Robust Gaussian Mixture Filter Based Mouth Tracking in a Real Environment

Friedrich Faubel, Munir Georges, Bo Fu, Dietrich Klakow
Spoken Language Systems,
Saarland University, D-66123 Saarbrücken, Germany
{friedrich.faubel, dietrich.klakow}@lsv.uni-saarland.de

***Abstract*—We have recently proposed a novel Gaussian mixture filter for nonlinear, non-Gaussian tracking problems. It is based on splitting and merging Kalman filters in order to increase the level of detail in likely regions of state space and reduce it in unlikely ones. In this work, we apply the above mentioned filter for tracking the mouth of a speaker under adverse conditions, i.e. in a shaky car environment, with off-the shelf camera equipment and severe compression artifacts. A Viola-Jones based detector identifies possible locations, which are then treated as multiple observations in a Bayesian filtering framework.**

I. Introduction

Tracking visual objects in a real environment is a nontrivial task. This is due to many reasons: light conditions can change abruptly; there might be compression artifacts; also, the cameras can shake when mounted on a movable object such as a laptop or, in our case, a car driving along a highway at 55 miles per hour [1]. The task of this work was to track the mouth of a speaker under such conditions – a necessary prerequisite for doing audio-visual speech recognition on the AVICAR [1] corpus. As tracking performance is crucial in this scenario, we developed a novel *Gaussian mixture filter* (GMF) [2], which, in contrast to the data association approaches taken in [3], [4], is truly able to handle concurrent hypotheses (multiple observations). Possible object locations are obtained with a Viola-Jones [5] detector, which we constrain to regions where the object is expected by the Bayesian filter. That discards unlikely locations and thereby increases the speed of both detection and tracking. The remaining part of this paper is organized as follows: In Section II we present the new filter. In Section III we describe the Viola-Jones based tracker.

II. Tracking with Multiple Observations

The objective of tracking can be formulated as to keep track of a system state x_t evolving in time t, where the evolution of the system state is considered to follow an underlying physical process, modeled by a process equation:

$$x_t = f_t(x_{t-1}, w_t), \;\; w_t \sim \mathcal{N}(w_t; 0, \Sigma_W). \tag{1}$$

In this equation, f_t is the state transition function, w_t is Gaussian process noise, which is introduced in order to account for uncertainties. In tracking applications, the system state itself is considered unknown but related to observations y_t through a measurement equation

$$y_t = h_t(x_t, v_t), \;\; v_t \sim \mathcal{N}(v_t; 0, \Sigma_V), \tag{2}$$

where h_t denotes the measurement function, v_t denotes Gaussian measurement noise. With these models at hand, we now briefly sketch the probably most well-known tracking algorithm: the *Kalman filter* (KF).

A. The Kalman Filter

For the Kalman filter, the functions f and h are required to be linear, as only then the filtering density – that is $p(x_t|y_{1:t})$ with $y_{1:t} \triangleq \{y_1, \ldots, y_t\}$ – remains Gaussian while being propagated in time. Then following [6], the operation of the Kalman filter can be described as:

1) **prediction**: constructing the joint predictive Gaussian distribution $p(x_t, y_t|y_{1:t-1})$ of the next state and observation, X_t and Y_t, given the process and measurement models – specified by (1) and (2) – as well as the observation history $y_{1:t-1}$.
2) **update**: conditioning that joint Gaussian distribution on the realized observation $Y_t = y_t$, in order to obtain $p(x_t|y_{1:t})$.

Alternating between these two steps propagates the filtering density $p(x_t|y_{1:t})$ in time. The minimum mean square error (MMSE) state estimate $\hat{x}_t$ is obtained by taking the expectation $E\{x_t|y_{1:t}\}$ of the filtering density.

B. Handling Multiple Observations by Splitting Filters

The Kalman filter was designed to receive a single observation y_t at time t. And though, in virtually any applied tracking scenario several, possible observations candidates $y_t = \{y_t^1, \ldots, y_t^{K_t}\}$ are available, some of which may be due to the object of interest, some of which may be due to clutter [1]. This gap is usually bridged by taking the single most likely observation or by applying the more sophisticated *Probabilistic Data Association Filter* (PDAF), which first combines the measurements in a weighted sum and according to their influence; then feeds the resulting, single observation into a Kalman filter [3]. This procedure is computationally simple, but clearly suboptimal as it disregards the individual hypotheses implicated by the observations. Moreover, the assumptions of the PDAF might be ill-suited to visual tracking, where objects can indeed spawn multiple observations and where clutter tends to occur in specific regions of the image. In this work, we try to approximate the ideal solution to the multiple observation problem, which consists in performing the update

[1] misdetections occurring at random, not related to the objects being tracked

stage of the KF once per observation. That can be achieved by duplicating each KF $K_t - 1$ times and then assigning each of the resulting filters to one of the observations. Propagating the filters as well as their posterior probabilities through time gives a Gaussian mixture filtering density:

Let there be N_{t-1} filters $\{\mathcal{K}_{t-1}^n | n = 1, \ldots, N_{t-1}\}$ at time $t-1$, with Gaussian filtering densities $p(\mathbf{x}_{t-1}|\mathcal{K}_{t-1}^n, y_{1:t-1})$ and weights $p(\mathcal{K}_{t-1}^n|y_{1:t-1})$. Then, at time t, all of the filtering densities are predicted according to step 1 of the KF; each $\mathcal{K}_{t-1}^n$ is split into K_t filters $\{\mathcal{K}_{t-1}^{n,k}|k = 1, \ldots, K_t\}$; $\mathcal{K}_{t-1}^{n,k}$ is assigned to the k-th observation y_t^k; and the corresponding filtering density $p(\mathbf{x}_t|\mathcal{K}_{t-1}^{n,k}, y_{1:t})$ is obtained by performing step 2 of the KF. Consequently, the overall filtering density $p(x_t|y_{1:t})$ becomes a Gaussian mixture:

$$p(x_t|y_{1:t}) = \sum_{n=1}^{N_{t-1}} \sum_{k=1}^{K_t} \underbrace{p(x_t|\mathcal{K}_{t-1}^{n,k}, y_{1:t}) p(\mathcal{K}_{t-1}^{n,k}|y_{1:t})}_{=p(\mathbf{x}_t, \mathcal{K}_t^{n,k}|y_{1:t})}, \quad (3)$$

where the $p(x_t|\mathcal{K}_{t-1}^{n,k}, y_{1:t})$ are multivariate Gaussian distributions and where the $p(\mathcal{K}_{t-1}^{n,k}|y_{1:t})$ are their weights. Similar as in [7], the weights – i.e. the posterior probabilities – can be evaluated with Bayes' rule:

$$p(\mathcal{K}_{t-1}^{n,k}|y_{1:t}) = \frac{p(y_t^k|\mathcal{K}_{t-1}^{n,k}, y_{1:t-1}) p(k) p(\mathcal{K}_{t-1}^n|y_{1:t-1})}{\sum_{k'=1}^{K_t} p(y_t^{k'}|\mathcal{K}_{t-1}^{n,k'}, y_{1:t-1}) p(k')},$$

where $p(\mathcal{K}_{t-1}^n|y_{1:t-1})$ is the filter's weight from the previous iteration and where $p(k) = 1/K_t$ is the prior probability, i.e. certainty, associated with observation y_t^k. The observation likelihood $p(y_t^k|\mathcal{K}_{t-1}^{n,k}, y_{1:t-1})$ of the (n,k)-th filter is obtained by marginalizing the joint predictive distribution from step 1 of the KF over x_t. At the end of each iteration, the filters are relabeled as $\mathcal{K}_t^l = \mathcal{K}_{t-1}^{n,k}$ with $l = (n-1)K_t + k$. After, N_t is set to $(N_{t-1} \cdot K_t)$ and t is incremented by 1.

C. Merging Filters

Running the ideal filter from above results in a Gaussian mixture filtering density, whose number of components grows exponentially in time. That is very impractical from a computational point of view. Hence, we reduce the number of components by merging the filters successively in pairs. For that, the filter $\mathcal{K}_t^n$ with the lowest posterior probability $p(\mathcal{K}_t^n)$ is selected and its similarity to all the other filters $\mathcal{K}_t^m$, $m \neq n$, is determined as described in [7]. Subsequently, $\mathcal{K}_t^n$ is merged with the most similar filter $\mathcal{K}_t^m$. Repeating this procedure until N filters remain limits the computational complexity to running $N \cdot \max(K_t)$ filters in parallel.

III. Viola-Jones Based Object Tracking

In order to track objects with a Kalman filter, we not only have to specify the process and measurement equations, but also need a detector that actually provides measurements of the specified type. In this work, we decided for the Viola-Jones detector [5] – a fast and accurate detector for visual objects that is based on a cascade of weak classifiers. The main motivation behind this choice is that by using Viola-Jones we avoid the problem of modeling and updating appearance (see [4]).

A. A Linear Model for Tracking

As a process model, we use a simple zeroth-order linear dynamic model that is based only on the object's position $\mathbf{p}_t$ as well as its scale s_t:

$$\begin{bmatrix} \mathbf{p}_t \\ s_t \end{bmatrix} = \begin{bmatrix} \mathbf{p}_{t-1} \\ s_{t-1} \end{bmatrix} + \mathcal{N}\left(\begin{bmatrix} \mathbf{0} \\ 0 \end{bmatrix}, \begin{bmatrix} \Sigma_{\Delta\mathbf{P}} & 0 \\ 0 & \Sigma_{\Delta S} \end{bmatrix} \right),$$

where $\Sigma_{\Delta\mathbf{p}}$ specifies the variability in position, $\Sigma_{\Delta S}$ specifies the variability in scale. For the measurements we consider a similar model with additive Gaussian noise.

B. Spatially Constrained Viola-Jones Detection

In initial experiments, the mouth location – as detected by the Viola-Jones detector – tended to jump in successive frames. In some cases, the mouth could not be found at all, probably due to poor image quality or due to the difficult lighting conditions in the driving car scenario. For these reasons, we tuned the detector to yield a large number of hypotheses and left it to the Bayesian filter to determine their relevance. That, however, caused a new problem: the large number of measurements immensely increased the computation time spent in the tracking algorithm. Hence, we decided to use the "gating" technique proposed in [8], which consists in calculating confidence ellipses for the predicted observation density $p(y_t|y_{1:t-1})$ from step 1 of the KF; and then discarding measurements that lie outside this region. We slightly modified this scheme by determining a rectangular region that includes all confidence ellipses – stemming from different filters – and then constraining the Viola-Jones detector to that region. The beauty of this approach is that the region searched by the detector is determined within the Bayesian filtering framework. Under normal conditions, only a small portion of the image is evaluated. With increasing state and, thereby, observation uncertainty, the search region increases automatically.

Acknowledgments

This work was supported by the Federal Republic of Germany through the Cluster of Excellence for Multimodal Computing and Interaction; and by the DFG under the IRTG 715 "Language Technology and Cognitive Systems".

References

[1] Bowon Lee et al., "AVICAR: Audio-visual speech corpus in a car environment," *Interspeech 2004*, pp. 2489–2492, Oct. 2004.

[2] D. L. Alspach and H. W. Sorenson, "Nonlinear Bayesian estimation using Gaussian sum approximations," *IEEE Trans. Autom. Control*, vol. 17, no. 4, pp. 439–448, Aug. 1972.

[3] C. Rasmussen and G. D. Hager, "Probabilistic data association methods for tracking complex visual objects," *IEEE Trans. PAMI*, vol. 6, no. 10, pp. 560–576, June 2001.

[4] X. Ren, "Finding people in archive films through tracking," *Proc. CVPR 2008*, June 2008.

[5] P. Viola and M. Jones, "Robust real-time object detection," *Intl. Workshop on Stat. and Comp. Theories of Vision*, July 2001.

[6] F. Faubel and D. Klakow, "A transformation-based derivation of the Kalman filter and an extensive unscented transform," *IEEE Workshop on Statistical Signal Processing*, Sept. 2009.

[7] F. Faubel, J. McDonough, and D. Klakow, "The split and merge unscented Gaussian mixture filter," *IEEE Signal Process. Lett.*, vol. 16, no. 9, pp. 786–789, Sept. 2009.

[8] Y. Bar-Shalom and T. Fortmann, *Tracking and Data Association*, Academic Press, 1988.