Probabilistic People Tracking for Occlusion Handling

Abstract

This work presents a novel people tracking approach, able to cope with frequent shape changes and large occlusions. In particular, the tracks are described by means of probabilistic masks and appearance models. Occlusions due to other tracks, or due to background objects and false occlusions are discriminated. The classification is exploited in a selective model update. The tracking system is general enough to be applied with any motion segmentation module, it can track people interacting each other and it maintains the pixel assignment to track even with large occlusions. At the same time, the update model is very reactive, so as to cope with sudden body motion and silhouette's shape changes. Due to its robustness, it has been used in many experiments of people behavior control in indoor situations.

1. Introduction

Tracking is one of the most critical step in processes of people motion capture, people behavior control and indoor video surveillance. The tracking module should be very efficient, in order not to affect the speed of the whole process and, at the same time, it should be very reactive, to adjust the model to sudden changes of silhouette's shape and very robust to occlusions due to other people or objects present in the environment.

In literature, many works address people tracking with occlusion handling, but only few of them manage the pixel assignment during the occlusion, in order to keep the knowledge of the track while the occlusion occurs. The works [1] and [2] address occlusions between tracks. In [1] classes of similar color defined with EM algorithm are defined to segment people, tracked frame by frame with a maximum a posteriori probability approach . In [2] pixels assignment is guided by color histograms that model the a priori probability and again a Bayes rule is used to form the posterior probability: thus a visibility index is built to provide information on the depth ordering of tracks.

The authors of [3] exploit a stereo vision system to deal with the occlusions and to correctly segment each person in the scene. Furthermore, similar to others [4,5], they use a mask and an appearance template for each track to resolve the temporal tracking. In [6] the tracking system is realized with the fusion of three co-operating parts: an Active Shape Tracker, a Region Tracker and a Head Detector. The Region Tracker exploits the other two modules to solve occlusions.

Following the related work [4,5,6], we describe the tracks with probabilistic mask and appearance models, enriched with a selective update function, specifically defined to cope with not rigid body motions and frequent shape changes. The defined tracking does not rely on a specific segmentation module: nevertheless, in order to overcame the unavoidable under-and over- segmentation errors, we introduce the definition of *macro-object*, to merge all segmented objects that could be potentially assigned to more than one track and define a probabilistic pixel to track assignment function.

However, the novelty of this work is the special focus on occlusions. Occlusions are classified as *track occlusions* (due to other people or moving objects), *background object occlusions* (due to fixed objects in the indoor environment), and *apparent or false occlusions* (due to sudden shape changes). The classification is exploited to model a non occlusion probability function, used as a priori probability to have a track and to provide a selective update of the probabilistic mask and the appearance model.

In the next section the detail of the defined tracking are described, while the last section shows some results carried out in real situations of indoor video surveillance.

2. The tracking system

The defined tracking is totally independent from previous steps of object segmentation. Assuming the acquisition from a single fixed camera, let us assume to have, for each frame *t*, a model of background and a V^t set of *Visual Objects*: $V^t = \left\{VO_1^t, \dots, VO_n^t\right\}, VO_j^t = \left\{BB_j, M_j, I_j, c_j\right\}$.

Each Visual Object VO_j^t is a set of connected points detected as in motion by the segmentation algorithm and described with a set of features: the bounding box BB_j, the blob mask M_j, the Visual Object's color template I_j and the centroid c_j. During the tracking execution, we compute a set of tracks T^t at each frame *t*, that represents the knowledge of the objects present in the scene: $T^t = \left\{T_1^t..T_m^t\right\}$ with $T_k^t = \left\{BB_k, AI_k, PM_k, PNO_k, c_k, \vec{e}_k\right\}$, where BB_k is the bounding box; AI_k is the *Appearance Image*, i.e. the estimated aspect (in RGB space) of the track, representing the "memory" of the object previously tracked; PM_k is the *probability mask*: each value of $PM_k(x)$ defines the probability that the point *x* belongs to the track T_k; PNO_k is the *probability of not occlusion* associated with the track, that is the probability that the

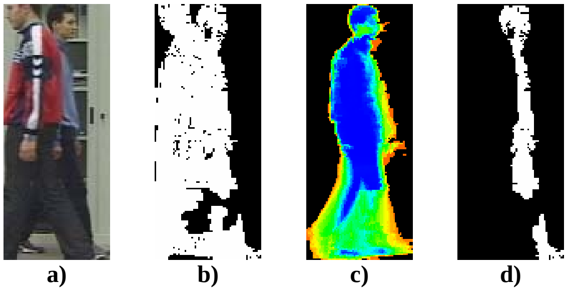

Figure 1: Depth order pixel assignment.

track k is not occluded by other tracks; $\vec{e}_k$ is the motion vector estimated for the next frame[1].

The tracking process is divided in the following steps: 1) *VO to Track mapping*, 2) *Track position refinement*, 3) M*O-pixel to track assignment*, 4) *Occlusion classification*, 5) *Selective track update*.

2.1. VO to Track mapping

As most of the tracking approaches, the process starts with the construction of a Boolean correspondence matrix C between the V^t and T^{t-1} sets. The element $C_{k,j}$ is set to one if the VO_j^t can be associated to the track T_k^{t-1}. The association is established if the track (shifted into its estimated position by means of the vector $\vec{e}_k$) and the *VO* can be roughly overlapped, or, in other words, if they have a "low" distance. It is computed as a Bounding Box Distance (*BBd*) as in the following equation:

$$BBd\left(VO_j, T_k\right) = \min_{x_k \in BB_k, y_j \in BB_j} \left(\min\left(\left\| c_j, x_k + \vec{e}_k \right\|, \left\| c_k + \vec{e}_k, y_j \right\| \right) \right). \quad (1)$$

In the matrix C five different cases can arise: 1) a track is not associated to any *VO*: the track is missed, 2) a *VO* is not associated to any *T*: a new object is entered into the scene and a new track is generated, 3) a *T* is associated to more than one *VO*, 4) many tracks are associated to the same *VO*, 5) many tracks are associated to many *VOs*. In the last three cases, the tracking system has to cope with problems of track split, track merge or track overlap. This work is specially oriented to solve these last cases, very frequent in indoor environments with people interactions. To this aim, we define the concept of *Macro-Object* (*MO*) as the union of the *VOs* associated to the same tracks. Initially a *MO* is created for each *VO*, then couples of *MOs* that have at least a track in common are merged. This step is iterated until each track is associated to a single *MO* only.

Thus, hereinafter, the *MOs* substitute the segmented *VOs* in the tracking module and allow to get rid of the problem of managing the many-to-many correspondence case. In general, in fact, a single segmented *VO*, generated from overlapped people, has points that should be assigned to different tracks, or some disjoint *VOs* (due to segmentation errors) should be associated to the same track.

2.2. Track position refinement

In this phase the estimated position of each track is refined with the displacement $\vec{\delta} = (\delta_x, \delta_y)$ that maximizes a fitting function P_{FIT}. This is iterated for all the tracks associated with a MO, with a order proportional to their probability of not occlusion. $\vec{\delta}$ is initialized with the value $\vec{e}_k$ and searched with a gradient descent approach:

$$P_{FIT}(T_k, \vec{\delta}) = \frac{\sum_{x \in MO} P_{APP}(I(x - \vec{\delta}), AI_k(x)) \cdot PM_k(x)}{\sum_{x \in T_k} PM_k(x)} \quad (2)$$

where $P_{APP}\left(RGB_i, RGB_j\right)$ measures the correspondence between the actual RGB color of the point in *MO* and the appearance model of the track. As in [5], we use a spherical Gaussian to approximate the pixel distribution around the mean stored by the model

$$P_{APP}(RGB_i, RGB_j) = (2\pi\sigma^2)^{-3/2} e^{-\frac{\left\| RGB_i - RGB_j \right\|^2}{2\sigma^2}} \quad (3)$$

where the norm used is the Euclidean distance in RGB space. The alignment is $\vec{\delta}_{BF}(T_k) = \arg\max_{\vec{\delta}}\left(P_{FIT}(T_k, \vec{\delta})\right)$. After each fitting computation, the points of the *MO* matching a track point with high P_{APP} are removed and not considered for the following tracks. Fig. 1 shows a single *MO* (two overlapped people) and two tracks: Fig. 1.d shows the *MO*'s points not assigned to the first track and than assigned to the second; Fig.1.c is the probability mask of the second track. To cope with large occlusions we refined the model by rewriting the Eq. 2 as:

$$P_{FIT}(T_k, \vec{\delta}_{BF}) = Likelihood \cdot Confidence =$$

$$\frac{\sum_{x \in MO} P_{APP}(I(x - \vec{\delta}_{BF}), AI_k(x)) \cdot PM_k(x)}{\sum_{x \in MO} PM_k(x)} \cdot \frac{\sum_{x \in MO} PM_k(x)}{\sum_{x \in T_k} PM_k(x)} \quad (4)$$

The first term is a measure of how similar are the corresponding pixels of the MO and the track; the second term is the percentage of track points, weighted with their probability, that are visible on the current frame and belonging to the *MO*. Accordingly, when the product of *Likelihood* and *Confidence* is low the track is considered totally occluded (and $\vec{\delta}_{BF}$ is not used). Instead, immediately after an occlusion we want to react without waiting for the best fit value to return to higher values: therefore if the Confidence value is growing with respect to the previous frame, the estimated position is updated anyway. For

[1] Hereinafter, in order to manage a point either of the VO or of the Track, we will write -improperly- $x \in VO, x \in T$, meaning that $x \in BB$ and either the VO's mask $M(x)$ or the probability mask $PM(x)$ of T in the point x is not zero.

instance, the backmost track in Fig. 1 has a high *Likelihood* (0.76) and a low, but growing, *Confidence* (0.32) and thus we accept the position refinement.

2.3. Pixel to track assignment

All *MO* points must be assigned to a track. If a *MO* is in correspondence with a single track, the assignment is straightforward. Instead, in presence of track occlusions, when two or more tracks contend points of the same *MO*, we exploit a Bayesian function to solve the assignment:

$$P(T_k|x) = P(x|T_k)P(T_k) \Big/ \sum_{i=1}^{m} P(x|T_i)\cdot P(T_i)\,. \qquad (5)$$

The conditional probability is the product of two terms: $P(x|T_k) = P_{APP}(x)\cdot PM_k(x)$. It takes into account the difference between the colors of the actual pixel and the track appearance one, weighted by the probability that the point belongs to the track. In order to cope with track-based occlusion, the $P(T_k)$ is suitably modeled as the *a priori probability* of seeing T_k, defined as a probability of not occlusion (see section 2.5 for details). Each point will be assigned to the track that maximizes $P(T_k|x)$ and the set of point assigned to the track T_k is named A_k.

2.4. Occlusion classification

Due to occlusions or shape changes, some points of the tracks remain without any correspondence with a *MO* point. Other proposed techniques that exploit probabilistic appearance model without coping with occlusions explicitly, use directly the set of assigned points (A_k) to guide the update process [5]; the mask probability in the points $x \in \{T - A_k\}$ decreases while in $x \in \{A_k\}$ is reinforced. In this work the adaptive update function is enriched by the knowledge of occlusion regions.

The set of not visible points $NV^t = \{T^t - A_k\}$ are the candidate points for occlusion regions: in general they are the points of the tracks that are not visible anymore at the frame *t*. After a labeling step, a set of *not visible regions* (of connected points of *MO*) is created, neglecting sparse points or too small regions. Non visible regions can be classified in three classes:

1) apparent occlusions R_{AO}: regions not visible because of shape changes, silhouette's motion, or self-occlusions;

2) *track-based occlusions* R_{TO}: due to overlap of another track closer to the camera that "wins" the assignment process of pixels;

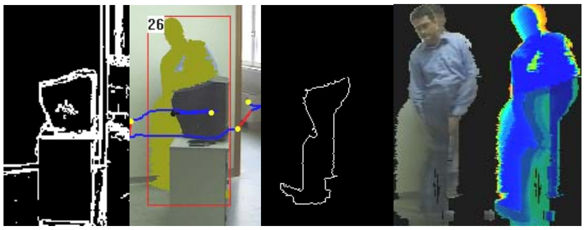

Figure 2: Edge pixel classification and selective update.

3) *background object-based occlusions* R_{BOO}: due to (still) objects, included in the background model, but ahead of the track.

The presence of occlusions can be detected with the *Confidence* value of Eq. 4 decreasing above an alerting value, since in case of occlusion the track's shape changes considerably. In case of actual occlusions (classes 2 and 3), the track model should not be updated since we do not want to lose the memory of the people appearance. Nevertheless, if the *Confidence* decreases due to a sudden shape motion (apparent occlusion), not updating the track would create an error. The solution is a *selective update* according to the region classification. The R_{TO} regions are distinguishable from previous steps: they are composed by the points shared between track T_k and other tracks T_i but not assigned to T_k. In order to distinguish between cases 1) and 3) without the knowledge of the background objects, an approximated technique based on an edge analysis is proposed. The edges of the not visible regions are extracted. At the same time, the edges of the background model are available. The edge pixels that touch visible pixels of the track are classified as *bounding pixels* while the others are said *not bounding pixels*. If the majority of the bounding pixels match background edges, we can suppose that an object hides a part of the track, and the region is labeled as R_{BOO}, otherwise as R_{AO}. In Fig. 2 an example is shown: a part of a person is occluded; this non visible region is classified as R_{BOO} and the correspondent probabilistic and appearance model is neither reinforced neither weakened.

2.5. Selective Track update

As the final step, the probability mask, the appearance mask and the probability of not occlusion are updated with adaptive functions. In particular, $\forall x \in T^t$

$$PM^t(x) = \begin{cases} \lambda PM^{t-1}(x) + (1-\lambda) & x \in A_K \\ PM^{t-1}(x) & (x \in R_{TO}) \vee (x \in R_{BBO}) \\ \lambda PM^{t-1}(x) & otherwise \end{cases} \qquad (6)$$

$$AI^t(x) = \begin{cases} \lambda AI^{t-1}(x) + (1-\lambda) I^t(x) & x \in A_K \\ AI^{t-1}(x) & otherwise \end{cases} \qquad (7)$$

When the track is generated $P_M{}^t(x)$ is initialized to an intermediate value (0.4 when λ=0.9) while the appearance image is initialized to the image $I^t(x)$. Defining $Po^t_{i\to k}$ as the probability that track T_i occludes T_k, the not-occlusion probability used in the Bayes rule is computed as a value proportional to the number $a_{i\to k}$ of shared points assigned to T_i and not to T_k. In particular:

$$PNO^t(T_k) = 1 - \max_{i=1..m}(Po^t_{i\to k})\,, \qquad (8)$$

calling $\beta_{i\to k} = \frac{a_{i\to k} + a_{k\to i}}{\left\| A_i \right\|}$, $Po^t_{i\to k}$ update model is:

$$Po^t_{i\to k} = \begin{cases} 0 & \beta_{i\to k} < \vartheta_{occl} \\ (1-\beta_{i\to k})Po^{t-1}_{i\to k} & a_{i\to k} = 0 \\ (1-\beta_{i\to k})Po^{t-1}_{i\to k} + \beta_{i\to k} e^{-\frac{a_{k\to i}}{a_{i\to k}}} & a_{i\to k} \neq 0 \end{cases} . \quad (9)$$

Finally, the motion vector $\vec{e}_k$ is estimated according to a constant speed assumption, but enforced by a segmented trajectory schema. Starting from a reference initial position, a certain number of successive motion vectors are linearly interpolated by finding the least squares solution. The solution vector is the motion estimation. In order to check if the interpolation describes correctly the last vector, we evaluate the ratio between the two eigenvalues of the principal direction computation and also if the angle or modulus has changed much from the first value. If the solution fails these checks, a new reference position is created and a new direction can be searched. In this way, an adaptive finite window is used to infer the future motion of the object, able to cope with change of direction in a robust way.

3. Results and conclusions

The system has been devised for a project of Domotic to control the people behavior in the house and detect dangerous situations, as people falling and lying on the floor motionless for a long time [7].

This complex but complete process has been tested over days of indoor video surveillance in two rooms equipped with fixed camera, with some actors and indoor furniture. Moreover it has been tested over the videos of PETS 2002, in which people walk and interact behind a shop window. Figures 1 and 2 are examples of frames of videos V2 and V3 respectively. Fig. 3 shows as the occlusion in V1 are correctly managed. Table 1 shows the performances of the system on eight sequences: #pe is the number of people present in the scene, #fr is the number of frames, #C is the number of correct assignments and FP and FN are the number of false positives and false negatives respectively against a manual ground-truth. The former are frames in which two or more tracks are assigned to the same person, while the latter are the number of frames in which no tracks are assigned to a person. In TR3 the high error rate in FP is due to the fact two people enter together in the scene and the system has not the possibility to see them as separate objects. However, in general the tracking results very robust. Also in V3 and V4 experiments, where large occlusions due to furniture and track overlaps occur, a percentage of about 88% of correct assignment is reached.

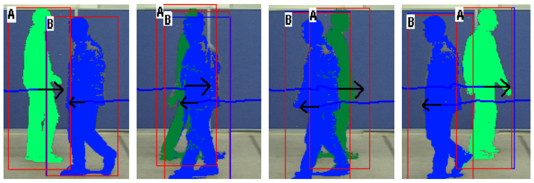

Figure 3: Correct track-based occlusion solve.

Video	#pe	#fr	#C	FP	FN
V1	2	1596	2346	0	0
V2	2	1753	1947	188	0
V3	3	1331	687	58	19
V4	3	1270	2698	25	329
PETS 2002 TR2	9	1471	1372	70	29
PETS 2002 TR3	10	1295	882	343	70
PETS 2002 TE1	5	653	538	0	115
PETS 2002 TE2	9	1753	1345	120	288

Table 1: System performances.

The tracking approach is not computationally intensive. In our experiment, the indoor video surveillance is able to process about fifteen frames per second on a standard PC including an initial visual object segmentation module with background suppression, the shadow removal module [8] and a further people posture classification process [7]. The edge-based method is able, on average, to correctly classify the 85% of non visible regions. This approach could be further refined but it is enough precise to allow a good reactivity to silhouette's shape change and, at the same time, a good memory of the appearance model also when a person remains occluded by static objects for a long time.

Therefore the proposed tracking module is a general scheme that exploits probabilistic function and appearance model to keep the knowledge of tracked objects even if they are partially hidden. The robustness and the reactivity is based on a selective update process, that manages differently visible pixels, pixels occluded by static or moving regions and pixels that are not visible anymore, due to shape changes self-occlusions or sudden silhouette's motion.

4. References

[1] S. Khan, M. Shah, "Tracking People in Presence of Occlusion", Asian Conf. on Computer Vision, Taiwan, Jan 2000.

[2] S.J. McKenna, et. al. "Tracking interacting people", IEEE Int. Conf. on Automatic Face and Gesture Recognition, France, Mar 2000, pp. 348-353.

[3] D. Beymer, K. Konolige, "Real-time tracking of multiple people using continuous detection", Int. Conf. on Computer Vision, 1999.

[4] A. J. Lipton, et al. "Moving target classification and tracking from real-time video" IEEE Image Understanding Workshop, 1998, pp. 129-136.

[5] A. Senior, et al. "Tracking people with probabilistic appearance models", Int. Workshop on Perf. Eval. of Tracking and Surveillance Systems, 2002.

[6] N.T. Siebel, S. Maybank, "Fusion of Multiple Tracking Algorithms for Robust People Tracking", 7th European Conf. on Computer Vision, Denmark, May 2002, vol. IV, pp. 373-387.

[7] -omitted for blind review- ACM Multimedia 2003.

[8] -omitted for blind review- PAMI 2003.