
planetaryimage Documentation

Release 0.4.1

PlanetaryPy Developers

May 12, 2016

1	planetaryimage - Open PDS and Isis CubeFiles in Python	3
1.1	Features	3
1.2	Quickstart	3
2	Installation	5
3	Usage	7
3.1	Quick Explanation	7
3.2	Examples	7
4	PlanetaryImage	13
5	PDS3Image	15
6	CubeFile	17
7	Supported Planetary Image Types	19
7.1	General Information	20
7.2	PDS List	20
8	Contributing	37
8.1	Types of Contributions	37
8.2	Get Started!	38
8.3	Pull Request Guidelines	38
8.4	Tips	39
9	Credits	41
9.1	Development Lead	41
9.2	Contributors	41
10	History	43
11	0.4.1 (2016-03-26)	45
12	0.4.0 (2016-03-05)	47
13	0.3.0 (2015-09-29)	49
14	0.2.0 (2015-06-17)	51

15	0.1.0 (2015-06-03)	53
16	Indices and tables	55
	Python Module Index	57

Contents:

planetaryimage - Open PDS and Isis CubeFiles in Python

NOTE This is Alpha quality software that is being actively developed, use at your own risk.

Planetary image parser

- Free software: BSD license
- Documentation: <https://planetaryimage.readthedocs.org>.

1.1 Features

- Reads in PDS Images as NumPy arrays.
 - Supports GZIP and BZ2 compressed PDS Images.
 - Supports writing out PDS3 images.
- Reads in Isis Cube Files as NumPy arrays.

Check out a few simple examples of [opening and viewing PDS and Isis CubeFiles in an IPython notebook](#).

1.2 Quickstart

The example below will walk you through setting up a Python virtual environment and installing the necessary software as well as a few handy extras. It then downloads a sample Pancam PDS image, opens and displays that image in your web browser in an [IPython Notebook](#). The example assumes you have Python, virtualenv, and pip installed on your system. If you don't, don't know what this means or aren't thrilled by the opportunity to learn what this means, this software may be a little too immature for you to use at this point.

Create and activate a virtual environment:

```
virtualenv venv
source venv/bin/activate
```

Upgrade pip, then pip install the package and IPython notebook and matplotlib to help display the image:

```
pip install -U pip
pip install planetaryimage matplotlib ipython[notebook]
```

This quick example will show how to open and display a Pancam PDS image using this module. First, grab a sample image:

```
wget http://pds-imaging.jpl.nasa.gov/data/mer/opportunity/mer1po_0xxx/data/sol2840/edr/1p380322615effbr43p244311m1.img
```

Now run python in an IPython Notebook (a browser window should pop up after entering the following command):

```
$ ipython notebook
```

Create a new notebook in your web browser and then paste the following code into a cell and execute it by pressing Shift+ENTER. This will load and display the image:

```
%matplotlib inline
import matplotlib.pyplot as plt
from planetaryimage import PDS3Image
image = PDS3Image.open('1p380322615effbr43p244311m1.img')
plt.imshow(image.image, cmap='gray')
```

See [Usage](#) for full documentation on how to use planetaryimage.

Installation

At the command line:

```
$ easy_install planetaryimage
```

Or, if you have virtualenvwrapper installed:

```
$ mkvirtualenv planetaryimage  
$ pip install planetaryimage
```

Usage

3.1 Quick Explanation

The example below shows how to use planetaryimage's PDS3Image class to open a PDS image, inspect it's label and display the image data:

```
>>> from planetaryimage import PDS3Image
>>> import matplotlib.pyplot as plt
>>> testfile = 'tests/mission_data/2p129641989eth0361p2600r8m1.img'
>>> image = PDS3Image.open(testfile)
>>> print(image.record_bytes)           # access attribute
128
>>> print(image.label['FILE_RECORDS'])  # access label
332
>>> plt.imshow(image.image, cmap='gray') # display image
```

3.2 Examples

Setup:

```
>>> %matplotlib inline
>>> from glob import glob
>>> import numpy as np
>>> import matplotlib.pyplot as plt
>>> import matplotlib.image as mpimg
>>> from planetaryimage import PDS3Image, CubeFile
```

Gather the Images:

```
>>> pdsfiles = glob("*.img")
>>> images = []
>>> for pdsfile in pdsfiles:
...     try:
...         images.append(PDS3Image.open(pdsfile))
...     except:
...         pass
>>> for n, image in enumerate(images):
...     print n, image
0 1p190678905erp64kcp2600l8c1.img
1 mk19s259.img
2 m0002320.img
```

```
3 mg00n217.sgr
4 h2225_0000_dt4.img
5 0044ML0205000000E1_DXXX.img
```

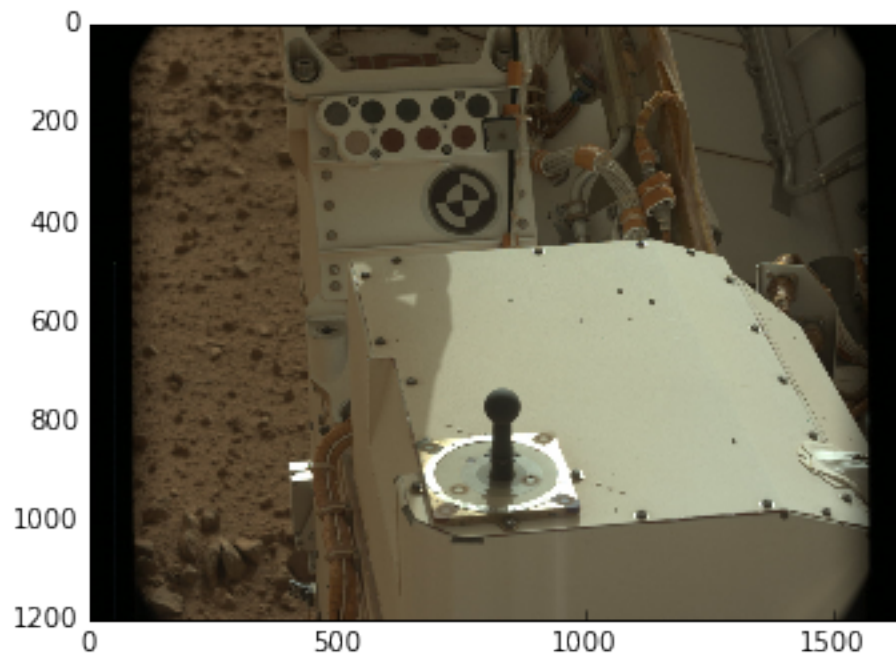
One can use the `try` statement in-case any of the images you have are not compatible with `PDS3image`. See [Supported Planetary Images List](#) to know what image types are compatible. The `for` loop will show what index number to use in future use of the image.

To see the information about each image:

```
>>> for i in images:
...     print i.data.dtype, i.data.shape, i.shape
>i2 (1, 1024, 32) (1, 1024, 32)
uint8 (1, 1331, 1328) (1, 1331, 1328)
uint8 (1, 1600, 384) (1, 1600, 384)
uint8 (1, 960, 964) (1, 960, 964)
>i2 (1, 10200, 1658) (1, 10200, 1658)
uint8 (3, 1200, 1648) (3, 1200, 1648)
```

To display a three band, color, image:

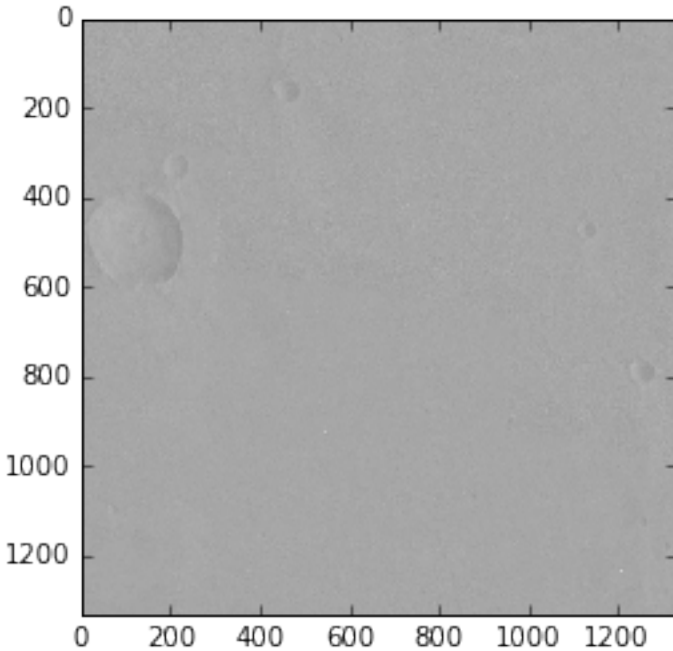
```
>>> caltarget = images[5]
>>> plt.imshow(caltarget.image)
<matplotlib.image.AxesImage at 0x10a13c250>
```



It is important to look at the first number in `i.shape` (See attributes) or the value from `i.bands`. If this number is 3, then the above example works, otherwise, you should use `cmap='gray'` parameter like in the below example.

To display a single band, grayscale, image:

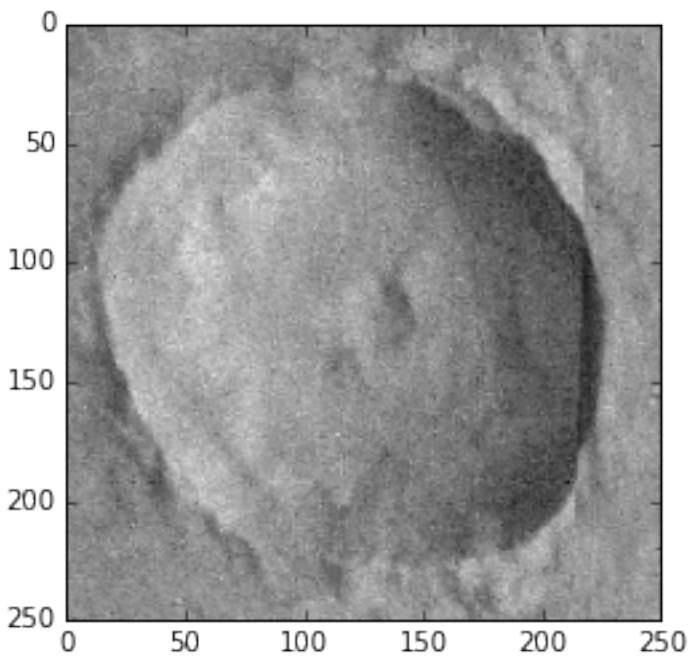
```
>>> image1 = images[1]
>>> plt.imshow(image1.image, cmap='gray')
<matplotlib.image.AxesImage at 0x125817a50>
```



The `cmap` keyword argument defines which `colormap` a grayscale image should be displayed with. In the case where `i.bands` is 3, it means the image is an RGB color image which does not need a colormap to be displayed properly. If `i.bands` is 1, then the image is grayscale and `imshow` would use its default colormap, which is not grayscale.

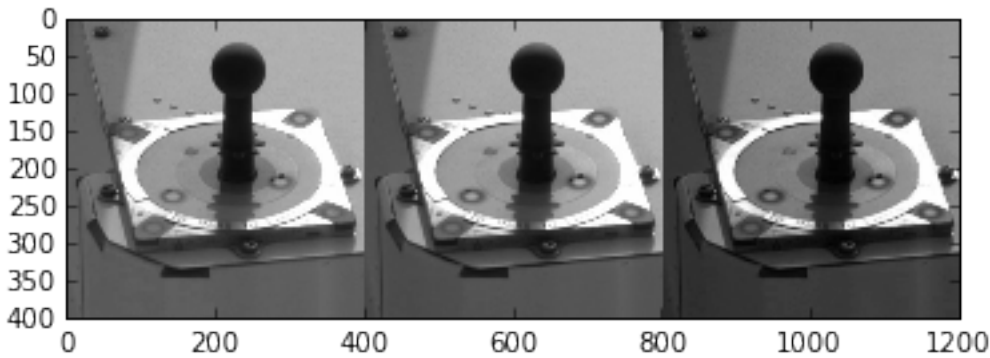
To see a subframe of an image:

```
>>> plt.imshow(image1.image[370:620, 0:250], cmap = 'gray')
<matplotlib.image.AxesImage at 0x11c014450>
```



To see the different bands of a colored image:

```
>>> plt.imshow(np.hstack([
    mcam1.image[700:1100, 600:1000, 0],
    mcam1.image[700:1100, 600:1000, 1],
    mcam1.image[700:1100, 600:1000, 2],
]), cmap='gray')
<matplotlib.image.AxesImage at 0x10fccd210>
```

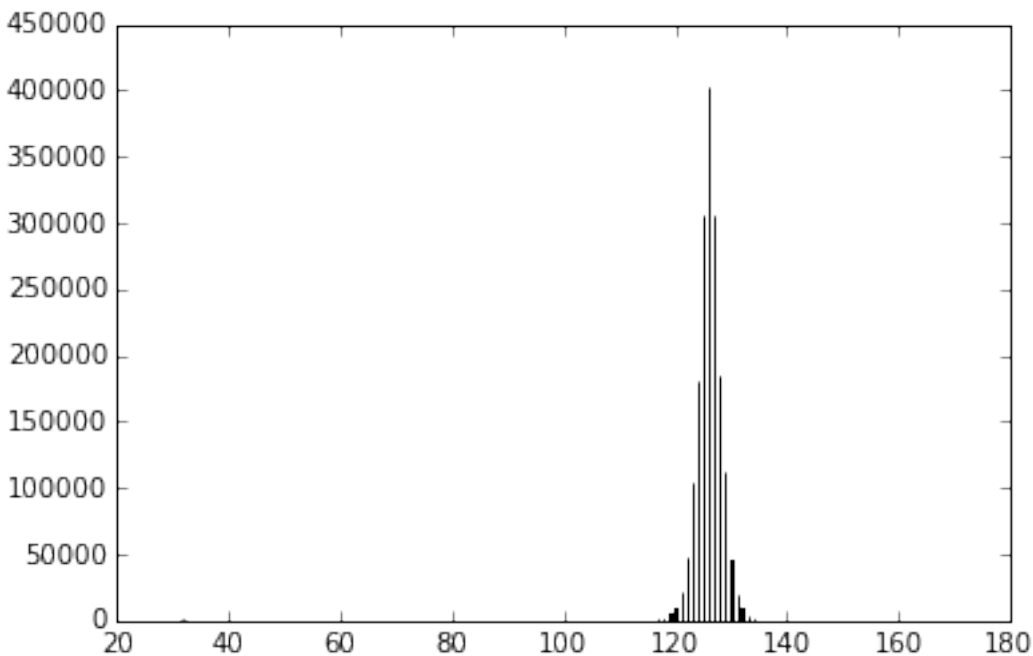


To save an image as a *.png* file for later viewing:

```
>>> crater = image1.image[370:620, 0:250]
>>> plt.imsave('crater.png', crater, cmap='gray')
```

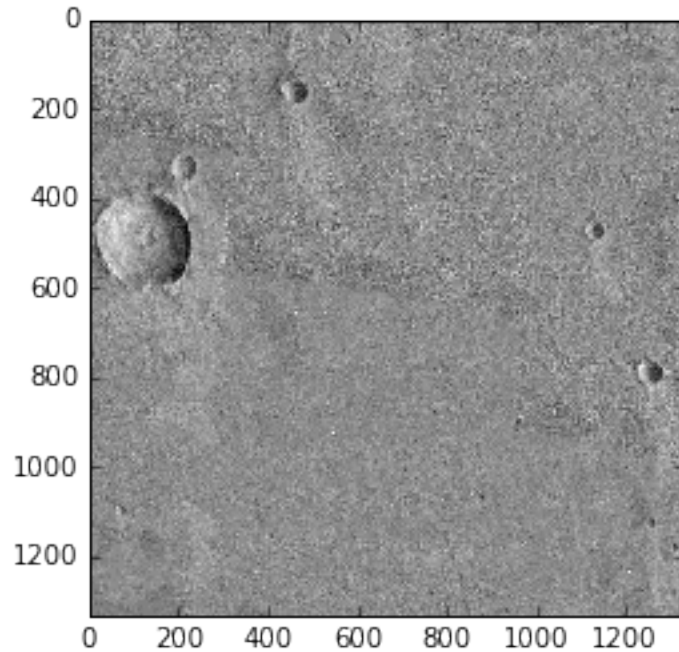
To do some image processing:

```
>>> plt.hist(image1.image.flatten(), 2000)
(array([ 2.,  0.,  0., ...,  0.,  0.,  1.]),
 array([ 32.,   32.036,  32.072, ..., 175.928, 175.964, 176.   ]),
 <a list of 4000 Patch objects>)
```



Using this Histogram can produce a clearer picture:

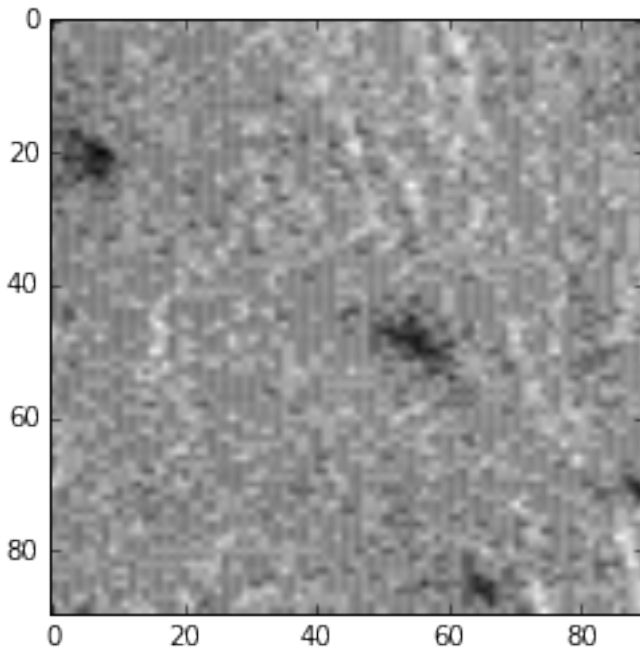
```
>>> plt.imshow(image1.image, cmap='gray', vmin=115, vmax=135)
<matplotlib.image.AxesImage at 0x1397a2790>
```



See [documentation for plt.imshow](#) and [Image tutorial for pyplot](#) to see more methods of image processing.

You can also use planetaryimage to process Isis Cube Files:

```
>>> from planetaryimage import CubeFile
>>> isisimage = CubeFile.open("tests/data/pattern.cub")
>>> isisimage.data.dtype, isisimage.data.shape, isisimage.shape
(dtype('<f4'), (90, 90), (1, 90, 90))
>>> plt.imshow(isisimage.image, cmap='gray')
<matplotlib.image.AxesImage at 0x114010610>
```



PlanetaryImage uses three main classes: *PlanetaryImage*, *PDS3Image*, and *CubeFile*. *PlanetaryImage* is the base class for *PDS3Image* and *CubeFile* and can be inherited for other image readers. *PDS3Image* and *CubeFile* are used to read images. See [Usage](#) to see more in depth examples using *PDS3Image* and *CubeFile*.

Contents

- *PlanetaryImage*
- *PDS3Image*
- *CubeFile*

PlanetaryImage

class planetaryimage.image.**PlanetaryImage**(*stream_string_or_array*, *filename=None*, *compression=None*)

A generic image reader. Parent object for PDS3Image and CubeFile

Parameters **stream**

file object to read as an image file

filename : string

an optional filename to attach to the object

compression : string

an optional string that indicate the compression type 'bz2' or 'gz'

Examples

```
>>> from planetaryimage import PDS3Image
>>> testfile = 'tests/mission_data/2p129641989eth0361p2600r8m1.img'
>>> image = PDS3Image.open(testfile)
>>> # Examples of attributes
>>> image.bands
1
>>> image.lines
64
>>> image.samples
64
>>> str(image.format)
'BAND_SEQUENTIAL'
>>> image.data_filename
>>> image.dtype
dtype('>i2')
>>> image.start_byte
34304
>>> image.shape
(1, 64, 64)
>>> image.size
4096
```

See <https://planetaryimage.readthedocs.io/en/latest/usage.html> to see how to open images to view them and make manipulations.

Attributes

compression	(string) Compression type (i.e. 'gz', 'bz2', or None).
data	(numpy array) A numpy array representing the image.
filename	(string) The filename given.
label	(pvl module) The image's label in dictionary form.

bands

Number of image bands.

data = None

A numpy array representing the image

data_filename

Return detached filename else None.

dtype

Pixel data type.

filename = None

The filename if given, otherwise none.

format

Image format.

image

An Image like array of `self.data` convenient for image processing tasks

- 2D array for single band, grayscale image data

- 3D array for three band, RGB image data

Enables working with `self.data` as if it were a PIL image.

See <https://planetaryimage.readthedocs.io/en/latest/usage.html> to see how to open images to view them and make manipulations.

label = None

The parsed label header in dictionary form.

lines

Number of lines per band.

classmethod open (filename)

Read an image file from disk

Parameters **filename** : string

Name of file to read as an image file. This file may be gzip (.gz) or bzip2 (.bz2) compressed.

samples

Number of samples per line.

shape

Tuple of images bands, lines and samples.

size

Total number of pixels

start_byte

Index of the start of the image data (zero indexed).

PDS3Image

class planetaryimage.pds3image.**PDS3Image**(*stream_string_or_array*, *filename=None*, *compression=None*)

Bases: *planetaryimage.image.PlanetaryImage*

A PDS3 image reader.

Examples

```
>>> from planetaryimage import PDS3Image
>>> testfile = 'tests/mission_data/2p129641989eth0361p2600r8m1.img'
>>> image = PDS3Image.open(testfile)
>>> # Examples of PDS3Image Attributes
>>> image.dtype
dtype('>i2')
>>> image.record_bytes
128
>>> image.data_filename
```

dtype

Pixel data type.

record_bytes

Number of bytes for fixed length records.

CubeFile

class planetaryimage.cubefile.**CubeFile**(*stream_string_or_array*, *filename=None*, *compression=None*)

Bases: *planetaryimage.image.PlanetaryImage*

A Isis Cube file reader.

Examples

```
>>> from planetaryimage import CubeFile
>>> image = CubeFile.open('tests/data/pattern.cub')
>>> # Examples of CubeFile Attributes
>>> image.base
0.0
>>> image.multiplier
1.0
>>> image.specials['His']
-3.4028233e+38
>>> image.tile_lines
128
>>> image.tile_samples
128
>>> image.tile_shape
(128, 128)
```

apply_numpy_specials(*copy=True*)

Convert isis special pixel values to numpy special pixel values.

Isis	Numpy
Null	nan
Lrs	-inf
Lis	-inf
His	inf
Hrs	inf

Parameters *copy* : bool [True]

Whether to apply the new special values to a copy of the pixel data and leave the original unaffected

Returns Numpy Array

A numpy array with special values converted to numpy's nan, inf, and -inf

apply_scaling (*copy=True*)

Scale pixel values to there true DN.

Parameters *copy*: bool [True]

Whether to apply the scaling to a copy of the pixel data and leave the original unaffected

Returns Numpy Array

A scaled version of the pixel data

base

An additive factor by which to offset pixel DN.

data_filename

Return detached filename else None.

get_image_array ()

Create an array for use in making an image.

Creates a linear stretch of the image and scales it to between 0 and 255. *Null*, *Lis* and *Lrs* pixels are set to 0. *His* and *Hrs* pixels are set to 255.

Usage:

```
from planetaryimage import CubeFile
from PIL import Image

# Read in the image and create the image data
image = CubeFile.open('test.cub')
data = image.get_image_array()

# Save the first band to a new file
Image.fromarray(data[0]).save('test.png')
```

Returns A uint8 array of pixel values.

multiplier

A multiplicative factor by which to scale pixel DN.

specials

Return the special pixel values

specials_mask ()

Create a pixel map for special pixels.

Returns An array where the value is *False* if the pixel is special
and *True* otherwise

tile_lines

Number of lines per tile.

tile_samples

Number of samples per tile.

tile_shape

Shape of tiles.

Supported Planetary Image Types

Table of Contents

- *General Information*
- *PDS List*
 - *Mars Science Laboratory*
 - *Lunar Reconnaissance Orbiter*
 - *Phoenix*
 - *Chandrayaan 1*
 - *Mars Reconnaissance Orbiter*
 - *MESSENGER*
 - *Mars Exploration Rover*
 - *ESA Mars Express*
 - *2001 Mars Odyssey*
 - *Cassini*
 - *Mars Pathfinder*
 - *Mars Global Surveyor*
 - *Shoemaker Levy 9*
 - *Clementine*
 - *Galileo*
 - *Magellan*
 - *Voyager*
 - *Viking Lander*
 - *Viking Orbiter*
 - *Mariner 10*
 - *Mariner 9*

7.1 General Information

Structure

- **Mission - Launch Year - Destination**
 - **Camera (Acronym)**
 - * **Volume - Description**
 - Symbol Image: Link_to_file
 - Label: Link_to_file
 - * **Volume - Description**
 - Symbol Image: Link_to_file
 - Label: Link_to_file
- Mission - Launch Year - Destination

This list is in chronological order, based on launch date, with the most recent mission at the top of the list and older missions at the bottom.

Symbols

-  = image works
-  = image does not work
-  = image has not been tested
-  = planetaryimage can open the file but `plt.imshow` does not work

Editing

If editing this document, to mark an image as good with a green square () add `|good|`, mark bad with a red square () add `|bad|`, mark that imshow does not work with a blue square () add `|nodisp|` and unknown/not tested with a white square () add `|none|` before “Image” like so:

```
|good| Image: image.img
|bad| Image: image.img
|none| Image: image.img
|nodisp| Image: image.img
```

See [Usage](#) for documentation on how to open the images.

7.2 PDS List

- **Mars Science Laboratory (MSL) - 2011 - Mars**
 - [Engineering Cameras](#)
 - * Volume: MSLHAZ_0XXX - Hazcam
 -  Image: FLB_431397159EDR_F0141262FHAZ00323M1.IMG

- Label: FLB_431397159EDR_F0141262FHAZ00323M1.LBL
- * Volume: MSLNAV_0XXX - Navcam
 -  Image: NLA_397671934ECS_N0010008AUT_04096M1.IMG
 - Label: NLA_397671934ECS_N0010008AUT_04096M1.LBL
- * Volume: MSLMOS_1XXX - Mosaics
 -  Image: N_A000_0012XEDR003CYPTUM0004XTOPMTM1.IMG
 - Label: N_A000_0012XEDR003CYPTUM0004XTOPMTM1.LBL
- Mast Camera (Mastcam)
 - * Volume: MSLMST_0002 - EDR
 -  Image: 0025ML0001270000100807E01_XXXX.DAT
 - Label: 0025ML0001270000100807E01_XXXX.LBL
 - * Volume: MSLMST_0002 - RDR
 -  Image: 0025ML0001270000100807E01_DRCL.IMG
 - Label: 0025ML0001270000100807E01_DRCL.LBL
- Mars Hand Lens Imager (MAHLI)
 - * Volume: MSLMHL_0002 - EDR
 -  Image: 0047MH0000110010100214C00_XXXX.DAT
 - Label: 0047MH0000110010100214C00_XXXX.LBL
 - * Volume: MSLMHL_0002 - RDR
 -  Image: 0047MH0000110010100214C00_DRCL.IMG
 - Label: 0047MH0000110010100214C00_DRCL.LBL
- Mars Descent Imager (MARDI)
 - * Volume: MSLMRD_0002 - EDR
 -  Image: 0000MD0000000000100027C00_XXXX.DAT
 - Label: 0000MD0000000000100027C00_XXXX.LBL
 - * Volume: MSLMRD_0002 - RDR
 -  Image: 0000MD0000000000100027C00_DRCL.IMG
 - Label: 0000MD0000000000100027C00_DRCL.LBL
- Lunar Reconnaissance Orbiter (LRO) - 2009 - Moon
 - Lyman-Alpha Mapping Project (LAMP)
 - * LROLAM_0007 - EDR
 -  Image: LAMP_ENG_0322531705_02.fit
 - Label: LAMP_ENG_0322531705_02.lbl

- * LROLAM_1010 - RDR
 -  Image: [LAMP_SCI_0345885974_03.fit](#)
 - Label: [LAMP_SCI_0345885974_03.lbl](#)
- * LROLAM_2001 - GDR
 -  Image:
[LAMP_80n_240mpp_long_dqual_01.img](#)
 - Label: [LAMP_80n_240mpp_long_dqual_01.lbl](#)
- Lunar Reconnaissance Orbiter Camera (LROC)
 - * LROLRC_0010 - Narrow Angle Camera - EDR
 -  Image: [M181639328RE.IMG](#)
 - * LROLRC_0010 - Wide Angle Camera - EDR
 -  Image: [M181648212CE.IMG](#)
 - * LROLRC_1015 - Narrow Angle Camera - CDR
 -  Image: [M1119524889RC.IMG](#)
 - * LROLRC_1015 - Wide Angle Camera - CDR
 -  Image: [M1119570719MC.IMG](#)
 - * LROLRC_2001 - RDR - Narrow Angle Camera
 -  Image: [NAC_ROI_FLMSTEEDHIA_E023S3168_20M.IMG](#)
 - * LROLRC_2001 - RDR - Wide Angle Camera
 -  Image: [WAC_ROI_FARSIDE_DUSK_P900S0000_100M.IMG](#)
- Lunar CRater Observation and Sensing Satellite (LCROSS)
 - * Volume 1 - Mid Infrared Camera 1 (MIR1)
 -  Image: [LCROSS_MIR1_CAL_20091009113134589.IMG](#)
 - Label: [LCROSS_MIR1_CAL_20091009113134589.LBL](#)
 - * Volume 1 - Mid Infrared Camera 2 (MIR2)
 -  Image: [LCROSS_MIR2_CAL_20091009113110458.IMG](#)
 - Label: [LCROSS_MIR2_CAL_20091009113110458.LBL](#)
 - * Volume 1 - Near Infrared Camera 1 (NIR1)
 -  Image: [LCROSS_NIR1_CAL_20091009113055156.IMG](#)
 - Label: [LCROSS_NIR1_CAL_20091009113055156.LBL](#)
 - * Volume 1 - Near Infrared Camera 2 (NIR2)

-  Image: LCROSS_NIR2_CAL_20091009113413068.IMG
 - Label: LCROSS_NIR2_CAL_20091009113413068.LBL
 - * Volume 1 - Near Infrared Spectrometer 1 (NISP1)
 -  Image: LCROSS_NSP1_CAL_20091009113218380.TAB
 - Label: LCROSS_NSP1_CAL_20091009113218970.LBL
 - * Volume 1 - Near Infrared Spectrometer 2 (NISP2)
 -  Image: LCROSS_NSP2_CAL_20091009113129443.TAB
 - Label: LCROSS_NSP2_CAL_20091009113130032.LBL
 - * Volume 1 - Total Luminence Photometer (TLP)
 -  Image: LCROSS_TLP_CAL_20091009104100.TAB
 - Label: LCROSS_TLP_CAL_20091009104100.LBL
 - * Volume 1 - Visible Camera (VIS)
 -  Image: LCROSS_VIS_RAW_20091009113358274.IMG
 - Label: LCROSS_VIS_RAW_20091009113358274.LBL
 - * Volume 1 - Visible Spectrometer (VSP)
 -  Image: LCROSS_VSP_CAL_20091009113528839.TAB
 - Label: LCROSS_VSP_CAL_20091009113529919.LBL
- Phoenix - 2008 - Mars
 - Optical Microscope (OM)
 - * Volume: phxom_0xx - Experiment Data Records
 -  Image: os017eff897721638_123d0mbm1.img
 - * Volume: phxsci_0xx - Science Reduced Data Records
 -  Image: os009rad897013001_119b0mbr1.img
 - Robotic Arm Camera (RAC)
 - * Volume: phxmos_0xx - Moasaics
 -  Image: r_013eff_cyl_sr11e4e_mdddm1.IMG
 - Label: r_013eff_cyl_sr11e4e_mdddm1.LBL
 - * Volume: phxrac_1xx - Reduced Data Records
 -  Image: rs004ff1896573565_10f86mdm1.img
 - * Volume: phxrac_0xx - Experiment Data Records
 -  Image: rs004eff896573565_10f86mdm1.img
 - * Volume: phxsci_0xx - Science Reduced Data Records

-  Image: rs003rad896482473_10e31mbr1.img
- Surface Stereo Imager (SSI)
 - * Volume: phxmos_0xx - Mosaics
 -  Image: s_000eff_cyl_sr10ca8_r222m1.img
 - Label: s_000eff_cyl_sr10ca8_r222m1.lbl
 - * Volume: phxsci_0xx - Science Reduced Data Records
 -  Image: ss000iof896227848_10c70r1t1.img
 - * Volume: phxssi_0xx - Experiment Data Records
 -  Image: ss000edn896227848_10c70r1m1.img
 - * Volume: phxssi_1xx - Reduced Data Records
 -  Image: ss004dil896560177_11684l1m1.img
- Chandrayaan-1 - 2008 - Moon
 - Moon Mineralogy Mapper (M³)
 - * CH1M3_0003
 -  Image: M3G20090714T080142_V03_LOC.IMG
 - Label: M3G20090714T080142_V03_L1B.LBL
 - Other: M3G20090714T080142_V03_LOC.HDR
- Mars Reconnaissance Orbiter (MRO) - 2005 Mars
 - High Resolution Imaging Science Experiment (HiRISE)
 - * Volume 1 (accumulating) - EDR
 -  Image: PSP_007978_2005_RED4_1.IMG
 - * Volume 1 (accumulating) - RDR
 -  Image: PSP_005109_1770_COLOR.JP2
 - Label: PSP_005109_1770_COLOR.LBL
 - Context Camera (CTX)
 - * Release 20
 -  Image: G13_023307_1051_XN_74S099W.IMG
 - Mars Color Imager (MARCI)
 - * Release 20
 -  Image: G12_022891_3112_MA_00N278W.IMG
- MESSENGER - 2004 - Mercury
 - Mercury Dual Imaging System (MDIS)

- * MSGRMDS_8001 - Regional Targeted Mosaic RDR (RTM) Narrow Angle Camera
 -  Image: [MDIS_RTM_N01_006966_5568032_0.IMG](#)
 - Label: [MDIS_RTM_N01_006966_5568032_0.LBL](#)
- * MSGRMDS_8001 - Regional Targeted Mosaic RDR (RTM) Wide Angle Camera
 -  Image: [MDIS_RTM_W11_006648_5217862_0.IMG](#)
 - Label: [MDIS_RTM_W11_006648_5217862_0.LBL](#)
- * MSGRMDS_7101 - High-Incidence Angle Basemap Illuminated from the West (HIW)
 -  Image: [MDIS_HIW_256PPD_H12NE0.IMG](#)
 - Label: [MDIS_HIW_256PPD_H12NE0.LBL](#)
- * MSGRMDS_6001 - MDIS 3-Color Map
 -  Image: [MDIS_MD3_128PPD_H11SW0.IMG](#)
 - Label: [MDIS_MD3_128PPD_H11SW0.LBL](#)
- * MSGRMDS_5001 - Multispectral Reduced Data Record (MDR)
 -  Image: [MDIS_MDR_064PPD_H10SW2.IMG](#)
 - Label: [MDIS_MDR_064PPD_H10SW2.LBL](#)
- * MSGRMDS_4001 - Basemap Reduced Data Record (BDR)
 -  Image: [MDIS_BDR_256PPD_H08NW0.IMG](#)
 - Label: [MDIS_BDR_256PPD_H08NW0.LBL](#)
- * MSGRMDS_3001 - Derived Data Record (DDR)
 -  Image: [DW1026713343K_DE_0.IMG](#)
- * MSGRMDS_2001 - calibrated data (CDR)
 -  Image: [CN1052412325M_IF_4.IMG](#)
 -  Image: [CN1052412325M_RA_4.IMG](#)
- Mars Exploration Rover (MER) - 2003 - Mars -Opportunity
 - Alpha Particle X-Ray Spectrometer
 - * Volume: mer1ao_0xxx - EDR
 -  Image: [1a468769014edrciq8n1419n0m1.dat](#)
 - Label: [1a468769014edrciq8n1419n0m1.lbl](#)
 - Moessbauer Spectrometer
 - * Volume: mer1bo_0xxx - EDR
 -  Image: [1b188656262ed564kcn1940n0m1.dat](#)
 - Label: [1b188656262ed564kcn1940n0m1.lbl](#)

- Descent Camera
 - * Volume: mer1do_0xxx - EDR
 -  Image: 1e128278505edn0000f0006n0m1.img
- Hazard Avoidance Camera
 - * Volume: mer1ho_0xxx - EDR
 -  Image: 1f161026369edn42d9p1111l0m1.img
 - * Volume: mer1ho_0xxx - RDR
 -  Image: 1f161026369uvl42d9p1111l0m1.img
 - * Volume: mer1om_0xxx - RDR Mosaics
 -  Image: 1rr012eff02vrt42p1211l000m2.img
 - * Volume: mer1mw_0xxx - RDR Meshes
 -  Image: 1f139471884xyl3000p1214l0m1.rgb
 - Label: 1f139471884xyl3000p1214l0m1.lbl
- Microscopic Imager
 - * Volume: mer1mo_0xxx - EDR
 -  Image: 1m298459885effa312p2956m2m1.img
 - * Volume: mer1mo_0xxx - RDR
 -  Image: 1m298459667mrda312p2956m2m1.img
 - * Volume: mer1ms_0xxx - Science Products EDR
 -  Image: 1m228942450eff81d2p2976m2f1.img
 - * Volume: mer1ms_0xxx - Science Products RDR
 -  Image: 1m140877373cfd3190p2936m2f1.img
- Navigation Camera
 - * Volume: mer1no_0xxx - EDR
 -  Image: 1n129510489eff0312p1930l0m1.img
 - * Volume: mer1no_0xxx - RDR
 -  Image: 1n129510489mrl0312p1930l0m1.img
 - * Volume: mer1om_0xxx - Navcam - RDR Mosaics
 -  Image: 1nn013ilf03cyl00p1652l000m2.img
 - * Volume: mer1mw_0xxx - RDR Meshes
 -  Image: 1n137786085xyl2300p1981l0m1.rgb
 - Label: 1n137786085xyl2300p1981l0m1.lbl

- Panoromic Camera
 - * Volume: mer1po_0xxx - EDR
 -  Image: 1p134482118erp0902p2600r8m1.img
 - * Volume: mer1po_0xxx - RDR
 -  Image: 1p134482118sfl0902p2600l8m1.img
 - * Volume: mer1pc_0xxx - EDRs
 -  Image: 1p190678905erp64kcp2600l8c1.img
 - * Volume: mer1pc_1xxx - RDRs
 -  Image: 1p144429114rat3370p2542l2c1.img
 - * Volume: mer1om_0xxx - Pancam - RDR Mosaics
 -  Image: 1pp081ilf11cyp00p2425l777m1.img
 - * Volume: mer1mw_0xxx - RDR Meshes
 -  Image: 1p137953271xyl2513p2366l7m1.rgb
 - Label: 1p137953271xyl2513p2366l7m1.lbl
- Mars Exploration Rover (MER) - 2003 - Mars - Spirit
 - Alpha Particle X-ray Spectrometer
 - * Volume: mer2ao_0xxx - EDR
 -  Image: 2a132656587edr1800n1438n0m1.dat
 - Label: 2a132656587edr1800n1438n0m1.lbl
 - Moessbauer Spectrometer
 - * Volume mer2bo_0xxx - EDR
 -  Image: 2b129423244ed50327n1940n0m1.dat
 - Label: 2b129423244ed50327n1940n0m1.lbl
 - Descent Camera
 - * Volume: mer2do_0xxx - EDR
 -  Image: 2e126462398edn0000f0006n0m1.img
 - Hazard Avoidance Camera
 - * Volume: mer2ho_0xxx - EDR
 -  Image: 2f130356488eff0800p1110r0m1.img
 - * Volume: mer2ho_0xxx - RDR
 -  Image: 2f130352973ilf0800p1120r0m1.img
 - * Volume: mer2mw_0xxx - Hazcam - RDR Meshes

-  Image: 2f132759178xyl2000p1212l0m1.rgb
- Label: 2f132759178xyl2000p1212l0m1.lbl
- * Volume: mer2om_0xxx - RDR Mosaics
-  Image: 2ff010eff02per11p1003l000m2.img
- Microscopic Imager
- * Volume: mer2mo_0xxx - EDR
-  Image: 2m130974443eff1100p2953m2m1.img
- * Volume: mer2mo_0xxx - RDR
-  Image: 2m130974067rst1100p2942m1m1.img
- * Volume: mer2ms_0xxx - Science Products EDR
-  Image: 2m133285881eff2232p2971m2f1.img
- * Volume: mer2ms_0xxx - Science Products RDR
-  Image: 2m132591087cfd1800p2977m2f1.img
- Navigation Camera
- * Volume: mer2no_0xxx - EDR
-  Image: 2n129472048eth0327p1874l0m1.img
- * Volume: mer2no_0xxx - RDR
-  Image: 2n129472048inn0327p1874r0m1.img
- * Volume: mer2mw_0xxx - RDR Meshes
-  Image: 2n131962517xyl1400p1917l0m1.rgb
- Label: 2n131962517xyl1400p1917l0m1.lbl
- * Volume: mer2om_0xxx - RDR Mosaics
-  Image: 2nn043ilf06cyp00p1817l000m1.img
- Panoromic Camera
- * Volume: mer2po_0xxx - EDR
-  Image: 2p129641989eth0361p2600r8m1.img
- * Volume: mer2po_0xxx - RDR
-  Image: 2p129641989mrd0361p2600r8m1.img
- * Volume: mer2mw_0xxx - Camera RDR Meshes
-  Image: 2p132046745xyl1500p2445l7m1.rgb

- Label: 2p132046745xy11500p244517m1.lbl
 - * Volume: mer2om_0xxx - Camera RDR Mosaics
 -  Image: 2pp062ilf13cyp00p2119l666m1.img
 - * Volume: mer2pc_0xxx - Science Products (EDRs)
 -  Image: 2p130614950erp09bvp2556r1c1.img
 - * Volume: mer2pc_1xxx - Science Products (RDRs)
 -  Image: 2p130975038rad1100p2820l4c1.img
- Rock Abrasion Tool
 - * Volume: mer2ro_0xxx - EDR
 -  Image: 2d147320057edr8600d2515n0m1.dat
- ESA Mars Express (MEX) - 2003 - Mars
 - High Resolution Stereo Camera (HRSC)
 - * mexhrsc_0001 - Radiometrically Calibrated Image
 -  Image: h9335_0000_p12.img
 - * mexhrsc_1001 - Map Projected Image
 -  Image: h5395_0000_p23.img
 - * mexhrsc_2001 - Orthophoto and DTM
 -  Image: h2225_0000_dt4.img
- 2001 Mars Odyssey - 2001 - Mars
 - Thermal Emission Imaging System (THEMIS)
 - * ODTGEO_v2 - Geometric Records
 -  Image: V52514013ALB.IMG
 - * ODTSDP_v1 - Standard Products
 -  Image: I53094006BTR.IMG
 -  Image: V48732003RDR.QUB
- Cassini - 1997 - Saturn
 - Imaging Science Subsystem (ISS)
 - * Volume: 1 - Saturn EDR
 -  Image: N1454725799_1.IMG
 - Label: N1454725799_1.LBL
 - * Volume: 1 - Narrow Angle Camera
 -  Image: 134600.img

- Label: 134600.lbl
- * Volume: 1 - Wide Angle Camera
 -  Image: 128078.img
 - Label: 128078.lbl
- Cassini Radar Instrument (RADAR)
 - * Volume: 35 - ABDR
 -  Image: ABDR_04_D035_V02.ZIP
 - Label: ABDR_04_D035_V02.LBL
 - * Volume: 35 - LBDR
 -  Image: LBDR_14_D035_V02.ZIP
 - Label: LBDR_14_D035_V02.LBL
 - * Volume: 35 - CALIB
 -  Image: BEAM1_V01.PAT
 - Label: BEAM1_V01.LBL
- Visual and Infrared Mapping Spectrometer (VIMS)
 - * Volume: covims-unks - QUBE EDRs
 -  Image: v1585148848_2.qub
 - Label: v1585148848_2.lbl
 - * Volume 5 - Spectral Cubes
 -  Image: v1477775070_4.qub
 - Label: v1477775070_4.lbl
 - * Volume: 35 - BIDR
 -  Image: BIBQD49N071_D035_T00AS01_V02.ZIP
 - Label: BIBQD49N071_D035_T00AS01_V02.LBL
 - NOTE: This only works *after* opening the zip file to retrieve the *.IMG* file.
- ISS RDR Cartographic Map Volumes
 - * Volume: coiss_3004 - RDR Cartographic Map
 -  Image: ST_1M_0_324_MERCATOR.IMG
- Mars Pathfinder - 1996 - Mars
 - Atmospheric Structure Instrument and Meteorology (ASI-MET)
 - * mpam_0001 - Entry, Descent, and Landing ERDR
 -  Image: r_sacc_s.tab
 - Label: r_sacc_s.lbl

- * mpam_0001 - Surface EDR
 -  Image: [se0732s.tab](#)
 - Label: [se0732s.lbl](#)
- * mpam_0001 - Surface RDR
 -  Image: [sr0893s.tab](#)
 - Label: [sr0893s.lbl](#)
- Imager for Mars Pathfinder EDRs
 - * mpim_0003 - Rover Cameras
 -  Image: [i277783l.img](#)
- Rover Cameras/Alpha X-ray Spectrometer (APXS)
 - * mprv_0001 - APXS EDR
 -  Image: [a1526159.tab](#)
 - * mprv_0001 - APXS DDR
 -  Image: [ox_perc.tab](#)
 - * mprv_0001 - Rover Cameras EDR
 -  Image: [r9599891.img](#)
 - * mprv_0001 - Rover Cameras Mosaicked Image Data Record
 -  Image: [r01090al.img](#)
 - Label: [r01090al.haf](#)
- Mars Global Surveyor (MGS) - 1996 - Mars
 - Mars Orbiter Camera (MOC)
 - * mgsc_0005 - Decompressed Standard Data Products
 -  Image: [sp246804.img](#)
 - * mgsc_1006 - Standard Data Records
 -  Image: [m0002320.imq](#)
 - * RDRs
 - This data set is being prepared for peer review; it has not been reviewed by PDS and is NOT PDS-compliant and is NOT considered to be Certified Data.
- Shoemaker-Levy 9 - Comet - 1994
 - Event K, N and W - Observed by Galileo Near Infrared Mapping Spectrometer (NIMS)
 - * c024895/
 -  Image: [0600g.img](#)
 - Near Infrared Mapping Spectrometer (NIMS)

- * aareadme
- Clementine - 1994 - Moon
 - Experiment Data Records
 - * cl_0072 - Ultraviolet/Visible (UV/VIS) Camera
 -  Image: lub0204c.313
 - * cl_0078 - NearInfraRed (NIR) Camera
 -  Image: lna3869l.335
 - * cl_0058/ - Long Wave InfraRed (LWIR) Camera
 -  Image: lla2531k.252
 - * cl_0065 - High Resolution (HiRes) Camera
 -  Image: lhd1540h.279
 - Lunar Basemap Mosaics
 - * cl_3013
 -  Image: bi24s333.img
 - Label: bi24s333.lab
 - Full Resolution UVVIS Digital Image Model
 - * cl_4009
 -  Image: ui45s015.img
 - Label: ui45s015.lab
 - High Resolution Mosaics
 - * cl_6016
 -  Image: h58n3118.img
- Galileo - 1989 - Jupiter
 - Solid State Imaging (SSI)
 - * Volume: go_0003 - Raw EDRs
 -  Image: 9500r.img
 - Label: 9500r.lbl
 - Near-Infrared Mapping Spectrometer (NIMS) EDRs
 - * go_1005
 -  Image: e4i015.edr
 - NIMS CUBEs
 - * go_1107

-  Image: e6e004ti.qub
- Magellan - 1989 - Venus
 - Mosaicked Image Data Records
 - * mg_0124
 -  Image: ff05.img
 - Label: ff05.lbl
 - Full Resolution Radar Mosaics
 - * mg_1194
 -  Image: fl05s205.img
 - Global Altimetry and Radiometry Data Records
 - * mg_3002 - Global Emissivity Data Record (GEDR)
 -  Image: f18.img
 - Label: f18.lbl
 - * mg_3002 - Global Reflectivity Data Record (GREDR)
 -  Image: f31.img
 - Label: f31.lbl
 - * mg_3002 - Global Slope Data Record (GSDR)
 -  Image: f26.img
 - Label: f26.lbl
 - * mg_3002 - Global Topography Data Record (GTDR)
 -  Image: f30.img
 - Label: f30.lbl
 - Synthetic-aperture radar (SAR) Experiment Data Records (EDRs)
 - * Vol 1046
 -  Image: EDR2856A.07
- Voyager - 1977 - Interstellar Space
 - Imaging Science Subsystem (ISS)
 - * vg_0011 - EDR
 -  Image: c1138206.imq
 - ISS Calibrated Data Products
 - * VGISS_0026 - RDR
 -  Image: C3289235_RAW.IMG

- Label: C3289235_RAW.LBL
- Viking Lander - 1975 - Mars
 - Experiment Data Records
 - * vl_0001 - Viking Lander 1
 -  Image: 12j017.n06
 - * vl_0002 - Viking Lander 2
 -  Image: 21e147.grn
 - Processed Images
 - * The following are NOT PDS formatted volumes. They were produced by the Science Digital Data Preservation Task by copying data directly off of old, decaying tape media onto more stable CD-WO media. They have not been otherwise reformatted.
 - * vl_2111 - Viking Lander 1
 -  Image: vl_0901.002
 - * vl_2112 Viking Lander 2
 -  Image: vl_0958.003
- Viking Orbiter - 1975 - Mars
 - Experiment Data Records
 - * vo_1063
 -  Image: f673b55.imq
 - Digital Image Map
 - * vo_2004
 -  Image: mi35n227.img
 - Digital Topographic Maps
 - * vo_2007
 -  Image: tg00n217.img
 - Digital Color Mosaics
 - * vo_2011
 -  Image: mg00n217.sgr
 - High Resolution Mosaicked Digital Image Maps
 - * vo_2020
 -  Image: mk19s259.img
- Mariner 10 - 1973 - Mercury and Venus
 - Experiment Data Records

* “The following are NOT PDS formatted volumes. They were produced by the Science Digital Data Preservation Task by copying data directly from old, decaying tape media onto more stable CD-WO media, then transferred online. They have not been otherwise reformatted.”

* mvm_0013

·  Image: [mve_050.080](#)

- Mariner 9 - 1971 - Mars

- Experiment Data Records

- * mr9iss_0007

-  Image: [10060584.img](#)

- Label: [10060584.lbl](#)

Contributing

Contributions are welcome, and they are greatly appreciated! Every little bit helps, and credit will always be given. You can contribute in many ways:

8.1 Types of Contributions

8.1.1 Report Bugs

Report bugs at <https://github.com/planetarium/planetariumimage/issues>.

If you are reporting a bug, please include:

- Your operating system name and version.
- Any details about your local setup that might be helpful in troubleshooting.
- Detailed steps to reproduce the bug.

8.1.2 Fix Bugs

Look through the GitHub issues for bugs. Anything tagged with “bug” is open to whoever wants to implement it.

8.1.3 Implement Features

Look through the GitHub issues for features. Anything tagged with “feature” is open to whoever wants to implement it.

8.1.4 Write Documentation

planetariumimage could always use more documentation, whether as part of the official planetariumimage docs, in docstrings, or even on the web in blog posts, articles, and such.

8.1.5 Submit Feedback

The best way to send feedback is to file an issue at <https://github.com/planetarium/planetariumimage/issues>.

If you are proposing a feature:

- Explain in detail how it would work.
- Keep the scope as narrow as possible, to make it easier to implement.
- Remember that this is a volunteer-driven project, and that contributions are welcome :)

8.2 Get Started!

Ready to contribute? Here's how to set up *planetaryimage* for local development.

1. Fork the *planetaryimage* repo on GitHub.
2. Clone your fork locally:

```
$ git clone git@github.com:your_name_here/planetaryimage.git
```

3. Install your local copy into a virtualenv. Assuming you have virtualenvwrapper installed, this is how you set up your fork for local development:

```
$ mkvirtualenv planetaryimage
$ cd planetaryimage/
$ pip install -r requirements.txt
```

4. Create a branch for local development:

```
$ git checkout -b name-of-your-bugfix-or-feature
```

Now you can make your changes locally.

5. When you're done making changes, check that your changes pass flake8 and the tests, including testing other Python versions with tox:

```
$ make lint
$ make test
$ make test-all
```

To get flake8 and tox, just pip install them into your virtualenv.

6. Commit your changes and push your branch to GitHub:

```
$ git add .
$ git commit -m "Your detailed description of your changes."
$ git push origin name-of-your-bugfix-or-feature
```

7. Submit a pull request through the GitHub website.

8.3 Pull Request Guidelines

Before you submit a pull request, check that it meets these guidelines:

1. The pull request should include tests.
2. If the pull request adds functionality, the docs should be updated. Put your new functionality into a function with a docstring, and add the feature to the list in README.rst.
3. The pull request should work for Python 2.6, 2.7, 3.3, 3.4, and 3.5. Check https://travis-ci.org/planetarypy/planetaryimage/pull_requests and make sure that the tests pass for all supported Python versions.

8.4 Tips

To run a subset of tests:

```
py.test tests/test_planetaryimage
```

Credits

9.1 Development Lead

- PlanetaryPy Developers <contact@planetarypy.com>

9.2 Contributors

- Trevor Olson <trevor@heytreavor.com>
- Nikhil Kalige <nikhilkalige@gmail.com>
- Austin Godber <godber@uberhip.com>
- Bhavin Nayak <bhavin.nayak@asu.edu>

History

0.4.1 (2016-03-26)

- Fixes to saving PDS3 images files after modification of the data.

0.4.0 (2016-03-05)

- Added basic support for saving PDS3 images.

0.3.0 (2015-09-29)

- Added support for opening PDS images that are gzip or bz2 compressed.

0.2.0 (2015-06-17)

- Improved support for float type PDS3 image types. Many types still not supported.

0.1.0 (2015-06-03)

- First release on PyPI. Basic PDS and Isis Cube File parsing works.

Indices and tables

- `genindex`
- `modindex`
- `search`

p

`planetaryimage.image`, [13](#)

A

`apply_numpy_specials()` (planetaryimage.cubefile.CubeFile method), 17
`apply_scaling()` (planetaryimage.cubefile.CubeFile method), 17

B

`bands` (planetaryimage.image.PlanetaryImage attribute), 14
`base` (planetaryimage.cubefile.CubeFile attribute), 18

C

`CubeFile` (class in planetaryimage.cubefile), 17

D

`data` (planetaryimage.image.PlanetaryImage attribute), 14
`data_filename` (planetaryimage.cubefile.CubeFile attribute), 18
`data_filename` (planetaryimage.image.PlanetaryImage attribute), 14
`dtype` (planetaryimage.image.PlanetaryImage attribute), 14
`dtype` (planetaryimage.pds3image.PDS3Image attribute), 15

F

`filename` (planetaryimage.image.PlanetaryImage attribute), 14
`format` (planetaryimage.image.PlanetaryImage attribute), 14

G

`get_image_array()` (planetaryimage.cubefile.CubeFile method), 18

I

`image` (planetaryimage.image.PlanetaryImage attribute), 14

L

`label` (planetaryimage.image.PlanetaryImage attribute), 14
`lines` (planetaryimage.image.PlanetaryImage attribute), 14

M

`multiplier` (planetaryimage.cubefile.CubeFile attribute), 18

O

`open()` (planetaryimage.image.PlanetaryImage class method), 14

P

`PDS3Image` (class in planetaryimage.pds3image), 15
`PlanetaryImage` (class in planetaryimage.image), 13
planetaryimage.image (module), 13

R

`record_bytes` (planetaryimage.pds3image.PDS3Image attribute), 15

S

`samples` (planetaryimage.image.PlanetaryImage attribute), 14
`shape` (planetaryimage.image.PlanetaryImage attribute), 14
`size` (planetaryimage.image.PlanetaryImage attribute), 14
`specials` (planetaryimage.cubefile.CubeFile attribute), 18
`specials_mask()` (planetaryimage.cubefile.CubeFile method), 18
`start_byte` (planetaryimage.image.PlanetaryImage attribute), 14

T

`tile_lines` (planetaryimage.cubefile.CubeFile attribute), 18
`tile_samples` (planetaryimage.cubefile.CubeFile attribute), 18
`tile_shape` (planetaryimage.cubefile.CubeFile attribute), 18