Model-Guided Grouping for 3-D Motion Tracking

Xun Li **David G. Lowe**

Technical Report 91-22
October 1991

Department of Computer Science
The University of British Columbia
Vancouver, B.C. Canada V6T 1Z2
xli@cs.ubc.ca
lowe@cs.ubc.ca

Model-Guided Grouping for 3-D Motion Tracking*

Xun Li **David G. Lowe**
Computer Science Department
The University of British Columbia
Vancouver, B.C. V6T 1Z2, Canada

ABSTRACT

The objective of this paper is to develop a robust solution to the correspondence problem in model-based motion tracking even when the frame-to-frame motion is relatively fast. A new approach called Model-Guided Grouping, which is used to derive intermediate-level structures as our matching tokens, is introduced. The groupings are guided and derived locally, with the contemporary use of model structures, around the predicted model during the object tracking. We choose junctions and parallel pairs as our matching tokens, thus the information coded in these structures is relatively invariant in consecutive frames. The matching strategy is coarse-to-fine, and partial matching will also be allowed when occlusions are present. The method for evaluation of probability of accidental match based on junction groupings will be discussed. Systematic testing shows that matches based on these new methods improve correspondence reliability by about an order of magnitude over previous methods based on matching individual line segments.

*Xun Li was supported by the Institute for Robotics and Intelligent Systems under the Networks of Center of Excellence program. David Lowe was supported by the Canadian Institute for Advanced Reseach and by NSERC grant OGP-36807.

1 Introduction

A model-based motion tracking system consists of three major parts: segmentation, correspondence and viewpoint verification. Most previous approaches to solving the correspondence problem have used lower-level primitives such as points [17] and lines [19, 8] as their matching tokens. In this paper, we propose a new approach called Model-Guided Grouping, which is used to derive intermediate-level structures as our matching tokens. The term Model-Guided comes from the fact that the groupings are guided and derived locally, with the contemporary use of model structures, around the predicted model during the object tracking. We choose junctions and parallel pairs as our matching tokens, thus the information coded in these structures is relatively invariant in consecutive frames.

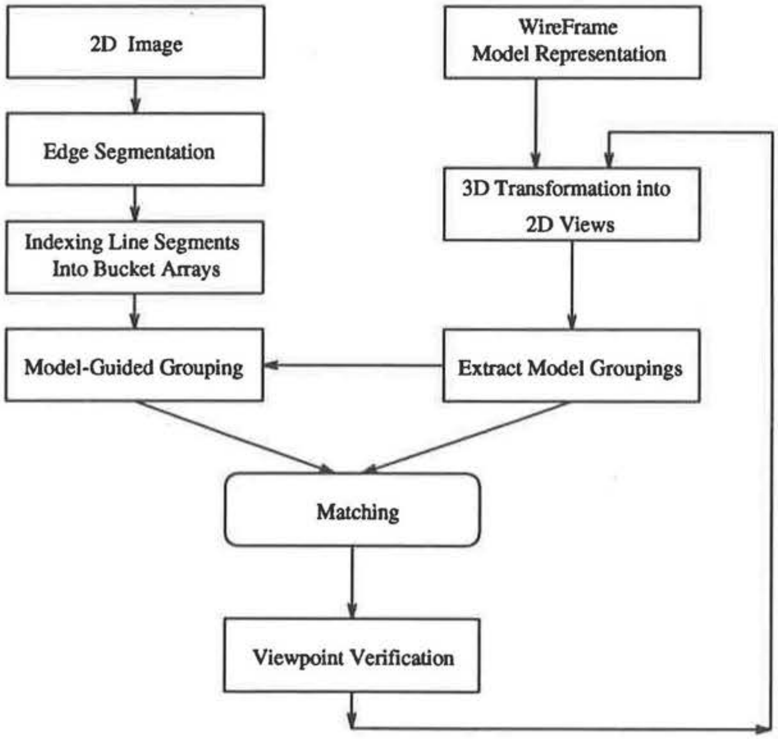
2D Image
WireFrame Model Representation
Edge Segmentation
3D Transformation into 2D Views
Indexing Line Segments Into Bucket Arrays
Model-Guided Grouping
Extract Model Groupings
Matching
Viewpoint Verification

Figure 1: The framework of our motion tracking system. Model-Guided Groupings are incorporated as our matching tokens.

Once the matching tokens have been derived, the problem to be considered is the pairing by the correspondence process of tokens in one frame with tokens in the subsequent frame. The key issue here is that the correspondence between isolated token pairs is governed by a certain *proximity* and *similarity* metric, termed *Correspondence Strength* [18]. We will discuss how to formulate the measure of correspondence strength based on model-guided groupings.

Correspondence is performed locally, with a coarse-to-fine strategy. Partial matching will also be allowed when occlusions are present or features fail to be detected for some other reason. Global matching is examined by viewpoint consistency, which is based on the Lowe's iterative method [7].

Comparisons are made between the current system and Lowe's previous one that used only line-to-line matching. The results show that with a slight increase in computational cost, the use of the higher-level groupings as matching tokens leads to a much more robust model-based motion tracking system.

1.1 Motivation

The robustness of the correspondence process is essential in any motion tracking system. Previous correspondence algorithms using line segments as the matching token are robust only when motion is very slow. If motion is relatively fast, the system fails due to an inevitable problem: selection from multiple matches – a token being tracked may find multiple matches in the current frame around the predicted region, especially when there are several scene tokens which are close to each other.

Two important issues concerning motion correspondence algorithms will be discussed:
1. What kind of tokens will be used in our matching process? How to derive these tokens?
2. How should correspondence strength of these tokens be measured?

In recent years, the use of perceptual organization has drawn great attention in the machine vision community. However, most aspects of perceptual organization remain untried. Inspired by the human capability of visual perception, our approach is to incorporate perceptual groupings – intermediate-level structures as the matching tokens. Unlike most perceptual grouping processes, model-guided grouping is used to derive perceptual groupings with the contemporary use of the model structures; therefore it can be computed much more efficiently. The correspondence is performed between intermediate-level structures from model and image pairings. Figure 1 illustrates the framework of our matching system.

2 Related Work

An early system for model-based motion tracking was studied by Gennery [2]. He combined a prediction and a matching process to adjust parameters. The prediction of the object position and orientation for the current frame is taken by extrapolating from the previously data. Matching is performed by searching nearest image edges to the predicted model line, if it is within five pixels. He also proposed a more elaborate method which varies the extent of searching according to the accuracy of the predicted data.

Two correspondence algorithms are described in Verghese's [17] work: hypothesize-and-test and dynamic edge tracking. From a set of slightly different viewpoints, the hypothesize-and-test algorithm uses the previous camera parameters to hypothesize model projections which are cross-correlated with edges detected in the latest image. The results of cross-correlation will be used to select the best match and to update the camera parameters. The second algorithm is divided into two problems: dynamic edge tracking and model-based segment tracking. Dynamic edge tracking is performed in a 3×3 neighbor around each model edge point; model-based segment tracking uses an adaptation of Lowe's viewpoint solution algorithm. They assume that the motion is very slow – the average motion displacement is 1 pixel per frame.

The correspondence algorithm in Lowe's [8] tracking system works through the following two steps: with projected model edges predicted from previous viewpoint parameters, he searches compatible edges around each projected model edge. Compatibility is defined in terms of location, length and orientation. Then the probabilities for accidental match will be measured and ranked for all compatible candidate edges associated with each model edge [6], and the lowest ranked ones are reported as the correct matches and will be used to update the viewpoint.

3 Model-Guided Grouping

A number of psychophysical studies concerning the detection, localization and tracking objects in the visual field have suggested a two-stage theory of human visual perception. The first stage is the "preattentive" mode [15], in which simple features are processed rapidly and in parallel over the entire visual field. In the second, "attentive" mode, a specialized processing focus, usually called focus of attention, is directed to particular locations in the visual field. Empirical and theoretical studies suggest that beyond a certain processing stage, the analysis of visual information proceeds in a sequence of operations, each one applied to a selected location.

The sequential application of operation to selected locations raises two central problems [4]. First, what are the operations that the visual system can apply to the selected locations? Second, how does the selection proceed? That is, what determines the next location to be processed.

Koch [4] suggested that the major rule for initial selection of a location is based on the conspicuity of that location, i.e. by how much its properties differ from the property of its neighborhood. We argue that perceptual groupings do preserve the properties different from its neighborhood.

Consider the second of questions in motion tracking, two rules for shifting from one selected location to another are based on (1) proximity preference and (2) similarity preference. Both mechanism are related to phenomena on perceptual grouping and "Gestalt effects" which occur as a function of object similarity and spatial proximity.

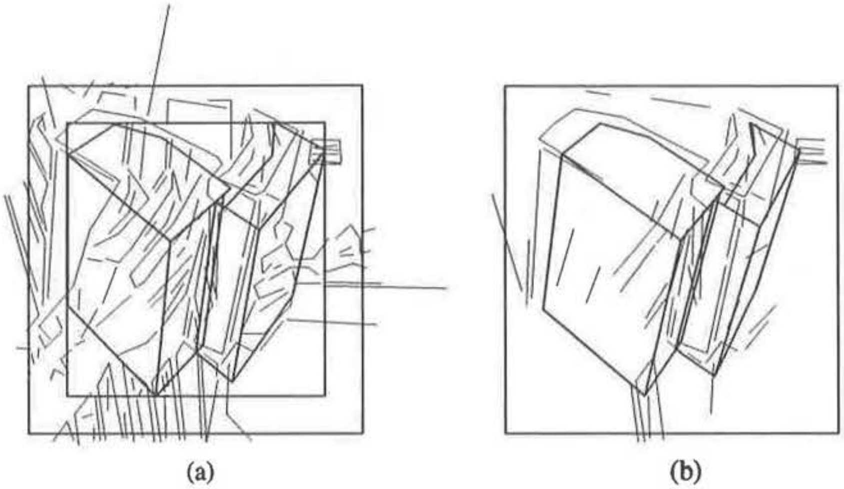

Figure 2: (a) *Local Focus Window* is an extended bound by superimposed model edges. (b) Compatible image edges will be derived after passing all the unary constraints.

3.1 Constraints Reduce the Search Space

From the computational point of view, the above two rules can be accomplished by imposing constraints, to simulate the focus of attention in the region of interest, and to perform the shifting rules based on proximity and similarity preference.

In order to focus the search space, all the grouping and matching processes should be focused within the *local focus window* – an extended boundary around superimposed model edges (Figure 2). The set of potentially matching image segments is obtained by applying constraints on the length, angle and distance of each image edge from the model prediction. The range of each constraint can be obtained from the covariance matrix of model parameters during motion tracking [8]. For the purpose of testing the role of grouping in the correspondence process, we have used fixed thresholds on each constraint as described later.

3.2 Indexing Line Segments

Attempts to reduce the computational complexity for detecting neighborhood relations between segments has been inspired by the methods in computational geometry – partition the whole image into buckets [1] or slabs [11]. In our implementation, we choose the former one. The overall image is partitioned into $m \times m$ squared windows, and to each window is attached a bucket - a list of segments according to their endpoints or middle points.

3.3 Model-Guided Grouping

Two line segments are identified as "potentially connected" if they satisfy the following constraints:

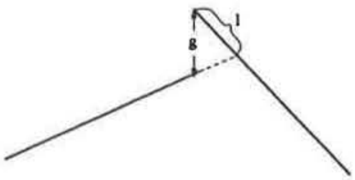

Figure 3: T junctions will be examined for potentially connected constraints.

1. their gap is small – less than a certain threshold (say 12 pixels),
2. they are not T junctions.

By T junction we mean that for two line segments the gap between their endpoints is less than a threshold and intersection point is inside one of the segment (Figure 3) with l longer than a certain threshold (say 5 pixels).

3.3.1 Model-Guided Junction Grouping

An ideal junction is a set of lines terminating at a common point. In real images however, the lines rarely terminate exactly at the same point due to noise or algorithms for constructing the line segments. Instead they terminate at a "small common region" (Figure 4). Hence the difficulty of detecting junctions is two fold [14]:

1. the complexity of considering all subsets of lines as potential junctions;
2. the lack of a simple and exact mathematical definition of a set of lines terminating at a common region.

We propose the concept of *pseudo-cotermination point* (**pcp**), which is defined as follows: Given a set of lines (more than two), the point which minimizes the sum of squares of perpendicular distance to each line is called the *pseudo-cotermination point*. A junction can be

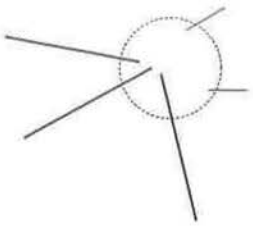

Figure 4: In the real image, lines are terminate at a small common region.

formed in the real image by judging the maximum distance from **pcp** to all the line segements considered [5].

Given a 3D polyhedral model, we present a method for extracting junctions which are compatible with the model junctions. Model junctions should be first extracted, and sorted according to the counter-clockwise orientation of their edges. With extracted model edges for each vertex, we first search all compatible edges associated with each model edge. By compatibility we mean that image edges should pass all the unary constraints. The search region is bounded by a circle with radius r centering around each model vertex.

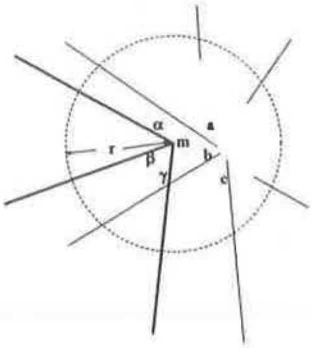

Figure 5: Grouping is focused around each model vertex (formed by thick lines).

Two directions can be followed to form the junctions from all combinations:

1. based on **pcp**: compute the **pcps** for all the junction combinations;
2. based on endpoint proximity: examine all adjacent pairs of potentially connected constraints.

Our implementation follows the later one, and test results give satisfactory formation and efficiency.

3.3.2 Model-Guided Parallel Grouping

Model-guided parallel grouping is performed in a similar way to the model-guided junction grouping. Salient parallel model pairs should be indexed first. This is done by checking all the segment pairs in each polygon. Then we will search all the compatible image edges associated with each salient parallel model edge. Salient parallel grouping is performed between compatible image edges associated with salient parallel model edges.

3.3.3 Model-Guided Collinearity Grouping

If two line segments are both compatible with the same model edge, they can be merged into one segment if they are collinear.

4 Matching with Model-Guided Groupings

Lowe's line-to-line matching system will be reviewed, then we explain how to extend the matching problem from line tokens to grouping tokens.

4.1 Probability Ranking of Accidental Match for Single Line Segment

All potentially matched image segments, which satisfy three unary constrains, will be identified locally around the predicted model edge position. Under the assumption that incorrect matches will be uniformly distributed in terms of position, orientation and scale, Lowe [7] formulates the expected number of lines within the given separation and angular difference:

$$E = \frac{4D\theta s}{\pi m} \tag{1}$$

Here, θ is the orientation difference, m is the length of the matched segment, D is a measure of the background desity of similar features, and s is the perpendicular distance from the center of the matched image segment to the predicted segment.

To verify a predicted model feature, he ranks the correspondence strengths – the inverse probabilities of accidental agreement for all potentially matched image segments:

$$p = \frac{E}{1+E} \tag{2}$$

4.2 Evaluation of Probability of Accidental Match with Model-Guided Junction Groupings

The goal of a vision system is to *minimize* the probability of making *incorrect* interpretations. All the matching algorithms are facing ambiguities from the input low-level features, due to noise from camera input as well as the imperfect output from the lower-level segmentation.

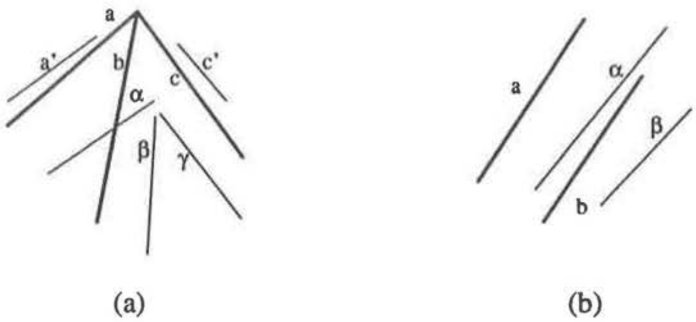

Figure 6: Improper matches arise due to spurious data or inconsistent local structures. Thick lines are predicted model edges, thin lines are image edges. Ambiguities can be reduced by using groupings. (a) Correct matches should be between a, b, c and α, β, γ, instead of local best matches between a, c and a', c'. (b) Correct matches should be between a, b and α, β instead of local best match between b and α.

Thus model-guided grouping can reduce the ambiguities by making full use of local consistent interpretation from image data.

Taking model-guided junctions as our matching tokens, our evaluation of the probability of accidental match makes use of the following considerations:

1. The probability should reflect the proximity and similarity rules.

2. The probability of accidental agreement should decrease when the number of compatible constituents in a junction increases. For example: suppose that a junction of a model consists of three edges. Then, a detected image junction compatible with three line segments would have a lower probability of accidental match than a detected image junction compatible with two line segments in a model junction.

3. When occlusion is present, partial matching among grouped junction segments should be allowed. Thus the evaluation should be flexible with respect to the number of the segments. For example: suppose that model junction consists of three edges, the maximum number of compatible image junction is two, this match should also be accepted.

4. If no junction groupings are present, this system should work at least as well as Lowe's line-to-line matching.

Following examination of these principles, we evaluate the accidental probability of the grouped feature as the product of each individual line-to-line accidental matching (2). Suppose that the number of compatible image segments is m with n model junction edges $(1 \leq m \leq n)$, and the probability of these m compatible image edges which can form a junction is $p(J)$. Each individual probability of accidental match is p_i. Making use of an assumption of conditional independence of the feature probabilities and Bayesian inference, the overall probability for accidental match, P(A), between an image junction with m segments and a model junction with n edges is:

$$p(A\&J) = p(A|J)p(J) = (p_1 \times p_2 \times \ldots \times p_m) \times p(J) \tag{3}$$

Since an image junction is derived with the known model structure, we assume that $p(J)$ is very close to 1. Thus equation 3 can be simplified as:

$$p(A\&J) = p_1 \times p_2 \times \ldots \times p_m \tag{4}$$

The inverse probabilities for accidental match will be ranked for all compatible candidate edges (both for groupings and single line segments), and duplicated ranked ones will be removed. Generally, the highest ranked ones are the complex junction groupings, while lowest ranked ones are single line segments. Top ranked ones are regarded as the matches and will be used to update the viewpoint.

4.3 Matching Strategy

Initial matching will be performed around each model vertex where the area for searching compatible candidate tokens is bounded by the circle (Figure 7-c) with radius of r. Top matches will be reported to update the viewpoint. Next viewpoint estimation will be performed, and the model structure will be updated (Figure 7-d). During this iteration, searching for compatible candidate tokens is bounded by a shrinking circle with radius of r' ($r' < r$). Matching refinement will be repeated until the agreement between model structures and image edges is close enough (Figure 7-f).

5 Implementation Results

Testing was performed on a SUN SPARC II among three images which were captured from different viewpoints. Comparisons were made with Lowe's line-to-line matching.

All the edges in the three images are extracted by marking zero-crossings [10], connected edges that are then linked together. Recursive contour splitting is applied for all the linked edges [3]. Finally, accurate line segments are formed by least-square-fitting [5] (Figure 7).

5.1 Model-guided Grouping

The projected positions of model edges can be computed directly from the known viewpoint. The model junction associated with each model vertex is extracted. During each iteration, viewpoint will be updated and the information of each model junction is obtained from the junction database, and are used to guide the grouping for image segments.

Images	Test Cases	Translation (169)	% Error
Image 1	Junction Groupings	169	0.0
	Line-to-Line	112	33.7
Image 2	Junction Groupings	164	3.0
	Line-to-Line	89	47.3
Image 3	Junction Groupings	168	0.6
	Line-to-Line	105	37.9

Table 1: Comparisons of error rates for matching based on junction groupings versus the previous line-to-line matching. The error rate is seen to greatly decrease with the use of junction groupings. This table reports results for only translational motion in 3-D.

Similar to the model-guided junction grouping, salient model parallel pairs will be formed into a database. Then they will be used to find the image parallel pairs around the predicted model position during each matching refinement.

5.2 Matching with Junction Grouping – Comparisons with Lowe's Line-to-Line Matching

The computation time for model-guided junction matching was measured by running 50 iterations. Roughly speaking, the average time for the use of model-guided junction groupings is slightly more than double the time for the use of single line segment.

The robustness of the system is tested in .the following two cases: 1) translation only, 2) combination of translation and rotation. With the known viewpoint for each image; we translate and rotate the model to a new position and determine the number of errors in matching from this position. The results are reported in Tables 1 and 2. For these tests, the thresholds for search range of line matching were set to an angle range of 30 degrees, and a length ratio in the range [0.25, 1.25].

The implementation results show that among 169 translational motions, almost all correspondences are correctly established by using junction groupings. If the translation amount is not large enough to move the junctions out of the searching region, all the matching can be correctly established with the use of model-guided junction groupings. We found however, based on "best" match selection, line-to-line matching failed due to inconsistency with local structures.

The tests for both translation and rotation are measured in two steps. We first translate the model in four directions to the adjacent 169 positions, at each position, we rotate the model in 8 different directions, up 15 degrees. The success rates with the use of groupings for all three images, are all over 94 percent. Failure has been identified if the junctions in the image fall

Images	Test Cases	Translation+Rotation (1352)	% Error
Image 1	Junction Groupings	1309	3.2
	Line-to-Line	758	43.9
Image 2	Junction Groupings	1275	5.7
	Line-to-Line	695	48.6
Image 3	Junction Groupings	1298	4.0
	Line-to-Line	712	47.3

Table 2: Comparisons between junction-based and line-to-line matching. The model has been displaced by both 3-D translation and rotation.

outside of the searching region.

6 Conclusions and Future Directions

The goal of our system is to minimize the probability of making incorrect interpretations. The use of model-guided groupings that we have presented assigns a central role to perceptual grouping as a way of reducing the ambiguities during the matching process in motion tracking. Most incorrect local matching, as shown from Lowe's line-to-line matching, can be eliminated through the use of the inherent contextual information formed from these groupings.

6.1 Conclusions

As the results show in section 5.2, we found that the use of junction grouping greatly improves the system performance. We also found that the model-guided junction groupings with three line segments are seldomly incorrectly matched, although the "junctions" with two line segments will more often find incorrect matching candidates. However, they could be eliminated through the ranking system. Since the final set of correspondences will be greatly overdetermined, the correct interpretation can still be reliably made even in the presence of some missing features (due to occlusion).

Lowe's line-to-line matching can successfully track slow motion (however, it depends heavily on the model structures from a particular viewpoint). In our implementation, larger frame-to-frame motion information, with a displacement up to 30 pixels, can be reliably recovered.

Since the metric of the evaluations of probabilities for different model-guided groupings are different, the optimal integration through the probability ranking is still unknown. As the authors noticed, the correct integration should not be performed from a simple weighting scheme. Instead, we should examine the local interpretation of matched parallel pairs.

For model-guided junction grouping, the running time for each iteration during the matching is only slightly more than double the time in comparison with line-to-line matching.

6.2 Future Directions

Current grouping is based on geometrical information only. We could group the features based on other information such as intensity or contrast across a line segment or gradient information. Model-guided grouping based on cotermination, parallelism and collinearity should be incorporated properly into one system in order to reduce all sources of ambiguities during the matching process. The current system is constrained by the use of line segments only. Provided the additional complexity of the segmentation is acceptable, the constraint can be removed by adding other features such as circular arc segments or general curve fragments.

References

[1] N. Ayache and B. Faverjon,"Efficient registration of stereo images by matching graph descriptions of edge segments," *Inter. Jour. of Computer Vision*, **1**, 2, 107-131, 1987.

[2] D. B. Gennery, "Tracking known three-dimensional objects," *Proceedings of the National Conference on Artificial Intelligence*, Pittsburgh, 13-17, 1982.

[3] S. L. Horowitz and T. Pavlidis, "Picture segmentation by a tree traversal algorithm," *Journal of ACM*, **23**, 368-388, 1976.

[4] C. Koch and S. Ullman, "Selecting one among the many: a simple network implementing shifts in selective visual attention," MIT AI Lab Memo 770, January, 1984.

[5] X. Li, *The Use of Model-Guided Grouping in Model-Based Motion Tracking*, Master Thesis, Computer Science Department, UBC, 1991.

[6] D. G. Lowe, *Perceptual Organization and Visual Recognition,* Kluwer Academic Publishers, 1985.

[7] D. G. Lowe, "Three-dimensional object recognition from single two-dimensional images," *Artificial Intelligence* **31**, 355-395, 1987.

[8] D. G. Lowe, "Integrated treatment of matching and measurement errors for robust model-based motion tracking," *Third International Conference on Computer Vision*, Osaka, Japan, 436-440, 1990.

[9] D. G. Lowe, "Fitting Parameterized Three-Dimensional Models to Images," *IEEE Transactions on PAMI*, **13**, 5, May, 1991.

[10] D. Marr and Hildreth, "Theory of edge detection," *Proc. Roy. Soc. London B*, **207**, 187-217, 1980.

[11] T. Matsuyama, H. Arita and M. Nagao, "Structural matching of line drawings using the geometric relationship between line segments," *CVGIP*, **27**, 177-194, 1984.

[12] R. Mohan and R. Nevatia, "Using perceptual organization to extract 3-D structures," *IEEE Transactions on PAMI*, **11**, 11, November, 1989.

[13] D.W. Murray, D.A. Castelow and B.F. Buxton, "From image sequences to recognized moving polyhedral objects," *Inter. Jour. of Computer Vision*, **3**, 1, 181-208, 1989.

[14] L. Quan and R. Mohr, "Matching Perspective Images Using Geometric Constraints and Perceptual Grouping," *Second International conference on Computer Vision*, Tampa, Florida, USA, 679-683, 1988.

[15] A. Treisman, "Preattentive processing in vision," *CVGIP* **31**, 156-177, 1985.

[16] S. Umeyama, T. Kasvand and M. Hospatal, "Recognition and positioning of three-dimensional object by combining matchings of primitive local patterns," *CVGIP*, **44**, 58-76, 1988.

[17] G. Verghese, C. R. Dyer, "Real time, model-based tracking of three-dimensional objects," TR-806, Computer Sciences Department, University of Wisconsin, November, 1988.

[18] S. Ullman, *The Interpretation of Visual Motion*, The MIT Press, 1979.

[19] Z. Zhang and O.D. Faugeras, "Tracking and grouping 3-D line segments," *Third International conference on Computer Vision*, Osaka, Japan, 577-580, 1990.

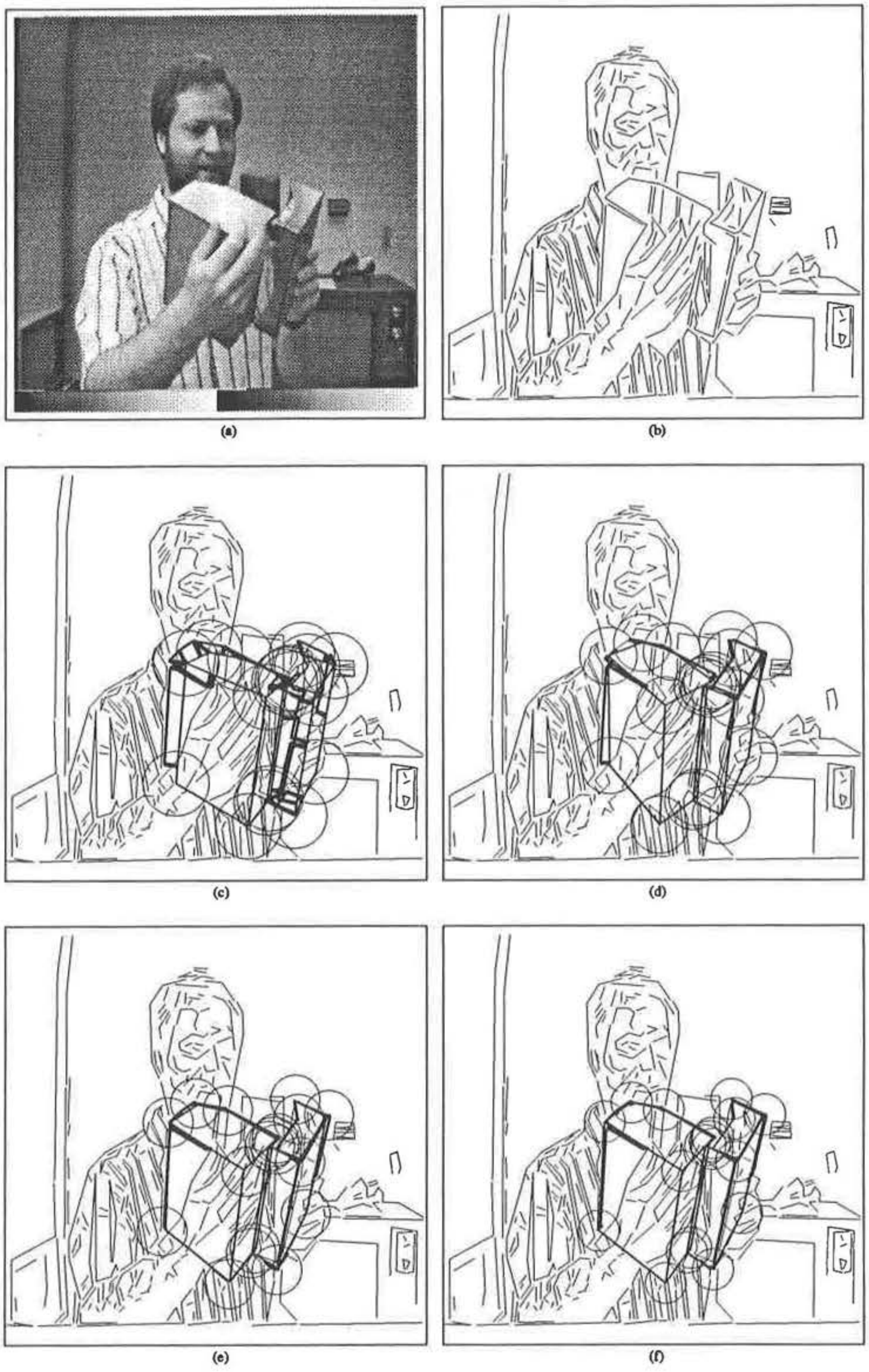

Figure 7: Matching refinement with model-guided junction groupings. The original image from a motion sequence (a). Line segments are formed in (b). Superimposed model edges are projected from its prior estimated viewpoint. Nearby are top ranked matching edges. Heavy bars are the perpendicular errors to be minimized. Searching regions (circles) will be reduced in consecutive iterations (c)-(f).