A METHOD FOR MULTIPLE RIGID-OBJECT MOTION SEGMENTATION BASED ON DETECTION AND CONSISTENT MATCHING OF RELEVANT POINTS IN IMAGE SEQUENCES

J. Domingo *G. Ayala* *E. Díaz* *

Instituto de Robótica de la Universidad de Valencia.
C/ Hugo de Moncada, 4. 46010-Valencia (Spain).
email: jdomingo@glup.eleinf.uv.es

ABSTRACT

This communication describes a method to detect and characterise the independent movement of several rigid objects in a sequence of images. No model is assumed for the moving objects, meanwhile a linear model in the image plane is accepted for the pixels of each object, which can be changed without loosing of generality. The approach works on each pair of images. Relevant points are detected on them and then each moving object is identified as two clusters of similar relevant points, one per image, that perform a consistent movement. Differently from previous approaches, the consistent movement constraint is incorporated into the cluster building phase. The main property of the proposed method is its robustness, since minimal assumptions are assumed. Results are provided with tests on two different kinds of scenes: a laboratory one and a natural traffic scene.

1. PROBLEM DESCRIPTION

The problem of movement estimation and object segmentation in a video image sequence has been widely analysed in the literature in the recent years. Time-varying imagery reveals valuable information about the environment, since the world is constantly in motion. Furthermore, some information is only possible to be obtained from an image sequence, not from a single image. For this reason, the analysis of image sequences to estimate 3D motion and scene structure has been at the heart of computer vision research for the last decade [13, 9, 5].

One of the difficulties it involves is that the useful content of an image sequence is intricately coded and implicit in an enormous volume of sensory data. Making this information explicit entails significant data reduction, to decode the spatio-temporal correlations of the intensity values and eliminate redundancy.

This paper aims at describing a method for significant object characteristics extraction and matching based simultaneously on spatial intensities and movement constraints.

The method can be applied to a wide range of cases since a few number of not very restrictive constraints are assumed. First, a single uncalibrated static camera is used. It is supposed that there are no total occlusions amongst objects. Objects are rigid and they move independently one from another. The number of objects in the scene has not to be constant, that is, objects may enter and quit from the scene. Finally, the trajectories and type of movement of each object are unrestricted. These ideas are developed in section 2. In section 3 the method is compared with other approaches. The algorithm is fully described at section 4. Results and guidelines for future developments are explained in section 5.

*The first two authors acknowledge the support of the General Direction of Universities and Research of the Valencian Regional Government through the project GV-2221-94. The third author acknowledges the support of CICYT (Spanish Commission of Science and Technology) through the project TIC95-076-C02-10.

2. APPROACH

The method is based on two main steps: for each image in the sequence, the object is located and described by means of a set of relevant points. The correspondence between the clusters of relevant points from the same object is established by describing the neighbourhood of each point, the geometry of the cluster and the motion needed in order to register the first cluster onto the second one. Let us see a detailed exposition:

(i) Firstly, detection of the relevant points in each image of the sequence has to be done. The concept of relevant point varies depending on the method used to detect them. But it essentially refers to points easily identifiable, independently from illumination, changes of perspective, etc. and sufficiently distinguishable from neighbour points. In our case, these requirements can be accomplished by corners and centres of circular features. The number of points detected must be sufficient to characterise the movement of each object, but small enough to allow fast matching between each point and its correspondent point in the next image, if it exists.

Methods tested for point detection have been Förstner's method [7] and Wang and Brady corner detector [11]. The former is based on a model proposed for two kinds of relevant points: corners and centres of circularly symmetric features. The model for corners supposes that a number of straight border segments ending in the same image area intersect at an unknown point, whose optimal position, found by regression, is the corner. By using a similar reasoning but taking the straight lines which will contribute in the direction of the gradient, as if they were the borders of a hypothetical circle to be found surrounding the points, centres of circular features are detected, too. On the other hand, Wang and Brady corner detector tries to find points where the curvature is a maximum and higher than a threshold, being at the same time the gradient modulus above another threshold, since a corner always belongs to an edge.

The parameters of Förstner's method can be adjusted so as to detect approximately the desired number of points. On the contrary, Wang and Brady's is more difficult to adjust, but it has a lower computational cost. In synthetic or controlled images, almost any method works. In real images, methods adapted to the case must be used.

Notice that, since clusters of relevant points probably correspond with moving objects, isolated relevant points (those with no neighbour within a given distance) will not be interesting for us, and consequently will be removed. This maximum distance to the nearest neighbour is related with the size of the objects in the image, that can be approximately known in advance. From now on, we consider only the remaining points.

(ii) Secondly, we pretend to associate to each moving object a cluster of relevant points in each image of the sequence. The method constructs two of these clusters simultaneously, by establishing the correspondence between the elements of both.

It is obvious that, in order to establish such a correspondence between points, a robust description of the neighbourhood of each relevant point (perhaps, invariant against rotations, translations and affine changes of intensity) is needed. The neighbourhood of a point will be described by means of the autocorrelation function, $C(t)$, which, for stochastic processes with distribution invariant under translations and rotations, is defined as the Pearson's correlation coefficient between random variables (considering as such the grey level at each point of the grid) observed at locations a distance t apart [4]. This function and the used estimator (equation 1) result to be invariant to rigid motions in the image plane and affine changes of intensity.

$C(t)$ has been estimated at t equal to one to six times the inter-pixel distance by means of

$$\hat{C}(t) = \sum_{h \in T} L(h) K(t - \| h \|) \qquad (1)$$

where

$$L(h) = \frac{\sum_{x: x, x+h \in W \cap T} I(x+h) I(x)}{\#\{x : x, x+h \in W \in T\}}$$

being W a circular window centred at each relevant point, T the digital grid of points and I the standardised image, i.e., $I(x)$ is the original grey level value minus the local mean (the mean intensity within the window) divided by the standard deviation. The Epanechnikov kernel has been used as K-function. It is defined as:

$$K(t) = \begin{cases} \frac{3}{4\sqrt{5}}(1 - \frac{1}{5}t^2) & \text{if } -\sqrt{5} \leq t \leq \sqrt{5} \\ 0 & \text{otherwise} \end{cases}$$

Other kernel functions could have been used, as described in [12]. The final result of this step is a vector of n (in our case, 6) components attached to each relevant point, containing the estimations of the autocorrelation function at distances of 1 to 6 pixels from the point.

As a measure of similarity between pairs of points from different images, the Euclidean distance between these vectors has been used. Values of this similarity are calculated for each pair of relevant points. It serves us to discard most of the possible matchings leaving only a small fraction of them as we are going to explain. Nevertheless, this does not happen when the image has several similar regions (see holes of long Meccano part in example 1, which would be a pathological example for purely spatial methods).

A point will be matched with one in the next image, adding both points to their respective clusters. A backtracking algorithm will be used, that builds both clusters at a time. To be included in them, the matching to which the points belong must be consistent. This means that:

- The vectors of local characteristics of both of its points have to be similar.
- In the case of being added, both new clusters must have a similar geometry.
- Finally, the geometric transformations that would lead from the first cluster to the second, with and without the last pair, are not too different.

Section 4 contains a possible (but not the unique) implementation of these ideas.

3. JUSTIFICATION OF THE APPROACH

The problem of independent motion of several objects has been treated in many different ways. We must justify our solution in front of others.

Against trivial image difference methods: they cannot be applied if any object in the next frame occupies all or part of the area occupied by a different object in the present frame. This is very common in car traffic sequences [8].

Against optical flow methods: in general, they have a high computational cost [1], and sometimes it is necessary to apply later split and merge algorithms to fuse all points with a similar movement direction (points of the same object).

Against other approaches based on the same principle (detection and matching of relevant points): one of the most significant works on this field is from Shapiro [11] in which:

- The similarity function for establishing initial correspondences is based on correlations between intensities surrounding each point, so they are invariant to affine intensity changes (if normalised) but not to rotations.
- Appearance or disappearance of relevant points is treated in a non elegant way in the form of forced matchings.
- The trajectories of each point in successive images are assigned to the same object by clustering them after built, so false matchings represent a potential source of error. Furthermore, affinity measures between trajectories are built supposing a model for object movement in the real world, which is not always known.

4. IMPLEMENTATION

The following algorithm is a full description of the implementation of the ideas proposed in section 2.

Step 0 Let $n = 1$ and (s_1, t_1) the pair of points (s_1 in the first image and t_1 in the second image) at minimum distance, i.e., they are the points whose neighbourhood is more similar from our chosen point of view (Euclidean distance between autocorrelation functions, other alternatives can be easily proposed). Let us call C_1 and C_2 the respective clusters of points in the first and second image. We initialise $C_1 = \{s_1\}$ and $C_2 = \{t_1\}$.

Step 1 Let us assume that:

(1) $\{s_1, \ldots, s_{n-1}\}$ and $\{t_1, \ldots, t_{n-1}\}$ are the current pair of corresponding clusters (one per image), being (s_i, t_i) with $i = 1, \ldots, n-1$ the correspondences previously established.

(2) Let us suppose that s_i can be written as

$$s_i = A_{n-1} t_i + B_{n-1} + \epsilon,$$

where A_{n-1}, B_{n-1} are 2×2 and 2×1 matrices; ϵ is a 2×1 random vector normally distributed with null means. Let $\hat{A}_{n-1} = [a_{kl}^{(n-1)}]$ and $\hat{B}_{n-1} = [b_k^{(n-1)}]$ the least-squares estimators of A_{n-1} and B_{n-1} obtained from the set of correspondences $\{(s_1, t_1), \ldots, (s_{n-1}, t_{n-1})\}$.

If the number of current correspondences $(n-1)$ is greater than a certain value, the algorithm finishes (in the following examples this maximum number of points has been fixed to 8 points per cluster); else, choose a pair of correspondent points, (s_n, t_n), candidates to be added. To add them, three conditions must be met:

Similar neighbourhood The similarity between s_n and t_n is the maximum between the *possible* correspondences at the present time, i.e., between pairs of points not previously matched.

Similar geometry The respective Euclidean distances, measured in the image plane, from the candidate s_n to s_i $(i = 1, \ldots, n-1)$ have to be similar to the corresponding distances from t_n to each t_i. A tolerance level is previously fixed.

Consistent movement Both geometric transformations must be similar: that determined by the former points, and that determined by them, together with the candidate couple. This is clearly equivalent to say that the new pair cannot be an influential observation. As a measure of the influence of a new correspondence, Cook's distance has been chosen [2, 3]. Other alternative measures can be found in [6]. Cook [2] proposed that the influence for the i-th component $(i = 1, 2)$ of the n-th correspondence be measured by the distance

$$D_i = (b_i^{(n)} - b_i^{(n-1)})^t X^t X (b_i^{(n)} - b_i^{(n-1)})$$

where X is the $n \times 3$ matrix whose k-th row is $(s_k^t, 1)$, X^t is the transpose of X and $(b_i^{(n-1)})^t = (a_{1i}^{(n-1)}, a_{2i}^{(n-1)}, b_i^{(n-1)})$, i.e., $b_i^{(n-1)}$ are the least-squares estimators of the i-th column of A_{n-1} and the i-th row of B_{n-1} when only the first $n-1$ correspondences are considered. $b_i^{(n)}$ has a similar definition when the n correspondences are considered. D_i is compared with $F(3, n-3, 1-\alpha/2)$ (the $1-\alpha/2$ quantile of a F distribution with 3 and $n-3$ degrees of freedom) for a selected α (a value of 0.05 has been used in section 5). The last correspondence is an influential one if $\max\{D_1, D_2\} > F(3, n-3, 1-\alpha/2)$. Note that two different tests have been applied (one for each component) and a Bonferroni's correction has been used [10].

If the correspondence (s_n, t_n) satisfies the just described three conditions, then it is added and the new clusters are updated to $C_1 = \{s_1, \ldots, s_{n-1}\} \cup \{s_n\}$ and $C_2 = \{t_1, \ldots, t_{n-1}\} \cup \{t_n\}$.

If no matching can be added to the clusters being currently built, the last matching (s_{n-1}, t_{n-1}) is forbidden and the algorithm backtracks to find new consistent matchings, until the desired number of matchings per cluster has been found, or until there are no consistent matchings to incorporate. We keep the best matching found, i.e., the correspondence between clusters for which the common cardinality of the cluster is a maximum and greater than a minimum level (three points in the following examples).

Step 2 Relevant points corresponding to the cluster detected in the previous step are removed. If there are no relevant points in either the first or the second image, the algorithm finishes. Otherwise, the current clusters are set to empty sets and we go back to step 0.

Notice that if the projected movement of the real object is such that it cannot be modelled as a linear geometric transformation, as specified by assumption 2 in step 1, a more complicated model can be adopted without changing the essence of the algorithm, since this assumption is only used in the checking of consistency between matchings, and in the worst case it is a model of movement in the image plane, so it can be determined by standard techniques from a sufficient number of labelled samples.

5. RESULTS AND FURTHER DEVELOPMENTS

The algorithm has been applied to two different kinds of sequences, from which two frames are shown: an image sequence generated in a controlled laboratory environment and a natural traffic image sequence. As it has been previously explained the minimum and maximum number of points per cluster have been fixed to 3 and 8. In the first case, most of the relevant points, either circular or corner points, are well detected. No false matching is generated, and 96% of the possible right matchings are detected, which proves the robustness of the algorithm. (By clarity, not all matchings are shown in the figure).

To check the performance of the algorithm with natural scenes, a traffic image sequence was chosen. In this case the number of interesting points detected and the ratio between the well-done correspondences and the potential number of them decreases due to the poor quality of the images. However, the results show that the algorithm can be likewise useful in this type of large and not controlled scenes, since more than 50% of the possible right matchings are found, which is sufficient to distinguish the cars and estimate their movement. (In our model, a linear transformation in the image plane, three matchings per object are sufficient). Again, not all the found matchings are shown.

The previous elimination of isolated relevant points does not alter the results of the algorithm, but improves the performance, since between 10 and 20% of the initial candidates matchings are removed. Other restrictions adapted to the particular cases can be used (for example, if it is known in advance that each point will have moved to fall into a bounded region).

To impose minimal and maximal cardinalities for the clusters also reduces computation time. Minimal size is compulsory, and has to be chosen at least, as the minimal number of points needed to estimate a transformation under the chosen model. Maximum size is an option, and is taken to avoid search at an excessive depth in the tree of possible matchings. The resulting clusters can be fused later at a lower computational cost, if they prove to be similar.

Formally speaking, we have two sets of points in different spaces (images in this case) and two different problems: first, to distinguish clusters of points (corresponding in our case to different objects) and, second, to establish the correspondences between the detected clusters from different spaces (or images in our application). A problem of clustering and a problem of matching jointly considered. The proposed methodology tries to solve both questions simultaneously. But a question remains unanswered: can this method be reformulated in the form of a single clustering (or simultaneous clustering) problem with a particular matrix of similarities (perhaps, related with the movement of the object)?

6. CONCLUSIONS

A method for multiple rigid motion segmentation has been designed and tested, which is based on the detection and matching of relevant points in a sequence of images. The main novelties consist on the use of local characteristics for the matching which are invariant under rigid motions and affine intensity changes (the autocorrelation function) and also the introduction of a check of consistent movement into the clustering process. Results show the robustness of the procedure when used in low quality natural images.

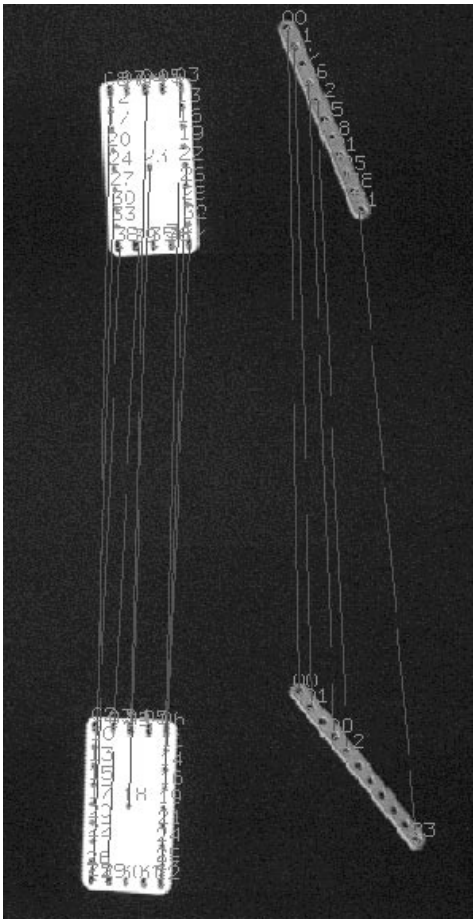

Figure 1. Laboratory sequence

REFERENCES

[1] G. Adiv. Determining 3d motion and structure from optical flow generated by several moving objects. *IEEE Trans. on Pattern Anal. and Machine Intelligence*, PAMI-7:384–401, July 1985.

[2] R.D. Cook. Detection of influential observations in linear regression. *Technometrics*, 19:15–18, 1977.

[3] R.D. Cook. Influential observations in linear regression. *Journal of the American Statistical Association*, 74:169–174, 1979.

[4] N. Cressie. *Statistics for Spatial Data. Second Edition.* John Wiley & Sons, 1993.

[5] D.Park. Adaptive bayesian decision model for motion segmentation. *Pattern Recognition Letters*, 15:1183–1189, 1994.

[6] N.R. Draper and H. Smith. *Applied Regression Analysis. Second Edition.* John Wiley & Sons, 1981.

[7] R.M. Haralick and L.G. Shapiro. *Computer and Robot Vision Volume I.* Addison-Wesley, Reading, Massachusetts, 1992.

[8] N. Hoose. *Computer Image Processing in Traffic Engineering.* John Wiley & Sons, 1991.

[9] H. Murase and R. Sakai. Moving object recognition in eigenspace representation: gait analysis and lip reading. *Pattern Recognition Letters*, 17:155–162, 1996.

[10] G.A.F. Seber. *Multivariate Observations.* John Wiley & Sons, 1984.

[11] L.S. Shapiro. *Affine Analysis of Image Sequences.* Cambridge University Press, Cambridge, UK, 1995.

[12] B.W. Silverman. *Density Estimation for Statistics and Data Analysis.* Chapman and Hall, 1986.

[13] Z. Zhang and O. D. Faugeras. Finding clusters and planes from 3d line segments with application to 3d motion determination. In *Proc. of the European Conference on Computer Vision*, pages 227–236. G. Sandini, 1992.

Figure 2. Laboratory sequence