

Automated designation of tie-points for image-to-image coregistration

R. E. KENNEDY

Department of Forest Science, 321 Richardson Hall, Oregon State University,
Corvallis, OR 97331, USA; e-mail: robert.kennedy@orst.edu

and W. B. COHEN

USDA Forest Service, Pacific Northwest Research Station, Forestry Sciences
Laboratory, Corvallis, OR 9733, USA; e-mail: warren.cohen@orst.edu

(Received 3 May 2001; in final form 13 June 2002)

Abstract. Image-to-image registration requires identification of common points in both images (image tie-points: ITPs). Here, we describe software implementing an automated, area-based technique for identifying ITPs. The ITP software was designed to follow two strategies: (1) capitalize on human knowledge and pattern-recognition strengths, and (2) favour robustness in many situations over optimal performance in any one situation. We tested the software under several confounding conditions, representing image distortions, inaccurate user input, and changes between images. The software was robust to additive noise, moderate change between images, low levels of image geometric distortion, undocumented rotation, and inaccurate pixel size designation. At higher levels of image geometric distortion, the software was less robust, but such conditions are often visible to the eye. Methods are discussed for adjusting parameters in the ITP software to reduce error under such conditions. For more than 1600 tests, median time needed to identify each ITP was approximately 8 s on a common image-processing computer system. Although correlation-based techniques—such as those implemented in the free software documented here—have been criticized, we suggest that they may, in fact, be quite useful in many situations for users in the remote sensing community.

1. Introduction

When two images of the same area are to be analysed, they must be geometrically registered to one another. Image-to-image registration involves: (1) the identification of many image tie-points (ITPs), followed by (2) a calculation based on those points for transforming one image's pixel coordinates into the coordinate space of the other image, followed finally by (3) a resampling based on that transformation. While steps 2 and 3 are relatively standardized processes, the manual identification of image tie-points may be time-consuming and expensive, and includes bias of the interpreter. For these reasons, automated techniques for detection of ITPs have been developed that are potentially cheaper and repeatable.

The variety of automated registration techniques has been amply summarized elsewhere (Bernstein 1983, Brown 1992, Fonseca and Manjunath 1996, Schowengerdt 1997). Briefly, the techniques to identify tie-points in two images fall into two main

categories: area-based matching and feature-based matching. Area-based matching approaches compare spatial patterns of pixel grey-scale values (digital numbers) in small image subsets (Pratt 1974, Davis and Kenue 1978). Match-points between image subsets are located by maximizing some function of similarity between an image subset in one image and the image subset in another image. The function of similarity may be a modified difference function or a more complex correlation function (Bernstein 1983, Schowengerdt 1997). Variations on this basic theme exist that seek to decrease computation time (Barnea and Silverman 1972) or to compute a simultaneous solution for multiple points, rather than using sequential image subsets (Rosenholm 1987, Li 1991). In contrast to area-based matching, feature-based matching does not attempt to directly match grey-scale pixel patterns between the two images. Rather, pattern-recognition algorithms are applied to grey-scale pixel patterns within each image separately to identify prominent spatial features, such as edges (Davis and Kenue 1978, Li *et al.* 1995) or unique shapes (Tseng *et al.* 1997). These features are described in terms of shape or pattern parameters that are independent of the image grey-scale values (Goshtasby *et al.* 1986, Tseng *et al.* 1997). The separate features in the two images are then matched based on those shape and pattern parameters.

Although these techniques have been in existence for decades and are well-documented in image-processing texts (Pratt 1991, Schowengerdt 1997), many users of satellite data still rely on manual designation of tie-points. To assess use of automated techniques, we surveyed every article published in the *International Journal of Remote Sensing* for the year 2001 (i.e. all of volume 22, not including the shorter communications in the *Letters* section). Of 206 articles surveyed, 59 included methodology that required co-registering of images, and thus could have benefited from the use of an automated approach. Only five of those (~8%) used some type of automated approach for locating ITPs, while 24 (~40%) appeared to use manual co-location of points (table 1). Another 13 studies (~22%) stated that they co-registered images, but provided no detail how they did so. However, we infer from the lack of attention paid to describing geo-registration methodology that most or all of these studies did not use techniques other than the traditional manual approach. Thus, it appears likely that over 60% of the studies used manual ITP identification, while fewer than 10% of the studies used an automated approach. The remaining 30% of the studies manually registered both images to a common system, which is another situation where automated approaches could have been used. In all, more than 90% of the studies that could have used automated techniques did not.

Table 1. Tallies of *International Journal of Remote Sensing* 2001 articles that co-registered images, broken down by the method used to identify ITPs.

Image co-registration method	Number of papers
Images separately registered to common system	17
Manual ITP selection	12
Manual ITP selection inferred, but not explicitly stated	12
Automated approach	5
Method not stated	13
Total	59

The cause for this avoidance of automated techniques is unclear. Certainly, reviews of automated techniques have stressed the weaknesses of both categories of automated tie-point designation. Area-based methods require significant computer time (Barnea and Silverman 1972), may only deal with translational offsets (Pratt 1974), and may not be useful for data fusion between optical and non-optical sensors (Rignot *et al.* 1991, Li *et al.* 1995). Feature-based methods require the existence of robust features that are relatively invariant between images, and also require use of sophisticated feature-extraction algorithms (Fonseca and Manjunath 1996). From the perspective of an applied user of satellite data, it may appear that the automated detection of tie-points holds more risk than potential reward.

We suggest that many applied users of remote sensing data could benefit from a simple automated approach. In this paper, we describe a publicly-available software package we have developed that uses a simple area-based approach for automated detection of image tie-points. We base the software on two strategies: (1) capitalize on user knowledge and pattern recognition ability, and (2) favour robustness to a variety of image conditions over optimality in any one condition. The former strategy eliminates the greatest unknowns and the costliest components of automated image-matching. The latter relies on the high speed of today's computers to overcome the time-consumption of the area-based approach. In addition to describing the software, we report on tests designed to characterize its usefulness under a range of potentially limiting conditions.

2. Methods

2.1. Automated location of tie-points

Locating each tie-point is a three-step process. In the first step, an approximate tie-point pair is defined in the two images. The approximate tie-point pair is fed to the area-based correlation module that derives the precise image tie-point pair. Finally, the new pair is validated. The next cycle begins by referring to the position of the previous tie-point pair.

Step 1: Locating approximate tie-point pairs. For each image pair, the user supplies the coordinates of the first approximate tie-point pair. This simple user input follows the strategy of capitalizing on human input. The match need not be a true image tie-point, only an approximation to seed the process (see section 4.7 for a brief discussion of this strategy). After the first image tie-point is found and validated (Steps 2 and 3 below), the program automatically locates the second and all subsequent approximate tie-point pairs along a regular grid pattern, working systematically left to right and up to down within the images. The spacing of the grid is adjustable.

For the grids to be scaled properly, the true pixel size (pixel-centre to pixel-centre distance on the ground) of each image must be known, as well as the rotation of the images relative to each other. The user supplies both of these values. Tests of the sensitivity of the process to errors in these estimates are described in section 2.2. Again, the rationale for requiring this user input is based on the strategy of capitalizing on human strengths: the human user likely knows this basic information about the two images, and even if these two values are not already known, they can be approximated easily from simple trigonometry. This is far more efficient than designing an iterative procedure to derive these values automatically under any set of circumstances.

Step 2: Locating potential tie-point pairs. An area-based matching approach is the core of the method. A small subset (dimensions W and H , in units of reference image pixels, adjustable by the user) is extracted from each of the two images (figure 1). The subsets are centred at the coordinates of the approximate tie-point pair supplied from Step 1. If the user has specified that the true pixel size differs between the two images, or that the images are rotated relative to one another, the program adjusts the input image subset to match the size and orientation of the reference image subset.

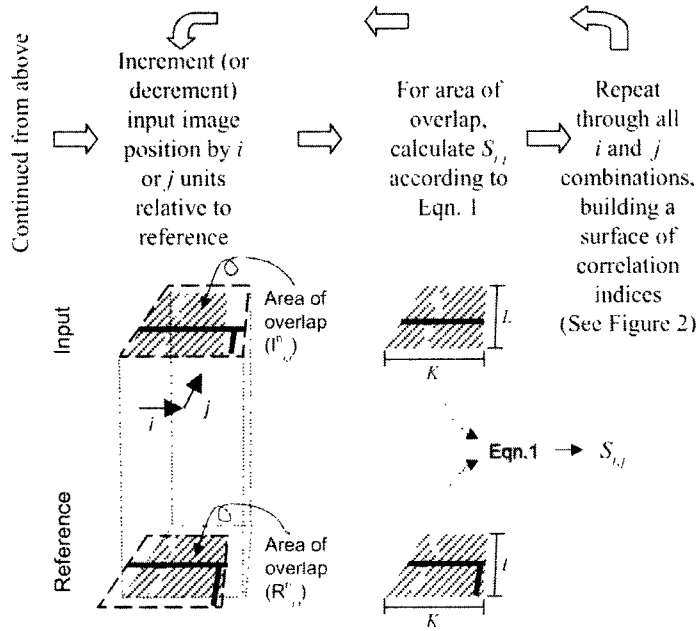
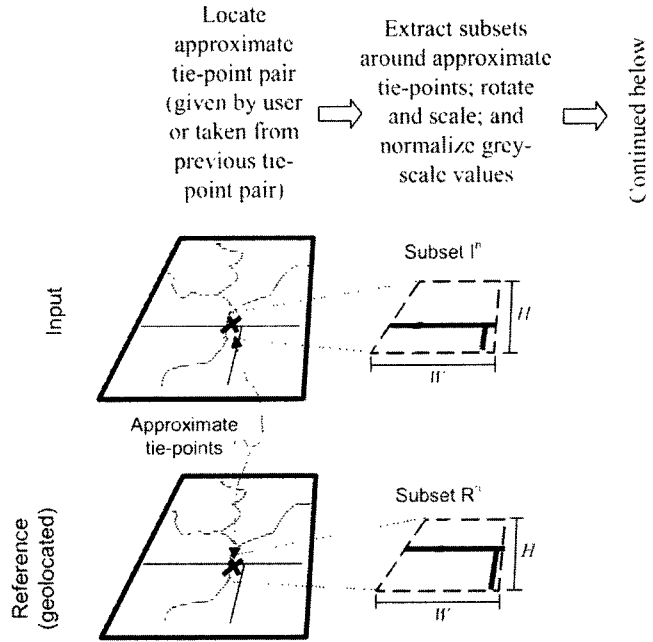
Once subsets have been defined, the process is identical to that described in Schowengerdt (1997). Conceptually, the reference image subset is anchored on a plane while the input image subset floats over it in incremental i and j pixel offset distances. At each offset, the pixel values in the intersection of the two subsets are used to calculate a spatial similarity index. As implemented in the routine, the two image subsets (hereafter \mathbf{R} and \mathbf{I} , for reference and image subsets, respectively) are first standardized to a mean of 0.0 and a standard deviation of 1.0, resulting in \mathbf{R}^n and \mathbf{I}^n . For each i, j offset, the intersection regions are identified as $\mathbf{R}_{i,j}^n$ and $\mathbf{I}_{i,j}^n$, and the mean values of the intersection regions are calculated ($\overline{\mathbf{R}_{i,j}^n}$ and $\overline{\mathbf{I}_{i,j}^n}$). For the pixels in the intersection, the following similarity index is calculated:

$$S_{i,j} = \frac{\sum_{k=1}^K \sum_{l=1}^L (r_{k,l} \cdot i_{k,l}) - (\overline{\mathbf{R}_{i,j}^n} \cdot \overline{\mathbf{I}_{i,j}^n})}{N} \quad (1)$$

where K and L are the dimensions of the intersection, $r_{k,l}$ and $i_{k,l}$ are grey-scale values of the normalized reference and input image subsets, respectively, at position k, l of the overlap area, and N is the number of pixels in the overlap area ($= K \cdot L$). For computational reasons, $S_{i,j}$ is formulated slightly differently than the normalized cross correlation (Fonseca and Manjunath 1996). Nevertheless, because the input images are standardized to mean 0.0 and standard deviation 1.0 before calculating the similarity index, the expected range of $S_{i,j}$ is still -1.0 to $+1.0$ (this result is similar to the observation that the square of standard normal distribution is a chi-square distribution with expected value of 1.0). The calculation of $S_{i,j}$ for all i, j pairs creates a surface of similarity index values (figure 2). The i, j corresponding to the peak of this surface marks the offset with the best match. Adjustments are made to the offset values based on image rotation or pixel size differences, and then the offset values are related to the original approximate tie-point coordinates of the two image subsets to yield the precise coordinates of the potential image tie-point pair. If no peaks are identified, the range of i, j offsets is doubled and peaks sought over the larger similarity surface. If range doubling happens twice with no success, the area is abandoned and the program moves onto the next approximate point.

An area-based approach allows sub-pixel scale estimation of match points. A common method is to fit a polynomial surface to the similarity index values and interpolate between integer i and j offsets (Bernstein 1983). However, in our examination of this technique we found that no rule for surface fitting was generically applicable across different image-to-image registration situations. Following our goal of robustness, we chose instead a simpler method for subpixel estimates: if the user desires subpixel matching, we simply interpolate the original digital image subsets

Figure 1. Schematic diagram of steps involved in calculating a similarity index surface (Step 1 and the first portion of Step 2 in the text, section 2.1).



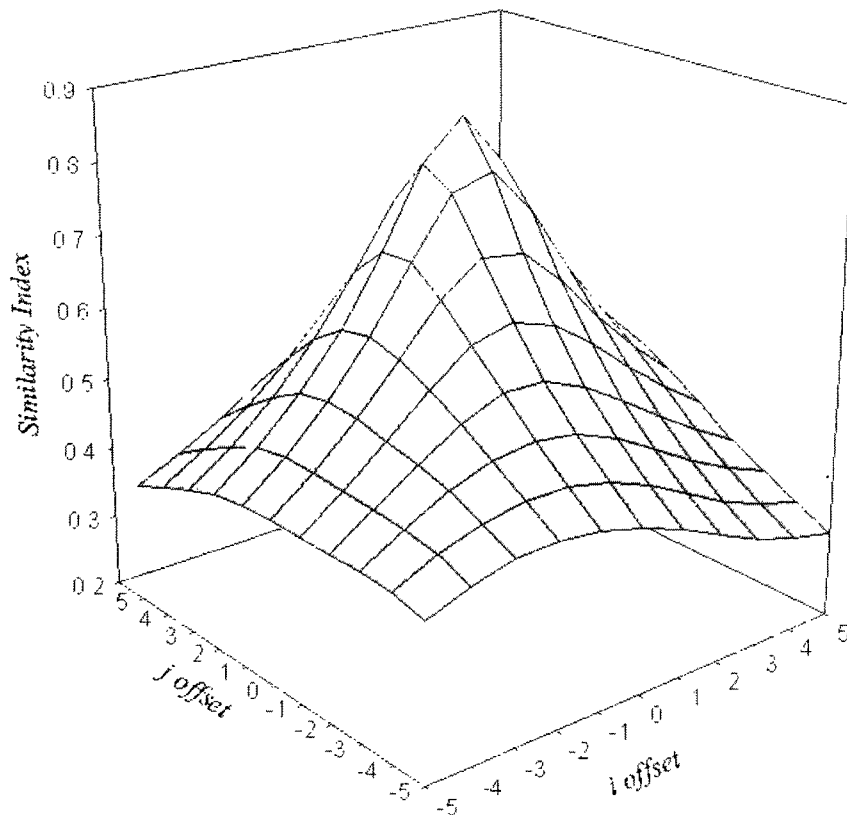


Figure 2. An example of a similarity index surface. The peak of this surface occurs at the i, j offset where the two subsets have the greatest similarity. Here, it is at $i=1, j=2$, indicating that the two image subsets are offset by 1 pixel and 2 pixels in the X and Y dimensions, respectively.

and re-calculate the similarity surface with i, j offsets corresponding to sub-pixel distances. This method is substantially less efficient, but appears to be much more robust across image conditions.

Step 3: Testing potential tie-point pairs. Because not every similarity peak necessarily corresponds to a meaningful image match, screening of points is the critical process in any automated tie-point detection approach. In our package, we use two tests for each point.

The first test is based solely on the similarity index value for the two image subsets. A tall peak in the similarity index is considered a good match. The height of the peak is defined as the distance from the peak to the plane describing the base of the peak. That plane is interpolated between the inflection points of the slope surface in each of the four cardinal directions in i, j offset space. Importantly, the height is scaled to the total range of similarity index values in the tested window to yield a score for the peak (with a range of 0.0 to 1.0). If the peak's score surpasses a defined threshold (an adjustable parameter), it is deemed valid. Because the height

of the peak is scaled to the values observed in each subset image, a general threshold can be used that appears to be robust for many different image conditions. If two or more peaks are determined to be valid with this test, the highest peak is accepted only if it exceeds the second-highest peak by a determined proportion. Based on the i, j offset of the peak, the coordinates of the image tie-points are calculated and a second test is performed.

The second test is based on the relationship between the newly-matched point and the body of existing tie-points. The distance between any two tie-points can be measured either in input image pixel counts, or in reference image map units. Dividing the former distance by the latter distance yields a quantity we refer to as the *pixel size ratio*. Unless there is severe distortion in one image, this ratio should be relatively similar across all tie-point pairs. For the second test, then, we estimate this ratio using only pairs involving the new point and compare that to the estimate using only pairs *without* the new point. If the comparison suggests that the new point is not from the same population (as determined by a user-alterable threshold), the point is discarded. Note that this second validity test can only be performed when a large enough sample of pre-existing image tie-points has been built, and thus cannot be applied to the first few image tie-points located by the methodology. In every run of the software reported in section 2.2, the second validity test was only used once 10 pairs of tie-points had accumulated.

Although the double-testing process screens out most poor ITPs, a low percentage of sub-optimal pairs may still persist. Before a final matrix of polynomial transformation coefficients is calculated from the points, human screening is advised as a last check. An initial model for the geometric projection solution is calculated and positional error calculated for all points. Those points with unusually high contributions to the error of the model should be investigated to determine if they are blatantly incorrect and should be screened. As described in section 2.2, it is also possible to automate this screening process.

Both the ITP software and the testing software were written in IDL v5.2 (Research Systems, Inc., Boulder, CO). They are available at <ftp://ftp.fsl.orst.edu/pub/gisdata/larse/itpfind.tar.gz>.

2.2. Method for evaluating the ITP location software

Ideally, the above procedure should be capable of locating ITPs on image pairs when there are several confounding factors. Broadly, those factors are: (1) the images are geometrically distorted relative to each other; (2) image metadata supplied by the user is not perfectly accurate; and (3) some parts of the images are obscured by clouds or have changed. To evaluate the influence of these factors, we developed a procedure to quantify error in the ITP process run on any image-to-image pair. We could then apply a confounding factor to an image pair, run the ITP software, and determine error. By repeating this across many levels of confounding factors, we were able to build a general evaluation of the conditions under which the software was appropriate.

The procedure to quantify error in a single image pair had three steps. In the first, an input image was created from a reference image. Reference images from five different landscapes were used, as described in section 2.2.1 below. Reference images were 601 by 601 pixels in size. Input images were created by projecting the reference image with a polynomial transformation coefficient matrix and resampling with a nearest-neighbour rule. By creating the input image from the reference image, we

were able to strictly control the types of confounding factors present in the matching process. Geometric distortions were applied by appropriate alteration of the transformation matrix prior to resampling. Landscape changes and cloud effects were simulated by replacing portions of the input image with randomly-placed solid disks of known density, size, and grey-scale value. Simulated inaccuracies in image metadata were passed to the ITP process along with the input image.

The second testing step involved running the automated ITP process to locate ITPs and then deriving a transformation matrix from those points. For this research, the subset image size was set to 60 reference pixels, with spacing of 80 reference pixels between points, resulting in a grid with a maximum of 60 ITP pairs. To facilitate the large number of tests, the human post-processing screen was replaced with an automated approach. In this approach, an initial transformation solution was derived after all ITP pairs had been designated. The contribution of each point to the total positional error was assessed. The point with the highest contribution to the error was eliminated and the transformation matrix re-calculated. Again, the point with the highest error was eliminated. The process was continued until the total root-mean square error for all points was less than one reference pixel. In cases where there were too few ITPs to calculate a valid transformation matrix, the entire run was declared invalid. This would only occur when the software had failed to adequately locate stable points, and thus the number of successful test runs served as an indicator of robustness of the ITP-locating process.

If the run was valid, the final derived transformation matrix was then used to project the input image back into the coordinate system of the original reference image. The distance between each pixel in the recreated reference image and its position in the original reference image was the error of position for that pixel. The average of all errors of position was calculated as the true root-mean-square error of position (true RMS error). The true RMS served as the final and most useful measure of the effectiveness of the software.

Several geometric distortions were used in the first step of the testing process. The first distortion involved rotating the input image relative to the reference image (ranging from 0 to 14°, in increments of 2°; compare figure 3(a) to figure 3(b)). The second was a simple scaling effect, where the input image was resampled to contain pixels of twice the true pixel size of the reference image. In both cases, the software was supplied with accurate information about these confounding factors, both of which regularly occur in real-world applications. These tests represent validation that the software will function when the user can supply accurate information about the relative characteristics of the images. Additionally, we tested two geometric distortions that cannot be specified in the software. The first, referred to here as *skew*, is the condition where true pixel size is a linear function of the pixel position. Visually, a skewed image appears compressed on one end and expanded on the other (figure 3(c)). The second distortion is similar to skew in that pixel size is a function of position, but in this case the function is curvilinear. The image appears bowed (figure 3(d)). It will be referred to as *warping*. The maximum of either effect was the condition where the skew or distortion would either completely compress or bow adjacent corners of the image. The magnitude of both effects was quantified as a proportion of a maximum (0.0 for no effect, 1.0 for maximum effect). We tested values from 0.02 to 0.18 for skew and -0.10 to 0.10 for warping.

The second type of confounding factor was inaccurate geometric information supplied by the user to the software. The first test involved inaccuracies in the 'true

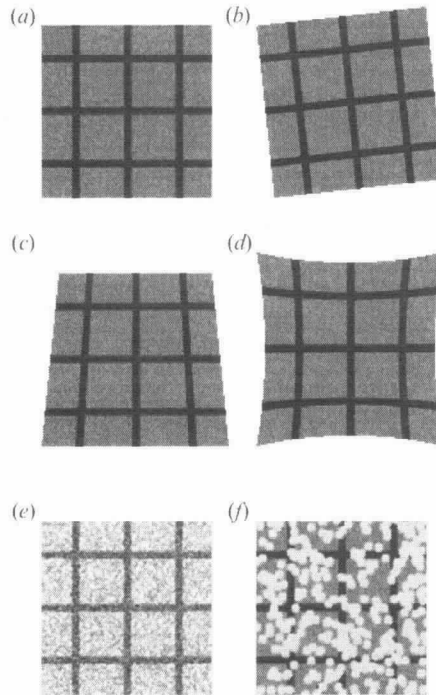


Figure 3. Examples of some of the confounding factors tested, using (a) an example image with no confounding factors as a base. Shown is the example image with: (b) rotation of 6° ; (c) skew of 0.10; (d) warp of 0.10; (e) noise of 2.0; and (f) change density of 50% and change level of 1.5.

pixel size' parameter (the pixel-centre to pixel-centre distance) of the input image. Here, the input image was maintained at the same pixel size as the reference image, but the program was incorrectly informed that the value for input image was either larger or smaller than that of the reference image. Hence, a 'true pixel size distortion' of 1.10 meant that the supplied parameter value for input image true pixel size was 110% that of the reference image. Tested values ranged from 0.85 to 1.20. The second type of inaccurate geometric information involved rotation of the input image. The input image was rotated relative to the reference image, but the program was informed that the two images were unrotated. This effect was tested for rotation mismatches of 1 to 10° .

The final type of confounding factor was change between input and reference images. Atmospheric noise was simulated by additive random noise (figure 3(e)). Coherent change was simulated by replacing portions of the input image with flat grey disks. The percentage of the surface area of the image covered by the disks varied from 0 to 50%. In one set of tests, the grey-scale value of the disks was set at 150% of the mean value of the image, simulating landscape changes (figure 3(f)). In a second set, the value was 250% of the mean value of the image, simulating bright objects such as clouds.

2.2.1. Sources of imagery

Because the ITP process is based on spatial pattern analysis, the spatial pattern in test images may be important. Therefore, we repeated all tests on Landsat TM images from five landscapes with different spatial patterns (see figure 4). From each landscape, we extracted a region approximately 1500 pixels square where the spatial patterns were relatively consistent. When each test was run, a reference image of 601 by 601 pixels was extracted at a random location within that larger region.

For each combination of landscape and confounding factor, seven repetitions of the entire testing process described in the previous paragraphs were conducted. For example, the influence of unknown rotation was tested across rotations of 1 to 10° with a single degree step, making 10 levels of the confounding factor. At each degree increment, the test was repeated seven times for each of the five landscapes, making 35 tests per degree increment, 350 tests in total, with a possible maximum of $350 \cdot 60 = 21\,000$ image tie-point pairs.

3. Results

3.1. Geometric distortions

Figure 5(a) shows how the ITP software performed for the conditions where the input image was rotated relative to the reference image, and that rotation was passed correctly to the ITP software by the user. The figure shows the average true RMS error for tests at rotations from zero to 14°. RMS error is reported in units of pixels.

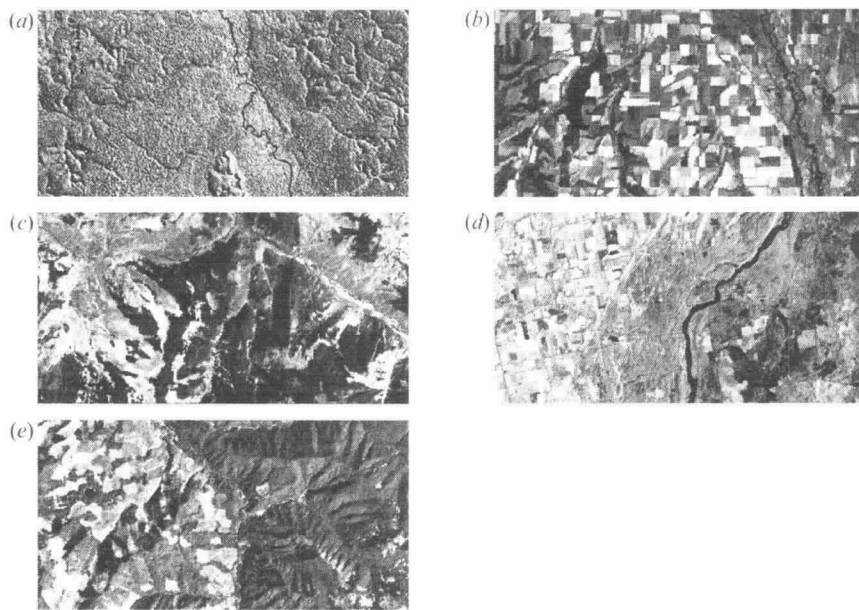


Figure 4. Representative regions from the five Landsat TM images used for testing, showing: (a) undisturbed tropical rainforest, southeast of Santarem, Brazil; (b) large-field agriculture in Gallatin Valley, Montana, USA; (c) large-block conifer forest and high-altitude open areas, Gallatin Mountains, Montana, USA; (d) small-field agriculture, hardwood, and mixed forest, St Croix River valley, Minnesota and Wisconsin, USA; (e) conifer forest with clearcut openings, Blue River, Oregon, USA.

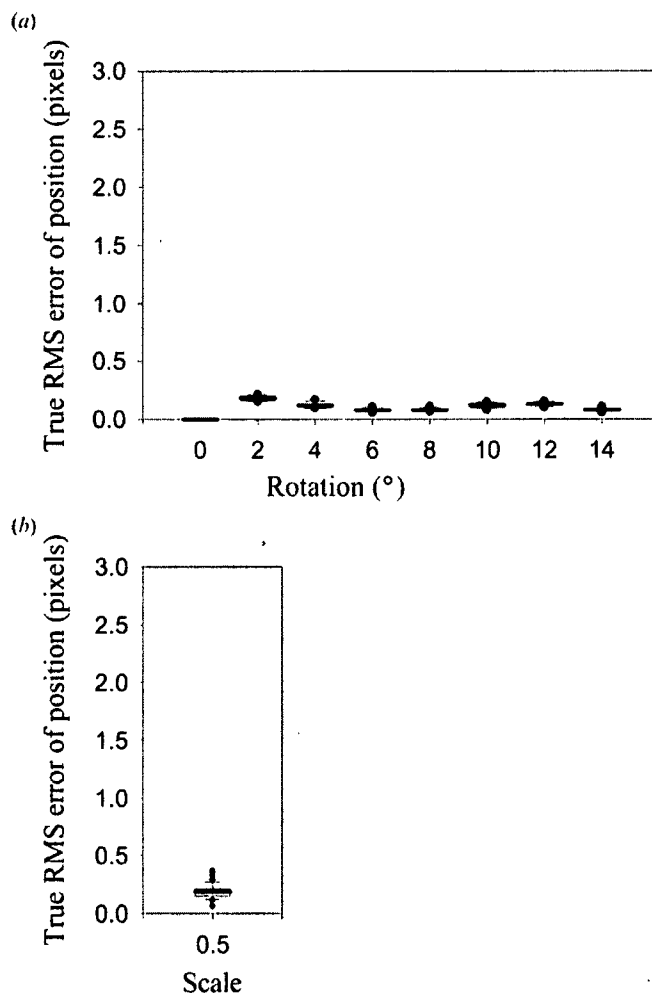


Figure 5. Test results when rotation and scale were accurately passed to ITP software: (a) for rotation of the input image from 0 to 14°; and (b) for pixel scale of 0.5, analogous to MSS to TM matching. True RMS error of position is the mean positional error (in pixels) of all pixels after being resampled using ITPs designated with the software. Thick horizontal line indicates mean value; thin horizontal line inside shaded box indicates median value; boundaries of shaded box represent 25th percentiles away from median; whiskers extend to 40th percentiles away from median; individual dots represent tests falling outside those percentile ranges. Except where noted, distribution statistics were calculated from $n=35$ tests.

In all cases for this figure, all 35 tests for each increment of rotation were successful. As would be hoped, the true RMS error was 0.0 when the images were identical (at a rotation of 0°). Rotations greater than zero had small but non-zero true RMS error. Because nearest-neighbour resampling was used in the creation of the input image from the reference image, slight rotation alters the interpixel geometry and

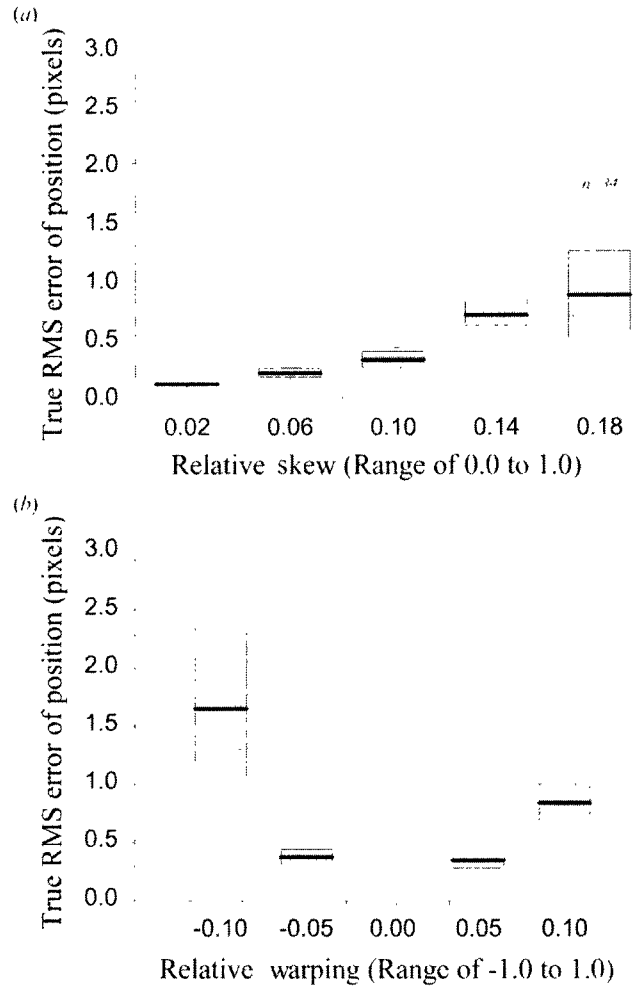


Figure 6. Test results when the input image was distorted relative to the reference image with: (a) skew of 0.02 to 0.18 (see figure 3(c) for an example of skew); and (b) warping of -0.10 to 0.10 (see figure 3(d) for an example of warping). Thick horizontal line indicates mean value; thin horizontal line inside shaded box indicates median value; boundaries of shaded box represent 25th percentiles away from median; whiskers extend to 40th percentiles away from median; individual dots represent tests falling outside those percentile ranges. Except where noted, distribution statistics were calculated from $n=35$ tests.

prevents zero-error co-registration. Figure 5(b) shows the condition where input image pixels had a true pixel size twice that of the reference image, and where that information was accurately passed to the ITP software. Again, true RMS error was small but non-zero.

Skew and warping caused more error. Figure 6(a) shows trends in true RMS error for increasing skew values. As skew increased, both the mean and variance

about the mean increased. For skew values of 0.02 to 0.10, the mean and median true RMS error values were below one-half pixel. An image with a skew of 0.10 will have one side with length 10% narrower and the other side 10% wider than the centre of the image, as illustrated in figure 3(c). In relation to skew, warping caused similar but amplified effects (figure 6(b)). At a warping of positive or negative 0.05, true RMS error was moderately low (means and medians less than 0.4 pixels). It was moderately higher for a warping value of positive 0.10, but much higher (greater than 1.0 pixel) for a warping value of negative 0.10. Figure 3(d) provides a sample warped image with a warping value of 0.10. An image with negative warping would have convex rather than concave edges. Recall that for both skew and warping tests, only the input image is distorted; the reference image geometry remains as in figure 3(a).

3.2. Inaccurate input information

The next set of tests involved situations where the user supplied the software with inaccurate information about the images. Figure 7(a) shows how true RMS error changed when true pixel size was inaccurate. When the true pixel size parameter of the input image was claimed to be within 5% of the actual value (true pixel size distortion of 0.95 to 1.05), the error was relatively small (less than 0.2 pixels). But when the distortion was 10%, true RMS error was near or over 0.5 pixels. By the time distortion was 15%, error was large (near 1.5 pixels) and several tests failed (note that successful test counts indicated in the figure are much lower than the possible 35), indicating that there were too few tie-points to adequately calculate geometric transformation matrices.

Undocumented rotation also caused error. In this situation, the input image was rotated relative to the reference image, but the software was not informed of that rotation. When the input image was rotated 5° relative to the reference image, the mean and median true RMS error were under 0.5 pixels (figure 7(b)). When rotation reached 6°, however, variability in true RMS was much higher, and the mean was much larger than the median. Three runs at a rotation of 6° had extremely high true RMS values (> 5 pixels). This is indicative of the breakdown of the software at higher rotations: by 10° of rotation, even the median true RMS value was near 2.5 pixels, and fewer than half of the runs (12 of 35) provided enough valid points to calculate a transformation matrix (figure 7(b)). For reference, figure 3(b) shows an example of an image with a rotation of 6°.

3.3. Change between images

The final tests were designed to emulate the situation where parts of the scene change between images. Additive noise did little to reduce the effectiveness of the routine (figure 8(a)), although a relative noise level of 2.0 confused the software enough to eliminate some tests from validity (only 28 of 35 runs successful). Simulated landscape change also had little effect (figure 8(b)). Even when 50% of the input image was covered with opaque disks masking underlying pattern (i.e. change density of 0.5; see figure 3(e) for an example of change density of 0.5), true RMS error was less than 0.2 pixels. The situation was less predictable when the opaque disks were much brighter than the background scene, as might be the case with clouds (figure 8(c)). True RMS error was much less predictable, and was generally higher, than for the simulated landscape changes. When 10% of the image was obscured with these brighter disks, all runs except one had true RMS error of a third of a

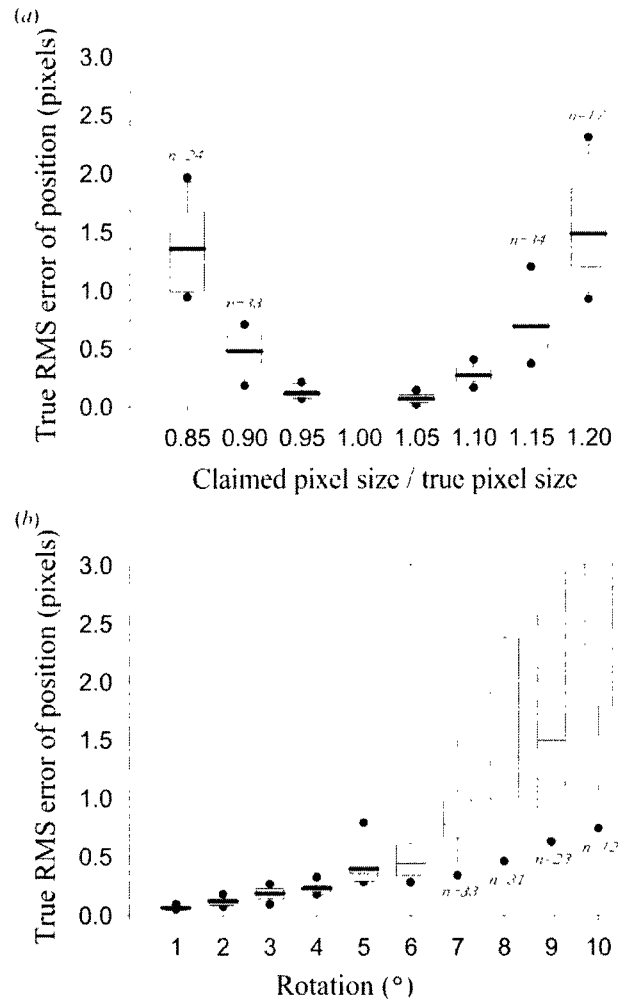


Figure 7. Test results when user-supplied information was inaccurate, for the situations where: (a) the input image pixel size was different than that claimed, indicated here as the ratio between the claimed and true values, and (b) where the input image was rotated relative to the reference image, in units of degrees. Thick horizontal line indicates mean value; thin horizontal line inside shaded box indicates median value; boundaries of shaded box represent 25th percentiles away from median; whiskers extend to 40th percentiles away from median; individual dots represent tests falling outside those percentile ranges. Except where noted, distribution statistics were calculated from $n = 35$ tests.

pixel or less, but one run had a true RMS error greater than 20, causing the mean to be extremely high. When 20% of the image was obscured, there were no such anomalous runs, and the mean and median values were very low. But again at 30%, and continuing at 40 and 50% coverage, several runs had dramatically high true RMS values, which increased mean values far above the medians.

3.4. Time per point

Figure 9 shows a histogram of the number of runs by the average number of seconds per tie point in that run. This figure incorporates all runs referenced in this paper. Recall that each run represents the average of up to 60 image tie points. Median time per point was 7.8 seconds. Because of the highly skewed distribution, the mean was much higher than the median (approximately 27 seconds per point). All runs were conducted on a Sun Ultra-1 workstation (1996 technology).

4. Discussion

4.1. Overview

Correlation-based identification of image tie points has been criticized for being too computationally expensive (Barnea and Silverman 1972, Brown 1992), and for being unable to handle certain types of image distortion (Pratt 1974) and cross-platform matching (Fonseca and Manjunath 1996). While these drawbacks certainly limit its usefulness in some cases, correlation-based matching may be very helpful for many users. Our goal has been to introduce a software package that uses correlation-based matching, and, more importantly, to test its constraints.

The tests provided herein were designed to be illustrative, not exhaustive. Nevertheless, we feel that the confounding factors we tested are an important subset of those that might influence the ITP software. Two unaddressed factors in particular should be considered before using the ITP software. First, to capture spatial pattern, the size of subset windows used to build the correlation indices (section 2.1) should be much larger than the dominant spatial patterns of the images being matched. As their sizes converge, precision will drop. Second, the method of locating maximum spatial correlation requires that dark and light features in one image be correspondingly dark and light in the other. This will not necessarily be the case for images acquired on different platforms, as noted in many reviews of the subject (Rignot *et al.* 1991, Fonseca and Manjunath 1996). In summary, the utility of the method will ultimately depend upon the unique combination of scene spatial features, sensor detection characteristics, and the parameters used to run the ITP software.

4.2. Known and unknown distortions

The tests reported here show that the ITP software can be useful under a wide variety of conditions, although it does have limits. When the images were exactly as the software expected i.e. when the user provided precisely accurate information about the images, the software was accurate and precise (figure 5(a)). The slight pixel-based error due to nearest neighbour resampling is an error source that was present in all of the tests, and was essentially impossible to separate on a test-by-test basis. Despite this error, the total error of position was minimal for both rotation and true pixel size differences, so long as those differences were accurately known (figure 5). This is likely the case for many users of satellite imagery: the images are minimally-distorted, any rotations are systematic and documented, and true pixel size is relatively consistent and well-known.

Even when rotation and true pixel size were slightly inaccurate, error was still relatively small (figure 7). This suggests that the user need only provide rough estimates of these two quantities. Following the strategy of capitalizing on human strengths in pattern recognition, the user can quickly locate two approximate match points and use simple trigonometry to gain these rough estimates.

Users of low-altitude aerial photography or of aerial scanners (such as AVIRIS)

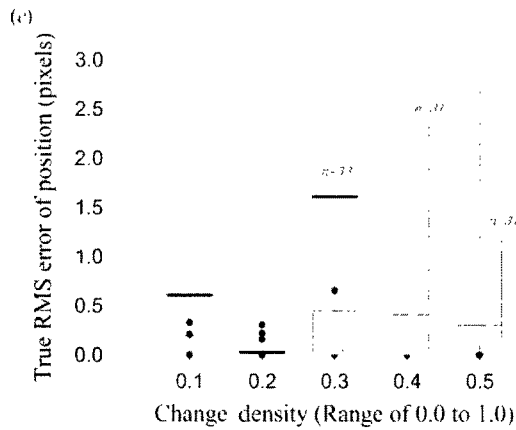
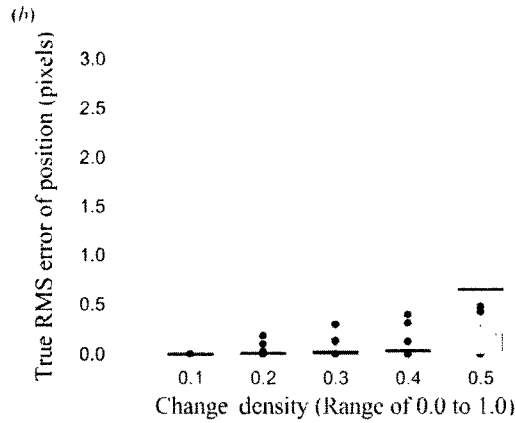
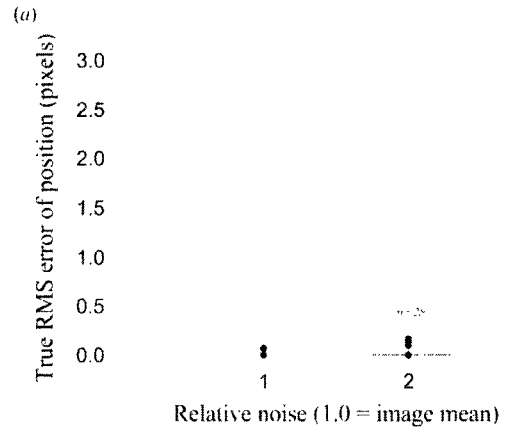
may find the results of the skew and warp tests relevant. Skew was a minor problem at low values, but became a significant issue when the distortion was quite noticeable to the eye (figure 6(a)). Warping was more of a problem than skewing (figure 6(b)). Even at a level of warping that appeared relatively minor (0.10), the true RMS error was considerable (near 1.0 pixel).

The cause of these errors, as well as those caused by inaccurate rotation or pixel size, may be traced to the process of locating points. Recall that an approximate match point is calculated from the prior point based on the supplied pixel size and rotation. If those two parameters are incorrect, the approximate match point will be far from the true match point. If the true match point is outside the range of i and j offsets used for building correlation surfaces (Step 2 in section 2.1), any false correlation peak that occurs by chance within the range of i and j offsets may be designated the true peak. Consequently, anything that causes true pixel size or rotation to be incorrect will increase the probability of false matches, and, hence, the total error of position of the matching. Warping and skewing cause local distortions in the true pixel size and rotation; incorrect user input causes image-wide error in pixel size and rotation.

These effects can be combated relatively easily. One approach is to increase the error tolerance by increasing the range of i, j offsets considered for correlation. The true match point will thus have a greater chance of being located on the correlation surface, and the probability of identifying false peaks will diminish. A second approach seeks to reduce error by reducing inter-point distances. Recall that approximate match points are calculated by multiplying the true pixel size by the desired distance between match points, while adjusting for rotation. As the distance between points increases, any given error in true pixel size or rotation is multiplied over a larger distance. This increases the error of position of the approximate match point, and consequently increases the probability of locating a false correlation peak. By decreasing the span between match points, the errors in true pixel size or rotation are multiplied over shorter distances, thus decreasing error of estimation and increasing the probability that the true match point will lie on the correlation surface. As a secondary benefit, this increases the density of the grid of points, providing a larger sample from which to calculate a transformation matrix. The primary drawback in both approaches is an increase in the total time spent on finding points.

The latter approach was used to match AVIRIS (Airborne Visible Infra-Red Imaging Spectrometer) imagery to TM imagery. As a whisk-broom scanner mounted on an aerial platform, AVIRIS has a different geometric distortion for every pixel.

Figure 8. Test results for situations where change has occurred between images. In part (a), random additive noise was added to input image only. A noise value of 2.0 indicates that the mean value for the noise was twice the mean value of the pre-noise image. In part (b), many small disks were added to the input image. Disk size was 10 reference pixels and the grey-scale value of the disks was 150% of the mean grey-scale value of the image. Change density refers to the proportion of the input image covered by opaque disks (see figure 3 for an example of change disks). Part (c) was as for part (b), but with disks having grey-scale values 250% of the mean grey-scale value of the image. Thick horizontal line indicates mean value; thin horizontal line inside shaded box indicates median value; boundaries of shaded box represent 25th percentiles away from median; whiskers extend to 40th percentiles away from median; individual dots represent tests falling outside those percentile ranges. Except where noted, distribution statistics were calculated from $n = 35$ tests.



As is commonly the case, the ephemeris and surface elevation data necessary to deterministically locate each pixel (such as in Meyer 1994) were not available at the level of precision necessary to accurately locate each point. In this case, the software was parameterized to run with a short inter-point separation. Although computationally expensive, the process was faster and the resultant dense grid of more than 500 tie-points far better than could be achieved manually (data not shown).

4.3. *Change between images*

Low to moderate levels of change between images may cause few problems. The ITP software was robust to many of the simulated scene-changes tested here, especially where the amplitude of the change was relatively moderate compared to that of the average pixel's grey-scale value. It is important to note, however, that only one size of disks was used to simulate change for these tests. It is expected that the spatial scale of the change would interact with the size of the image subset window to affect error. While it is not reasonable to test all permutations of size combinations, exploratory tests on simulated images (data not shown) indicated that this interaction did influence error, but that it did not alter the underlying conclusion that the method is relatively robust to moderate levels of scene change. These results are promising, as they suggest that the method could be used without penalty in change-detection studies.

The change-detection results are encouraging for another situation: where image matching across sensor-platforms is desired. For correlation approaches to function there must be positive correlation across images in response to scene components. The disks we used to test change were completely uncorrelated across images, and still the software was relatively robust. This suggests that even when sensor spectral response is not perfectly correlated across every feature in the image, the software can still provide useful results.

Although the tests reported here show that clouds may cause potential problems, such problems can be circumvented with a relatively simple technique. If clouds are bright enough to cause problems for the software, they can readily be identified by simple thresholding. It is thus possible to set clouds and shadows to the background value, alert the software to ignore those areas in its calculations, and run the software in an otherwise normal mode. This process was used to match TM images from the Amazon. To illustrate the success of the process, a tiled image composite of the input and reference images is shown in figure 10. Note the match of drainage patterns across the tile boundaries.

4.4. *Comparisons to other techniques*

The case of the cloudy images in Brazil illustrates the flexibility of the correlation-based technique. Not only can the software 'work around' the clouds, but it can locate tie-points on a landscape where man-made features like roads are not present. Both of these situations could pose problems for feature-based techniques (Fonseca and Manjunath 1996). Another advantage of an area-based approach over a feature-based approach is in its ability to create regular grids of points. Although strategies for approximating a regular grid with feature-based approaches have been proposed (Davis and Kenue 1978), they are still constrained by the presence or absence of a robust feature in a given grid cell. Moreover, area-based techniques can create grids of tie-points at any arbitrary density.

Another useful technique for tie-point designation is a sequential-similarity

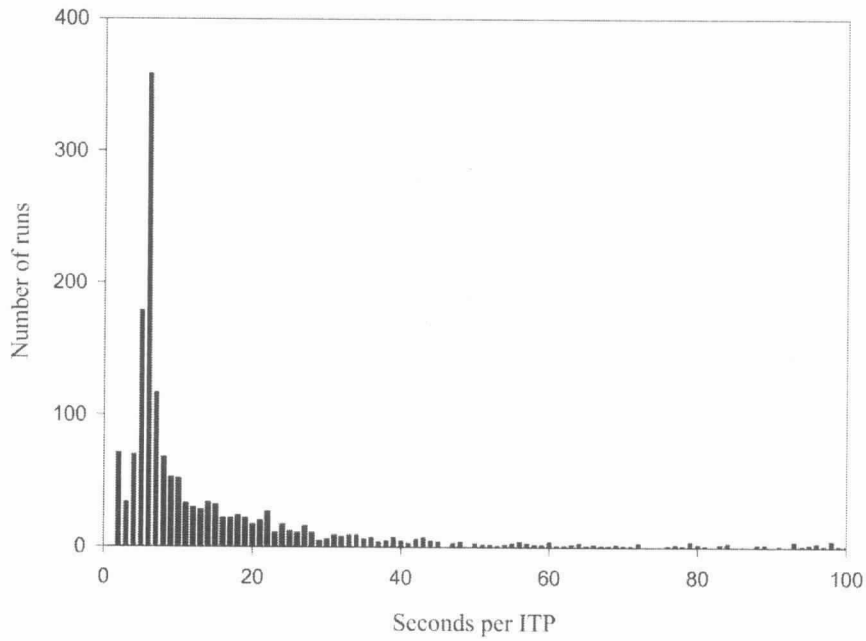
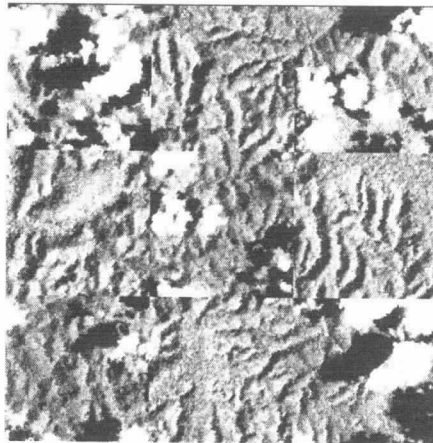


Figure 9. Histogram of the average time needed for each ITP within each run. The average time per run could include as many as 60 ITPs. All runs were conducted on a Sun Ultra-1 workstation with IDL 5.2. Median 7.7 s over 1696 runs.

(a)



(b)

1984	1986	1984
1986	1984	1986
1984	1986	1984

Figure 10. Registration matching example for cloudy images in the Brazilian rainforest, showing: (a) a checkboard of image tiles taken alternately from a 1984 TM image and 1986 TM image registered to it using the ITP software; and (b) the key for the image tiles shown in part (a). Note continuance of drainage patterns, as well as density of clouds and shadows obscuring the surface.

detection algorithm (SSDA; Barnea and Silverman 1972). Like a correlation approach, an SSDA builds a similarity surface for a range of i, j offsets. For a given offset, absolute differences between pixel values are added sequentially in an increasing radius around a random point until a threshold is met. The similarity measure is the number of pixels added before the threshold is met; the matchpoint lies at the maximum. Although clearly faster than a correlation based approach, it may not be as robust across situations. Bernstein (1983) notes difficulty in designating a threshold that is generally applicable. Moreover, because the method includes only a sample of the image subset pixels, it may be more susceptible to isolated anomalous features, such as regions of changed pixels. In the search for a general approach, then, the SSDA may not be as useful as a correlation-based approach.

In contrast to both feature-based approaches and the SSDA strategy, correlation techniques allow for subpixel approximation of matchpoint position (Bernstein 1983). This is again computationally expensive, but may be useful where high precision is required.

4.5. Computational costs

The biggest criticism of correlation-based approaches is their high computational expense. Yet with today's computers, the computational cost of the correlation-based approaches may be small enough to be insignificant for many potential users. Approximately 75% of the runs reported here had a mean time per ITP under 20 seconds, and 90% were under 1 minute. In practice, we have found that new TM images can be accurately, repeatably matched and resampled in under 45 minutes, with user attention required for only a few minutes of that time.

4.6. Examples of applications

We have applied the software successfully in many situations and environments. In fact, for Landsat TM to TM matching, a single set of parameters has been successful for images from the forest, agriculture, and mountainous regions sampled for the tests in this study. We used it to co-register five dates of Landsat TM imagery in forested and agriculture regions of western Oregon (Oetter *et al.* 2001), and to register five years of Landsat TM imagery in Rondonia, Brazil (Guild 2000). By slightly altering the parameters, we have used the software to match TM and MSS imagery in the Rocky Mountains (Parmenter *et al.* in press). We have also used it to register small subsets of TM imagery to panchromatic high-resolution digital imagery (figure 11), to register multitemporal AVHRR imagery to TM imagery (figure 12), and to register images of canopy heights derived from small-footprint LIDAR (Light Detection and Ranging) to those derived from ground measurements (Dr Michael Lefsky, personal communication 2002, Hudak *et al.* 2002).

4.7. Evaluation of strategy

The strategies underlying the ITP software were: (1) require basic user input and (2) choose robustness and flexibility over efficiency, relying on availability of cheap computational speed. These strategies make the software inappropriate for quickly processing large batches of imagery (hundreds of images per day), but entirely appropriate for use in any study where manual interpretation would otherwise occur. The basic user input required is minimal, and can be estimated by the user directly from the imagery with minimal effort. Moreover, if many copies of the same type of imagery are to be processed, the software could be adapted to eliminate the need

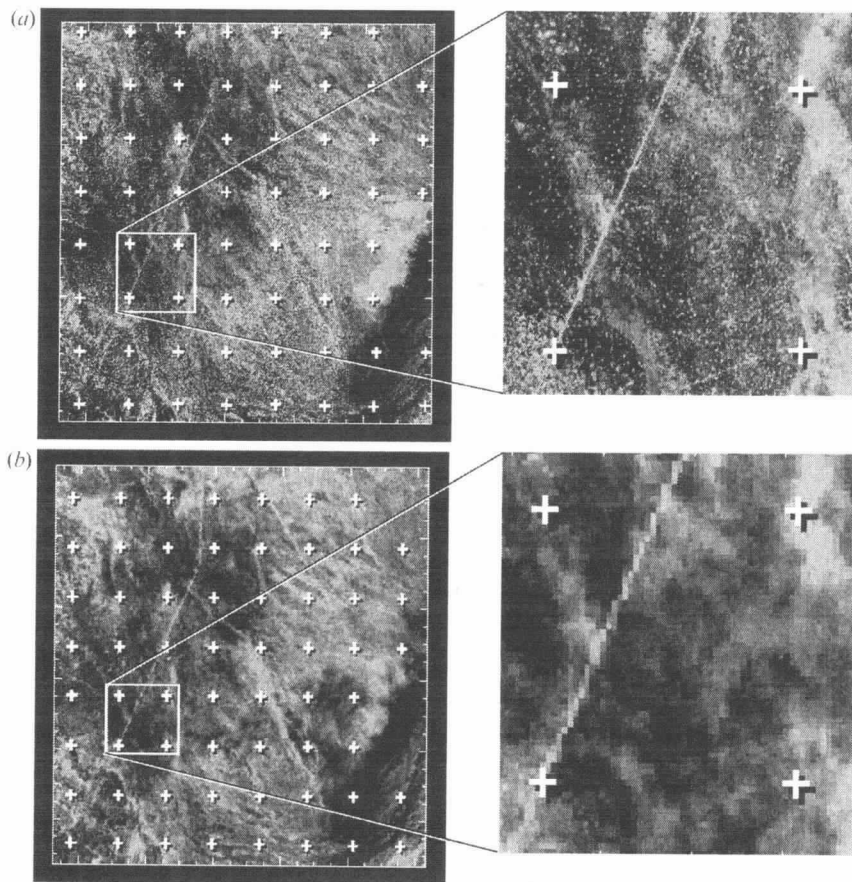


Figure 11. Example of tie-points created with the software described herein for images with unknown geometric relationships. (a) Near-infrared band from an IKONOS image (4 m nominal pixel size) at the Sevilleta Long Term Ecological Research Site, New Mexico, US, with enlarged area showing detail of four ITPs. (b) A corresponding subset of the near-infrared band of a Landsat ETM+ image (28.5 m nominal pixel size) for the same area at approximately the same time of year, with a similar enlarged area.

for image-by-image input altogether. The computational burden is relatively minor compared to many other image processing tasks common today, and is likely not to be an issue for many users.

5. Conclusions

We feel that the correlation-based matching approach has great potential for use in the general satellite remote sensing community. Because the grid of tie-point pairs produced by the software is regular, it is optimal for capturing the geometric relationships of images. The computational costs are higher than for other methods, but these costs are relatively small in today's terms. They are certainly lower than costs of manual interpretation. Moreover, the technique is repeatable, ensuring that

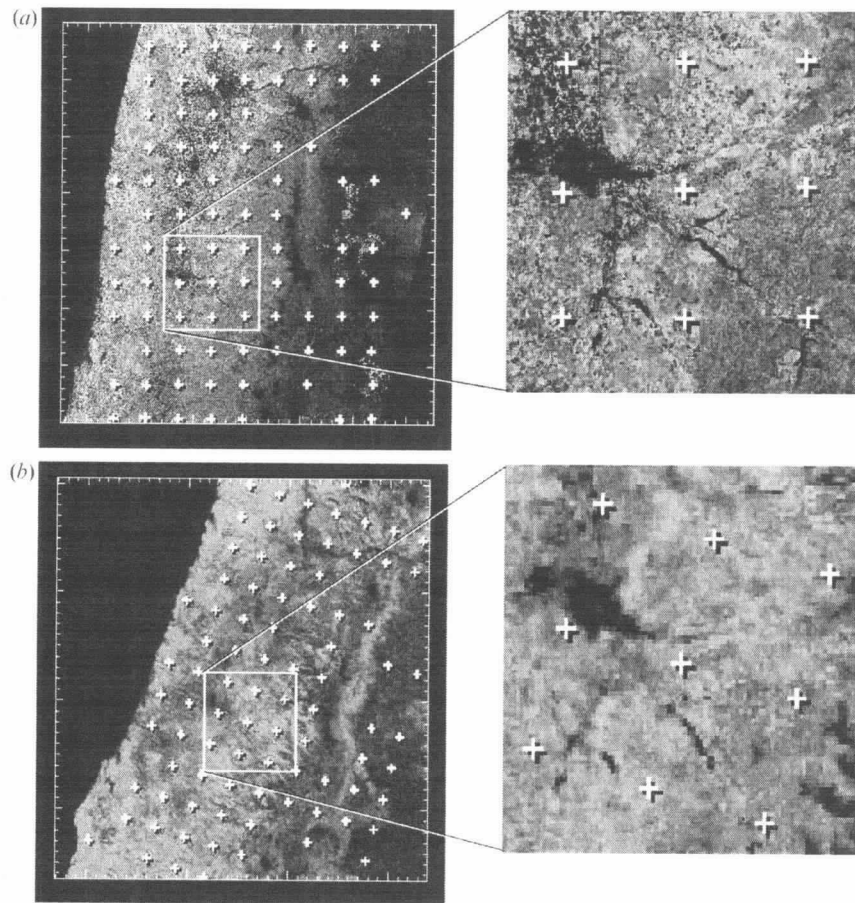


Figure 12. As figure 11, but for imagery acquired over western Oregon, US. (a) The greenness band from a tasseled-cap transformation of Landsat TM data (28.5 m nominal pixel size) mosaicked from five scenes, with an enlargement to show detail around nine ITPs. (b) An NDVI image composited from two weeks of AVHRR imagery (nominal pixel size 1 km; data courtesy Brad Reed, USGS EROS Data Center) around the same time as the TM scene, with an enlargement showing detail for an area of nine ITPs.

image libraries built up over time have consistent geometric properties. It is relatively robust to simple distortions and inaccuracies, although special attention must be paid to warping and to inaccuracies in pixel size estimation. Nevertheless, the correlation-based approach is flexible, and the software allows complete control over image-matching parameters. By focusing on robustness over absolute efficiency, and by relying on simple user input, we feel that correlation-based software can be useful to many users in the remote-sensing community.

Acknowledgments

Funding for this project was provided from NASA through the Terrestrial Ecology Program, the EPSCOR program, and the Land Cover Land Use Change

program. Additional support came from the NSF-funded LTER program at the H. J. Andrews Experimental Forest, and from the USDA Forest Service Pacific Northwest Station's Ecosystem Processes program. The authors would like to thank Karin Fassnacht and Andrew Hudak for helpful comments on the manuscript, Andy Hansen for collaboration on one of the funding projects. Greg Asner for images from Brazil, Brad Reed for AVHRR composite imagery, and Alisa Gallant, Douglas Oetter, Tom Maiersperger, Keith Olsen, Michael Lefsky, Mercedes Berterreche, and Andra Wright Parmenter for comments on the software package described herein.

References

- BARNEA, D. I., and SILVERMAN, H. F., 1972. A class of algorithms for fast digital image registration. *IEEE Transactions on Geoscience and Remote Sensing*, **21**, 179–186.
- BERNSTEIN, R., 1983. Image geometry and rectification. In *Manual of Remote Sensing*, edited by R. Colwell (Falls Church, VA: American Society of Photogrammetry), pp. 873–922.
- BROWN, L. G., 1992. A survey of image registration techniques. *ACM Computing Surveys*, **24**, 325–376.
- DAVIS, W. A., and KENUE, S. K., 1978. Automatic selection of control points for the registration of digital images. *Proceedings of the Fourth International Joint Conference on Pattern Recognition, Kyoto, Japan* (New York: IEEE), pp. 936–938.
- FONSECA, L. M. G., and MANJUNATH, B. S., 1996. Registration techniques for multisensor remotely sensed imagery. *Photogrammetric Engineering and Remote Sensing*, **62**, 1049–1056.
- GOSHIASBY, A., STOCKMAN, G. C., and PAGE, C. V., 1986. A region-based approach to digital image registration with subpixel accuracy. *IEEE Transactions on Geoscience and Remote Sensing*, **24**, 390–399.
- GUILD, L. S., 2000. Detection of deforestation and conversion and estimation of atmospheric emissions and elemental pool losses from biomass burning in Rondonia, Brazil. PhD thesis, Oregon State University. 120 pp.
- HUDAK, A. T., LEFSKY, M. A., COHEN, W. B., and BERTERRETICHE, M., 2002. Integration of lidar and Landsat ETM+ data for estimating and mapping forest canopy height. *Remote Sensing of Environment*, **82**, 397–416.
- LI, H., MANJUNATH, B. S., and MITRA, S. K., 1995. A contour-based approach to multisensor image registration. *IEEE Transactions on Geoscience and Remote Sensing*, **4**, 320–334.
- LI, M., 1991. Hierarchical multipoint matching. *Photogrammetric Engineering and Remote Sensing*, **57**, 1039–1047.
- MEYER, P., 1994. A parametric approach for the geocoding of airborne visible/infrared imaging spectrometer (AVIRIS) data in rugged terrain. *Remote Sensing of Environment*, **49**, 118–130.
- OETTER, D. R., COHEN, W. B., BERTERRETICHE, M., MAIERSPERGER, T. K., and KENNEDY, R. E., 2001. Land cover mapping in an agricultural setting using multi-seasonal Thematic Mapper data. *Remote Sensing of Environment*, **76**, 139–155.
- PARMENTER, A. W., HANSEN, A., KENNEDY, R., LANGNER, U., COHEN, W., LAWRENCE, R., MAXWELL, B., ASPINALL, R., and GALLANT, A., in press. Land use and land cover change in the Greater Yellowstone Ecosystem: 1975–1995. *Ecological Applications*.
- PRATT, W. K., 1974. Correlation techniques of image registration. *IEEE Transactions on Aerospace and Electronic Systems*, **10**, 353–358.
- PRATT, W. K., 1991. *Digital Image Processing* (New York: John Wiley and Sons, Inc.).
- RIGNOT, E. J., KOWK, R., CURLANDER, J. C., and PANG, S. S., 1991. Automated multisensor registration: requirements and techniques. *Photogrammetric Engineering and Remote Sensing*, **57**, 1029–1038.
- ROSENHOLM, D., 1987. Multi-point matching using the least-squares technique for evaluation of three-dimensional models. *Photogrammetric Engineering and Remote Sensing*, **53**, 621–626.
- SCAMBOS, T. A., DUTKIEWICZ, M. J., WILSON, J. C., and BINDSCHADLER, R. A., 1992. Application of image cross-correlation to the measurement of glacier velocity using satellite image data. *Remote Sensing of Environment*, **42**, 177–186.

- SCHOWENGERDT, R. A., 1997, *Remote Sensing: Models and methods for image processing* (San Diego: Academic Press).
- TSENG, Y.-H., TZAN, J.-J., TANG, K.-P., and LIN, S.-H., 1997, Image-to-image registration by matching area features using fourier descriptors and neural networks. *Photogrammetric Engineering and Remote Sensing*, **63**, 975–983.