Software

3D Orbit Visualization for Earth-Observing Missions

This software visualizes orbit paths for the Orbiting Carbon Observatory (OCO), but was designed to be general and applicable to any Earth-observing mission. The software uses the Google Earth user interface to provide a visual mechanism to explore spacecraft orbit paths, ground footprint locations, and local cloud cover conditions. In addition, a drill-down capability allows for users to point and click on a particular observation frame to pop up ancillary information such as data product filenames and directory paths, latitude, longitude, time stamp, column-average dry air mole fraction of carbon dioxide, and solar zenith angle.

This software can be integrated with the ground data system for any Earth-observing mission to automatically generate daily orbit path data products in Google Earth KML format. These KML data products can be directly loaded into the Google Earth application for interactive 3D visualization of the orbit paths for each mission day. Each time the application runs, the daily orbit paths are encapsulated in a KML file for each mission day since the last time the application ran. Alternatively, the daily KML for a specified mission day may be generated.

The application automatically extracts the spacecraft position and ground footprint geometry as a function of time from a daily Level 1B data product created and archived by the mission's ground data system software. In addition, ancillary data, such as the column-averaged dry air mole fraction of carbon dioxide and solar zenith angle, are automatically extracted from a Level 2 mission data product. Zoom, pan, and rotate capability are provided through the standard Google Earth interface. Cloud cover is indicated with an image layer from the MODIS (Moderate Resolution Imaging Spectroradiometer) aboard the Aqua satellite, which is automatically retrieved from JPL's OnEarth Web service.

This work was done by Joseph C. Jacob, Lucian Plesea, Brian G. Chafin, and Barry H. Weiss of Caltech for NASA's Jet Propulsion Laboratory. Further information is contained in a TSP (see page 1).

This software is available for commercial licensing. Please contact Daniel Broderick of the California Institute of Technology at danielb@caltech.edu. Refer to NPO-47316.

MaROS: Web Visualization of Mars Orbiting and Landed Assets

Mars Relay operations currently involve several e-mails and phone calls between lander and orbiter teams in order to settle on an agreed time for performing a communication pass between the landed asset (i.e. rover or lander) and orbiter, then back to Earth. This new application aims to reduce this complexity by presenting a visualization of the overpass time ranges and elevation angle, as well as other information. The user is able to select a specific overflight opportunity to receive further information about that particular pass.

This software presents a unified view of the potential communication passes available between orbiting and landed assets on Mars. Each asset is presented to the user in a graphical view showing overpass opportunities, elevation angle, requested and acknowledged communication windows, forward and back latencies, warnings, conflicts, relative planetary times, ACE Schedules, and DSN information.

This software is unique in that it is the first of its kind to visually display the information regarding communication opportunities between landed and orbiting Mars assets. The software is written using ActionScript/FLEX, a Web language, meaning that this information may be accessed over the Internet from anywhere in the world.

This work was done by Michael N. Wallick, Daniel A. Allard, Roy E. Gladden, and Franklin H. Hy of Caltech for NASA's Jet Propulsion Laboratory. For more information, contact iaoffice@jpl.nasa.gov.

This software is available for commercial licensing. Please contact Daniel Broderick of the California Institute of Technology at danielb@caltech.edu. Refer to NPO-47413.

RAPID: Collaborative Commanding and Monitoring of Lunar Assets

RAPID (Robot Application Programming Interface Delegate) software utilizes highly robust technology to facilitate commanding and monitoring of lunar assets. RAPID provides the ability for inter-center communication, since these assets are developed in multiple NASA centers.

RAPID is targeted at the task of lunar operations; specifically, operations that deal with robotic assets, cranes, and astronaut spacesuits, often developed at different NASA centers. RAPID allows for a uniform way to command and monitor these assets. Commands can be issued to take images, and monitoring is done via telemetry data from the asset.

There are two unique features to RAPID: First, it allows any operator from any NASA center to control any NASA lunar asset, regardless of location. Second, by abstracting the native language for specific assets to a common set of messages, an operator may control and monitor any NASA lunar asset by being trained only on the use of RAPID, rather than the specific asset.

RAPID is easier to use and more powerful than its predecessor, the Astronaut Interface Device (AID). Utilizing the new robust middleware, DDS (Data Distribution System), developing in RAPID has increased significantly over the old middleware. The API is built upon the Java Eclipse Platform, which combined with DDS, provides platform-independent software architecture, simplifying development of RAPID components. As RAPID continues to evolve and new messages are being designed and implemented, operators for future lunar missions will have a rich environment for commanding and monitoring assets.

This work was done by Recaredo J. Torres, David S. Mittman, Mark W. Powell, Jeffrey S. Norris, Joseph C. Joswig, Thomas M. Crockett, Lucy Abramyan, Khawaja S. Shams, and Michael N. Wallick of Caltech; Mark Allan of Ames Research Center; and Robert Hirsh of Johnson Space Center for NASA's Jet Propulsion Laboratory. For more information, contact iaoffice@jpl.nasa.gov.

This software is available for commercial licensing. Please contact Daniel Broderick of the California Institute of Technology at danielb@caltech.edu. Refer to NPO-46332.

Image Segmentation, Registration, Compression, and Matching

A novel computational framework was developed of a 2D affine invariant matching exploiting a parameter space. Named

as affine invariant parameter space (AIPS), the technique can be applied to many image-processing and computer-vision problems, including image registration, template matching, and object tracking from image sequence.

The AIPS is formed by the parameters in an affine combination of a set of feature points in the image plane. In cases where the entire image can be assumed to have undergone a single affine transformation, the new AIPS match metric and matching framework becomes very effective (compared with the state-of-the-art methods at the time of this reporting). No knowledge about scaling or any other transformation parameters need to be known *a priori* to apply the AIPS framework.

An automated suite of software tools has been created to provide accurate image segmentation (for data cleaning) and high-quality 2D image and 3D surface registration (for fusing multi-resolution terrain, image, and map data). These tools are capable of supporting existing GIS toolkits already in the marketplace, and will also be usable in a stand-alone fashion. The toolkit applies novel algorithmic approaches for image segmentation, feature extraction, and registration of 2D imagery and 3D surface data, which supports first-pass, batched, fully automatic feature extraction (for segmentation), and registration.

A hierarchical and adaptive approach is taken for achieving automatic feature extraction, segmentation, and registration. Surface registration is the process of aligning two (or more) data sets to a common coordinate system, during which the transformation between their different coordinate systems is determined.

Also developed here are a novel, volumetric surface modeling and compression technique that provide both quality-guaranteed mesh surface approximations and compaction of the model sizes by efficiently coding the geometry and connectivity/topology components of the generated models. The highly efficient triangular mesh compression compacts the connectivity information at the rate of 1.5–4 bits per vertex (on average for triangle meshes), while reducing the 3D geometry by 40–50 percent.

Finally, taking into consideration the characteristics of 3D terrain data, and using the innovative, regularized binary decomposition mesh modeling, a multi-stage, pattern-drive modeling, and compression technique has been developed to provide an effective framework for compressing digital elevation model (DEM) surfaces, high-resolution aerial imagery, and other types of NASA data.

Σ Image Calibration

Calibrate_Image calibrates images obtained from focal plane arrays so that the output image more accurately represents the observed scene. The function takes as input a degraded image along with a flat field image and a dark frame image produced by the focal plane array and outputs a corrected image. The three most prominent sources of image degradation are corrected for: dark current accumulation, gain non-uniformity across the focal plane array, and hot and/or dead pixels in the array. In the corrected output image the dark current is subtracted, the gain variation is equalized, and values for hot and dead pixels are estimated, using bicubic interpolation techniques.

◎ Rapid ISS Power Availability Simulator

The ISS (International Space Station) Power Resource Officers (PROs) needed a tool to automate the calculation of thousands of ISS power availability simulations used to generate power constraint matrices. Each matrix contains 864 cells, and each cell represents a single power simulation that must be run. The tools available to the flight controllers were very operator intensive and not conducive to rapidly running the thousands of simulations necessary to generate the power constraint data.

SOLAR is a Java-based tool that leverages commercial-off-the-shelf software (Satellite Toolkit) and an existing in-house ISS EPS model (SPEED) to rapidly perform thousands of power availability simulations. SOLAR has a very modular architecture and consists of a series of plug-ins that are loosely coupled. The modular architecture of the software allows for the easy replacement of the ISS power system model simulator, re-use of the Satellite Toolkit integration code, and separation of the user interface from the core logic.

Satellite Toolkit (STK) is used to generate ISS eclipse and insulation times, solar beta angle, position of the solar arrays over time, and the amount of shadowing on the solar arrays, which is then provided to SPEED to calculate power generation forecasts.

The power planning turn-around time is reduced from three months to two weeks (83-percent decrease) using SOLAR, and the amount of PRO power planning support effort is reduced by an estimated 30 percent.