MVA'90 IAPR Workshop on Machine Vision Applications Nov. 28-30,1990, Tokyo

A COMPUTER VISION SYSTEM THAT ASSEMBLES CANONICAL JIGSAW PUZZLES USING THE EUCLIDEAN SKELETON AND ISTHMUS CRITICAL POINTS.*

Roger W. Webster, Ph.D. and Paul W. Ross, D.E.
Robot Vision and Artificial Intelligence Laboratory
Department of Computer Science
Millersville University
Millersville, PA, USA 17551

Paul S. LaFollette, M.D. and Robert L. Stafford, Ph.D.
Department of Computer Science
Machine Vision Laboratory
Temple University
Philadelphia, PA, USA 19122

ABSTRACT

This paper describes a computer vision system which can assemble canonical jigsaw puzzles. The most novel aspect of this system is that the methodology presented here derives a new set of critical points which define a feature which can be used in matching partial boundaries (or contours) of planar regions. This global feature, called an *Isthmus*, can be efficiently and reliably computed from the Euclidean skeleton or Medial Axis Transformation (MAT) of an object. A heuristic matching technique using Isthmus critical points is applied to the partial boundary matching problem of jigsaw puzzle fitting. The Isthmus feature may also be a useful feature in a broader class of image processing problems such as: the narrowing of arteries in medical applications, geographic images, collision avoidance problems in robot path planning, and any application in which the point of narrowest necking of a planar region needs to be located.

* This work is based upon a project supported by the National Science Foundation (NSF) under Grant No. USE-9050371 and the Faculty Grants Committee of Millersville University.

1. INTRODUCTION.

Partial boundary matching (PBM) is the process of matching shapes based upon matching pieces of their respective boundaries. PBM techniques are useful in shape fitting or assembly applications where the entire boundary may not be available. Such applications include: industrial vision systems where parts touch or overlap, environments where shadows may affect portions of the boundary, and in scene analysis with occluded objects.

It is well known that the points which segment the boundary are referred to as *critical points*. Critical points which have been used include: sharp corners (discontinuity in curvature), points of inflection, and curvature maxima and minima. These points can be computed using local border processing (chain encoding) originally defined by Freeman [1]. In this paper we present a method for deriving a new set of critical points which describe a global feature (an *Isthmus*) which can be used in matching partial boundaries of planar regions.

1.1 The Jigsaw Puzzle Problem.

The task at hand is to assemble canonical jigsaw puzzles in a robust fashion. The jigsaw puzzle problem is originally summarized by Radack and Badler in [2] as: "Given a set of simply connected planar regions (silhouetted puzzle pieces), rotate and translate each piece so that the pieces fit together into one region, with no *significant* area gaps or overlapping pieces."

The jigsaw puzzle problem was chosen because it contains a number of problems endemic to many machine vision applications: shape description, partial boundary matching, pattern recognition, feature extraction, and heuristic matching. A survey of the different approaches to the jigsaw puzzle problem and previous work in partial boundary matching can be found in [3].

A puzzle piece is defined as a simply connected planar region. A jigsaw puzzle will be defined as a set of puzzle pieces which when properly assembled fit together into one region. All puzzle pieces are processed as binary images. Only shape information will be utilized in the solution. Pictorial information or scene analysis will not be a factor. Many apictorial jigsaw puzzles require a non-trivial degree of intelligence to assemble. The jigsaw puzzles used in this paper will have a unique solution, i.e., each puzzle piece has a unique location and orientation within the puzzle.

The essence of the jigsaw puzzle problem can be found where puzzle pieces mate. Two pieces which mate share a common border called the **Match Segment** (Fig. 1). The main task of a computer vision program to assemble a jigsaw puzzle is to locate and mate these Match Segments.

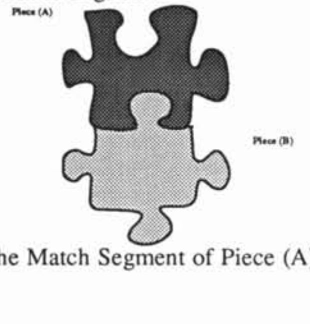

Fig. 1. The Match Segment of Piece (A) and Piece (B) .

Some jigsaw puzzles may have mating pieces in which the match segment contains no distinguishable or easily extractable critical points. The process of finding the match of puzzle pieces with match segments of this type becomes an arduous computational task. There is little alternative but to attempt to match each point with every other point along every other puzzle piece.

The previous methodologies used to perform jigsaw puzzle matching, Radack and Badler [2], Freeman and Garder [4], Hirota and Yoshikazu [5], Nagura, et. al. [6], are all based on the notion of extracting critical points from *local* border information. One of the problems with using these critical points: sharp corners, points of inflection, curvature maxima, and curvature minima, is that the number of match segments can become many times the number of puzzle pieces. If the number of critical points extracted is sufficiently large, the matching algorithm can take a prohibitive amount of computation time.

The puzzle pieces depicted in Fig. 2 would generate many inflection points and sharp corners that can yield large computation times for the matching process.

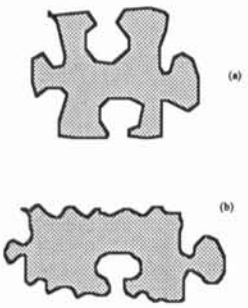

Fig. 2. Puzzle pieces which introduce many false candidate match segments. (a) Sharp corners as critical points. (b) Inflection points as critical points.

We present an approach to this problem based on a global feature called an Isthmus. The Isthmus feature yields a *pair* of critical points which are bounded together to produce a candidate match segment. These Isthmus Critical Points can be extracted from a global shape descriptor, the Medial Axis Transformation (MAT) or Euclidean skeleton. The MAT is a technique originally developed by Blum [7] which can reduce a planar object to a skeleton or stick figure.

This methodology of using a higher order entity (Isthmus Critical Points) rather than just single critical points (e.g. sharp corners), helps control the number of matches to evaluate. For example, a puzzle piece with N single critical points would yield potentially N * (N-1) candidate match segments. Thus, a piece with only 10 critical points may have, as an upper bound, 90 candidate match segments. The matching process can quickly get unruly. It should be noted that N * (N-1) is an upper bound and in the case of no backtracking not all of these segments would need to be checked.

2. ISTHMUS CRITICAL POINTS AND INTERLOCKING PUZZLES.

An inspection of conventional jigsaw puzzles shows that a large class of the match segments of pieces which mate also interlock. Isthmus Critical Points can be used to detect some interlocking shapes.

Fig. 3 illustrates the notion of Isthmus Critical Points. It can be defined as follows: "Given a planar region, an *Isthmus Line* is a chord partitioning the region whose length is locally minimum. An *Isthmus Point* is the midpoint of the Isthmus Line. The endpoints of the Isthmus Line will be called *Isthmus Critical Points* . " An Isthmus is a global feature of a planar region. Isthmus Critical Points can be derived from this feature.

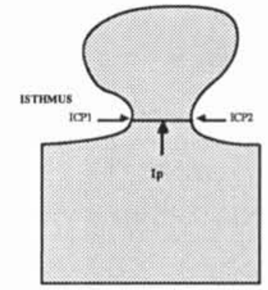

Fig. 3. Sample Isthmus Critical Points. The Isthmus of this shape is the line segment across the piece. The Isthmus Point is the midpoint of the Isthmus denoted by Ip. The Isthmus Critical Points, denoted by ICP1 and ICP2, are the border points which are minimally distant from Ip.

Isthmus Critical Points have several beneficial features: (a) An Isthmus defines a *pair* of critical points which are bounded together and yield a candidate match segment, (b) They can be efficiently extracted from a global shape descriptor, (c) They are rotation and translation invariant, (d) They are invariant to equal scaling in X and Y, (e) They can detect some interlocking shapes (puzzle pieces).

Two puzzle pieces are said to be *interlocking* iff they can not be assembled or disassembled (pulled apart) without taking the pieces out of the XY plane (see Fig. 4). Not all interlocking pieces contain Isthmus points. The "snake" example in Fig. 5b shows an interlocking puzzle piece without an Isthmus. However, a sampling of conventional jigsaw puzzles shows that many of the interlocking pieces do have Isthmus points. The algorithm for jigsaw puzzle matching described here is concerned with match segments which are interlocking and contain at least one unique Isthmus.

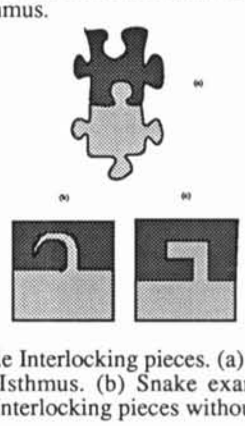

Fig. 4. Sample Interlocking pieces. (a) Conventional pieces with Isthmus. (b) Snake example without Isthmus. (c) Interlocking pieces without Isthmus.

Fig. 5 illustrates the notion of *positive* Isthmii and *negative* Isthmii. The matching algorithm attempts to mate a positive Isthmus of one piece with the matching negative Isthmus of another piece.

One of the more elegant features of Isthmus Critical Points is that they can be extracted from a global shape descriptor, the Medial Axis Transformation (MAT). Of course, since the MAT is equivalent to the border in terms of information content it follows that anything which can be found in the MAT can also be found from the border points. The MAT, however, contains global information, namely the opposite points of the boundary at each skeleton point. This makes the extraction of Isthmus Critical Points from the skeleton computationally more efficient than from local border processing.

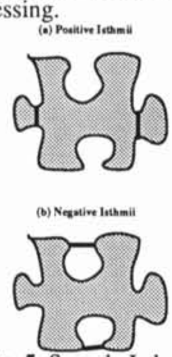

Fig. 5. Sample Isthmii.

3. A MULTIPLICITY OF SKELETONS.

The term "skeleton" has been widely referenced in the literature. This term has a number of aliases which include: Medial Axis Transform (MAT), Medial Axis (MA), Medial Axis Skeleton (MAS), Symmetric Axis Transform (SAT), Stick Figure, and Distance Transform. Indeed, the concept of a skeleton has been used to refer to a number of different constructs. The basic premise, however, is to reduce a planar shape in a digital image to a line drawing or stick figure. The stick figure allows for easier extraction of shape information.

Skeleton algorithms are used in numerous applications including: printed circuit board inspection, asbestos fiber counting, character recognition, chromosome shape analysis, soil cracking patterns, fingerprint classification, facsimile, and data reduction for map storage. Since Rosenfeld and Pfaltz's paper in 1967 [8] there have been many algorithms proposed to develop the skeleton of a discrete planar region.

The various skeleton algorithms, however, do not produce the same results. Each algorithm produces slightly different skeletons. The fundamental problem with the development of the *actual* Euclidean skeleton in a digital image is the difficulty of measuring the equality of distance between pixels in a grid (the equidistant property problem). The jigsaw puzzle problem requires a skeleton algorithm which uses Euclidean distances in order for the skeleton to be rotation invariant. We use our own Euclidean Skeleton algorithm which can be found in [3].

4. HARDWARE AND SOFTWARE.

The vision system used is the Imaging Technology FG101 image processing system. The frame buffer provides 1024 x 1024 pixel resolution with a 12 bit grey shade. A CCD color camera is used as the input device. A DEC MicroVAXII™ computer is the main computer vision controller. The operating system is UNIX™ (Ultrix-32m™). The vision boards are addressed via a device driver. The UNIX device driver and all the application software is written in the 'C' language. MicroVAXII and Ultrix-32m are trademarks of the Digital Equipment Corporation. Unix is a trademark of AT&T.

5. THE PUZZLE MATCHING PROCESS.

The process starts by placing each puzzle piece at a random orientation in the field of view of the camera. A backlight box is used to minimize the effects of shadowing. This produces a crisp binary image. After a frame grab operation is performed, the next task is to find the object's border. The puzzle piece's location in the image is not known a priori. A simple border following algorithm returns the array of border points and the area of interest (AOI) box.

5.1. Computing Isthmus Points from the Skeleton.

Using the area of interest (AOI) box a Euclidean skeleton algorithm is then performed on the piece. The next step is to recursively traverse the branches of the skeleton looking for local minima. A *local distance minima* is defined as follows: Let S be a skeleton. A skeleton is a line drawing comprised of line segments. All such segments are connected. Let a skeleton segment S[i] be the union of all skeletal points between an endpoint and an intersection point. The union of all S[1], S[2], S[3]... S[n] for 1 <= i <= n of S[i] = S. Also, S[i] intersected with S[j] is null for all i <> j. Along each skeleton segment is a function that maps the position on the segment into elevations.

A local minima point along a skeleton segment indicates an *Isthmus Point.* The Isthmus Critical Points (ICP) are computed as the endpoints of the Isthmus Line. The algorithm then marks those border points on the puzzle piece which are Isthmus Critical Points. This process recursively traverses the skeleton segments. The algorithm when finished stores all the ICPs in the main data structure and accents them on the monitor. Fig. 6 shows the Euclidean *endoskeleton* skeleton extracted from the Euclidean distances and the positive Isthmus critical points.

The process of extracting the ICPs from the *exoskeleton* is also performed and the negative Isthmus Critical Points are stored (Fig. 7). The process of extracting Isthmus Critical Points is then repeated for each puzzle piece and stored in the system data structure.

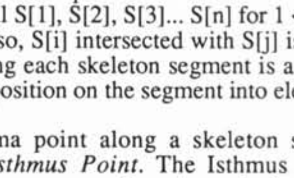

Fig. 6. Positive Isthmus Critical Points.
The *endoskeleton* is the internal skeleton of a planar region.

The *exoskeleton* is the external skeleton of a planar region (the skeleton of the area outside the object). The exoskeleton calculation is bounded by the AOI of the object.

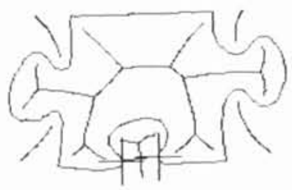

Fig. 7. Negative Isthmus Critical Points.

5.2. Heuristic Matching.

The matching algorithm begins by attempting to mate *positive* Isthmus match segments with *negative* Isthmus match segments. This yields candidate match segments.

The number of matches of all positive match segments to all negative match segments can be very large. In a typical 24 piece jigsaw puzzle there can be over 2000 matches to process. To reduce the search space many matches can be eliminated with some heuristic processing. For example: IF the Distance between the negative Isthmus Critical Point (ICP) pair is less than the distance between the positive ICP pair THEN discard from the matching process. Isthmus points are the points of narrowest necking. If the negative Isthmus point opening is less than the positive's Isthmus distance it can't possibly fit.

Another heuristic used is: IF the absolute value of the difference of the two path lengths (of the match segments) is greater than 50% of the longer segment THEN discard from the match. This heuristic is used to simply weed out two match segments in which their respective lengths are not even close.

After the heuristic preprocessor has removed a number of potentially bad candidate match segments then the viable candidate matches are stored in the system data structure. Next, some method of partial boundary matching is necessary. The method used here is a correlation of two functions called the Isthmus Distance Function.

5.3. The Isthmus Distance Function.

The *Isthmus Distance Function* (IDF) is derived by taking the distance to the border points from the Isthmus Point. The IDF is a function of path length. The pair of ICPs serve as the start and stop points of the IDF function. An Isthmus defines a pair of ICPs which are bounded together by the Isthmus extraction process. The IDF is derived for both positive and negative Isthmii. Fig. 8 and Fig. 9 illustrate the IDF of two Isthmii that mate.

The IDF's are then correlated to determine best fit. The correlation procedure depicted in Eq. 1 is used. The maximum value stored in R(m) indicates the offsets where g(x-m) best fits f(x). In Eq. 1 R(m) is the correlation array f(x) and g(x-m) are the two IDF functions to be correlated. The summations are taken over the range where g(x-m) is defined. R(m) is scaled to the range -1 to 1.

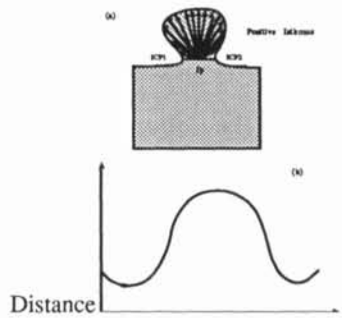

Fig. 8. The Isthmus Distance Function (IDF) Positive Isthmus.

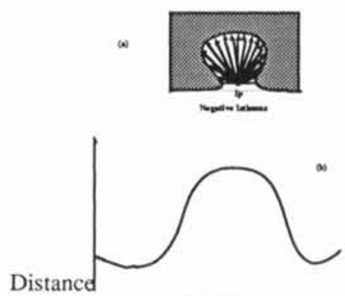

Fig. 9. The Isthmus Distance Function (IDF) Negative Isthmus.

$$R(m) = \frac{\sum \left[f(x) - \frac{\Sigma f(x)}{n} \right] \left[g(x-m) - \frac{\Sigma g(x-m)}{n} \right]}{\sqrt{\sum \left[f(x) - \frac{\Sigma f(x)}{n} \right]^2 \sum \left[g(x-m) - \frac{\Sigma g(x-m)}{n} \right]^2}}$$

(Eq. 1).

The correlation values are then used to determine where the best fit occurs within the two match segments. The offset produced by the correlation function is used to align the two match segments. The area or GAP is then calculated. This GAP measure is the amount of space in between the two puzzle pieces when they are fitted or assembled. The GAP measure is the sum of both the positive gap (POS-GAP) which is the *overlapping* portions of the match and negative gap (NEG-GAP) which is the *underlapping* portions.

The GAP represents how well the shapes (pieces) fit. Normally, one would expect that the smaller the gap, the better the match. However, it is possible that a short match segment (small path length) will have a small GAP but not necessarily be a good fit. Thus, the average gap as a function of path length is calculated. The PATH-LENGTH is the length of the match segment.

The average gap is represented by taking the gap divided by the path length of the match segment:

AVE-GAP = GAP / PATH-LENGTH. The matching algorithm takes the smallest value of AVE-GAP for the best fit. The program then rotates and translates the puzzle pieces into assembled position. The correlation offset is used to properly align the two mating puzzle pieces.

After the program has mated two match segments it removes them from the list of candidate matches. It also gets rid of impossibilities, i.e. after mating piece 1 with piece 2 one can not thereafter have a match of piece 2 with piece 1 with different Isthmii. A rudimentary check is then performed to determine if there is too much overlap. The program proceeds down the list of AVE-GAP values until all puzzle pieces are mated or a stopping condition arises.

Puzzle Matching Algorithm.

```
Algorithm Puzzle Piece Matching.
Begin
   For each Negative Isthmus on each Piece
   Begin
      For each Positive Isthmus on the N-1
          pieces
      Begin
         If any Heuristic Tests Fail Then
             Discard from the match table.
         Else
          Begin
             Correlate the two IDFs.
             Store Highest Correlation
               value in the match table.
              Compute AVE-GAP.
          End {else}
        End {for other N-1 pieces}
      End {for each Negative}

      QuickSort  match table by AVE-GAP.
       For each entry in match table:
       Begin
             Print Puzzle Piece Mating.
             Move Puzzle Pieces.
             (Rotate and Translate in Image).
             Eliminate impossibilities.
             Check intolerable overlaps.
        End  {for each entry}
End. {end Algorithm }
```

6. EXPERIMENTAL RESULTS

The 24 piece jigsaw puzzle used as a test case was chosen at random from a set of conventional off the shelf jigsaw puzzles. The puzzle is a difficult one because it contains many different Isthmii and many different match segments. There are some match segments which appear, to the human eye, to be very similar.

The shape matching program calculated 2377 possible matches. The heuristic processing reduced the search space to 126 possible matches (2251 matches eliminated). For each of the 126 possible matches the two corresponding match segments were correlated as per the equation in Eq. 1. Correlation is expensive so it is important to reduce the search space via the heuristics. After the correlation, additional heuristics can be performed to further reduce the search space. Correlation values less than 0.80 are discarded from the space because this suggests a poor correlation (poor fit).

The GAP is then calculated on the reduced set of 126. Recall, the GAP is calculated as: GAP = POS-GAP + NEG-GAP where POS-GAP is the amount of overlap and NEG-GAP is the underlap of two pieces assembled together. Any match with a GAP to POS-GAP ratio of greater than 50% can be eliminated because this constitutes an intolerable overlapping condition.

The puzzle matching algorithm can not be used blindly, however. As the algorithm assembles more and more pieces eventually it will incorrectly assemble two pieces. As with human puzzle assembly, if two pieces actually do fit together they will be assembled. Later the human discovers that this is an incorrect match because there are pieces that do not fit anywhere or two puzzle sections can not be coagulated. The human performs backtracking to discover the mismatch.

Our computer program correctly assembled almost all of the 37 actual matches in the 24 piece puzzle used as a test case. The one mismatch (Fig. 10) caused a *coagulation* problem which forced a stopping condition. This mismatch can be handled with backtracking.

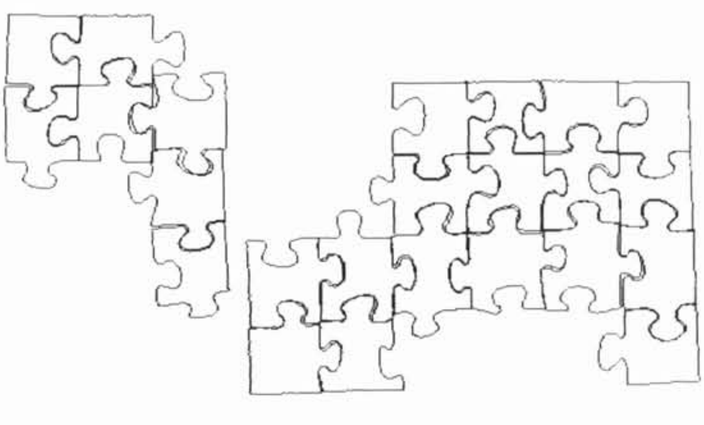

Fig. 10. Assembly Mismatch.

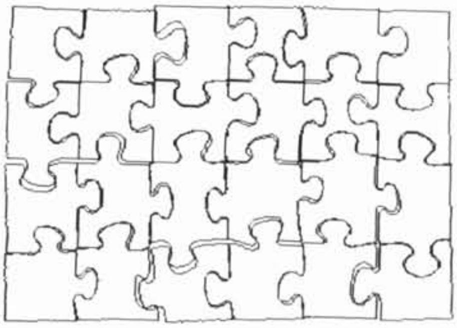

Fig. 11. Completely Assembled Puzzle.

Fig. 11 shows the entire puzzle assembled with backtracking implemented. The puzzle pieces are linearly scaled down in x and y to fit on the monitor.

Although the test case 24 piece puzzle is a completely interlocking puzzle, we expect that the puzzle matching algorithm will work on a subset of non-completely interlocking puzzles as well. For example, Fig. 12 depicts three pieces in which Piece A is assembled with Piece C even though they are not interlocking. This is because Piece A is assembled to Piece B and Piece B to Piece C. Piece A and B are interlocking. Piece B and C are interlocking. However, Piece A to C is not interlocking. It is therefore possible that the algorithms described in this paper will correctly assemble some non-completely interlocking jigsaw puzzles as well.

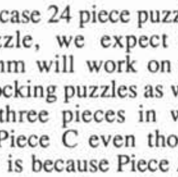

Fig. 12. Non-completely Interlocking Pieces.

7. CONCLUSION

In addition to an effective solution to the jigsaw puzzle problem, we have advanced the notion of a global feature of a planar object, called an *Isthmus*. This paper has presented a method for reliably computing the Isthmus feature from the Euclidean skeleton of an object and deriving a new set of critical points which describe the feature.

The key idea is that the Isthmus may also be a useful feature in a broader class of image processing problems such as: the narrowing of arteries in biomedical applications, geographic images, the obstacle avoidance problem in robot path planning, and any application in which one needs to find the point of narrowest necking of a planar region.

The Euclidean skeleton and the notion of Isthmii may be applied to the obstacle avoidance problem in robot path planning. The exoskeleton of a room with obstacles delineates the possible paths across the room that are equidistant from the obstacles that are in the way. The Isthmii of the exoskeleton are the points of narrowest necking, i.e., points where the robot might get stuck.

In terms of jigsaw puzzle matching, the methodology presented here uses a higher order entity (Isthmus Critical Points) rather than just single critical points (e.g. sharp corners, inflection points), which helps control the number of matches to evaluate. The class of jigsaw puzzles that this method can solve is conventional interlocking puzzles in which each match segment contains a unique Isthmus. Non-unique Isthmii (match segments) can be resolved with the use of backtracking.

The techniques and algorithms described in this paper were implemented in a set of computer programs and applied to the partial boundary matching problem of jigsaw puzzle fitting. An illustration of an actual assembly of a 24 piece jigsaw puzzle by these programs was given.

Acknowledgements

The authors wish to thank Dr. Giorgio Ingargiola and Dr. Charles Kapps of the Department of Computer Science at Temple University and Dr. Marvin Ziskin of the Temple University School of Medicine for many valuable discussions and comments on earlier drafts of this paper.

REFERENCES

[1] H. Freeman, "Boundary Encoding and Processing", In *Picture Processing and Psychopictorics*, ed. B. Lipkin and A. Rosenfeld, (New York: Academic Press, 1970), pp. 101-120.

[2] G. Radack and N. Badler, "Jigsaw Puzzle Matching Using a Boundary-Centered Polar Encoding," *Computer Graphics, and Image Processing*, Vol. 19, pp. 1-2, May, 1982.

[3] R. Webster, "Partial Boundary Matching and Shape Fitting Using the Medial Axis Transformation,", (*Ph.D. Dissertation, Temple University)*, Ann Arbor, Michigan: University Microfilms Incorporated, pp. 1-208, May, 1988.

[4] H. Freeman and L. Garder, "Apictorial Jigsaw Puzzles: The Computer Solution to a Problem in Pattern Recognition," *IEEE Transactions on Electronic Computers*, EC-13, pp. 118-127, April, 1964.

[5] Hirota and O. Yoshikazu, "Image Recognition in Jigsaw Puzzle Assembly Robot System (JPARS)," in *Bulletin of the College of Engineering* (Hosei University, Japan), pp. 87-93, March, 1986.

[6] K. Nagura et al., "Partial Contour Processing Using Curvature Function - Assembly of Jigsaw Puzzle and Recognition of Moving Figures," *Systems Computing* ,Vol 2, pp. 30-39, February, 1986.

[7] H. Blum, "A Transformation for Extracting New Descriptors of Shape," *Proceedings of the Symposium on Models for Perception of Speech and Visual Form*, ed. Weiant Whaten-Dunn, (Cambridge, Massachusetts: MIT Press, 1967), pp. 362-380.

[8] J.L. Pfaltz and A. Rosenfeld, "Computer Representation of Planar Regions by Their Skeletons," *Communications of the ACM*, Vol. 10, no.2, pp. 119-122, February, 1967.