory. For more information, contact iaoffice@jpl.nasa.gov. NPO-41278*

## Robotic Vision-Based Localization in an Urban Environment

A probability distribution of location is superimposed on an approximate map.

NASA's Jet Propulsion Laboratory, Pasadena, California

A system of electronic hardware and software, now undergoing development, automatically estimates the location of a robotic land vehicle in an urban environment using a somewhat imprecise map, which has been generated in advance from aerial imagery. This system does not utilize the Global Positioning System and does not include any odometry, inertial measurement units, or any other sensors except a stereoscopic pair of black-and-white digital video cameras mounted on the vehicle. Of course, the system also includes a computer running software that processes the video image data.

The software consists mostly of three components corresponding to the three major image-data-processing functions:

- *Visual Odometry*

This component automatically tracks point features in the imagery and computes the relative motion of the cam-

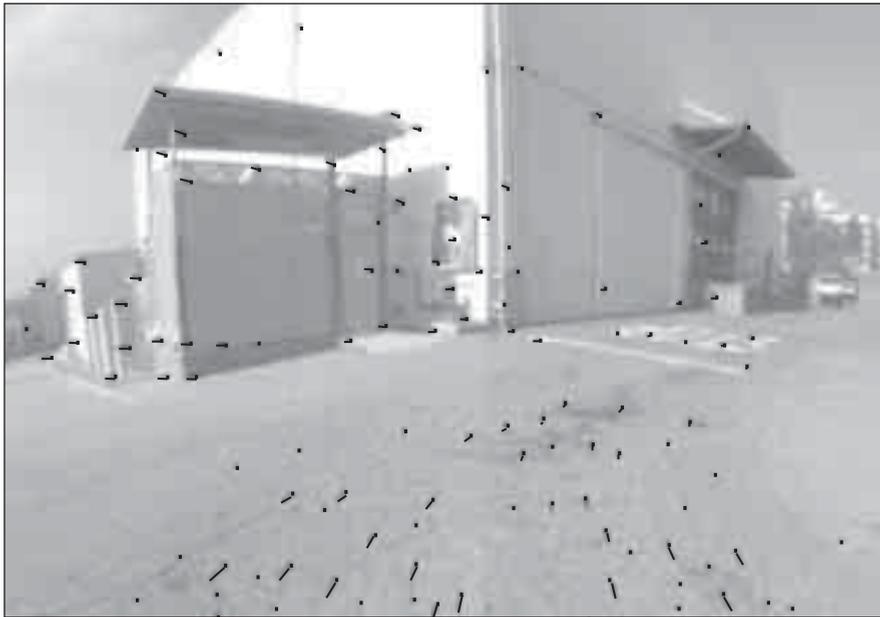
eras between sequential image frames. This component incorporates a modified version of a visual-odometry algorithm originally published in 1989. The algorithm selects point features, performs multiresolution area-correlation computations to match the features in stereoscopic images, tracks the features through the sequence of images, and uses the tracking results to estimate the six-degree-of-freedom motion of the camera between consecutive stereoscopic pairs of images (see figure).

- *Urban Feature Detection and Ranging*

Using the same data as those processed by the visual-odometry component, this component strives to determine the three-dimensional (3D) coordinates of vertical and horizontal lines that are likely to be parts of, or close to, the exterior surfaces of buildings. The basic sequence of processes performed by this component is the

following:

1. An edge-detection algorithm is applied, yielding a set of linked lists of edge pixels, a horizontal-gradient image, and a vertical-gradient image.
2. Straight-line segments of edges are extracted from the linked lists generated in step 1. Any straight-line segments longer than an arbitrary threshold (e.g., 30 pixels) are assumed to belong to buildings or other artificial objects.
3. A gradient-filter algorithm is used to test straight-line segments longer than the threshold to determine whether they represent edges of natural or artificial objects. In somewhat oversimplified terms, the test is based on the assumption that the gradient of image intensity varies little along a segment that represents the edge of an artificial object.



In this **Example Showing Features Used in Visual Odometry**, black lines superimposed on the image represent vectors proportional to displacement of features between sequential images.

4. A roof-line-detection algorithm identifies, as candidate roof lines, line segments (a) that exceed a threshold length and (b) above which there are no other such lines.
5. The 3D positions of line segments detected in the preceding steps are computed from either (a) ordinary stereoscopic imagery acquired simultaneously by the two cameras or (b) wide-baseline stereoscopic imagery synthesized from imagery acquired in two successive frames, using relative camera positions determined by visual odometry. The choice between (a) and (b) is made on the basis of which, given certain

parameters of the viewing geometry, is expected to enable a more accurate triangulation.

6. A heuristic pruning algorithm filters the remaining line segments: All lines that are not approximately vertical or horizontal are discarded, horizontal lines longer than 2 m are selected, vertical lines that extend above 2 m are selected, and sets of parallel lines are selected.

- **Particle-Filter-Based Localization**

The outputs of the visual-odometry and urban-feature-detection-and-ranging components are fed to this component, which implements a particle-filter-based localization algorithm. In

the theory of particle-filter-based robot localization, the key notion is that a particle filter produces an approximate probability density function for the position and heading of a robot by use of Monte Carlo techniques. This theory makes it possible to incorporate knowledge of measurement uncertainty in a rigorous manner. Because the open source particle-filter-based localization software (developed by Prof. Sebastian Thrun of Stanford University) used in a prototype of the system is based on a planar model, the input data fed to this component are preprocessed into simulated single-axis range data (in effect, simulated LIDAR range data) by projecting all 3D features onto a horizontal plane and sampling the field of view at small angular intervals ( $1^\circ$ ).

Notwithstanding the oversimplification inherent in this approach, success in localization has been achieved in initial experiments.

*This work was done by Michael McHenry, Yang Cheng, and Larry Matthies of Caltech for NASA's Jet Propulsion Laboratory.*

*In accordance with Public Law 96-517, the contractor has elected to retain title to this invention. Inquiries concerning rights for its commercial use should be addressed to:*

*Innovative Technology Assets Management  
JPL*

*Mail Stop 202-233  
4800 Oak Grove Drive  
Pasadena, CA 91109-8099  
(818) 354-2240*

*E-mail: iaoffice@jpl.nasa.gov*

*Refer to NPO-41881, volume and number of this NASA Tech Briefs issue, and the page number.*

## Programs for Testing an SSME-Monitoring System

*Marshall Space Flight Center, Alabama*

A suite of computer programs has been developed for special test equipment (STE) that is used in verification testing of the Health Management Computer Integrated Rack Assembly (HMC-IRA), a ground-based system of analog and digital electronic hardware and software for "flight-like" testing for development of components of an advanced health-management system for the space shuttle main engine (SSME). HMC-IRA units are designed to be integrated into a test facility wherein they enable additional engine monitoring during SSME

hot-fire tests. Running on a control processor that is part of the STE, the STE software enables the STE to simulate the SSME Controller, the SSME itself, and interfaces between the SSME and the HMC-IRA.

The STE software enables the STE to simulate the analog input and the data flow of an SSME test firing from start to finish. The STE software also provides user interfaces, error injection, data storage, and board-level test routines. Accompanying the STE software is a suite of post-processing programs that

convert stored data from the HMC-IRA and STE to readable textual and graphical formats, extract timing and statistical data, and provide for calibration of analog circuit cards.

*These programs were written by Andre Lang, Jimmie Cecil, Ralph Heusinger, Kathleen Freestone, Lisa Blue, DeLisa Wilkerson, Leigh Anne McMahon, Richard B. Hall, Kosta Varnavas, Keary Smith, and Donna Kaukler of Marshall Space Flight Center and James Hall of Sverdrup Technology, Inc. Further information is contained in a TSP (see page 1).  
MFS-32385-1.*