Multi-Camera People Tracking with a Probabilistic Occupancy Map

François Fleuret Jérôme Berclaz Richard Lengagne
Pascal Fua
École Polytechnique Fédérale de Lausanne
Lausanne, Switzerland
{francois.fleuret,jerome.berclaz,pascal.fua}@epfl.ch
richard.lengagne@ge.com

Technical Report EPFL/CVLAB2006.07

Abstract

Given three or four synchronized videos taken at eye level and from different angles, we show that we can effectively combine a generative model with dynamic programming to accurately follow up to six individuals across thousands of frames in spite of significant occlusions and lighting changes. In addition, we also derive metrically accurate trajectories for each one of them.

Our contribution is twofold. First, we demonstrate that our generative model can effectively handle occlusions in each time frame independently, even when the only data available comes from the output of a simple background subtraction algorithm and when the number of individuals is unknown *a priori*. Second, we show that multi-person tracking can be reliably achieved by processing individual trajectories separately over long sequences, provided that a reasonable heuristic is used to rank these individuals and avoid confusing them with one another.

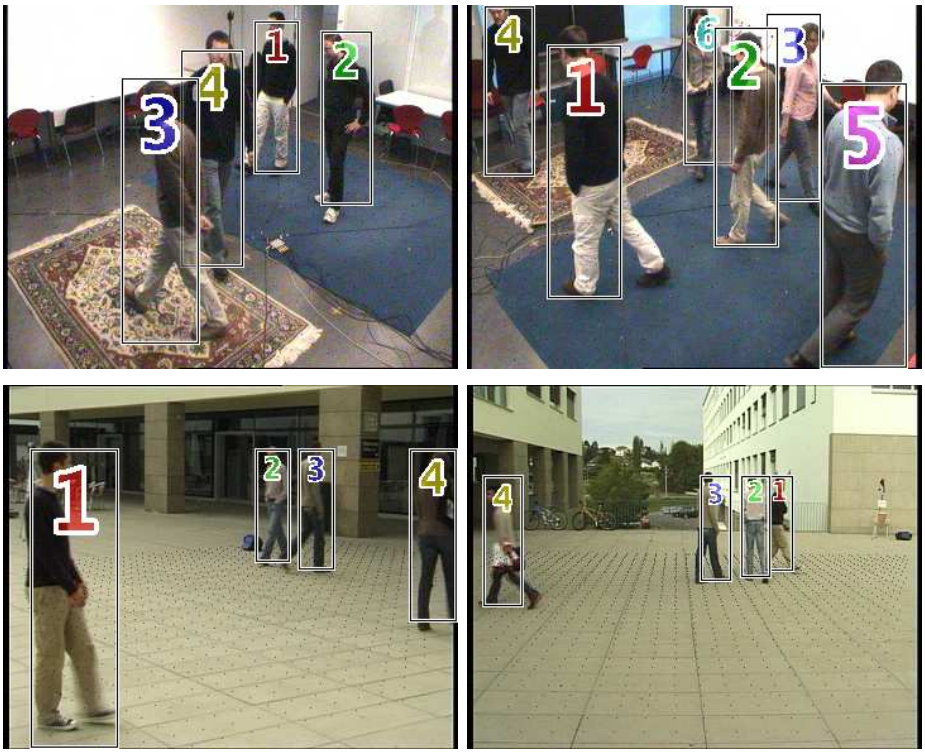


Figure 1: Images from two indoor and two outdoor multi-camera video sequences we use for our experiments. At each time step, we draw a box around people we detect and assign to them an Id number that follows them throughout the sequence.

1 Introduction

In this paper, we address the problem of keeping track of people who occlude each other using a small number of synchronized videos such as those depicted by Fig. 1, which were taken at head level and from very different angles. This is important because this kind of set-up is very common for applications such as video-surveillance in public places.

To this end, we have developed a mathematical framework that allows us to combine a robust approach to estimating the probabilities of occupancy of the ground plane at individual time steps with dynamic programming to track people over time. This results in a fully automated system that can track up to 6 people in a room for several minutes using only four cameras, without producing any false positives or false negatives in spite of severe occlusions and lighting variations. As shown in Fig. 2, our system also provides location estimates that are accurate to within a few tens of centimeters and there is no measurable performance decrease if as many as 20% of the images are lost, and only a small one if 30% are. This involves the two following algorithmic steps:

1. We estimate the probabilities of occupancy of the ground plane given the binary image obtained from the input images via background subtraction. At this stage, the algorithm only takes into account images acquired at the *same time*. Its basic ingredient is a generative model that represents humans as simple rectangles that it uses to create synthetic ideal images we would observe if people were at given locations. Under this model of the image given the true state, we approximate the occupancy probabilities as the marginals of a product law minimizing the Kullback-Leibler divergence from the "true" conditional posterior distribution. This leads to a method we refer to as a *Fixed Point Probability Field* (FPPF) algorithm because it finds the marginal probabilities of the product law as the fixed point of a large system of equations.

2. We then combine these probabilities with a color and a motion model and use a Viterbi algorithm to accurately follow individuals across thousands of frames. To avoid the combinatorial explosion that would result from explicitly dealing with the joint posterior distribution of the locations of individuals in each frame over a fine discretization, we use a greedy approach. We process trajectories individually over sequences that are long enough so that using a reasonable heuristic to choose the order in which they are processed is sufficient to avoid confusing people with each other.

In contrast to most state-of-the-art algorithms that recursively update estimates from frame to frame and may therefore fail catastrophically if difficult conditions persist over several consecutive frames, our algorithm can handle such situations, since it computes global optima of scores summed over many frames. This is what gives it the robustness that Fig. 2 demonstrates.

In short, we combine a mathematically well-founded generative model that works in each frame individually with a simple approach to global optimization. This yields excellent performance using basic color and motion models that could be further improved. Our contribution is therefore twofold. First, we demonstrate that a generative model can effectively handle occlusions at each time frame independently even when the input data is of very poor quality, and therefore easy to obtain. Second, we show that multi-person tracking can be reliably achieved by processing individual trajectories separately over long sequences.

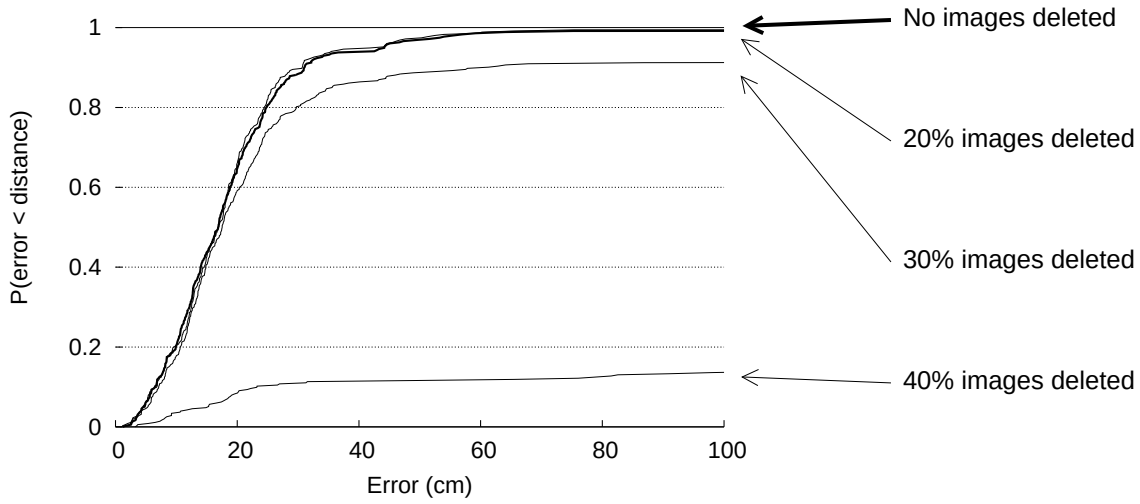


Figure 2: Cumulative distributions of the position estimate error on a 3800-frame sequence. See §6.3.1 for details.

In the remainder of the paper, we first briefly review related works. We then formulate our problem as one of estimating the most probable state of a hidden Markov process using dynamic programming and show how to compute the individual marginals. Finally, we present our results on several very long sequences.

2 Related Work

State-of-the-art methods can be divided into monocular and multi-view approaches that we briefly review in this section.

2.1 Monocular approaches

Monocular approaches rely on the input of a single camera to perform tracking. This method provides a simple and easy-to-deploy setup, but it must compensate for the lack of 3D information in a single camera view.

2.1.1 Blob-based methods

Many algorithms rely on binary blobs extracted from single video [8, 4, 7]. They combine shape analysis and tracking to locate people and maintain appearance models in order to track them even in presence of occlusions. The *BraMBLe* system[9], for example, is a multi-blob tracker that generates a blob-likelihood based on a known background model and appearance models of the tracked people. It then uses a particle filter to implement the tracking for an unknown number of people.

Approaches that track in a single view prior to computing correspondences across views extend this approach to multi camera setups. However, we view them as falling into the same category because they do not simultaneously exploit the information from multiple views. In [11], the limits of the field of view of each camera are computed in every other camera from motion information. When a person becomes visible in one camera, the system automatically searches for him in other views where he should be visible. In [3], a background/foreground segmentation is performed on calibrated images, followed by human shape extraction from foreground objects and feature point selection extraction. Feature points are tracked in a single view and the system switches to another view when the current camera no longer has a good view of the person.

2.1.2 Color-based methods

Tracking performance can be significantly increased by taking color into account. For example, in [12], the images are segmented pixel-wise into different classes, thus modeling people by continuously updated Gaussian mixtures. A standard tracking process is then performed using a Bayesian framework, which helps keep track of people even when there are occlusions. In such a case, models of persons in front keep being updated, while the system stops updating occluded ones, which may cause trouble if their appearances have changed noticeably when they reemerge.

More recently, multiple humans have been simultaneously detected and tracked in crowded scenes [1] using Monte-Carlo-based methods to estimate their number and positions. In [16], multiple people are also detected and tracked in front of complex backgrounds using mixture particle filters guided by people models learnt by boosting. In [6], multi-cue 3D object tracking is addressed by combining particle-filter based Bayesian tracking and detection using learnt spatio-temporal shapes. This approach leads to impressive results but requires shape, texture and image depth information as input. Finally [18] proposes a particle-filtering scheme that relies on MCMC optimization to handle entrances and departures. It also introduces a finer modeling of interactions between individuals as a product of pairwise potentials.

2.2 Multi-view Approaches

Despite the effectiveness of such methods, the use of multiple cameras soon becomes necessary when one wishes to accurately detect and track multiple people and compute their precise 3D locations in a complex environment. Occlusion handling is facilitated by using two sets of stereo color cameras[13]. However, in most approaches that only take a set of 2D views as input, occlusion is mainly handled by imposing temporal consistency in terms of a motion model, be it Kalman filtering or more general Markov models. As a result, these approaches may not always be able to recover if the process starts diverging.

2.2.1 Blob-based Methods

In [14], Kalman filtering is applied on 3D points obtained by fusing in a least-squares sense the image-to-world projections of points belonging to binary blobs. Similarly in [2], a Kalman filter is used to simultaneously track in 2D and 3D, and object locations are estimated through trajectory prediction during occlusion.

In [5], a best-hypothesis and a multiple-hypotheses approaches are compared to find people tracks from 3D locations obtained from foreground binary blobs extracted from multiple calibrated views.

In [17] a recursive Bayesian estimation approach is used to deal with occlusions while tracking multiple people in multi-view. The algorithm tracks objects located in the intersections of 2-D visual angles, which are extracted from silhouettes obtained from different fixed views. When occlusion ambiguities occur, multiple occlusion hypotheses are generated given predicted object states and previous hypotheses, and tested using a branch-and-merge strategy. The proposed framework is implemented using a customized particle filter to represent the distribution of object states.

2.2.2 Color-Based Methods

[15] proposes a system that segments, detects and tracks multiple people in a scene using a wide-baseline setup of up to 16 synchronized cameras. Intensity information is directly used to perform single-view pixel classification and match similarly labeled regions across views to derive 3D people locations. Occlusion analysis is performed in two ways. First, during pixel classification, the computation of prior probabilities takes occlusion into account. Second, evidence is gathered across cameras to compute a presence likelihood map on the ground plane that accounts for the visibility of each ground plane point in each view. Ground plane locations are then tracked over time using a Kalman filter.

In [10], individuals are tracked both in image planes and top view. The 2D and 3D positions of each individual are computed so as to maximize a joint probability defined as the product of a color-based appearance model and 2D and 3D motion models derived from a Kalman filter.

While our own method shares many features with these techniques, it differs in three important respects that we will highlight. First, we rely on dynamic programming to ensure greater stability in challenging situations by simultaneously handling multiple frames. Second, it relies on a discretization of the full area of interest, and is therefore able to deal with very flat distributions. Finally, our approach combines the usual color and motion models with a sophisticated estimation of the probability of occupancy.

3 Problem Formulation

Our goal is to track an *a priori* unknown number of people from a few synchronized videos taken at head level. In this section, we formulate this problem as one of finding the most probable state of a hidden Markov process given the set of images acquired at each time step, which we will refer to as a *temporal frame*. We then briefly outline the computation of the relevant probabilities using the notations summarized by Tables 1 and 2, which we also use in the following two sections to discuss in more details the actual computation of those probabilities.

3.1 Computing The Optimal Trajectories

We process the video sequences by batches of $T = 100$ frames, each of which includes C images, and we compute the most likely trajectory for each individual. To achieve consistency over successive batches, we only keep the result on the first ten frames and slide our temporal window. This is illustrated on Fig. 3.

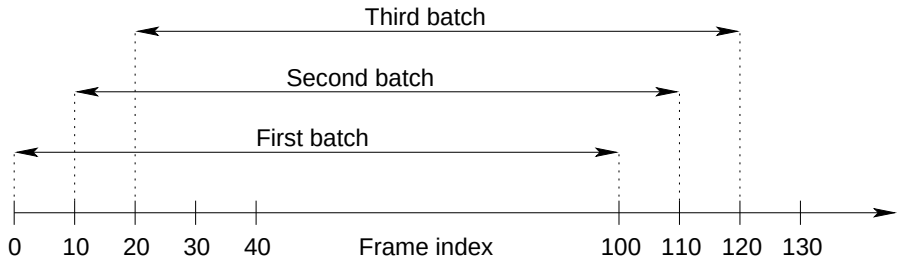


Figure 3: Video sequences are processed by batch of 100 frames. Only the first 10% of the optimization result is kept, and the rest is discarded. The temporal window is then slided forward and the optimization repeated on the new window.

We discretize the visible part of the ground plane into a finite number G of regularly spaced 2–D locations, and we introduce a virtual hidden location $\mathcal{H}$ that will be used to model entrances and departures from and into the visible area. For a given batch, let $\mathbf{L}_t = (L_t^1, \ldots, L_t^{N^*})$ be the hidden stochastic processes standing for the locations of individuals, whether visible or not. The number N^* stands for the maximum allowable number of individuals in our world. It is large enough so that conditioning on the number of visible ones does not change the probability of a new individual entering the scene. The L_t^n variables therefore take discrete values in the range $\{1, \ldots, G, \mathcal{H}\}$.

Given $\mathbf{I}_t = (I_t^1, \ldots, I_t^C)$, the images acquired at time t for $1 \leq t \leq T$, our task is to find the values of the $\mathbf{L}_t$ that maximize

$$P(\mathbf{L}_1, \ldots, \mathbf{L}_T \,|\, \mathbf{I}_1, \ldots, \mathbf{I}_T). \tag{1}$$

As will be discussed in §4.1, we compute this MAP in a greedy way, processing one individual at a time, including the hidden ones who can move into the visible scene or not. For each one, the algorithm performs the computation under the constraint that no individual can be at a visible location occupied by an individual already processed.

In theory, this approach could lead to undesirable local minima, for example by connecting the trajectories of two separate people. However this does not happen often because our batches are sufficiently long. To further reduce the chances of this, we process individual trajectories in an order that depends on a reliability score so that the most reliable ones are computed first, thereby reducing the potential for confusion when processing the remaining ones. This order also ensures that if an individual remains in the hidden location, all the other people present in the hidden location will also stay there, and therefore do not need to be processed.

Our experimental results show that our method does not suffer from the usual weaknesses of greedy algorithms, such as a tendency to get caught in bad local minima. We therefore believe that it compares very favorably to stochastic optimization techniques in general and more specifically particle filtering, which usually requires careful tuning of meta-parameters.

3.2 Stochastic Modeling

We will show in §4.2 that since we process individual trajectories, the whole approach only requires us to define a valid motion model $P(L_{t+1}^n \,|\, L_t^n = k)$ and a sound appearance model $P(\mathbf{I}_t \,|\, L_t^n = k)$.

The motion model $P(L_{t+1}^n \,|\, L_t^n = k)$, which will be introduced in section §4.3, is a distribution into a disc of limited radius and center k, which corresponds to a loose bound on the maximum speed of a walking human. Entrance into the scene and departure from it are naturally modeled thanks to the hidden location $\mathcal{H}$, for which we extend the motion model. The probabilities to enter and to leave are similar to the transition probabilities between different ground plane locations.

In section §4.4, we will show that the appearance model $P(\mathbf{I}_t \,|\, L_t^n = k)$ can be decomposed into two terms. The first, described in section §4.5, is a very generic color-histogram based model for each

Table 1: Notations (deterministic quantities)

$W \times H$	image resolution.
C	number of cameras.
G	number of locations in the ground discretization ($\simeq 1000$).
T	number of frames processed in one batch ($= 100$).
t	frame index.
$I \otimes J$	intersection of images, $\forall (x,y),\ (I \otimes J)(x,y) = I(x,y)J(x,y)$.
$I \oplus J$	disjunction of images, $\forall (x,y),\ (I \oplus J)(x,y) = 1 - (1 - I(x,y))(1 - J(x,y))$.
Ψ	a pseudo-distance between images.
Q	the product law used to approximate, for a fixed t, the real posterior distribution $P(\cdot \mid \mathbf{B}_t)$.
E_Q	Expectation under $\mathbf{X} \sim Q$.
q_k	the marginal probability of Q, that is $Q(X_k = 1)$.
ϵ_k	the prior probability of presence at location i, $P(X_k = 1)$.
λ_k	is $\log \frac{1-\epsilon_k}{\epsilon_k}$, the log-ratio of the prior probability.
$\mathcal{A}_k^c$	the image composed of 1s inside a rectangle standing for the silhouette of an individual at location i seen from camera c, and 0s elsewhere.
N^*	virtual number of people, including the non-visible ones.
μ_n^c	color distribution of individual n from camera c.

Table 2: Notations (random quantities)

$\mathbf{I}_t$	images from all the cameras $\mathbf{I}_t = (I_t^1, \ldots, I_t^C)$.
$\mathbf{B}_t$	binary images generated by the background subtraction $\mathbf{B}_t = (B_t^1, \ldots, B_t^C)$.
$\mathbf{T}_t$	texture information.
A_t^c	ideal random image generated by putting rectangles $\mathcal{A}_k^c$ where $X_t^k = 1$, thus a function of $\mathbf{X}_t$.
$\overline{A}_{k,\xi}^c$	compact notation for the average synthetic image $E_Q(A^c \mid X_k = \xi)$, see Figure 6.
$\mathbf{L}_t$	vector of people locations on the ground plane or in the hidden location $\mathbf{L}_t = (L_t^1, \ldots, L_t^{N^*})$. Each of these random variables takes values into $\{1, \ldots, G, \mathcal{H}\}$, where $\mathcal{H}$ is the hidden place.
$\mathbf{L}^n$	trajectory of individual n, $\mathbf{L}^n = (L_1^n, \ldots, L_T^n)$.
$\mathbf{X}_t$	vectors of boolean random variable $(X_t^1, \ldots, X_t^G)$ standing for the occupancy of location k on the ground plane $\left(X_t^k = 1\right) \Leftrightarrow (\exists n, L_t^n = k)$.

individual. The second, described in Section §5, approximates the marginal conditional probabilities of occupancy of the ground plane given the results of a background subtraction algorithm, in all views acquired at the same time. This approximation is obtained by minimizing the Kullback-Leibler divergence between a product law and the true posterior. We show that this is equivalent to computing marginal probabilities of occupancy so that, under the product law, the images obtained by putting rectangles of human sizes at occupied locations are likely to be similar to the images actually produced by the background-subtraction. Since these marginal probabilities are computed independently at each time step, they say nothing about identity or correspondence with past frames. The appearance similarity is entirely conveyed by the color histograms, which has experimentally proved sufficient for our purposes.

4 Computation of the trajectories

In section §4.1, we break the global optimization of several people's trajectories into the estimation of optimal individual trajectories. In section §4.2, we show how this can be performed using dynamic programming. This requires a motion model given in section §4.3 and an appearance model described in §4.4, combining a color model given in section §4.5 and a sophisticated estimation of the ground plane occupancy detailed in §5.

We partition the top view into a regular grid as shown in Figures 5(c) and 6, and from the camera calibration, we define for each camera c a family of rectangular shapes $\mathcal{A}_1^c, \ldots, \mathcal{A}_G^c$ which correspond to crude human silhouettes of height 175cm and width 30cm located at every position on the grid.

4.1 Multiple trajectories

Recall that we denote by $\mathbf{L}^n = (L_1^n, \ldots, L_T^n)$ the trajectory of individual n. Given a batch of T temporal frames $\mathbf{I} = (\mathbf{I}_1, \ldots, \mathbf{I}_T)$, we want to maximize the posterior conditional probability

$$P(\mathbf{L}^1 = \mathbf{l}^1, \ldots, \mathbf{L}^{N^*} = \mathbf{l}^{N^*} \,|\, \mathbf{I}) = P(\mathbf{L}^1 = \mathbf{l}^1 \,|\, \mathbf{I}) \prod_{n=2}^{N^*} P(\mathbf{L}^n = \mathbf{l}^n \,|\, \mathbf{I}, \mathbf{L}^1 = \mathbf{l}^1, \ldots, \mathbf{L}^{n-1} = \mathbf{l}^{n-1}). \quad (2)$$

Simultaneous optimization of all the L^is would be intractable. Instead, we optimize one trajectory after the other, which amounts to looking for

$$\hat{\mathbf{l}}^1 = \arg\max_l P(\mathbf{L}^1 = l \,|\, \mathbf{I}), \quad (3)$$

$$\hat{\mathbf{l}}^2 = \arg\max_l P(\mathbf{L}^2 = l \,|\, \mathbf{I}, \mathbf{L}^1 = \hat{\mathbf{l}}^1), \quad (4)$$

$$\vdots$$

$$\hat{\mathbf{l}}^{N^*} = \arg\max_l P(\mathbf{L}^{N^*} = l \,|\, \mathbf{I}, \mathbf{L}^1 = \hat{\mathbf{l}}^1, \mathbf{L}^2 = \hat{\mathbf{l}}^2, \ldots). \quad (5)$$

Note that under our model, conditioning one trajectory given other ones simply means that it will go through no already occupied location. In other words,

$$P(\mathbf{L}^n = l \,|\, \mathbf{I}, \mathbf{L}^1 = \hat{\mathbf{l}}^1, \ldots, \mathbf{L}^{n-1} = \hat{\mathbf{l}}^{n-1}) = P(\mathbf{L}^n = l \,|\, \mathbf{I}, \forall k < n, \forall t, L_t^n \neq \hat{l}_t^k), \quad (6)$$

which is $P(\mathbf{L}^n = l \,|\, \mathbf{I})$ with a reduced set of the admissible grid locations.

Such a procedure is recursively correct: if all trajectories estimated up to step n are correct, then the conditioning only improves the estimate of the optimal remaining trajectories. This is the case if the image-data were informative enough so that locations could be unambiguously associated to individuals.

In practice, this is obviously rarely the case. Therefore, this greedy approach to optimization could lead to undesired side effects. For example, due to partly missing localization information for a given trajectory, the algorithm might mistakenly start following another person's trajectory. This is especially likely to happen if the tracked individuals are located close to each other.

Recall that we process the images by batches of 100. To avoid the kind of undesirable behavior described above, we first extend the trajectories that have been found with high confidence – as defined below – in the previous batches. We then process the lower confidence ones. As a result a trajectory which was problematic in the past and is likely to be problematic in the current batch will be optimized last and thus prevented from "stealing" somebody else's location. Furthermore, this approach increases spatial constraints on such a problematic trajectory when we finally get around to estimating it.

We use as a confidence score the concordance of the estimated trajectories in the previous batches and the localization cue provided by the Fixed Point Probability Field described in §5. Since there is a high degree of overlapping between successive batches, the challenging segment of a trajectory – due to failure of the background subtraction or change in illumination for instance – is met in several batches before it actually happens during the ten kept frames. Thus, the heuristic would have ranked the corresponding individual in the last ones to be processed when such problem occurs.

4.2 Single trajectory

Let us now consider only the trajectory $\mathbf{L}^n = (L_1^n, \ldots, L_T^n)$ of individual n over T temporal frames. We are looking for the values $(l_1^n, \ldots, l_T^n)$ in the subset of free locations of $\{1, \ldots, G, \mathcal{H}\}$ where G is the number of points in the discretization of the visible area and $\mathcal{H}$ is a hidden location. The initial location l_1^n is either a known visible location if the individual is visible in the first frame of the batch, or $\mathcal{H}$ if she is not. We therefore seek to maximize

$$P(L_1^n = l_1^n, \ldots, L_T^n = l_t^n \,|\, \mathbf{I}_1, \ldots, \mathbf{I}_T) \quad = \quad \frac{P(\mathbf{I}_1, L_1^n = l_1^n, \ldots, \mathbf{I}_T, L_T^n = l_T^n)}{P(\mathbf{I}_1, \ldots, \mathbf{I}_T)}. \tag{7}$$

Since the denominator is constant with respect to $\mathbf{l}^n$, we simply maximize the numerator, that is, the probability of both the trajectories and the images. Let us introduce the maximum of the probability of both the observations and the trajectory ending up at location k at time t

$$\Phi_t(k) = \max_{l_1^n, \ldots, l_{t-1}^n} P(\mathbf{I}_1, L_1^n = l_1^n, \ldots, \mathbf{I}_t, L_t^n = k). \tag{8}$$

Under a Markov model of the process $\mathbf{L}^n$, we have the classical recursive expression

$$\Phi_t(k) = \underbrace{P(\mathbf{I}_t \,|\, L_t^n = k)}_{\text{Appearance model}} \max_{\tau} \underbrace{P(L_t^n = k \,|\, L_{t-1}^n = \tau)}_{\text{Motion model}} \Phi_{t-1}(\tau) \tag{9}$$

to perform a global search with dynamic programming, which yields the classic Viterbi algorithm. This is straightforward since the L_t^n are in a finite set of size $G + 1$.

4.3 Motion model

We chose a very simple and unconstrained motion model

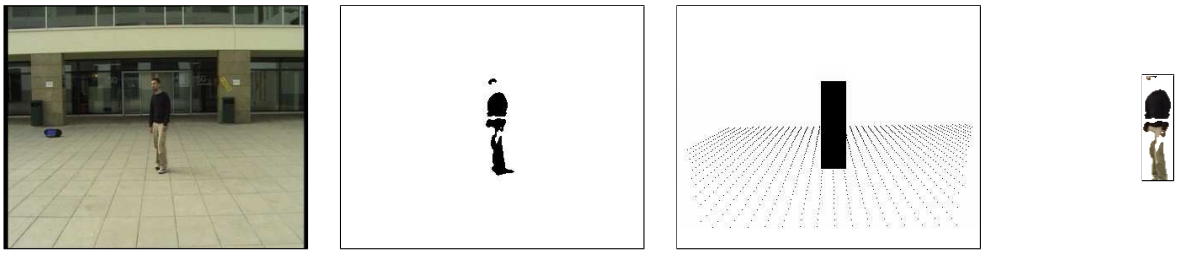
$$P(L_t^n = k \,|\, L_{t-1}^n = \tau) = \begin{cases} 1/Z \cdot e^{-\rho ||k-\tau||} & \text{if } ||k - \tau|| \leq c \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

where the constant ρ tunes the average human walking speed and c limits the maximum allowable speed. This probability is isotropic, decreases with the distance from location k and is zero for $||k-\tau||$ greater than a constant maximum distance. We use a very loose maximum distance c of one square of the grid per frame, which corresponds to a speed of almost 12mph. We also defined explicitly the parts of the scene that are connected to the hidden location $\mathcal{H}$. This is a single door in the indoor sequences and all the contours of the visible area in the outdoor sequences of Fig. 1.

4.4 Appearance Model

From the input images $\mathbf{I}_t$, we use background subtraction to produce binary masks $\mathbf{B}_t$, such as those of Fig. 4. We denote as $\mathbf{T}_t$ the colors of the pixels inside the blobs and treat the rest of the images as background, which is ignored.

Let X_k^t be a boolean random variable standing for the presence of an individual at location k of the grid at time t. In Appendix A, page 21, we show that



I_t^c B_t^c $\mathcal{A}_k^c$ $T_t^c(k)$

Figure 4: The color model relies on a stochastic modeling of the color of the pixels $T_t^c(k)$ sampled in the intersection of the binary image B_t^c produced by the background subtraction and the rectangle $\mathcal{A}_k^c$ corresponding to the location k.

$$\overbrace{P(\mathbf{I}_t \,|\, L_t^n = k)}^{\text{Appearance model}} \;\propto\; \underbrace{P(L_t^n = k \,|\, X_t^k = 1,\, \mathbf{T}_t)}_{\text{Color model}}\; \underbrace{P(X_t^k = 1 \,|\, \mathbf{B}_t)}_{\text{Ground plane occupancy}}. \tag{11}$$

The ground occupancy term will be discussed in §5, and the color model term is computed as follows.

4.5 Color model

We assume that if someone is present at a certain location k, her presence influences the color of the pixels located at the intersection of the moving blobs and the rectangle $\mathcal{A}_k^c$ corresponding to the location k. We model that dependency as if the pixels were independent and identically distributed and followed a density in the RGB space associated to the individual. This is far simpler than the color models used in either [15] or [10], which split the body area in several sub-parts with dedicated color distributions, but has proved sufficient in practice.

If an individual was present in the frames preceding the current batch, we have an estimation of her distribution, since we have previously collected the pixels in all frames at the locations of her estimated trajectory. If she is at the hidden location $\mathcal{H}$, we consider that her color distribution is flat.

Let $T_t^c(k)$ denote the pixels taken at the intersection of the binary image produced by the background subtraction from the stream of camera c at time t and the rectangle corresponding to location k in that same field of view (see Fig. 4). Note that even if an individual is actually at that location, this intersection can be empty if the background subtraction fails.

Let $\mu_1^c, \ldots, \mu_{N^*}^c$ be the color distributions of the N^* individuals present in the scene at the beginning of the batch of T frames we are processing, for camera c. The distribution may vary with the camera, due to difference in the camera technology or illumination angle. We have

$$\overbrace{P(L_t^n = k \,|\, X_t^k = 1,\, \mathbf{T}_t)}^{\text{Color model}} = \frac{P(L_t^n = k,\, X_t^k = 1,\, \mathbf{T}_t)}{P(X_t^k = 1,\, \mathbf{T}_t)} \tag{12}$$

$$= \frac{P(L_t^n = k,\, X_t^k = 1,\, \mathbf{T}_t)}{\sum_q P(L_t^q = k,\, X_t^k = 1,\, \mathbf{T}_t)} \tag{13}$$

$$= \frac{P(L_t^n = k,\, \mathbf{T}_t)}{\sum_q P(L_t^q = k,\, \mathbf{T}_t)} \tag{14}$$

$$= \frac{P(\mathbf{T}_t \,|\, L_t^n = k)}{\sum_q P(\mathbf{T}_t \,|\, L_t^q = k)} \tag{15}$$

where

$$P(\mathbf{T}_t \,|\, L_t^n = k) = P(T_t^1(k), \ldots, T_t^C(k) \,|\, L_t^n = k) \quad (16)$$

$$= \prod_{c=1}^{C} \prod_{q \in T_t^c(k)} \mu_n^c(q). \quad (17)$$

5 Probabilistic Occupancy Grid

In the previous section, we showed that, to find the MAP estimate of Eq. 1, we must supply accurate approximations of $P(X_t^k = 1 \,|\, \mathbf{B}_t)$, the probability of occupancy of grid location k at time t that appears in Eq. 11, for all k and all t.

In this section, we estimate an approximation of the $P(X_t^k = 1 \,|\, \mathbf{B}_t)$ by first introducing a generative model of $P(\mathbf{B}_t \,|\, \mathbf{X}_t)$ and then estimating the product law $Q(\mathbf{X}_t) = \prod_n Q(X_t^n)$ minimizing the Kullback-Leibler divergence to the true conditional law on $\mathbf{X}_t$ given $\mathbf{B}_t$ under this model. This lets us write the probabilities of occupancy as the solution of a system of coupled equations and leads to a very simple algorithm to reliably estimate them that we dubbed a "Fixed Point Probability Field" (FPPF).

Since we do this at each time step separately, we drop t from all notations in the remainder of this section for clarity.

In section §5.1 we introduce two independence assumptions under which we derive the analytical results of the other sections, and argue that they are legitimate. In section §5.2 we propose our generative model of $P(\mathbf{B} \,|\, \mathbf{X})$, which involves measuring the distance between the actual $\mathbf{B}$ and a crude synthetic image that is a function of the $\mathbf{X}$. From these assumptions and model, we derive in section §5.3 an analytical relation between estimates of the marginal probabilities of occupancy q_k by minimizing a Kullback-Leibler divergence between the corresponding product law and the true posterior. This leads to the Fixed Point Probability Field, a fast iterative algorithm that estimates them as the solution of a fixed point problem, as shown in section §5.4.

5.1 Independence Assumptions

We introduce here two assumptions of independence that will allow us to derive analytically the relation between the optimal q_ks.

Our first assumption is that individuals in the room do not take into account the presence of other individuals in their vicinity when moving around, which is true as long as avoidance strategies are ignored. This can be formalized as

$$P(X^1, \ldots, X^G) = \prod_k P(X^k). \quad (18)$$

Our second assumption involves considering that all statistical dependencies between views are due to the presence of individuals in the room. This is equivalent to treating the views as functions of the $X^1, \ldots, X^G$ vectors plus some independent noise. This implies that, as soon as the presence of all individuals is known, the views become independent. This is true as long as we ignore other hidden variables such as morphology, skin color or garment textures, that may simultaneously influence several views. This assumption can be written down as

$$P(B^1, \ldots, B^C \,|\, \mathbf{X}) = \prod_c P(B^c \,|\, \mathbf{X}). \quad (19)$$

5.2 Generative Image Model

To relate the values of the X_ks to the images produced by background subtraction $B^1, \ldots, B^C$, we propose here a model of the latter given the former. We use that relation in section §5.3 to compute an analytical relation between the optimal q_ks.

Let A^c be the synthetic image obtained by putting rectangles at locations where $X^k = 1$, thus $A^c = \oplus_k X^k \mathcal{A}_k^c$, where $\oplus$ denotes the "union" between two images. Such an image is a function of $\mathbf{X}$ and thus a random quantity. We model the images B^c produced by the background subtraction as if it was this ideal image with some random noise.

It appears empirically that the noise increases with the area of the ideal image A^c. We therefore introduce a normalized pseudo-distance Ψ to account for this asymmetry. For any gray-scale image $A \in [0,1]^{W \times H}$ we denote by $|A|$ the sum of its pixels, and we denote by $\otimes$ the product per-pixel of two images. We introduce Ψ defined by

$$\forall\, B,\ A \in [0,1]^{W \times H}, \quad \Psi(B, A) = \frac{1}{\sigma} \frac{|B \otimes (1 - A) + (1 - B) \otimes A|}{|A|}. \tag{20}$$

We model the conditional distribution of the background subtraction images given the true hidden state $P(B^c \,|\, \mathbf{X})$ as a density decreasing with the distance $\Psi(B^c,\, A^c)$ between the image produced by the background subtraction and an image obtained by putting rectangular shapes where people are present according to $\mathbf{X}$. We end up with the following model

$$\begin{aligned} P(\mathbf{B} \,|\, \mathbf{X}) &= \prod_c P(B^c \,|\, \mathbf{X}) && (21) \\ &= \prod_c P(B^c \,|\, A^c) && (22) \\ &= \frac{1}{Z} \prod_c e^{-\Psi(B^c,\, A^c)}. && (23) \end{aligned}$$

The parameter σ accounts for the quality of the background subtraction. The smaller σ the more B^c is picked around its ideal value A^c.

5.3 Relation between the q_k

We denote by E_Q the expectation under $\mathbf{X} \sim Q$. Since we want to minimize the Kullback-Leibler divergence between the approximation Q and the "true" posterior $P(\,\cdot \,|\, \mathbf{B})$, we use the following form of its derivative with respect to the unknown q_k (see Appendix B, page 22)

$$\begin{aligned} &\frac{\partial}{\partial q_k} KL(Q,\, P(\,\cdot \,|\, \mathbf{B})) \\ &\quad = \log \frac{q_K\,(1 - \epsilon_k)}{(1 - q_k)\,\epsilon_k} + E_Q\left(\sum_c \Psi(B^c, A^c) \,\middle|\, X^k = 1 \right) - E_Q\left(\sum_c \Psi(B^c, A^c) \,\middle|\, X^k = 0 \right). \end{aligned} \tag{24}$$

Hence, if we solve

$$\frac{\partial}{\partial q_k} KL(Q,\, P(\,\cdot \,|\, \mathbf{B})) = 0 \tag{25}$$

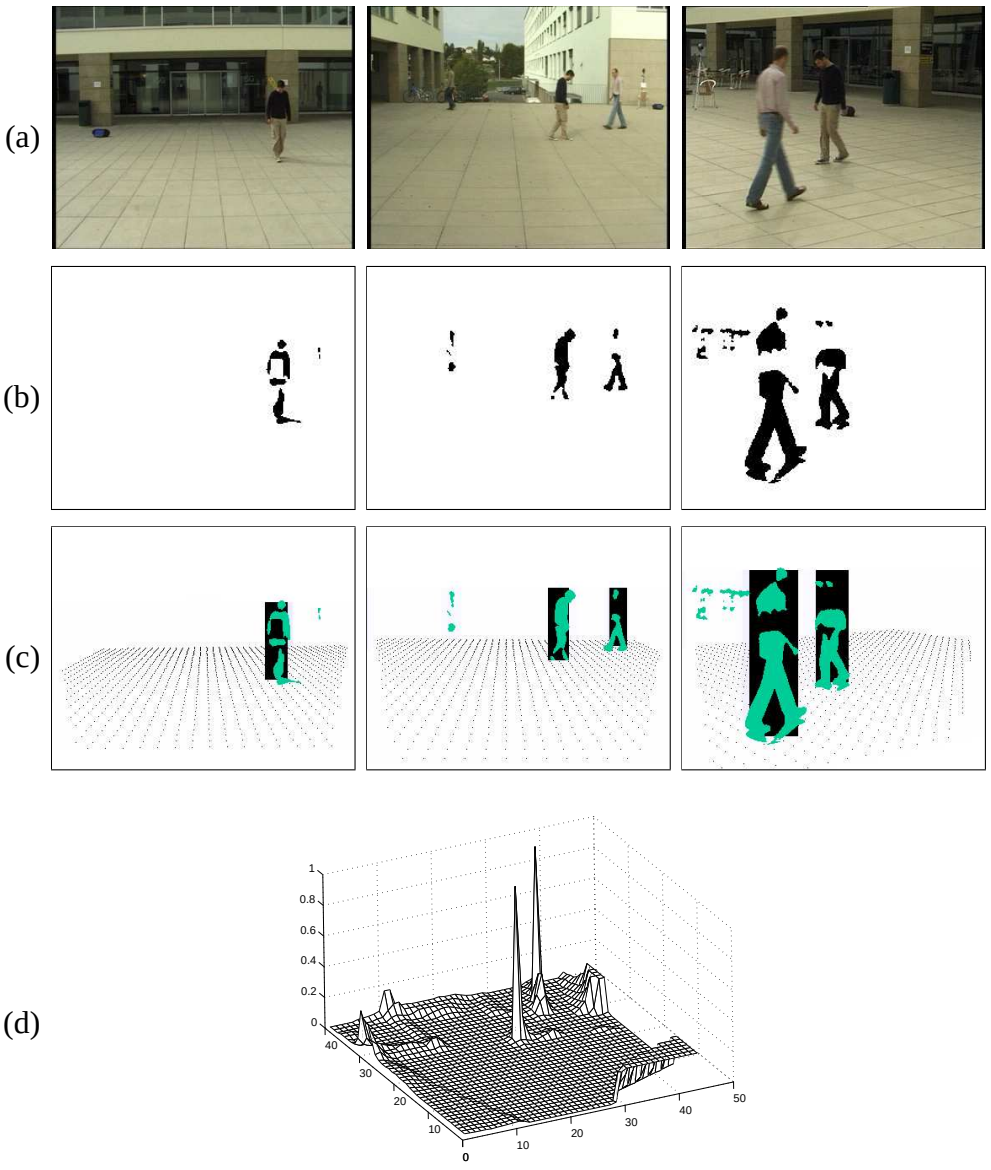


Figure 5: Original images from three cameras (a), binary images produced by background subtraction (b) and synthetic average images computed from them by the fixed-point algorithm (c). The graph (d) represents the corresponding occupancy probabilities q_k on the grid.

we obtain

$$q_k = \frac{1}{1 + \exp\left(\lambda_k + \sum_c E_Q(\Psi(B^c, A^c) \,|\, X^k = 1) - E_Q(\Psi(B^c, A^c) \,|\, \mathbf{X}) \,|\, X^k = 0)\right)}, \tag{26}$$

with $\lambda_k = \log \frac{1-\epsilon_k}{\epsilon_k}$.

Unfortunately, the computation of $E_Q(\Psi(B^c, A^c) \,|\, X^k = \xi)$ is untractable. However, since under $\mathbf{X} \sim Q$ the image A^c is concentrated around B^c, we approximate, $\forall \xi \in \{0,\, 1\}$

$$E_Q(\Psi(B^c, A^c) \,|\, X^k = \xi) \simeq \Psi(B^c, E_Q(A^c \,|\, X^k = \xi)) \tag{27}$$

leading to our main result

$$q_k = \frac{1}{1 + \exp\left(\lambda_k + \sum_c \Psi(B^c, E_Q(A^c \,|\, X^k = 1)) - \Psi(B^c,\, E_Q(A^c \,|\, X^k = 0))\right)}. \tag{28}$$

Note that the conditional synthetic images $E_Q(A^c \,|\, X^k = 0)$ and $E_Q(A^c \,|\, X^k = 1)$ are equal to $E_Q(A^c)$ with q_k forced to 0 or 1 respectively, as show on Fig. 6. Since Q is a product law, we have for any pixel (x, y)

$$\begin{aligned} E_Q(A^c(x,y)) &= Q(A^c(x,y) = 1) && (29)\\ &= 1 - Q(\forall k, \mathcal{A}_k^c(x,y)\, X_k = 0) && (30)\\ &= 1 - \prod_{k : \mathcal{A}_k^c(x,y) = 1} (1 - q_k) && (31) \end{aligned}$$

Finally, $E_Q(A^c \,|\, X^k = \xi)$ are functions of the $(q_l)_{l \neq k}$ and Equation (28) can be seen as one equation of a large system whose unknowns are the q_ks. Fig. 7 shows the evolution of both the marginals q_k and the average images $E_Q(A^c)$ during the iterative estimation of the solution.

Intuitively, if putting the rectangular shape for position k in the image improves the fit with the actual images, the score $\Psi(B^c, E_Q(A^c \,|\, X^k = 1))$ decreases, $\Psi(B^c, E_Q(A^c \,|\, X^k = 0))$ increases, and the sum in the exponential is negative, leading to a larger q_k. Note that occlusion is taken into account naturally: If a rectangular shape at position k is occluded by another one whose presence is very likely, the value of q_k does not influence the average image and all terms vanish but λ_k in the exponential. Thus the resulting q_k remains equal to the prior.

5.4 Fast Estimation of the q_ks

Let $\oplus$ denote the pixel-wise disjunction operator between binary images (the "union" image), $\otimes$ the pixel-wise product (the "intersection" image), $|I|$ the sum of the pixels of an image I and let 1 be the constant image whose pixels are all equal to 1.

We estimate the q_k as follows: We first give them a uniform value and use them to compute the average synthetic images $\overline{A}^c_{k,\xi} = E_Q(A^c \,|\, X^k = \xi)$. We then re-estimate every q_k with equation (28) and iterate the process until a stable solution is reached.

The main remaining issue is the computation of $\Psi(B^c,\, \overline{A}^c_{k,\xi})$ which has to be done G times per iteration for as many iterations as required to converge, which is usually of the order of 100. Fortunately, by using integral images, given any image I and any rectangular shape $\mathcal{A}$, the cost of the computation of $|I \otimes \mathcal{A}|$ does not depend on $\mathcal{A}$, apart from a pre-computation whose cost is proportional to the area of I. We organize the computation to take advantage of that trick, and finally perform the following steps at each iteration of our algorithm

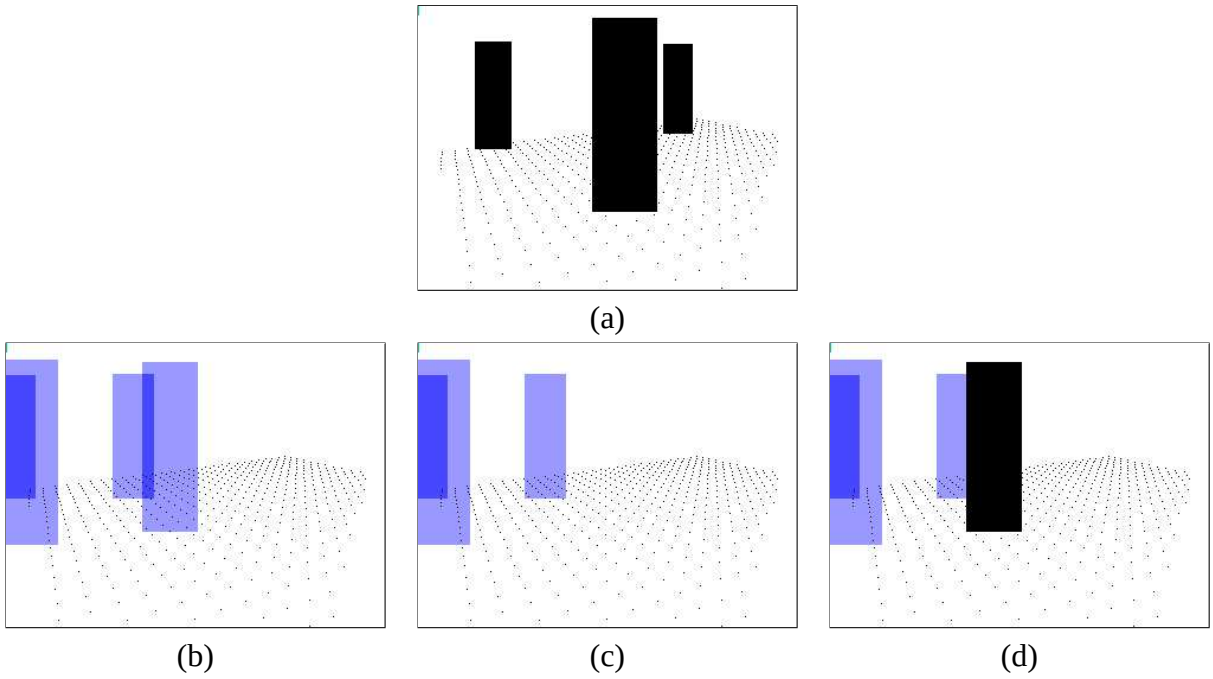


Figure 6: Picture (a) shows a synthetic picture A^c with three X^ks equal to 1. Picture (b) shows the average image $E_Q(A^c)$ where all q_k are null but four of them equal to 0.2. Pictures (c) and (d) show $\overline{A}^c_{k,0} = EQ(A^c \,|\, X^k = 0)$ and $\overline{A}^c_{k,1} = E_Q(A^c \,|\, X^k = 1)$ respectively, where k is the location corresponding to the black rectangle in (d).

$$\overline{A}^c = 1 - \otimes_k (1 - q_k \mathcal{A}^c_k) \tag{32}$$

$$|\overline{A}^c_{k,\xi}| = |\overline{A}^c| + \frac{\xi - q_k}{1 - q_k} |(1 - \overline{A}^c) \otimes \mathcal{A}^c_k| \tag{33}$$

$$|B_c \otimes \overline{A}^c_{k,\xi}| = |B_c \otimes \overline{A}^c| + \frac{\xi - q_k}{1 - q_k} |B_c \otimes \left(1 - \overline{A}^c\right) \otimes \mathcal{A}^c_k| \tag{34}$$

$$\Psi(B_c, \overline{A}^c_{k,\xi}) = \frac{1}{\sigma} \frac{|B_c| - 2\,|B_c \otimes \overline{A}^c_{k,\xi}| + |\overline{A}^c_{k,\xi}|}{|\overline{A}^c_{k,\xi}|} \tag{35}$$

$$q_k \leftarrow \frac{1}{1 + \exp\left(\lambda_k + \sum_c \Psi(B_c, \overline{A}^c_{k,1}) - \Psi(B_c, \overline{A}^c_{k,0})\right)} \tag{36}$$

At each iteration and for every c, step (32) consists of computing the average of the synthetic image under Q with the current estimates of q_ks and has a cost proportional to the sum of the areas of the $\mathcal{A}_k$. Step (33) (respectively (34)) computes the sum of the pixels of the conditional average images, given X_k (respectively of the same image multiplied pixel-wise by the result of the background subtraction) which is done at constant time for every k, given the pre-computation of the integral images of $1 - \overline{A}^c$ (respectively of $B_c \otimes \left(1 - \overline{A}^c\right)$). Finally, steps (35) and (36) compute the distance between the result of the background subtraction and the conditional average synthetic under Q, and the corresponding updated marginal probability. Except for the exponential in the last step, which has to be repeated for every location, the computation only involves sums and products and is therefore fast.

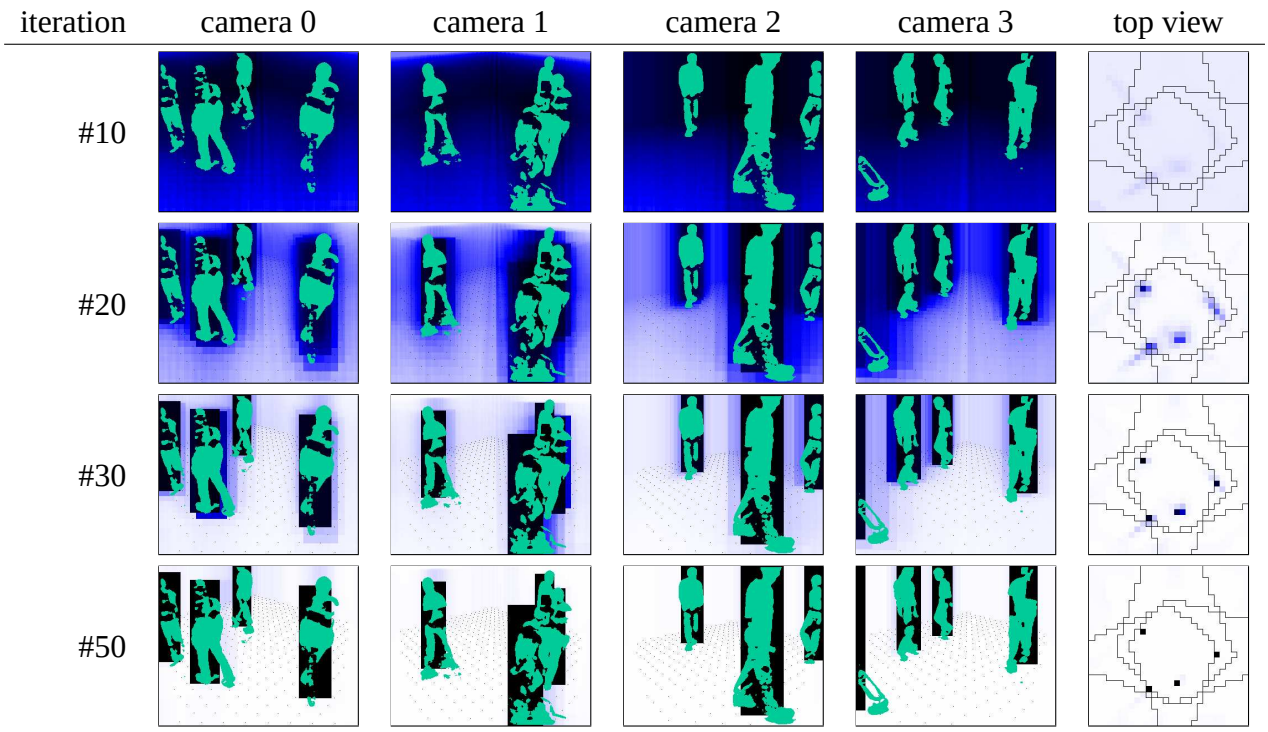


Figure 7: This illustrates the convergence process of the ground occupancy algorithm, both in top and camera views. The green color shows the background subtraction blobs, and the blue the probability of occupancy. The darker, the more probable.

6 Results

In our implementation, we first compute the probabilities of occupancy of Section §5 separately at each time step and then use the results as input to the dynamic programming approach of Section §4. Here we first present the results obtained at individual time steps and then the result of our global optimization.

6.1 Video sequences

We use here two indoor and two outdoor sequences that were shot on our campus.

6.1.1 Indoor sequences

The two sequences depicted by the upper row of Fig. 1 and three upper rows on Fig. 9 were shot by a video-surveillance dedicated setup of 4 synchronized cameras in a 50m^2 room. Two cameras were roughly at head level ($\simeq$ 1.80m) and the two others slightly higher ($\simeq$ 2.30m). They were located at each corners of the room. The first sequence is 3800 frame long and shows four individuals entering the room and walking continuously. The second contains 2800 frames and involves six individuals. This actually results in a more complex reconstruction problem than usually happens in real-life situations, mostly because people tend to occlude each other much more often.

In these sequences, the area of interest was of size 5.5m $\times$ 5.5m $\simeq$ 30m^2 and discretized into G =28$\times$28=794 locations, corresponding to a regular grid with a 20cm resolution.

6.1.2 Outdoor sequences

The outdoor sequences shown in the lower row of Fig. 1 and three lower rows on Fig. 9 were shot in front of the entrance of a building on our campus. We used three standard and unsynchronized Digital Video cameras and synchronized the video streams by hand afterward. All cameras were at head level ($\simeq$ 1.80m) covering the area of interest from three angles. The ground is flat with a regular pavement.

The area of interest is of size 10m $\times$ 10m and discretized into G =40$\times$40=1600 locations, corresponding to a regular grid with a resolution of 25cm. Up to four individuals appear simultaneously.

6.2 Occupancy Probabilities at Individual Time Steps

Fig. 7 displays the convergence of the ground occupancy part of our algorithm alone on a single time frame, while Fig. 8 shows its detection results on both the indoor and outdoor sequences. These results are only about *detection*. The algorithm operated on a frame-by-frame basis, with no time consistency and thus does not maintain identities of the people across frames.

As can be seen from the screenshots, the accuracy of ground occupancy algorithm is generally very good. Its performance is however correlated with the quality of the background subtraction blobs. An inaccurate blob detection as shown in frame #1048 of Fig. 8 can lead to incorrect people detection.

We have computed precise error rates by counting in each frame the number of actually present individuals, the number of detected individuals, and the number of false detections. We defined a correct detection as one for which the reprojected boxes intersect the individual on at least three camera views out of four. Such a tolerance accommodates for cases where, due to optical deformations, while the estimated location is correct, the reprojection does not match the real location on one view. With such a definition, the estimated false negative error rate on the indoor video sequence with 4 people is 6.14% and the false-positive error rate is 3.99%. In other words, in absolute terms, our detection performance is excellent considering that we have used only a small proportion of the available image information.

6.3 Global Trajectories Estimated over the whole Sequence

Since the observed area is discretized into a finite number of positions, we improve the result accuracy by linearly interpolating the trajectories on the output images.

6.3.1 Indoor sequences

On both of those sequences, the algorithm performs very well and does not lose a single one of the tracked persons. To investigate the spatial accuracy of our approach, we compared the estimated locations with the actual locations of the individuals present in the room as follows.

We picked 100 frames at random among the complete four individual sequence and marked by hand a reference point located on the belly of every person present in every camera view. For each frame and each individual, from that reference point and the calibration of the four cameras, we estimated a ground location. Since the 100 frames were taken from a sequence with four individuals entering the room successively, we obtained 354 locations.

We then estimated the distance between this ground-truth and the locations estimated by the algorithm. The results are depicted by the bold curve on Fig. 2. More than 90% of those estimates are at a distance of less than 31cm and 80% of less than 25cm. We also computed similar curves after having replaced a certain percentage of images taken randomly over the complete sequence by blank images. The accuracy remains unchanged for an erasing rate as high as 20%. The performance of the algorithm starts to get worse when we get rid of one third of the images, as shown with the thin curves on Fig. 2, page 3.

Figure 8: Results of the ground occupancy algorithm. Shown are the background subtraction images, as well as the corresponding detection after thresholding. On the top right column are displayed the occupancy probability map, without any post processing. Time frame #2643 illustrates a failure case, in which the algorithm detects a person at the wrong place, due to the bad quality of background subtraction.

Figure 9: Results of the tracking algorithm. Each row displays several views of the same time frame coming from different cameras. The column for camera 4 is left blank for the outdoor scenes, for which only three cameras were used. Note that, on frame #1689, people #2 and #3 are seen by only one camera, but are nonetheless correctly identified.

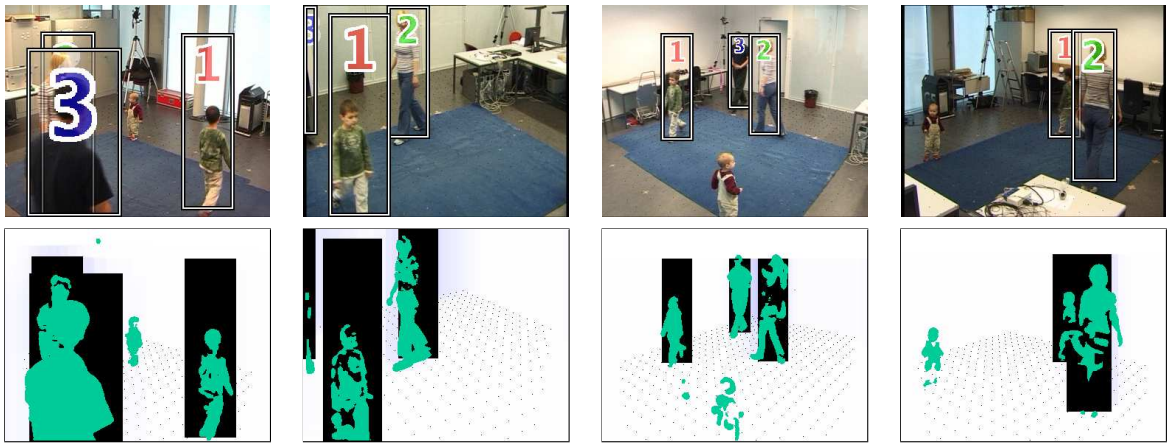


Figure 10: This figure illustrates the sensitivity of the ground occupancy algorithm to people size. The first row shows the four camera views of a single time frame, as well as the detection results. The second row displays the synthetic images corresponding to the ground occupancy estimation made by our algorithm. In green is the output of the background subtraction. We can see that, although quite small, the boy is correctly detected. The little girl however, is too small to be detected.

6.3.2 Outdoor sequences

Despite disturbing influence of external elements such as shadows, the sliding door visible on camera #1, cars passing by in camera #2, and the fact that people can enter and exit the tracked area from anywhere, the algorithm performs well and follows people accurately. In many cases, because the cameras are not located ideally, individuals appear on one stream alone. They are still correctly localized due to both the time consistency and the rectangle-matching of the ground occupancy estimation which is able to exploit the size of the blobs even in a monocular context. Such a case appears for individuals #2 and #3 in frame #1689, Fig. 9. On both outdoor sequences, the algorithm does not produce one false positive or false negative, nor make confusion between individuals.

6.3.3 Rectangle size influence

We checked the influence of the size of the rectangular shapes we use as models: The results of the ground occupancy algorithm are almost unchanged for model sizes between 1.7 m and 2.2 m. The performance tends to decrease for sizes noticeably smaller. This can be explained easily: if the model is shorter than the person, the algorithm will be more sensitive to spurious binary blobs that it may explain by locating a person in the scene, which is less likely to happen with taller models.

To further investigate how our algorithm behaves in presence of people whose size is significantly different from our standardized 1.75m high rectangles, we have shot a video sequence involving both adults and children. Besides the two adults, it features a 4 year old boy who is just above 1 meter high and a 2 year old girl who measures about 80 centimeters. Fig. 10 shows the results of our tracking algorithm on this sequence. It is able to accurately follow the boy throughout the whole sequence. The little girl, however, is not detected. The blobs that she creates cover approximately a quarter only of an adult blob, and the ground occupancy algorithm therefore does not see her most of the time.

Observing the sequence carefully, we noticed that, due to the boys small size, its detection is a bit less reliable than adults one. However, the global optimization smooths the trajectory and compensates for the missing detection so that it is not noticeable on the results. On the contrary, the little girl is rarely detected, and her occupancy information is discarded as noise by the global optimization. This example clearly shows the limits of our system, but also emphases its consistency.

7 Conclusion

We have presented an algorithm that can reliably track multiple persons in a complex environment and provide metrically accurate position estimates. This is achieved by combining the Fixed Point Probability Field, which provides a very robust estimation of the occupancy on the ground plane in individual time frame, with a global optimization of the trajectories of the detected individuals over 100-frame batches. This optimization scheme introduces a 4 second delay between image acquisition and output of the results, which we believe to be compatible with many surveillance applications given the robustness increase if offers.

There are many possible extensions of this work. The most obvious ones are improvements of our stochastic model. The color model could be refined by splitting bodies into several uniform parts instead of relying on the i.i.d assumption. Similarly, the motion model could take into account consistency of speed and direction. Modeling the avoidance strategies between people would also help. However, it is worth noting that, even with our very simple models, we already obtain very good performance, thus underlining the power of our trajectory optimization approach.

Beside those straightforward improvements, a more ambitious extension would be to use the current scheme to automatically estimate trajectories from a large set of video, from which one could then learn sophisticated appearance and behavior models. These models could in turn be incorporated into the system to handle the increasingly difficult situations that will inevitably arise when the scenes become more crowded or we use fewer cameras.

A Appearance model

Recall that our appearance model is given by

$$P(\mathbf{I}_t \,|\, L_t^n = k), \tag{37}$$

where $\mathbf{I}_t$ are the input images at time frame t and L_t^n is the random variable representing the location on the grid of individual n, also at time t. From the input images $\mathbf{I}_t$, we use background subtraction to produce binary masks $\mathbf{B}_t$. We denote as $\mathbf{T}_t$ the colors of the pixels inside the blobs and treat the rest of the images as background, which is ignored.

Let X_k^t be a boolean random variable standing for the presence of an individual at location k of the grid at time t. Then we have

$$\begin{aligned}
\overbrace{P(\mathbf{I}_t \,|\, L_t^n = k)}^{\text{Appearance model}} &= \frac{P(\mathbf{I}_t)}{P(L_t^n = k)} P(L_t^n = k \,|\, \mathbf{I}_t) && (38)\\
&\propto P(L_t^n = k \,|\, \mathbf{I}_t) && (39)\\
&= P(L_t^n = k \,|\, \mathbf{B}_t,\, \mathbf{T}_t) && (40)\\
&= P(L_t^n = k,\, X_t^k = 1 \,|\, \mathbf{B}_t,\, \mathbf{T}_t) && (41)\\
&= P(L_t^n = k \,|\, X_t^k = 1,\, \mathbf{B}_t,\, \mathbf{T}_t)\, P(X_t^k = 1 \,|\, \mathbf{B}_t,\, \mathbf{T}_t)\\
&= \underbrace{P(L_t^n = k \,|\, X_t^k = 1,\, \mathbf{T}_t)}_{\text{Color model}}\ \underbrace{P(X_t^k = 1 \,|\, \mathbf{B}_t)}_{\text{Ground plane occupancy}}. && (42)
\end{aligned}$$

Equality (38) follows directly from Bayes formula. Equality (39) is true since the probability of the image – without conditioning – does not depend on the trajectory and the prior on the trajectories is flat. Equality (40) is true under the assumption that all information is carried by the product of the background subtraction and the set of the blob pixel colors. Equality (41) is true since $L_t^n = k \Rightarrow X_t^k = 1$, and finally equality (42) is true under the assumptions that the occupancy of a location X_t^k

provides strictly more information about someone being at location k than the result of the background subtraction, and that given the result of the background subtraction, the color of the blobs does not provide information about the occupancy.

B Relation between the q_ls

We are looking for an approximation of $P(X_t^k = 1 \,|\, \mathbf{B}_t)$. Having introduced a generative model of $P(\mathbf{B}_t \,|\, \mathbf{X}_t)$, we estimate the product law $Q(\mathbf{X}_t) = \prod_n Q(X_t^n)$ minimizing the Kullback-Leibler divergence to the true conditional law on $\mathbf{X}_t$ given $\mathbf{B}_t$ under this model. We denote by E_Q the expectation under $\mathbf{X} \sim Q$ and we derive the Kullback-Leibler divergence with respect to the unknown q_k

$$
\begin{aligned}
& \frac{\partial}{\partial q_k} KL(Q,\, P(\,\cdot\,|\,\mathbf{B})) \\
= \;& \frac{\partial}{\partial q_k} E_Q\left(\log \frac{Q(\mathbf{X})}{P(\mathbf{X}|\mathbf{B})}\right) && (43)\\
= \;& \frac{\partial}{\partial q_k} E_Q\left(\log \frac{Q(\mathbf{X})}{P(\mathbf{X})}\frac{P(\mathbf{B})}{P(\mathbf{B}|\mathbf{X})}\right) && (44)\\
= \;& \frac{\partial}{\partial q_k} E_Q\left(\sum_l \log \frac{Q(X^l)}{P(X^l)} + \log P(\mathbf{B}) - \log P(\mathbf{B}|\mathbf{X})\right) && (45)\\
= \;& \frac{\partial}{\partial q_k} E_Q\left(\log \frac{Q(X^k)}{P(X^k)} - \log P(\mathbf{B}|\mathbf{X})\right) && (46)\\
= \;& \frac{\partial}{\partial q_k}\, q_k\left(\log \frac{q_k}{\epsilon_k} - E_Q\big(\log P(\mathbf{B}|\mathbf{X}) \,|\, X^k = 1\big)\right) \\
& + \frac{\partial}{\partial q_k}\,(1-q_k)\left(\log \frac{1-q_k}{1-\epsilon_k} - E_Q\big(\log P(\mathbf{B}|\mathbf{X}) \,|\, X^k = 0\big)\right) && (47)\\
= \;& \log \frac{q_k}{\epsilon_k} + 1 - E_Q\big(\log P(\mathbf{B}|\mathbf{X}) \,|\, X^k = 1\big) - \log \frac{1-q_k}{1-\epsilon_k} - 1 + E_Q\big(\log P(\mathbf{B}|\mathbf{X}) \,|\, X^k = 0\big) && (48)\\
= \;& \log \frac{q_K\,(1-\epsilon_k)}{(1-q_k)\,\epsilon_k} - E_Q\big(\log P(\mathbf{B}|\mathbf{X}) \,|\, X^k = 1\big) + E_Q\big(\log P(\mathbf{B}|\mathbf{X}) \,|\, X^k = 0\big) \\
= \;& \log \frac{q_K\,(1-\epsilon_k)}{(1-q_k)\,\epsilon_k} - E_Q\left(-\sum_c \Psi(B^c, A^c) \;\middle|\; X^k = 1\right) + E_Q\left(-\sum_c \Psi(B^c, A^c) \;\middle|\; X^k = 0\right) && (49)
\end{aligned}
$$

Equality (43) is the definition of the Kullback-Leibler divergence, (44) is obtained by applying Bayes's rule to $P(\mathbf{X}|\mathbf{B})$. Equality (45) is true under our assumption of independence of the X_ks and (46) by removing terms which are constant with respect to q_k. We develop the expectation by conditioning on X_k to get (47), do formal differentiation to obtain (48), and finally introduce our model of $P(\mathbf{B}|\mathbf{X})$ and assumption of conditional independence of the B^c given $\mathbf{X}$ to get (49).

References

[1] T. Z. and R. Nevatia. Tracking multiple humans in crowded environment. In *Conference on Computer Vision and Pattern Recognition*, 2004.

[2] J. Black, T. Ellis, and P. Rosin. Multi-view image surveillance and tracking. In *IEEE Workshop on Motion and Video Computing*, 2002.

[3] Q. Cai and J. Aggarwal. Automatic tracking of human motion in indoor scenes across multiple synchronized video streams. In *ICCV*, 1998.

[4] R. T. Collins. Mean-shift blob tracking through scale space. In *Conference on Computer Vision and Pattern Recognition*, volume 02, page 234, Los Alamitos, CA, USA, 2003. IEEE Computer Society.

[5] D. Focken and R. Stiefelhagen. Towards vision-based 3d people tracking in a smart room. In *IEEE International Conference on Multimodal Interfaces*, 2002.

[6] J. Giebel, D. Gavrila, and C. Schnorr. A bayesian framework for multi-cue 3d object tracking. In *Proceedings of European Conference on Computer Vision*, 2004.

[7] M. Han, W. Xu, H. Tao, and Y. Gong. An algorithm for multiple object trajectory tracking. In *Conference on Computer Vision and Pattern Recognition*, volume 1, pages 864–871, June 2004.

[8] I. Haritaoglu, D. Harwood, and L. Davis. Who, when, where, what: A real time system for detecting and tracking people. In *FGR*, pages 222–227, 1998.

[9] M. Isard and J. MacCormick. Bramble: a bayesian multiple-blob tracker. In *Conference on Computer Vision and Pattern Recognition*, volume 2, pages 34–41, July 2001.

[10] J. Kang, I. Cohen, and G. Medioni. Tracking people in crowded scenes across multiple cameras. In *Asian Conference on Computer Vision*, 2004.

[11] S. Khan, O. Javed, and M. Shah. Tracking in uncalibrated cameras with overlapping field of view. In *2nd IEEE Workshop on Performance Evaluation of Tracking and Surveillance*, 2001.

[12] S. Khan and M. Shah. Tracking people in presence of occlusion. In *Asian Conference on Computer Vision*, 2000.

[13] J. Krumm, S. Harris, B. Myers, B. Brummit, M. Hale, and S. Shafer. Multi-camera multi-person tracking for easy living. In *IEEE Workshop on Visual Surveillance*, 2000.

[14] I. Mikic, S. Santini, and R. Jain. Video processing and integration from multiple cameras. In *Proceedings of the 1998 Image Understanding Workshop, Morgan-Kaufman, San Francisco*, 1998.

[15] A. Mittal and L. Davis. M2tracker: A multi-view approach to segmenting and tracking people in a cluttered scene. *International Journal of Computer Vision*, 51(3):189–203, 2003.

[16] K. Okuma, A. Taleghani, N. de Freitas, J. Little, and D. Lowe. A boosted particle filter: multitarget detection and tracking. In *ECCV*, Prague, Czech Republic, May 2004.

[17] K. Otsuka and N. Mukawa. Multi-view occlusion analysis for tracking densely populated objects based on 2-d visual angles. In *Conference on Computer Vision and Pattern Recognition*, 2004.

[18] K. Smith, D. Gatica-Perez, and J.-M. Odobez. Using particles to track varying numbers of interacting people. In *CVPR*, 2005.