Image Analysis Method for Overcoming Source Distortion Using Algebraic Invariant Methods*

Boris Kovalerchuk, Richard Chase
Department of Computer Science, Central Washington University,
Ellensburg, WA 98926-7520
borisk@cwu.edu

ABSTRACT

For multispectral sensory and geospatial data to be properly integrated they must be co-registered with known data which is a difficult and time consuming process. A persistent problem with new unregistered data is geometric image distortion. This paper deals with distortion due to disproportional transformation. Images can be disproportionally transformed because of a specific angle of data acquisition, sensor and lens distortions, atmospheric effects, and others factors. This research is focused on developing a method to overcome such distortion effects and to provide computational tools to automate a large portion of the process without relying on the sensor geometry and model that may not be known. Current methods of image analysis and feature recognition rely heavily on geometric shapes and/or the topological nature of data contained within the image. In addition to geometric shapes and topological data, features and images can also be compared algebraically. Algebraic structures have been defined with which comparisons can be made between geometric components such as relative angles, and lengths. Invariant point placement and feature comparison methods are developed here that can overcome the effect of distortion and disproportional scaling. Deriving a method that is invariant to disproportional scaling that is based on an algebraic invariant method is a new approach to solving this problem and represents a new mathematical language for the processing of image data.

INTRODUCTION

The problem of imagery integration includes: (1) Registration (geo-referencing) and co-registration of raster imagery, (2) Conflation of vector data (matching features, geo-referencing features and transfering feature attributes between vector datasets), and (3) Fusion of raster and vector data. Progress in image integration is critical for change detection, automatic target recognition and content based search of images. Images to be registered invariably contain different information. At the conclusion of every registration process, there are ambiguities and potential inconsistencies that must be estimated to judge process quality. Robust measures of these differences are necessary to understand the complex variability of image matching and to guide the process. A final step of having an expert review and modify registrations is often essential.

The possibility of using photogrammetric models to integrate imagery often is limited – such models can be unavailable or inaccurate. The traditional "model-free" integration method uses simple features – points that *have no internal structure*. This is a fundamental problem of this approach. In the traditional approach: (1) A user finds matching points called Control Points or Ground Control Points (GCPs) interactively, (2) Software computes coefficients of a mapping polynomial P, and (3) Image are transformed by P. This method is simple and intuitive for the user. This is the main advantage of this method and the reason that it is implemented in every commercial GIS and imagery analysis product. Its disadvantage is a direct result of its advantages -- matching beyond a few control points is questionable and needs human intervention to verify that the computed result is correct. With a growing stream of imagery such verification is becoming less and less realistic. The use of features that have internal structure is a promising approach to deal with this problem. Figure 1 illustrates some challenges of the control point approach. The upper part (a) shows the problem of matching area of interests (AOIs) when control points are located outside of AOIs and the lower part (b)

* This work has been supported by the grant from the National Geospatial-Intelligence Agency that is gratefully acknowledged.

shows the problem of matching AOIs when control points are located on AOIs. The matching point for point Q is not clear because it was not a control point. The match will be enforced by relatively arbitrarily selected polynomial P with only empirical evidence that it is in between two control points.

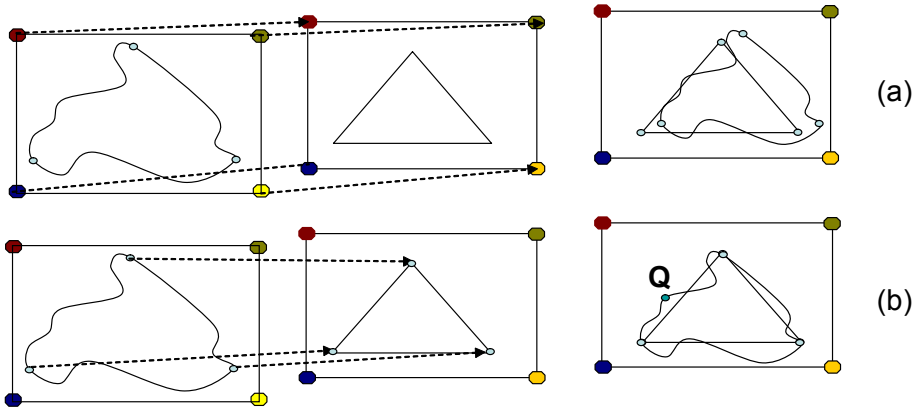


Figure1. Control point based image integration.

Previous work on registration has been reviewed recently by Zitová & Flusser[1]. They list hundreds of papers published in the last 10 years. The review for the previous 10 years[2] also listed hundreds of papers. However, the problem is still unsolved as recently indicated by broad agency announcements (BAA) from NGA, NSF, DARPA and Air Force.

What is the reason for that? At first registration is not a single problem, it is a collection of many problems with an umbrella term "registration". Adding more sensor modalities to the loop permanently expands that the domain. The second reason is that there is *no adequate mathematics* to develop a robust registration method for a variety of image registration situations[3,4]. Ignoring this math deficiency may lead to hundreds more specialized methods in next 10 years again and no final solution. We are advocating a more general mathematical framework/paradigm to get a smaller number of more universal methods. These new methods should be able to provide an equivalent accuracy now reached by specialized methods and cover more types of images. Traditional correlation methods work only for images that are shifted. While other methods work with rotated images, they do not work with scaled images and images that only partially overlap. New methods should work for these situations. There are three mathematical paradigms that provide foundations for image integration techniques. We loosely call them geometry, algebra, and topology. Figure 2 compares these mathematical paradigms for image integration problems.

Geometry operates with geometric entities: coordinates of points, angles, and areas, and quantities computed from them such as distances. Geometry is the base for integration techniques which use Euclidian and Hausdorff distances. The Euclidian distance is used for the Root Mean Square Error (RMSE) optimization criteria between matching points in algorithms that based on control points, correlation, and polynomial interpolation. The Hausdorff distance is often used to measure similarity in methods that are based on polylines instead of control points. The transition from visual Greek geometry to modern numerical geometry with Cartesian coordinates was a revolution in science, but also made it less intuitive and less based on human visual perception that is critical in image integration.

In this context, algebra focuses on relations between geometric entities or entities derived from them. In general, algebraic order relations ">" and "≥" are defined for any numeric and non-numeric entities including points, angles, areas, shapes, polylines, etc. Partial order relations can also be defined for many entities. For instance, intervals on a straight line are only partially ordered. Let [a1,a2] and [b1,b3] be two intervals where a2 ≥ a1 and b2 ≥ b1 then relation "≥" can be defined as follows, [a1,a2] ≥ [b1,b2] ⇔ a1≥ b2, that is interval [a1,a2] starts at the point a1 that is greater than the end point b2 of [b1,b2]. The linear algebra operates with relations between linear expressions of x, y, and z that can be coordinates of a point p=(x,y,z) such as an equation relation ax+by+cz=d or an inequality relation ax+by+cz ≥ d.

There are two quite different relations: (1) Relations between directly observable geometric entities, e.g., angles L1 and L2, L1>L2 and (2) Relations between mathematical expressions based on observable geometric entities, e.g., distances |between lines F1, F2 and F3, ||F1,F2|| < ||F1,F3|| or relations between linear functions ax+by+cz < sx+ty+qz. The algebraic relations (1) between observable geometric entities often have more intuitive, visual, and direct physical meaning than relations (2) between less empirical mathematical expressions. Relations (1) often represent empirical and

structural relations between geometric entities. Such relations can be very useful for building robust image integration methods as we show below.

Topology focuses on relations between geometric entities that do not change under "rubber sheeting" transformations. For instance, a T-intersection A formed by three lines is topologically equal to another T-intersection B with any other angles between the lines, A=B. However, T-intersection A is not topologically equal to a triangle C, A≠C, because there is no "rubber sheeting" transformation that can transform A to C.

Figure 2 presents an example of integrating two images where the middle line F1 is from image A and lines F2 and F3, respectively on the top and on the bottom are from another image B. Co-registration of these images needs to resolve line match ambiguity. Which line of two lines F2 or F3 should be matched with the middle line F1? The traditional approach would compute the distances between lines (e.g., Housdorff distance), ||F1,F2|| and ||F1,F3|| and match F1 with F2 if ||F1,F2|| < ||F1,F3||. This is shown in the left column of pictures in Figure 2. The second column shows the same match situation, but with slight differences in mutual location of lines. Different noises, camera geometries and other factors can cause such differences. None of these images may represent "ground truth" correctly. Now we can compute Hausdorff distances for the second case and may find that the distance relation has changed, to the opposite one, ||F1,F2|| > ||F1,F3||. Thus, the line match is changed to the opposite one, now we match F1 with F3. This means that a distance based geometric approach is too variant for resolving match ambiguity.

Pictures in columns 3 and 4 illustrate how structural algebraic relations can provide a robust feature match resolving match ambiguity. The lower pictures in the columns 3 and 4 show all three features presented by their generalized structure that an image analyst can observe. Structurally, the line in the middle is closer to the lower line in both cases in spite of some changes in distances. Pictures in the columns 5 and 6 show topological relations between the same three lines. All of them are topologically equivalent to straight lines and thus do not provide any information to resolve match ambiguity. Thus structural algebraic relations promise to be very useful tools for developing robust image integration algorithms.

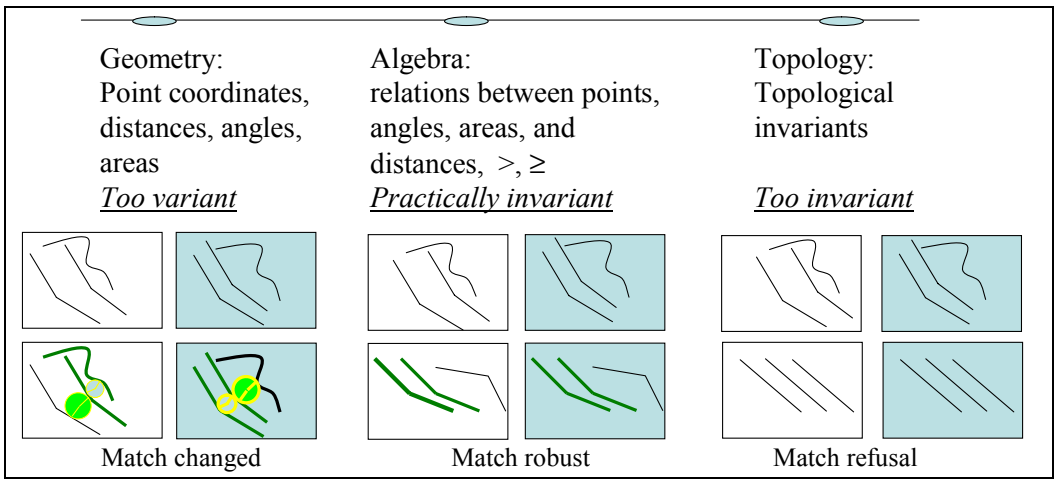


Figure 2. Comparison of mathematical paradigms

ALGEBRAIC STRUCTURAL ANALYSIS

Algebraic structural relational invariants[4] provide a basis for the registration/conflation of images (in raster or vector format) from many sources with various resolutions and reliabilities giving them common scales and coordinates. This is done by (1) Extracting curvilinear features using standard algorithms; (2) Building generalized features from extracted features that may contain discontinuities and combine several original extracted features; (3) Calculating algebraic structural characteristics of these features using techniques from abstract algebra; and (5) Registering images by matching features based on their intrinsic algebraic structure.

This method of *algebraic invariants*: (1) Does not rely on the identification of *control points*[1,2], (2) Does not require *common scales* and (3) Does not require that the *orientations* of individual images be known. This method differs from other commonly used[5,6,7,8,9,10,2,1]. The combination of this method with photogrammetric and other sensor information can be useful, especially if the images are taken by cameras with very different locations or resolutions.

The algebraic invariant approach assumes that each image has several well-defined "features". In this paper we explore features that can be represented as polylines (continuous chains of line segments). Such features are a wider concept than is commonly used in image sciences. *Anything with a reasonably well-defined shape will work as a feature.* A closed polygon is also a feature in this sense. The only requirement is that the feature can be fit with a polyline. It is not necessary to know what it is or if there are any correspondences with polylines in other images.

The algebraic technique we use[11] differs from traditional algebraic techniques such as linear algebra. The following definitions make this distinction clear. For more detail see Kovalerchuk & Schwing[4], chapter 19.

Definition. A pair $\mathbf{a} = \langle A, \Omega \rangle$ is called an ***algebraic system*** if A is a set of elements, Ω_a is a set of predicates {P} and operators {F} on A and on its Cartesian products, where

$$P: A\times A\times ...\times A \rightarrow [0,1] \text{ and } F: A\times A\times ...\times A \rightarrow A.$$

A set of elements can be more complex and contain several sets (e.g., A and R). In this case, the system is called *multisort algebraic system.*

A multisort algebraic system $\boldsymbol{a} = \langle A,R,\Omega_a \rangle$ is called a ***linear feature*** if R is a set of real numbers, Ω_a consists of two operators (functions) $D(a_i)$ and $L(a_i , a_j)$ and three predicates (linear order relations) $>_a , \geq_D , \geq_L$. Operators $D(a_i)$ and $L(a_i , a_j)$ satisfy some additional axiomatic properties [Kovalerchuk, Schwing, 2001, 2004] that allow them to represent such polyline characteristics as lengths of intervals and angles.

Definition. An algebraic system $\tilde{\boldsymbol{a}}$ is called an ***abstracted linear feature*** of feature $\boldsymbol{a}$ if $\Omega_{\tilde{a}}$ consists of three predicates (linear order relations) $>_a , \geq_D , \geq_L$, with $\Omega_a = \langle >_a , \geq_D , \geq_L \rangle$ from the linear feature ***a.***

Definition. Linear features $\boldsymbol{a} = \langle A, R, \Omega_a \rangle$ and $\boldsymbol{b} = \langle B, R, \Omega_b \rangle$ are *co-reference candidates* if they are homeomorphic and have isomorphic linear subfeatures

$$\boldsymbol{e_a} = \langle E_a, R, \Omega_e \rangle \text{ and } \boldsymbol{e_b} = \langle E_b, R, \Omega_e \rangle,$$

where $\boldsymbol{e_a} \subseteq \boldsymbol{a}$ and $\boldsymbol{e_b} \subseteq \boldsymbol{b}$.

With these, theorems such as the high speed of the registration process can be proven[4].

Theorem: If the number of elements in linear features $\boldsymbol{a}$ and $\boldsymbol{b}$ equals n, then their maximum co-reference subsystem $\boldsymbol{e}$ can be found in $O(n^3)$ matrix comparisons and $O(n^5)$ binary number comparisons for the worst-case scenario.

The number of points n needed to identify the feature is much smaller than the number of pixels in the whole image that autocorrelation methods use. The registration and conflation method based on algebraic invariants using polylines may be done in several ways. Consider the two satellite images of the Kyrgyz lake Sonkyl in Figure 3.

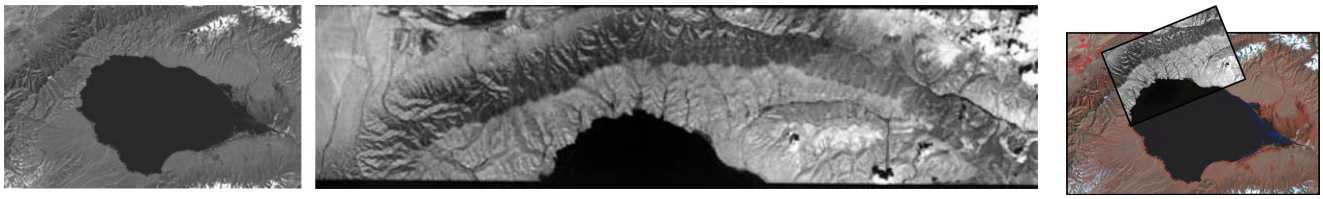
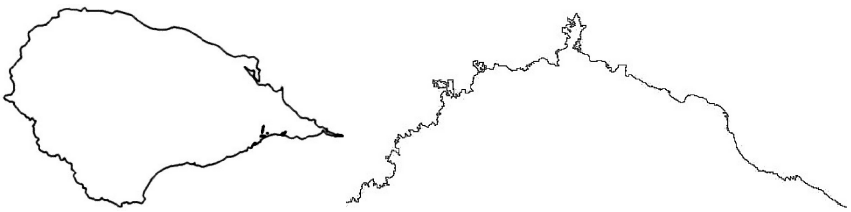


Figure 3. Two images of the lake Sonkyl in Kyrgyzstan and their match.

Feature extraction programs can be used to construct numerous polylines, with the most obvious one being the shoreline of the lake. The results are shown in Figure 4. While the overall structure of the extracted shorelines is apparent, the polylines differ in detail for a variety of reasons. Robust ways of comparing these polylines are necessary to determine image transformations and to assess the quality of the result.



(a) (b)

Figure 4. Lake shores extracted from the satellite images of Sonkyl.

BINARY STRUCTURAL DIVISION (BSD)

The angles between segments and individual segment lengths are two algebraic characteristics of polylines that we use. For smooth features extracted from images with comparable scales and resolutions, either comparison works well. When there are marked differences in image scale and resolution, the choice of angles or lengths becomes more important. We examine characteristics of extracted polylines and how they may be interpolated and compared. The issue of variability of points that form a polyline also needs to be addressed. Different feature extraction algorithms and imagery analysts can assign points differently on the same physical feature. This can affect finding co-reference candidate features. The algebraic technique called Binary Sequential Division (BSD) method[12,4] addresses this problem in a computationally efficient way.

Consider the polyline in Figure 5 and the successive approximations defined by taking points defined by the mid point of lengths along the polyline as indicated. The BSD method computed by finding a curve **middle point** along the curve, then repeated for each half, halves of halves and so on. In this process a recursive function *G(n)* is used to denote the *n*-th interpolation of a polyline. Note that above $n=2^k$, and *k* is called the **BSD level.** Now we can use such interpolation of each feature with uniformly allocated points to build an algebraic representation of features.

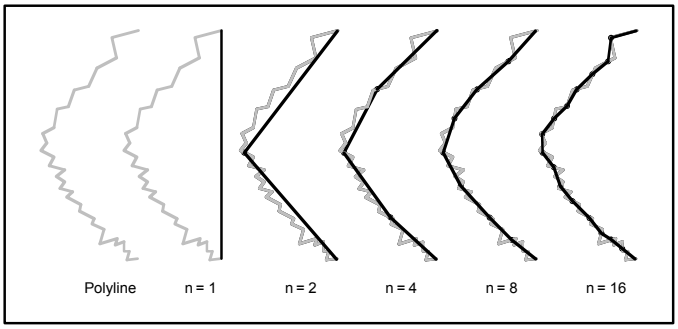


Figure 5. Structural interpolations of a polyline**.** Our experiments show that 8 binary sequential divisions with $256=2^8$ linear segments is typically sufficient for interpolation.

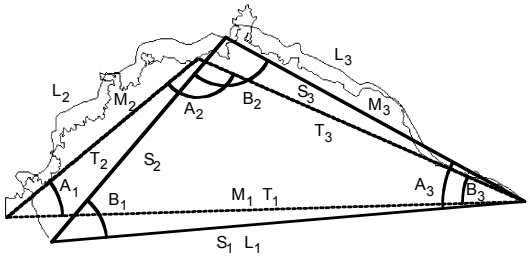


Figure 6. Sections of the extracted shorelines with the first level BSD interpolation with k=1 and n=2.

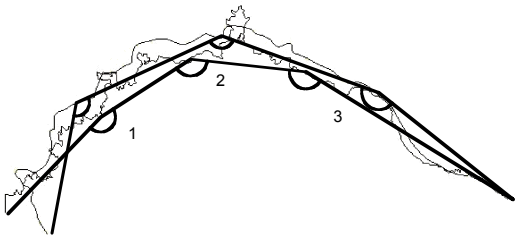


Figure 7. Fragment of BSD level 2 for the two polylines.

Figures 6 and 7 depict BSD levels 1 and 2 for the vectorized features shown in Figure 4. The middle points of each feature are as shown, computed along each line. Significant fluctuations have been lost in the lower resolution image. Feature M as interpolated has angles A_1, A_2, and A_3. Feature L as interpolated has angles B_1, B_2, and. B_3. Next, each BSD level 1 in Figure 6 is converted to abstract algebraic systems $\boldsymbol{a} = \lesssim A, R, \Omega_a >$ and $\boldsymbol{b} = \lesssim B, R, \Omega_b >$, respectively for

features L and M shown in Figure 6. These systems contain order relations for lengths, $\geq_D$,and angles, $\geq_L$. For instance, we can notice that length M_2 is greater than M_3, $M_2 \geq M_3$ and angle A_1 is smaller than A_2. Next the similarity of systems ***a*** and ***b*** is explored and if they are similar (homeomorphic) then we conclude that features L and M are structurally similar. After that the similarity of BSD level 2 for the same features shown in Figure 7 is explored. The process continues until the highest BSD level is reached where the similarity of feature structure is confirmed. In this way the match of images shown in Figure 3 was made.

GENERALIZATION OF THE APPROACH: EXPANDING ABSTRACT FEATURE CONCEPT

The concept of an abstract feature that represents feature structure can be expanded beyond matrixes of angular relations and edge lengths relations. Matrixes of relations between edge intensity/color relations, edge texture relations, and spectral relations are natural candidates for such expansion. Figure 8 illustrates this idea showing a variety of order relations. Each of them can be viewed as a simple algebraic system with a single predicate, $<A_i,P_i>$ as shown in Figure 8. These matrixes represent relations between components of features. For instance, elements in the second table show that edge a1 is greater than edge a3. Similarly, in Table 3 the intensity of edge a1 is greater than intensity of edge a3 and in Table 4 a selected spectral characteristic of edge a1 is greater than the same characteristic for edge a4. All these systems can be combined together and from a system that contains all of them, $<A_1,A_2,A_3,A_4, P_1,P_2,P_3,P_4>$. In Figure 8 the last three sets of elements A_2, A_3 and A_4 are the same (edges), but predicated are different, thus this system can be written as $\boldsymbol{a}=<A_1,A_2,P_1,P_2,P_3,P_4>$. We may have another feature and can build an abstract algebraic system, $\boldsymbol{b}=<B_1,B_2,R_1,R_2,R_3,R_4>$ for this feature. Now we can explore their relations as a structural match between features in these abstract terms, using concepts of isomorphism, homomorphism and homeomorphism.

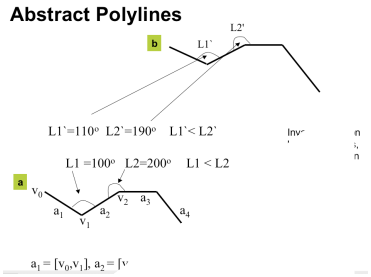
Another way to define a structure of the feature would be a set of matrixes of ratios. It can be ratios for angle values, edge lengths, edge intensities/colors, edge textures, and edge spectral data. Such generalization needs another expansion of the concept of abstract feature. It can be a multisort system with not only sets A1,A2,A3,A4 and relations on them, but also a domain R of possible ratio values, where $F: A \times A \rightarrow R$.

Angle	L1	L2	L3	L4	L5	L
L1	1	0	0	1	0	1
L2		1	0	1	0	0
L3			1	0	1	0
L4				1	1	0
L5					1	1
L6						1

$<A_1, P_1(L_i,L_j)>$ abstract feature: angle

Edge	a1	a2	a3	a4	a5	a6
a1	1	0	1	0	0	1
a2		1	1	0	0	0
a3			1	0	1	0
a4				1	1	0
a5					1	1
a6						1

$<A_2, P_2(a_i,a_j)>$ abstract feature: edge length



Edge	a1	a2	a3	a4	a5	a6
a1	1	1	0	1	0	1
a2		1	1	0	0	0
a3			1	0	1	0
a4				1	1	0
a5					1	1
E6						1

$<A_2, P_3(a_i,a_j)>$ abstract feature: edge intensity

Edge	a1	a2	a3	a4	a5	a6
a1	1	1	0	1	0	1
a2		1	1	0	0	0
a3			1	0	1	0
a4				1	1	0
a5					1	1
E6						1

$<A_2, P_4(a_i,a_j)>$ abstract feature: edge spectral

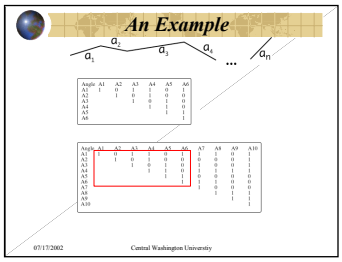
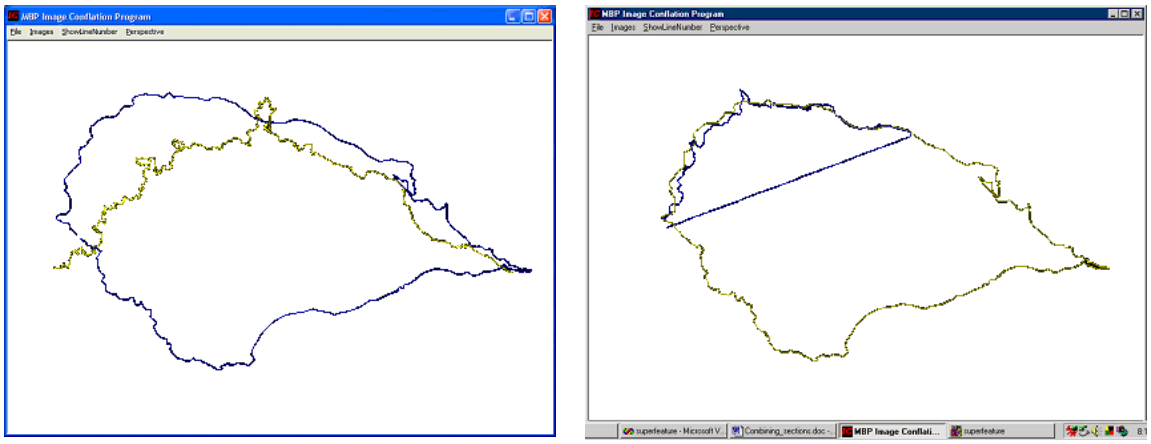


Figure 8. Expanding abstract feature concept.

UNKNOWN PROPORTIONAL SCALES

Figures 9 and 10 show different possible matches when scales of two images are unknown. These matches have been found by using the BSD method. The image in Figure 9(a) shows feature proportions as they were extracted and vectorized from the two original images. Our experiments had shown that the BSD method was able to capture the right

scale, rotation, and shift. The second image in Figure 9 shows the correct match. Images in Figure 10 illustrate the variability of possible matches when the scale ratio of two images varies between 1:1 and 1:2.



(a) Original feature proportions. (b) Correct match found by BSD method

Figure 9. Original feature proportions and correct match.

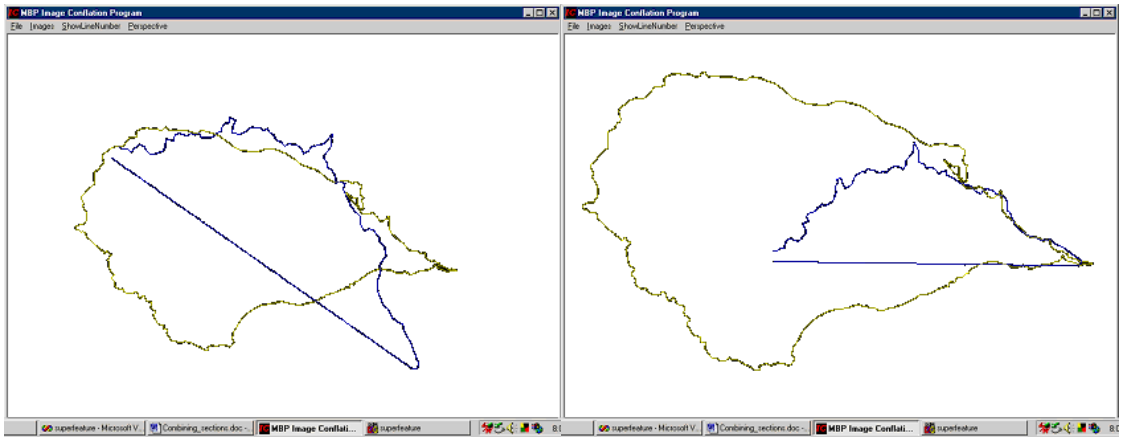


Figure 10. Alternative matches found by using BSD method.

UNKNOWN DISPROPORTIONAL SCALES

Images that are disproportionally scaled present more challenges than those that are proportionally scaled since BSD middle points are not invariant. This makes BSD matching problematical.

We have developed an **algorithm to detect the rotation and disproportional scale factors** for a given set of features or subsections of features. After transforming, we can place points in a consistent manner such that we can determine if the candidate sections are indeed matching or not. This removes a difficulty that occurs in general point placement algorithms that produce points that are not invariant to the combination of rotation and disproportional scaling. Even using a Ramer algorithm that interpolates polylines using the furthest points for each segment does not place points in the same places because of the nature of real data. Ramer algorithms are invariant only for ideal data without noise and require different threshold values for different parts of rotated and disproportionally scaled features. It is impossible to calculate these thresholds without knowing the transformation metrics in advance.

A half circle is used to demonstrate disproportional scaling mathematically and visually. We find that when disproportional scaling is applied, the relationships of the initial x coordinates to each other remain the same, and the relationships of the initial y coordinates to each other remain the same. However, the relationships between initial x and y coordinates do not remain the same. This has interesting implications for comparing polylines features that have different orientations. Consider the deformation of 200% scaling in x (horizontal) direction only (see Figure 11):

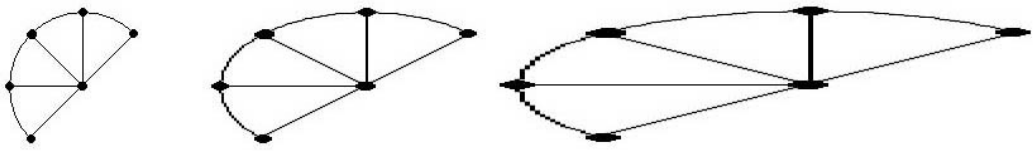


Figure 11. Disproportional scaling

It is both clear from observation (see Figure 11) and mathematically provable that the relationships of x coordinates of the points remain the same in all three shapes. The same is true for y coordinate values taken separately from x coordinates. If the center of the shape is considered as the origin, the x coordinates have simply been scaled by a factor of 2 (for the second shape) and by a factor of 4 (for the third shape) with the y coordinates staying the same (scaled by a factor of 1). Let us parameterize the location of each point in the range of the values in the x and y independently. Assume that the lowest x value for the five points on the curve has a parameter t=0.0% and the highest value has t=100.0%. Similarly the parameter u is set up for the y coordinate, with u=0.0% and u=100.00% for the two extremes respectively. In this notation the lowermost point is (t,u)=(16.0%,0.0%) and the rightmost point is (t,u)=(100.0%,75%). Thus, if the x and y coordinates are parameterized to the range of the values in the x and y independently, it is found that starting from the lower left point:

X coordinates: {16.0%, 0.0%, 16.0%, 58.0%, 100.0%}
Y coordinates: { 0.0%, 37.5%, 75.0%, 100.0%, 75.0%}

These values are the same for all cases (a), (b), and (c) in Figure 11. Now consider the same feature with a different orientation as shown in Figure 12.

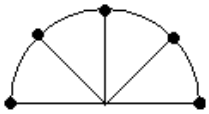


Figure 12. The feature with a different orientation

The relationships of these points parameterized using the same method are very different:

X coordinates: {0.0%, 12.5%, 50.0%, 87.5%, 100.0%}
Y coordinates: {0.0%, 70.0%, 100.0%, 70.0%, 0.0%}

To make these representations, the same the feature must be rotated until the same relationships are found. This can be done by sequential iteration giving the rotational metric and solving for rotational registration.

With the rotational orientation known, simple geometric properties can be used to find the change in the x and y directions. The coordinate values can be used to derive the solution to reverse the effect of the disproportional scaling. To use our previous example of the slanted half circle and the second shape (first mutation), we find the transformation in the x by dividing the range of the x in the second shape by the range of the y in the first shape to find the scaling coefficient is 2.0 since the range of the x values in the second shape is twice the range of the first shape. Similarly in Figure 11 we could find the scale factor for the case (c) is 4.0 to the shape (a) and 2.0 to the shape (b). The only factor that we had to guess was the rotation that occurred prior to the disproportional scaling.

We also have the challenge of correctly positioning the points in the features such that we have a one-to-one correspondence in the different shape representations. This can be done using a BSD method following a first level Ramer algorithm implementation. In this method we do not use a threshold epsilon value which is a part of the Ramer algorithm. We simply divide the features into a factor of 2 sections using the extreme point as the first division (assuming there is no ambiguous situation of where to place the point). This representation by itself is in fact enough to establish the rotational metric by which the disproportional scaling can be derived. Using this information, we can disproportionally scale the original data and then place points in the appropriate positions using a point placement technique such as pure BSD. Now we can describe the Reverse Disproportional Scaling (RDS) algorithm more specifically.

- Pick a pair of polyline subsections for comparison. Only compare subsections that are in the scale differential range chosen, for example, 1:2 to 2:1. This significantly reduces the number of subsections that must be tested.
- Rotate both features to be horizontal (if necessary).
- Find the maximum deviation point in the same manner that the Ramer algorithm does, but without using epsilon and only iterating the process once.
- Parameterize the point set for x and y values as a percentage of each of the ranges independently.
- Use the parameterized values of the midpoint placements to identify the rotational difference between the features when the disproportional scaling was applied. This can be done by iteration, binary search method, or by direct calculation. Then rotate both features to this orientation.
- In practice we use a double loop to find the rotational combination that produces the best match of the parameterized values. Feature A is rotated from 0 to 90 degrees in half degree increments. For each of these cases feature B is rotated from 0 to 90 degrees in half degree increments and the parameterized values of the rotated versions are compared. The absolute value of the difference in the x parameters is added to the absolute values of the difference in the y parameters. This combined value gives the degree of parameter matching. The lowest value is used to identify the best rotational combination. A comparison value less than 1% means that neither the x parameter nor the y parameter values are more than 1% different and the combined sum of them is less than 1% as well.
- Rotate the features by the best-fit parametric values.
- Determine the ranges of the x and y values for both. Then use the ratio of the x ranges to find the disproportional scaling factor for the x, and the ratio of the y ranges to find the disproportional scaling factor for the y. Disproportionality scale factors are thus computed.
- The disproportional scale factors are applied to the original data to make two new proportionally scaled images.

Now the BSD method than needs proportional scales can be applied with confidence to confirm or negate the possibility that these are indeed *matching sections*.

The C++ RDS algorithm has been found to successfully identify the spatial relationship at which the disproportionality scaling took place, and subsequent scale factors. Thus, we can transform the distorted version back into the original without having known the relationship initially. For the shapes used above in Figure 11 we find a best rotation for feature A to be 45 degrees and the best rotation for feature B is 26.5 degrees, and at this combination the computed scale factor is 2.0 in the horizontal and 1.0 in the vertical which has been confirmed to be correct within the error range of the algorithm. The actual rotation value for the second feature is 26.51 degrees, thus an error of 0.01 degrees, which is well within the half degree resolution being used (+/- 0.25 degrees).

COMPARISON WITH OTHER APPROACHES

Probabilistic approach. The Bayesian probabilistic approach has been developed for pattern recognition problems. The main idea of this approach is assigning matching labels to features from two images[13]. Algorithms that provide matching labels combine: (1) A set of prior probabilities and (2) A set of measurements to create a further set of probabilities (the posterior probabilities) of feature match. This sequential process of adding more measurements (taken from images) ends up in the selecting labeling that provides the maximum of the combined probability of labeling.

The bottleneck of this approach is in obtaining probabilities for feeding this process. It is acknowledged[14,15] that if the probabilistic approach is to be practical, several assumptions need to be made. These assumptions are heuristics in essence. At first the prior probabilities are assigned to be equal using the principle of insufficient reason. While it is practical but it does not speed up a matching process and does not reflects any specifics of particular images.

The probabilistic approach assumes that measurements are instances of *random variables* with some *distributions*. In fact, there are two assumptions here: (a) measurements are random variables, and (2) distributions of random variables can be identified accurately enough. These assumptions need to be confirmed and an actual process of assigning relative values to all possible values of the corresponding random variables should be designed. To be practical it was assumed by Christmas[14] that measurements follow the Gaussian normal distribution with only justification for this assumption as "…this was *felt somehow to fit* with what was observed from a subjective examination of typical images." Even if one would agree with such heuristic, parameters of Gaussian distributions (covariance matrixes) should be found. Next to make the use of Bayesian probabilities practical some other heuristic assumption are needed such as conditional independence, thresholds, and robustness of measurements to noise.

Affine invariants. Affine invariants attracted attention in image matching for a long time[16,17,18,19]. One of computationally intensive ideas was producing various geometric transformations of a set of points to find possible affine invariants and distribution of non-invariant transformations. Less computationally intensive approaches are based on

using affine invariants associated with specific objects (sets of points) in the images, such as area, lines connecting centers of gravity of areas, and parallel lines. These methods assume that sets of objects can be extracted from images and noise does not change these sets dramatically. Such invariants include distance ratio[20], area ratio[21], and affine moment invariant (AMI)[16]. Distance ratio and area ratio are defined for pairs of objects, and AMI are defined for individual objects. The first two may have an ordering problem as noticed by Ahmadyfard & Kittler[17], but these invariants have the important advantage of intuitive meaning for analysts. Relative size of objects (area ratio) and mutual location of two objects (distance ratio) are easily captured by humans who match images.

AMI can potentially provide computation gain in matching of shapes because they are defined for each object (area) individually, but AMI can provide many false matches because of this in images with similar shapes. Next AMI are also much less intuitive than area ratios and distance ratios. If the goal is a fully automated registration, with minimal human quality control AMI may not be the best choice. It is possible that AMI will produce a result that human will reject. Similarly pixel based approaches may fail human quality control because an analyst can use concepts (features) of higher level than pixels to judge the match.

A classical pattern (target) recognition task has been studied with an affine invariant technique[17,18]. This task differs from the image integration/registration task in several important aspects. The authors search for an object in the *scene* assuming that the object constitutes the whole another image and this object can appear very differently in the scene because of possible differences in lighting, occlusion, as well as object and camera positions. Thus, the first image is the scene image and the second one is the object image. Typically in the pattern recognition problem the object is known in advance (e.g., tank), and its image has a good quality in comparison with the scene. It can be an image with good lighting, and without any occlusion of important parts of the object.

The situation in the image integration/registration/conflation tasks can be quite different. Both images can be of low quality, taken by different cameras and with different light source positions. Objects can be completely unknown, both images may contain many objects, and only parts of some objects may belong to both images. Associated object occlusions also may not be known and quite different. Even if the second image contains a single object the task is still very different from the pattern recognition task. The object may still be unknown, that is, we have no prior information about this object that can be used for guiding the image matching process.

Summarizing this discussion we will assume that the integration task should be solved under the following conditions: (1) Both images are scene images with many objects on them, and (2) Geographic areas covered by scene images overlap. There are also important modifications of the image integration task for images of adjacent scenes or for one image as a small fraction of the scene presented in another image. In spite of differences there is a fundamental similarity in recognition and registration tasks associated with the search for match in features.

SUMMARY

The BSD method has been shown to work well in cases where there is no disproportional scaling problem. By guessing that the initial subsections were in fact correctly a matching pair, we can apply the Reverse Disproportional Scaling (RSD) algorithm and compute the scale factors required to compensate for any possible disproportional scaling. From here we can use standard methods or our own BSD method for confirming or negating the correspondence of the two datasets.

The Reverse Disproportional Scaling (RDS) method is rather simple. It does not require an extensive amount of computation and so can be implemented with a minimum of code. We must be careful however that we do not identify a matching section that has a disproportional scale factor (the ratio between the scaling in the x and the scaling in the y) that is different than that found for the majority of matching sections found. It is thus very important to screen the transformation data to identify the dominant cluster. The transformation should be applied to the entire image and a statistical evaluation performed using proximity measures such as were used during the grid evaluation fine-tuning method.

REFERENCES

1. Zitová T., Flusser J., Image registration methods: a survey, Image and Vision Computing. 21 (11), 2003, pp. 977-1000.
2. Brown, L. A Survey of Image Registration Techniques, ACM Computing Surveys,vol.24 (4), pp. 325--376, 1992.
3. Bern, M., Eppstein, D., Agarwal, P., Amenta, N, Chew, P. at al. Emerging challenges in computational topology, 1999, citeseer.nj.nec.com/bern99emerging.html
4. Kovalerchuk B., Schwing J. (eds.), Visual and Spatial Analysis: Advances in Data Mining, Reasoning and Problem Solving (Eds.), Springer, 2005.
5. Bartl, R., and W. Schneider. Satellite image registration based on the geometrical arrangement of objects. *Proc. SPIE, v. 2579*, p. 32-40, 11/1995.
6. Cohen, S. and L. Guibas, Partial Matching of Planar Polylines under Similarity Transformations. *Proceedings of the Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*, 777-786, January 1997.
7. Dey, T., Edelsbrunner, H., Guha, S., Computational Topology. In *Advances in Discrete and Computational Geometry (Contemporary mathematics 223)*, ed. B. Chazelle, J. E. Goodman, and R. Pollack, American Mathematical Society, 109—143, 1999. http://citeseer.nj.nec.com/dey99computational.html
8. Eppstein D., Geometry in action: Cartography and Geographic Information Systems, 1999 http://www.ics.uci.edu/~eppstein/gina/carto.html
9. Hirose, M., Furuhashi, H., Kitamura, N., and Araki, K. Method for automatic registration using real-time multiview range images. *Proc. SPIE Vol. 4572*, p. 174-182, 10/2001.
10. Pinz, A., Prantl, M., and Ganster, H. A robust affine matching algorithm using an exponentially decreasing distance function. J.UCS - Journal of Universal Computer Science, Springer, 1(8), 1995.
11. Mal'cev A.I. Algebraic Systems, Springer-Verlag, New York, 1973.
12. Kovalerchuk, B., Sumner W., Curtiss, M., Kovalerchuk, M., and Chase, R., Matching Image Feature Structures Using Shoulder Analysis Method, In: Algorithms and technologies for multispectral, hyperspectral and ultraspectral imagery IX. Vol. 5425, International SPIE military and aerospace symposium, AEROSENSE, Orlando, FL, April 12-15, 2004.
13. Rosenfeld,A., Hummel, R. Zucker, S., Scene labeling by relaxation operations, IEEE Trans. on System and Cybernetics, pp. 420-433, 1976.
14. Christmas, W., Structural matching in computer vision using probabilistic reasoning, Ph. D. Dissertation, University of Surrey, UK, 1995, 93 p.
15. Christmas, W., Kittler, J., Petrou, M., Structural matching in computer vision using probabilistic relaxation, IEEE Trans. on pattern analysis and machine intelligence, pp. 749-764, 1995.
16. Flusser, J., Suk, T., A moment-based approach to registration of images with affine geometric distortion", IEEE Transactions on Geoscience and Remote Sensing, vol. 32, no. 2, March, 1994.
17. Ahmadyfard, A., Kittler, J. A Comparative Study of Two Object Recognition Methods, Electronic Proceedings of The 13th British Machine Vision Conference, University of Cardiff, 2-5 September 2002, www.bmva.ac.uk/bmvc/2002/papers/115/full_115.pdf
18. Ahmadyfard, A., Object recognition by region matching using relaxation with relational constraints, Ph.D Dissertation, Univ. of Surrey, UK, 2003. ftp://ftp.ee.surrey.ac.uk/pub/vision/papers/Ahmadyfard_thesis.pdf
19. Matsakis, P. Keller, J., Sjahputera, O., Marjamaa, J., The Use of Force Histograms for Affine-Invariant Relative Position Description, IEEE Transactions on Pattern Analysis And Machine Intelligence, Vol. 26, No. 1, January 2004.
20. Huttenlocher, D. P. Fast affine point matching: An output-sensitive method, CVPR91, pp. 263-268.
21. Chan, H.B., Hung, Y. S. Matching patterns of line segments by eigenvector decomposition, SSIAI'02, 2002, pp. 286-289.