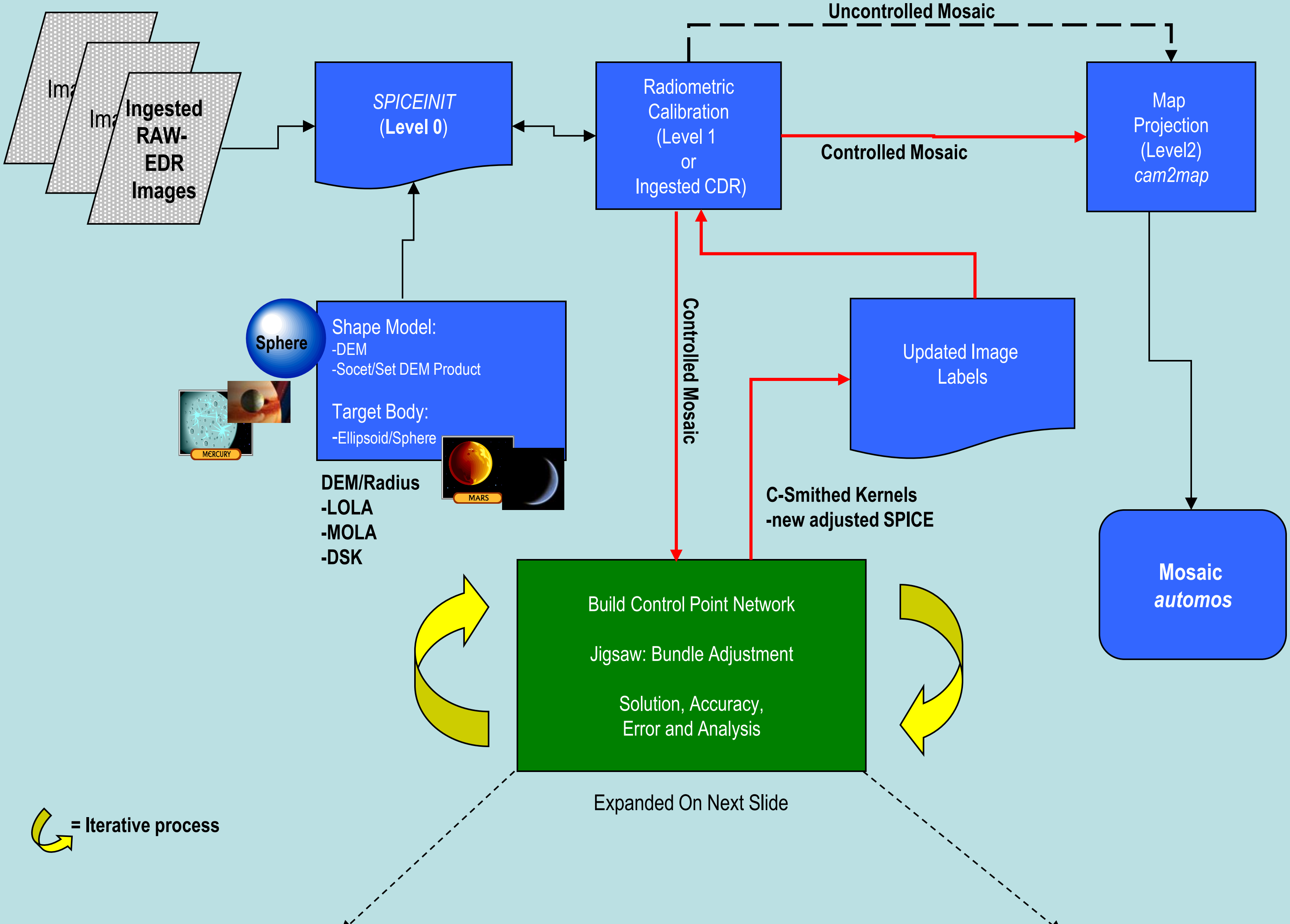
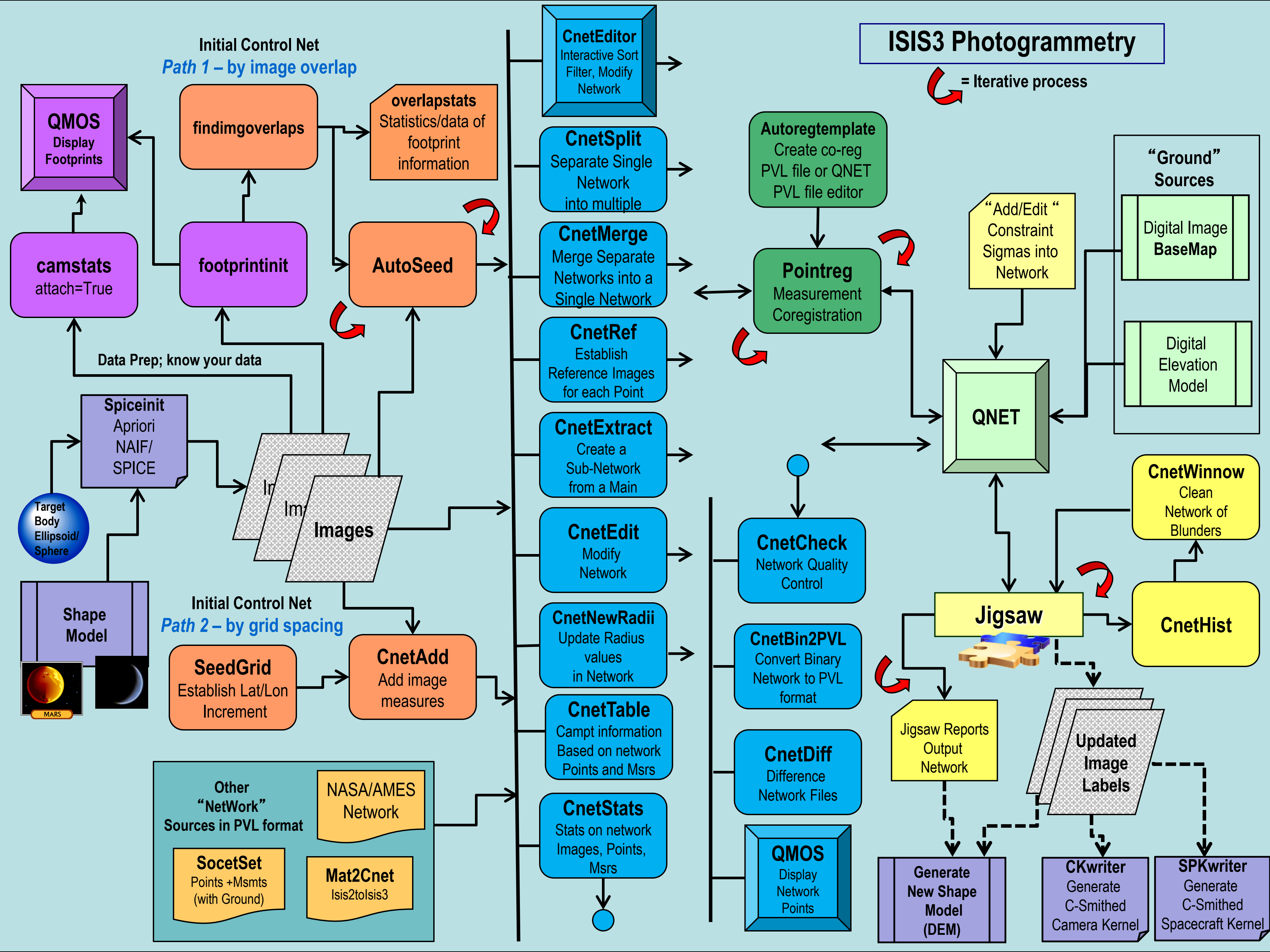
ISIS3 Controlled Map Product Development




ISIS3 Photogrammetry
= Iterative process
Initial Control Net
Path 1 – by image overlap
QMOS
Display
Footprints
findimgoverlaps
overlapstats
Statistics/data of footprint information
camstats
attach=True
footprintinit
AutoSeed
Data Prep; know your data
Spiceinit
Apriori
NAIF/
SPICE
Target Body Ellipsoid/ Sphere
Images
Shape Model
MARS
Initial Control Net
Path 2 – by grid spacing
SeedGrid
Establish Lat/Lon Increment
CnetAdd
Add image measures
Other "NetWork" Sources in PVL format
NASA/AMES Network
SocetSet
Points +Msmts (with Ground)
Mat2Cnet
Isis2toIsis3
CnetEditor
Interactive Sort Filter, Modify Network
CnetSplit
Separate Single Network into multiple
CnetMerge
Merge Separate Networks into a Single Network
CnetRef
Establish Reference Images for each Point
CnetExtract
Create a Sub-Network from a Main
CnetEdit
Modify Network
CnetNewRadii
Update Radius values in Network
CnetTable
Campt information Based on network Points and Msrs
CnetStats
Stats on network Images, Points, Msrs
Autoregtemplate
Create co-reg PVL file or QNET PVL file editor
Pointreg
Measurement Coregistration
"Add/Edit " Constraint Sigmas into Network
"Ground" Sources
Digital Image BaseMap
Digital Elevation Model
QNET
CnetCheck
Network Quality Control
CnetBin2PVL
Convert Binary Network to PVL format
CnetDiff
Difference Network Files
QMOS
Display Network Points
CnetWinnow
Clean Network of Blunders
Jigsaw
CnetHist
Jigsaw Reports Output Network
Updated Image Labels
Generate New Shape Model (DEM)
CKwriter
Generate C-Smithed Camera Kernel
SPKwriter
Generate C-Smithed Spacecraft Kernel

Initiating a control network involves a choice of two paths

FIRST: Know your data!

- *camstats*
  - camera statistics for a single image or all images
  - *attach=true* for *qmos*
  - Increase *linc* & *sinc* to 10 to 100 the default = 1 (slow)
- *campt*
  - geometric and photometric information for a single pixel location
  - reports additional information camstats does not
    - For example, azimuth angles (spacecraft and solar)
- *footprintinit - create polygons from images for qmos*
- *qmos* – displays image footprints in a map projection and shows areas of overlap and basic camera statistics
- *qview* - image quality, pointing quality, overlap amounts
- *mosrange* – computes latitude/longitude range and resolution range for all images and reports in map template format
- *caminfo* - good for a large amount of images. camera statistics, geometry information, DN statistics, label and polygon information (in WKT format) for a single image or all images

Familiarity with your data regarding…

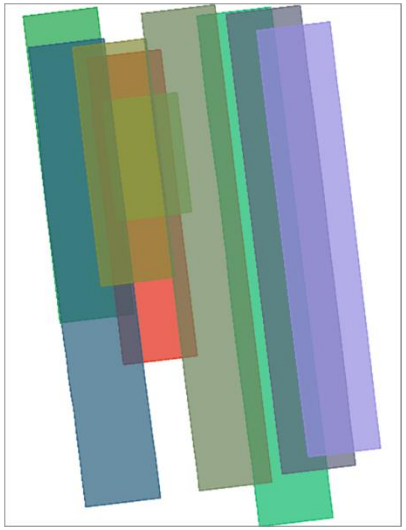
1) how the images overlap
2) range of resolution
3) viewing & illumination conditions

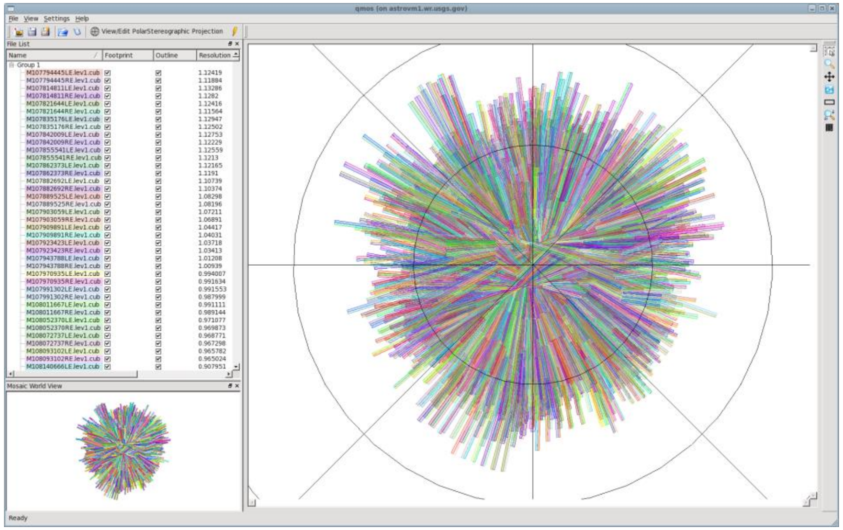
will help determine the best path for establishing your network…

by image overlaps (*path1*) and/or
by a gridded control point distribution (*path2*)

**There is no right or wrong method, there could be a <u>combination</u> of the paths*

Initiating a control network involves a choice of two paths





CTX Footprints
Candidate for **either path**

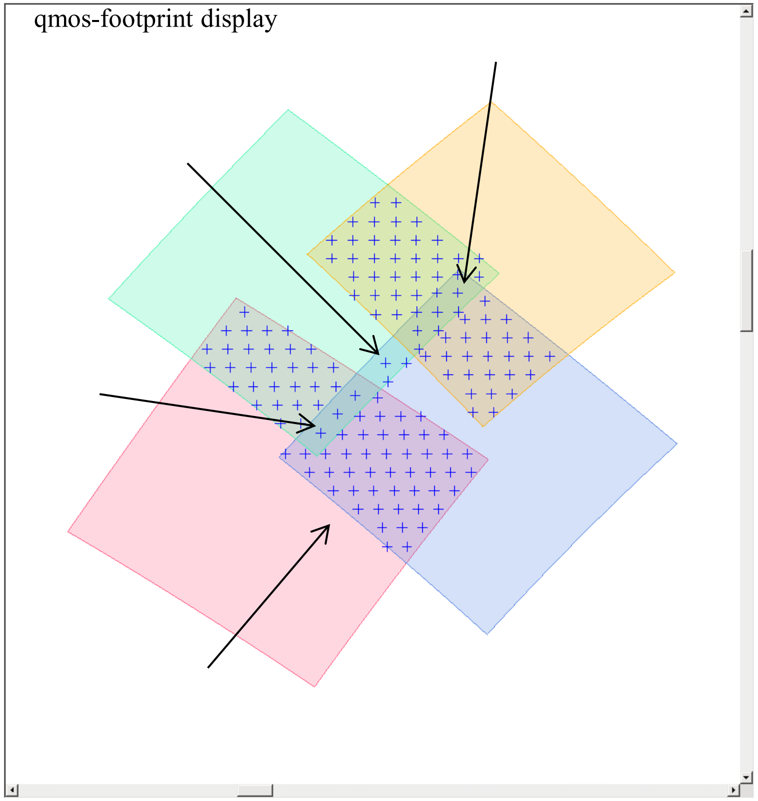
Messenger Flyby Footprints:
Candidate for ***autoseed*** path

LROC NAC Polar Footprints
Candidate for ***seedgrid*** path

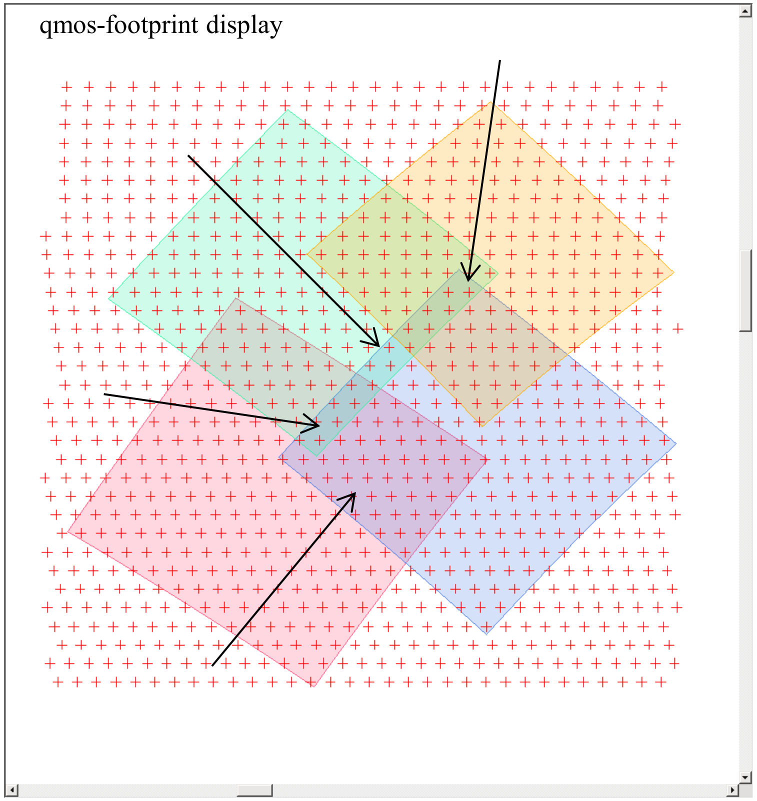
Starting a Control Network from scratch with Image Overlaps v.s. Gridded Points

The initial goal is to have a sufficient distribution of control point image measures across and through the image overlaps and across your images where possible

findimageoverlaps
autoseed *– based on your image overlaps*

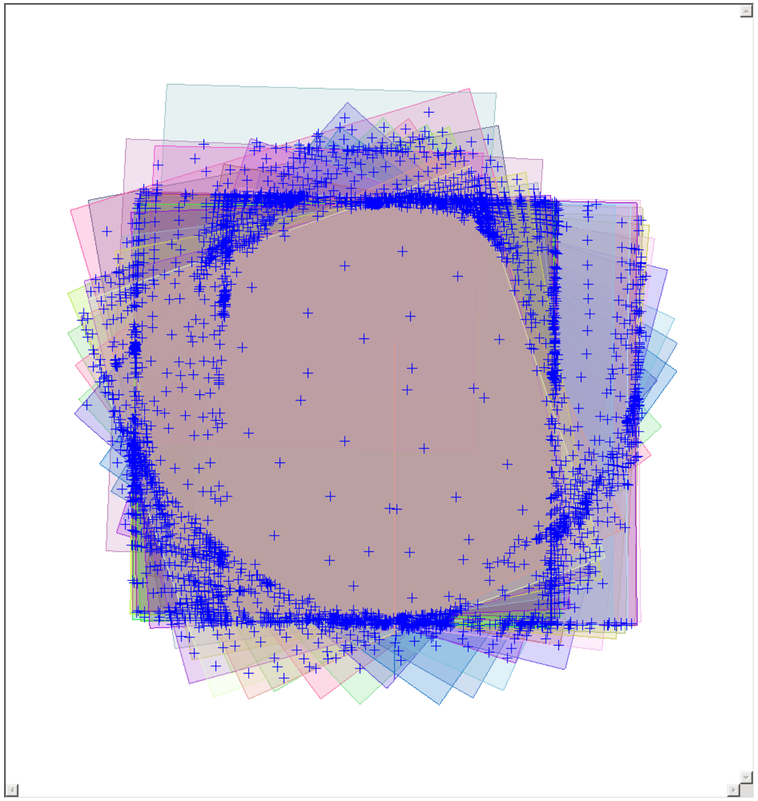


seedgrid *– doesn't know a thing about your images, only the lat/lon/spacing*



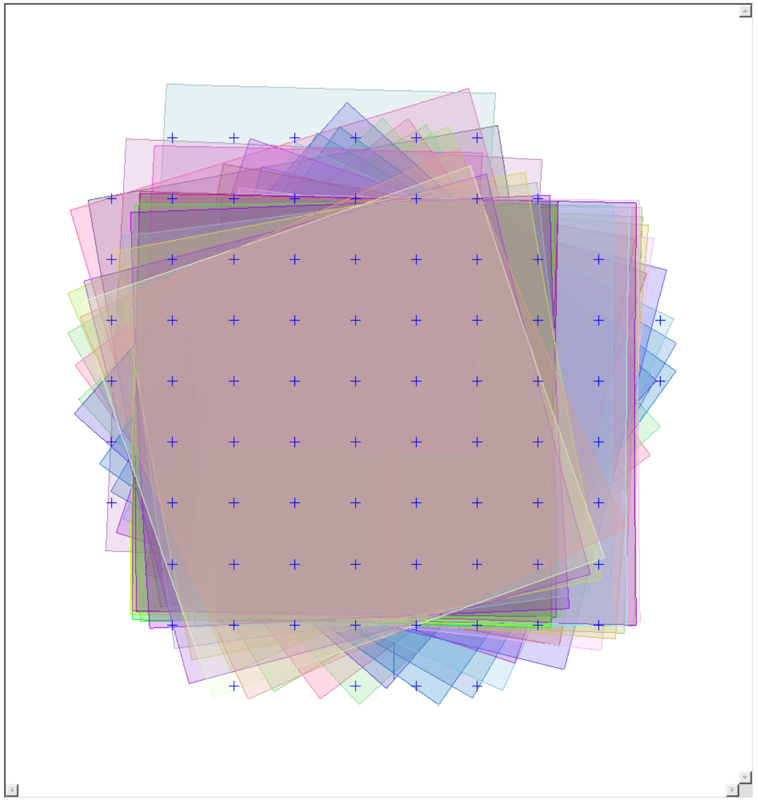
findimageoverlaps
autoseed

**Has the potential to see every single overlap polygon*



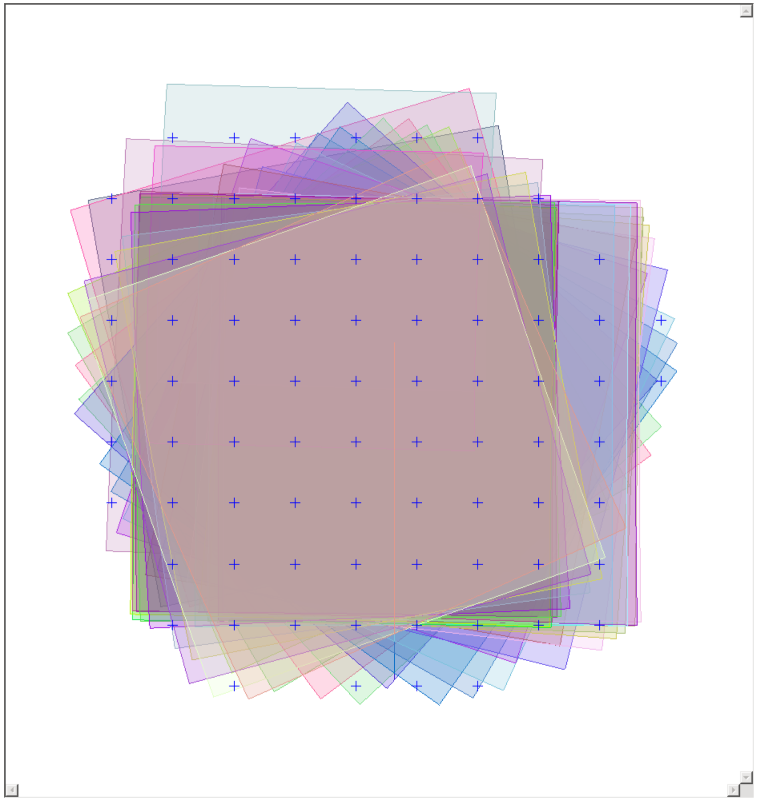
seedgrid map=spolar_projection.map
cnetadd retrieval=point extract=modified

**Every image in the stack needs a good distribution*

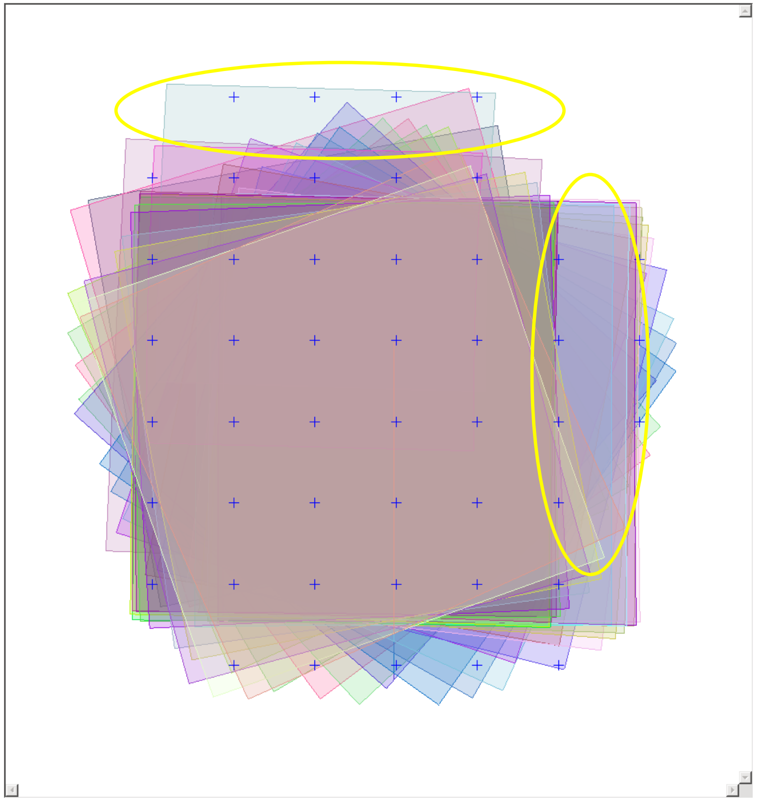


- X/Y Spacing starting point
  - Determined by resolution(s), overlap area(s) and desired number of points in overlaps
- Try to strike a balance with the number of points seeded
  - Too many points
    - Increases program runtime
    - Increases potential number of registration false positives
  - Too few points
    - May result in poorly connected the data
    - May result in more manual point picking than desired

seedgrid
with moderate X/Y Spacing

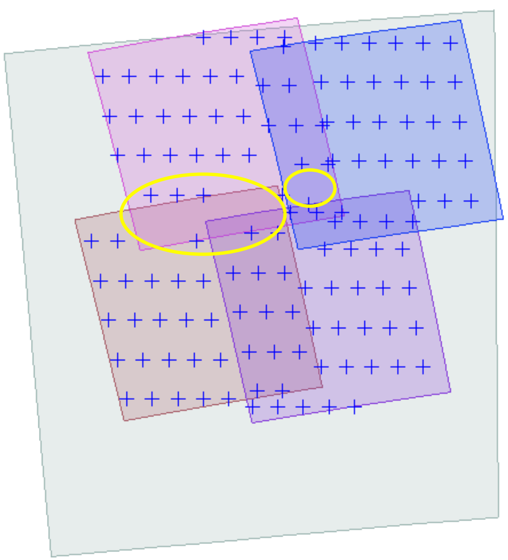


seedgrid
with less dense X/Y Spacing



findimageoverlaps
autoseed – based on your image overlaps

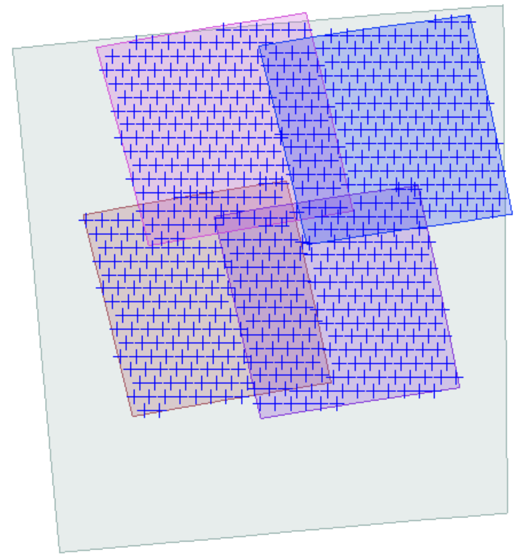
Spacing too sparse for small overlaps



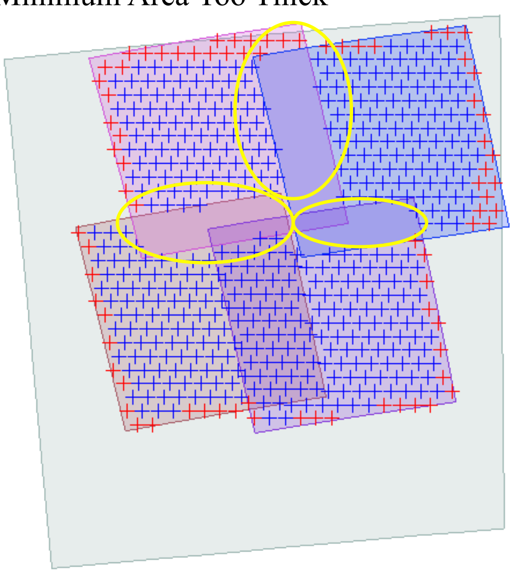
:::::::::::::
grid.def
:::::::::::::

```
Group = PolygonSeederAlgorithm
  Name             = Grid
  MinimumThickness = 0.1
  MinimumArea      = 56000
  XSpacing         = 15000  #40000
  YSpacing         = 15000  #40000
End_Group
```

Spacing a little too dense maybe?



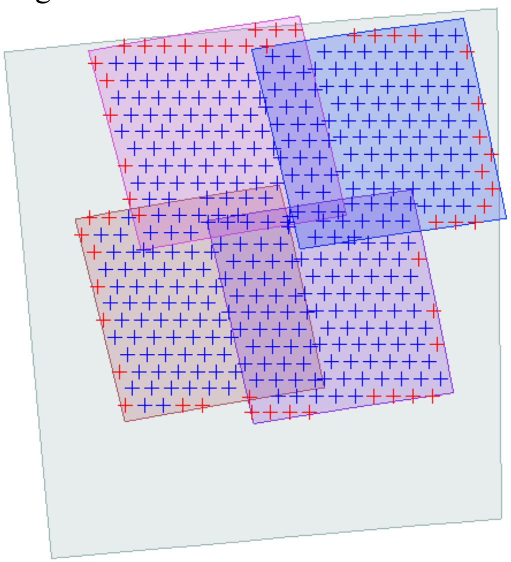
Minimum Area Too Thick



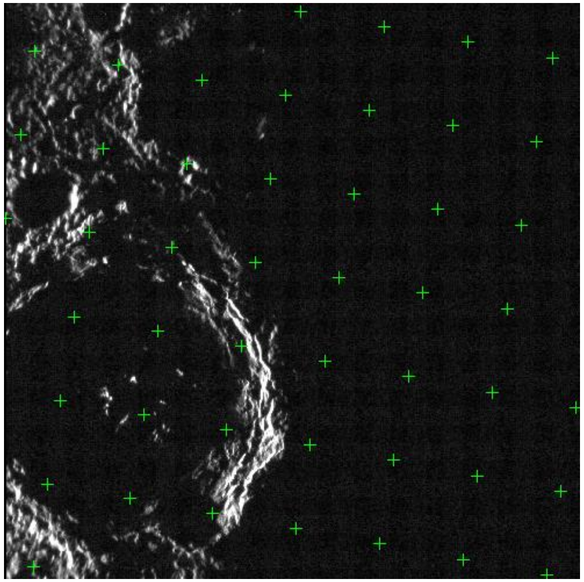
:::::::::::::
grid.def
:::::::::::::

```
Group = PolygonSeederAlgorithm
  Name             = Grid
  MinimumThickness = 0.3
  MinimumArea      = 56000
  XSpacing         = 25000
  YSpacing         = 25000
  PixelsFromEdge = 35
End_Group
```

Just right



- Avoiding undesirable conditions for measurements (shadows, limb, terminator, edges, no data, etc.)
  - Use valid measure *deffile* and desired optional constraints
    - $ISIS3DATA/base/templates/cnet_validmeasure/validmeasure.def
    - Can be applied anytime in an application that offers a ‘deffile’ parameter
      - cnetedit deffile=valid.def
      - *findimageoverlaps/autoseed:* autoseed deffile=valid.def
      - *seedgrid/cnetadd:* cnetadd deffile=valid.def



```
Group = ValidMeasure
   MinDN = 0.001
   MaxDN = double
   MinEmission = double  (limb/oblique criteria)
   MaxEmission = double
   MinIncidence = double (terminator/shadow)
   MaxIncidence = 75  (degrees)
   MinResolution = double
   MaxResolution = double
   PixelsFromEdge = integer OR MetersFromEdge = double
EndGroup
```

In the picture above, the green “**+**” marks are considered valid point locations.

Many of the point locations fall in deep crater shadows and terminator region.

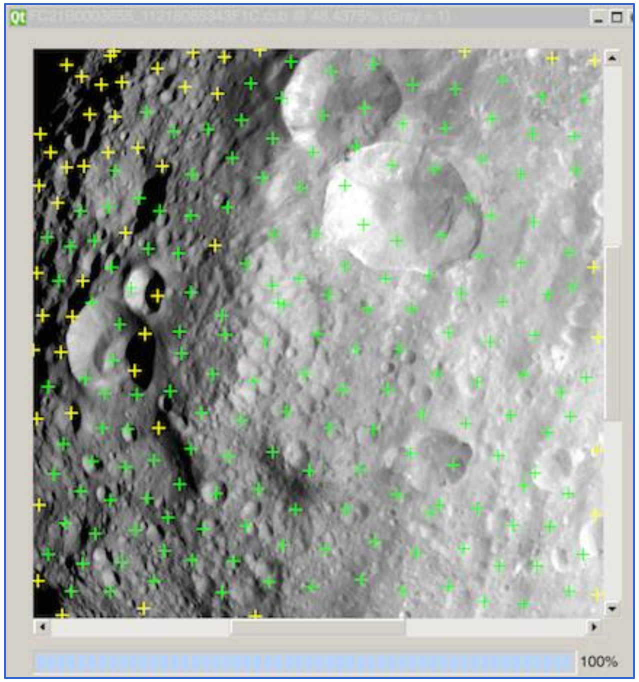
To identify these points as invalid (Ignore=true), constraints can be set for image conditions at these locations.

Possible constraints are in **Red** in the valid measure definition example above.

- ➢ Avoiding undesirable conditions for measurements (shadows, limb, terminator, edges, no data, etc.)
  - Use valid measure *deffile* and desired optional constraints $ISIS3DATA/base/templates/cnet_validmeasure/validmeasure.def
    - Can be applied anytime in an application that offers a ‘deffile’ parameter
      - cnetedit deffile=valid.def
      - *findimageoverlaps/autoseed:* autoseed deffile=valid.def
      - *seedgrid/cnetadd:* cnetadd deffile=valid.def

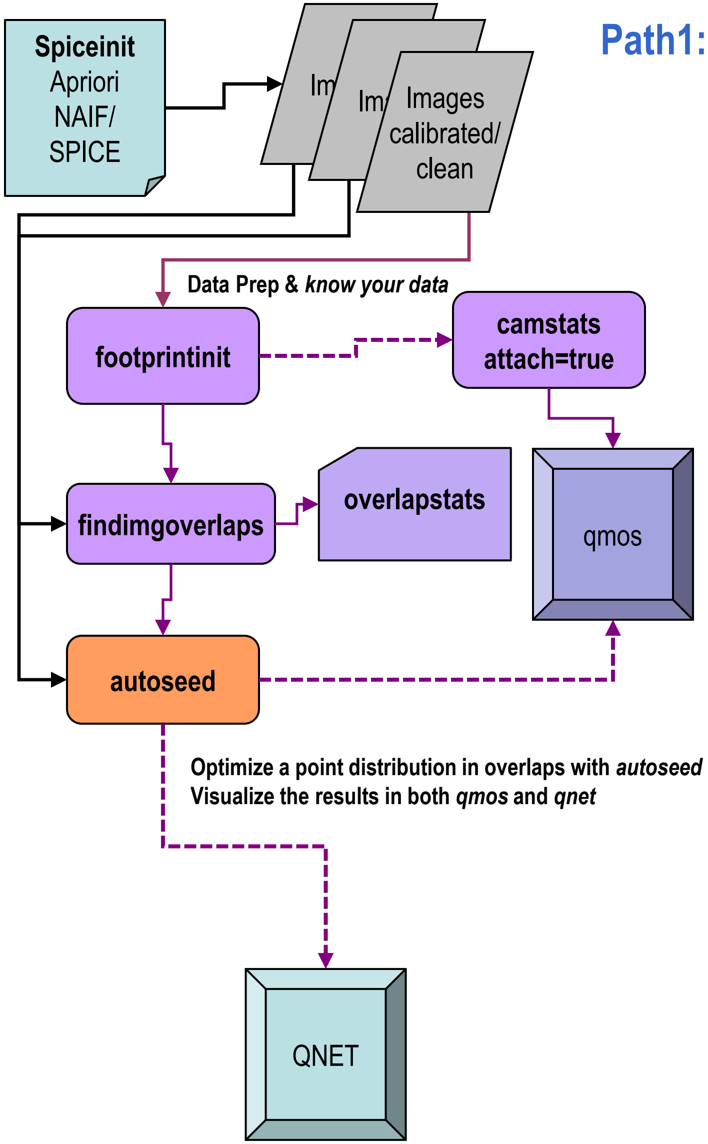
*Avoid control point image measures in the shadows with a **MinDN** and along the edges of the image with **PixelsFromEdge***

- Green “**+**” = Valid Measures;   Yellow “+” = Invalid Measures (Ignore = True)




```
Group = ValidMeasure
   MinDN = 1.5
   MaxDN = double
   MinEmission = double  (limb/oblique criteria)
   MaxEmission = double
   MinIncidence = double (terminator/shadow)
   MaxIncidence = 85
   MinResolution = double
   MaxResolution = double
   PixelsFromEdge = 20
EndGroup
```

Path1: Starting a Control Network from scratch using Image Overlaps



•**footprintinit:** Create footprint polygons for each image
- Provides footprint polygons for *qmos*
  - Required for *findimageoverlaps* & *autoseed*
  - Set *maxemission* and *maxincidence* for limb and terminator data
  - Consider a smaller *linc* and *sinc* for limb, polar or irregularly shaped data
  - Set *increaseprecision=true* for automatic adjustment with *linc* and *sinc* to smaller values if current values result in invalid polygons

•**findimageoverlaps**: Create a binary file of overlapping polygons

•**overlapstats:** Compute overlap polygon statistics
- Input file is the *findimageoverlaps* output
- View output of *overlapstats* for help with setting initial parameter values in the *autoseed* 'deffile'

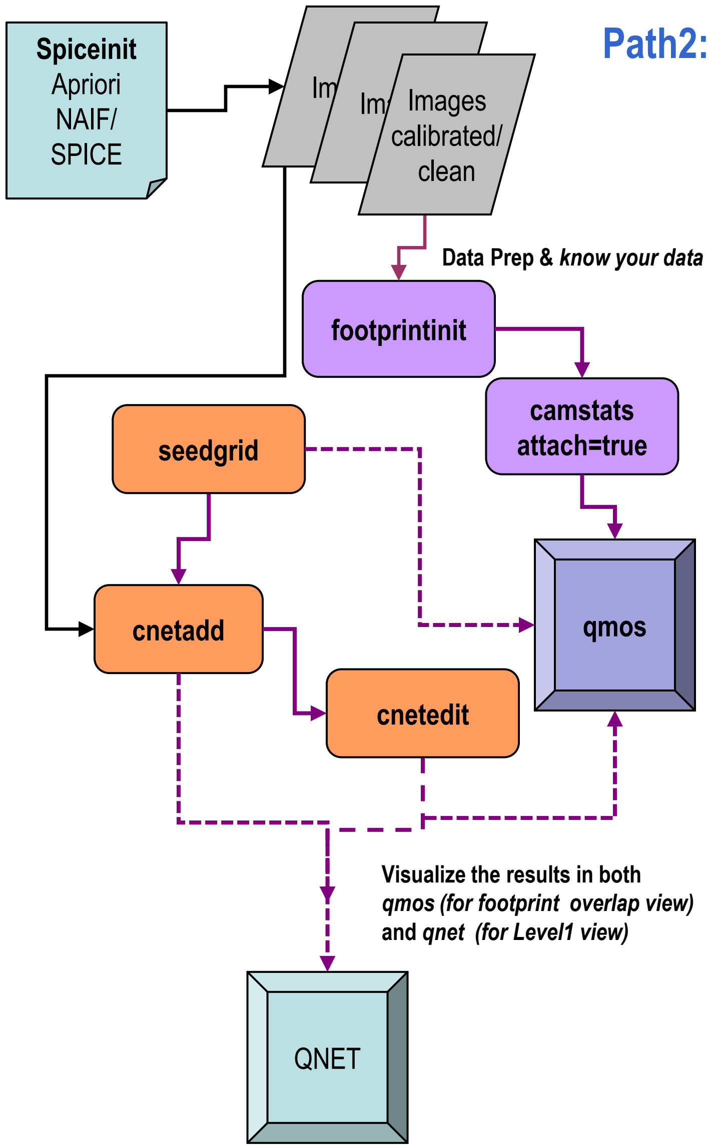
•**autoseed:** Create a distribution of control points across the areas of overlap
- *autoseed* Workshop!!
  - http://isis.astrogeology.usgs.gov/IsisWorkshop/index.php/Autoseed
- *autoseed* ***deffile*** templates
  - cp $ISIS3DATA/base/templates/autoseed/grid.def mygrid.def
  - Set optional 'valid measure' parameters within the *autoseed* deffiile (mygrid.def)
    - $ISIS3DATA/base/templates/autoseed/gridExtraKeywords.pvl
    - Examples-avoid the edges of an image, high emission angles, shadows

Commands:

> footprintinit from= linc= sinc= maxema= maxinc= (use –batchlist)
> findimageoverlaps fromlist= overlaplist=*overlap.dat*
> overlapstats fromlist= overlaplist=*overlap.dat* to=*overlap.csv*
> autoseed fromlist= overlaplist=*overlap.dat* ***deffile=****mygrid.def* onet=*autoseed.net*
pointid="*prefix*_???"

qnet - Load the image list and autoseed.net file; view some number of cubes to evaluate the control point distribution in Level1 (raw camera space)

qmos - Overlay the output autoseed.net file on top of the image footprints to evaluate control point distribution



Path2: Starting a Control Network from Scratch Using Gridded Points

•**footprintinit:** Create footprint polygons for each image (for qmos)

•**seedgrid:** Create a distribution of control points across the areas of overlap
- Establish grid by XY meters or latitude/longitude increment (spacing parameter)
- Note how many control points are created!!
- See Autoseed Worshop for **tips** on setting *spacing*
  - http://isis.astrogeology.usgs.gov/IsisWorkshop/index.php/Autoseed

•Create a 'valid measure' parameter file
- Possible parameters for valid criteria with an image measure are listed in
  - $ISIS3DATA/base/templates/cnet_validmeasure/validmeasure.def.example
- Create your own parameter file to edit
  - cp $ISIS3DATA/base/templates/cnet_validmeasure/validmeasure.def.example *validmsr.def*
- Edit the *validmsr.def* to establish optional criteria
  - Examples-avoid the edges of an image, high emission angles, shadows

•**cnetadd:** Add the image measures
- Based on who overlaps the control point lat/lon's created in *seedgrid*
- Supply the valid measure criteria parameter file (deffile)
  - Any measure added to a control point considered invalid will have Ignore=True

•**cnetedit:** Delete the 'Ignored' points from the network
- NOTE: future application *cnetref* might fail if this step is elminated
  - Potential problem with control points that contain no image measures (no one overlaps that point)

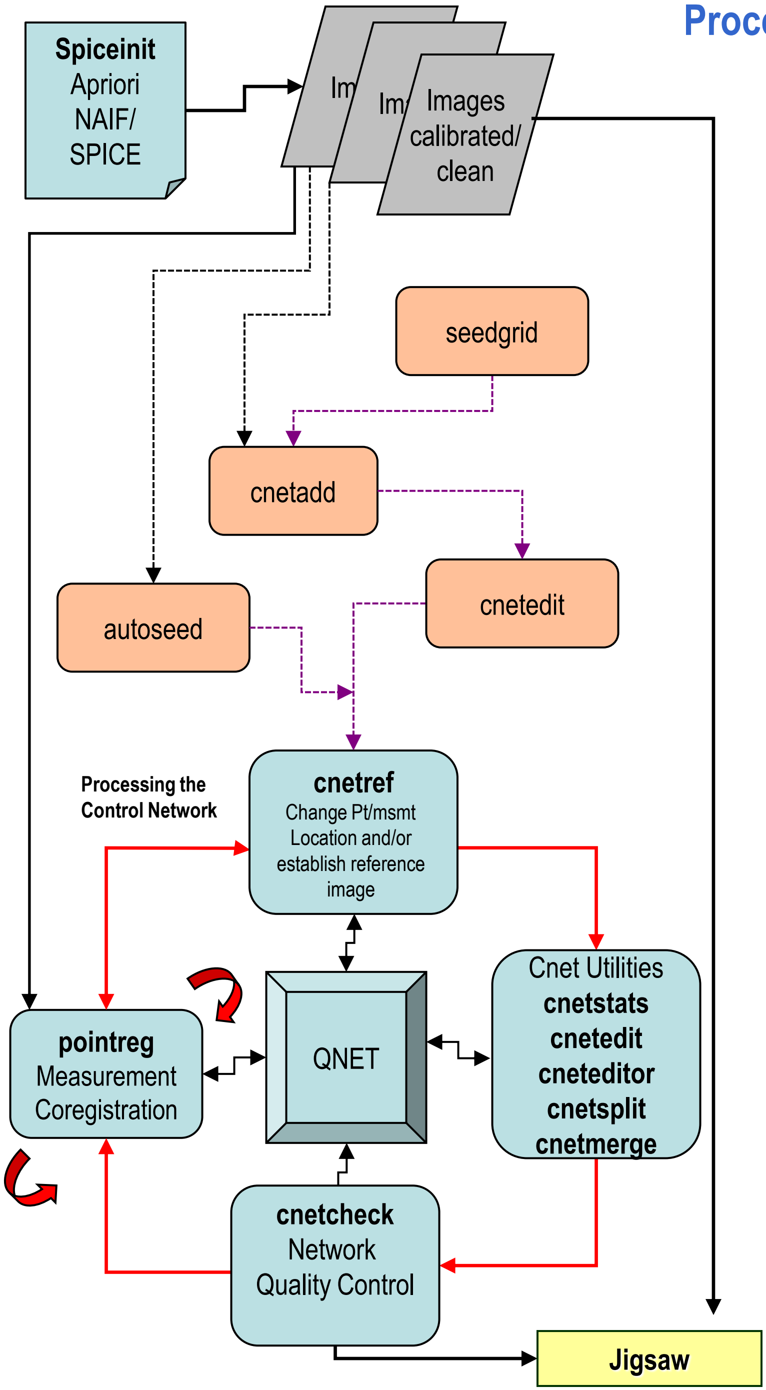
Commands:

> footprintinit from= linc= sinc= maxema= maxinc= (use –batchlist)
> seedgrid target= minlat= maxlat= minlon= maxlon= spacing= desc= pointid="prefix_???"
> cnetadd fromlist= ***deffile***=*validmsr.def* retrieval=point onet=cnetadd.net polygon=true extract=modified
>cnetedit cnet=cnetadd.net onet=cnetadd-edt.net

>qnet - Load the image list and cnetadd-edt.net file; view a number of cubes to evaluate the control point distribution in Level1 (raw camera space)

>qmos - Overlay the output cnetadd-edt.net file on top of the image footprints to evaluate control point distribution

Processing the Control Network



The next goal is to maximize the success and accuracy results of the sub-pixel registration amongst the image measures within each tie-point.

- **cnetref:** Establish the reference images for each control point (Optional-Use with Caution)!
  - The measure (image) within a control point with Reference=True becomes the “pattern” chip in *pointreg* for the remaining measures
  - The Reference measure is implicitly defaulted to the 1st measure encountered within a control point
    - unless it is overridden interactively in *qnet* or *cnetref*
  - cnetref will assign the image measure with Reference=True by a selection of criteria
    - Criteria=emission (avoid the limb; the image measure with the lowest ema = reference)
    - Criteria=incidence (avoid the terminator; the measure with the lowest inc=reference)
    - Criteria = resolution (multiple criteria choices available to select the reference)
  - Can use the same valid measure parameter file that is passed to *cnetadd* or …create/edit
    - cp $ISIS3DATA/base/templates/cnetref/no_operator.def.example ***validmsr.def***
  - Criteria=interest (“nudge” the control point locations to new coordinates in areas of interest)
    - cp $ISIS3DATA/base/templates/cnetref/standard_deviation_operator.def.example ***std.def***

- **pointreg:** Pattern matching amongst the measures within each control point
  - Subpixel registration matches the measures to the reference image
  - Autoreg ***deffile*** template (create your own and edit)
    - cp $ISIS3DATA/base/templates/autoreg/coreg.maxcor.p2020.s5050.def ***reg.def***
  - Pattern Match Workshop and Document
    - http://isis.astrogeology.usgs.gov/documents/PatternMatch/PatternMatch.html
    - http://isis.astrogeology.usgs.gov/IsisWorkshop/index.php/Automatic_Registration

- **cnetedit:** Delete the ‘Ignored’ measures and points

- **cnetcheck:** Used to double check the quality of the network right before jigsaw

- **qnet:** Used through the entire process for visual and manual adjustments

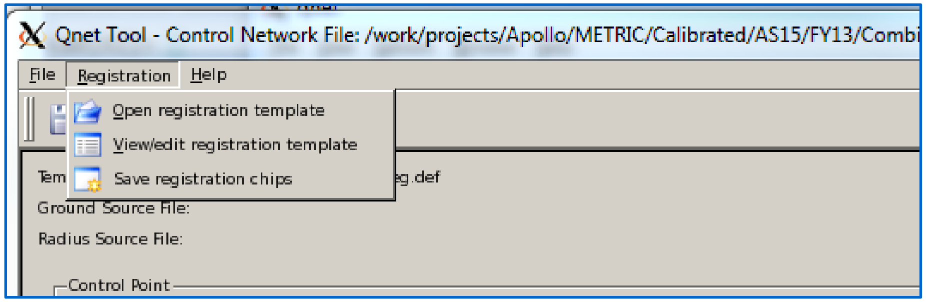
- **jigsaw:** Least squares bundle adjustment-default is to solve for camera angles & twist

Commands:

> cnetref fromlist= cnet= onet=cnetref.net ***deffile=*** criteria= type=
> pointreg fromlist= cnet=cnetref.net onet=pointreg.net ***deffile=reg.def*** points=all measures=all
>cnetedit cnet=pointreg.net onet=pointreg-edt.net
>cnetcheck fromlist= cnet=pointreg-edt.net prefix=pointreg-edt-CHK
> jigsaw fromlist= cnet=pointreg-edt.net onet=jig.net file-prefix=jig

Pattern Matching Image Measures within a Point

- Use *qnet* to
  - Define the *autoreg* parameter file (deffile)
  - View registration 'chips' to iterate and optimize the parameters for automated success
  - Refer to: http://isis.astrogeology.usgs.gov/IsisWorkshop/uploads/8/80/Qnet_demo.pdf



- Pattern Chip
  - The more 'bland' the data is, the larger the Pattern Chip box should be

- Search Chip
  - Should always be larger than Pattern chip
  - The size needs to account for feature offset (misregistration)
    - SearchChipSize = PatternChipSize + (OffsetPixelEstimate * 2)
  - It should be unnecessarily large for the Pattern Chip to avoid a 'false positive' match

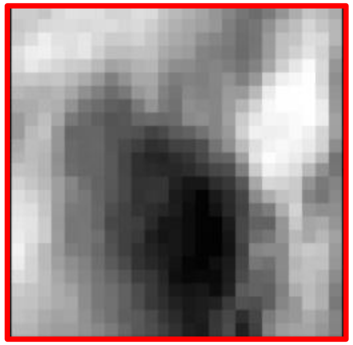
Pattern Matching Image Measures within a Point

- Dimensions
  - The overall shape of Search and Pattern chips should mimic your image data and how it overlaps where possible
  - Long and skinny images and overlaps, then try long and skinny Pattern and Search chip size dimensions

- Fit Chip
  - Will match the dimension of the Search Chip
  - Filled with the resulting Goodness of Fit values at every position that the Pattern Chip has been walked across the Search Chip computing a fit correlation value
  - The Fit chip is only good to a maximum of ONE pixel accuracy

- Strategize with complex datasets
  - More than one registration definition file may be necessary to register all measures and points
    - 1st time: pointreg cnet=in.net onet=out-1.net deffile=reg-1.def points=all measures=***all***
    - 2nd time: pointreg cnet=out-1.net onet=out-2.net deffile=reg-2.def points=all measures=***candidates***

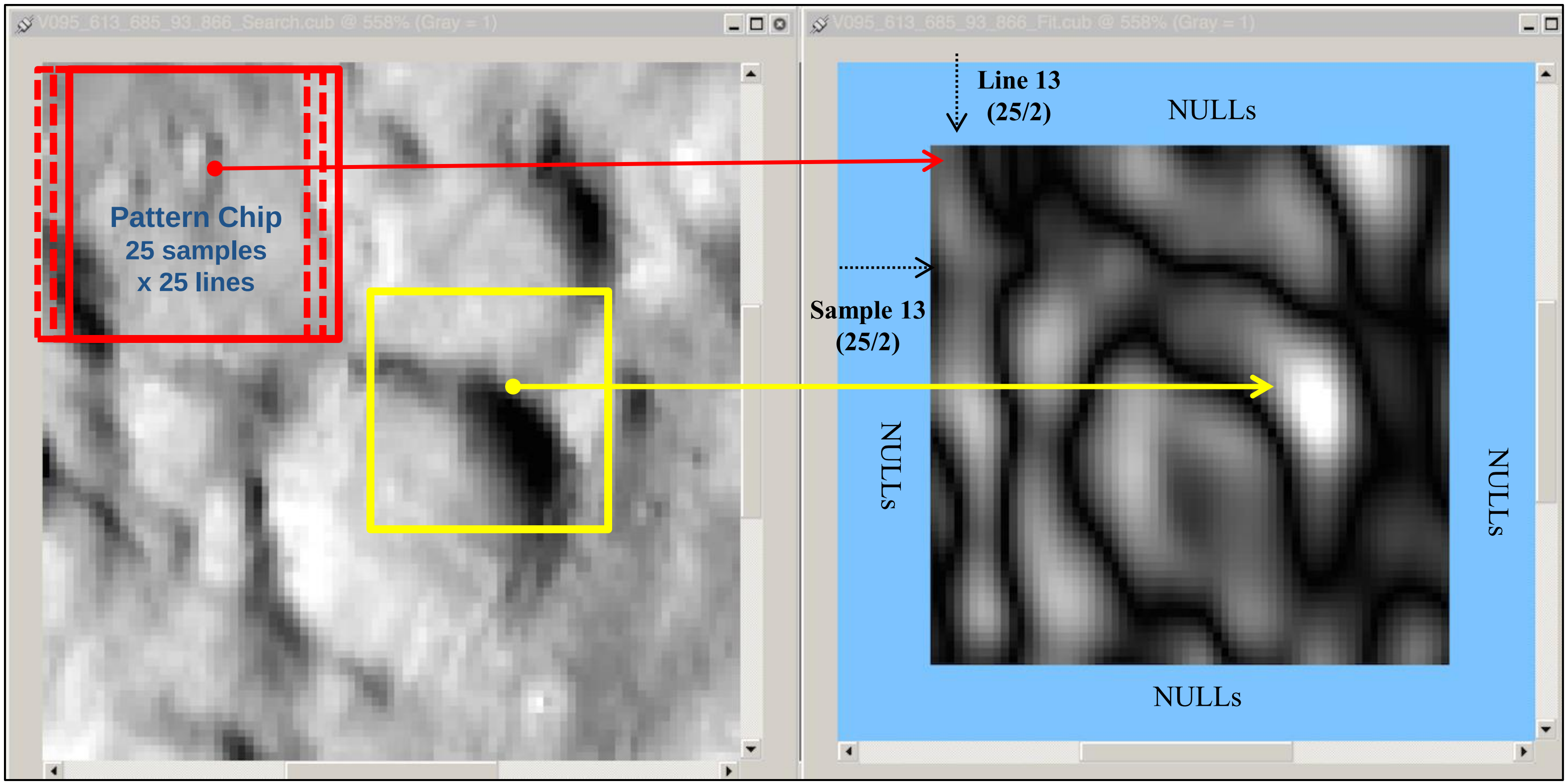
Pattern Chip
25 samples x 25 lines



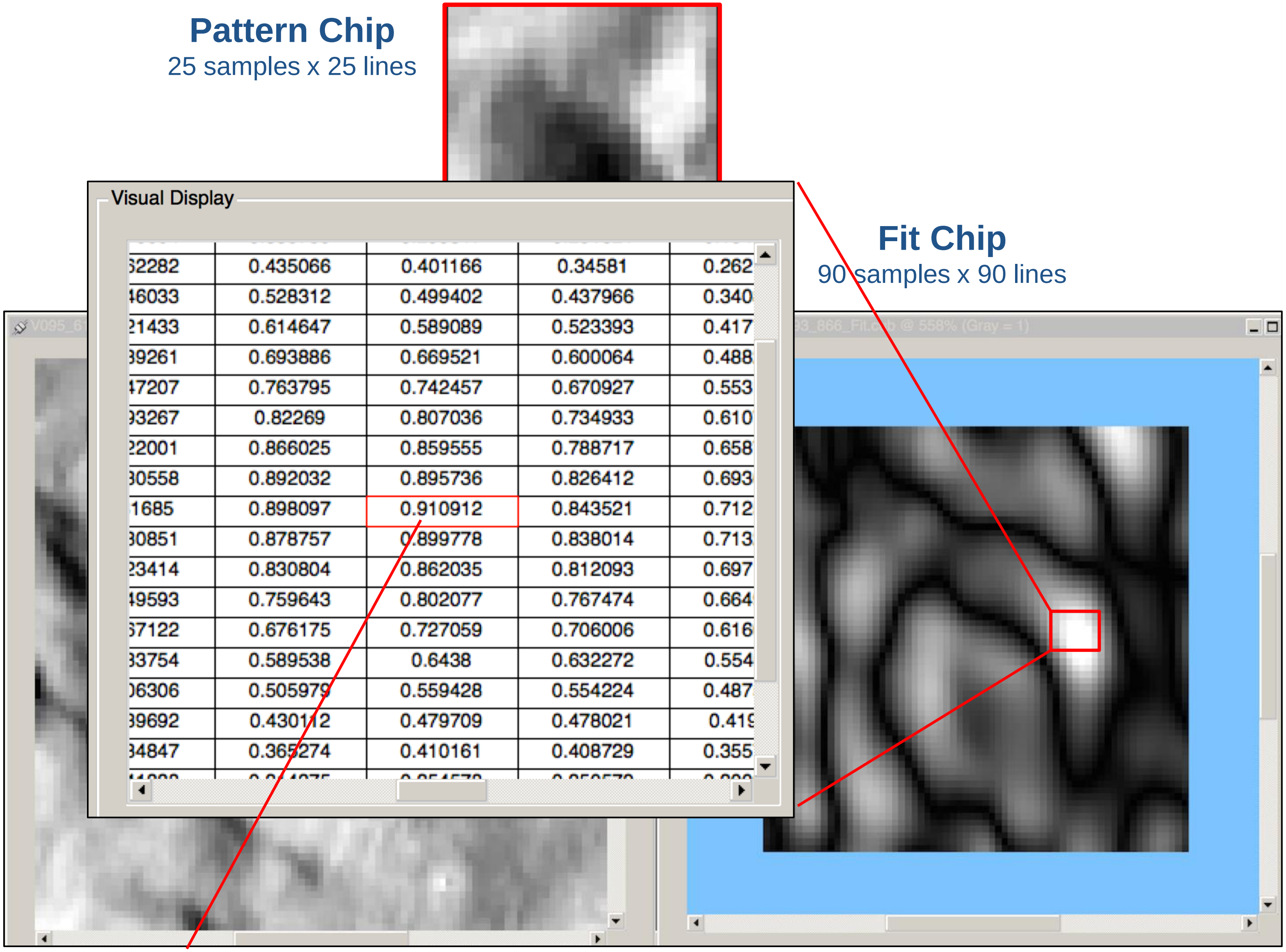
The Pattern Chip is a sub-area extracted from the Reference image measure within a control point. The center of the chip is the coordinate sample and line

Search Chip
90 samples x 90 lines

Fit Chip
90 samples x 90 lines



The Fit Chip is filled with the resulting Goodness of Fit (correlation value) value at every position that the Pattern Chip has been walked across the Search Chip



52282	0.435066	0.401166	0.34581	0.262
46033	0.528312	0.499402	0.437966	0.340
21433	0.614647	0.589089	0.523393	0.417
39261	0.693886	0.669521	0.600064	0.488
47207	0.763795	0.742457	0.670927	0.553
93267	0.82269	0.807036	0.734933	0.610
22001	0.866025	0.859555	0.788717	0.658
30558	0.892032	0.895736	0.826412	0.693
1685	0.898097	0.910912	0.843521	0.712
30851	0.878757	0.899778	0.838014	0.713
23414	0.830804	0.862035	0.812093	0.697
49593	0.759643	0.802077	0.767474	0.664
57122	0.676175	0.727059	0.706006	0.616
33754	0.589538	0.6438	0.632272	0.554
06306	0.505979	0.559428	0.554224	0.487
39692	0.430112	0.479709	0.478021	0.419
34847	0.365274	0.410161	0.408729	0.355

The sample, line coordinate of the highest value within the Fit Chip is used to compute the Adjusted Registered sample, line coordinate of the image measure.

The Fit Chip is filled with the resulting Goodness of Fit (correlation value) value at every position that the Pattern Chip has been walked across the Search Chip

Quality Control of a Network

Before running *jigsaw*, the completeness and “health” of the input control network should be double checked

-***cnetcheck*** creates certain output files only when an issue has been encountered

-These reported issues need to be addressed before running jigsaw

-Always reported are output files with suffix:

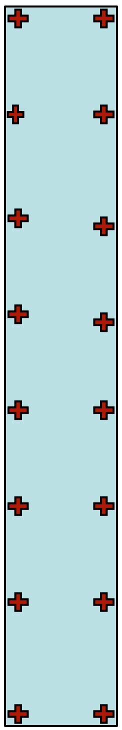
- ***_Island.1*** (this means you have one single network of points - *this is good*)
- ***_LowCoverage.txt*** – the default for the convex hull value is 1.0; this results in always reporting the hull for all images
    - The convex hull indicates the distribution of points across the image (described in next slide)
    - Make sure each image contains greater than 3 or 5 control points
    - Sort the hull value in ascending order for the images with the lowest hull to display on top
        - command: *sort -nb -t, -k3 your-prefix_LowCoverage.txt | more*

-Common issues

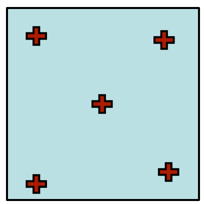
- Identify single measures in a point (only one image measured at a control point location)
    - *cnetcheck* creates a file with suffix = ***_SinglePointCubes.txt***
    - Optimum is more than 3 images per point at varying viewing angles to support convergence
    - Single measures can be okay with Fixed or Constrained points with a known lat/lon/radius (use with caution)
    - Either delete the points listed (*qnet, cnetedit*) or find additional overlapping images (*qnet, cnetadd*)
- Avoid *more than* 1 “island”, if all images are tied together then only a single island is reported (described above)
    - If cnetcheck creates files for more than one island (***_Island.1, _Island.2, _Island.3***…)
        - These multiple Island.# files will contain the filenames identified in each island
        - This suggests a disconnect or lack of points shared between images
            - Use *qmos* to overlay the network on the image footprints to identify the disconnected areas
- Identify the images that have only one point (hull=0.0000)
    - cnetcheck creates a file with suffix = ***_SingleCube.txt***

Quality Control of a Network

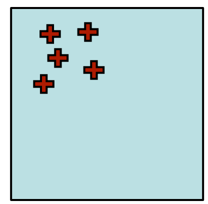
Control Point Distribution Across an Image



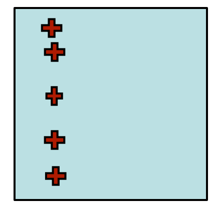
Good distribution
Convex Hull = ~0.99

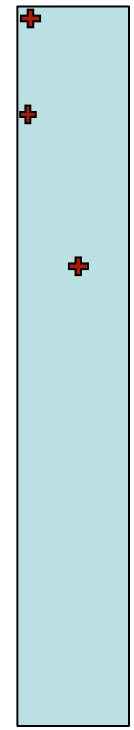


Bad distribution
Convex Hull < 0.5



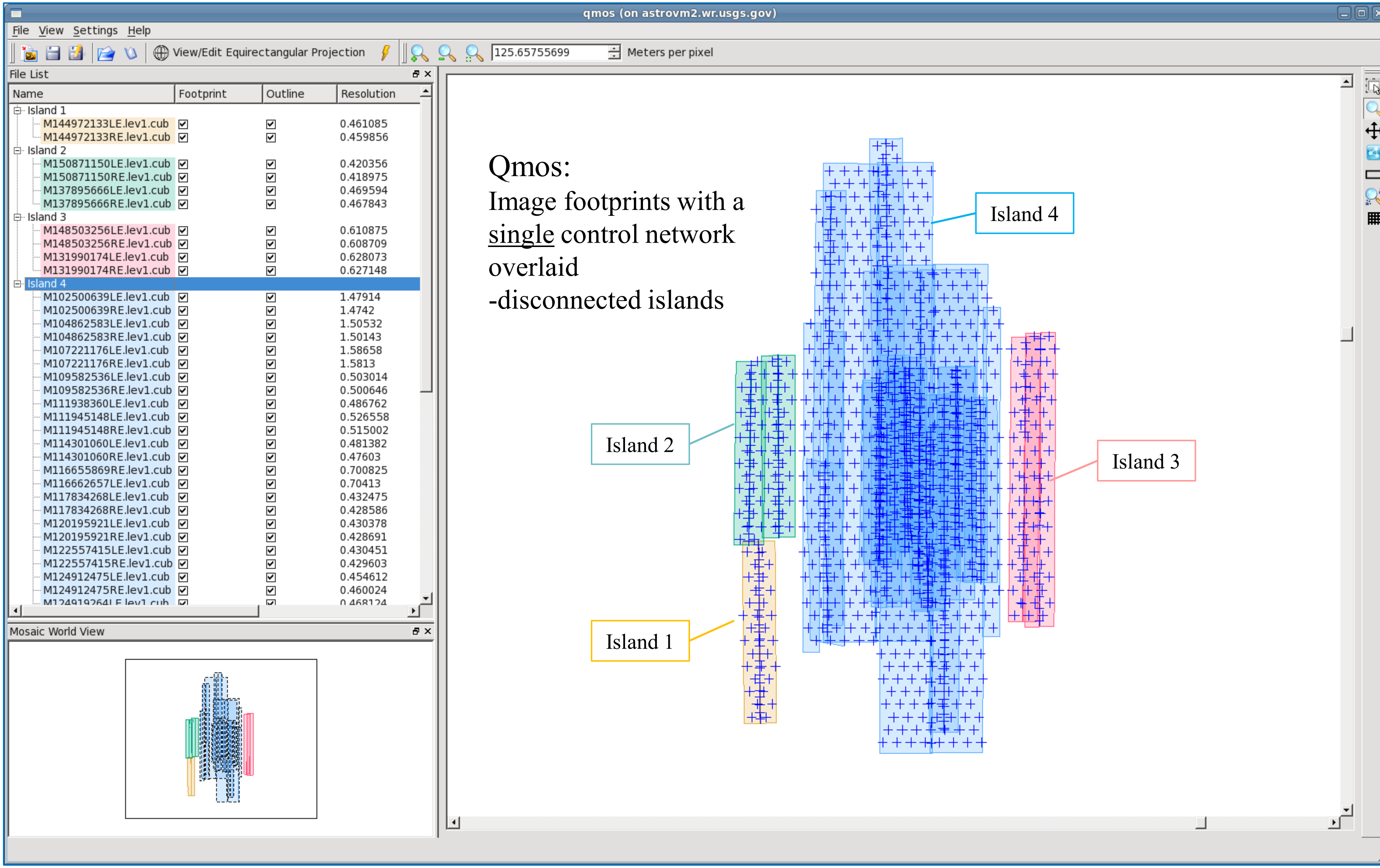
Bad distribution
Convex Hull < 0.5





- The point measurement distribution across the area of the image is critical for both frame cameras and line scan cameras. Think of the measurements as anchor ties.
- In addition, the distribution of the points in the context across the overlapping areas of the images is also important and should be thought of as anchor ties (tacks) through the ‘stack’ of images

Example of control network “Islands”



The **blue “+”** marks above are the control points overlaid on top of the image footprints by latitude and longitude position.
This is a single control net file and simply demonstrates what are considered islands where the images do not overlap.
Also, even though the images in Island1 overlap slightly with the images in Island2, qmos has indicated that these images do not share a control point

Control Network Terminology

<u>Point Type</u>

Free (no constraints) – also referred to as "tie-points"
Constrained (weighted to a 'ground') - also referred to as "ground-points"
Fixed (highest constraint possible to 'ground') - also referred to as "ground-points"

Apriori Latitude, Longitude, Radius

The location of the control point; a priori based on original SPICE
Computed by the first measure in the control point, or an average, or set when measured to a Ground reference (type=constrained or fixed)

Apriori Latitude, Longitude, Radius Sigmas

Parameters for weighing/constraining the location of the control point with type Contrained of or Fixed (set in qnet or cneteditor)

Adjusted Latitude, Longitude, Radius

The converged location of the control point after jigsaw

Ignore = True/False

If True, the control point and all measures are ignored by pointreg (optional) and jigsaw; cnetedit will remove these measures by default

<u>Measurement</u> The sample and line pixel coordinate of an image

Serial Number A unique identifier for every image constructed "on-the-fly" in the *cnet* applications using the instrument group keyword values. Internal to the control network; most applications will include the associated filename. To view a Serial Number run 'getsn from=image.cub'

Sample, Line The image pixel coordinate

Measurement Type

Candidate: least accurate pixel measurement-usually based on apriori SPICE
Manual: measured manually in qnet
RegisteredSubPixel: output coordinate resulting from *pointreg*, or *qnet-"Register" feature*

Ignore = True/False (Measure)

If True, the measurement will be ignored by pointreg (optional) and jigsaw; cnetedit will remove these measures by default

Reference=True (Measure)

The image that is used as the 'Pattern' chip for the pattern matching of all remaining measures (image coordinates) within a control point
The 1st measure listed within a control point is implicitly set as Reference=true by most applications
*This can be overridden by the 'cnetref' application or interactively by the user within qnet (always the left hand image in the control point editor)
Will often remain measurement-type=candidate after pointreg

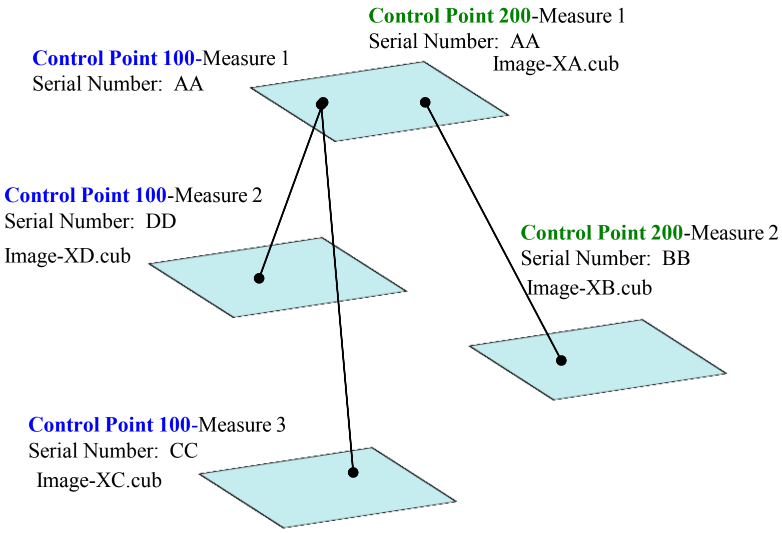
Goodness of Fit (Measure)

The pattern match correlation result for co-registration of measures to the reference

Sample/Line Residual (Measure)

Jigsaw residuals in pixels

Control Network Representation
(Abbreviated Version)



Contol Point Id 100

Type=Free, Constrained, Fixed
Apriori Latitude=
Apriori Longitude=
Apriori Radius=
Ignore=False/True

Measurement 1

Serial Number=AA
MeasureType=Candidate, Sub-Pixel, Manual
Reference=True
Sample=
Line=
Ignore=False/True

Measurement 2

Serial Number=DD
MeasureType=Candidaet,Sub-Pixel,Manual
Reference=False
Sample=
Line=
Ignore=False/True

Measurement 3

Serial Number=CC
MeasureType=Candidate,Sub-Pixel,Manual
Reference=False
Sample=
Line=
Ignore=False/True

Control Point Id 200

Type=Free, Constrained, Fixed
Apriori Latitude=
Apriori Longitude=
Apriori Radius=
Ignore=False/True

Measurment 1

Serial Number=AA
MeasureType=Candidate, Sub-Pixel,Manual
Reference=False
Sample=
Line=
Ignore=False/True

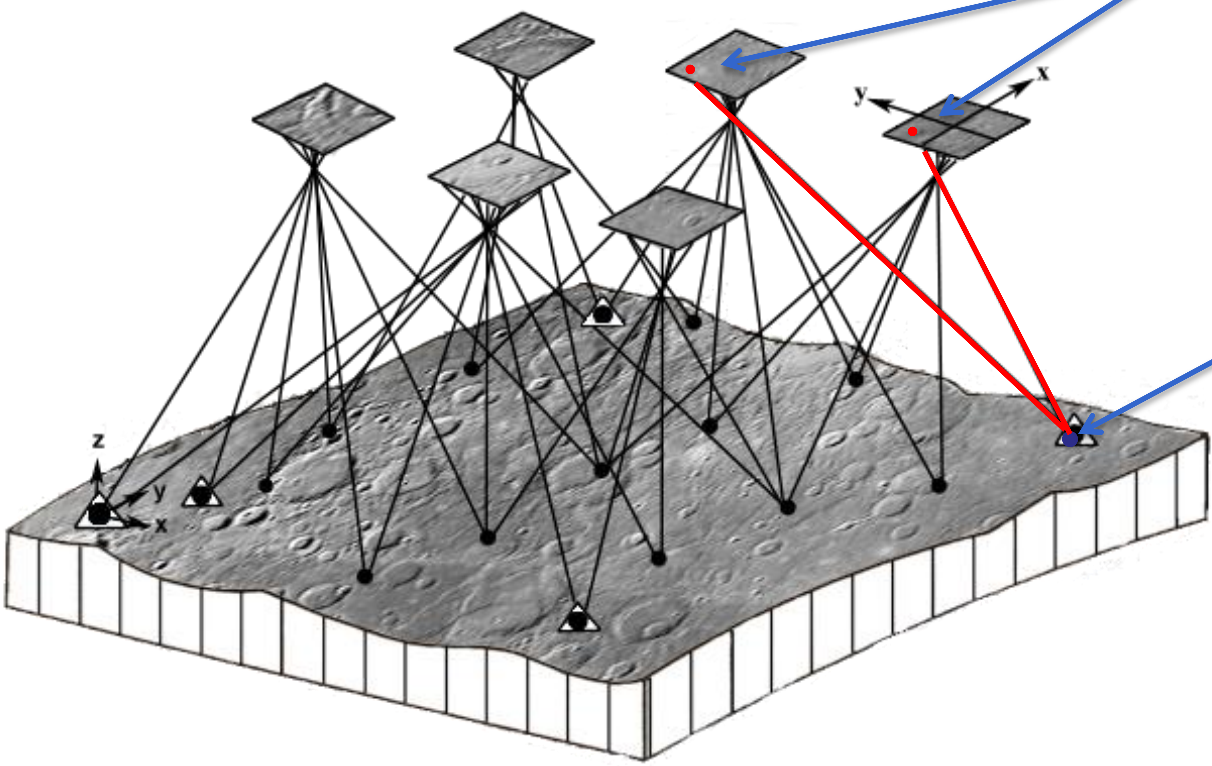
Measurement 2

Serial Number=BB
MeasureType=Candidate,Sub-Pixel,Manual
Reference=True
Sample=
Line=
Ignore=False/True

Control Point Type

Free (no constraints) – also referred to as "tie-points" or "match-points"
Constrained (weighted to a 'ground') - also referred to as "ground-points"
Fixed (highest constraint possible to 'ground') - also referred to as "ground-points"

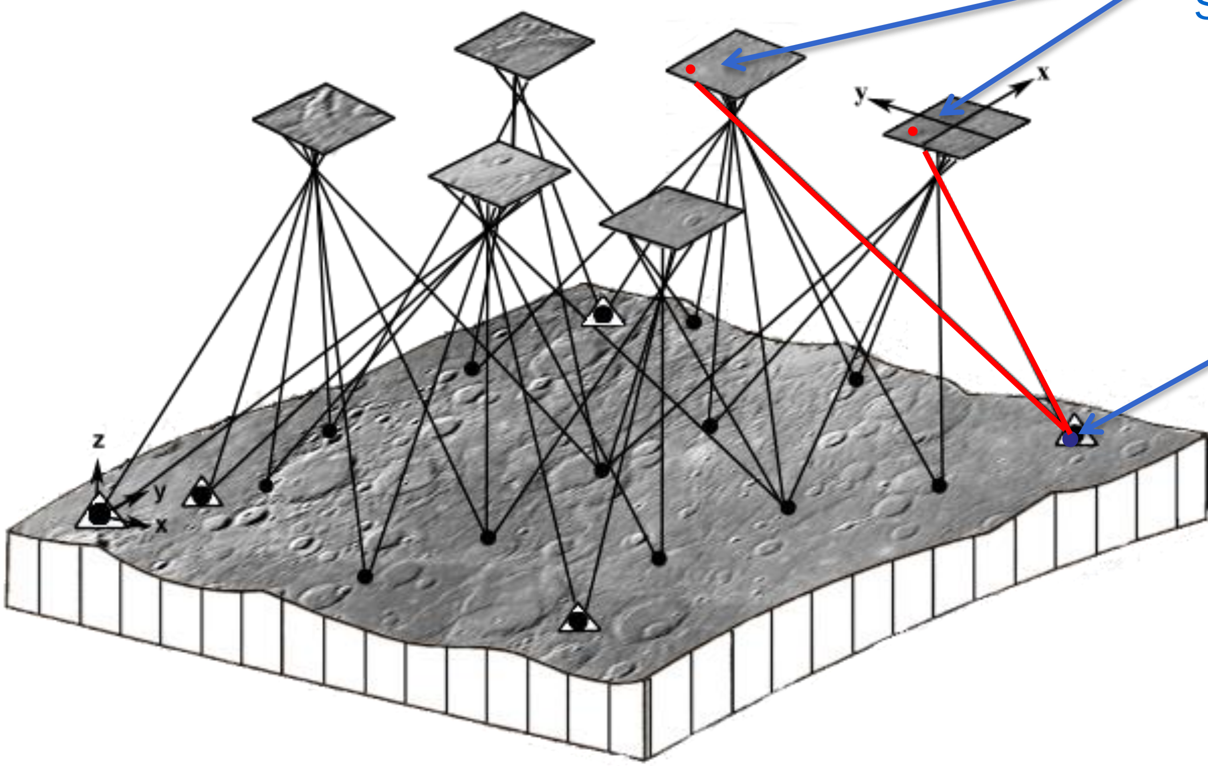
Control Network Representation



Measurement
Image (Serial Number)
Sample Coordinate (x)
Line Coordinate (y)

Point Id
Latitude (x)
Longitude (y)
Radius (z)
Latitude Sigma
Longitude Sigma
Radius Sigma

Control Network Representation



Measurement

Image (Serial Number)

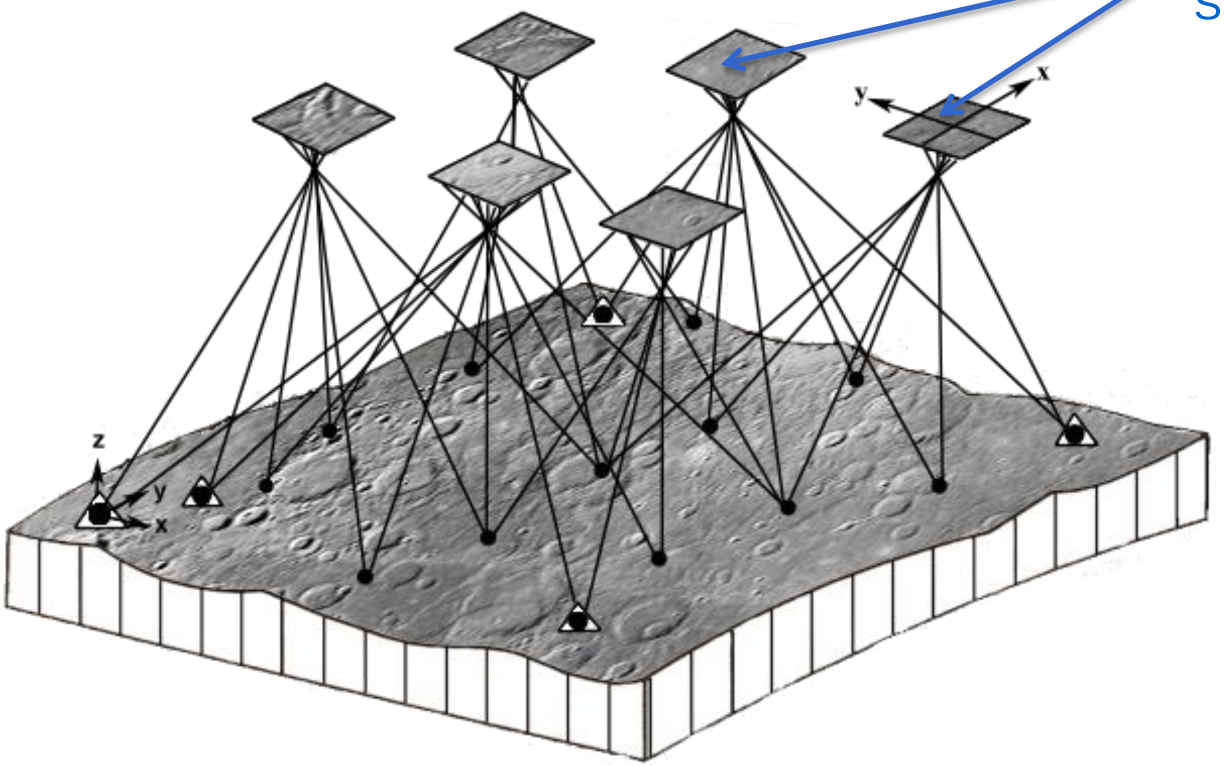
Sample Coordinate (x)

Line Coordinate (y)

Point Type

- Constrained
  - -Assigned Weights
- Free
  - -No Weights Set
- Fixed
  - -No Movement

Control Network Representation



Measurement
Image (Serial Number)
Sample Coordinate (x)
Line Coordinate (y)

* Serial Number

Unique Identifier for each image generated “in real-time” and stored by the Isis control network applications. The serial number is based on combined values from mission and instrument specific keywords

Control (Ground) Points
Type=Constrained or Fixed

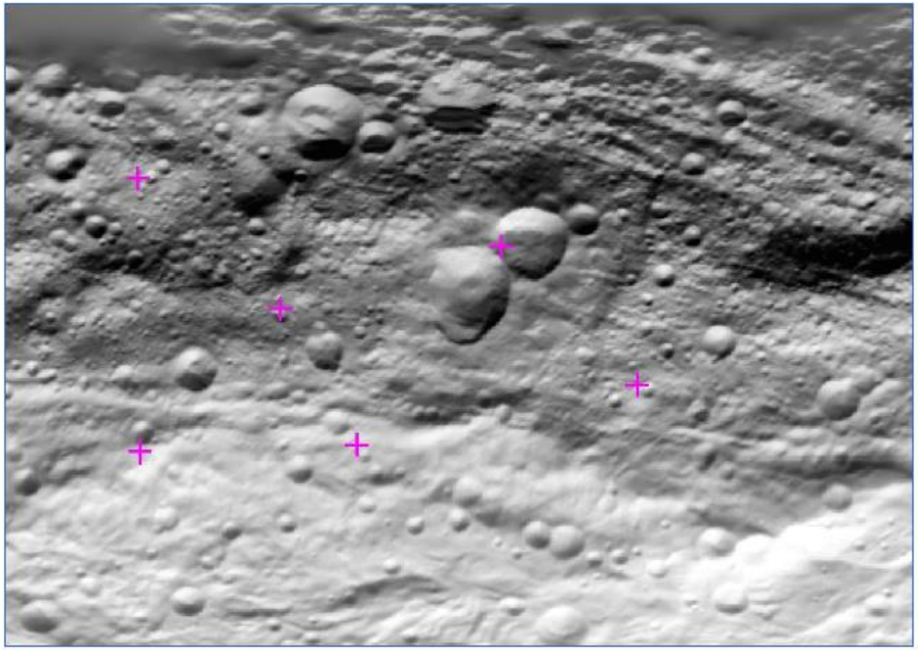
- Control Point or Ground Point
  - Represents a known location on the ground (latitude/longitude/radius)
  - Sigma values or weights are assigned to each Control Point (indication of how accurate)
  - Example known 'Ground' sources
    - For radius/topography values at a location
      - LOLA for Moon, MOLA for Mars, Gaskell DEM or DSK for Vesta
    - For 'horizontal' latitude and longitude location values
      - DEMs
      - Basemaps

- Distribution
  - Control points need to be distributed across the ground coverage of the images
  - Control points need to distributed across/within the overlaps of the images
  - Refer to: http://isis.astrogeology.usgs.gov/IsisWorkshop/uploads/8/80/Qnet_demo.pdf

Control (Ground) Point Distribution
Type=Constrained or Fixed

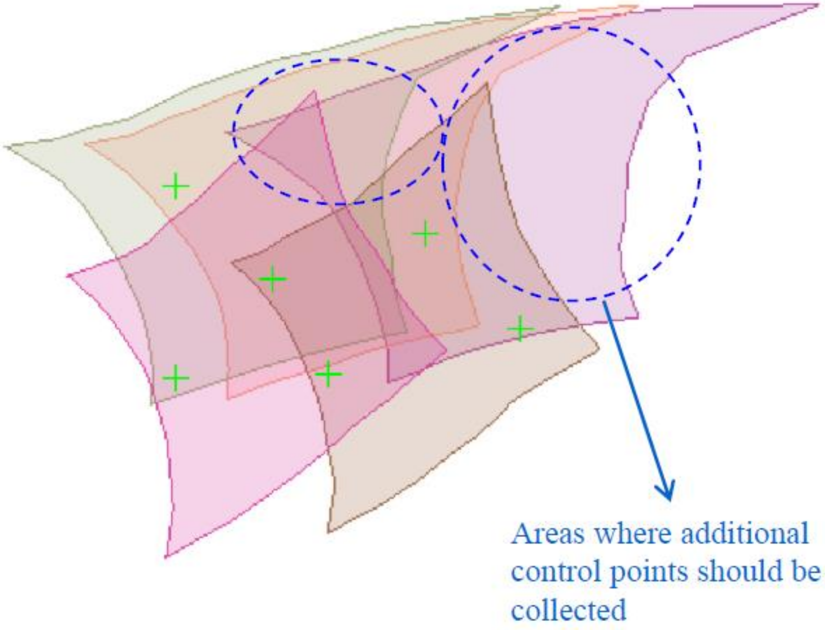
Vesta – Gaskell DEM

- Equirectangular
- Shaded version
- Pink marks indicate "ground" control point locations
  - This example is somewhat sparse



Vesta – DAWN Image Overlaps

- Equirectangular
- Green marks indicate "ground" control points



Jigsaw

While iterating with jigsaw, do not select ‘update=true’ (do not update labels) until a satisfactory solution has been reached

*file_prefix*_**bundleout.txt** = Summary file of bundle results for images and points

*file_prefix*_**bundleout_images.csv** = comma delimited file on image results

*file_prefix*_**bundleout_points.csv** = comma delimited file on point results

*file_prefix*_**residuals.csv** = residual (in pixels) information for every measure

Most often measurements with the highest residuals need to be investigated and corrected in qnet

Sort the residual reported for both sample/line in descending order (10th column)

If desired, only keep columns 1, 2, and 10 (PointId, Filename, Sample/Line_Residual)

>sort -rnb -t, -k10 file_prefix_residuals.csv | *cut –f1,2,10 –d, > jigout_residual_sorted.csv*

Jigsaw – bundleout.txt

IMAGE EXTERIOR ORIENTATION *(units currently not labeled in bundleout.txt)*

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

Image Serial Number: ….

LOCATION OF SPACECRAFT POSITION COORDINATES (J2000)

Image Parameter	Initial Value	Total Correction	Final Value	Initial Accuracy	Final Accuracy
	(kilometers)	(kilometers)	(kilometers)	(meters)	(kilometers)
X	1518.61974167	-0.12851163	1518.49123004	800.00000000	1.14898008
Y	-453.22482389	-0.40930647	-453.63413037	800.00000000	1.12170768
Z	1433.20602286	0.21146863	1433.41749149	800.00000000	1.15443907

THREE ANGLES OF CAMERA POINTING

RA (a)	-133.72803633	1.89167694	-131.83635939	-1.00000000	0.12562597
DEC (a)	103.72686026	-1.94435795	101.78250231	-1.00000000	0.12354186
TWI (a)	46.74631395	1.65568489	48.40199884	-1.00000000	0.4954500