# Image Alignment by Line Triples

*Scott O. Mason** and *Kam W. Wong*
University of Illinois at Urbana-Champaign, 2213 Newmark Civil Engineering Laboratory, 205 N. Mathews Ave., Urbana, IL 61801

ABSTRACT: An algorithm has been developed to automatically align the two images of a stereo pair by using line features called line triples. Each triple consists of three line segments. The center segment can be either a straight line or a circular arc, while the two end segments must be straight. A line triple is characterized by the type of center segment, the length of the individual segments, and the magnitude of the two deflection angles. The alignment process yields the conformal transformation parameters (two translations, one rotation, and one scale factor) between the two images. The algorithm has been successfully tested using several stereo pairs, two of which are reported in this paper.

## INTRODUCTION

DIGITAL PHOTOGRAMMETRY plays a key role in the automation of many industrial processes. Through the use of solid-state video cameras, image processing techniques, and methods of analytical photogrammetry, systems can be developed to provide real-time geometric measurements for the control of manufacturing processes. The potential for fully automated operation, free of the need for human involvement, will eventually open many new areas of application for photogrammetry.

The process of extracting three-dimensional scene information from stereo images can consist of four basic operations, including (1) preliminary alignment of the two images in the stereo pair, (2) relative orientation, (3) image correlation to generate three-dimensional (3D) model coordinates, and (4) absolute orientation. In topographic mapping, aerial photography of the terrain must be flown with specified forward and side overlaps. The differences in tilt, kappa rotation, and scale between two successive photos are usually small. Preliminary alignment of the two photos in a stereo pair is thus provided by the inherent overlapping geometry of the photo pair. Hannah (1988, 1989) developed an algorithm to automatically identify the necessary corresponding image points needed to determine the relative orientation between two photos. Hannah's algorithm identifies a small window (called an interest point) in one image where there is sufficient grey intensity texture, and then searches for a matching window in the other image within a small neighborhood defined by the overlapping geometry of the stereo pair. The algorithm requires some *a priori* knowledge on the overlapping geometry, and that there be no major differences in scale and kappa rotations between the two images. Greenfeld (1990) used edge vertices as matching features to locate corresponding image points for automatic relative orientation. However, this method also depends on some *a priori* knowledge of the overlapping geometry to restrict the search area.

In industrial and other close-range applications, the locations and orientations of the cameras are usually dictated by many constraints that cannot be easily altered. Under such circumstances, differences in the tilts and kappa rotations of the two images usually exceed the 3 or 4 degrees that are the acceptable limits in aerial photographs. Differences in scales exceeding 10 percent are also not uncommon. Preliminary alignment of the two images is essential before algorithms such as those of Hannah and Greenfeld can be used for automatic determination of relative orientation.

This paper presents an algorithm, together with some experimental results, for the automatic alignment of two images by matching line features called line triples. The purpose of the algorithm is to determine the approximate alignment of the two images in a stereo pair, as represented by four conformal transformation parameters: two translations, one rotation, and one scale factor. The algorithm is based on the fundamental assumption that there are similar features in the two images, and that these features can be extracted from the two images in the form of unique sequences of line segments.

## THE RATIONALE FOR LINE TRIPLES

The most important properties of features to be used in image matching are (1) distinctness from their neighbors, (2) invariance with respect to expected geometric and radiometric distortions, (3) stability, and (4) seldomness (Förstner, 1986). The features should have unique geometric characteristics, and should seldom occur more than once, in identical form, within an image. The features should also be stable in the sense that they retain the same general geometric characteristics in multiple views of the same scene.

Points, straight lines, and arcs are three basic types of geometric features that are usually present in an image. Point features, however, such as Hannah's interest points and Greenfeld's edge vertices are inherently lacking in seldomness to be useful for image matching without *a priori* knowledge to narrow the search space. Long straight lines and long arcs are more suitable for matching. Such features are represented by edges in the image, and commonly correspond to structural details in the scene. For this reason, line and arc features are generally stable insofar as they appear about the same in multiple views of the same scene, and are relatively invariant to geometric and radiometric distortions. However, single lines and arcs also lack seldomness, because such features usually appear in large numbers in an image. Thus, a more logical choice of image features for matching purposes are extended geometric features consisting of lines and arcs, as evidenced by works in the field of object recognition.

Many algorithms have been developed for the automatic recognition of specific objects in images by matching polygons derived from high-contrast intensity edges, typically object silhouettes (e.g., McKee and Aggarwal, 1977; Price, 1984; Ayache and Faugeras, 1986). The shapes and sizes of the polygons are characterized by the lengths of their segments, the angles included between two successive line segments, and the orders of the segments within the polygons. Thus, knowing the polygon representation of an object, its presence or absence in an image can be verified by identifying all polygons in the image and matching them with the known representation. Clark *et al.* (1980) used line triples, consisting of three consecutive, straight-line segments, as the basic geometric unit to match image features in low-tilt aerial photos of natural terrain. The image fea-

*Presently at the Institut für Geodäsie und Photogrammetrie, ETH-Zürich, Switzerland.

PHOTOGRAMMETRIC ENGINEERING & REMOTE SENSING,
Vol. 58, No. 9, September 1992, pp. 1329–1334.

0099-1112/92/5809–1329$03.00/0

tures were represented by polygons, from which the line triples could be directly derived.

For the purpose of automatic alignment of two images, it is not necessary to use complex geometric features. It was decided, therefore, to use the line triples introduced by Clark *et al.* (1980) as the basic geometric feature for matching. The matching attributes of the line triples were selected to meet the specific needs of the image alignment problem and, therefore, were different from those used by Clark *et al.* In order to enhance the seldomness characteristic of the line triples, the middle segment was allowed to be either a circular arc or a straight line. The angular relationship between two adjacent line segments in a triple was represented by the change in direction between the segments, which is equivalent to the deflection angles commonly used in engineering surveying. By using the change in direction instead of the direction of each segment as the attribute for matching, the algorithm becomes independent of the actual kappa rotation of the two images. Thus, the following attributes were devised to characterize a line triple:

- length of each line segment;
- type of middle line segment, straight or arc; and
- magnitude of each of the two deflection angles.

In the current algorithm, the two outside segments of a line triple must be straight lines, while the inside segment can be either straight or circular.

Importantly, only one pair of matching line triples is needed to determine the relative alignment of two images. For example, in Figure 1, line triple AB-BC-CD in the left image corresponds to line triple ab-bc-cd in the right image. Because there are four nodes (e.g., nodes A, B, C, and D on the left image in Figure 1) in a line triple, the corresponding coordinates of the four nodes from the two images permit determination of the conformal transformation parameters between the two images. If more than one pair of matching line triples is found from different parts of the images, a stronger solution can be obtained and matching errors can possibly be detected and eliminated.

## THE ALGORITHM

The image alignment algorithm consists of four basic steps:

(1) extraction of edges,
(2) formation of line segments,
(3) selection of line triples, and
(4) triple matching and image alignment.

### EXTRACTION OF EDGES

A simplified implementation of the Canny edge detector (Canny, 1986) is employed to locate the edge points in an image. A single width ($\sigma$) Gaussian operator is used for smoothing the image, as opposed to the multi-width approach proposed by Canny. As with the two threshold values used by the edge detector to avoid streaking (so-called hysteresis), $\sigma$ is a parameter that can be tuned to the content of each image.

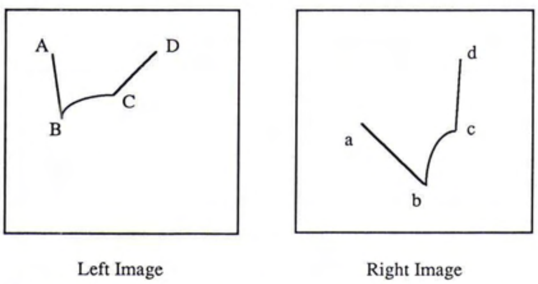


FIG. 1. Matching line triples in a stereo pair of images.

Adjoining edge points are linked together to form edges by the eight-connectivity approach (Jain, 1986). The end outputs of this step are edges, each of which is represented as a string of $(x,y)$ coordinates of the edge points. Each string can either be open-ended or form a closed loop. Figure 2 shows the edges extracted from the left image shown in Figure 4a.

### FORMATION OF LINE SEGMENTS

The purpose of this step is to form line segments from each string of edge pixels. The outputs from this step are strings of line segments, with each segment being defined by its end coordinates and classified as either a straight line or a circular arc.

The $x,y$ coordinates of the edge points in a string are first transformed into $(\psi, s)$ coordinates, in which $\psi$ is the direction angle of the tangent at an edge point and $s$ is the distance of the edge point from the starting point of the string. The parameter $\psi_i$ for pixel $i$, which has the coordinates $x_i$ and $y_i$, is computed as follows:

$$\psi_i = \frac{1}{2}\left[\tan^{-1}\left(\frac{y_i - y_{i-1}}{x_i - x_{i-1}}\right) + \tan^{-1}\left(\frac{y_{i+1} - y_i}{x_{i+1} - x_i}\right)\right]$$

where $\psi_i$ has a value ranging from $-\pi$ to $+\pi$.

Characteristics of the $\psi$-$s$ representation were described by Ballard and Brown (1982), and by Greenfeld and Schenk (1989). In $\psi$-$s$ space, straight lines and circular arcs in $x$-$y$ space appear as horizontal lines and inclined straight lines respectively. Figure 3a shows a string of edge points defined by their $x$-$y$ coordinates, and Figure 3b shows the corresponding points after being transformed into $\psi$-$s$ space.

It is obvious in Figure 3a that the string of edge points can be segmented into four segments: ab, bc, cd, and de. The points b, c, and d are called the dominant points in the string (Teh and Chin, 1988). The locations of these dominant points can be determined by analysis of the derivative function of the $\psi$-$s$ representation of an edge. In our current algorithm, the slope ($\kappa_i$) at edge point $i$ in $\psi$-$s$ space is computed as follows:

$$\kappa_i = \frac{\psi_{i+1} - \psi_{i-1}}{s_{i+1} - s_{i-1}}$$

The dominant points are located as points where the values of $\psi$ are local maxima or minima with absolute values exceeding a user-specified amount, such as $\pi/4$. Figure 3c shows a plot of the slope ($\kappa$) against the distance s for the string of edge points shown in Figure 3a. The dominant points are located at points b, c, and d.

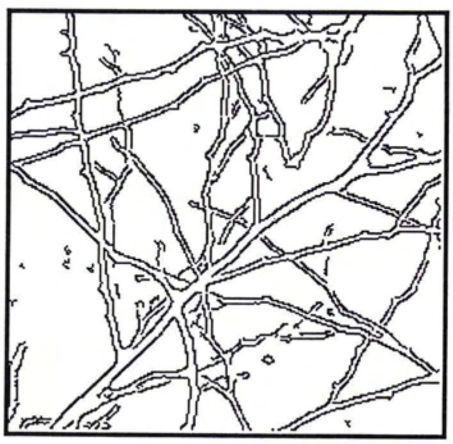

FIG. 2. Example of edges extracted by the Canny edge detector.

Once these dominant points have been found, the edge points between two adjacent dominant points are fitted to a straight line in $\psi$-$s$ space. Of these lines, those having a slope exceeding a prescribed threshold are classified as circular arc segments (in $x$-$y$ space), while the other line segments are classified as straight lines. For the string of edge points shown in Figure 3a, the resulting line segments are shown in Figure 3d.

After some initial testing, it was found that a one-pass segmentation scheme as described above often yielded unsatisfactory results. If the threshold for $\kappa_i$ were set too low, there would be too many dominant points. On the other hand, if the threshold were set too high, there would be too few dominant points. This problem was resolved by adopting a two-pass process. In the initial pass, the threshold for $\kappa_i$ was set relatively low, such as $\pi/4$ or $\pi/6$ radian/pixel. After the line segments had been formed as described above, any two adjoining segments which did not differ significantly in slope were merged to form a single segment. This procedure was found to yield satisfactory results.

### Selection of Line Triples

The line segments in each string are grouped into line triples, each consisting of three segments. For example, two triples, (ab, bc, cd) and (bc, cd, de), can be formed from the string of line segments shown in Figure 3d. For each triple, the lengths of the three individual segments and the two included deflection angles are computed. A line triple is selected for the matching operation at the next step only if (1) the length of each of the three segment is more than a pre-specified quantity, such as 10 pixels; (2) each of the two deflection angles exceeds a pre-specified minimum value, such as 0.3 radian; and (3) the end segments are straight. Of the two triples shown in Figure 3d, the triple (ab, bc, cd) meets all three criteria, while triple (bc, cd, de) fails to meet criterion 3.

### Triple Matching and Image Alignment

The triple in the left image with the longest center segment is first selected as the target for matching. All the triples in the right image are then checked for similarity with this target triple. A triple in the right image is considered as a potential match candidate if the following similarity conditions are satisfied:

- the center segment is of the same type (i.e., either straight or circular arc),
- the difference in length between each of three segments is within the pre-specified threshold; and
- the difference in each of the two deflection angles is within a pre-specified tolerance.

This set of criteria places emphasis on matching the general shapes of the triples. The orientations of the triples within each image are irrelevant, thus permitting large differences in kappa rotation between the two images. Because there is only one correct matching triple on the right image for a given target triple on the left image, the similarity thresholds for lengths and deflection angles should be set relatively loose so that the possibility of rejecting the correct match at this stage is minimized.

All the triples on the right image meeting the above similarity criteria are considered as potential match candidates, and are further evaluated. Suppose that a potential match candidate (ab, bc, cd) has been found for the triple (AB, BC, CD). The coordinates ($x$, $y$) of the dominant points (A, B, C, D, a, b, c, and d) are then used in a least-squares solution to compute the conformal transformation parameters of the left image with respect to the right image. In order to evaluate the validity of this match, all remaining triples on the left image are transformed to the coordinate system of the right image using the computed transformation parameters. Ideally, every transformed triple from the left image should be validated by a similar triple (i.e., similar in lengths, deflection angles, and types of middle segment) at or near its projected location of center of gravity on the right image. Because of occlusion, imperfection in the edge extraction and line segmentation processes, and difference in viewing angles of the two images, there may not be a one-to-one correspondence of triples in the two images. However, the larger the number of transformed triples that are each validated by a similar triple at or near its projected center of gravity on the right image, the greater is the probability that (ab, bc, cd) is the correct match for (AB, BC, CD). If no triple, other than itself, can be validated after this coordinate transformation, the triple (ab, bc, cd) is eliminated from further consideration.

All the potential match candidates for (AB, BC, CD) are similarly evaluated. The match candidate having the largest number of validated triples after coordinate transformation is considered the most likely match. All the corresponding triples, paired together from the validation process, are then used together to provide an improved solution for the transformation parameters. The image alignment process is thus successfully concluded.

If a match cannot be found for the triple on the left image with the longest center segment, then the above process is repeated for the triple with the next longest center segment. If

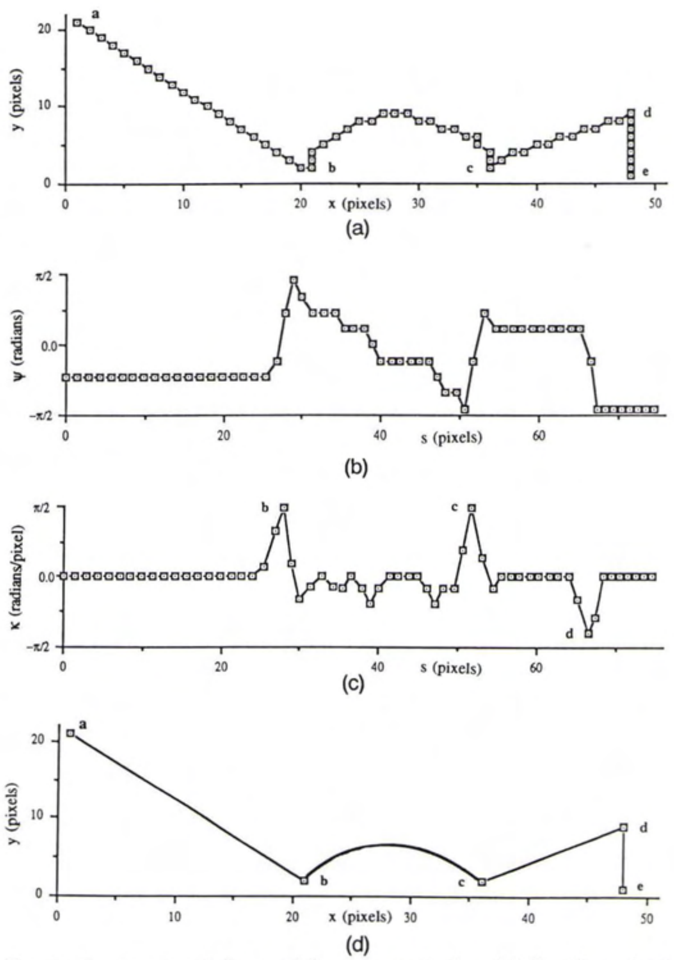


FIG. 3. $\psi$-$s$ representation and line segmentation. (a) A string of edge points in $x$-$y$ space. (b) $\psi$-$s$ representation of the edge. (c) Derivative function of the $\psi$-$s$ representation. (d) Resulting segmentation of the edge into straight lines and a circular arc.

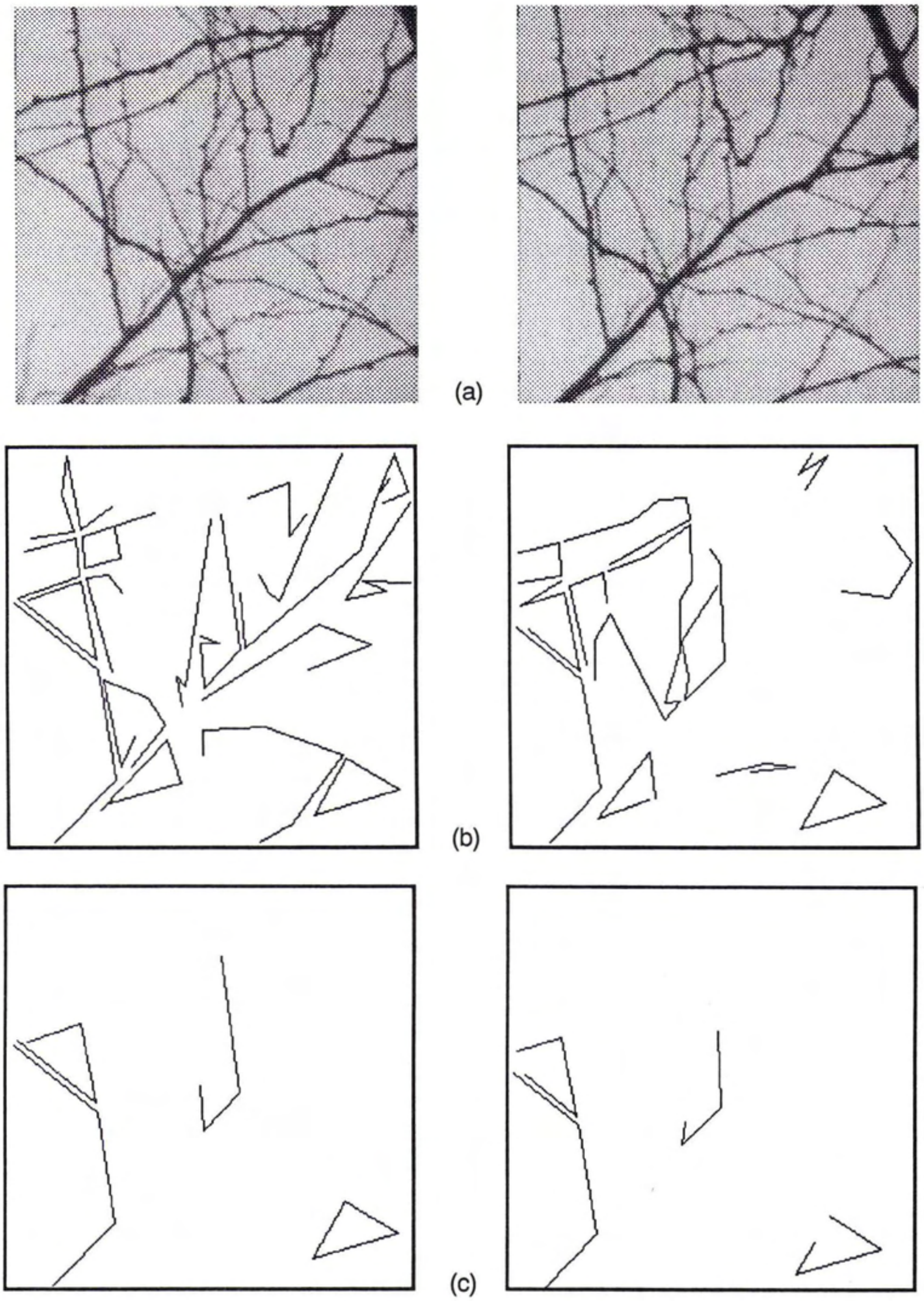


FIG. 4. Test results of a tree scene. (a) Left and right images of a stereo pair. (b) Line triples selected for matching. (c) Successfully matched line triples.

no match is found after all the triples on the left image have been attempted, then the image alignment ends in failure.

## EXPERIMENTAL RESULTS

The algorithm has been successfully tested using several pairs of stereo images. Figures 4 and 5 show the results obtained for two pairs. Figure 4a shows images of a tree scene, which were included in the standard test images provided by Working Group III/4 of ISPRS(Gulch, 1988). The images, each composed of 240 by 240 pixels, were reported to be digitized from photographs taken with an amateur camera. The image scale is approximately 1/30. The pair was categorized by ISPRS Working Group III/4 as "showing large differences in depth, several edges and possibly some occlusions." The line triples selected by the algorithm for matching are shown in Figure 4b, and the successfully matched triples are shown in Figure 4c. Tolerance thresholds of 50 percent and 0.5 radian were used for matching lengths of line segments and the deflection angles, respectively. While a successful alignment of the two images was achieved, Figure 4 shows that the algorithm clearly did not make use of the dominant features in the image, such as the large tree branch that spans from the lower left corner of the image to the upper right

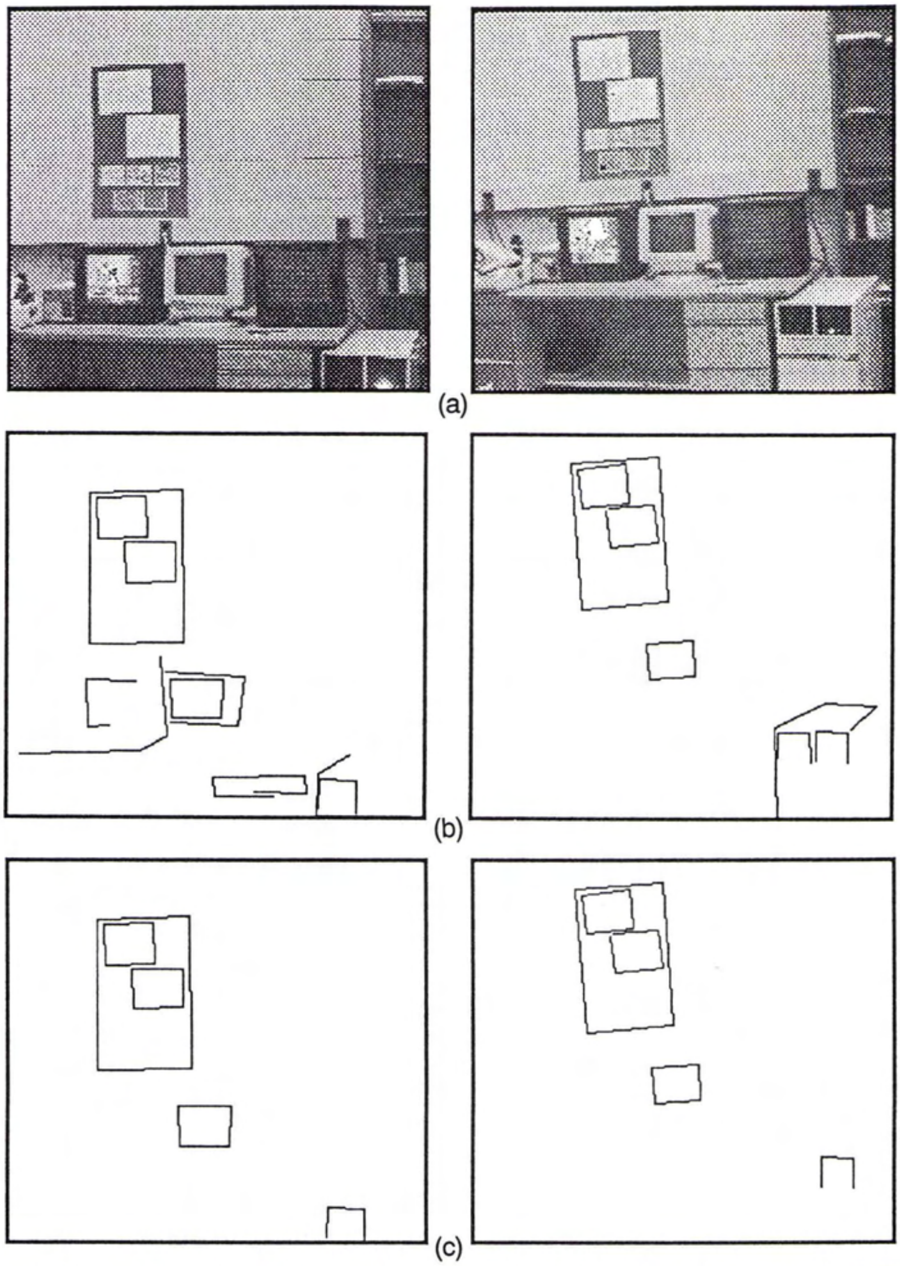

FIG. 5. Test results of a computer scene. (a) Left and right images of a stereo pair. (b) Line triples selected for matching. (c) Successfully matched line triples.

corner. This is due to the fact that the current algorithm employs only line segments (i.e., edges) for image alignment. It does not recognize image features that have an areal extent, and it makes no attempt to associate the different edges to form object entities. While human vision can instinctively identify the dominant linear pattern from the edges in Figure 2, the current algorithm can only associate the edge segments in term of triples.

Figure 5a shows the stereo images of a computer station. The images were acquired using two CCD video cameras. Each image consists of 256 by 256 pixels. This scene includes several objects that are rectangular in shape, but different in size. The line-triples selected for matching are shown in Figure 5b, and the successfully matched triples in Figure 5c. Tolerance thresholds of 30 percent and 0.5 radian were used for matching lengths of line segments and the deflection angles, respectively. The lower threshold for length was needed to avoid mismatching of similar rectangular-shaped objects in the scene. In fact, there is a mismatch in the lower right corners of the images in Figure 5c. Even by reducing the length threshold to 10 percent, this mismatch could not be avoided. The two rectangular features in the right image are simply too similar in size and shape, and too close to each other. It can also be noted from Figure 5 that the line-triple algorithm does not use for matching such prominent features as the long vertical edge of the bookcase, and the long horizontal edge of the desk.

## CONCLUSION

The test results demonstrated the promise as well as limitations of the current algorithm. It can be seen from Figures 4c and 5c that the number of matched triples in both test cases was much smaller than one might expect for scenes with such a high content of line structures. The reason for this is directly attributable to the non-optimality of the edge extraction process. Like all edge detectors based on the step-edge assumption, the Canny edge detector performs poorly at non-step edges, edge corners, and edge junctions. It was not designed to handle such events. As a result, many of the edges extracted from the two test scenes were discontinuous at corners and junctions. However, the reliable extraction of line triples, as used in this algorithm, is directly dependent on how well extended edges, containing corners and junctions, can be identified. One possible solution to this problem is to use an explicit corner/junction detector to supplement the Canny edge detector.

Another limitation of the current approach is the need for proper selection of thresholds for edge extraction, line segmentation from the $\psi$-$s$ representation, and triple matching. The values for these thresholds depend largely on image content, image quality, and imaging geometry. In any application environment, experience can be gained through trial and error to provide guidelines for the selection of these threshold values. Nevertheless, because the goal is complete automation of the alignment process, there is need to further study the sensitivity of this algorithm to these thresholds and to modify the algorithm to reduce this sensitivity.

Can the algorithm be used to align two images that have large differences in tilt, kappa rotation, and/or scale? Because the algorithm attempts to find matching features according to their sizes and shapes, some corresponding features must appear similar in the two images. If there is a large difference in scale between the two images, the magnitude of difference must be approximately known. Difference in kappa rotation (i.e. rotation about the optical axis) will only result in a different orientation of the corresponding features in the two images and, therefore, should have no adverse effect on the matching algorithm. Small differences in shapes caused by tilt can be accommodated by proper selection of the thresholds used in matching. However, if the difference in tilt is such that no similar feature can be found in the two images, then the algorithm would fail.

The algorithm can be made more robust in general, and more tolerant to tilt differences in particular, by increasing the potential pool of matchable features in an image. The definition for a line triple can be expanded to include triples that have two or three circular segments, as well as triples that have three line segments meeting at a single junction point. Other basic line features, such as long straight segments, circles, and ellipses, can be incorporated into the algorithm. Large regions of uniform image intensity, as well as interest points which represent small windows possessing characteristic image features, may also be used to supplement line triples as matching features.

In conclusion, this study has demonstrated that the basic approach of using simple image features for automatic image alignment is feasible; and continuing effort is being directed at implementing some of the above ideas for potential improvements.

## ACKNOWLEDGMENT

The research reported in this paper was conducted as part of the University of Illinois Advanced Construction Technology Center research program sponsored by the U.S. Army Research Office under the DoD-University Research Initiative Program.

## REFERENCES


Ayache, N., and O.D. Faugeras, 1986. HYPER: A New Approach for the Recognition and Positioning of Two-Dimensional Objects, *IEEE PAMI*, 8(1):44-45.

Ballard, D.H., and C.M. Brown, 1982. *Computer Vision*, Prentice-Hall Inc., N. J.

Canny, J., 1986. A Computational Approach to Edge Detection, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, PAMI-8:679-698.

Clark, C.S., D.K. Conti, W.O. Eckhardt, T.A. McCulloh, R. Nevatia, and D.Y. Tseng, 1980. Matching of Natural Terrain Scenes, *Int. Conf. on Pattern Recognition*, Miami, Florida.

Förstner, W., 1986. A Feature Based Correspondence Algorithm for Image Matching, *International Archives of Photogrammetry and Remote Sensing*, 26-3/3:150-156.

Greenfeld, J.S., 1990. Automatic Relative Orientation Based on Computer Vision Principles. *Tech. Papers: ACSM-ASPRS Annual Convention*, Vol. 5, Denver, Colorado.

Greenfeld, J.S., and A. F. Schenk, 1989. Experiments with Edge-Based Stereo Matching, *Photogrammetric Engineering & Remote Sensing*, 55(12):1771-1777.

Gulch, E., 1988. Results of Test on Image Matching of ISPRS WG III/4, *International Archives of Photogrammetry and Remote Sensing*, 27-III:254-271.

Hannah, M.J., 1988. Digital Stereo Image Matching Techniques, *Int. Archives of Photogrammetry and Remote Sensing*, 27(3):280-293.

———, 1989. A System for Digital Stereo Image Matching. *Photogrammetric Engineering & Remote Sensing*, 55(12):1765-1770.

Jain, A.K., 1986. *Fundamentals of Digital Image Processing*, Prentice-Hall.

McKee, J.W., and J.K. Aggarwal, 1977. Computer Recognition of Partial Views of Curved Objects, *IEEE Computers*, C-26(8):790-800.

Price, K.E., 1984. Matching Closed Contours, *Workshop on Computer Vision: Representation and Control*, IEEE Comput. Society, pp. 130-134.

Teh, C.-H., and R.T. Chin, 1988. A Scale-Independent Dominant Point Detection Algorithm, *Proc. CVPR*, Ann Arbor, pp. 229-234.


(Received 21 February 1991; revised and accepted 30 December 1991)

## OFFICIAL NOTICE TO ALL CERTIFIED PHOTOGRAMMETRISTS

The ASPRS Board of Directors approved an expansion of the Certified Photogrammetrist Program that goes into effect **January 1, 1992.** After that date, all Certified Photogrammetrists **MUST** submit an application for recertification as a Photogrammetrist or for certification as a "Certified Mapping Scientist--Remote Sensing," or "Certified Mapping Scientist--GIS/LIS." Recertification is required every five years; fee for recertification application and evaluation is $125 for members of the Society, and $225 for non-members. Those that do not recertify, will be transferred into either an "Inactive" or "Retired" status.

**If you were certified between January 1, 1975 and January 1, 1987, (anyone with a certificate number lower than 725), you must comply with this notice.**

Each affected Certified Photogrammetrist has been sent a letter by Certified Mail, with the new forms and procedures. If you are reading this notice and have NOT received a letter, please call me immediately at 301-493-0290.

—William D. French, CAE
Secretary, ASPRS