Stereopsis and Image Registration from Extended Edge Features in the Absence of Camera Pose Information

N. M. Vaidya and K. L. Boyer
Signal Analysis and Machine Perception Laboratory
Department of Electrical Engineering
The Ohio State University
Columbus, OH 43210

Abstract

We solve the stereo correspondence problem in uncalibrated domains using extended edge contours as a source of primitives, as opposed to traditional point-based algorithms. This work represents a significant new approach to implementation of the structural stereopsis concept of Boyer and Kak [1], in particular with regard to speed. Judiciously exploiting the contiguity relation among primitives, correspondence solutions without prior knowledge of the epipolar condition for feature-rich stereopairs previously requiring several days of processing by the Boyer and Kak algorithm are now acquired in tens of seconds on the same equipment. This approach offers potentially in rapid automatic rectification of stereo images for which the camera geometry is known only vaguely a priori.

I Introduction

This work addresses the correspondence problem that arises in stereo vision systems. The stereo correspondence problem has been pursued by many researchers. (We recommend [2] for a review). Most techniques generally make only weak use of any sort of structural representation of the scene. Some like [3, 4] do incorporate the use of relational constraints in matching edge features, but almost all of these paradigms require a well-calibrated camera system, since this provides geometric (epipolar) constraints which are used to direct the search process(es) for homologous entities in the two images.

When dealing with some applications of passive stereopsis, such as aerial photogrammetry, the positions and orientations of the cameras are not known, even relative to one another, a priori, and we must determine them from the stereo image data itself and not from special calibration scenes. In robotics, it is advisable to check the camera parameters periodically. Robust estimation from analytical photogrammetry may be applied to this calibration problem, but requires that many corresponding points be identified in the two images. Due to the lack of imaging-geometry based search-constraints, rapid identification of correspondents automatically by computer for an uncalibrated camera pair is difficult. This paper presents an important step in developing such a system: fast stereopsis from extended photometric edge features.

Boyer and Kak [1] showed that the rich information content of a structural description of the scene, constructed from extended primitives, may be used in solving the stereo correspondence problem for unconstrained situations. The appeal of the structural approach is that, firstly, it permits greater use of inter-primitive relational constraints and, secondly, it requires only an approximate (probabilistic) characterization of the image-to-image structural distortion, and not detailed knowledge of the epipolar geometry. However in the form presented in [1], such an approach requires enormous computational burden and would literally take days to run on images of detail rich scenes.

In this work, we retain the mathematical formulation of the information-theoretic approach to structural matching of Boyer and Kak, but dispense with their backtracking tree search algorithm. We have reformulated the search process in an opportunistic framework which exploits the contiguity relation among primitives to maximum advantage in building the interprimitive mapping function. This has resulted in a search speed several orders of magnitude faster.

II Review and Drawbacks of Structural Stereopsis

Very briefly, we review the mathematical formulation of structural stereopsis and the original Boyer and Kak search technique to contrast that work with the present effort. Assume one has extracted primitive features (by a low level process like edge detection) and
their relations in each of two images of a scene. Let $D_L = (P, R)$ and $D_R = (Q, S)$ represent the structural descriptions of the left and right images, respectively. The sets $P$ and $Q$ are primitives of the left and right views, respectively, while $R$ and $S$ are the sets of named $N$-ary parametric relations over the respective primitive sets. A primitive is characterized by a set of attribute-value pairs as $P = \{p_1, \ldots, p_n\}$ and $Q = \{q_1, \ldots, q_m\}$ with each primitive defined in the product set $p_i, q_i \subseteq A \times V$ with $A$ a set of attributes and $V$ a set of values for those attributes. The relations $R$ and $S$ over $P$ and $Q$ respectively are given as $R = \{R_1, \ldots, R_K\}$ and $S = \{S_1, \ldots, S_K\}$, where for $k = 1, \ldots, K$, $R_k = (NR_k, R_k, fR_k)$ and $S_k = (NS_k, S_k, fS_k)$. Here $NR_k$ and $NS_k$ represent the name of the $k^{th}$ relation and $R_k$ and $S_k$ are the sets of $M_k$-tuples of primitives having that relationship. The terms $fR_k$ and $fS_k$ represent parameter functions applied to the tuples in $R_k$ and $S_k$. The solution to the correspondence problem is an interprimitive mapping function, $h: P \rightarrow Q$, taking elements of the left image primitive set to elements of the right image primitive set.

The inexact matching of parametric structural descriptions entails modeling a primitive as an information source, and the correspondence process as an information channel. The distance between two primitive sets $P$ and $Q$ under the mapping $h$ is the summation of the conditional information derived by placing the primitives $p_i$ and $q_j$ into correspondence:

$$\text{DIST}_h(P, Q) = \sum_{(i,j) \in A} \sum_{a \in A} \log \left[ \frac{1}{p[a(q_j)|a(p_i)]} \right]$$

where $a$ denotes an attribute from the set $A$, and $p[a]$ is the conditional probability that for a certain attribute $a$, the value taken on by the corresponding right image primitive $q_j$ is $a(q_j)$ given that the left image primitive has a value $a(p_i)$.

The total (in)consistency between the relation sets $R$ and $S$ under the mapping $h$ is the summation of the conditional information we receive upon being told that tuple $S_{k_j}$ corresponds to tuple $R_{k_j}$:

$$\text{INC}_h(R, S) = \sum_{k} \sum_{(i,j) \in R_{k_j} \circ h} \log \left[ \frac{1}{p[fS_k(S_{k_j})|fR_k(R_{k_j})]} \right]$$

where $fR_k(R_{k_j})$ is the parameter of $R_{k_j}$ in $k^{th}$ relation in $P$-space, $fS_k(S_{k_j})$ is the parameter of $S_{k_j}$ in $k^{th}$ relation in $Q$-space, $p[a]$ is the conditional probability that the tuple $S_{k_j}$ takes the parameter value $f_k(S_{k_j})$ given that the parameter value taken by $R_{k_j}$ is $f_k(R_{k_j})$. The notation $(i, j) \in R_k \circ h$ means that the composition of the $i^{th}$ source tuple with mapping function $h$ yields an image tuple identical to the $j^{th}$ destination tuple. The lower the value of $\text{INC}_h(R, S)$, the smaller the relational inconsistency (or the larger the consistency) between the two images for a given mapping $h$.

The overall structural distance measure is defined as:

$$\text{INC}_h(R, S) + \text{DIST}_h(P, Q) = \rho \times M_h(D_L, D_R)$$

where $\rho$ is the cardinality of the mapping function. Since we seek the best match, the desired mapping function $h_0$ is that which minimizes $M_h$:

$$M_h(D_L, D_R) = \min h \in \text{pool}(p_i)$$

where $p_i$ is the nil possibility is also added to the pool if the best fitting potential correspondent exhibits an interprimitive distance exceeding a nilmapping threshold. This allows a relationally consistent mapping to be obtained when the correct correspondent does not exist in the matchpool due to processing errors, occlusion, or other effects. In anticipation of the order of instantiation in the search that is to follow, the entries in each pool($p_i$) are sorted according to their interprimitive distance values. Finally, a backtracking tree search is undertaken to recover the numerically best interprimitive mapping function. Each level of the tree corresponds to the instantiation of one left image primitive. In order to keep the tree size as small as possible, the primitives are instantiated in a manner such that the one at the top-most level has the smallest matchpool size, and the pool size increases for the primitives corresponding to successive levels. The tree is traversed depth first by tentatively pairing a right image primitive out of the matchpool of the left image primitive at that level. A relational consistency check is made after each pairing to ensure that $\text{INC}_h(R, S)$ stays finite.

The backtracking tree search approach to the construction of the inter-primitive mapping function is quite adequate in dealing with images with few primitives: typically $20$ to $30$. However, when confronted with images of detail rich scenes, each having hundreds of primitives, it is unable to arrive at even a single
mapping function (corresponding to one complete descent down the tree) without enormous computational burden since, early in the process, thrashing sets in.

A second drawback relates to the initial process of computing a matchpool for each left image primitive, before the tree search begins. Attributes of every right image primitive have to be compared to those of the left image primitive in order to construct the latter's matchpool. Assuming about \( N \) primitives in both the left and right images, building all \( N \) matchpools requires \( O(N^2) \) comparisons. For large images this represents a significant computational expense. We remind the reader that since we lack the luxury of camera pose information, epipolar constraints cannot be invoked to limit the search area for matchpool computation.

III The New Matcher

A. Motivation

A significant feature of the previous matching scheme is that the relational consistency is used only as a means of (eventually) inhibiting the growth of an inconsistent mapping function. There is no provision to actively use \( R \), the relations among primitives, in seeking matches that have a high chance of being correct. Only the information in the attributes of the primitives is used for this purpose since that completely determines both (a) the contents of the matchpools and (b) the order in which the primitives are instantiated. Seen in the context of the cost function which the structural matcher seeks to minimize, \( \text{INC}_h(R, S) + \text{DIST}_h(P, Q) \), this step is essentially an attempt to keep the change (always an increase) in the \( \text{DIST}_h(P, Q) \) term to a minimum. No particular concern is shown for the magnitude of the change in \( \text{INC}_h(R, S) \) as long as it is not infinity, i.e., as long as the new pairing is not irreconcilably inconsistent with the previously matched primitives. This approach is the most natural path to take if the primitives are so rich in information content that the values of their attributes afford a sufficiently large amount of discriminatory power. But as the number of primitives gets large, less successful, in terms of time-efficiency, this reliance on attributes becomes. So it seems logical to investigate whether a matching scheme that leans on minimizing the \( \text{INC}_h(R, S) \) term can be devised, and if so, whether it would be faster.

With this in mind, we turn attention to the fact that in unconstrained domains, matching extended primitives must necessarily replace point-based matching. So, ideally, we would like to partition the edge image into a few, and hence long, primitives so that the combinatorics of the matching are relatively benign. The drawback with using arbitrarily long edge contours as primitives for matching is that the difference in vantage points of the two cameras makes it very unlikely that the shape of such contours would be preserved exactly from image to image. Therefore the contours are partitioned into smaller fragments, which in our case, are segments of approximately constant curvature. Correspondences can then be established for at least a subset of these primitives because primitives resulting from the partitioning of a single contour in the left image are not restricted to finding homologous primitives that are all derived by the partitioning of a single right image contour.

Having said this, we would now like to backtrack and say that in a wide variety of images, a large fraction of the contiguous primitives in the left image have homologues which continue to maintain contiguity. By contiguous primitives, we mean those primitives that adjoin end-to-end because they were extracted from the same edge-contour. Using contiguity to guide the search for correspondents therefore holds out a distinct promise of success for the task of minimizing increments in the inconsistency term in structural matching. Therefore at any given stage of the mapping process, we should instantiate that left image primitive which is literally adjacent (along the edge contour) to the primitive that was paired up in the immediately preceding step. We must emphasize that we use this relation not as a constraint to be satisfied by primitives in getting mapped, but only as a pointer to locations in the image where there is a high likelihood of finding matches in whose correctness we can place a very high confidence. Since there is now a special spatial relation between successively instantiated left image primitives, we can expect the increments in the inconsistency term to be small. Furthermore, as the number of primitives that chain together in mapping onto contiguous primitives in the other image increases, our confidence in this set of matches increases at least proportionately.

B. The Matching Algorithm

We extract intensity edges from the stereo pair of gray level images by applying the LoG. The edge contours are partitioned into segments of approximately constant curvature (extracted according to the algorithm in [9]) and are used as primitives to build the parametric structural descriptions. The primitives re-
resulting from the decomposition of distinct contours are kept in distinct lists. Each such list is referred to as a section of (contiguous) primitives. When a subsection of left image primitives maps onto a subsection of right image primitives, the resulting contiguity-preserving matches would be said to constitute a mapsection. The algorithm divides naturally into three parts: Extract some reliable matches, establish constraints on the search space by requiring subsequent potential matches to be consistent with these, and finally carry-out well-constrained mapping along with correction of blunders.

1. Initial Matches

Our aim, at first, is to extract a few very reliable matches. As hinted before, when matches constitute a mapsection, the chances of their being blunders decreases as the cardinality of the mapsection increases. (In practice we have seldom seen blunders when the cardinality is 3 and almost never when it is 4 or more.) Hence we first concentrate on finding mapsections of cardinality greater than some number \( \mu \). To do so with minimum effort, the largest (in terms of number of constituents) left image section, say \( lsec \), is picked. Starting from one end of \( lsec \), for every \( \mu \)th primitive, a matchpool is built by exhaustive comparison with all right image primitives. We therefore have a long list of tentative starting (or seed) matches of the form \((pseed, qseed)\) where \( pseed \) ranges over all those members of \( lsec \) for which matchpools have been built and \( qseed \) ranges over all members of each \( pseed \)’s matchpool. Next, for all these pairs, we try to extend the mapping \((pseed, qseed)\) to other \( p_i \in lsec \) and other \( q_j \in rsec \) (where \( rsec \) is the right image section to which \( qseed \) belongs) to see if a mapsection could result.

Starting from \( pseed \), its adjoining neighbor on one side, say \( pseed-1 \), is considered. Its matchpool, \( pool(pseed-1) \), is checked to see if one of the adjoining neighbors of \( qseed \) is present. An affirmative answer is followed by a limited relational consistency check of this match with the previously made matches (which at this stage, is just \((pseed, qseed)\)). The relational checks are the same as those made in the old structural matcher. If the pairing is declared consistent, we move on in the same direction within both \( lsec \) and \( rsec \) and next consider \( pseed-2 \). This process of walking and mapping along \( lsec \) and \( rsec \) is continued until one of three things happen: (a) We run off the end of \( lsec \) and/or \( rsec \). (b) The \( rsec \) member under current consideration is not in the matchpool of the current \( lsec \) member. (c) The relational consistency check declares the current (tentative) pairing to be inconsistent. After reaching the end on one side of \( pseed \), we reverse directions and walk along the section starting from \( pseed+1 \). In this way, the initial seed pairing is grown to the fullest extent in each direction.

These steps are applied to the entire list of seed pairs. Since the extraction of these mappings is guided by moving along a section, the process gets carried out very rapidly. A number of large and small mapsections result, all of them internally consistent, but some constituted of spurious matches (especially those with cardinality 2). Usually, at least one mapsection with cardinality \( \geq \mu \) gets extracted in the above steps.

We keep repeating the entire procedure above, choosing the largest left image section as \( lsec \) at each stage, until \( \nu \) mapsections of cardinality \( \geq \mu \) are obtained. We have chosen \( \mu = 4 \) and \( \nu = 5 \) in our implementations. Artificially contrived scenes with precisely repetitive structure could possibly lead to a large yet spurious mapsection. To avoid this pitfall, we carry out relational consistency checks between the \( \nu \) large mapsections, discarding the one(s) that are not compatible with the majority. The matches making up this majority of large sections are our initial reliable matches.

2. Establishing constraints on the search-space

We have, at this stage, a set of matches that we have very high confidence in, leading us to make the assumption that these matches would never get backtracked over. We can therefore restrict the entry of right image primitives into the matchpools of the remaining left image primitives by requiring aspirants to satisfy the (inexact) relational consistency checks at the very outset. The two parametric relations measured between primitives are pairwise separation of the centroids and pairwise orientation of the line joining the centroids of the two primitives, where the associated parameters are the numerical values of the separation and the angle, respectively. Without entering into the details of estimating the transition probabilities from the left to right images for the parameters, it suffices for present purposes to consider just the maximum permissible distortions, \( \Delta \theta \) and \( \Delta \rho \), on angle and separation parameters, respectively. For a right image primitive \( q_j \) to be considered for attribute checking with a left image primitive \( p_i \), the tentative pairing must be consistent with the initial reliable matches.

The values of \( \Delta \theta \) and \( \Delta \rho \) are used to predict a window in the right image in which the centroid of \( q_j \) must lie.
To illustrate this, we select any one of the initial few, but reliable, matches, say \((p_a, q_b)\) as shown in Fig. 1. The centroid locations of the primitives are denoted by the points \(a\) and \(b\) respectively. An unmapped primitive \(p_i \in P\) in the left image, with centroid denoted by \(i\), will have coordinates \((\rho_p, \theta_p)\), in a polar coordinate system centered at \(a\). Clearly, if we desire consistency of the pairing of the points \(a\) and \(b\), then \((p_a, q_b)\) with the match \((p_a, q_b)\), then the centroid of \(y_j\) must have coordinates close to \((\rho_p, \theta_p)\) in a coordinate system centered at \(b\). The discrepancy we are willing to tolerate is dictated by \(\Delta \rho\) and \(\Delta \theta\). Thus, a window is constructed in the right image for each left image primitive, with polar dimensions \(2\Delta \rho\) and \(2\Delta \theta\), centered at \((\rho_p, \theta_p)\), as illustrated in Fig. 1. Only those primitives that lie in this window and, furthermore, are consistent with the other initial matches, qualify for consideration in building the matchpool, \(pool(p_i)\). Thus in spite of the lack of epipolar constraints, we achieve a very effective confinement of the regions in the right image over which matchpool computations must occur.

3. Well-constrained matching

After constraining the search space for correspondents, the mapping of the remaining primitives is very fast. First, the contiguity preserving map-sections are sought. This is achieved by walking and mapping along left and right sections in a manner similar to that followed in extracting the initial matches. The map-sections that are of cardinality \(\geq 3\), and are relationally consistent with the earlier map-sections are accepted first, followed by those of cardinality \(\leq 2\).

After garnering matches having a high degree of reliability, the consistency check mechanism, which by nature is such that it grows more effective as the matches accumulate, is at a stage where it rules out almost immediately any pairing that is not of true homologues. Therefore at the last stage, when the only left image primitives remaining to be matched are those that have failed to become part of a map-section, simply picking that right image primitive out of the matchpool which leads to the smallest increment in the cost function, \(INC_a(R, S) + Dist_b(P, Q)\), leads to the correct result i.e. for all practical purposes, a hill climbing search at this juncture is just as effective for extending the interprimitive mapping function as the backtracking search of the original matcher. We contend that any spurious match that might have occurred in earlier stages can be detected by keeping track of tentative pairings that are unacceptable due to relational inconsistency with just one of the earlier matches (and consistent with all others). If a sufficiently large number (3, in our implementations) of such pairings are recorded, the grievance causing earlier match is removed and the pairings are accepted as matches. Thus we arrive at a complete interprimitive mapping function with almost no thrashing.

IV Results and Conclusions

We demonstrate the performance of the algorithm on two stereo pairs from two different domains. One is a pair of aerial images (241 \(\times\) 241 pixels) and the other is of an indoor scene (256 \(\times\) 256 pixels). The indoor-scene images, originally shot at the University of Massachusetts, have been obtained from the vision-list-archives.

Both stereo pairs were processed using no knowledge of the exact camera geometry. We applied the LoG with \(\sigma = 5\) and detected the zero crossing contours. Whole contours with shallow average zero crossing slope (< 0.65) were deleted. The remaining contours were partitioned into constant curvature segments according to the algorithm in [5]. Contour segments with average zero crossing slopes less than 1.1 were then deleted. The remaining segments, our primitives for matching, were described by the attributes: centroid, edge-length, mean-orientation, eight-ratio (which is length divided by the eight-connected distance between endpoints; basically a measure of straightness), mean absolute slope (of the zero crossings), and plus ratio (ratio of number of positive tangential-slope values to length). The relations used were pairwise-orientation and pairwise-separation. The conditional probability density estimators of the attributes and the relational parameters were constant-slope triangular functions with the following widths of the triangle bases: Length: 5 times the square root of the (left) contour segment length. Eight-Ratio: 0.8, a constant 0.4 to either side of the left primitive value. Mean-Orientation: 80°, 40° to either side of the left primitive value. Mean-Slope: 0.5 times the left primitive’s average slope, 0.25 to either side. Plus-Ratio: 0.5, a constant 0.25 to either side. Pairwise-Orientation: 60°, a constant 30° to either side. Therefore, \(\Delta \theta\) is chosen to be 30°. Pairwise-Separation: 0.15; for this relational parameter, distances between primitive centroids from 0 to 500 are mapped into the range [0,1]. Hence, \(\Delta \rho\) is chosen as 38 pixels.

Contiguity-exploiting matching is performed over the primitives, with the goal of constructing a 1:l mapping. Fig. 2 shows the performance of the algorithm.
on the two stereo-pairs. The detailed statistics of the
matching results for both pairs are shown in the table
below. The time performance results are for execution
on a TI Explorer II+ LISP machine.

The situations where not backtracking over matches
could cause blunders are those where the scene has
a number of repeated structures. The indoor images
are of a contrived scene, replete with rectangular floor
mats and wall hangings. Our results show that even
under artificially created adverse conditions, the algo-
rithm works well. The few erroneous matches were of
primitives that were on the borders of the images, and
hence lacked an adequate number of relational consis-
tency checks to steer them clear of the wrong match.
We must mention that the process of consistent de-
composition of the edge contours to create the prim-
itives in the two images of the stereopair is crucial,
since its performance sets a hard limit on the number
of extended primitives that can be matched.

The purpose of structural matching of extended
primitives is not to deliver a depth map by itself, but
to transform an underconstrained situation into a well
constrained one. Then a process like Dynamic Edge
Warping, as described in [6], takes over and provides
pointwise matching along the homologous contour seg-
ments and autonomous calculation of the epipolar condi-
tion from those point matches. From this new-
found knowledge of the epipolarity condition, we may
proceed to apply any of the existing approaches to
stereo which require this knowledge, or otherwise it-
ervatively refine the resulting stereo model. The most
effective means of refining the scene depth map is an
open question at this point, and we continue to pursue
it actively.

## References

[1] K. L. Boyer and A. C. Kak, “Structural stere-
opsis for 3-d vision,” *IEEE Transactions on Pat-
tern Analysis and Machine Intelligence*, vol. 10,

- a review,” *IEEE Transactions on Systems, Man,

tion of stereo images by matching graph descrip-
tions of edge segments,” *International Journal of

through feature grouping and maximal cliques,”
*IEEE Transactions on Pattern Analysis and Ma-
chine Intelligence*, vol. 11, pp. 1168–1180, Nov.
1989.

tour decomposition using a constant curvature cri-
terion,” *IEEE Transactions on Pattern Analysis
and Machine Intelligence*, vol. 13, pp. 41–51, Jan.

namic edge warping: An experimental system for
recovering disparity maps in weakly constrained
systems,” *IEEE Transactions on Systems, Man,

<table>
<thead>
<tr>
<th>Observed Quantity</th>
<th>Aerial Scene</th>
<th>Indoor Scene</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of left image primitives</td>
<td>303</td>
<td>260</td>
</tr>
<tr>
<td>Number of right image primitives</td>
<td>335</td>
<td>274</td>
</tr>
</tbody>
</table>
| Total number of correctly matched primi-
  tives                                  | 111          | 134          |
| Number of correct contiguity preserving
  matches                                | 30           | 101          |
| Number of other correct matches          | 18           | 33           |
| Number of erroneous matches              | 3            | 3            |
| Time taken, Initial matches              | 64 s         | 43 s         |
| Time taken, Rest of contiguity preserving
  matches                                | 45 s         | 46 s         |
| Time taken, Other matches                | 35 s         | 42 s         |
Figure 1: Search Window for homologue of $p_1$

Figure 2: The gray level images, the edge contours, and the matched segments.