Titan through the time of Cassini: a database of VIMS observations

Paulo Penteado (penteado@astro.iag.usp.br)
Instituto de Astronomia, Geofísica e Ciências Atmosféricas, Universidade de São Paulo

Introduction

Since Cassini's arrival at Saturn, VIMS has recorded over 10^4 cubes, containing over 10^7 spectra. This still increasing amount of observations precludes direct inspection of all data, either to select the observations, or to identify the occurrence and time variation of specific spectral or spatial features. Additionally, many VIMS observations are taken as a large number of cubes with small spatial extent, which cannot be meaningfully visualized without assembling mosaics. This work presents titan_browse, a tool developed to deal with these difficulties. Titan_browse comprises both a database of observations, and a visualization tool to inspect them. The database contains every VIMS observation of Titan in the PDS archive, and provides a flexible query system, which can select individual cubes or spatial pixels based on arbitrary functions of the instrumental or photometric data. Once observations are selected, titan_browse can be used to directly inspect them, through mosaics in several map projections, or displaying images of selected bands, or spectra of selected spatial pixels. This allows users to interactively explore the data, to refine queries to obtain those most useful to the intended analysis. The selected cubes or spectra can them be directly exported from the database, either to an IDL session, or to files (one format being the original cubes, in ISIS format). The cubes used in the database were processed to contain more geometric information than either the original PDS files or those that are produced by the VIMS pipeline, including the coordinates of the edges of each spatial pixel (necessary for precise mosaics), and more information on the illumination angles (to aid in analyses of specular reflections). This version is a complete reimplementation of it's the previous, titan_browse, to overcome the previous coverage and performance limitations, and is the first to be made publicly available.

Data and calibration

The raw data used is publicly available at NASA's Planetary Data System (PDS) Imaging Node (http://pds-imaging.jpl.nasa.gov/index.html). At the time of this writing, 35 datasets have been released for VIMS. As the most recent observations are 9-12 months old, they currently cover the observations up to March 2009.

The cubes in the datasets are unprocessed, and before they can be incorporated into the database, they need radiometric and geometric calibration. For this purpose, a custom pipeline was developed, making use of the default pipeline software (included in the PDS datasets), and the data made available by NASA's Navigation and Ancillary Information Facility (NAIF, http://naif.jpl.nasa.gov/naif/index.html), incorporated through the use of NAIF's SPICE library. The pipeline developed produces cubes with more geometric data than the standard pipeline, including:

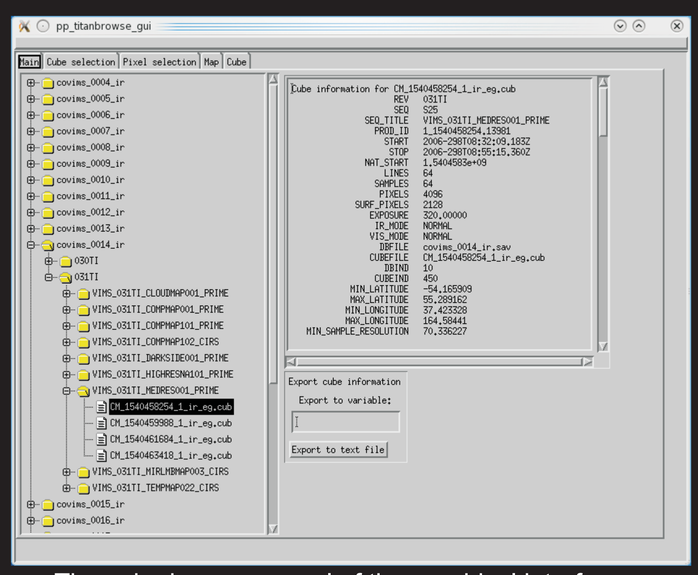
- The geometric data is calculated for the center and each corner of every pixel, necessary to determine the projection of every pixel on Titan. This is of particular relevance for cubes taken at low spatial resolution, cubes with a small length in one dimension (even 1 pixel wide cubes), and cubes with large pointing changes between pixels.
- The geometric data is calculated even for pixels that do not intercept the surface, in which case they refer to the point in the line of sight nearest to the surface. This is essential for studies of Titan's atmosphere, with spectra taken over Titan's limb, which are unique as they provide direct vertical resolution of the atmosphere.
- More illumination data is calculated, including the location of the specular reflection point, and its distance from each pixel. This data is of particular relevance for detection of liquid surfaces, and studies of the surface's scattering function. These data are calculated for each pixel, to account for cubes where a large geometry change occurs between the time each pixel was recorded.

The geometric data is incorporated into the cubes into their header (for data which are constant over the cube), and into backplanes (for data which vary for each pixel). The current implementation produces these 52 backplanes:

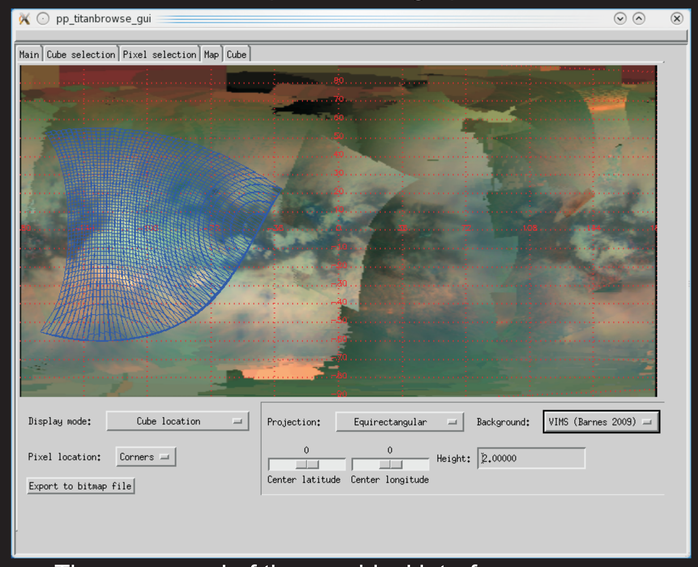
LATITUDE	NaN	Latitude of the center of pixel line of sight, if it intercepts the surface (from ISIS)
LONGITUDE	NaN	Longitude of the center of pixel line of sight, if it intercepts the surface (from ISIS)
SAMPLE_RESOLUTION	NaN	Sample resolution of the center of pixel line of sight, if it intercepts the surface (from ISIS)
LINE_RESOLUTION	NaN	Line resolution of the center of pixel line of sight, if it intercepts the surface (from ISIS)
PHASE_ANGLE	NaN	Phase angle of the center of pixel line of sight, if it intercepts the surface (from ISIS)
INCIDENCE_ANGLE	NaN	Incidence (solar) angle of the center of pixel line of sight, if it intercepts the surface (from ISIS)
EMISSION_ANGLE	NaN	Emission (observer) angle of the center of pixel line of sight, if it intercepts the surface (from ISIS)
NORTH_AZIMUTH	NaN	North azimuth of the center of pixel line of sight, if it intercepts the surface (from ISIS)

For the following backplanes, the data refer to the point nearest to the surface in the line of sight, of the pixel's center (_0), top-left corner (_1), top-right corner (_2), bottom-right corner(_3), and bottom-left corner (_4):

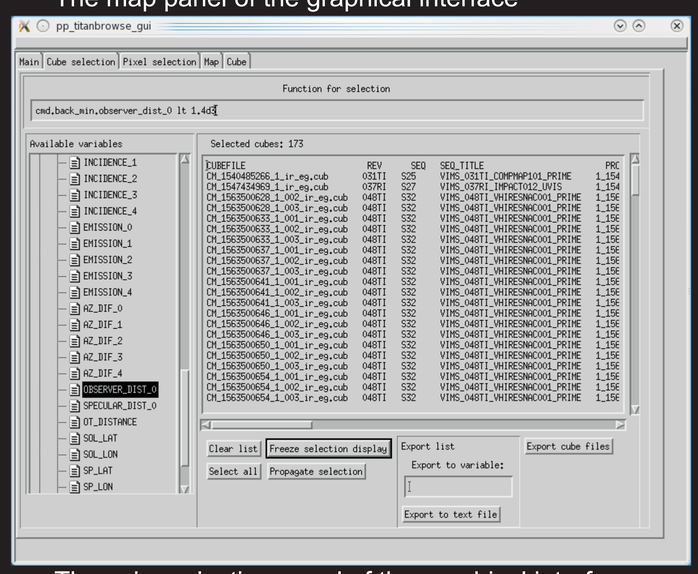
LAT_0	-38.839169	Latitude
LON_0	160.40419	Longitude
ALT_0	102.46656	Altitude to the surface (0 means the pixel intercepts the surface)
PHASE_0	97.228409	Phase angle
INCIDENCE_0	52.578320	Incidence angle
EMISSION_0	89.999977	Emission angle
AZ_DIF_0	99.115982	Azimuth difference between the Sun and the observer
OBSERVER_DIST_0	26201.770	Slant distance from the point to the observer
SPECULAR_DIST_0	93.634842	Angular distance from the point to the specular reflection point
OT_DISTANCE	6338.215	Distance between the target and observer
SOL_LAT	-3.9179378	Subsolar latitude
SOL_LON	-156.21873	Subsolar longitude
SP_LAT	19.903276	Specular point latitude
SP_LON	-197.25456	Specular point longitude
OBS_LAT	32.293877	Subobserver latitude
OBS_LON	111.95390	Subobserver longitude



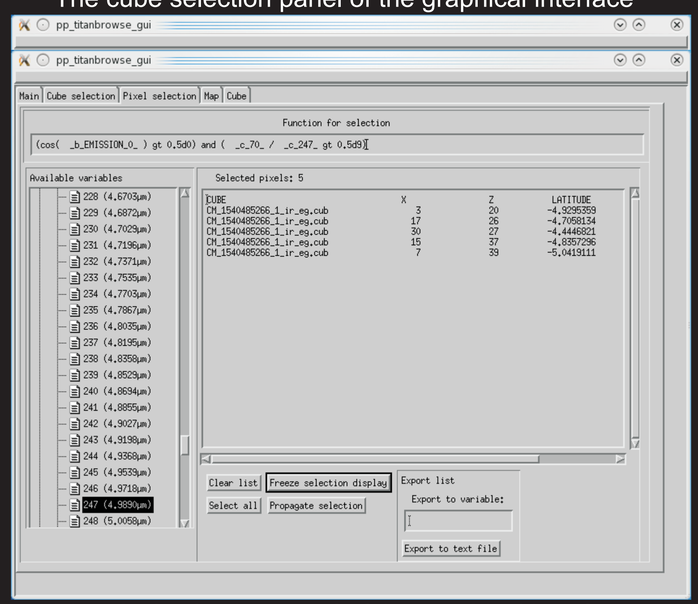
The cube browser panel of the graphical interface



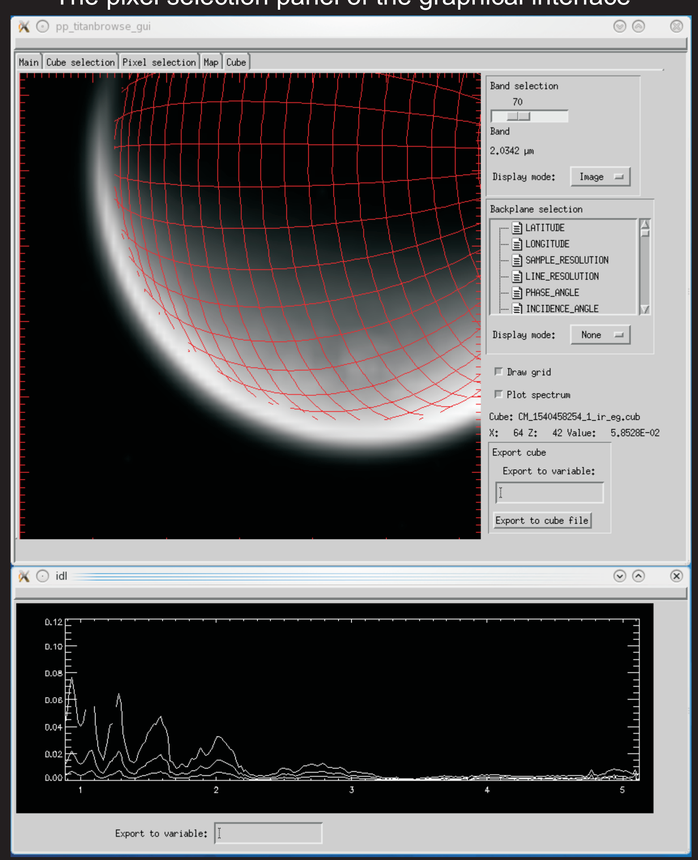
The map panel of the graphical interface



The cube selection panel of the graphical interface



The pixel selection panel of the graphical interface



The visualization panel of the graphical interface

Database implementation and functionality

The database was implemented in IDL, instead of a dedicated database system, for several reasons:

- The data structures and the usual criteria for its access and selection are very array-oriented, and IDL provides ample support for efficient array processing.
- Dynamic interpretation of complex functions to be used for data selection and visualization, making use of IDL's standard library and user-defined functions that may be the same used in the data analysis.
- Integration of the database use with an environment commonly used for the subsequent data analysis. For instance, selected metadata, spectra and entire cubes retrieved from the database can be immediately used for analysis, with no need for separate data export/import through files.
- Integration between database access and visualization of the results, making use of the variety of visualization tools provided by IDL, including map projection functionality.
- Platform independency.
- Ease of development and maintenance.

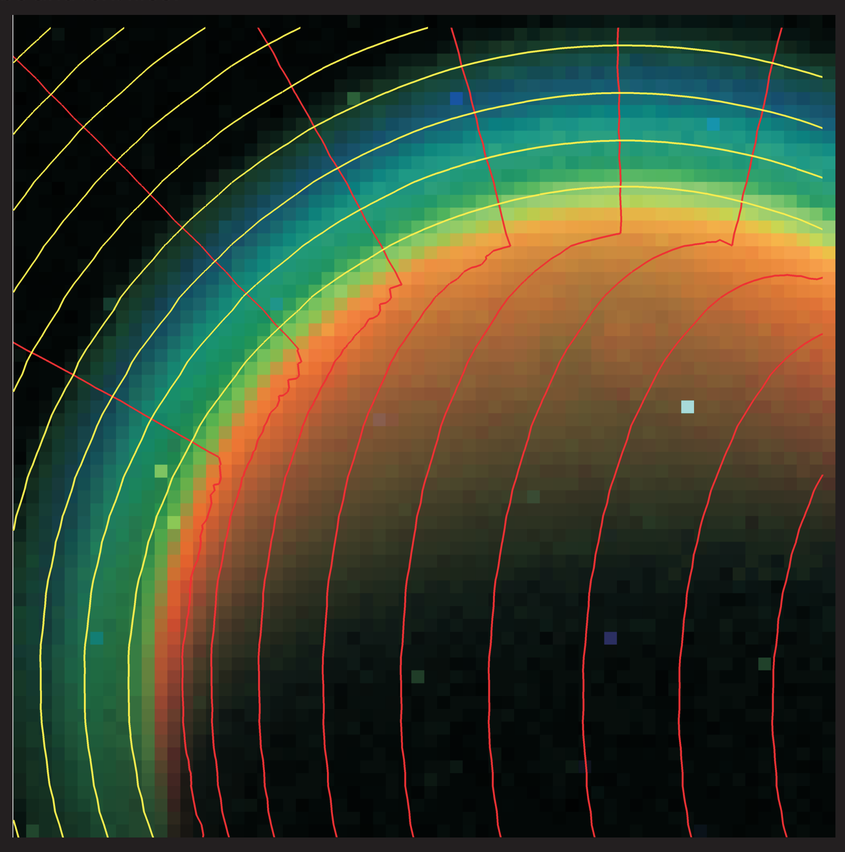
The database works on the principle of making selections of entire cubes or individual spectra (spatial pixels). Starting from a selection that comprises the entire domain, arbitrary user functions of the cube metadata (for cube selection) or pixel data (for pixel selection) are used to filter the selections, reducing them to the data with the selected properties. The data that can be used in functions for pixel selection include all core bands, and all (currently 52) backplanes. The data that can be used for cube selection includes the ranges of the core and backplane values, and (currently) the following metadata:

REV	093TI	REV identifier
SEQ	S45	Sequence identifier
SEQ_TITLE	VIMS_093TI_GLOBMAP001_CIRS	Sequence title identifier
PROD_ID	1_1605808125.14960	Cube identifier
START	2008-324T17:09:20.747Z	Cube start time
STOP	2008-324T17:20:43.821Z	Cube stop time
NAT_START	1.6058081e+09	Cube native clock start time
LINES	24	Number of lines in the cube
SAMPLES	64	Number of samples in the cube
PIXELS	1536	Number of spatial pixels in the cube (lines*samples)
SURF_PIXELS	1201	Number of pixels whose center line of sight intercept the surface
EXPOSURE	420.00000	Exposure time
IR_MODE	NORMAL	Resolution mode for the IR channel
VIS_MODE	NORMAL	Resolution mode for the VIS channel
DBFILE	covims_0032_ir.sav	Database file this comes from
CUBEFILE	CM_1605808125_1_ir_eg.cub	Cube file
DBIND	2	Index that identifies the database file
CUBEIND	1	Index that identifies the cube in the database file

Internally, the database objects contain all the processed cubes, stored as a collection of objects of a class called pp_editablecube. This class provides a convenient way to read, edit, write and store ISIS cubes, allowing access (and modifications) to the entire contents of a cube (header, core bands, backplanes, sideplanes and bottomplanes) through an interface more complete and easier to use than any other routines that could be found. Cubes can be kept stored directly by these objects, and ISIS (.cub) files can be recreated with a simple call to one of their methods. No external libraries (such as ISIS) are needed to use the database or the cube objects.
For each PDS dataset, the database keeps its data in two files: one contains cube metadata, and all cubes as pp_editablecubes, thus in a "cube-major" order. For more efficient access during pixel selections, the second file used by the database contains the cube core and backplanes in a "band-major" order, so that only the needed bands get read from disk. This allows the database to handle every VIMS titan cube recorded, and conveniently provide query and access to them, without the need to have the original cube files.

Graphical interface and visualization

Though the entire database was implement to be accessed programatically, to aid in selection and visualization, tiranbrowse includes a graphical interface. In addition to the functionality provided by the API, this GUI provides visualization of individual cubes and spectra, and geographical mapping of the selected pixels, to aid in interactive exploration of the data. As from the API, the selected data and metadata can be directly exported to variables to be used in the IDL session, or to cube and text files.



An example of 3 bands of a cube mapped into RGB space, with contours of constant latitude (red) and altitude (yellow) for the pixel centers, to illustrate the geometric data included with the cube, which extends to the pixels that do not intercept the surface.

This work is supported by the São Paulo state government through FAPESP