Data Introduction

Stardust: Navigation Camera (NavCam) Images

Instrument Operations

NavCam is an Optical CCD 1024^2 pixel camera developed using a Voyager Wide Angle Optical Assembly and has a focal length of 200 mm with f/3.5. NavCam had a filter wheel with 8 filters however, one year after launch the filter wheel got stuck on the OpNav filter which transmits light from ~400 - ~900 nm. NavCam uses a Petzval lens system which, combined with its wide bandpass, causes images to suffer from some chromatic aberration. As a result the intrinsic point spread function is ~2.3 pixels. Additionally, the anti-reflection coating for the camera lens was developed in the 1970's and performs poorly causing images to have broad shallow skirts of scattered light.

NavCam's purpose was to navigate Stardust during its approach to a comet and regulate distance during the flyby. NavCam utilizes a scan mirror and periscope to assist in this. The scan mirror allows the stationary optics to keep the comet in its view during the flyby by rotating at the proper rate. The periscope is an optical assembly that allows the scan mirror to see over the Whipple shield when viewing in the forward direction where the shield would otherwise obstruct the mirror. NavCam data can be used to obtain science.

Reading the Data

The data is primarily stored as FITS files with detached PDS labels; one label and one FITS file for each exposure. Version 1.0 of the raw data is an exception and is stored as stored as IMG rather than FITS files. Filenames are chronologically ordered. A typical filename is something like 'n2057we02r.fit' or 'n0494c202.fit'. The 'n' stands for Navigation Camera. This is followed by a 4 digit image sequence number, in these cases '2057' or '0494'. The next two characters tell us which mission phase the data was taken in. The 'we' stands for the Wild 2 encounter phase and the 'c2' stands for the Cruise 2 phase. This is followed by 2 digits that tell us the version number of the data such as the '02' for version 2 data. If the data has been radiometrically calibrated an 'r' is appended to the end of the filename, otherwise the filename ends with the version number. If an image has been radiometrically calibrated it will give radiance in units of W/cm^2/nm/sr.

Calibrated encounter images are 1022 pixels across in one direction rather than the full 1024 pixels. This is because the first and last columns in that direction are uncalibratable and removed during calibration. In version 2.0 of the raw and calibrated images LINE_DISPLAY_DIRECTION and SAMPLE_DISPLAY_DIRECTION in the label give the orientation as seen on the sky.

In the case of navigation data usually not the entire frame but only the values of certain sections of the CCD were downlinked. Non-downlinked pixels are filled with hex null values (i.e. Zeroes). Non-null regions are called pixel windows and their location is given in the OBJECT DESCRIPTION of the corresponding label. A navigation image is the same size as an encounter image except the pixel values are zero outside the pixel windows.

The design of the shutter causes an offset between the commanded exposure time and the true exposure time. The exposure can flip between too long and too short. This has been corrected in version 2.0 of the calibrated and raw data sets so that the EXPOSURE_DURATION keyword contains the true exposure time.

If your favorite analysis environment is IDL, you can use the package readpds.pro, which is available at PDS-SBN (http://pdssbn.astro.umd.edu/tools/). If you type “data = readpds(<filename.lbl>)”, it will read the data, including all extensions, into an IDL structure containing all the parts of the data product. To see the various pieces, type “help, /struct, data” and it will list the pieces of the highest level of the structure. Some of those will be themselves be structures and you can type, “help, /struct, data.piece1” to find out what is in the sub-structure piece1. If your favorite environment is ISIS, there is a routine pds2isis, although we have not exercised this routine.

PDS does not explicitly support FITS, but if your favorite analysis environment is based exclusively on FITS, you can read the FITS file directly (with extension .fit), ignoring the PDS label (the file with .lbl extension), but you need to be aware that PDS does not validate or officially support the FITS standard. We strongly encourage the use of the freely available routine fv (http://heasarc.gsfc.nasa.gov/ftools/fv/) to read the entire file and determine which extensions are of interest.