FEATURE GUIDED PIXEL MATCHING AND SEGMENTATION IN MOTION IMAGE SEQUENCES

Ram Charan and Narendra Ahuja

Beckman Institute for Advanced Science and Technology
University of Illinois at Urbana-Champaign, Urbana, Illinois, U.S.A. 61801

ABSTRACT

The problem of feature correspondences and trajectory finding for a long image sequence has received considerable attention. Most attempts involve small numbers of features and make restrictive assumptions such as the visibility of features in all the frames. In this paper, a coarse-to-fine algorithm is described to obtain pixel trajectories through the sequence and to segment into subsets corresponding to distinctly moving objects. The algorithm uses a coarse scale point feature detector to form a 3-D dot pattern in the spatio-temporal space. The trajectories are extracted as 3-D curves formed by the points using perceptual grouping. Increasingly dense correspondences are obtained iteratively from the sparse feature trajectories. At the finest level, matching of all pixels is done using intensity correlation and the finest boundaries of the moving objects are obtained.

Keywords: Motion Estimation, Motion Segmentation, Perceptual Grouping, Pixel Matching, Triangulation, Feature Matching

1. INTRODUCTION

This paper describes work aimed at interpretation of image sequences. The goal is to analyze the two-dimensional motion of objects in the image plane, with at most nearer-farther type understanding of their three-dimensional characteristics. Given an image sequence containing an arbitrary number of rigid objects in motion, the objectives are to identify feature points in the scene, obtain spatially dense trajectories of those points, segment moving objects, compute image flow at each pixel, and derive a qualitative description of the scene structure and dynamics from the image sequence. The qualitative description consists of scene characterization such as identifying frames when an object enters or exits the scene, detection of occlusion and occluding and occluded objects, detection of small objects moving at high speeds and counting the number of objects ever seen in the scene. Such interpretation of the image sequence is useful for a variety of applications such as traffic scene analysis, biological image analysis, aerial image understanding, in each of which the differences in the depths of moving objects are small. Next section summarizes some related previous work. Section 3-5 give details of our algorithm. Section 6 presents experimental results and section 7 presents concluding remarks.

2. PREVIOUS WORK

Several researchers have addressed the problem of feature correspondence in the past. Sethi and Jain[1] formulate this problem as an optimization problem and propose an iterative algorithm which they call the Greedy Exchange algorithm. Sethi et al.[2] propose a relaxation algorithm for feature point matching where the formation of smooth trajectories over space and time is favored. This method requires the correct initial correspondence and was used on very few feature points. Rangarajan and Shah [3] have proposed a noniterative polynomial time approximation algorithm by minimizing a proximal uniformity cost function. Cheng and Aggarwal[4] propose a two stage hybrid approach to the trajectory finding problem. The first stage extends the trajectories and the second one attempts to correct any errors. Debrunner[5] uses a two stage method to finding trajectories. The first step computes short feature paths of constant velocity. The second step deals with joining the feature paths into trajectories.

Grouping of image features is a central operation in the work done here. Point features are detected and the trajectories are formed by a grouping process. Gestalt psychologists were the first to study the grouping phenomenon in the human vision system [6, 7]. They proposed a set of criteria to form groups of image tokens [7]. These include: proximity, similarity, continuity and closure. They also studied their relative significance. Lowe [8] and Marr [9] have discussed roles of perceptual grouping in object recognition and early visual processing. Ahuja and Tuceryan [10] have developed a compu-

0-8186-7190-4/95 $4.00

tational approach to extracting perceptual structure in dot patterns. A similar approach is used in this work for perceptual grouping in 3D dot patterns.

3. SPARSE FEATURE POINT MATCHING AND SEGMENTATION

This section describes the first two steps of the algorithm to analyze a single batch of frames. The goal here is to find correspondences for sparsely placed feature points and segment the moving objects. The correspondences could also be used by any of a number of motion and structure algorithms [11, 12, 13] that require point feature correspondences. We have no knowledge of number of moving objects or their motions in the scene. The only assumptions made here are that they are rigid objects and are moving smoothly. A perceptual grouping technique is used to achieve this goal. Feature points are detected in each image. The images in a batch are stacked to form a 3D dot pattern. The batch size must be chosen carefully. If the number of frames in the batch is small, then the Voronoi tessellation may not represent the 3D structure well due to lack of data along time axis. A large batch size, on the other hand, may contain frames in which a moving object enters or exits the visual field which results in trajectories that last for only part of the batch. This makes their detection more difficult. Further it significantly adds to the computational load. Since in the algorithm used in this work, the batch analysis is used to estimate final correspondences for a central pair of frames, it is sufficient to ensure that the Voronoi structure associated with the dots in several central frames are correct, i.e., are not affected by lack of image frames. We have found that a batch of seven to nine frames suffices to yield valid Voronoi structure for the central frames. The batch size of seven frames is used in the implementation as shown in Fig.1.

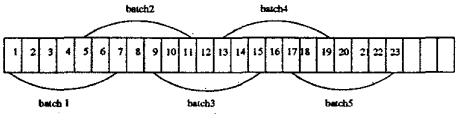

Figure 1: The image sequence is analyzed a batch at a time. Here successive overlapping batches of seven frames are shown.

3.1 Feature Matching using Perceptual Grouping: A feature correspondence represents how a physical point located in a given frame has moved to another location in the next frame. To find these correspondences using perceptual grouping, it is necessary to extract the perceptual segments of dots that group together because they belong to a perceived curve. A feature detector is used to obtain feature points from a set of image frames. These feature points are stacked to form a 3-D dot pattern. The Voronoi tessellation of the dot pattern is computed. This associates with each dot a polyhedral region, or neighborhood, that represents the dot's geometric environment. The geometric structure of the dot pattern is represented in terms of certain geometric properties of the Voronoi neighborhoods of the dots. Then each edge joining two neighboring dots in 3-D Voronoi tessellation is labeled as a *curve* or a *noncurve* edge by applying heuristic rules to the geometric properties of the polyhedron of each dot. The two vertices of an edge which is labeled as a curve, represent the correspondence or match of one vertex to another.

The geometric structure of the dot pattern is represented in terms of certain geometric properties of the Voronoi neighborhoods of the dots. We consider the neighborhood of a point P as the region enclosed by the Voronoi polyhedron containing P. This is an intuitively appealing approach because the local environment of a point in a given pattern is reflected in the geometrical characteristics of its Voronoi polyhedron. Most of the perceptually significant characteristics of a dot's environment are captured in the geometric properties of the Voronoi neighborhood. Such properties have to be specified to complete the representation of the geometric structure. They include: volume, eccentricity and elongation of the polyhedron of a dot, and relative distances between dots. The selection of these properties is based on intuition. The computation of these properties for a given dot pattern gives the basic data on which the procedures are applied that extract perceptual structure. Volume of a polyhedron is related to density of dots. The direction of eccentricity indicates the direction in which the density of dots increases. The elongation carries with it the direction of its major and minor axes.

3.2 Algorithm: The features detected in a batch of frames yield a 3D dot pattern when the frames are stacked together as spatio-temporal data. First, the Voronoi tessellation of the dot pattern is computed. Then the geometric properties of the Voronoi tessellation described above are computed. These parameters are then combined to obtain evidence in support of the different possible perceptual roles of Delaunay edges. We analyze the local geometric structure of the dot pattern, and compute a probability vector for each Delaunay edge such that this vector represents various perceptual roles based on local evidence. The computation of each vector is done using a probablistic relaxation labeling process which assigns the labels curve

and noncurve to Delaunay edges. The initial probabilities, compatibilities and updating of the above labels is done by using the various geometric properties of Voronoi cells and Delaunay edges. Such details of the relaxation formulation [14] will be skipped here for brevity.

3.3 Sparse image point correspondences: Let us consider a physical point in the scene which is visible in all the seven frames and whose corresponding feature point is also detected in all seven frames. The result is a curve segment joining the feature points in all frames. In general some feature points are missed and some new feature points appear from frame to frame. Therefore, the length of the curve segments or trajectories for the batch is between 1 to 6. For a fully connected trajectory the length is 6, and a segment between only two frames is of length 1. The feature point locations along a trajectory in frames 3 and 4 are considered as correspondences.

3.4 Segmentation of moving objects: Once the feature point correspondences are known, the feature points in each frame are segmented into different moving objects based on similarity of motion. Local adjacency among points is made explicit through the Delaunay triangulation whenever a Delaunay edge connects a point with its voronoi neighbors. Segmentation is then achieved by identifying Delaunay edges connecting points belonging to different objects as well as those inside a single object. In general two independently moving objects differ in the magnitude and direction of their 3D motion. However 2D direction alone is a strong basis to discern if two points belong to the same or different objects, and in fact is stronger cue for motion boundary perception in human vision. Edge identification is done by comparing the motion vectors at its two vertices. For an intra-object edge, the two end motion vectors are similar, i.e. the length and orientation of the vectors are approximately the same. For an inter-object edge the length and orientation of these two vectors are independent, and therefore different in general. A connected component algorithm is run to identify the sets of edges each comprising a different moving object. This also identifies the feature points belonging to each moving object.

4. DENSE CORRESPONDENCES

This section describes matching of finer level features with the help of coarsest level matches already identified through perceptual grouping as discussed in section 3. The finer level features are more densely distributed and therefore they improve the accuracy of detected moving object shapes relative to those segmented at the coarsest level. The sequence of coarse-to-fine matching steps described in the following subsections is iterated to the finest level of detected features, yielding the highest density of feature matches.

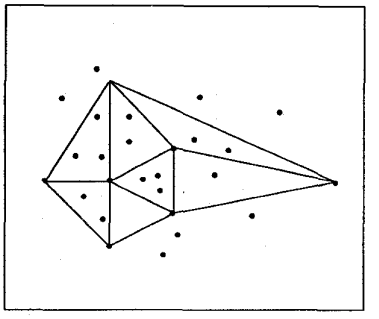

Figure 2: Vertices of the triangles are the matched coarse level feature points and the other dots are new finer level feature points.

4.1 Computing coarse estimates of fine matches: Consider the finer level features detected in a pair of frames. The motions of these denser features are predicted based on the known motions of nearby, coarser level features. The coarser level features near a detected fine level feature may belong to one or more differently moving objects. Therefore the detected feature may have any of these motions. Accordingly all different motions are considered and the one that gives the best intensity correlation with the next frame is selected. Specifically, the Delaunay triangulation defined by the coarse level features is superposed on the image containing the finer level features. Each detected feature belongs to a triangle of the above triangulation (Fig 2). The detected feature is assumed to have a motion which is the same as one of the motion values of the vertices of the triangle. The three vertices may belong to one, two or three diferent moving objects, thus having one, two or three different motions (Fig 3). Thus, these are the following three cases to consider:

- If all the vertices of a triangle are from the same object then the motion of the feature points inside the triangle is estimated from the motion of all three vertices, as weighted sum of the three vertex motions. In Figure 3 the triangle *abc* is of this type.
- For the triangle *bcd*, two of the vertices *b* and *c* are from one object and vertex *d* is from another object. A feature point *p* inside this triangle, may belong to either of the objects. Therefore, the point is assigned two different possible estimates corresponding to the two object motions, one of which must then be selected.

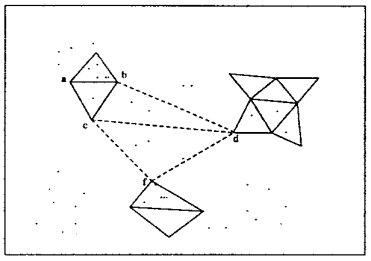

Figure 3: Triangle *abc* has all vertices from same object, triangle *bcd* has its vertices from two objects and triangle *cdf* has its vertices from three objects.

- Each vertex of the triangle *cdf* belongs to a different object. Therefore, a new feature point inside such a triangle is assigned all three vertex motions as estimates, one of which must then be selected

4.2 Fine match selection: For a new fine level feature, each of the available motion estimates predicts a different matching location in the next frame. Only one of which is correct and to be selected. Around each such predicted match, feature points are tested to identify those whose neighbors gray levels are well correlated with those of the fine feature points is being examined. Since the estimates derived from coarse level are more approximate due to lower feature density, the newly identified candidate matches serve as more accurate alternative motion estimates. Selection among these candidates is now performed by enforcing the spatial continuity of motion. In particular, this is achieved by employing the constraint that nearby features have similar motion directions. The relaxation algorithm is used to select the best candidate for each feature point. It may happen that two or more feature points may get matched to the same point in the next frame. Such matches are discarded and only those feature point pairs which are matched uniquely are considered for further processing.

4.3 Segmentation: The unique matched pairs selected for fine level feature points comprise denser correspondences than the coarse level correspondences inherited from the coarser level. These finer level correspondences can again be segmented into distinctly moving objects in the same way as done at the coarser level, namely, by grouping features having similar motion directions as discussed in the previous section. The resulting segmentation follows the object boundaries more accurately. These fine level features along with the segmentation are the final result of the iteration. The process reiterates starting with the first step described in Sec. 4.1 using the next finer level of features.

The coarse to fine motion estimation and segmentation is continued at increasingly fine spatial scales until the feature detector no longer gives useful new features.

5. PIXEL CORRESPONDENCES

Once the finest level features are found and matched, the matching of remaining pixels must use raw intensity information. This is done using intensity correlation as described below. The result is pixel level correspondence and segmentation.

5.1 Obtaining the pixel match: Consider a pair of frames (frame1 and frame2) after the finest level feature matching. The Delanauy triangulation is computed for the matched points in frame1. The 2-D motion of every pixel in this triangle is computed using a procedure similar to that described in the previous section for feature points. The three vertices of a triangle may belong to one, two or three different objects in the scene. Therefore, all pixels in a triangle will have one, two or three motions. To obtain the candidate matches for a pixel in frame1, gray level correlation is used. A relaxation algorithm is used to find the best match for each pixel among these candidate matches. The support for a certain candidate match for a pixel is computed from the four adjacent pixels analogous to the Voronoi neighbors in the case of feature points.

Here two or more pixels may get matched to the same pixel in the next frame. Therefore, we find pixel matches from frame2 to frame1 also and discard those matches which do not match both ways. Near the boundary of moving objects, matches will not be obtained for those pixels corresponding to the scene points which are visible in one frame but not in the other frame. This yields thick bands of unmatched pixels comprising self occlusion regions of a moving object.

5.2 Segmentation of moving objects: The boundaries of the moving objects are obtained based on the similarity of the motion field, in a manner similar to that described in the previous section. However, the detected object boundaries will have errors whenever the motion estimates of pixels are erroneous. This will happen wherever, for example, the number of features in an image part is sparse, leading to rather large triangles. Therefore, the estimates of candidate matches of points within the triangle (for finer level features or pixels) will contain large errors since the estimates are based on linear interpolation of the vertex motions. This will propagate errors down to both feature matching at the finest scale and pixel matching. The intensity structure can be used to help overcome some of these shortcomings.

6. EXPERIMENTAL RESULTS

Experimental results obtained by the algorithm presented are shown in this section for three different image sequences. One of these was obtained in our laboratory and the other two show outdoor scenes.

For the first example a sequence was taken by moving objects with known motion between successive frames as shown in Fig. 4. Only two objects are moving during the first 8 frames, then 3rd object enters in the scene in 9th frame. The small conical shape object is in front and it occludes the bigger cylindrical object in some frames. Figure 4a is the 3rd frame in the sequence. Point correspondences between frame-3 and frame-4 are shown in Fig. 4b. The segmentation of moving objects in frame-3 is shown in Fig. 4c. Segmentation results in frame-8 and frame-12 are shown in figures 4d and 4e respectively. Fig. 4f shows the motion field (for every 5th row and 5th column pixel) between frame-3 and frame-4.

The second example is a sequence of a natural scene with tree Fig. 5a is the third frame in the sequence. Fig. 5b shows the motion field (for every 5th row and 5th column pixel) between frame-3 and frame-4.

The third example is a sequence of a natural scene with cars on a highway obtained from University of California, Berkeley. Fig. 5c is the third frame in the sequence. Fig. 5d shows the motion field (for every 5th row and 5th column pixel) between frame-3 and frame-4.

7. CONCLUSIONS

In this paper, an algorithm is proposed to obtain flow of image points across a sequence starting from sparse point features which yields trajectories, down to pixel level flow. Features as well as pixels are segmented into different moving objects. Experiments were conducted with both laboratory image sequences as well as natural sequences. Here, Intensity maxima and minima were used as features. Other feature point detector may be used to obtain the feature points required for this algorithm.

Perceptual grouping yields reliable correspondences. Although they are sparse, most correspondences found are correct. Inside homogeneous regions, sometimes the pixels are not matched uniquely. When an object also has a motion boundary with little change in texture/gray level across it, the boundary may not be found accurately.

8. REFERENCES

[1] I. K. Sethi and R. Jain, "Finding trajectories of feature points in a monocular image sequence," *IEEE Transactions on Pattern Analysis and Machine intelligence*, vol. PAMI-9, January 1987.

[2] I. K. Sethi, V. Salari, and S. Vemuri, "Feature point matching using temporal smoothness in velocity," in *Pattern Recognition Theory and Applications* (P. A. Devijver and J. Kittler, eds.), pp. 119–131, Springer–Verlag, June 1986.

[3] K. Rangarajan and M. Shah, "Establishing motion correspondences," *CVGIP: Image Understanding*, vol. 54, pp. 56–73, July 1991.

[4] C.-L. Cheng and J. K. Aggarwal, "A two-stage hybrid approach to the correspondence problem via forward sezrching and backward correcting," in *Proceedings of the International Conference on Pattern Recognition*, pp. 173–179, 1990.

[5] C. H. Debrunner, *Structure and Motion from Long Image Sequences.* PhD thesis, University of Illinois at Urbana-Champaign, Urbana, Illinois, 1990.

[6] K. Koffka, *Principles of Gestalt Psychology.* Hartcourt, Brace, New York, 1935.

[7] M. Wertheimer, "Laws of organization in perceptual forms," in *A Source Book of Gestalt Psychology* (W. D. Ellis, ed.), pp. 71–88, Hartcourt, Brace, New York, 1938.

[8] D. G. Lowe, *Perceptual Organization and Visual Recognition.* PhD thesis, Stanford University, 1984.

[9] D. Marr, *Vision.* W. H. Freeman and Company, 1982.

[10] N. Ahuja and M. Tuceryan, "Extraction of early perceptual structure in dot pattern: Integrating region, boundary, and component gestalt," *Computer Vision Graphics and Image Processing*, vol. 48, pp. 304–356, December 1989.

[11] J. K. Aggarwal and Y. F. Wang, "Analysis of a sequence of images using point and line correspondences," in *Proceedings International Conference on Robotics and Automation*, 1987.

[12] J. Weng, N. Ahuja, and T. Huang, "Motion and structure from point correspondences: A robust algorithm for planar case with error estimation," in *Proceedings of the International Conference on Pattern Recognition*, 1988.

[13] R. Tsai, T. Huang, and W.-L. Zhu, "Estimating three-dimensional motion parameters of a rigid planar patch, ii: Singular value decomposition," *IEEE Transactions on Acoustics Speech and Signal Processing*, vol. ASSP-30, no. 4, pp. 525–534, 1982.

[14] A. Rosenfeld, R. Hummel, and S. Zucker, "Scene labeling by relaxation operations," *IEEE Trans. on Systems, Man and Cybernetics*, vol. SMC-6, pp. 420–433, 1976.

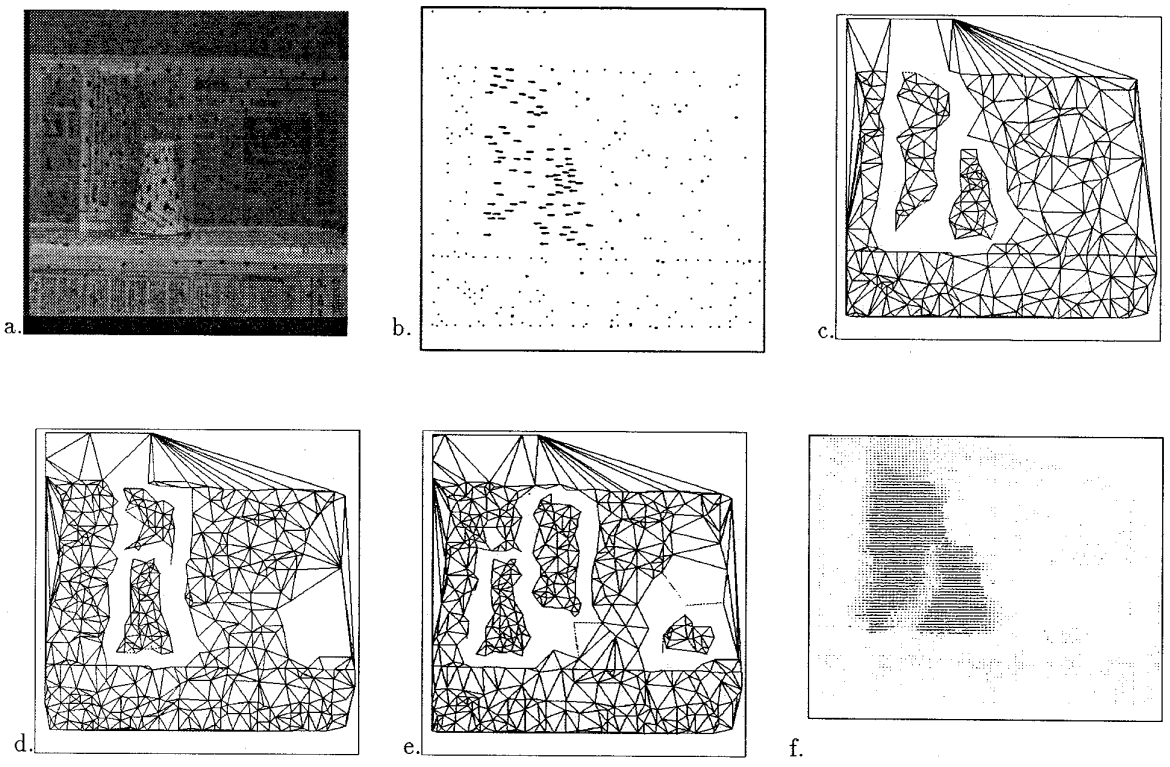

Figure 4: (a) Frame number 3 in the sequence. (b) Feature point matching between frame- 3 and frame-4 (coarsest level) The arrows denote the magnitude and direction of motion. (c) Segmentation of moving objects in frame-3 at the coarsest level. (d,e) Segmentation of moving objects in frames 8 and 12 at the finest level. (f) Display of motion field (shown for every 5th row and 5th column) betw een frame-3 and frame-4.

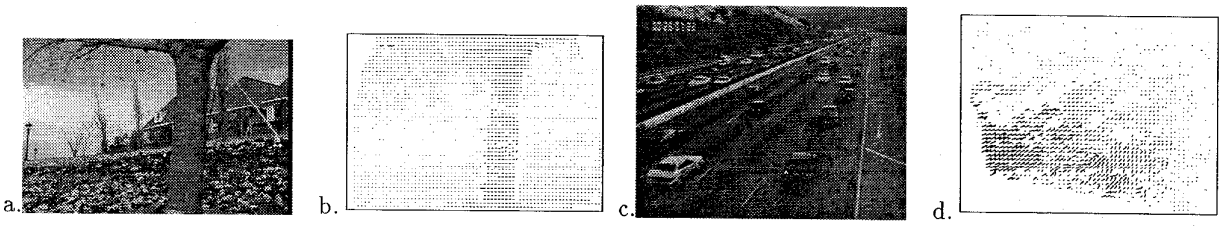

Figure 5: (a) Frame number 3 in the sequence with tree. (b) motion field (for every 5th row and 5th column pixel.) (c) Frame number 3 in the sequence with cars. (d)motion field (for every 5th row and 5th column pixel) between frame-3 and frame-4.