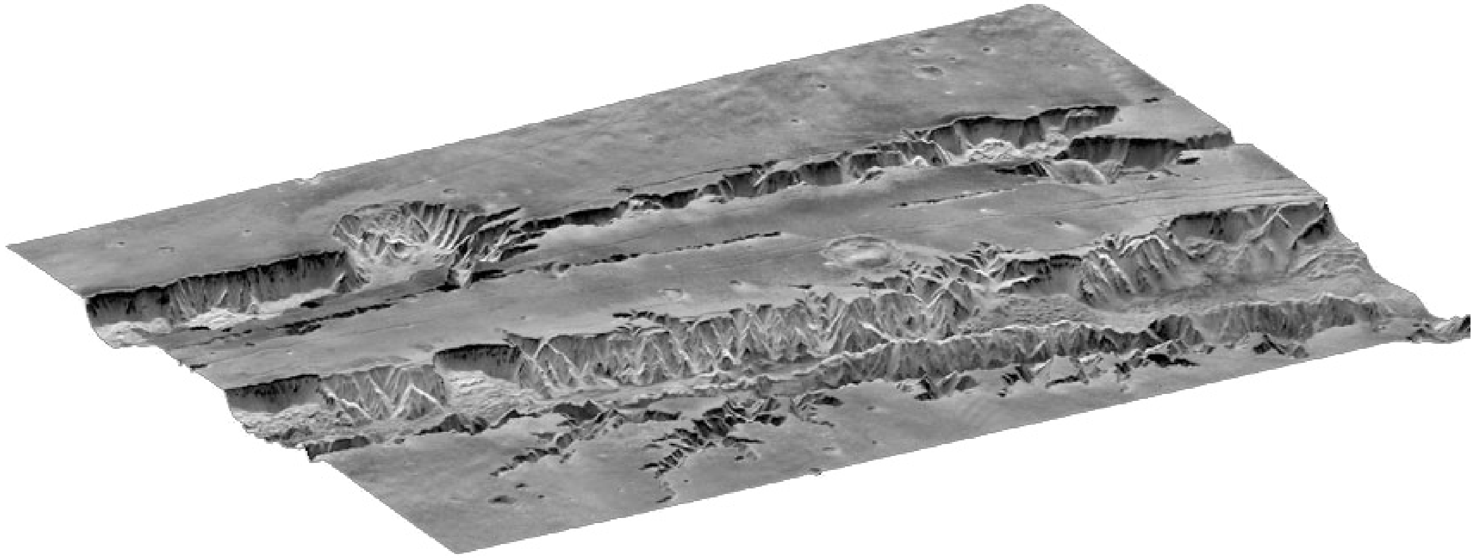
# HRSC Introduction

Angelo Pio Rossi

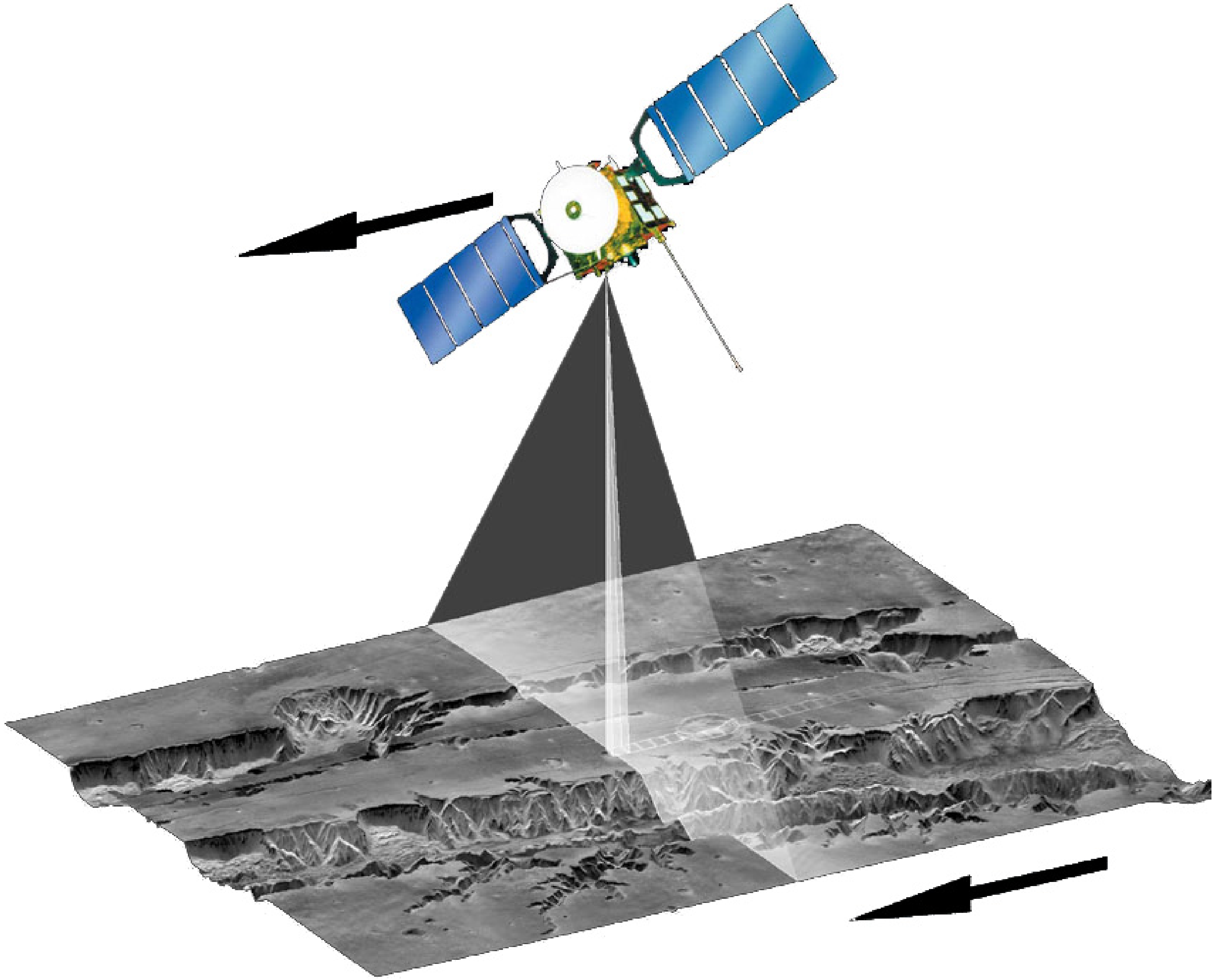
# HRSC stereo

HRSC stereo
imaging principle



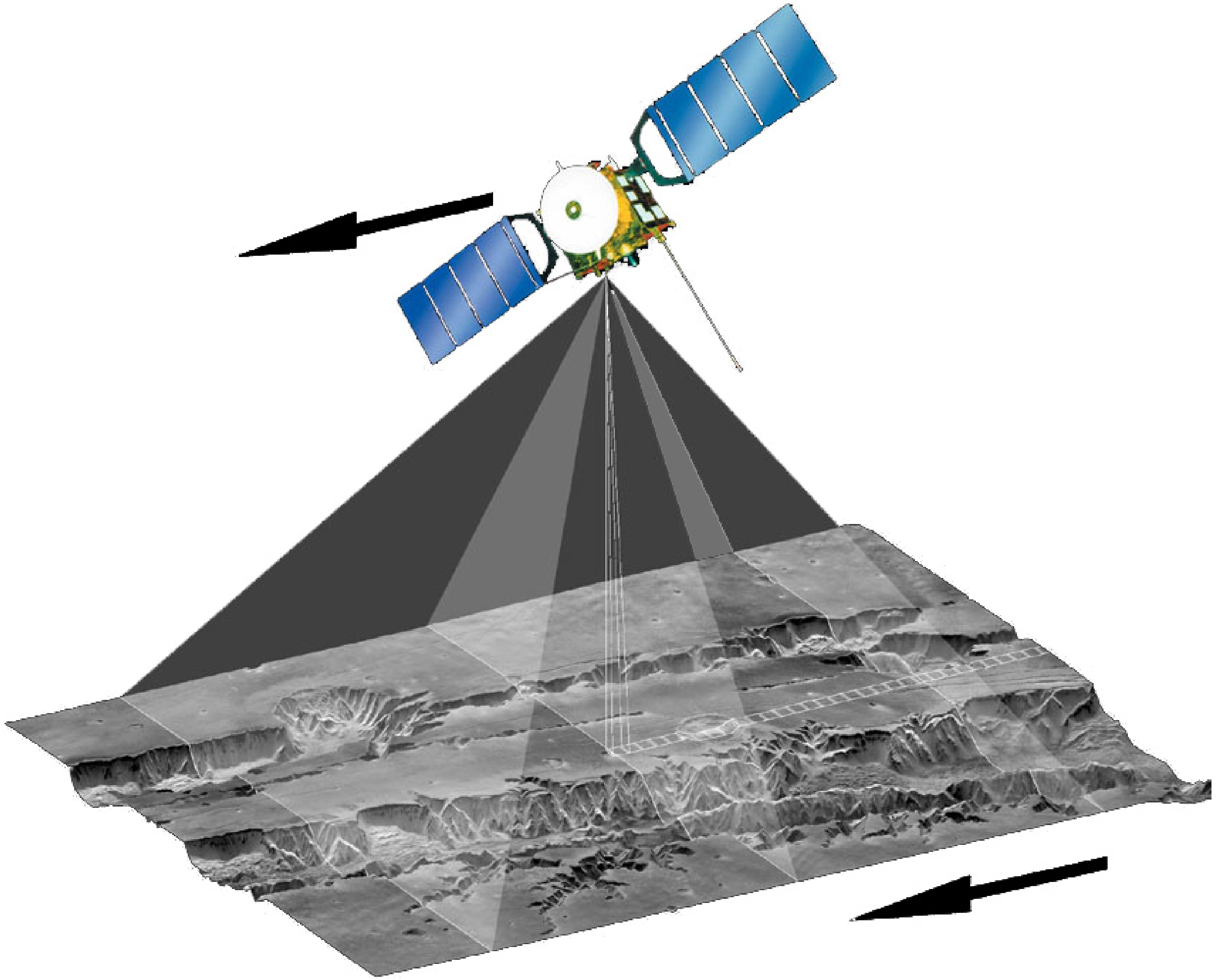
# HRSC stereo

HRSC stereo
imaging principle



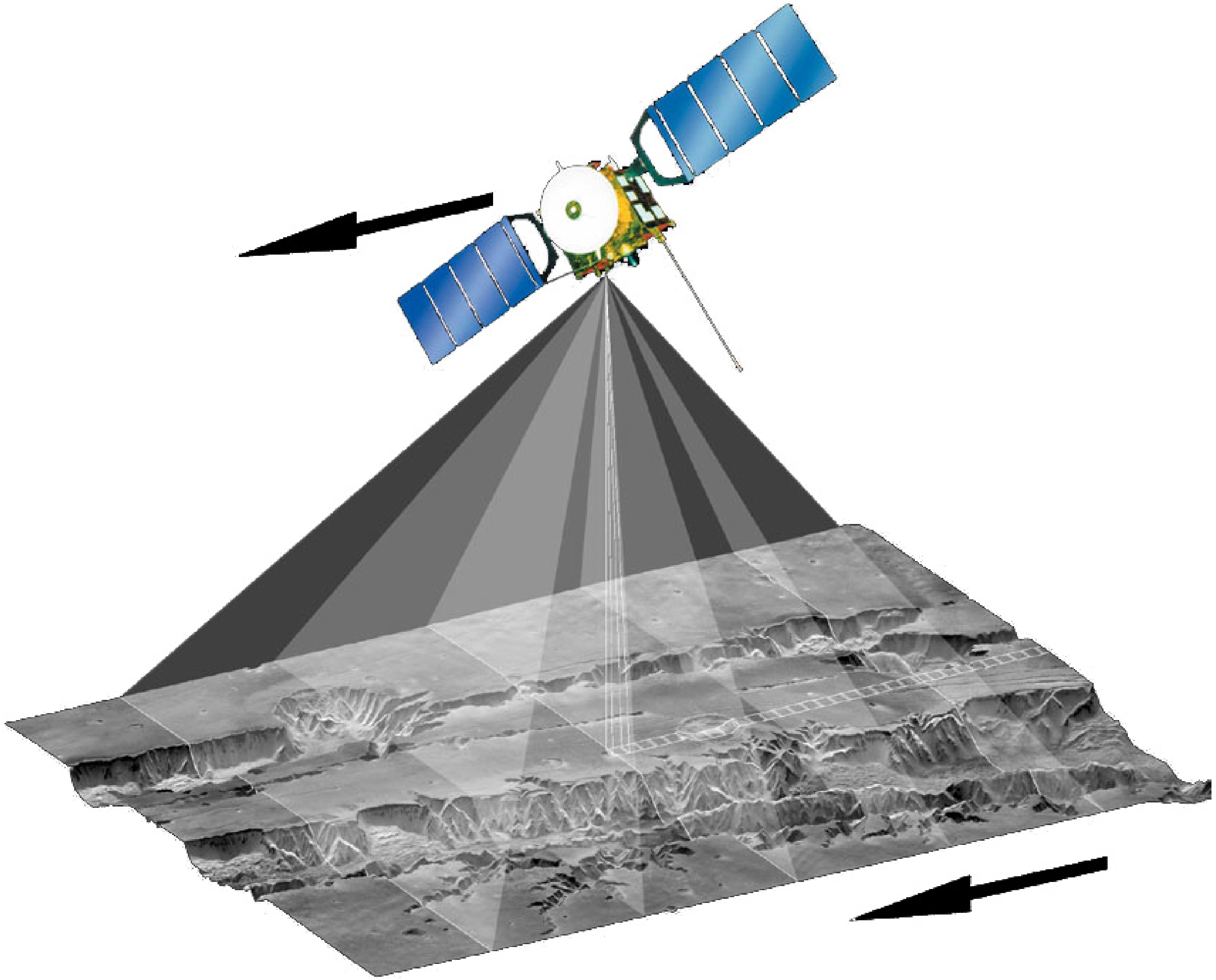
# HRSC stereo

HRSC stereo
imaging principle



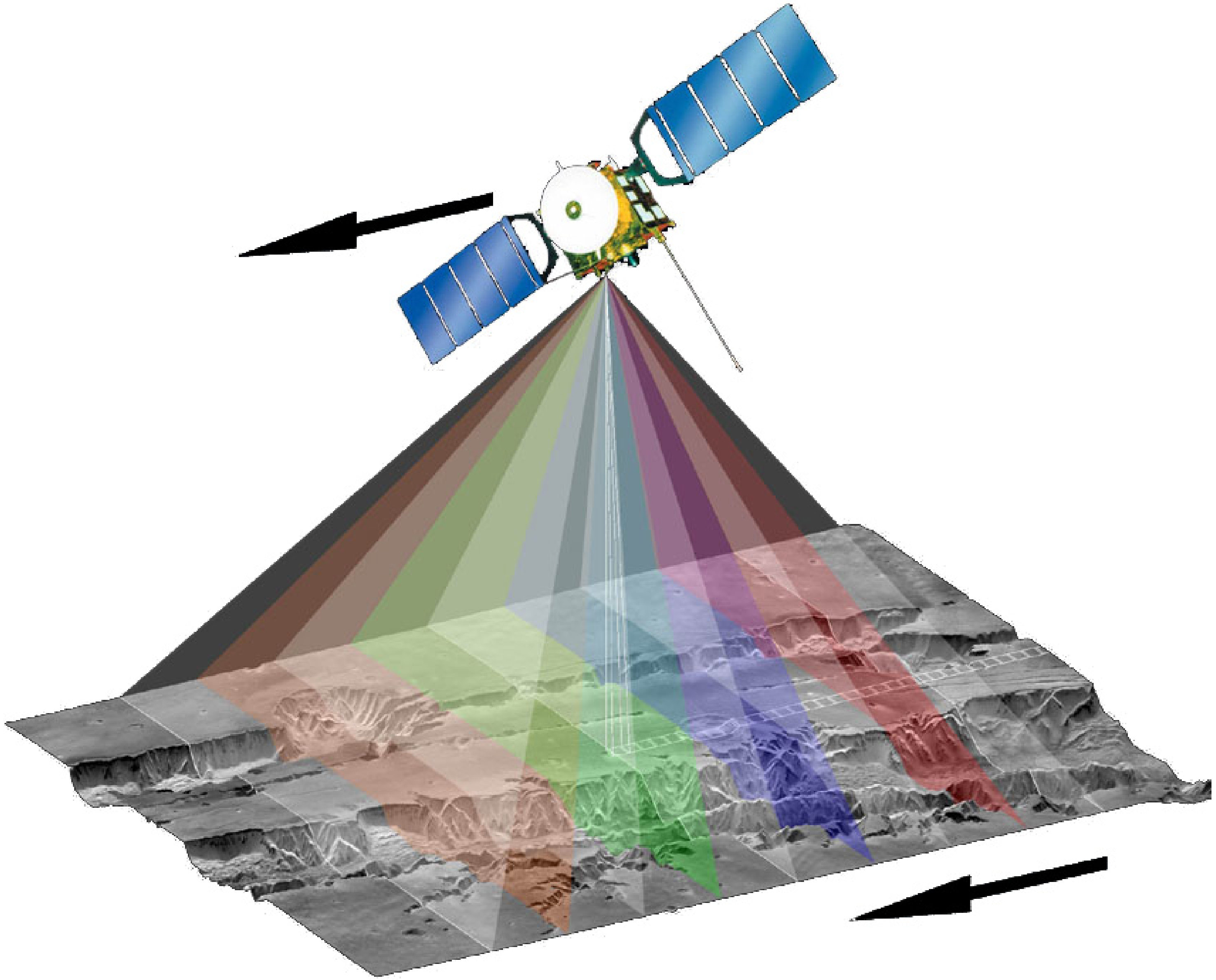
# HRSC stereo

HRSC stereo
imaging principle



# HRSC stereo

HRSC stereo
imaging principle



# HRSC data products

- Level-2: Radiometrically Calibrated DataInput for Photogrammetric Processing
- Level-3: Geometrically Corrected and Map ProjectedStandard Products for Geology
- Based on available SPICE kernels (may be predicted SPK, CK)

# HRSC data products

- Radiance = RADIANCE_OFFSET +DN * RADIANCE_SCALING_FACTOR (Unit is W*m**-2*sr**-1)

- Reflectance = DN * REFLECTANCE_SCALING_FACTOR

# Data type

- The data will be signed16-bit data either from data transmitted as 14-bit data (SRC only) or 8-bit data.
- The format of the data will be in raster form.
- All dark, dummy, isolation, and affected pixel are removed from the image.

# Level2 data

Error pixel are marked with the following negative values:

**-1** blemish pixel one pixel for SRC for one blemish

one whole column for HRSC for one blemish

**-2** bad pixel (telemetry or decompression problems)

typical size: 8 lines x 512 pixels

**-8** saturated pixel

Bad pixels affect other pixels during compression (8x8) !

# File naming conventions

The following structure of filenames will be used:

Hoooo_mmmm_DDl.IMG

| | |
|---|---|
| - camera id | H |
| - orbit number | oooo |
| - image number in one orbit | mmmm |
| - detector id | dd |
| (can be nd, s1, s2, p1, p2, re,ir, gr, bl, 10, 11, sr) | |
| - level (0,1,2,3) | l |
| -version | vv |
| -example: | H1234_0023_ND2.IMG |

(small letters at PDS)

# Level3 data

MEX-M-HRSC-5-REFDR-MAPPROJECTED-V2.0

- Sinusoidal projection for |center_latitude| < 85°, radius = 3396.19

- Stereographic projection |center_latitude| > 85°, radius = 3376.2

# HRSC - name examples

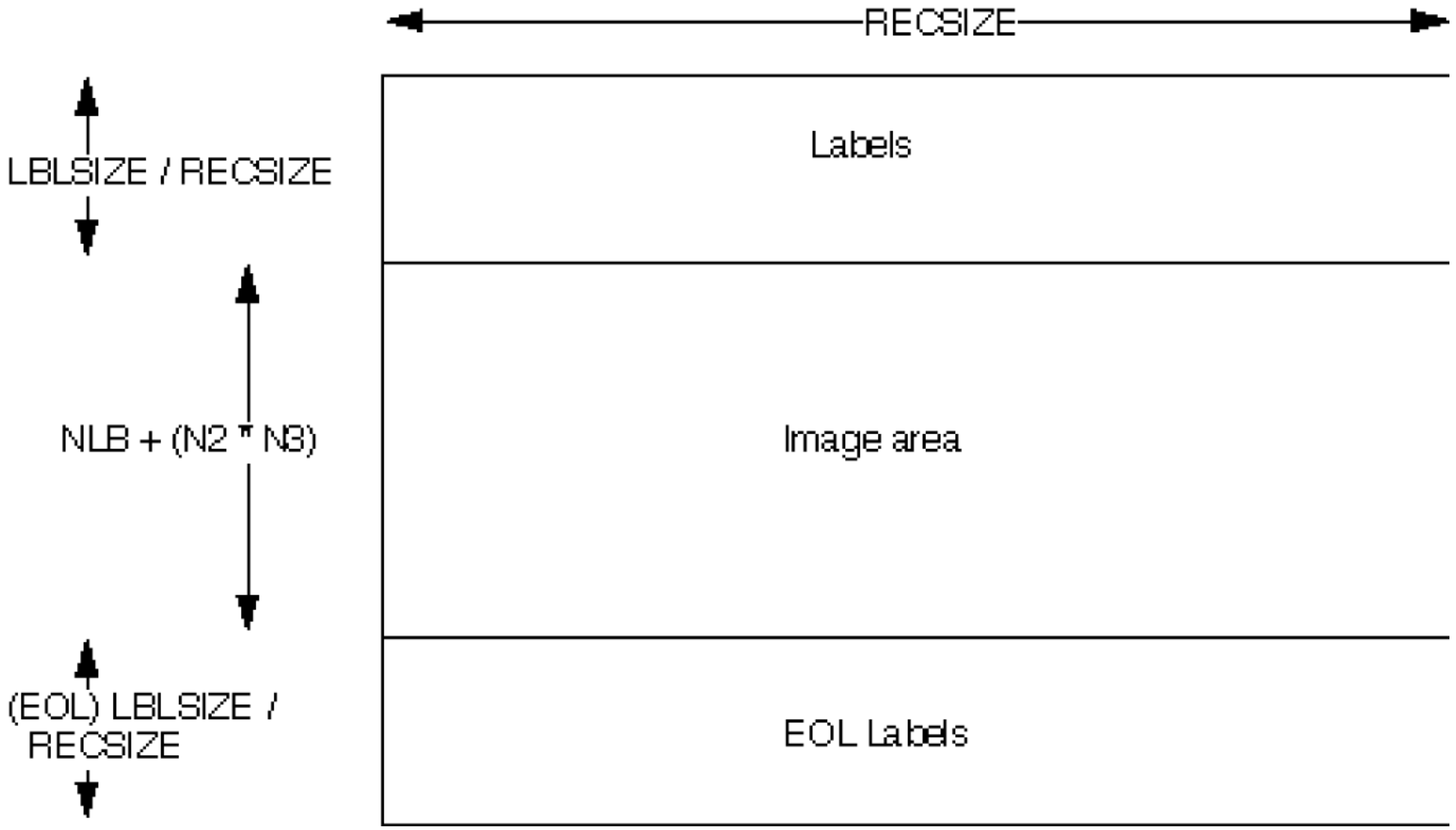
```
H0000_0000_ND2.IMG      ND = nadir
H0000_0000_S12.IMG      S1 = stereo1
H0000_0000_S22.IMG      S2 = stereo2
```

processing level
(Level2)

# miniVICAR - history

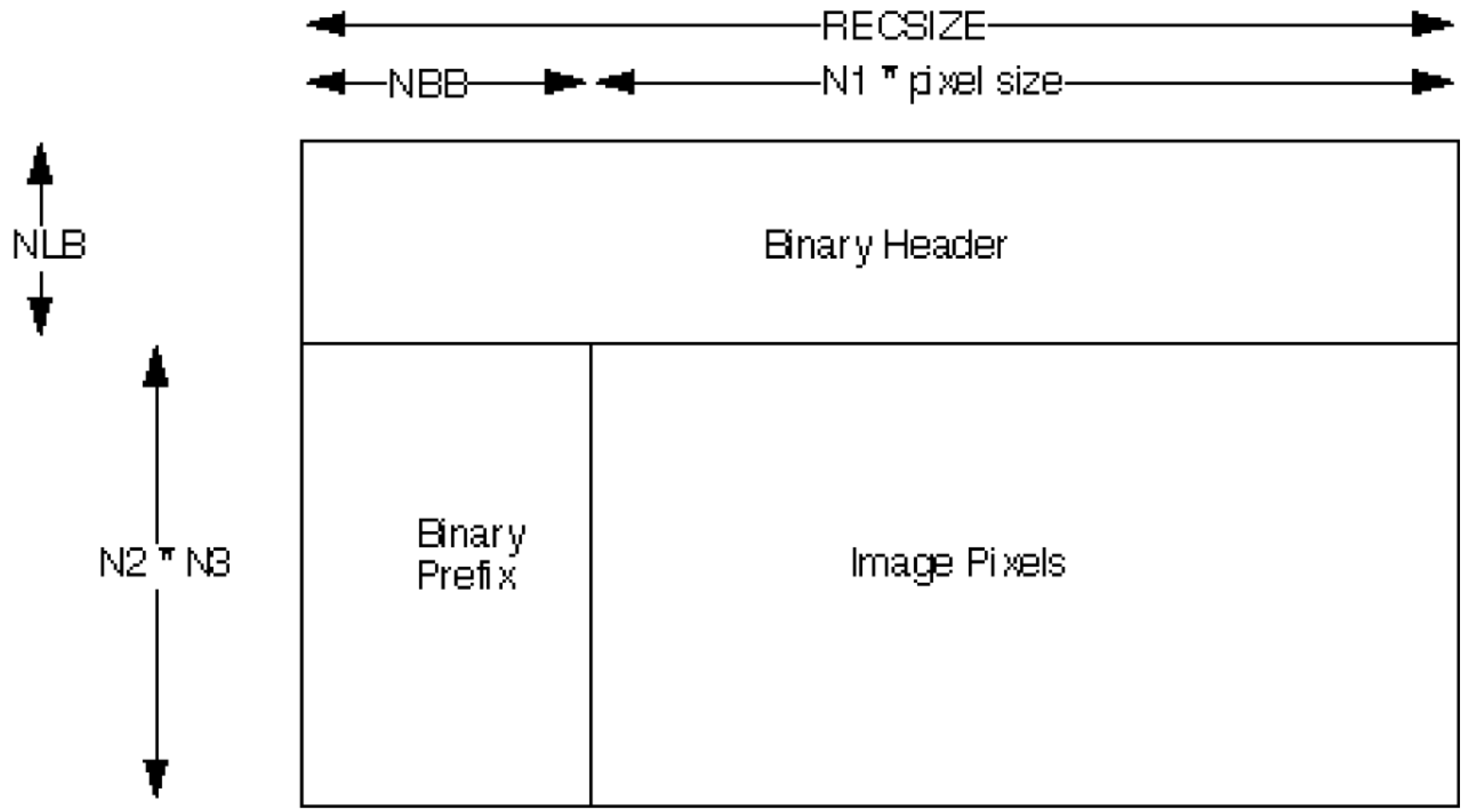
- Implemented in 1966 at the Jet Propulsion Laboratory to process image data produced by the planetary exploration program
- Originally designed for operation with the IBM 360/44 Programming System (44PS)
- Ported to IBMOS/360, VAX-VMS, and Alpha-VMS
- Now running under Sun-Solaris and x86-Linux

# miniVICAR



| | |
|---|---|
| NLB | Number of Lines of binary header |
| N2 | Number of Images Lines (for BSQ) |
| N3 | Number of Bands (for BSQ) |

# miniVICAR



| | |
|---|---|
| NLB | Number of Lines of Binary header |
| NBB | Number of Bytes of Binary prefix |
| N1 | Number of Image Samples (for BSQ) |
| N2 | Number of Images Lines (for BSQ) |
| N3 | Number of Bands (for BSQ) |

# Data types

-BYTE: unsigned single byte, 0 / 255

-HALF: signed two byte integer, -32768  / 32767

-FULL: signed four byte integer, -2147483648 / 2147483647

-REAL: four byte real

-DOUB: eight byte real

-COMP: ever used ?

# miniVICAR programs

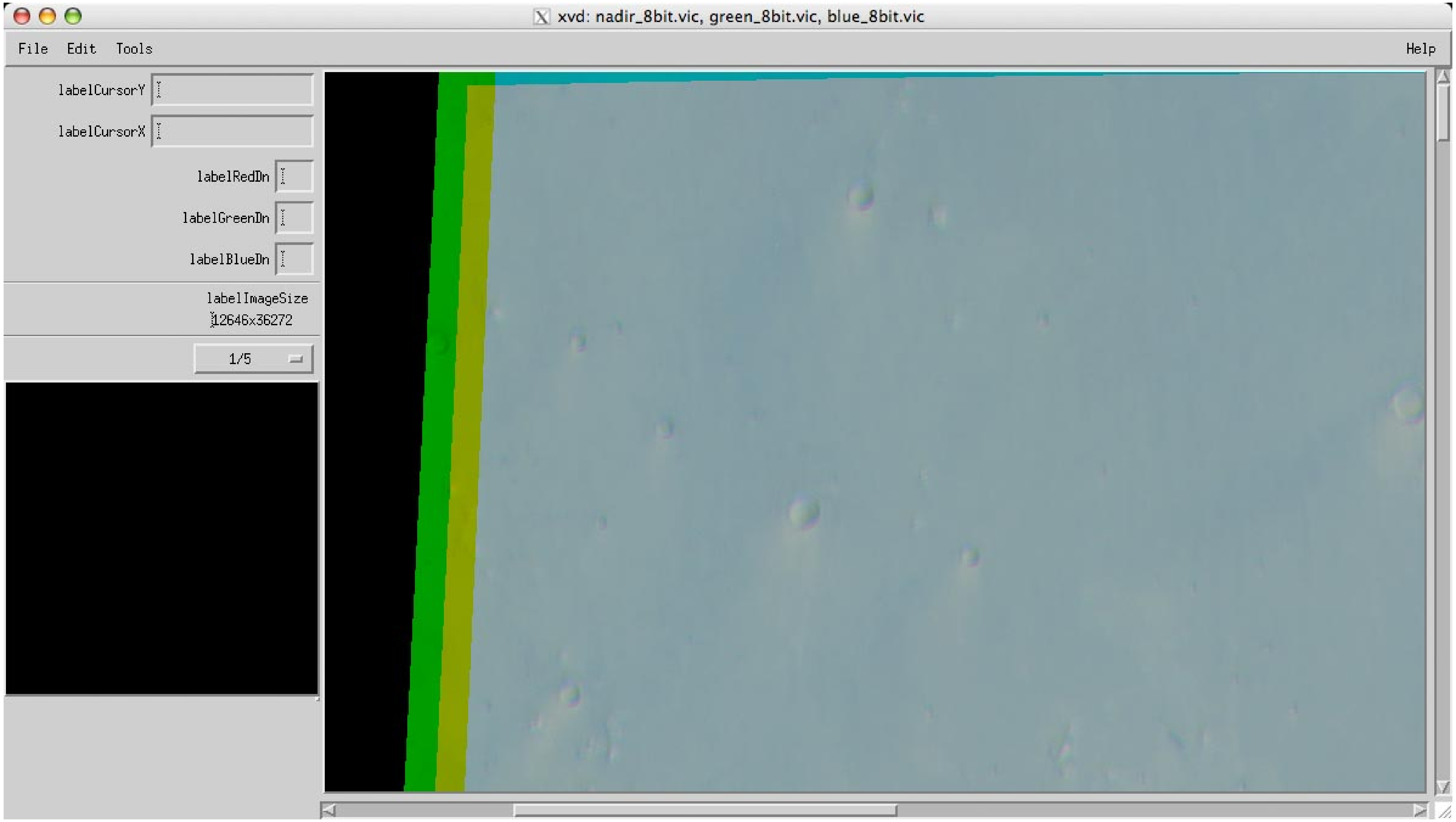
- Every program consists of an executable and a PDF(program definition file),
- Must be in the same directory
- PDF is an ASCII file and contains detailed help for the program
- Software distributed as source code
- Executables available from PSA

ftp://gorilla.estec.esa.int/pub/projects/workshop/04_MEX_DW_june_2007/software_data/minivicar

# General programs

- xvd: PDS and VICAR display program
- Label: prints and modifies the labellabel -list  test.dat
- label -remove test.dat test.rawlabel -create test.raw test2.dat nl=100 ns=50 -ha
- dlrvic2png: converts VICAR to PNG(Problem:
- VICAR signed 16-bit, PNG unsigned 16-bit)

# xvd



labelImageSize
12646x36272

1/5

# hrfill

Syntax:

```
$HWLIB/hrfill inp=... out=... [optionals]
```

- This program changes the DN value of all pixels which were flagedas bad during the calibration process

- hrfill tries to fill the pixels with useful values:= Maximum_of_valid_pixels+1

- For the saturated pixels= Median of the surrounding valid pixels otherwise

- Output will be written only if something has to be filled !

# dlrto8

Syntax:

```
$HWLIB/dlrto8 inp=... out=... [optionals]
```

- dlr12to8 uses image histograms of signed 16-bit integerimages to derive look-up tables for the conversion into 8-bit images using a linear transformation.

- The lower and upper borders of the input data range canbe defined using

  a) explicitevalues (DNMIN, DNMAX)

  b) percentages of image pixels (LOWER, UPPER).

- DNs lower than the LOWER or higher than the UPPER percentages of all pixels are ignored in defining the transformation function.

# hrortho

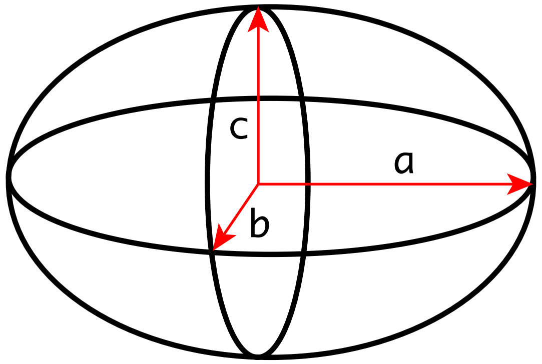
Syntax:

```
$HWLIB/hrortho inp=... ori=spice dtm=... out=... [optionals]
```

| | |
|---|---|
| dtm=... | dtm-file or height above datum in meter |
| inp=... | Input image used if |
| out=... | Output image generated |
| ori=spice | (spice KERNELS used) |
| fitto=... | File to which OUT should fit. |
| sL_inp=... | starting lines of input Level2 image |
| nl_inp=... | no. lines starting from nl_inp to be processed |

“hrortho” produced map projected HRSC images

# hrortho



```
mp_type=...   type of output projection
outmax=...    size limit for output image [in MegaByte]
a_axis=...    value of the a-axis of a solar system body
b_axis=...    value of the b-axis of a solar system body
c_axis=...    value of the c-axis of a solar system body
     ( DEFAULT a_axis=b_axis=3396.19 c_axis=3376.2 )
mp_scale=... measured in kilometers per pixel
cen_lat=...   reference latitude for certain map projections
cen_lon=...   reference longitude for certain map projections
```

For other parameters, have a look at:

`$V2TOP/hw/lib/x86-linux/hrortho.pdf`

(it's ASCII file,  NOT an Adobe .pdf)

# frameortho

Syntax:

```
$HWLIB/frameortho inp=... out=... [optionals]
```

- map projection program for frame sensor data
- input are image, SPICE kernels and geometriccalibration files
- user can specify:
    - map resolution
    - map projection type
    - center latitude and center longitude
    - many other parameters

# SPICE kernels to use

- IK, FK, PCK, SCLK

  latest version

- SPK and CK :

  - Check START_TIME in image label

  - select corresponding Kernel

# Also in miniVICAR

- vicar
  - subset of VICAR from MIPL/JPL
  - DLR programs to process HRSC data
- geocal:
  - Geometric calibration files for line sensors
- data:
  - global Mars-DTM derived from MOLA data to be used as reference body in *ortho (default is the IAU ellipsoid)

# Enviromental Variables

```
#!/bin/tcsh

# MINIVICAR VARIABLES
setenv V2TOP /<PATH>/minivicar/vicar
source $V2TOP/vicset1.csh
source $V2TOP/vicset2.csh
setenv M94GEOCAL $V2TOP/../GEOCAL
setenv PATH $V2TOP:{$PATH}

# VARIABLES FOR KERNELS AND DATA
setenv LEAPSECONDS $V2TOP/../kernels/NAIF0008.TLS
setenv CONSTANTS $V2TOP/../kernels/PCK00008.TPC
setenv SUNKER $V2TOP/../kernels/DE405S.BSP
setenv HWSPICE_TF $V2TOP/../kernels/MEX_V08.TF
setenv HWSPICE_TI $V2TOP/../kernels/MEX_HRSC_V03.TI
setenv HWSPICE_TSC $V2TOP/../kernels/MEX_070321_STEP.TSC
setenv HWSPICE_BC ./ATNM_P030602191822_00135.BC
setenv HWSPICE_BSP ./ORMM__050301000000_00117.BSP
```