Probabilistic Matching Algorithm for Keypoint Based Object Tracking Using a Delaunay Triangulation

Trichet Rémi, Mérialdo Bernard
Eurecom Institut, Sophia-Antipolis, France
Remi.trichet@eurecom.fr, Bernard.Merialdo@eurecom.fr

Abstract

This article presents a matching algorithm developed for a generic object tracking system. Matching is a critical part for the effectiveness of tracking. The proposed method is a probabilistic algorithm inspired from the emerging "discriminative random fields". Points are associated according to their visual similarity and to spatial relations in their neighborhood, based on a Delaunay triangulation. Experimental results are presented to validate this contribution.

1. Introduction

Our work takes place in the context of the Portivity European project. This project aims at developing an interactive television system, which can realize direct interactivity with moving objects on hand-held receivers. In this framework, the need of a generic tracking system, able to deal with all kind of videos has arisen. In our previous work **[1]**, a keypoint based tracking system was build up in order to fulfill the specific requirements of such an application. It rapidly came out that, the matching algorithm was the angular stone of our system: matching errors propagate to the subsequent steps of the tracking process and directly influence the quality of tracking. As a consequence, our efforts have been directed towards the improvement of the matching algorithm.

During our first experiments, we have stated several weaknesses of the matching algorithm in various cases. For instance, some points, while perfectly localized, were not matched because the distance between their descriptors was slightly below the threshold. In the case of spatially close points, the descriptors are frequently similar and sometimes lead to point inversion during matching. These mistakes were further disturbing the motion model. In order to reduce this problem, we have decided to broaden our matching criterions by using not only visual information, but also spatial information.

We have developed a probabilistic algorithm assessing the likelihood of the association of two given points according to both their visual descriptors and the correspondence of the surrounding points.

This article is organized as follows. The next section will summarize the main issues and the existing techniques in the domain of primitive tracking. The third section will describe our algorithm and some results will be presented in the fourth one. Finally, the last section will conclude and discuss possible enhancements.

2. Previous work in primitive tracking

Primitive tracking consists in matching together points in consecutive images. Each point belongs to a trajectory. However, because of sensors perturbation, flaws of the algorithms, occlusions of the tracked primitive, or shifts of the primitive characteristics, trajectories can appear and disappear. A point in image *t* may have one or no equivalent in image *t+1* and sites of image *t+1* may have no associated points. In order to solve these ambiguities, constraints are defined. These constraints can refer to characteristics of the primitives, to their motion, or to relations between them. Algorithms can be classified based on the constraints they use and their way to exploit them.

The first class of methods focuses on the association of a point to a trajectory. A comparison is available in **[2][3]**. The approach developed by **[4]** first links the closest points and then modifies the associations in order to build smooth trajectories. In **[6]** a method proposes to first associate the most confident pairs of points keeping the hardest ones for the end. Unfortunately, this method needs a finite number of points. The originality of **[2]** approach lies on the use of a triplet of images for a finer analysis of the motion. Veenman **[3]** associates a point to a trajectory according to the minimization of a cost function assessing the totality of the image.

The RANSAC (RANdom SAmple Consensus) algorithm, developed by **[6]** matches points according to a given model and is able to deal with an important amount of noise. Further versions **[7]** have considerably improved in speed.

The multi-hypothesis tracking **[8]** is based on the generation, at each step of the tracking, of possibilities that stem from the situation. Then a hypothesis tree is built up and updated during the tracking. The branches to be eliminated or refined correspond to assumptions that appear wrong or possibly true. This technique offers the opportunity to delay the decision at an extra computation cost. The same concept is used by the more recent particle filters **[9]**, for which the particle models randomly select hypotheses in relation with the model. At each step, the previous hypotheses are evaluated according to the present observations and the model is updated in consequence.

The second type of constraints is specific to image analysis. Shape, color or texture characteristics can be extracted from an interest point and its neighborhood, in order to increase its specificity. This technique has the advantage that it only requires two images for estimating the distance between points, while motion characterization requires at least three images.

3. Algorithm description

The basic idea is to use the neighborhood around the points, and not solely their descriptors, to decide of the correspondence between a model point and an observed point in the image. The neighborhood relationships are modeled using a Delaunay triangulation. Indeed, with the quality criterion of the minimum angle, the Delaunay triangulation has no flat average triangle, which is adequate for representing the notion of proximity. The triangulation is performed on the points of the observed image rather than on the model ones. Two factors have influenced this decision. First of all, the number of points of the observed image is always inferior or equal to the model one, so that the triangulation is faster. Second, triangulation can be used to update the motion of non-matched points based on their matched neighbors.

In order to exploit the neighborhood relationships jointly with the visual information, we build up a probabilistic algorithm that stems from an emerging technique: the discriminative random fields **[10]**. We compare the *n* model points $\{x_{t-1}^i\}, \quad i \in 1...n$, from the image *t-1*, to the *m* sites $\{x_t^j\}, \quad j \in 1...m$ of the observed image *t*. The algorithm operates in two steps. First, the *matching potential*s *PA₀* are initialized. This probability evaluates the possibility to associate a point to a site using only the visual descriptors:

$$P_0(x_t^j, x_{t-1}^i) = sim(desc(x_t^j), desc(x_{t-1}^i)) \quad 0 \le P_0(x_t^j, x_{t-1}^i) \le 1$$

where *sim(desc1,desc2)* is the comparison metric of two descriptors *desc1* and *desc2*, and *desc(x)* the function giving the descriptor of the location *x*. At the end of this step, we have defined, for each site of the observed image, a set of *k* (*k* ≥ *0*) candidates for the matching. In order to limit the computation, only the best three candidates are kept (*k* ≤ *3*). In a second step, we iteratively compute for each candidate point, the *matching potentials* PA_i, using the *interaction potential*s PI_i. This value estimates the probability of a matching hypothesis (i.e. the matching of a candidate point to a site) in relation to the credibility of the surrounding configuration. An association is no longer considered in an isolated manner but jointly with the matching of the neighbors. To quantify the probability of a configuration implies to evaluate the probability of the neighborhood. This task is realized according to four parameters:

- The *matching potentials* $PA_{i-1}(x_t^j, x_{t-1}^i)$ and $PA_{i-1}(x_t^l, x_{t-1}^k)$ of the two points x_{t-1}^i and x_{t-1}^k, for their respective sites x_t^j and x_t^l.
- The angle $\left(\overrightarrow{x_t^j x_{t-1}^i}, \overrightarrow{x_t^l x_{t-1}^k}\right)$ between the two vectors $\overrightarrow{x_t^j x_{t-1}^i}$ and $\overrightarrow{x_t^l x_{t-1}^k}$ built from the two points and their associates in the observed image. This value lies between 0 and 180 degrees. This angle is further converted into a likelihood P_{angle}, an angle of 180 degrees being considered as impossible, and a 0 degree angle leading to an absolute confidence in the matching.
- The difference of the Euclidian distance between the two vectors $\overrightarrow{x_t^j x_{t-1}^i}$ and $\overrightarrow{x_t^l x_{t-1}^k}$:

$$d3(x_t^j, x_{t-1}^i, x_t^l, x_{t-1}^k) = \max\left(d1(x_t^j, x_{t-1}^i, x_t^l, x_{t-1}^k), d2(x_t^j, x_{t-1}^i, x_t^l, x_{t-1}^k)\right)$$

with *d1* and *d2* the following distances:

$$d1(x_t^j, x_{t-1}^i, x_t^l, x_{t-1}^k) = 1 - \left| d(x_t^j, x_{t-1}^i) - d(x_t^l, x_{t-1}^k) \right| / 10$$

$$d2(x_t^j, x_{t-1}^i, x_t^l, x_{t-1}^k) = \frac{\min\left(d(x_t^j, x_{t-1}^i), d(x_t^l, x_{t-1}^k)\right)}{\max\left(d(x_t^j, x_{t-1}^i), d(x_t^l, x_{t-1}^k)\right)}$$

and *d* the euclidian distance between two points. The distance d3 is a combination between *d1* yielding good results for a small *d*, and *d2* being accurate in the case of large values of *d*

The importance of those parameters could be adjusted with a system of weights, but the saliency of these four factors varies according to the motion regularity of the tracked object, which itself depends on the kind of video studied. As our system aims at being generic, all video genres should be expected. Thus, we have set up equivalent weights for each of the parameters. Their mean offers a satisfying assessment for the probability of a neighborhood, except when $x_{t-1}^{i} = x_{t-1}^{k}$ which is an impossible configuration. Indeed, the same point could be affected to two sites. In this case, the likelihood is null. More formally, we have:

$$PI_i\left(x_t^j, x_{t-1}^i \middle| x_t^k, x_{t-1}^{kl}\right) = \begin{cases} \left(PA_{i-1}(x_t^j, x_{t-1}^i) + PA_{i-1}(x_t^l, x_{t-1}^k) + P_{angle} + d3\right)/4 \\ 0 \quad si \quad x_{t-1}^i = x_{t-1}^k \end{cases}$$

In order to evaluate the global likelihood of a configuration, for a fixed site x_t^j and a matching candidate x_{t-1}^i, the interaction probabilities for all the neighbors, and for each possible surrounding configuration of a candidate should be combined.. Actually, if, for each site, we could have up to three possible candidates, it will result a multitude of possible neighborhoods. Two formulations are considered:

$$PA_i\left(x_t^j, x_{t-1}^i \middle| x_t^k, x_{t-1}^{kl}\right) = \sum_l \left[\max_k \left(PI_i\left(x_t^j, x_{t-1}^i \middle| x_t^k, x_{t-1}^{kl}\right)\right)\right]$$

$$PA_i\left(x_t^j, x_{t-1}^i \middle| x_t^k, x_{t-1}^{kl}\right) = \sum_l \left[\prod_k PI_i\left(x_t^j, x_{t-1}^i \middle| x_t^k, x_{t-1}^{kl}\right)\right]$$

In order to respect the uniqueness rule (one point could only be associated to one site) the configurations containing twice the same point are avoided. The first possibility selects the most probable configuration whereas the second one considers the information from all the configurations in an equivalent manner. For practical purposes, it frequently exists a predominant configuration for many minor cases. Hence, we preferred to implement and experiment the first formulation.

So, the *matching potentials PA_i* are iteratively recomputed until the end of the algorithm. Two stopping criterions could be figured out:

- Continue the algorithm till, for each site, a matching assumption is retained. A matching hypothesis could be considered as validated, if, a candidate point has a probability higher that a given threshold *s2*, or if the likelihood of all the candidate points is lower that a threshold *s3*.
- Execute *n* iterations, and then, affect to each site the point having the highest matching probability, if this last is superior to a user fixed threshold *s1*. We can notice that with *n=0*, no iterations are performed and only the visual descriptors are considered. Setting *n* high will give a better estimation of the spatial relationships but improves the risk of neglecting the visual information. So, for the rest of this paper, this criterion will be chosen, with *n=1*.

4. Experimental results

In order to evaluate our algorithm, we have chosen to compare the performances for two different versions of our tracking system **[1]**. The first one includes our matching algorithm and is using the Delaunay triangulation in order to update the points of the model according to their neighbors. The second one contains a basic matching algorithm only based on the visual descriptors and the points are updated according to the global object motion. In both cases, the remaining ambiguities are solved by trusting the point with the highest value (respectively the highest probability or the smallest distance between the descriptors). The points are extracted with a Harris-Laplace detector **[11]**, enriched with color moments **[12]** computed on a circular region centered on the point, of a size proportional to the detected scale. The descriptors are compared using the Mahanalobis distance. The trackers are evaluated according to the comparison between the bounding box *A* returned by the tracking system and a hand-labeled ground truth bounding box *B*. The following classical formula is used:

$$\mathrm{d(A,B)} = \frac{A \cap B}{A \cup B}$$

The description of the various video used and their results using the two trackers are displayed in table 1. Except for the “cognac” sequence, our algorithm leads to better tracking results, and the object is never lost. This confirms that our tracker is robust for a greater variety of video genres.

5. Conclusion

In this article, we have presented a probabilistic matching algorithm satisfying the conditions of a generic tracking system where no specific model has to be defined. The experimental results show good performances for object tracking. Though it was developed for an object tracking application, it could also easily be adapted to other kinds of applications.

The current version only uses the previous image. It will be interesting to investigate a wider temporal use of the images, in order to maintain smooth trajectories.

Table 1: comparison of a tracker using our matching algorithm, and a basic tracker. For a given frame, the number displayed is the average performance over all the previous frames. Best results are highlighted in yellow.

Video Name	Object Size	Difficulties	Description	Frame	Basic Tracker	Improved Tracker
Fashion	Big	None	woman turning back	30	89,57%	88,77%
				60	79,17%	78,56%
				90	77,57%	76,15%
				120	78,4%	78,83%
Soccer	Small	None	Football player tracking	30	70,62%	81,5%
				70	72,56%	74,04%
Cooking	Medium	scale change, cluttered background	Marmite tracking with camera movement	30	90,48%	93,26%
				60	75,78%	81,9%
Cognac	Small	Occlusions, fast & irregular motion, cluttered background	Cognac bottle tracking	15	72,43%	54,79%
				30	53,21%	27,35%
Jellyfish	Medium	low contrast, fast object change	jellyfish swimming	15	65,07%	76,39%
				30	52,73%	69,61%
Frying pan	Medium	cluttered background	Cook showing a frying pan	25	92,17%	85,83%
				50	81,89%	79,89%
				75	79,89%	79,49%
Bottle	Small	Occlusions, cluttered background	bottle passing from hand to hand	20	96,75%	96,75%
				40	73,7%	70,53%
				60	62,35%	58,21%

Combining this algorithm with multi-hypothesis tracking for instance could further improve the quality of the tracking.

6. Acknowledgements

This research was supported by the European Project PorTiVity IST-027098 and by Institut Eurécom's industrial members

7. References

[1] R. Trichet and B. Mérialdo, “Generic Object Tracking for Fast Video Annotation”, VISAPP, Barcelona, Spain, 2007.

[2] D. Chetverikov and J. Verestoy, “Tracking feature points: a new algorithm”, *Proc. Int. Conf. on Pattern Recognition*, 1436--1438, 1998.

[3] C. J. Veenman, M. J. T. Reinders, and E. Backer, “Resolving motion correspondence for densely moving points”, *IEEE Trans. on PAMI*, vol. 23, 1: 54-72, Jan 2001.

[4] I.K. Sethi, and R. Jain, “Finding Trajectories of Feature Points in a Monocular Image Sequence”, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, pp. 56-73, Jan 1987.

[5] K. Rangarajan, Shah M., “Establishing motion correspondence”, *CVGIP: Image Understanding*, 54:56-73, 1991.

[6] M. A. Fischler and R. C. Bolles, “Random sample consensus: A paradigm for model fitting with application to image analysis and automated cartography”. *Comm.* ACM 24 (6), pp 381-395, 1981.

[7] O. Chum, J. Matas, “Matching with PROSAC - progressive sample consensus”**,** CVPR 2005, Vol. 1, 20-25, pp 220-226 vol. 1, June 2005.

[8] D. B. Reid, “An algorithm for tracking multiple targets”, *IEEE Transactions on Automatic Control*, AC-24(6):843-854, December 1979.

[9] Isard M. and MacCormick J., “BraMBLe: A Bayesian Multiple-Blob Tracker” Proc Int. Conf. Computer Vision, vol. 2, 34-41, 2001.

[10] S. Kumar and M. Hebert, “Discriminative Fields for Modeling Spatial Dependencies in Natural Images**”**, *NIPS 16*, 2004.

[11] K. Mikolajczyk, C. Schmid, « Indexation à l'aide de points d'intérêt invariants à l'échelle » *Journées ORASIS GDR-PRC Communication Homme-Machine.*, May 2001.

[12] F. Mindru, T. Tuytelaars, L. Van Gool, “Moment Invariants for Recognition under Changing Viewpoint and Illumination”, *Theo Moons,ACM*, Jul.2003.