A Fast and Robust Point Tracking Algorithm

C.J. Veenman, E.A. Hendriks and M.J.T. Reinders
Delft University of Technology, Dept. of Information Technology and Systems,
Information & Communication Theory Group,
P.O. Box 5031, 2600 GA Delft, the Netherlands
Email: {C.J.Veenman, E.A.Hendriks, M.J.T.Reinders}@its.tudelft.nl

Abstract

We present an algorithm that efficiently tracks a predefined set of landmark points in a time sequence of images. The algorithm iteratively optimizes the correspondences between the point measurements in the images, while allowing for spurious and missing point measurements. This trajectory based approach hypothesizes missing points by interpolation. Spurious measurements are either left out because they do not form the optimal correspondences or are removed afterwards if they have the smoothness or other constraint exceed its predetermined maximum.

1. Introduction

For image sequence analysis the problem of tracking landmark points is important for a variety of applications, like among others 3D scene geometry estimation, pose estimation or object tracking. Our interest lies in real-time tracking of object deformations. In particular we are interested in the tracking of changing facial expressions within the interest of performer-driven animation or model based coding.

To realize real-time landmark point tracking we consider landmark points that have relatively simple but unique image properties, for example points with a specific (enhanced) color or high hessian points. Then, the detection can be done relatively simple, e.g. by thresholding, at the cost of losing the identification of these points. In this paper we present a fast and efficient solution for tracking these detected landmark points over a number of images, allowing an imperfect detection scheme hence missing or spurious detected points.

Several techniques with polynomial approximations for this exponential problem have been published [1]. The Multiple Hypothesis Tracker (MHT) originally developed by Reid [5], for which Cox and Hingorani [2] made an efficient approximation, is able to track a variable number of points and explicitly models spurious and missing point measurements. The Joint-Probabilistic Data-Association Filter (JPDAF) algorithm by Fortmann, Bar-Shalom and Scheffe [3] suits best, when there are a predefined number of points to track. These techniques are specifically designed to deal with a lot of spurious point measurements and hence are rather complex, conceptually as well as computationally. In our application, with little spurious and missing measurements, a less complicated technique suffices.

The algorithm we propose, is based on Sethi and Jain [7]. Their algorithm is a greedy optimization algorithm (Greedy Exchange, GE) in that it iteratively optimizes point correspondences. This trajectory based approach optimizes inherently local and it is sensitive to the initial correspondences. In addition it explicitly models the uniqueness constraint, that is one landmark point uniquely matches one point measurement. Sethi and Jains algorithm does however not model spurious and missing point measurements. A modified version of this algorithm [4, 6] solves both problems a great deal, but it may lead to a number of short unconnected trajectories and wrong correspondences.

The algorithm we propose enhances the GE algorithm differently by preserving as much motion information as possible, especially in case of missing measurements. It does not divide trajectories up easily and makes better correspondences.

After making the problem statement in the next section, we continue with a description of the proposed algorithm and end with the conclusions. In the remainder of this paper we abbreviate landmark points to *points* and point measurements to *measurements*.

2. Problem statement

We define the tracking problem as follows. Given a sequence of n time instances. At each time instance t_k there are m_k measurements x_{ik}, with $1 \leq i \leq m_k$ and $1 \leq k \leq n$. At t_1, $\boldsymbol{m}$ points ($\boldsymbol{m} \leq m_1$) are identified among the m_1 measurements by an operator or otherwise. The task is to track these $\boldsymbol{m}$ measurements over the whole sequence. In other words: return a set of $\boldsymbol{m}$ trajectories, that represent the motion of the $\boldsymbol{m}$ points from $t_1, t_2, .. \text{ to } t_n$. A trajectory is a list of measurements: $x_{i_1 1}, x_{i_2 2}, ..., x_{i_n n}$, with $1 \leq i_k \leq m_k$.

.

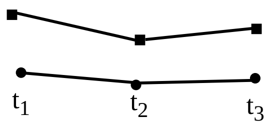

Figure 1. Two moving points, that are each measured three times. The lines represent the point correspondences in time**.**

3. The Greedy Exchange Algorithm

The GE algorithm by Sethi and Jain [7] hypothesizes the correspondences between the measurements and then repeatedly exchanges correspondences between trajectories, that better match the optimization criterion which consists of local smoothness in both the velocity and the direction.

The original GE algorithm has two major drawbacks. First, it assumes that there are no missing measurements by occlusion or otherwise. Second it does not allow for spurious measurements or false alarms. That is, in the GE algorithm m_k always equals $\boldsymbol{m}$.

Salari and Sethi [6] solve the missing and spurious measurements problem by introducing phantom points. First these phantom points replace missing measurements, while satisfying local constraints. By having the maximum allowed local smoothness criterion (ϕ_{max}) and a maximum displacement (d_{max}), they also replace spurious measurements. That is, when spurious measurements are present in a trajectory they will be replaced by phantom points, if the phantom points cause a lower criterion value.

This approach generally works fine except that missing measurements always have the maximum criterion and displacement. For instance if point p has not been measured at t_k, the algorithm can easily associate a measurement of p at t_{k+1}, to another point which is within the criterion range ϕ_{max}. Moreover, the phantom points only enforce that the local movement constraints are satisfied, but by putting a phantom point in a trajectory, the trajectory is in fact divided into two trajectories. In other words, the maximum criterion approach solves the correspondence problem up to the maximum criterion. Choosing a low maximum leads to too many undecided trajectory parts and a high maximum leads to possibly wrong correspondences.

Here, we propose a solution to this problem by interpolating the missing measurement locations using preceding and succeeding measurements in the trajectory established so far. In this way we retain as much motion information as possible and we are therefore able to make more plausible correspondences.

3.1. Algorithm description

We represent the problem data in a matrix with m_{max} rows and n columns, with m_{max} is the maximum among all m_k's. The measurements are filled in from the first row in the column and when there are less than m_{max} measurements the last rows are kept empty. The first $\boldsymbol{m}$ rows represent the trajectories of the $\boldsymbol{m}$ landmark points (see Figure 2.).

The algorithm repeatedly optimizes the local criterion from t_1 to t_n. At t_k, we consider the trajectories correct up to t_k and we check if correspondence exchange between any two trajectories at t_{k+1} improves the criterion the most.

In our application the (facial) point movements are hardly restricted by inertia. Therefore, the smoothness assumption, which [7] uses, will not hold in this situation. The closeness criterion, which minimizes the total motion between two time instances suits better in this case. We define the closeness criterion at t_k from row r_i to r_j ($1 \leq i \leq m_k$, $1 \leq j \leq m_{k+1}$) as:

$$c_{ij}^{k} = |x_{ik} - x_{ik-1}| + |x_{jk+1} - x_{ik}|$$

To cope with missing data, we estimate the distance $|x_{ik} - x_{ik-1}|$ by scanning back in the row to collect two measurements in the nearest past and normalize this distance by the number of columns (time instances) in-between them plus 1. If x_{jk+1} is present and x_{ik} is missing, we estimate the distance $|x_{jk+1} - x_{ik}|$ likewise. We do not consider a missing measurement at t_{k+1}, because then there is nothing to exchange.

Besides the closeness criterion, in some experiments we use the smoothness criterion as in [7] (see 4. Experiments). For this criterion the missing data are interpolated similar as described above, which gives estimates for the vectors $\overrightarrow{x_{ik-1}x_{ik}}$ and $\overrightarrow{x_{ik}x_{ik+1}}$.

In the case of spurious measurements there are m_{max} - $\boldsymbol{m}$ spurious trajectories. Clearly we only minimize the criterion within the $\boldsymbol{m}$ point trajectories. Eventually the spurious measurements migrate to the spurious trajectories.

$$\left.\begin{bmatrix} x_{11} & x_{12} & x_{13} & x_{14} & x_{15} & \cdots & x_{1n} \\ x_{21} & x_{22} & x_{23} & x_{24} & x_{25} & \cdots & x_{2n} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{\boldsymbol{m}1} & x_{\boldsymbol{m}2} & \varnothing & x_{\boldsymbol{m}4} & x_{\boldsymbol{m}5} & \cdots & x_{\boldsymbol{m}n} \\ \varnothing & \varnothing & \varnothing & \varnothing & x_{m_5 5} & \cdots & \varnothing \end{bmatrix}\right\} m_{max}$$

Figure 2. An m_{max} x n matrix, representing the trajectories of point 1 to point m from t_1 to tn. At t_3 there is one missing measurement and at t_5 there is one spurious measurement.

Per time instance the algorithm minimizes the criterion by exchanging measurements that maximize the gain in the criterion for the point trajectories including the measurements in the spurious trajectories.

We define the gain in the point trajectories ($1 \leq i, j \leq \boldsymbol{m}$) in the following way:

- $g_{ij}{}^k = c_{ii}{}^k + c_{jj}{}^k - (c_{ij}{}^k + c_{ji}{}^k)$, when both x_{ik+1} and x_{jk+1} exist.
- $g_{ij}{}^k = c_{ii}{}^k - c_{ji}{}^k$, when only x_{ik+1} exists.
- $g_{ij}{}^k = c_{jj}{}^k - c_{ij}{}^k$, when only x_{jk+1} exists.

As the spurious trajectories do not need to meet any criterion they are very willing to give up their measurements. That is, if the point trajectory has any favor by taking a measurement from a spurious trajectory, it may take it.

Therefore between the point and spurious trajectories ($1 \leq i \leq \boldsymbol{m}$ and $\boldsymbol{m} + 1 \leq j \leq \boldsymbol{m} + m_{k+1}$) we define the gain:

- $g_{ij}{}^k = c_{ii}{}^k - c_{ij}{}^k$, both x_{ik+1} and x_{jk+1} exist.

This leads to the following algorithm.

1. *Initialization:* Initialize the point trajectories with nearest neighbors. That is, starting at t_2 assign all measurements at t_k to the trajectory with the nearest measurement at t_{k-1}, for $2 \leq k \leq n$. Ignore ambiguities during the assignment, like among others a different number of measurements between t_k and t_{k-1}. That is, assign arbitrarily in such cases.
2. *Exchange loop:* For $t = t_2$ to t_{n-1}:
   a. Compute gain within point trajectories; i.e. for $i = 1$ to $\boldsymbol{m} - 1$ and $j = i + 1$ to $\boldsymbol{m}$.
   b. Compute gain between point trajectories and spurious trajectories; i.e. for $i = 1$ to $\boldsymbol{m}$ and $j = \boldsymbol{m} + 1$ to $m_{\boldsymbol{k}}$.
   c. If there is any gain achievable, then exchange those measurements that result in the highest gain.
3. *Termination:* If the exchange loop did not yield gain for any *t*, then terminate. Otherwise go back to the exchange loop.

Because the optimization criterion is always positive or zero, there will eventually be no gain achievable. Consequently the algorithm always terminates.

So far we assumed that missing and spurious measurements do not occur simultaneously. In such cases some of the presumed spurious measurements remain in the point trajectories. To discover those points, we calculate the criterion in all established point trajectories after the optimization algorithm has terminated. If the criterion $c_{ij}{}^k$ exceeds a certain threshold (ϕ_{max}) in any trajectory, we assume that it is caused by a misinterpreted spurious measurement. We then remove that measurement and restart the optimization algorithm. This procedure surely terminates, since the number of measurements is finite.

4. Experiments

We first tested the algorithm on a sequence of facial images with enhanced color marks. In this sequence we used the *closeness criterion* as described above. Since we do landmark point segmentation based on a fixed color threshold, once in a while some measurements can be missed. To detect outliers (spurious measurements) we define the maximum closeness criterion ϕ_{max} = 100 pixels in within a trajectory (see Figure 4.).

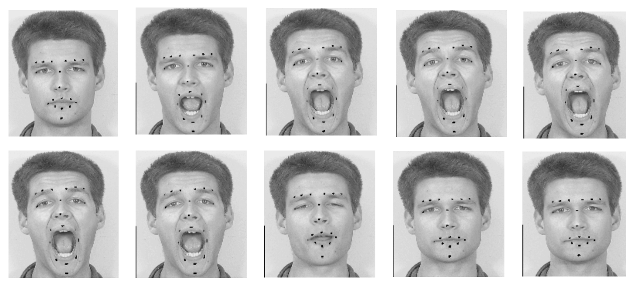

Figure 3. A sequence of 10 720x576 images with overlaid detected points. In image 3 to 6 the left upper lip location is missed

The number of spurious measurements and simultaneously missing and spurious measurements is usually low. Therefore we used synthetic data to test the algorithm in these circumstances and to compare it to Salari and Sethi (S&S) [6].

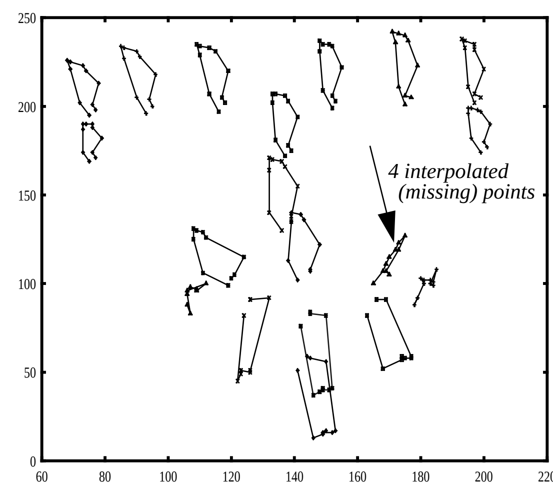

Figure 4. This figure shows 19 trajectories of enhanced color landmark points taken from the moving face sequence (see Figure 3.). The left upper lip point wa not measured from t_3 until t_6. and has interpolated points in this picture.

In all synthetic data examples we use the *smoothness of velocity and direction criterion* as do [6, 7]. To resolve the unsmooth start problem, we optimize the trajectories forwards and backwards until there is no gain achievable in either direction [7]. As an aside, although Sethi and Jain [7] did not mention it, this way of optimizing may not terminate.

The first synthetic sequence shows two pairs of moving landmark points. One of the four points has not been measured during two time instances. Moreover, there are a number of (randomly inserted) spurious measurements. After putting the measurements in random order (per time instance), the algorithm makes correct correspondences and correctly handles the spurious and missing measurements. That is, the spurious measurements end up in the spurious trajectories and the trajectory of the point with missing measurements has indeed two empty positions, which we filled with linear interpolated values (see Figure 5.).

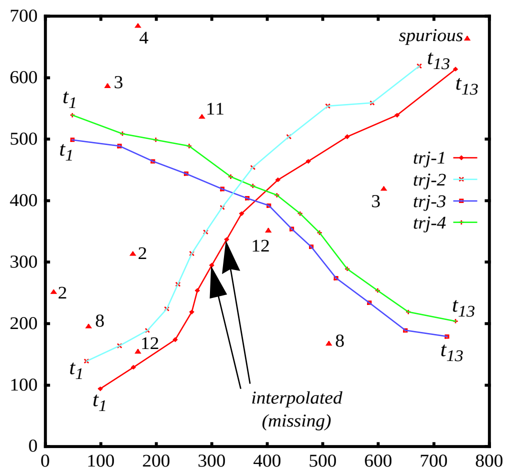

Figure 5. Result of correspondence optimization in a data set with both spurious and missing measurements. In trajectory 1 the missing measurements have been filled in with linear interpolations.

The last example shows two crossing landmark points with a missing measurement at t_3 for the first and at t_4 for the second point (see Figure 6.). Our algorithm correctly finds the point trajectories (see Figure 9.). The S&S version either leads to wrong correspondences or to separate trajectories. We use two different settings of ϕ_{max} to show the shortcomings of the S&S version of the GE algorithm.

First with a high ϕ_{max} (> 0.10) the algorithm makes wrong correspondences (see Figure 7.). The algorithm prefers points that have been measured in the last time instance, if they obey the maximum allowed criterion.

On the other hand, when ϕ_{max} is lower (0.05) the algorithm separates four trajectory parts (see Figure 8.) and correspondences between the trajectory parts have to be made afterwards..

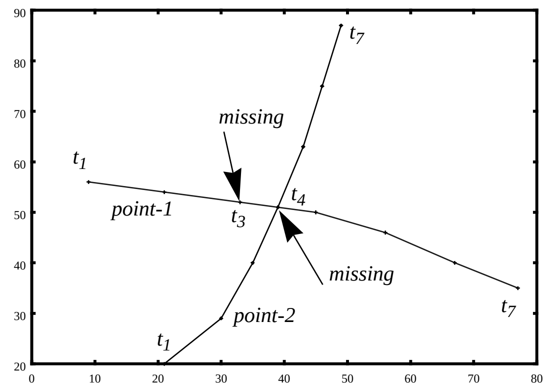

Figure 6. The input measurements with 7 time instances, but at t_3 a measurement for point-1 is missing and at t_4 a measurement of point-2 is missing.

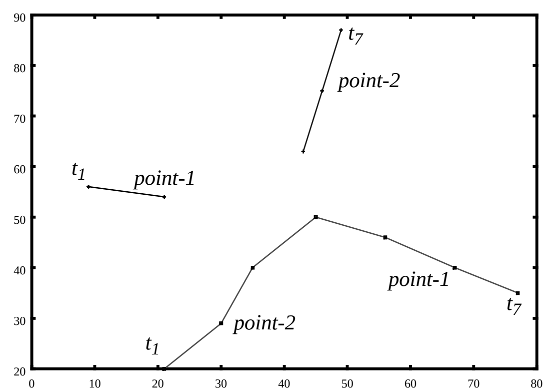

Figure 7. Three trajectory parts that are found by the S&S algorithm using ϕ_{max} > 0.10 and d_{max} = 20. Clearly the correspondence conflict is wrongly solved at t_4.

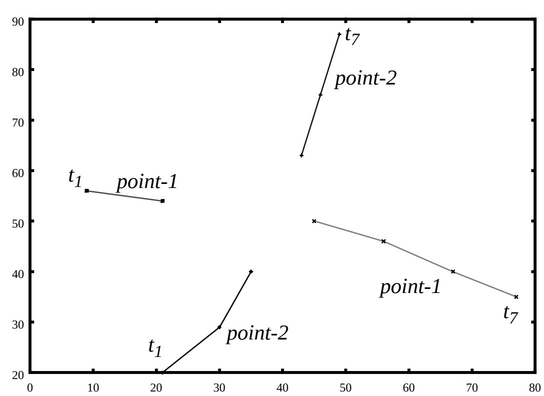

Figure 8. Four trajectory parts that are found by the S&S algorithm using ϕ_{max} = 0.05.

Our GE algorithm interpolates the missing measurements and forms the most plausible trajectories. Here we give some example exchange decisions:

At t_2: $c_{1.}^{2}$ uses x_{11}, x_{12}, x_{j3} and c_{2j}^{2} uses x_{21}, x_{22} and again x_{j3}. Clearly x_{j3} in trajectory 2 gives the highest gain (j is the trajectory the third measurement is in).

At t_3: c_{1j}^{3} uses x_{11}, x_{12}, x_{j4} and c_{2j}^{3} uses x_{22}, x_{23} and again x_{j4}. Now x_{j4} clearly fits better in trajectory 1 (j is the trajectory the fourth measurement is in).

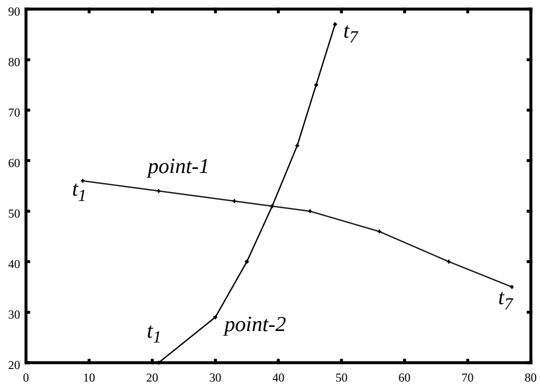

Figure 9. Resulting trajectories found by our algorithm. In the figure we interpolated the missing points, because our algorithm connects the trajectories. We used $\phi_{max} \geq 0.05$.

5. Conclusions

In this paper we showed that our modified version of Sethi and Jains greedy exchange algorithm works well even in the presence of spurious measurements and when some measurements are missing. We also showed that the phantom points approach [6] to overcome these imperfect detection problems is in some cases less robust.

In our experiments we used the closeness and smoothness criterion. The algorithm also applies with other criteria, as long as they are positive and monotonically increasing. For instance, the number of previous measurements taken into account in the local criterion computation, can be increased to get higher order smoothness. Besides, including more measurements will also increase the interpolation quality.

For the missing measurement determination we now unrestrictedly gather previous measurements in a trajectory. Interpolating the most recent measurements this way, will probably lead to problems in case of very long occlusions. That is, in such cases the motion information of the last measurements seen, is probably unrelated to the current motion of these points.

As mentioned, the optimization scope of the algorithm is limited, because its greedy nature and the way the gain is computed. The algorithm only considers an exchange of *two* measurements, making a cyclic exchange between three measurements hard to find. Increasing this number leads to an exponential growth in the number of gain computations, which is $O(\boldsymbol{m}^2)$, in the current scheme.

Finally, in contrast to Salari and Sethi our algorithm is currently not able to track a changing number of landmark points. Including more trajectory constraints, enables the algorithm to be also applicable in those circumstances.

6. References

1 I.J. Cox, "A Review of Statistical Data Association Techniques for Motion Correspondence", *International Journal of Computer Vision*, 10 (1), 1993, pp.53-66.

2 I.J. Cox, and S.L. Hingorani, "An Efficient Implementation of Reid's Multiple Hypothesis Tracking Algorithm and Its Evaluation for the Purpose of Visual Tracking", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18 (2) February 1996.

3 T.E. Fortmann, Y. Bar-Shalom, and M. Scheffe, "Sonar Tracking of Multiple Targets Using Joint Probabilistic Data Association". *IEEE Journal of Oceanic Engineering*, 8 (3) July 1983.

4 R. Jain, R. Kasturi, and B.G. Schunck, *Machine Vision*, McGraw Hill, New York, 1995, pp. 443-451.

5 D.B. Reid, "An Algorithm for Tracking Multiple Targets", *IEEE Transactions on Automatic Control*, 24 (6) December 1979.

6 V. Salari, and I.K. Sethi, "Feature Point Correspondence in the Presence of Occlusion", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12 (1) January 1990.

7 I.K. Sethi, and R. Jain, "Finding Trajectories of Feature Points in a Monocular Image Sequence". *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9 (1) January 1987.