Motion of Disturbances: Detection and Tracking of multi-Body non-Rigid Motion *

Gilad Halevi and Daphna Weinshall†
Institute of Computer Science, The Hebrew University,
91904 Jerusalem, Israel

Abstract

We present a new approach to the tracking of very non rigid patterns of motion, such as water flowing down a stream. The algorithm is based on a "disturbance map," which is obtained by linearly subtracting the temporal average of the previous frames from the new frame. Every local motion creates a disturbance having the form of a wave, with a "head" at the present position of the motion and a historical "tail" that indicates the previous locations of that motion. These disturbances serve as loci of attraction for "tracking particles" that are scattered throughout the image. The algorithm is very fast and can be performed in real time. We provide excellent tracking results on various complex sequences, using both stabilized and moving cameras, showing: a busy ant column, waterfalls, rapids and flowing streams, shoppers in a mall, and cars in a traffic intersection.

1. Introduction

The tracking of motion in computer vision can be divided into two subtopics: the motion of rigid bodies and the motion of nonrigid bodies. In the latter more complicated case it is often assumed that the changes in the shape of the object are relatively slow and that it is therefore possible to compare local features, such as edges. However, the shape constancy assumption is not always valid, and there are cases in which large changes occur in objects from frame to frame. An example of this is the case of water flowing down a rushing stream. In addition, it is not always possible to extract local features since this process requires successful object segmentation, which cannot always be accomplished reliably (e.g., under camouflage).

Is local shape information the only information that can assist tracking? Introspection tells us that it is far easier to identify objects in motion than stationary objects, and that the identification of motion may preceed the identification of shape. This observation leads us to the following idea: the spatial information in a picture may be used as global information that helps to stabilize large regions in an image, whereas motion is identified by means of temporal changes (disturbances) that are detected within the stabilized region. These disturbances are tracked without regard to the spatial shape of the object being tracked.

Having defined the problem in this way, our solution is the following: we compare the present frame with a "background" image obtained by averaging over previous frames, thus obtaining an effect similar to the effect obtained by a photographer who tries to capture motion by exposing film for a long time: in the over-exposed picture every moving object creates a smeared image in the direction in which it moves. Now all that is left to do is to scatter "tracking particles" in the image, which will lock on these trails and follow them to the present position of each object or motion pattern. By limiting ourselves to following these well-defined trails, we obtain highly stable tracking even when different objects pass in close proximity to each other.

The algorithm basically does not make any assumptions regarding the smoothness of the motion of the objects, the ability to distinguish between objects and the background, or a restriction on the magnitude of the changes that can occur in an object upon passage from frame to frame. Collisions between objects, however, require special treatment. In addition, there is a certain constraint on the velocity of the objects, and the assumption is that the motion of an object from frame to frame does not greatly exceed the dimensions of the object. This assumption nearly always holds, and it is far weaker than the restrictions imposed by most of the existing algorithms in regard to the velocity of objects between consecutive frames.

Thus highly stable tracking is obtained in the following difficult situations, when: (1) the simultaneous tracking of a large number of objects is required; (2) it is difficult to distinguish between the objects and the background (camouflage); (3) the objects undergo complex, non-rigid and varying motion; (4) the shape of the objects varies fast relative to the frame rate.

We shall illustrate the algorithm in the difficult examples of an ant column, water flowing down streams and waterfalls, shoppers in a mall, and cars at a traffic intersection. A review of related literature can be found in the full version of this paper [1].

*This research was sponsored by ARPA through the U.S. Office of Naval Research under Grant N00014-93-1-1202, R&T Project Code 4424341—01.

†To whom correspondence should be addressed: daphna@cs.huji.ac.il

2. Tracking in Stabilized Images

We want to treat cases in which the objects to be tracked undergo significant changes in shape upon transition from frame to frame, e.g., water flowing down a waterfall. Our claim is that in these cases we can still observe a dynamic sequence of abrupt changes in the picture, to be called ”disturbances”, which successively appear and vanish. For simplicity, we begin with cases in which the camera is stabilized, and thus complete frames can be compared.

2.1. The Disturbance Field

In this section we define a disturbance by means of temporal changes between the present frame and previous frames, and we shall obtain a disturbance field, in which every moving object creates a disturbance. The basic structure of the disturbance does not depend on the changes in shading or shape that the object undergoes, but only on its motion. (Cf. to [2] where such changes were used for figure/ground segmentation.)

A **disturbance** is an abrupt change in grey-levels that appears (and disappears) in a certain region and at a certain time. When there are many moving objects, a complete field of disturbances is obtained (see example for one sequence in Fig. 1-middle). In this smooth field every disturbance acts as a locus of attraction, which attracts ”tracking particles” found in the history of that disturbance. The field is obtained by linear subtraction of the temporal average of the previous frames from the last frame in the following manner:

1. The new temporal average at time t is computed as follows, where A denotes the temporal average image, I the actual image after initial smoothing, and $0 < w < 1$ a history factor:

$$A_t = (1-w)I_t + wA_{t-1} \tag{1}$$

2. The disturbance field Δ is computed using linear subtraction of the previous temporal average from the new frame, followed by smoothing:

$$\Delta_t = I_t - A_{t-1}$$

Thus every moving object creates a disturbance in the field in the form of a wave, which includes one extremum at the present position of the object - to be called the ”head” of the disturbance, and a ”tail” having the opposite sign of the head - indicating the previous positions of the object. There is a smooth monotonic path between the extremum at the tail of the disturbance to the extremum at its head (see detailed discussion in [1]).

2.2. Tracking Particles:

Above we defined a disturbance field and showed how every moving object in an image creates a disturbance that includes a ”head” and a historical ”tail”. We shall now define ”tracking particles,” which are attracted to these disturbances. The form of every disturbance is such that it attracts only particles found in its tail, i.e., in its previous positions, and does not attract other particles even if they happen to be closer. In this way excellent separation between different trajectories is achieved, and highly stable tracking is obtained.

Disturbances as attractors: In order to track disturbances, we shall utilize data structures called ”tracking particles”, or simply ”particles”. Every particle contains the following information: (1) location in the picture; (2) tracking state (inactive, tracking, holding), and how much time it has been in the last state; (3) is the object being tracked brighter or darker than the particle; (4) history of the objects' motion (e.g., previous positions, velocities). The third point, describing the relative shading of the object, is a feature which is maintained for a long period of time; therefore, it is sufficient to determine it at the beginning of the tracking (see discussion in [1]).

After we set the relative shading of the object and place the particle on it during initialization, we move on to the tracking stage based on the disturbance field. W.l.o.g. we shall henceforth assume that the shading of the object is bright relative to the background. In this case there is a negative change at the previous position of the object, and the tracking particle will find itself at the minimum at the tail of a new disturbance. The maximum at the head of the disturbance marks the new position of the object and is the location to which the particle must be attracted. Utilizing the fact that there is always a smooth monotonic path between the two extrema, the tracking particle moves along this monotonic path from the previous position of the object (the minimum) to its new position (the maximum).

In order to identify cases in which the object disappears, for example, as a result of occlusion, a very low threshold level is set so that if the extremum found is smaller in absolute value than this level, it is assumed that tracking is lost. In this case the particle switches to a holding state and remains at the last point where the object was detected. During the wait period, the particle continues to search for the object, as we shall see below. If renewed detection is not achieved within a certain time period, the particle goes into the inactive state and searches for a new object to track.

Initialization and revision of tracking particles: The particle initialization stage is designed to find one of the following: (1) a group of “good” objects to track, and the type of object (bright or dark); (2) new objects to track. The number of objects that can be tracked simultaneously is restricted by the number of free particles. Thus, if we have n inactive particles that are not in the tracking state, we can assign to them the n best (i.e., those with the highest absolute value) disturbances. In addition, we must determine the position of the head of the disturbance for each disturbance (or, equivalently, the relative shading of the object). The entire process can be described as follows:

- First the system is allowed to stabilize for time t, so that the average will faithfully reflect the background.

- Next the grid of the disturbance pattern is scanned, and the grid points are sorted in a list in decreasing order of intensity.
- We assign free particles to grid points in the list by scanning the grid points, starting from the highest value. An assigned particle is then drawn to its nearest extremum.
- Each particle is now located either at the head of the disturbance or at its tail. To determine head from tail, the nearest extremum of opposite sign is found, and the absolute values of the maximum and the minimum are compared, selecting the larger of the two as the head of the disturbance.
- Should the disturbance be “vacant,” we position the particle there and switch it to the tracking state; otherwise, we move the particle to the next point in the list of grid points and repeat the process.
- We continue until all the particles that were in the inactive state are used up or until the intensity of the disturbance that we have reached is less than a predefined threshold.

The above process can be actually performed very fast, since we do not examine the entire frame but only a certain number of selected points. The above process repeats itself in each frame after the positions of the active particles are revised. In this way it is possible to discover new disturbances that were not identified previously.

2.3. Summary of The Algorithm

1. Read a new frame and revise the disturbance map (as described in Sec. 2.1).
2. Move particles in the tracking state toward the heads of the disturbances, and assign inactive particles to new disturbances that are not yet covered (as described in Sec. 2.2).

The free parameters in the algorithm are: (1) the threshold level for identifying a new disturbance; (2) The threshold level for identifying the disappearance of a disturbance (less or equal to the former level); (3) the number of tracking particles; (4) The history factor w. Our initial experiments showed robustness to the choice of the parameters: different parameter values did not lead to significantly different results. Thus, in all the different experiments to be described, we retained the same parameter values without change. Parameters were *not* tuned to obtain optimal results per each sequence.

2.4. Results

Fig. 1 shows the results using a sequence depicting the activity in a column of ants that move along fast while changing their trajectories and velocities almost randomly

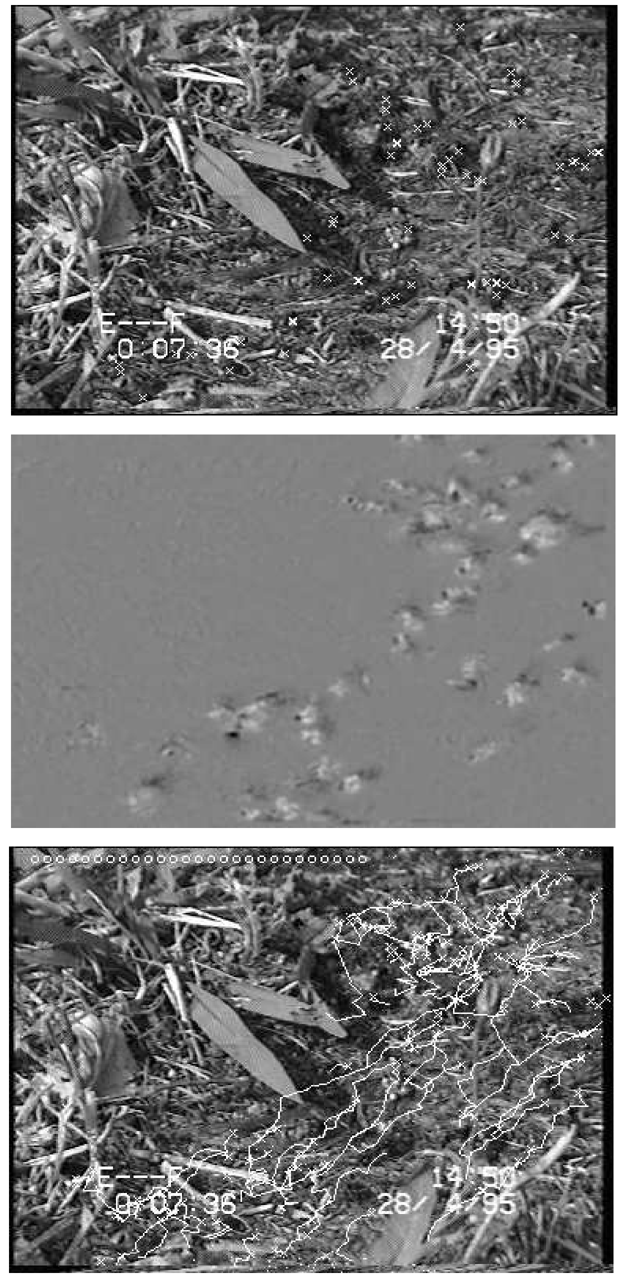

Figure 1. Results with the ant sequence. *Top:* **ants detection: each disturbance attracts one particle.** *Middle:* **the disturbance field, where the bright areas denote the disturbances heads while the dark denote their tails.** *Bottom:* **the trajectories of the ants after 30 frames; note that the trajectories are smooth despite the complexity of the motion. (See demonstration of these results in http://www.cs.huji.ac.il/˜ daphna.)**

in their attempt to overcome various obstacles. In the top picture, each x indicates the position of a particle, corresponding to an individual ant; the bottom picture shows the trajectories of the disturbances (ants) after 30 frames. Apart from the fact that the shading of the ants is very similar to the background and it is very difficult to discern them when they are not in motion, we are dealing here with a very complex scene: the ants are avoiding obstacles, climbing on branches, or passing beneath leaves, being partially or fully occluded. They bump into one another and intermittently change their angle and shape. Despite all this, stable tracking over a long period of time is attained, and it appears that the trajectories are mostly accurate except for isolated points.

Fig. 2 shows the results obtained for various sequences of waterfalls and water flowing down a stream. In these cases the changes from frame to frame are enormous. Nevertheless, good tracking of the flow is attained.

3. Tracking in Non-Stabilized Images

The computation of the disturbance field until now was based only on temporal changes. This requires that the region under consideration be stabilized. When the camera moves, we perform affine registration of the average for each new frame, and then we compare and revise the average for the ensuing frames. Since the comparison to the average drastically lowers the sensitivity of the computation to noise, better results are also obtained in cases in which the registration is not perfect. In this section we discuss how to incorporate affine stabilization into our algorithm, and illustrate the results on a few examples.

The frame stabilization is a process that is based entirely on spatial information, and its purpose is to achieve optimal overlap between a pair of frames on the basis of a limited number of parameters. In our case the frames that we want to compare are the new frame and the picture of the average of the previous frames, which reflects the background.

If we would perform registration of the *last frame* to the background frame, as is normally done, we would very quickly reach a situation in which there are almost no areas of overlap between the new frames and the average due to the motion of the camera and its increased distance from the first frame. For this reason we employ the opposite approach of repeated registration of the *background* to the last frame. In this way maximum overlap between the average and the new frame is always maintained, cf. [10].

The stabilization process is based on a search for affine correspondence between the frames. In [8] it was proved that this computation always converges to a result that correctly reflects the most dominant motion in the scene. In most cases it is the motion of the background. Therefore, this stabilization method can also be applied to scenes that include the independent motion of many objects relative to the background.

After performing the registration and revising the disturbance map, registration of the particles should be performed according to the parameters found. In this way we maintain

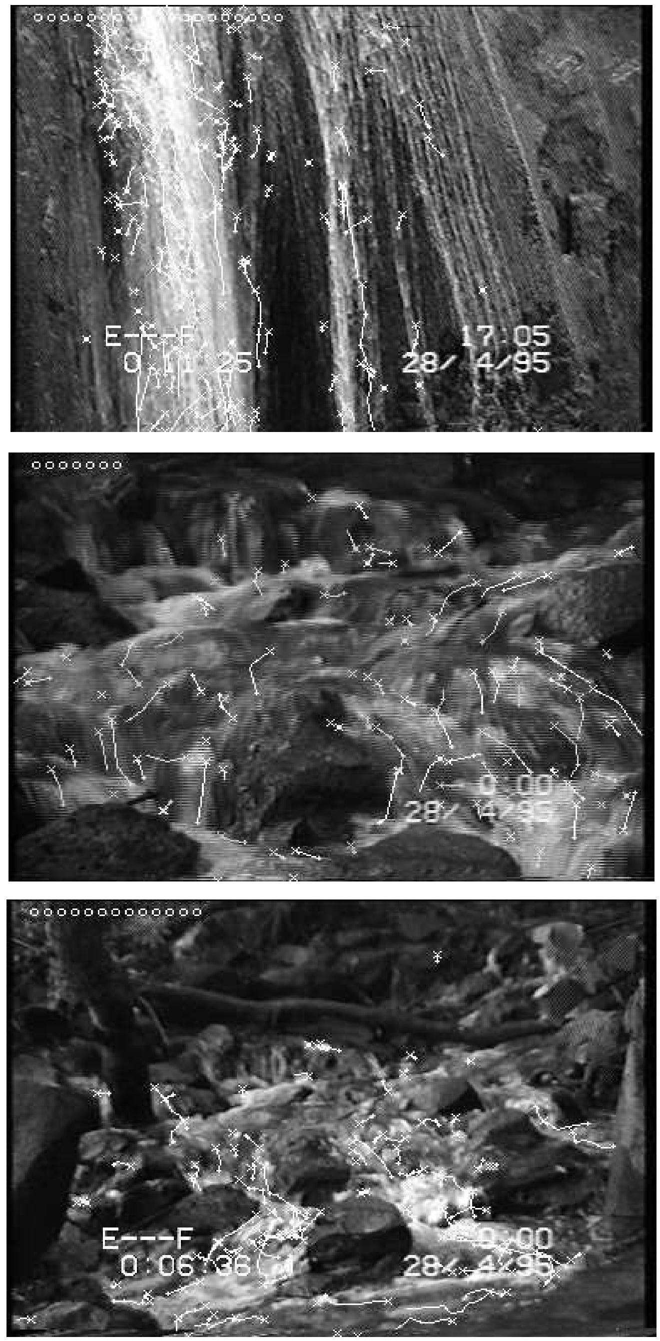

Figure 2. Sequences of waterfalls and river flows: *Top:* **the computed trajectories super-imposed on one frame from the waterfall sequence.** *Middle:* **the computed trajectories super-imposed on one frame from a complex rapids sequence; the trajectories follow the flow of the water correctly (when looking at a movie of the particles, they appear to be carried downstream by the water, see http://www.cs.huji.ac.il/~ daphna).** *Bottom:* **similarly for another rapids sequence.**

the stability of the particles positions relative to the background. Only after this step can the particles be displaced according to the new disturbance map.

Fig. 3 shows 3 frames from a 60 frames movie taken from the second floor of a shopping mall while the camera was moving downward. Three figures moving along the entire sequence, for which individual stable continuous tracking trajectories were obtained, can be discerned (see trajectories in frame 3). Stable tracking was not achieved for the fourth figure, the soldier moving along the lower right-hand side of the picture, possibly because his direction of motion was similar to the camera's motion. In this example we used tracking particles which have a dynamic size, i.e., the particle size varied during tracking in order to optimally fit itself to the size of the disturbance that it is tracking.

Fig. 4 shows a group of frames from a sequence of 60 frames showing vehicles going through a traffic intersection. Many of the vehicles make a left turn accompanied by a change in their two-dimensional projection. Nevertheless, the tracking remains stable along the entire sequence (see trajectories in last frame).

4. Comparison with Other Methods

Most of the published methods for tracking nonrigid objects are not suitable for handling the examples presented here. The basic assumption is almost always that the shape of the object changes slowly (small deformation), as in the methods based on the occluding contours of objects [3], or methods based on extreme points in the object [4]. Clearly, these approaches are not suitable to handle flowing water (what are the "objects" in this case?), or camouflaged ants where occluding contours cannot be discerned reliably. Also, an approach like that described in [6], in which the minimum of the Hausdorff distance between prominent points sampled from the frame is sought, requires that the shape of the object changes slowly between consecutive frames, and thus it does not meet our needs. The only methods that are not based on the assumption of slow changes, such as the method described in [5], require a geometric model that is confined to very specific cases. Typically, however, we do not have a general geometric model for flowing water or the motion of ants.

The methods that could potentially be successful are the conventional methods for computing a motion field on the basis of brightness changes. These algorithms start out from the erroneous assumption (erroneous in our case) that all the changes in shading are caused only by motion and are not caused by any other factor, particularly not by changes in shape; at the same time, however, they are general enough to deal with diverse kinds of motions and do not require the isolation of the objects being tracked or the knowledge of their shape.

We performed extensive experiments with the algorithm of Lucas & Kanade [11]. With all our efforts, the optical flow results obtained by this algorithm using the ants sequence (which was very challenging for this algorithm since it violated all the algorithm's basic assumptions) were far

Figure 3. Three frames from a long sequence, taken with a downward moving camera while the people in the scene are moving in different directions. There are 60 frames between the first picture (top) and the last one (bottom); still, 3 figures are reliably tracked, and the tracking is very stable, as indicated by the faint trajectories in black and white (see demonstration in http://www.cs.huji.ac.il/˜ daphna).

Figure 4. Three frames from another long sequence, taken by a non-stabilized camera in a traffic intersection. The cars which turned to the left changed their $2D$ projection non-rigidly. There are 60 frames between the first picture (top) and the last one (bottom). The computed trajectories are shown with faint white lines (see demonstration in http://www.cs.huji.ac.il/˜ daphna).

from the true motion, and stable tracking was not possible. Using the waterfalls sequences, the direction of the optical flow typically matched the actual motion, but the magnitude was far from the true speed, rendering tracking along a few frames impossible. These experiments, as well as comparisons to point matching algorithms, are described in [1].

In conclusion, in our experiments our algorithm was very reliable and performed better than other algorithms on sequences with many independently moving objects or complex patterns of motion. The reason for the superior performance seems to be the use for tracking of a shape invariant property, whose character is only weakly influenced by the shape of the moving objects. In addition, there was a difference of orders of magnitude between the run time of our algorithm (a few frames per second) to that of the multiscale optical flow algorithm (a few minutes per frame).

References

[1] G. Halevi and D. Weinshall, "Motion of Disturbances: Detection and Tracking of multi-Body non-Rigid Motion", *CVPR*, San-Juan, 1997; full version can be obtained from "http://www.cs.huji.ac.il/˜ daphna/".

[2] H. H. Nagel, "Formation of an Object Concept by Analysis of Systematic Time Variation in the Optically Perceptible Environment", *CVIP,* 7:149-194, 1978.

[3] A. Kass, A. Witkin and D. Terzopoulos, "Snakes: Active Contour Modes", *IJCV,* 1(3):321-331, 1988.

[4] I. Cohen, N. Ayache and P. Sulger, "Tracking Points on Deformable Objects, Using Curvature Information", *ECCV 92,* pp. 458-465, 1992.

[5] J. P. Berroir, I. Herlin and I. Cohen,. "Non-Rigid Motion Without Relying on Local Features: ...", *INRIA Repport,* No. 2684, 1995.

[6] D. P. Huttenlocher, J. J. Noh and W. J. Rucklidge, "Tracking Non-Rigid Objects in Complex Scenes", *ICCV 93*, pp. 93-101, 1993.

[7] H. Applebaum, M. Werman, "The detection of motion in noisy environments using a 3D f* algorithm", *PhD thesis*, Hebrew University, 1995.

[8] P. J. Burt, R. Hingorani and R. J. Kolezynski, "Mechanism for isolating component patterns in the sequential analysis of multiple motion", *IEEE Workshop on Visual Motion*, pp. 187-193, 1991.

[9] B. K. P. Horn and B.G. Schunck, "Determining Optical Flow", *Art. Intel.* 17:185-203, 1981.

[10] M. Irani, B. Rousso and S. Peleg, "Computing Occluding and Transparent Motion", *International Journal of Computer Visionq*, 1993.

[11] B. D. Lucas, T. Kanade. "An Iterative Image Registration Technique with an Application to Stereo Vision", *IU workshop*, pp. 121-130, 1981.