Quad-G+: Automated Georeferencing of Scanned Map Images

User Manual

Version 1.1
8/30/2018

James E. Burt, Gregory Allord, Ken Then, Jeremy White, A-Xing Zhu

Overview

This document describes software for georeferencing scanned topographic quadrangles and other map images. In other words, the software converts an image from a scanner's coordinate system to a known spatial reference system (SRS). The SRS can be a geographic (longitude/latitude) system defined by a datum, or a projected coordinate system defined by a datum and map projection. Usually the SRS will be that of the original map, but that is not required by Quad-G+. The software operates in much the same way as other georeferencing tools: a small number of points are located in both image coordinates and geographic coordinates. These so-called *ground control points* (*GCPs*) are used to establish a relationship between the image and the SRS. This relationship is operationalized as a polynomial model linking the two systems. When applied to a scanned map, the model produces a new raster image in the desired SRS.

Quad-G+ differs from standard georeferencing tools in that the control points are found automatically rather than provided by a user through on-screen digitizing. In essence, it substitutes computer search and pattern matching for manual panning, clicking and data entry. This allows the software to process an arbitrarily large batch of scanned images without operator supervision. Generally speaking, Quad-G+ has advantages over conventional tools whenever one has more than a few maps in a series that require georeferencing, or when one wants to georeference a single map using more than a few GCPs.

A companion article (*CAGIS*, 2018, submitted) discusses the theory, underlying assumptions and behavior of Quad-G+ in detail. Our focus here is on program use, including details about program input and output.

Quad-G+ is distributed as a MS Windows™ executable under the GNU General Public License. The program is available at www.???, as are sample input datasets and output files. Quad-G+ supersedes an earlier version---Quad-G---which was developed as part of the USGS Historical Quadrangle Scanning Project.[1] Quad-G+ has many features Quad-G lacked, including:

- Output images warped to any projected or geographic coordinate system
- Checkboxes for UTM and polyconic output
- Output of raw scan with GCPs
- CSV or XML batch file input
- Images are opened as GDAL DataSet objects supporting 100+ input raster formats and their variants
- Scan resolution determined from file if not specified on input
- Input SRS specification can be given as WKT, proj4, EPSG, etc.,
- Input SRS can be projected or geographic coordinate system
- Flexible quadrangle description
- Optional compressed file output
- Exclusion of individual mark grid locations

Use of Quad-G+ rests on the following assumptions:

1) The mapped quadrangle area comprises a rectangle in latitude and longitude outlined by a black neatline. The four corners of the rectangle are typically used as GCPs. Additional printed control marks (if any) comprise a regular lon/lat grid on the perimeter and within the quadrangle. For example, Figure 1 shows the 16 controls marks in a 7.5 minute USGS topographic quadrangle. In this case the grid is 4x4 with marks every 2.5 minutes. In general, the grid need not have the same number of rows as columns, and the spacing in longitude need not match the latitude spacing.

2) As seen in Figure 1, a standard set of shapes is assumed for the control marks. Note that interior, corner and edge marks each have a distinct shape. These shapes are built into Quad-G+ in the orientations shown. Thus an interior mark is always a "+" shape, and a top edge mark is always a "T" shape.

3) The program was designed for large-scale maps whose meridians and parallels are close to straight lines intersecting at nearly 90°. As a practical matter, Quad-G+ is adequate for scales of 1:250,000 and larger.

4) The map sheet must be scanned within 5° of cardinal directions. In other words, neatlines are assumed to be within 5° or vertical and horizontal.

[1] Quad-G was used by the USGS to georeference hundreds of thousands of quadrangles comprising the National Map's Historical Topographic Map Collection.

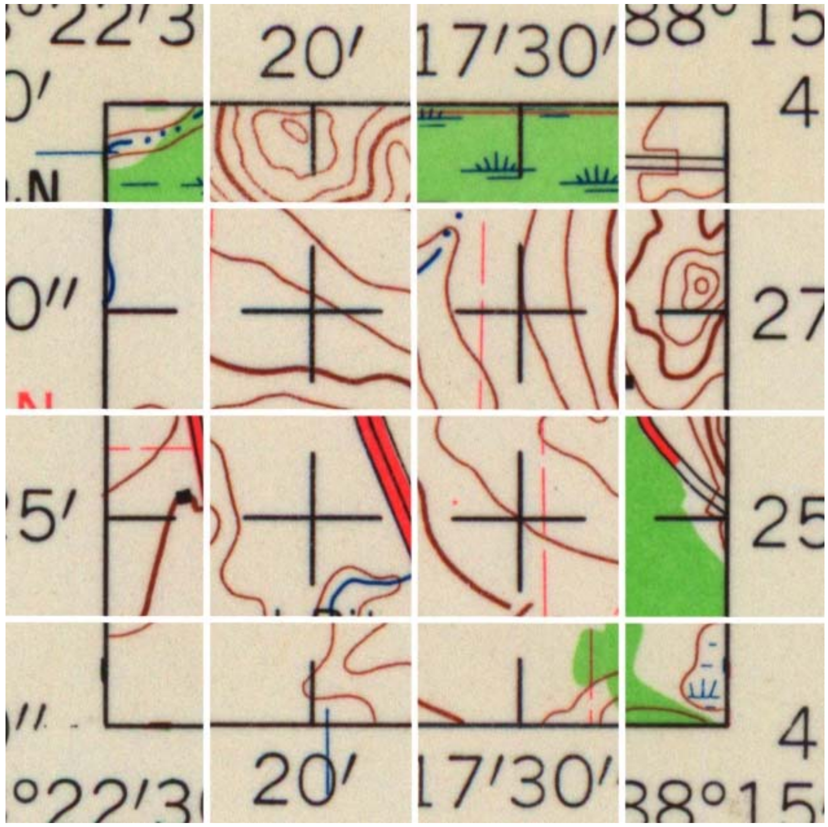

Figure 1. Standard control marks in the 4x4 layout of 7.5-minute quadrangles.

5) We assume that the map has been scanned in either a north-up or east-up orientation. As support for this assumption, we have observed that scan operators typical feed map sheets in a "normal" orientation so that the northern border is toward the top of the image. However, sheets that are much wider than they are tall are sometimes fed transversely, with the eastern border at the top. As a working hypothesis Quad-G+ assumes that the map has been scanned in north-up orientation. For quads with more columns than rows of control marks the program compares the aspect ratio of the image to the aspect ratio of the latitude-longitude quadrangle. If the image is inconsistent with the quadrangle, the program assumes the map has been scanned east-up and image coordinates are rotated accordingly. Please note that regardless of input orientation the output image file will always have north toward the top.

6) As is standard with other georeferencing tools, we assume a low-degree polynomial adequately represents the relationship between SRS and image coordinates. As explained in the companion article, the polynomial accounts for image rotation, map sheet distortions and other deformations.

Use of Quad-G+ is straightforward: a data file is provided that describes the images to be georeferenced. The required information for each image includes the image file name, map extent, control mark layout, and several other parameters described below. Quad-G+ reads the data file and processes each image in turn. Alternatively, there are options for manual selection and processing of individual images listed in the input file and for direct input of image data via a dialog. Diagnostic statistics are written to a log file for examination outside of Quad-G+. Visual quality assurance is provided by extracting and searching control subareas from output images. Output images are stored as GeoTIFF rasters in a location determined by the user. (Please see http://trac.osgeo.org/geotiff/ for information about the GeoTIFF format.)

Installation

Quad-G+ is provided as Windows™ binary that can be executed from any directory. There is no install procedure to run nor are any registry changes required. The "installation" consists of copying the distribution package to any directory. As with any application, users can create desktop or program menu shortcuts pointing to the executable file. The program requires the Microsoft .NET framework version 4 or higher.

Quad-G+ uses the GDAL raster package and related software. All dependencies are distributed with Quad-G+ in a subfolder, so GDAL itself need not be resident. Changes to the system environment (path, etc.), persist only while the program is running.

Input Data File

A single data file provides information for a suite or "batch" of input images. Typically all of the images in a batch come from one map series, but the definition of a batch is left completely to the user. The only requirement is that all images within a batch reside in the same folder. Likewise, all output images for a batch are placed in a single folder. (Input and output folders are specified by the user.) Batch data must be provided as a Comma Separated Value (CSV) or Extensible Markup Language (XML) file. CSV input requires a header row naming the columns present. Each record after the header represents a single scan. The XML file consists of a standard preamble and a list of scans. Within the scan list each image is described in an XML block delimited by <scan> and </scan> tags.

The choice between CSV and XML input formats is purely a matter of user preference---they have the same capabilities. Moreover, corresponding CSV column names and XML tags are identical, and Quad-G+ attempts to recognize obvious aliases for field names and tags.

For example, the following CSV and XML listings shows a data file for georeferencing the same two scan images:

File	SRS	Scale	North	South	East	West	Cols	Rows	Mark Size	Output Root
301.tif	NAD27	1:100000	36	35	-100	-102	9	5	.1in	Amarilla
502.tif	UTM4.wkt	0	67 45	67 30	-156	-156.5	4	4	.1in	Ambler

```
<?xml version="1.0"?>
<ScanList>
	<Scan>
		<FileName>301.tif</FileName>
		<SRS>NAD27</SRS>
		<Scale>1:100000</Scale>
		<North>36</North>
		<South>35</South>
		<East>-100</East>
		<West>-102</West>
		<ControlMarkLayout>5x9</ControlMarkLayout>
		<ControlMarkSize>0.1in</ControlMarkSize>
		<OutputRootName>Amarilla</OutputRootName>
	</Scan>
	<Scan>
		<FileName>502.tif</FileName>
		<SRS>UTM4.wkt</SRS>
		<North>67.75</North>
		<South>67.5</South>
		<East>-156</East>
		<West>-156  30 00</West>
		<MarkLayout>4x4</MarkLayout>
		<MarkSize>0.1</MarkSize>
		<OutputRootName>Ambler</OutputRootName>
	</Scan>
<</ScanList>
```


A complete description of batch files is presented below, but for now note the mix of decimal degrees and degrees/minutes/seconds for longitude and latitude. The quadrangle location and extent is given in terms of boundary lon/lat values. The mark spacing is not given, but can be found knowing the number of marks and the quadrangle size.

Scale is set to "0" for the second scan in the CSV file and is not present in the corresponding XML entry. In this case Quad-G+ will estimate map scale from the image size and other information for that scan. The general point is that Quad-G+ uses batch information where available; otherwise it attempts to determine required values internally. Note also there are varying names for the same field, such as "ControlMarkLayout" and "MarkLayout". Wherever possible the program will recognize names from the input strings provided. Of course, the exact names given later in the document will always work.

Table 1 shows input field names and their meanings. Entries in the first column of Table 1 will appear in the header of CSV files or as tags in an XML file. CSV headers are not case-sensitive and blanks are removed. Each cell shows alternative field names---if any.

Column Name or XML Tag	Meaning	Examples
FileName	Input image file, relative to source image directory. If	Jacoma.tif, 167001.tif

	MapName is not specified, the file name extension is removed and used as a prefix in naming output images.	
MapName	Optional name for output image files. This is used as a prefix. Various suffixes are appended depending on the output file as described in a later section.	West Madison Quadrangle
SRS	SRS of original map. This may be a datum only or datum and projection. It can be given as any string recognizable by GDAL or in one of the other forms discussed in note 1. Defaults to NAD27.	NAD27, WGS84, UTM21s.wkt, EPSG:4131, NAD83+UTM, NAD27+Polyconic
North	Northern quad boundary (see note 2 below)	35, 35.5, 30 30
West	Western quad boundary (see note 2 below)	-88.5, -88 30 00, 88.5W -81.375, -81 22 30, W88
East	Eastern quad boundary (see note 2 below). "Longitude" is retained for backward compatibility with Quad-G.	-88.5, -88 30 00, -81.375, -81 22 30, 88
South	Southern quad boundary (see note 2 below) "Latitude" is retained for backward compatibility with Quad-G.	42.125, 42 7.5 00, S 30 30, -15.5, -15 30 00, 30 south, 30 22 30 S
Resolution DPI	Scan resolution in dots per inch. If not present in the batch file or if set to 0 it must be present in the image file. (The image resolution and units are supplied as standard Tiff tags by most scanners.)	600, 0
Scale	Optional map scale: specify as denominator in representative fraction or use colon-separated ratio. Providing this tag will result in a better edge search. See note 3 below.	24000, 62500, 1:24000, 1:250000
Rows	Number of rows in control mark grid	2,4,5
Columns Cols	Number of columns in control mark grid	2, 3, 9
MarkLayout	Number of rows and columns of control marks on map when viewed in normal orientation. Do not adjust for transverse orientation. Note rows first, then columns. Used only if either Rows or Columns not specified.	4x4 (a square map in lat-long) 5x9 (5 rows in latitude, 9 columns in longitude)
MarkSpacing	Control mark grid spacing. Use only if mark spacing is the same in longitude and latitude (see note 2 below)	00 10, 2.5, 0.041666

LonSpacing MarkLonSpacing ControlMarkLonSpacing	Longitude distance between control marks (see note 2 below)	00 10, 2.5, 0.041666
LatSpacing MarkLatSpacing	Latitude distance between control marks (see note 2 below)	00 10, 2.5, 0.041666
TickLength MarkSize	Size of control mark legs. E.g., half the entire width of a “+” mark. Allowed units are mm (millimeter), in (inches), and pixels. Pixels assumed if no units are given.	0.25in, 5mm, 25pixels, 0.2
TickWidth	Width of quad neatlines and control mark segments. Defaults to 1 pixel	10, 3, 0.5mm
PerimeterMarksOnly Perimeter	Some maps lack interior marks (i.e., “+” sign in Figure1.) This optional flag indicates interior marks are absent or should be ignored.	True, T, t, False, F, f, 1, 0

Table 1. Input CSV fields and XML tags.

Note 1: The SRS represented in this field must contain a valid datum. Optionally, it may be a complete map SRS; i.e., a datum and projection. The datum given is always imputed to the GCPs and therefore it should match that of the original map. If not, output lon/lat values will not be correct. The projection, if any, is also applied to GCPs, but Quad-G+ computes projected coordinates from lon/lat values. Thus so long as the datum is correct for the map, the projected values will also be correct. This means output images will be correctly rendered in the output SRS whether or not it is the same as the original map projection.

The SRS value can a GDAL-style string, which means it can be a complete SRS specification, or a file name containing a specification, or one of the common datum names (NAD27, NAD83, WGS72, WGS84). The program will also accept a combination *datum+proj* where *datum* is one of those four and *proj* is either “UTM” or “Polyconic”. In this case QuadG+ builds the corresponding projection using the quad mid-point for the central meridian and the southern boundary for the UTM hemisphere or latitude of origin.

Note 2: Input files can specify values for latitude, longitude, and grid spacing in either decimal degrees or degrees, minutes, seconds. To obtain a value the program looks for a text string with one to three fields separated by blanks. If there is only one field, it is interpreted as degrees. If the second field is present, it is taken as minutes. If there is a third field, it is interpreted as seconds. Any of the three can have a decimal point. A minus sign or a “W”, “w”, “S” or “s” anywhere means the entire value is negative. South or west can be spelled out. Table 2 provides a few examples.

Input String	Interpretation
44.5	44.5 º (or 44º 30’)
44 30	44.5 º (44 º 30’)
44 30 30	44.50833 º (44º 30’ 30”)
44.50833	44.50833 º (44º 30’ 30”)
00 2.5	2.5’
-81 30	-81.5º (west longitude)
-5	-5 º (west longitude or south latitude)
5 S	5 south latitude
5 west	5 west longitude

Table 2. Examples of input for latitude and longitude.

If longitude boundaries differ in sign we assume the narrower of the two possible quads. E.g., west and east boundaries of -179 and 179 are assumed to represent a quad 2º wide spanning 180ºE.

QuadG+ must be able to determine the quad extent and mark layout from the values provided. For the east-west layout you can provide any of the following

a) East and west edges and number of marks (mark spacing will be calculated)
b) One boundary along with the number of marks and their spacing (the other boundary will be calculated)
c) Both edges and the mark spacing (number of marks will be calculated). If necessary, the mark spacing will be adjusted so that the number of marks is an integer.

The control mark grid can have rectangular cells with different spacing in latitude and longitude. In that case use the ControlMarkLatSpacing and ControlMarkLonSpacing fields to specify respective values. If the grid is uniform with square cells the same value can be supplied for each, or the field ControlMarkSpacing can be used.

Note 3: Map scale is used to guess at the quad width and height in pixels. These are used to evaluate various quadrilaterals as potential quadrangle boundaries (neatlines). If the scale is not specified (or set to zero), Quad-G+ compares aspect ratios rather than dimensions. Such a test is obviously not as stringent but often suffices. After the quad has been identified map scale can be computed.

There is an important distinction between the input *data values* described here and user *settings* or *preferences* discussed below. Obviously, data values can vary from scan to scan. One image within a batch could have a scale of 1:24000, whereas another might be at 1:62500. By contrast, user preferences apply to the entire batch. Preferences can be changed in a Quad-G+ session, but not while a single image or batch of images is being processed. For example, *mark search radius* is a preference. All scans in a given batch are processed using the same search radius.

Output Map Files

Up to four output images can be created for any scan. All output images are GeoTiFF files with embedded georeferencing information, but various combinations of image compression and other attributes are possible. Collectively, the options form a suite of output file properties. These are specified through the dialog shown in Figure 2.

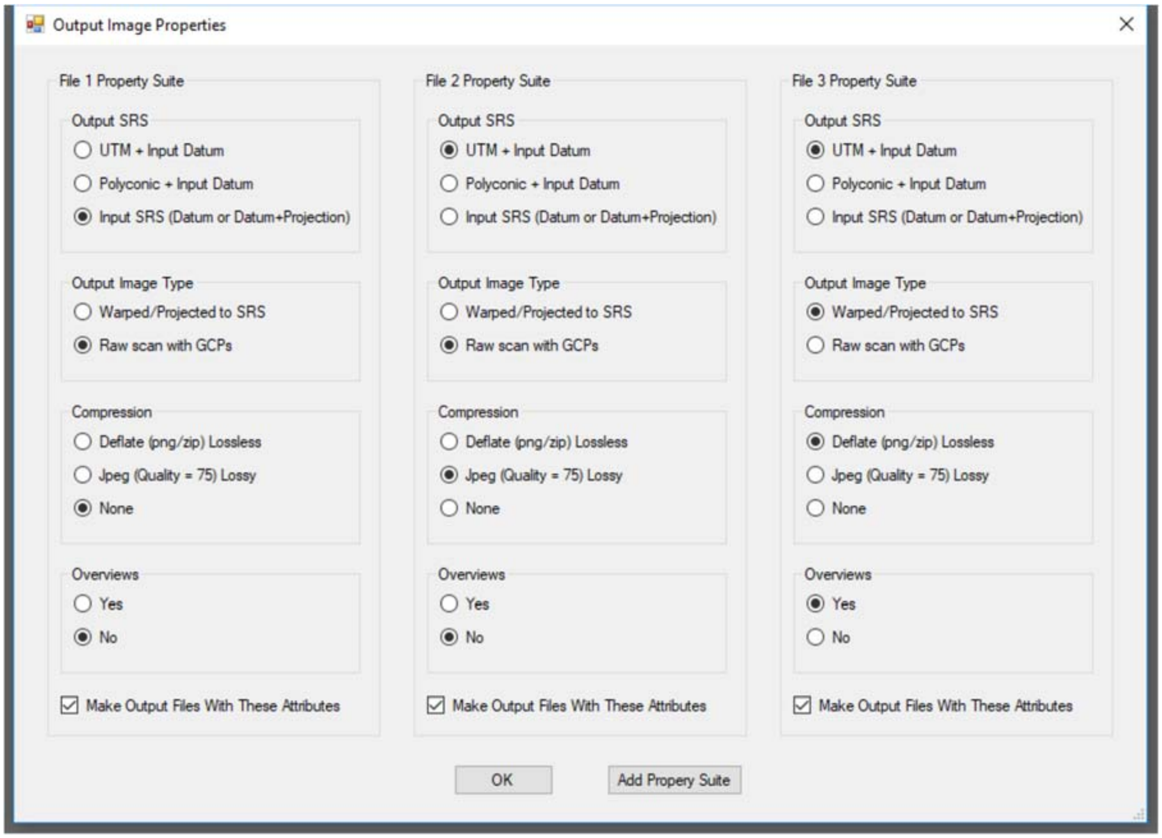

Figure 2. Output Image File Types

This example calls for three output images, each with a different combination of properties. Radio button meanings are as follows:

Output SRS:

- **UTM + Input Datum** The output file will have a UTM projection whose zone is appropriate for the map's longitude midpoint and northern latitude. The datum is copied from the input SRS.
- **Polyconic + Input Datum** The output file will be a polyconic with a central meridian equal to the longitude midpoint. The datum is copied from the input SRS.
- **Input SRS (Datum or Datum+Projection)** If the input SRS is only a datum, the output file will not have a projection---only the input datum is used. Otherwise the output file will have the full datum and projection.

Output Image Type:

- **Warped/Projected to SRS** Each output pixel is a square in the output SRS. If the SRS is a projection, they are squares in map projections units (e.g., meters). Otherwise the pixels are squares in longitude and latitude.
- **Raw Scan with GCPs** The input scan raster is replicated in the output file; every input pixel appears in the output file. GCPs are embedded in the output file using the specified SRS.

Compression:

- **Deflate** The compression method used in png and zip files, with no loss of information (change in pixel colors). Files are smaller but might take longer to load than uncompressed files.
- **Jpeg** The compression method most often used for photographs. The image raster is modified in order to reduce storage requirements.
- **None** No compression.

Overviews:

- **Yes** The output file contains the full raster along with smaller, lower resolution versions. Such "pyramid" files are useful for zooming in and out.
- **No** Only the full-resolution raster appears in the output file.

Individual files are named either *`root_srs`*`.tif` or *`root_srs`*`_GCP.tif` where *`root`* is the MapName (if provided) or the input file name without extension. The *`srs`* field corresponds to the file's output SRS (e.g., UTM, Polyconic). The GCP suffix is used for GCP files. Output images are made only for property suites whose checkbox is checked.

Georeferencing Steps

The complete sequence is as follows:

1) **Preference File Processing:** When Quad-G+ is loaded into memory it searches for an XML file saved from a previous session. Program options are read from the file if it exists, otherwise they are set to default values. Preference files are not maintained meant for user modification, thus the format is not described here. Preference values are changed via a dialog box described in the next section.

2) **Input File Processing.** All data for an image is processed to obtain the latitude and longitude coordinates for all control points and other map properties. If not provided, image resolution is determined from the scan file. The image orientation (normal or transverse), is also determined in this step.

3) **Find Corners.** The input image is searched for the bounding neatline. The search is informed by the quad aspect ratio, which is known given the control mark layout. If the map scale is provided the approximate pixel dimensions of the quad are known and also used in the corner search. The search first finds candidate

edges---nearly vertical and horizontal lines[2] that might be part of the quadrangle boundary. All combinations of candidate edges are projected to intersections that define a set of candidate quadrangles. Intersections are searched for nearby corner marks and if any are missing the quadrilateral is rejected. Remaining quadrilaterals are scored based on their various measures of size, shape, and control mark match. The best-scoring quadrilateral is used for image coordinates of each corner (Figure 3). After the dialog is dismissed, display windows are placed around all four corners for possible manual adjustment.

[2] In our experience scanned images are typically rotated by a few tenths of degree from vertical. In addition, except for rectangular projections, convergence of meridians guarantees that the quadrangle is not a true rectangle. Thus individual edges are only assumed to be within 5° of horizontal or vertical.

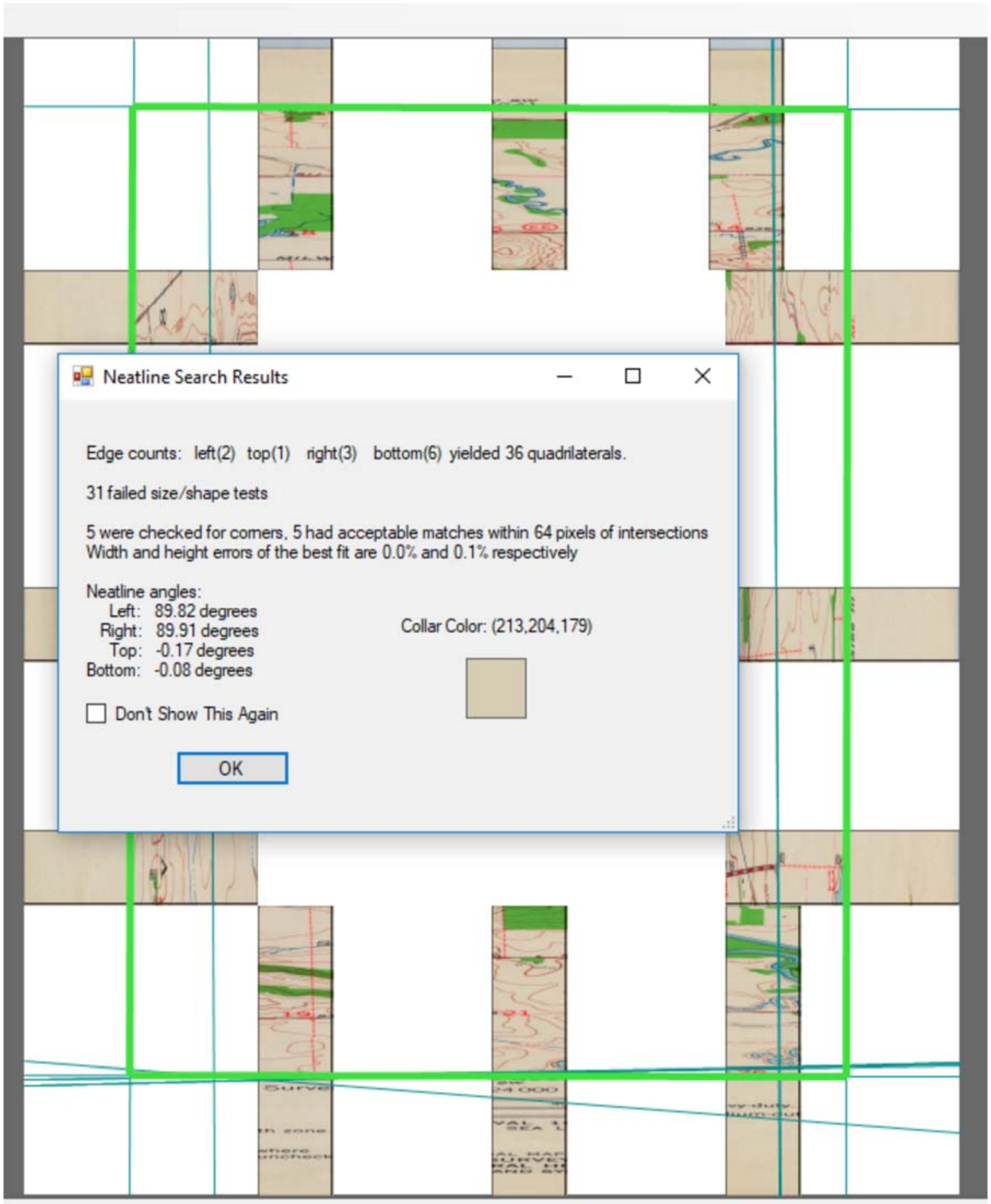

Figure 3. Results of search for quadrangle boundaries. The green line shows the best highest-scoring quadrilateral.

4) **Adjustment of Corners (optional, manual mode only).** If an actual quad corner is not within its window, it and other control marks will not be successfully located during the pattern search. An option is therefore provided for user adjustment of corner locations. The operator clicks on a window and drags until

the actual corner is reasonably close to the center of the window. Arrow keys can be used to move the image one pixel at a time. A shift-arrow combination accelerates movement to 10 pixels. There is no need for fine adjustment of corners; approximate image locations are sufficient.

5) **Identify Search Windows.** The program uses the grid layout and the image corner locations to guess at control mark locations. That is, a control mark grid is established based on the presumed corner locations. These become the center of search windows for all the GCPs, including the corner marks. If the PerimeterOnly flag is set, no interior windows are found (right-hand side of Figure 4).

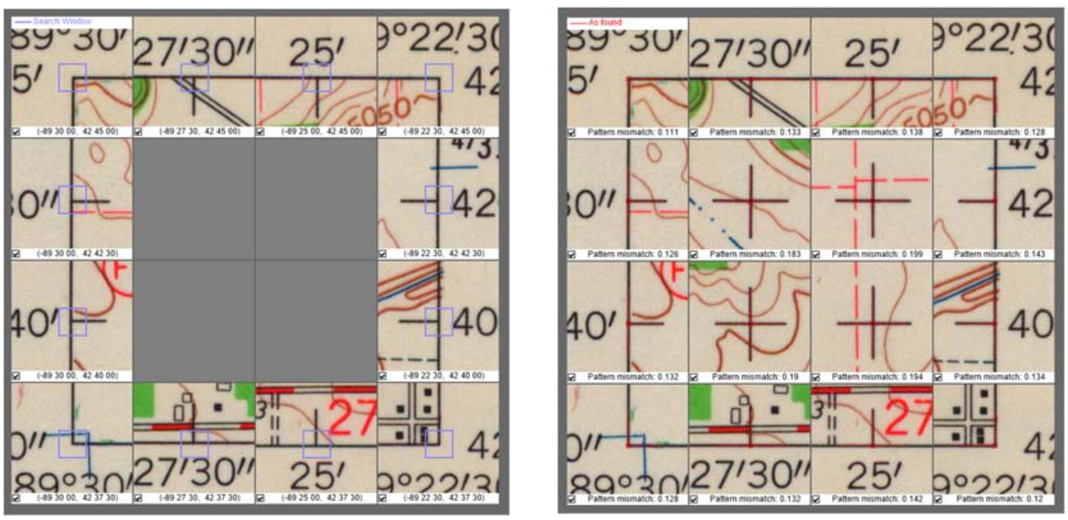

Figure 4. Search windows as established from quadrilateral corners.

6) **Adjust Search Windows (optional, manual mode only).** If a control mark is not within its window, it will not be successfully located during the pattern search. In this case the user should move the window until it is roughly centered over the correct control mark. The adjustment procedure is the same as for corner adjustment. Only approximate placement is required.

7) **Search Windows.** Each window is searched for its control mark. The area searched within each display window is determined by the “search radius” preference value described below. The proper control mark template is placed over every pixel in the search area, and deviations between the template and the underlying image are noted. As seen in Figure 5, the pixel location with smallest deviation is taken as the control mark location. [Please see the companion article for details.]

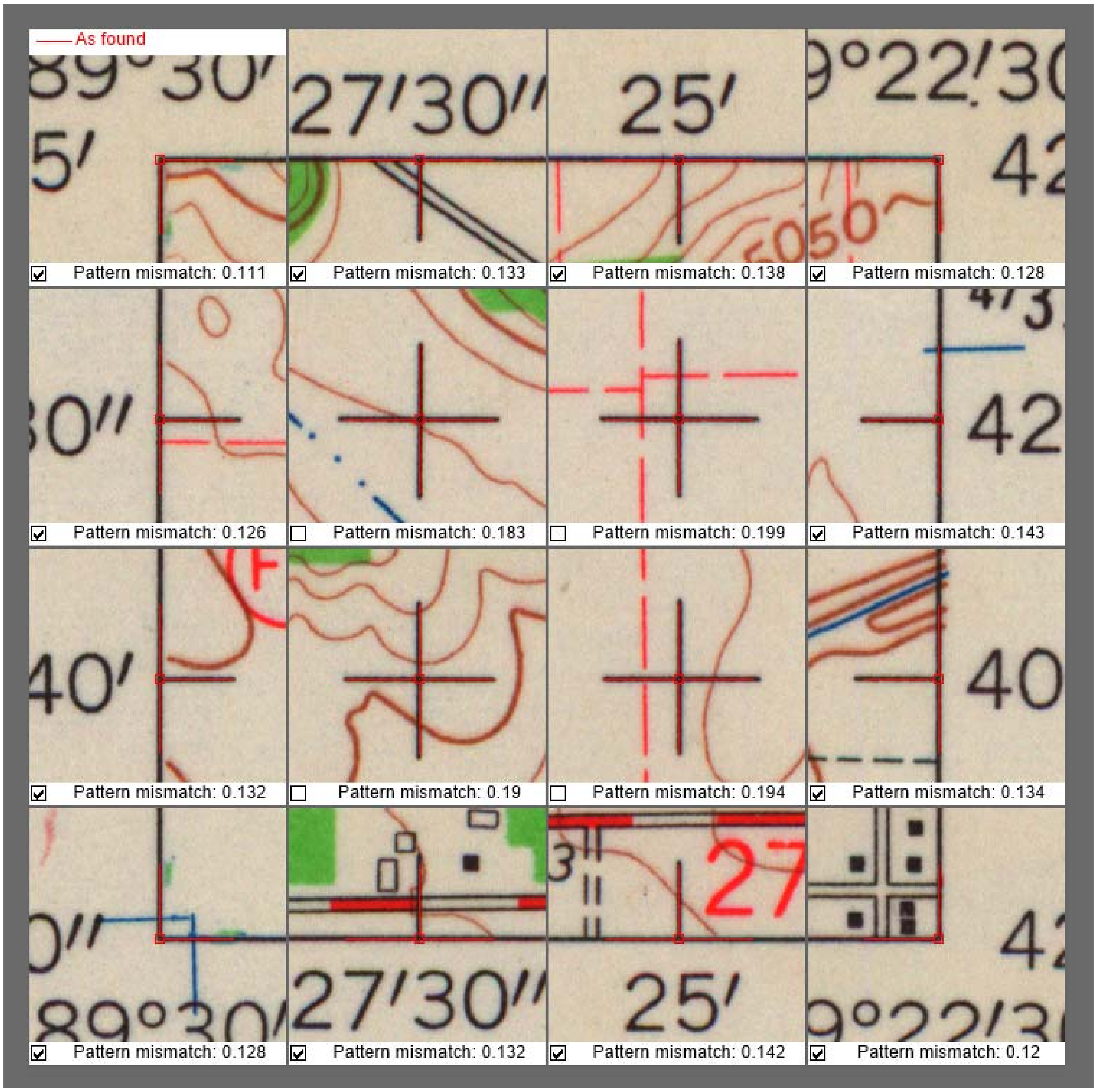

Figure 5. Results from pattern search.

8) **Control Mark Adjustment (optional, manual mode only).** If the search results are unsatisfactory, individual control marks (including corners) can be adjusted before going on to the next step. Because final positions will be used as input data for the fitting procedure, operators should position control marks as precisely as possible. The adjustment procedure is the same as for corner adjustment.

9) **Least-squares Fitting and Error Analysis.** The input file determines a longitude, latitude pair (λ,ϕ) for every control point. The desired output SRS gives a corresponding map coordinate (X,Y). Obviously, if the SRS is a datum, (X,Y) = (λ,ϕ) . The pattern search gives image coordinates (u,v) for the same marks in the scanned map. This can be visualized in table form:

Control Mark No.	Map X (known)	Map Y (known)	u (search result)	v (search result)
1	X_1	Y_1	u_1	v_1
2	X_2	Y_2	u_2	v_2
3	X_3	Y_3	u_3	v_3
…				
n	X_n	Y_n	u_n	v_n

We seek a function $\mathbf{f}(X,Y)$ yields a predicted (u,v) for any location. By default, Quad-G+ uses 2nd degree polynomials for the components of $\mathbf{f}$. That is

$$\hat{u} = \mathbf{f}_1(X,Y) = a + bX + cY + dX^2 + eY^2 + fXY \qquad (1)$$
$$\hat{v} = \mathbf{f}_2(X,Y) = g + hX + iY + jX^2 + kY^2 + lXY \qquad (2)$$

where $(\hat{u}, \hat{v})$ is the predicted image location. If only the four corner control marks are available, Quad-G+ uses a first degree polynomial (in effect d, e, f, j, k, l are zero). A first degree polynomial can compensate for image rotation, differential stretching in u and v, and shearing. A second degree polynomial is obviously even more flexible. In particular, it captures variations in projection scale that a linear function cannot.

The task is to choose the coefficients a through l that are optimal in some sense. As is standard practice, Quad-G+ finds coefficients that reproduce the observed (u,v) as closely as possible in a least-squares sense. That is, we solve the following optimization problem:

$$\text{Find } \mathbf{f}_1(X,Y) \text{ to minimize } S_u = \sum_{i=1}^{n}[\mathbf{f}_1(X_i,Y_i) - u_i]^2 = \sum_{i=1}^{n}[\hat{u}_i - u_i]^2 \qquad (3)$$
$$\text{Find } \mathbf{f}_1(X,Y) \text{ to minimize } S_v = \sum_{i=1}^{n}[\mathbf{f}_2(X_i,Y_i) - v_i]^2 = \sum_{i=1}^{n}[\hat{v}_i - v_i]^2 \qquad (4)$$

The coefficients enter partial derivatives of S_u and S_v linearly, so this is a problem in linear regression. However, because the image coordinates result from an automated procedure there is no guarantee they are accurate or even feasible. Precautions are therefore essential to ensure the program does not fail catastrophically during the fitting step. To this end Quad-G+ uses a technique known as singular value decomposition (SVD). SVD is certain to return a solution even in pathological situations, such as collinear control points. Numerical accuracy of the fit is improved by scaling all values (X,Y,u,v) to the unit square before optimization.

Solutions to (3) and (4) give the best-fitting polynomial transformation based on all control points. Cross-validation is used to assess the ability of the polynomial to capture the pattern of the control points. If the transformation is a good one, it ought to be able to successfully predict the location of a "new" control point not part of the fitting procedure. Cross-validation implements this idea by excluding a control point from (3) and (4), and using the resulting functions to predict that excluded point. Within Quad-G+, the cross-validation error is reported as the distance in pixels between the predicted and search-window locations of the control mark:

$$e_{(i)} = \sqrt{(\hat{u}_{(i)} - u_i)^2 + (\hat{v}_{(i)} - v_i)^2} \qquad (6)$$

The predicted values $(\hat{u}_{(i)}, \hat{v}_{(i)})$ in (6) are those generated by a model with the i^{th} data point omitted. Each control point is dropped in turn, and the resulting error is computed.

Note there are two predictions and two "errors" for each point. First, there are the model predictions and corresponding residuals. In addition, we have the cross-validation predictions and errors. By default he program displays the model predictions and errors in the thumbnail windows. Selecting "CV-predictions" changes to cross-validation values. Either method provides visual feedback as to whether or not the polynomial successfully models various parts of the map. In addition, a global error measure is found by summing over all excluded points. Cross-validation errors are larger than S_u and S_v, because the errors are measured at points not used during optimization.

If the model error exceeds a user threshold the input scan information is written to and "error" XML file. This file can be loaded later to quickly work through scan failures. Otherwise---if model error is acceptably small---the following steps are executed. In either case results of model fitting are displayed as in Figure 6.

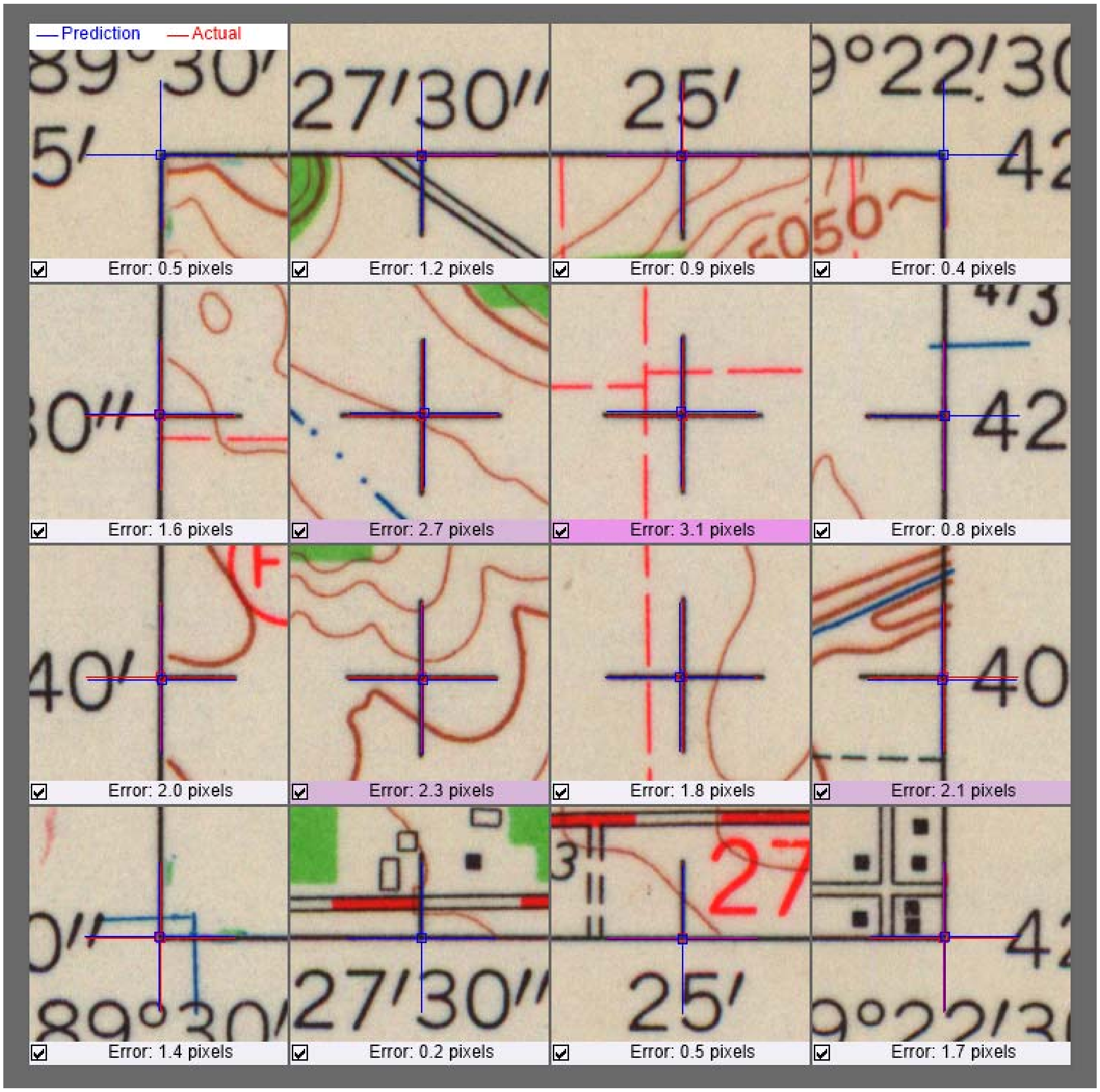

Figure 6. Results from polynomial least-squares fit.

10) **Creation of Output GCP Text (optional)**. If requested, a CSV or XML file is created containing GCP locations and errors for each scan.

11) **Creation of Output Georeferenced Images (optional)**. Requested output images are created by spawning a batch file. The batch file uses GDAL raster utilities with appropriate flags for compression, overviews, etc.

12) **Quality Analysis of Georeferenced File (optional)**. If available, an output image is opened and windows surrounding all GCPs are extracted. Each window is searched for a control mark. Pixel distances between the locations found and the locations expected from the output SRS are noted. The results are displayed and--if requested—saved as a png file for later visual inspection (see Figure 7).

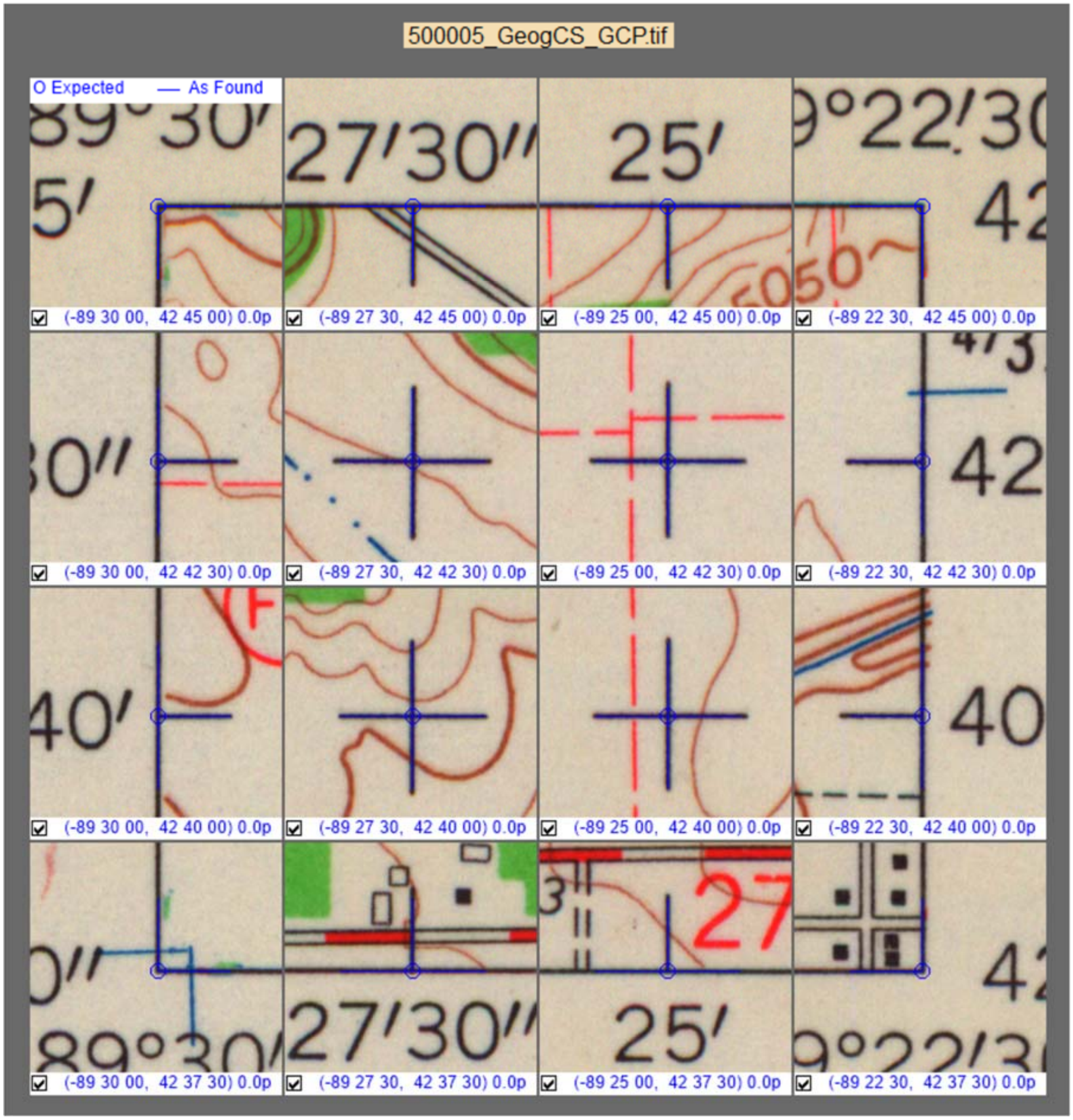

Figure 7. Quality Analysis Display

Using Quad-G+

Quad-G+ is a standard Windows™ application. All of its functionality is accessed through menus and buttons displayed on the main screen (Figure 8). The main screen is also used to display diagnostic text and graphical elements enabling a user to monitor progress on a batch of images. This section describes how to use and interpret Quad-G+ features.

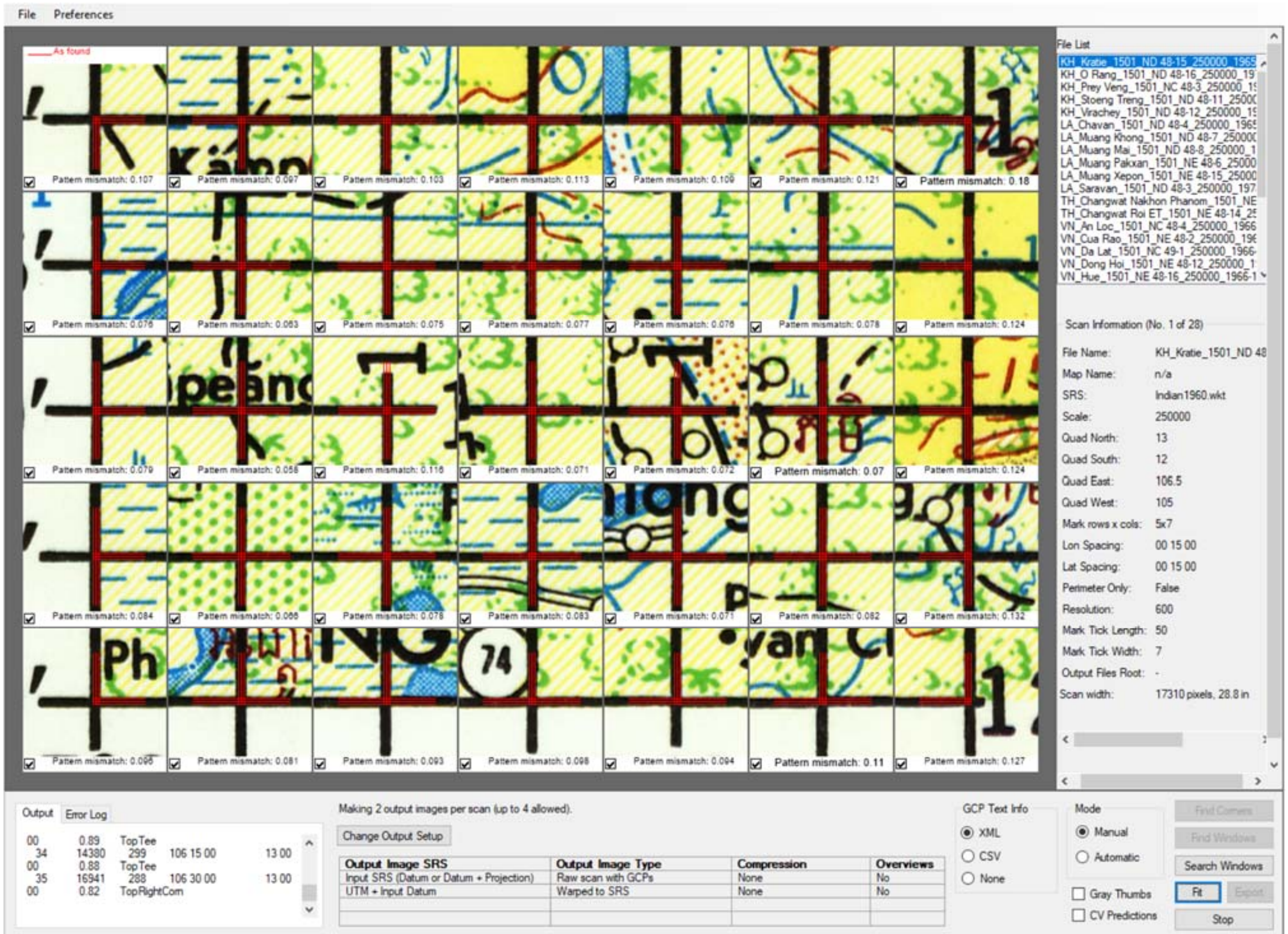

Figure 8. Main Program Screen.

File Menu (Figure 9). This menu is used to open an input XML file, input parameters for a single scan file, and to exit the program. Typically a user will load an XML file for a batch of scans, which has the effect of populating the "File List" box seen in the upper-right corner of Figure 8. The list can then be processed with no operator intervention in automatic mode, or individual files can be highlighted and processed in manual mode. Typically a user will first employ automatic mode for a batch, and then return to any problem files in manual mode. The single scan option provides an alternative to input via an XML input file. In this case a dialog opens allowing for direct entry of scan file parameters (see Fig 9b).

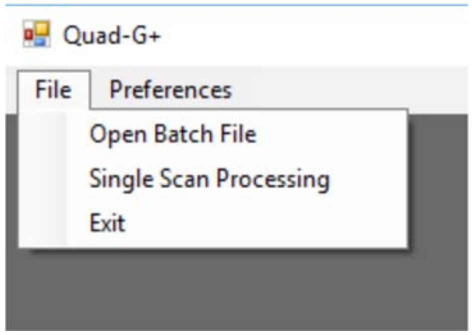

Figure 9. File Menu

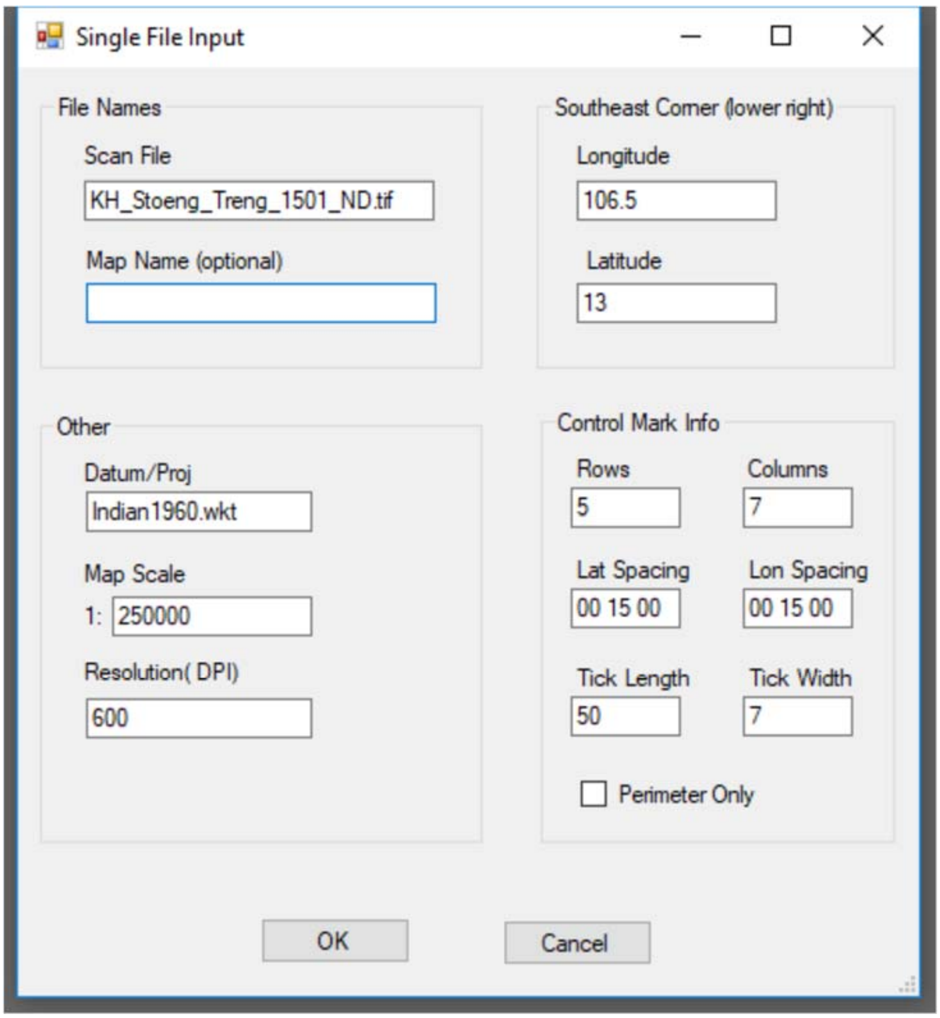

Figure 9b. Direct Input Dialog

Preferences Menu and Dialog. Opening the Preferences menu exposes the dialog shown in Figure 10.

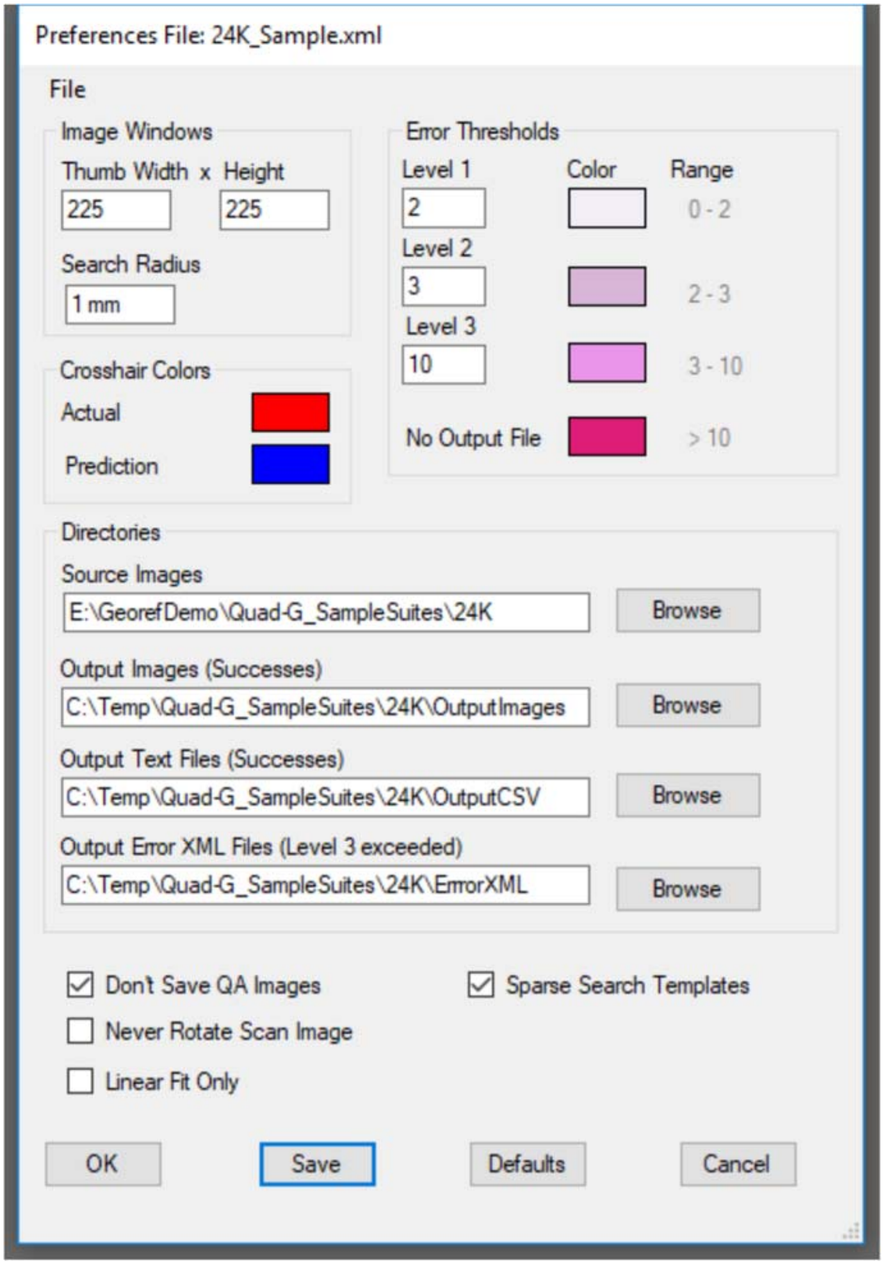

Figure 10. Preference dialog.

The dialog's File submenu is used to read an existing preferences XML file. This allows settings saved previously to be easily re-established. The File submenu also provides for saving a modified suite of preferences as a new file. Meanings of the settings are as follows:

> **Thumb Dimensions:** controls the display size of search windows. Can be specified in pixels, millimeters (use "mm") or inches (use "in"). If the thumbnail size is too large for all search windows to fit in the panel, the display size is

adjusted downward. However, the search radius is unaffected by this. It remains at the value shown.

Search Radius: determines size of image subareas searched for control marks. The default search radius is 2 mm, meaning an area 4mm x 4mm is searched. Large high-resolution scans sometimes benefit from a larger window size in order to find a control mark. Smaller search radii result in somewhat faster searches and fewer false matches.

Crosshair Colors: Sets color of cross-hairs used to display search window locations and model predictions.

Error Thresholds and Colors: model errors are flagged with colors indicating increasing levels of severity. Colors are changed by clicking the color boxes. Values are changed by keyboard entry in edit boxes. Levels 1-3 are simply visual cues. By contrast, the "No Output" level determines whether or not an output image is produced. A georeferenced image will be generated only if all errors are below the threshold.

Source Image Directory: location of input images

Output Image Directory: location of output georeferenced images and quality analysis images.

Output Text Directory: location of CSV or XML files for successfully transformed images (errors below threshold). Control mark locations, transformation errors, etc. are placed in these files. The file format is determined the "GCP Text Info" group on the main form.

Output Error XML Directory: location of XML information for images whose maximum error threshold was exceeded. Information for all scans is put in a single file. This file can be opened later as an input file for individual manual processing of problematic scans.

Don't Save QA Images: Quality analysis is performed and displayed (Figure 7), but not saved to the output image directory.

Never Rotate: disable automatic detection and rotation of transverse scans.

Linear Fit Only: use linear rather than quadratic polynomial regardless of control mark count. This can prevent unrealistic extrapolation in areas far from control marks.

Sparse Search Templates: For tick thickness greater than 1 pixel, the default is to use filled template search patterns. A sparse template uses fewer pixels for pattern matching, resulting in faster searches. Compare left and right of Figure 10 below.

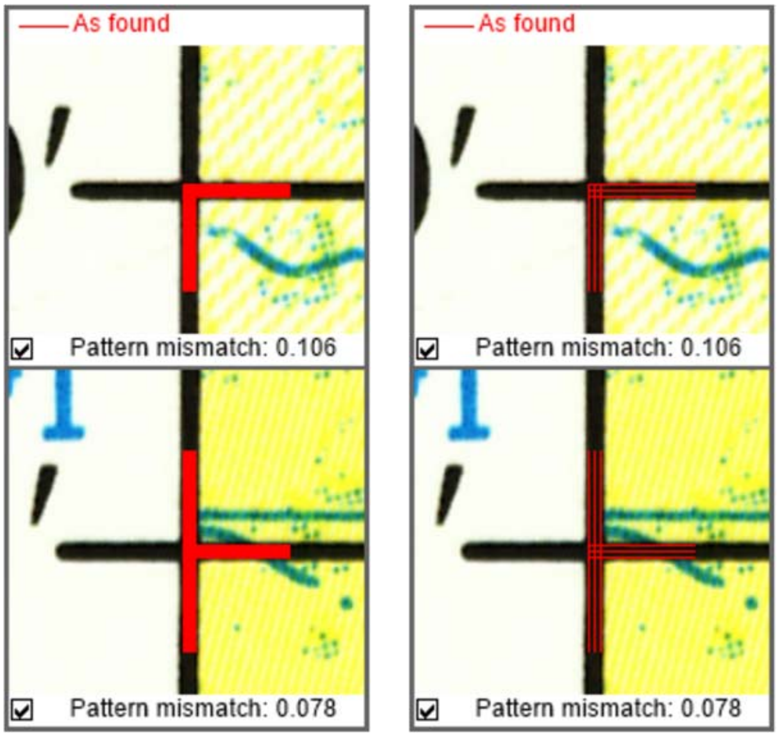

Figure 10. Filled and sparse templates, both with a 7-pixel tick thickness.

OK Button: Apply changes to the current Quad-G+ session and close dialog. Changes are not saved in the external preference file.

Save Button: Apply changes and save settings to the current preference file.

Defaults Button: Restore default values for all settings. Dialog remains open.

Cancel: Close the dialog without using or saving changes.

Process Buttons: This group, shown in Figure 11, invokes the processing steps discussed above.

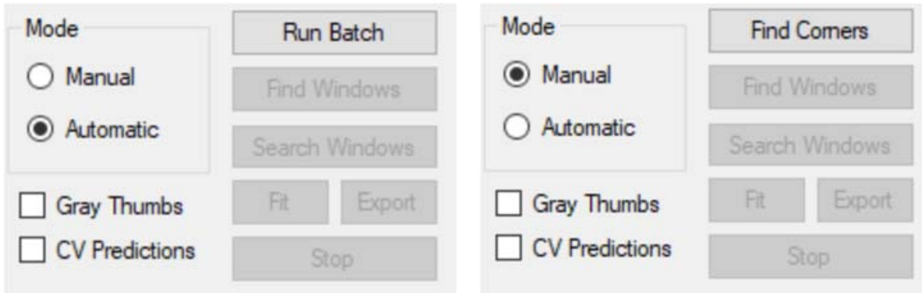

Figure 11. Process buttons for automatic and manual modes.

Automatic Mode. In automatic mode the user clicks "Run Batch" and the full sequence is of steps executed for the batch starting with the selected file and continuing to the end of the file list. After a batch starts the "Stop" button becomes enabled and can be used to interrupt the process (Figure 11, left).

Manual Mode. In manual mode (Figure 11, right) a user selects an input image and then progresses through the sequence of processing steps by clicking a button for each step. Buttons are enabled when the corresponding processing step is appropriate. The "export" button generates requested output image and text files.

Gray Thumbs: displays scan image data as gray-scale rather than full-color in thumbnail windows. This has no effects on scan processing.

CV Predictions: Mark model errors are reported as Cross-Validation errors in thumbnails.