

A Probabilistic Framework for Combining Tracking Algorithms

Ido Leichter, Michael Lindenbaum and Ehud Rivlin

Department of Computer Science

Technion – Israel Institute of Technology

Technion City, Haifa 32000, Israel

{idol|mic|ehudr}@cs.technion.ac.il

Abstract

For the past few years researches have been investigating enhancing tracking performance by combining several different tracking algorithms. We propose an analytically justified, probabilistic framework to combine multiple tracking algorithms. The separate tracking algorithms considered output a probability distribution function of the tracked state, sequentially for each image. The algorithms may output either an explicit probability distribution function, or a sample-set of it via CONDENSATION. The proposed framework is general and allows the combination of any set of separate tracking algorithms of this kind, even on different state spaces of different dimensionality, under a few reasonable assumptions. In many of the investigated settings, our approach allows us to treat the separate tracking algorithms as “closed boxes”. In other words, only the state distributions in the input and output are needed for the combination process. The suggested framework was successfully tested using various state spaces and datasets.

1. Introduction

For the past few years researches have been investigating enhancing tracking performance by combining several different tracking algorithms. In [14] multiple window-based trackers simultaneously track the same feature. Different situations in which miss-tracking occurs are categorized and dealt with by the appropriate trackers. In [4] the individual estimates of two tracking systems, based on the bounding-box and the 2D pattern of the targets, are combined to produce a global estimate for vehicle tracking. The combination is performed by utilizing instantaneous performance coefficients. In [11] two tracking algorithms, a region tracker and an edge tracker, are run in parallel. The two algorithms, having complementary failure modes, correct each other based on their confidence measures. In [7] the results of two corner trackers, global correspondence and local relaxation, are merged using a classification based approach and reliability attributes for each corner match. In [12] a system containing three co-operating detection and

tracking modules is devised for tracking people in an indoor environment. The three modules are an active shape tracker, a region tracker featuring region splitting and merging for multiple hypothesis matching, and a head detector to aid in the initialization of tracks. Another related work is [3], where a 2D-location tracker performs on-line switching between different features, according to their quality of discrimination between object and background pixels. Recently, the so-called *democratic integration* scheme [15], was used in [13] to combine multiple cues for visual tracking.

The algorithm combination methods mentioned are either heuristic or specifically tailored to the particular separate algorithms used. For example, the combination in [12] consists of running in parallel multiple algorithms, where each one is devised for a specific sub-task, and is dependent on the output of other algorithms for initialization or correction. The algorithms described in [11] run in parallel, and switching between their outputs is done according to ad-hoc rules. Perhaps the closest work to our approach is [13], where multiple algorithms are run in parallel to track the same object, provide location distribution estimates, and are heuristically combined by weighted sum.

In this paper we propose an analytically justified, probabilistic framework to combine multiple tracking algorithms. The framework investigated here has the following significant advantages: 1. It is general. That is, it may be used for combining any set of tracking algorithms that provide a PDF (Probability Distribution Function) estimate representation of the tracked object state as their output, even on different state spaces. 2. There is no need to devise confidence measures for the output of each separate tracking algorithm in order to explicitly switch between them or differently weight them in the final estimate. 3. It enables the combination of tracking algorithms using state spaces of different dimensionality. 4. Even if all the separate algorithms provide multi-modal PDFs, the combination may be done, and the combined PDF may still be unimodal. 5. In many settings, the proposed combination uses only the state distributions in the algorithms' input and output for the combination pro-

cess (i.e., “closed boxes approach”). See more details in the conclusion.

We start by defining the framework considered (Section 2). Then we propose methods to combine tracking algorithms, both explicit-PDF yielding ones (Section 3), as well as CONDENSATION-based (Section 4). The combination methods are first developed for the case where the separate algorithms track in a common state space, and then generalized for the case of different state spaces. We continue by presenting experimental results (Section 5), and conclude with a short summary (Section 6).

2. Framework

We assume the following regarding the separate algorithms:

1. The algorithms provide a PDF estimate representation of the tracked state, sequentially for each image. A specific algorithm outputs either an *explicit* PDF, or a sample-set of it via CONDENSATION. Such algorithms are very common. For example, any algorithm using Kalman filtering explicitly provides a Gaussian PDF of the tracked state (e.g., [2]). Other tracking algorithms employing a general discrete probability distribution for tracking are given in [9], [10] and [1]. Algorithms providing samples from general PDFs using CONDENSATION are also widely used.

2. The algorithms are *conditionally independent*, i.e., each algorithm relies on features which, given the tracked state, are conditionally independent of the features used by the other algorithms. That is, if algorithm A_i uses features \mathbf{z}_i and algorithm A_j uses features \mathbf{z}_j ($i \neq j$), then we assume for every object state \mathbf{x} that

$$f(\mathbf{z}_i, \mathbf{z}_j | \mathbf{x}) = f(\mathbf{z}_i | \mathbf{x}) \cdot f(\mathbf{z}_j | \mathbf{x}). \quad (1)$$

While this assumption is indeed difficult to verify, there are many cases where it is true or approximately true. Moreover, note that tracking algorithm combination is most beneficial when the separate trackers use conditionally independent features in the first place, reinforcing the applicability of this assumption.

For clearer presentation we will discuss combining *two* tracking algorithms, but the entire discussion and results may be easily generalized to combine an arbitrary number of tracking algorithms.

3. The explicit PDF case

3.1. Same state space

Let A_1 and A_2 be two conditionally independent tracking algorithms, tracking the object in a common state space. Let \mathbf{z}_1 and \mathbf{z}_2 be the features used by A_1 and A_2 , respectively. Each algorithm receives at its input $f_{t-1}(\mathbf{x})$ – the tracked object state PDF at time $t-1$, and I_t – the image at time t (the t^{th} frame). Each algorithm A_i extracts from the image

its relevant features \mathbf{z}_i , and outputs $f_t(\mathbf{x} | \mathbf{z}_i, f_{t-1})$ – the PDF of the tracked state at time t , given the features \mathbf{z}_i in the t^{th} frame and the state PDF at time $t-1$. The overall system of the two separate algorithms is illustrated in Fig. 1.

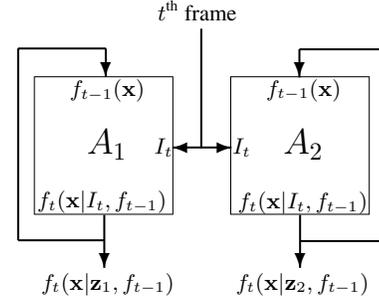


Figure 1: Two separate explicit PDF-yielding tracking algorithms. Both algorithms track in the same state space.

Let us now derive $f_t(\mathbf{x} | \mathbf{z}_1, \mathbf{z}_2, f_{t-1})$ – the state PDF of the combined system. In the derivation we apply Bayes’ rule [8] and assumption (1). Also, it is obvious that if the current object state is given, then any additional knowledge on the state at previous times has no influence on the current image. Specifically,

$$f_t(\mathbf{z} | \mathbf{x}, f_{t-1}) = f_t(\mathbf{z} | \mathbf{x}). \quad (2)$$

Now,

$$\begin{aligned} f_t(\mathbf{x} | \mathbf{z}_1, \mathbf{z}_2, f_{t-1}) &= \frac{f_t(\mathbf{z}_1, \mathbf{z}_2 | \mathbf{x}, f_{t-1}) f_t(\mathbf{x} | f_{t-1})}{f_t(\mathbf{z}_1, \mathbf{z}_2 | f_{t-1})} \\ &= f_t(\mathbf{z}_1, \mathbf{z}_2 | \mathbf{x}) \cdot \frac{f_t(\mathbf{x} | f_{t-1})}{f_t(\mathbf{z}_1, \mathbf{z}_2 | f_{t-1})} \\ &= f_t(\mathbf{z}_1 | \mathbf{x}) \cdot f_t(\mathbf{z}_2 | \mathbf{x}) \cdot \frac{f_t(\mathbf{x} | f_{t-1})}{f_t(\mathbf{z}_1, \mathbf{z}_2 | f_{t-1})} \\ &= f_t(\mathbf{z}_1 | \mathbf{x}, f_{t-1}) \cdot f_t(\mathbf{z}_2 | \mathbf{x}, f_{t-1}) \cdot \frac{f_t(\mathbf{x} | f_{t-1})}{f_t(\mathbf{z}_1, \mathbf{z}_2 | f_{t-1})} \\ &= \frac{f_t(\mathbf{x} | \mathbf{z}_1, f_{t-1}) f_t(\mathbf{z}_1 | f_{t-1})}{f_t(\mathbf{x} | f_{t-1})} \cdot \frac{f_t(\mathbf{x} | \mathbf{z}_2, f_{t-1}) f_t(\mathbf{z}_2 | f_{t-1})}{f_t(\mathbf{x} | f_{t-1})} \cdot \frac{f_t(\mathbf{x} | f_{t-1})}{f_t(\mathbf{z}_1, \mathbf{z}_2 | f_{t-1})} \\ &= \frac{f_t(\mathbf{x} | \mathbf{z}_1, f_{t-1}) f_t(\mathbf{z}_1 | f_{t-1}) f_t(\mathbf{x} | \mathbf{z}_2, f_{t-1}) f_t(\mathbf{z}_2 | f_{t-1})}{f_t(\mathbf{x} | f_{t-1}) f_t(\mathbf{z}_1, \mathbf{z}_2 | f_{t-1})} \\ &= \frac{f_t(\mathbf{x} | \mathbf{z}_1, f_{t-1}) f_t(\mathbf{x} | \mathbf{z}_2, f_{t-1}) f_t(\mathbf{z}_1 | f_{t-1}) f_t(\mathbf{z}_2 | f_{t-1})}{f_t(\mathbf{x} | f_{t-1}) \int_{\mathcal{S}} f_t(\mathbf{z}_1, \mathbf{z}_2 | f_{t-1}, \mathbf{x}') f_t(\mathbf{x}' | f_{t-1}) d\mathbf{x}'} \end{aligned}$$

where \mathcal{S} denotes the space of the tracked state. This expression can be written as the product

$$f_t(\mathbf{x} | \mathbf{z}_1, \mathbf{z}_2, f_{t-1}) = k \cdot \frac{f_t(\mathbf{x} | \mathbf{z}_1, f_{t-1}) f_t(\mathbf{x} | \mathbf{z}_2, f_{t-1})}{f_t(\mathbf{x} | f_{t-1})}, \quad (3)$$

where $k = \frac{f_t(\mathbf{z}_1 | f_{t-1}) \cdot f_t(\mathbf{z}_2 | f_{t-1})}{\int_{\mathcal{S}} f_t(\mathbf{z}_1, \mathbf{z}_2 | \mathbf{x}', f_{t-1}) f_t(\mathbf{x}' | f_{t-1}) d\mathbf{x}'}$. It may be shown that $k = \frac{1}{\int_{\mathcal{S}} \frac{f_t(\mathbf{x}' | \mathbf{z}_1, f_{t-1}) f_t(\mathbf{x}' | \mathbf{z}_2, f_{t-1})}{f_t(\mathbf{x}' | f_{t-1})} d\mathbf{x}'}$, which does

not depend on the state \mathbf{x} . Note that the multiplication by k in (3) is equivalent to scaling $\int_{\mathcal{S}} f_t(\mathbf{x} | \mathbf{z}_1, \mathbf{z}_2, f_{t-1}) d\mathbf{x}$ to one. Now we see that A_1 and A_2 may be easily combined by multiplying their corresponding PDFs $f_t(\mathbf{x} | \mathbf{z}_1, f_{t-1})$ and $f_t(\mathbf{x} | \mathbf{z}_2, f_{t-1})$, dividing by the PDF $f_t(\mathbf{x} | f_{t-1})$, and scaling to have unit integral.

Note that even if the conditional independence assumption (1) is false, (3) is still reasonable: Applying Bayes’

rule, (3) may be rewritten as

$$f_t(\mathbf{x}|\mathbf{z}_1, \mathbf{z}_2, f_{t-1}) = k' \cdot f_t(\mathbf{x}|\mathbf{z}_1, f_{t-1})f_t(\mathbf{z}_2|\mathbf{x}, f_{t-1}), \quad (4)$$

where k' is a constant independent of \mathbf{x} . Even in the extreme case where, given the state \mathbf{x} , \mathbf{z}_2 is a function of \mathbf{z}_1 , falsely assuming conditional independence results in multiplying the correct PDF $f_t(\mathbf{x}|\mathbf{z}_1, \mathbf{z}_2, f_{t-1}) = f_t(\mathbf{x}|\mathbf{z}_1, f_{t-1})$ by the likelihood of the state $f_t(\mathbf{z}_2|\mathbf{x}, f_{t-1})$, which yields a sharpened version of the correct PDF.

The PDF $f_t(\mathbf{x}|f_{t-1})$ is usually referred to as the *prior* PDF – the PDF of the tracked state predicted prior to the measurements in the t^{th} frame. In our experiments, we make a worst case simplification and assume that no knowledge regarding the tracked object dynamics is given in the combination. (The separate tracking algorithms, however, do use a motion model.) Thus, we set the prior PDF to a uniform one, which allows us to treat the separate algorithms as “closed boxes”. The combined system is illustrated in Fig. 2.

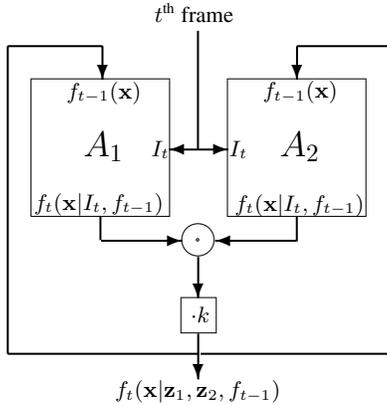


Figure 2: Combining A_1 and A_2 of Fig. 1: two explicit PDF-yielding tracking algorithms that use a common state space.

Algorithms A_1 and A_2 might, for example, be Kalman filters. If so, it can be shown (see [6]) that the combined PDF remains Gaussian $N(\mu, C)$, with the covariance matrix and mean being

$$\begin{aligned} C^{-1} &\triangleq C_1^{-1} + C_2^{-1} - C_3^{-1} \\ \mu &\triangleq C(C_1^{-1}\mu_1 + C_2^{-1}\mu_2 - C_3^{-1}\mu_3), \end{aligned} \quad (5)$$

where $N(\mu_1, C_1)$ and $N(\mu_2, C_2)$ are the PDFs provided by the separate algorithms, and $N(\mu_3, C_3)$ is the prior PDF. Being Gaussian, the combined PDF may be fed back into A_1 and A_2 . Note that treating the prior PDF as uniform (as in our experiments), is achieved by letting C_3 approach the identity matrix scaled to infinity. Then C_3^{-1} approaches the zero matrix, making the combined PDF $N(\mu, C)$ independent of the prior PDF, as expected.

In case the probability distributions are discrete and compact (or may be approximated as such), A_1 and A_2 may be general distribution filters [9].

3.2. Different state spaces

Often we can construct conditionally independent trackers on different state spaces, and then would like to use them to enhance the tracking performance. This is indeed possible (if the state spaces are related). In order to combine the algorithms we require that a probability distribution on any state space, conditioned on the state in any of the other state spaces, be supplied. In other words, if we denote the state space of algorithm A_i by \mathcal{S}_i , then for all $i \neq j$, $f_{\mathcal{S}_i \rightarrow \mathcal{S}_j}(\mathbf{x}_j|\mathbf{x}_i)$ where $\mathbf{x}_i \in \mathcal{S}_i$ and $\mathbf{x}_j \in \mathcal{S}_j$, has to be given. Another approximation we make here is that

$$f_t(\mathbf{x}_i|\mathbf{z}_j, f_{\mathbf{x}_i, t-1}) \approx f_t(\mathbf{x}_i|\mathbf{z}_j, f_{\mathbf{x}_j, t-1}), \quad (6)$$

where $f_{\mathbf{x}_i, t-1}$ is the PDF on state space \mathcal{S}_i at frame $t-1$. Note that this approximation is exact when no knowledge regarding the tracked object dynamics is given in the combination process, as is the case in our experiments.

We turn now to obtain the combination of A_1 and A_2 , two conditionally independent tracking algorithms, tracking an object in different state spaces. Denote by \mathcal{S}_1 and \mathcal{S}_2 the corresponding spaces, and denote by \mathbf{z}_1 and \mathbf{z}_2 the features used by A_1 and A_2 , respectively. The illustration in Fig. 1 given earlier for the case of a common state space applies also here, though note that the PDFs related to A_1 are on space \mathcal{S}_1 and the PDFs related to A_2 are on space \mathcal{S}_2 .

In order to combine the algorithms' estimations, each provided PDF has to be translated into a PDF in the state space of the other algorithm. This may be accomplished given the PDFs of the state in one space conditioned on the state in the other space – $f_{\mathcal{S}_2 \rightarrow \mathcal{S}_1}(\mathbf{x}|\mathbf{y})$ and $f_{\mathcal{S}_1 \rightarrow \mathcal{S}_2}(\mathbf{y}|\mathbf{x})$ ($\mathbf{x} \in \mathcal{S}_1$, $\mathbf{y} \in \mathcal{S}_2$). Using these PDFs we can compute $f_t(\mathbf{x}|\mathbf{z}_2, f_{\mathbf{y}, t-1})$ and $f_t(\mathbf{y}|\mathbf{z}_1, f_{\mathbf{x}, t-1})$. Relying on (6), these PDFs are approximately $f_t(\mathbf{x}|\mathbf{z}_2, f_{\mathbf{x}, t-1})$ and $f_t(\mathbf{y}|\mathbf{z}_1, f_{\mathbf{y}, t-1})$, respectively. Once we have the last two PDFs, we can combine the algorithms as in the case of the common state space. Observe that now the combined system has two outputs, one in each state space. See Fig. 3 for the combined system.

4. The CONDENSATION case

4.1. Same state space

Many tracking algorithms use CONDENSATION to propagate an approximated state PDF, represented by a sample. This approach is used when an analytic form of the PDF is unknown or unjustified. When using CONDENSATION, sample-sets of PDFs are given rather than explicit PDFs as in the previous cases. We use the same notations as in [5] and denote an N -sample-set of the tracked state PDF at time t by $\{\mathbf{s}_t^{(n)}, \pi_t^{(n)}, n = 1, \dots, N\}$, $\mathbf{s}_t^{(n)}$ being the sampled states and $\pi_t^{(n)}$ being the corresponding probabilities. Let C_1 and C_2 be two conditionally independent CONDENSATION-based algorithms using the same state space. Since

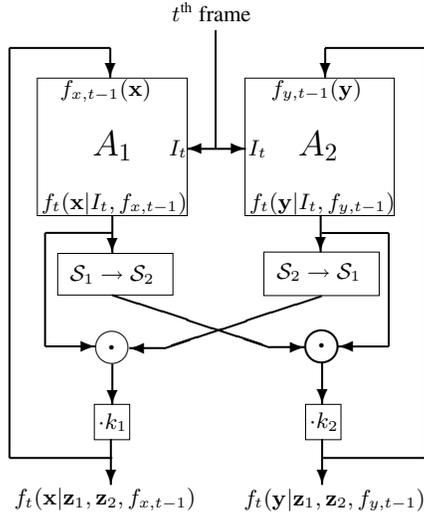


Figure 3: Combining A_1 and A_2 , two explicit PDF-yielding tracking algorithms using different state spaces.

both algorithms track the same object, they may use the same stochastic dynamics. In such a case, their prediction phases are identical, both in the deterministic drift and in the stochastic diffusion (see [5] for details). Denote by \mathbf{z}_1 and \mathbf{z}_2 the features used by C_1 and C_2 , respectively. The overall system of the two separate algorithms is illustrated in Fig. 4.

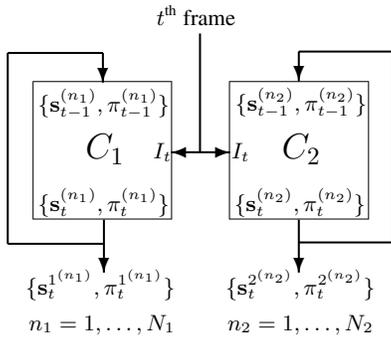


Figure 4: Two separate CONDENSATION-based tracking algorithms. Both algorithms track in the same state space.

To combine C_1 and C_2 we propose to set the number of samples in each algorithm to be the same ($N \triangleq N_1 = N_2$), and let the samples of both sample-sets to be of the same states, i.e., $\mathbf{s}_t^{(n)} \triangleq \mathbf{s}_t^{1(n)} = \mathbf{s}_t^{2(n)}$, $n = 1, \dots, N$. Then, the outputs of C_1 and C_2 may be combined into a single sample-set containing the same states by multiplying corresponding probabilities and normalizing such that $\sum_{i=1}^N \pi_t^{(i)} = 1$ (No division this time; proof

follows):

$$\{\mathbf{s}_t^{(n)}, \pi_t^{(n)} = k \pi_t^{1(n)} \pi_t^{2(n)}, n = 1, \dots, N\}, \quad (7)$$

$$\text{where } k = \left(\sum_{i=1}^N \pi_t^{1(i)} \pi_t^{2(i)} \right)^{-1}.$$

A sample-set of common states for the two algorithms may be achieved by initializing them with the same initial sample-set for the state at the first frame, and feeding them from the same random number generator. (We remind the reader that both algorithms have identical prediction phases, and thus the states in the sample-set remain identical in the two algorithms after the prediction phase.)

The combination rule (7) is justified by the equivalence of the combined system to a single CONDENSATION-based algorithm, whose selecting and predicting steps are identical to those in the separate algorithms, and whose measurement step is composed of the measurements of both C_1 and C_2 . The conditional observation density of the latter CONDENSATION-based algorithm is the product of the conditional observation densities of C_1 and C_2 . Since the observations made by C_1 are conditionally independent of the observations made by C_2 (Assumption (1)), this conditional observation density is correct. This concludes the correctness of the whole latter CONDENSATION-based algorithm, and thus also the correctness of combination (7).

4.2. Different state spaces

Consider now the combination of CONDENSATION-based algorithms corresponding to different state spaces. Unlike the previous cases, here we could not devise a method to combine the algorithms by solely using their inputs and outputs. In this combination, we need functions from *one* of the state spaces to all others. For example, a CONDENSATION-based algorithm that approximates the contour of the tracked object may be combined with a CONDENSATION-based algorithm estimating the bounding box of the tracked object, since there is a (straightforward) function relating a contour to a bounding box. Formally, in this case we need functions F_j relating one of the state spaces \mathcal{S}_i to all other state spaces \mathcal{S}_j . Another assumption we make here is that for *that* i and for all $j \neq i$

$$f(\mathbf{z}_j | \mathbf{x}_i) = f(\mathbf{z}_j | F_j(\mathbf{x}_i)), \quad \forall \mathbf{x}_i \in \mathcal{S}_i. \quad (8)$$

Notice that if the F_j s are bijections, no assumption is made, and if not, this assumption is very realistic; since naturally the observations made by any separate algorithm are dependent only on the object parameters sought by that specific algorithm.

We turn now to obtain the combination of C_1 and C_2 , two conditionally independent CONDENSATION-based tracking algorithms, using different state spaces. Denote by \mathcal{S}_1 and \mathcal{S}_2 the corresponding spaces, and denote by \mathbf{z}_1

and \mathbf{z}_2 the features used by C_1 and C_2 , respectively. The illustration in Fig. 4 given earlier for the case of a common state space may be appropriate also here, though one should bear in mind that the states in the sample-sets of C_1 are of space \mathcal{S}_1 and the states in the sample-sets of C_2 are of space \mathcal{S}_2 . Given a function F from one of the state spaces to the other, and noting that a CONDENSATION-based algorithm always has the “Select-Predict-Measure” phases, we can combine C_1 and C_2 by formulating a single CONDENSATION-based algorithm, the measurement phase of which is just a little more complicated. Without loss of generality, assume $F : \mathcal{S}_1 \rightarrow \mathcal{S}_2$. Then the composed observation density of \mathbf{z}_1 and \mathbf{z}_2 conditioned on the state $\mathbf{x}_1 \in \mathcal{S}_1$ and on the state $\mathbf{x}_2 = F(\mathbf{x}_1) \in \mathcal{S}_2$, $f(\mathbf{z}_1, \mathbf{z}_2 | \mathbf{x}_1, \mathbf{x}_2)$, equals $f(\mathbf{z}_1, \mathbf{z}_2 | \mathbf{x}_1, F(\mathbf{x}_1)) = f(\mathbf{z}_1, \mathbf{z}_2 | \mathbf{x}_1)$. By (1) the last term equals $f(\mathbf{z}_1 | \mathbf{x}_1) \cdot f(\mathbf{z}_2 | \mathbf{x}_1)$, which equals $f(\mathbf{z}_1 | \mathbf{x}_1) \cdot f(\mathbf{z}_2 | F(\mathbf{x}_1))$ by (8). We conclude for the composed conditional observation density:

$$f(\mathbf{z}_1, \mathbf{z}_2 | \mathbf{x}_1) = f(\mathbf{z}_1 | \mathbf{x}_1) \cdot f(\mathbf{z}_2 | F(\mathbf{x}_1)). \quad (9)$$

Letting $N = \max\{N_1, N_2\}$, the new CONDENSATION-based algorithm has a sample-set of size N in space \mathcal{S}_1 , and contains the following phases:

1. **Selection phase:** As in the selection phase of C_1 .
2. **Prediction phase:** As in the prediction phase of C_1 .
3. **Measurement phase:**
 - (a) Perform the measurement phase of C_1 .
 - (b) Perform the measurement phase of C_2 on the states in \mathcal{S}_2 corresponding to the states in \mathcal{S}_1 obtained in step 2.
 - (c) Multiply the probabilities obtained in step (a) with the corresponding probabilities obtained in step (b), and normalize to a unit sum to yield the final probabilities.

The obtained algorithm is illustrated in Fig. 5.

5. Experiments

We present here two experiments in which we used standard image sequences from *Performance Evaluation of Tracking and Surveillance* (PETS) workshops (see <http://www.visualsurveillance.org> and related links). The first experiment tested the combination of explicit PDF-yielding algorithms, where a pair of simple tracking algorithms using different state spaces of different dimensionality was combined. The second experiment tested the combination of CONDENSATION-based algorithms of different state spaces. Additional experimental results are given in [6]. All experiments’ results validate the significantly improved performance of the combined algorithm over the separate algorithms.

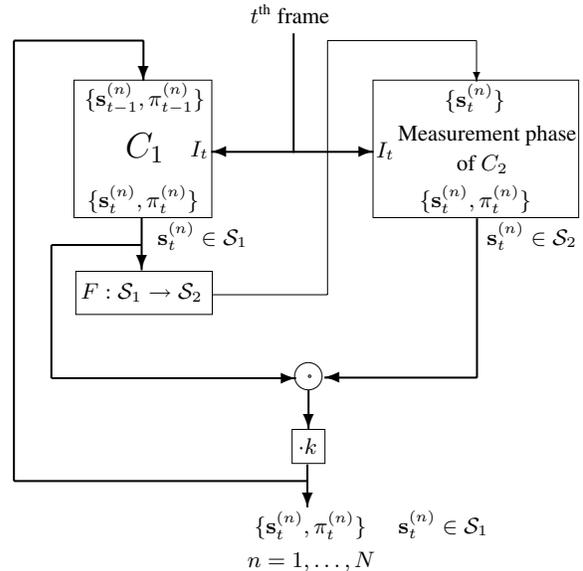


Figure 5: Combining C_1 and C_2 , two CONDENSATION-based tracking algorithms of different state spaces, into a single CONDENSATION-based algorithm.

Experiment I: Here we tracked a walking person in an outdoor scene, using a 1:2 down-sampled gray-level version of the image sequence from the PETS2000 workshop. Two simple tracking algorithms were implemented. The first algorithm tracks the center of the person by a template search of part of his body (the area of the stomach and chest). The state space is composed of four parameters – the 2D center coordinates in the current frame and in the previous frame. The second algorithm tracks the bounding box of the person using background differencing. The state space is composed of six parameters: The center, height and width of the box in the current frame, and the center in the previous frame. The two states were manually initialized, and propagated using a Kalman filter. A few of the frames with the tracking results imposed are shown in Fig. 6. The template-based algorithm fails already at the beginning due to an occlusion by a tree. Afterwards, the background differencing-based algorithm fails due to the proximity of the moving car to the person.

Next, the two algorithms are combined using only their outputs (Section 3.2). The PDF provided by the second algorithm is translated into the space of the first by making the PDF of the template center equal to the PDF of the bounding box center, and discarding the height and width parameters. For the other direction, the PDF of the bounding box center is made equal to the PDF of the template center, and augmenting that PDF with two additional independent Gaussian random variables with very large variances, as the height and width. The resulting PDFs remain Gaussian, and their Gaussian combinations (using (5)) make the

feedback to the Kalman filters feasible. The combined algorithm now overcomes all the situations in which the separate algorithms failed, and tracks the person successfully until he leaves the scene. Note that, as expected, a PDF provided by a successful tracking algorithm is not degraded when combined with the output of an unsuccessful algorithm, due to the implicit small weighting of the PDF of large covariances. A few representative frames are shown in Fig. 7.

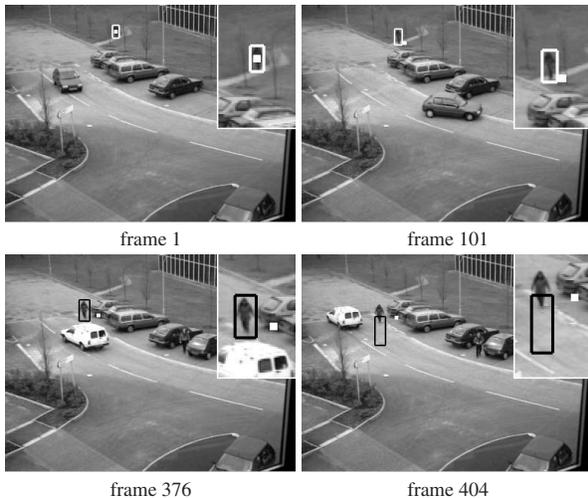


Figure 6: A few frames of the sequence used in Experiment I, with the tracking results of the two separate algorithms imposed (marked by boxes and dots). The template-based algorithm fails already in the beginning due to an occlusion by a tree. Afterwards, the background differencing based algorithm fails due to the proximity of the moving car to the tracked person.

Experiment II: This experiment consisted of tracking the contour of the head of a person taking part in a “smart meeting”. We used a 1:2 down-sampled version of an image sequence from the fourth PETS workshop (PETS-ICVS, scenario A, camera 2). The head’s shape is approximated as an ellipse (defined by five parameters: center coordinates, main and secondary axes, and angle). The state space consists of seven dimensions – the ellipse parameters and the velocity of its center. Unlike the previous experiment, the tracking is performed here via a CONDENSATION-based algorithm, using color edges as its observations: a hypothesized ellipse is weighted according to the amount of edges near the ellipse’s contour. The initialization is performed manually. A few of the frames with the tracking results imposed are shown in Fig. 8 (the mean of the sample-set is shown). Due to many edges in the interior of the tracked object region, the algorithm yields poor results.

In order to improve the tracking, we combined the last algorithm with another CONDENSATION-based algorithm, which tracks the vertical axis of the head (center coordinates

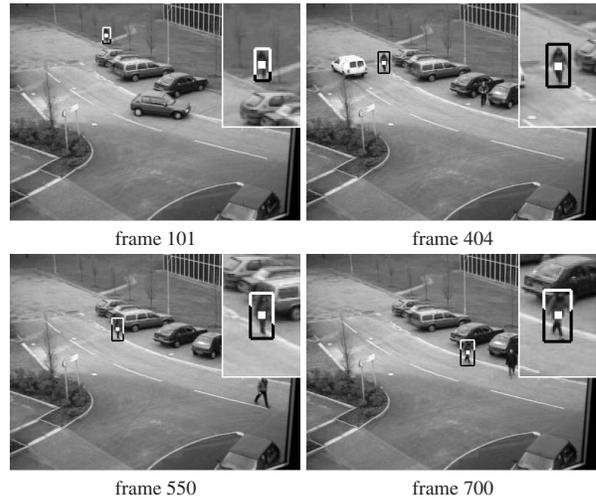


Figure 7: Results of Experiment I using the combined algorithm. The tracking succeeds until the person leaves the scene.

and length). The observations performed in this algorithm are the difference between the color at the top of the head’s vertical axis and the hair color, the difference between the color at the bottom of the head’s vertical axis and the skin color, and the difference between the color below the head’s vertical axis and the color of the shirt. The algorithms were combined as described in Section 4.2. The function F transforming a state from the space of the first algorithm into the space of the second sets the center of the vertical axis and its length to be the center and the major axis of the input ellipse, respectively. The composed algorithm yields better results, as may be seen in Fig. 9.

6. Conclusion

A general, analytically justified framework for combining tracking algorithms providing a PDF of the tracked state or a sample-set of it was developed. The framework proposed has the following significant advantages:

1. The framework is general. That is, it may be used for combining any set of tracking algorithms which provide a PDF estimate representation of the tracked object state as their output, even on different state spaces, under the assumption of conditional independence and some other few reasonable assumptions.

2. There is no need to devise auxiliary confidence measures for the output of each separate tracking algorithm in order to explicitly switch between them or differently weight them in the final estimate. The weighting is implicitly performed by the combination of the PDF representations. For example, if an algorithm provides an approximately uniform PDF containing no information on the tracked state, it does not affect the final PDF.



Figure 8: A few frames of the sequence used in Experiment II, with the tracking results of the edge-based algorithm imposed. Due to many edges in the interior of the tracked object region, the algorithm yields poor results.

3. The framework enables the combination of tracking algorithms using state spaces of different dimensionality. Thus, a tracker in a high-dimensional state space may be assisted by a tracker in a low-dimensional state space, and vice versa.

4. In most tracking algorithm combinations, at least one of the separate algorithms has to yield a unimodal estimate of the tracked state for the switching or correction of the other algorithms to be meaningful. Here, even if all the separate algorithms provide multi-modal PDFs, the combination may be done, and the combined PDF may still be unimodal.

5. In many settings (in particular, when the separate algorithms provide an explicit PDF of the tracked state), the proposed framework may be used by treating the separate algorithms as “closed boxes”. In other words, only the state distributions in their input and output are needed for the combination process. This may ease the implementation of the suggested framework from software aspects.

The framework was successfully tested using various state spaces and datasets.

References

- [1] T. Baker and M. Strens. Representation of uncertainty in spatial target tracking. In *ICPR 98*, pages 1339–1342, 1998.
- [2] A. Blake, R. Curwen, and A. Zisserman. A framework for spatio-temporal control in the tracking of visual contours. *IJCV*, 11(2):127–145, 1993.
- [3] R. Collins and Y. Liu. On-line selection of discriminative tracking features. *ICCV*, pages 346–352, 2003.



Figure 9: Results of Experiment II using the combined algorithm. The algorithm overcomes now the distractions caused by the edges inside the tracked head’s region.

- [4] S. Gil, R. Milanese, and T. Pun. Combining multiple motion estimates for vehicle tracking. *ECCV*, 2:307–320, 1996.
- [5] M. Isard and A. Blake. CONDENSATION - conditional density propagation for visual tracking. *IJCV*, 29(1):5–28, 1998.
- [6] I. Leichter, M. Lindenbaum, and E. Rivlin. A probabilistic framework for combining tracking algorithms. CIS-2004-02, Comp. Sci. Department, Technion – Israel Inst. Tech., 2004.
- [7] B. McCane, B. Galvin, and K. Novins. Algorithmic fusion for more robust feature tracking. *IJCV*, 49(1):79–89, 2002.
- [8] A. Papoulis. *Probability, Random Variables, and Stochastic Processes*. McGraw-Hill, 3rd edition, 1991.
- [9] Y. Rosenberg and M. Werman. A general filter for measurements with any probability distribution. *CVPR*, pages 654–659, 1997.
- [10] Y. Rosenberg and M. Werman. Representing local motion as a probability distribution matrix applied to object tracking. *CVPR*, pages 106–111, 1997.
- [11] K. Shearer, K.D. Wong, and S. Venkatesh. Combining multiple tracking algorithms for improved general performance. *PR*, 34(6):1257–1269, 2001.
- [12] N.T. Siebel and S. Maybank. Fusion of multiple tracking algorithms for robust people tracking. *ECCV*, 4:373–387, 2002.
- [13] M. Spengler and B. Schiele. Towards robust multi-cue integration for visual tracking. *Machine Vision and Applications*, 14:50–58, 2003.
- [14] K. Toyama and G.D. Hager. Tracker fusion for robustness in visual feature tracking. *SPIE*, 2569:38–49, 1995.
- [15] J. Triesch and von der C. Malsburg. Self-organized integration of adaptive visual cues for face tracking. *4th Intl. Conf. on Automatic Face and Gesture Recognition*, 2000.