```
#THIS EXAMPLE SHOWS THE PROCESS FOR USING ISIS TO CREATE A
COLOR-RATIO IMAGE FROM DAWN IMAGES OF VESTA
#CODE WRITTEN BY CHRISTOPHER LUKE HAWLEY AND MODIFIED BY JULIA
DE LEON

# TO MAKE ISIS3 WORK, YOU HAVE TO WRITE THE FOLLOWING LINES:

echo $SHELL

# FOR BASH SHELL:

export ISISROOT="/net/japon/scratch/ISIS3/isis"
source ${ISISROOT}/scripts/isis3Startup.sh


# FOR TCSH SHELL:

setenv ISISROOT "/net/japon/scratch/ISIS3/isis"
source ${ISISROOT}/scripts/isis3Startup.csh

============================================================

# LETS START!!!!!

#THIS CREATES A BASE NAME LIST OF FILES, WITH NO EXTENSIONS

ls -1 *.IMG | sed 's/\.IMG//' > base.lis

# NOW GET THE APPROPRIATE FORMAT TO BE USED WITH ISIS3:

dawnfc2isis -batchlist=base.lis from=\$1.IMG to=\$1.cub

# KNOW YOUR DATA: READ THE HEAD OF ONE IMAGE:

catlab from=FC21B0007101_11273005958F3G.cub to=ImageLabel.txt

# ASSIGN THE DEM SHAPEMODEL AND ASSOCIATED PCK (Planetary
Constants Kernel). spiceinit ADDS CAMERA POINTING AND
SPACECRAFT INFO TO AN IMAGE CUBE:

spiceinit -batchlist=base.lis from=\$1.cub shape=user
model=vesta_gaskell_512_110825_dem.cub

# KNOW YOUR DATA. WRITE CAMERA INFORMATION FROM A SINGLE POINT
LOCATION

campt from=FC21B0007101_11273005958F3G.cub

# KNOW YOUR DATA. CREATE A TEXT FILE WITH GEOMETRY STATISTICS
```

```
camstats -batchlist=base.lis from=\$1.cub attach=true linc=10
sinc=10 to=Camstats.txt

# HAVE A LOOK AT YOUR DATA

qview FC21B0007101_11273005958F3G.cub &

# WE NOW CREATE A LIST OF ALL THE IMAGES THAT WILL BE USED BY
mosrange:

ls *G.cub > range.lis

# AND WE EXECUTE mosrange TO CREATE A MAP THAT WILL BE USED TO
MAP PROJECTING THE REST OF THE IMGES

mosrange fromlist=range.lis to=equi.map
projection=equirectangular

# THE MAP IS USED TO PROJECT THE FIRST IMAGE, AND THEN THE
REST OF THE IMAGES ARE PROJECTED TO THE FIRST ONE:

cam2map from=FC21B0007101_11273005958F3G.cub
to=FC21B0007101_11273005958F3G-CO-EQ.cub map=equi.map
pixres=map lonseam=continue

cam2map from=FC21B0007102_11273010009F4G.cub
to=FC21B0007102_11273010009F4G-EQ.cub
map=FC21B0007101_11273005958F3G-CO-EQ.cub matchmap=true

cam2map from=FC21B0007106_11273010035F8G.cub
to=FC21B0007106_11273010035F8G-EQ.cub
map=FC21B0007101_11273005958F3G-CO-EQ.cub matchmap=true

# COMPARE THE NON-PROJECTED AND THE PROJECTED IMAGES:

qview FC21B0007102_11273010009F4G.cub
FC21B0007102_11273010009F4G-EQ.cub &


# USING coreg WE ASSURE THAT EACH FILTER IS CORRECTLY ALIGNED

coreg from=FC21B0007102_11273010009F4G-EQ.cub
match=FC21B0007101_11273005958F3G-CO-EQ.cub
to=FC21B0007102_11273010009F4G-CO.cub deffile=coreg.def
onet=FC21B0007102_11273010009F4G.co.net transform=translate
interp=bilinear rows=32 columns=32

coreg from=FC21B0007106_11273010035F8G-EQ.cub
match=FC21B0007101_11273005958F3G-CO-EQ.cub
to=FC21B0007106_11273010035F8G-CO.cub deffile=coreg.def
```

```
onet=FC21B0007106_11273010035F8G.co.net transform=translate
interp=bilinear rows=32 columns=32


# photomet PHOTOMETRICALLY CORRECTS EACH IMAGE BASED OFF OF
PUBLISHED VALUES FOR EACH FILTER

photomet -batchlist=base.lis from=$(sed -n '3p' base.lis)-
CO.cub to=$(sed -n '3p' base.lis)-pho.cub phtname=hapkehen
theta=18.1 wh=0.408 hg1=0.233 hg2=1 hh=0.07 b0=1.7
zerob0standard=false normname=albedo incref=30.0 incmat=0.0
thresh=10e30 albedo=1.0 USEDEM=TRUE

photomet -batchlist=base.lis from=$(sed -n '2p' base.lis)-
CO.cub to=$(sed -n '2p' base.lis)-pho.cub phtname=hapkehen
theta=18.7 wh=0.377 hg1=0.231 hg2=1 hh=0.07 b0=1.7
zerob0standard=false normname=albedo incref=30.0 incmat=0.0
thresh=10e30 albedo=1.0 USEDEM=TRUE

photomet -batchlist=base.lis from=$(sed -n '1p' base.lis)-CO-
EQ.cub to=$(sed -n '1p' base.lis)-pho.cub phtname=hapkehen
theta=17.6 wh=0.534 hg1=0.222 hg2=1 hh=0.07 b0=1.7
zerob0standard=false normname=albedo incref=30.0 incmat=0.0
thresh=10e30 albedo=1.0 USEDEM=TRUE

# WE CREATE A LIST WITH THE PHOTOMETRICALLY CORRECTED IMAGES
AND PERFORM SOME DIVISIONS USING algegra:

ls *pho.cub > pho.lis

algebra from=$(sed -n '1p' pho.lis) from2=$(sed -n '3p'
pho.lis) to=750over430.cub operator=divide
algebra from=$(sed -n '1p' pho.lis) from2=$(sed -n '2p'
pho.lis) to=750over900.cub operator=divide
algebra from=$(sed -n '3p' pho.lis) from2=$(sed -n '1p'
pho.lis) to=430over750.cub operator=divide

# WE THEN PUT THE RATIO IMAGES INTO ONE SINGLE CUBE AND TRIM
THE AREAS THAT DO NOT OVERLAP:

ls *over*.cub > Ratset.lis

cubeit from=Ratset.lis to=Phoset.cub

bandtrim from=Phoset.cub to=Oppia_Ratio.cub
```