JOINT AUDIO-VIDEO OBJECT LOCALIZATION USING A RECURSIVE MULTI-STATE MULTI-SENSOR ESTIMATOR

N. Strobel, S. Spors, R. Rabenstein

Telecommunications Laboratory
University of Erlangen-Nuremberg
Cauerstrasse 7, 91058 Erlangen, Germany
E-mail: {strobel, spors, rabe}@LNT.de

ABSTRACT

Object localization based on audio and video information is important for the analysis of dynamic scenes such as video conferences or traffic situations. In this paper, we view the the dynamic audio-video object localization problem as a joint recursive estimation problem. It is solved using a decentralized Kalman filter fusing both audio and video position estimates. To better take into account different object maneuvers, multiple state-space equations are also incorporated. The result is a recursive multi-state multi-sensor estimator. Experiments show that it yields significantly improved joint position estimates compared to results achieved by using either an audio or a video system only.

1. INTRODUCTION

Various techniques exist both for the localization of sound sources by microphone arrays and for the tracking of visible objects in image sequences. All these methods inevitably suffer from reflections, background noise, illumination changes, and alike. Rather than improving upon localization techniques for a single modality, we present here a framework for joint localization combining audio and video signals.

We assume that audio and video observations are independent of each other, given the source position. This is to say that the audio measurements only depend on the speaker position and not on the video measurements and vice versa. Under this condition, we can decompose the joint estimation problem for the source position into two separate object localization problems based on audio and video measurements, respectively. Once acoustic and visual position estimates are available, they can be combined using a decentralized Kalman filter.

This paper is structured as follows. First, we concentrate on how the position estimates are obtained at the individual audio or video sensor. Then we show how they can be combined to a global estimate using either a single or multiple state models. In the next steps, we introduce the multi-state multi-sensor estimator and present the underlying system model. Afterwards experimental outcomes are provided to validate our approach. Finally we discuss our results and offer some conclusions.

2. RECURSIVE MULTI-SENSOR PARAMETER ESTIMATION

In this section we first show how the Kalman filter recursively computes position estimates using either audio or video measurements. Then we explain how to combine both estimates.

2.1. Separate State Estimates

In situations where the system dynamics can be described by a state-space model, the Kalman filter algorithm provides an efficient computational solution for estimating the state of a system. The discrete Kalman filter assumes that the state-space model is given through a linear stochastic difference equation and that measurements are provided through a linear measurement channel [1].

We describe the system dynamics of both audio ($i = 1$) and video ($i = 2$) observations by a general state space model

$$\mathbf{x}_i[k+1] = \mathbf{A}[k]\,\mathbf{x}_i[k] + \mathbf{b}[k]\,u[k] + \mathbf{v}_i[k] \quad (1a)$$

$$\mathbf{y}_i[k] = \mathbf{C}[k]\,\mathbf{x}_i[k] + \mathbf{n}_i[k]. \quad (1b)$$

The random variables $\mathbf{v}_i[k]$ and $\mathbf{n}_i[k]$ model the process and measurement noise. They are assumed to be independent of each other and from the system state $\mathbf{x}_i[k]$. Furthermore it is assumed that they are normally distributed with zero mean and covariance matrixes $\mathbf{R}_{vv}^{(i)}[k]$ and $\mathbf{R}_{nn}^{(i)}[k]$

$$\mathbf{v}_i[k] : N[\mathbf{0}, \mathbf{R}_{vv}^{(i)}[k]] \quad (2a)$$

$$\mathbf{n}_i[k] : N[\mathbf{0}, \mathbf{R}_{nn}^{(i)}[k]]. \quad (2b)$$

Note that we assume identical state space models for the audio and video system. They only differ in the additive noise components. Since the local states $\mathbf{x}_i[k]$ are driven by the local observations $\mathbf{y}_i[k]$, $i = 1, 2$, the two local state vectors are, however, usually different.

Internal Consistency Check

To avoid the assimilation of estimation errors, a malfunctioning sensor must be detected. To this end we perform an internal con-

sistency check by inspecting the innovation sequence

$$\boldsymbol{\eta}_i[k] = \left[\mathbf{y}_i[k] - \mathbf{C}[k]\hat{\mathbf{x}}_i[k|k-1]\right]. \quad (3)$$

In Eq. (3), the variable $\hat{\mathbf{x}}_i[k|k-1]$ denotes the a priori state estimate at the i-th sensor. The actual measurement of the i-th sensor is compared with the predicted measurement $\mathbf{C}[k]\hat{\mathbf{x}}_i[k|k-1]$ of the i-th sensor. The consistency check requires that the statistical properties of the innovation sequence $\boldsymbol{\eta}_i[k]$ are monitored. Under normal conditions the mean and the covariance matrix of the innovation sequence can be calculated form the distributions of the variables involved. Ideally, the vector $\boldsymbol{\eta}_i[k]$ should be normally distributed with zero mean and covariance matrix $\mathbf{P}_{\eta_i\eta_i}[k]$ [5]. This implies that the scalar $\boldsymbol{\eta}_i^T[k]\mathbf{P}_{\eta_i\eta_i}^{-1}[k]\boldsymbol{\eta}_i[k]$ is χ^2 distributed. To verify if this is actually the case, a χ^2 test is applied. The resulting value is called the *consistency statistic* $K[k]$. If $K[k]$ falls in between the limits $a(\beta)$ and $b(\beta)$, the measured statistics are assumed to agree with their theoretical counterparts. In this case, the measurements are said to be consistent, and the state estimate is accepted. Otherwise its counterpart as predicted by the Kalman filter is used. Note that $a(\beta)$ and $b(\beta)$ dependent on a preselected false-alarm probability β.

2.2. Joint State Estimation

In the previous subsections we introduced two separate position estimates. The first position estimate was based on audio measurements, and the second one included video observations. This section shows how to arrive at a joint position estimate using a decentralized Kalman filter recursively combining both audio and video modalities [2].

2.2.1. Single State Model

The decentralized Kalman filter (DKF) as used for the fusion of audio and video position estimates is a multi-sensor Kalman filter that has been divided up into two modules associated with the audio system and with the video system, respectively. Each node computes a local a posteriori estimate, $\hat{\mathbf{x}}_i[k|k],\ i = 1, 2$, of the object position. These partial estimates are finally assimilated to provide a global a posteriori estimate $\hat{\mathbf{x}}[k|k]$ in the fusion center. Figure 1 illustrates the structure of the decentralized Kalman filter.

The time-update equations and measurement-update equations of a DKF with M sensors can, e.g., be found in [2]. If the measurement noise components of audio and video observations are independent, the centralized state estimate can be separated. Then the global state-space equation can be expressed in the same way as the local system dynamics, i.e.,

$$\mathbf{x}[k+1] = \mathbf{A}\,\mathbf{x}[k] + \mathbf{b}\,u[k] + \mathbf{v}[k]. \quad (4)$$

Only the global noise component $\mathbf{v}[k]$ differs.

Hashemipour et al. showed in [2] that the global a posteriori state estimate can be expressed as

$$\hat{\mathbf{x}}[k|k] = \mathbf{P}[k|k]\left(\mathbf{P}^{-1}[k|k-1]\,\hat{\mathbf{x}}[k|k-1] + \sum_{i=1}^{2}\{\mathbf{P}_i^{-1}[k|k]\,\hat{\mathbf{x}}_i[k|k] - \mathbf{P}_i^{-1}[k|k-1]\,\hat{\mathbf{x}}_i[k|k-1]\}\right). \quad (5)$$

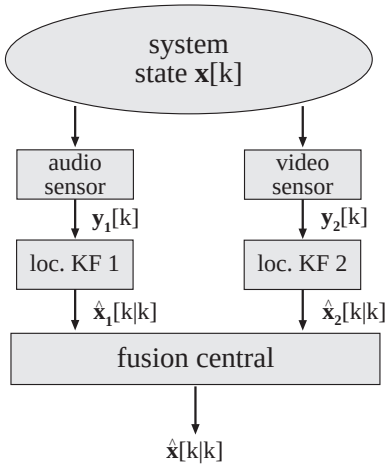

Figure 1: Structure of the decentralized Kalman filter. Two local Kalman filters (KF) provide their estimates to the fusion center. The fusion center combines the local estimates to compute a global estimate of the system state.

The matrices $\mathbf{P}[k|k-1]$ and $\mathbf{P}[k|k]$ denote the global a priori and a posteriori error estimate covariances, respectively, while $\mathbf{P}_i[k|k-1]$ and $\mathbf{P}_i[k|k],\ i = 1, 2$, are their counterparts at the two local processors. The vector $\hat{\mathbf{x}}[k|k-1]$ is the global a priori state estimate, and $\hat{\mathbf{x}}_i[k|k-1]$ together with $\hat{\mathbf{x}}_i[k|k],\ i = 1, 2$, denote the local a priori and local a posteriori state estimates, respectively. The second term on the right hand side in Eq. (5) involving the intermediate state estimates can be viewed as a *state error information* vector.

The global a posteriori error covariance is given by

$$\mathbf{P}^{-1}[k|k] = \mathbf{P}^{-1}[k|k-1] + \sum_{i=1}^{2}\{\mathbf{P}_i^{-1}[k|k] - \mathbf{P}_i^{-1}[k|k-1]\}. \quad (6)$$

Equations (5) and (6) summarize the parallel Kalman filter algorithm. In the measurement-update equation (5), the fusion center needs the central a priori state estimate, $\hat{\mathbf{x}}[k|k-1]$, the associated global a priori covariance matrix, $\mathbf{P}[k|k-1]$, the a posteriori covariance matrix, $\mathbf{P}[k|k]$, and the state error information vector together with variance error information matrix. There is no need for communications from the local processors to the fusion center during the prediction-update, provided it can store the matrices $\mathbf{A}$, $\mathbf{b}$, and $\mathbf{C}_i$.

Theoretically, there is no performance loss in the decentralized system. However, the algorithm does assume that the local processors work in sync at the same speed. In general, this cannot be assumed. A solution to the problem of asynchronous operation can be found in [5].

2.2.2. Multiple State Models

In situations where objects can perform different types of motion, it will be difficult to find one state-space model that always fits. Assuming that we can find appropriate state-space models for different parts of the object trajectory, an adaptive Kalman filter can learn from the measurements which of these models is the right

one. In [1] is shown that for multiple state-space models the optimal a posteriori state estimate is given by

$$\hat{\mathbf{x}}[k|k] = \sum_{i=1}^{L} \hat{\mathbf{x}}_{\alpha_i}[k|k] f(\alpha_i|\mathbf{Y}[k]) \tag{7}$$

where α_i denotes the state-space model used. The optimal a posteriori state estimate $\hat{\mathbf{x}}[k|k]$ is the sum of the a posteriori state estimates $\hat{\mathbf{x}}_{\alpha_i}[k|k]$ of Kalman filters incorporating the model α_i weighted with the model probability $f(\alpha_i|\mathbf{Y}[k])$. Each of these filters can then be implemented using a decentralized Kalman filter. Note that all Kalman filters are observing the same measurement sequence

$$\mathbf{Y}[k] = [\mathbf{y}[k] \ \ldots \ \mathbf{y}[0]] \,. \tag{8}$$

Figure 2 shows a block diagram of the adaptive Kalman filter. The model probability can be computed recursively [1].

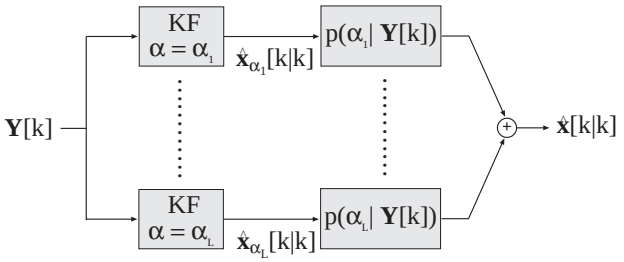

Figure 2: Block diagram of the adaptive Kalman filter

2.3. The Extended Kalman Filter

In the previous section, we described the decentralized (or parallelized) Kalman filter for linear systems. For nonlinear systems, a decentralized extended Kalman filter must be used. To this end, we rewrite the nonlinear plant equation

$$\mathbf{x}[k+1] = \mathbf{f}(\mathbf{x}[k],\ \mathbf{u}[k],\ \mathbf{v}[k]). \tag{9}$$

by introducing local states $\mathbf{x}_i[k]$ and mutually independent process noise components, $\mathbf{v}_i[k]$. Following the construction of the decentralized linear Kalman filter, we assume identical nonlinear plant equations $\mathbf{f}(.)$, and identical control inputs, $\mathbf{u}[k]$ at the two local processors. The result is

$$\mathbf{x}_i[k+1] = \mathbf{f}(\mathbf{x}_i[k],\ \mathbf{u}[k],\ \mathbf{v}_i[k]), \quad i = 1,\ 2. \tag{10}$$

The measurement models of the distributed sensors need, however, not be identical, i.e., different nonlinear measurement equations, $\mathbf{h}_i(.)$, are possible. In the case of two distributed sensors, we get

$$\mathbf{y}_i[k] = \mathbf{h}_i(\mathbf{x}_i[k],\ \mathbf{n}_i[k]), \quad i = 1,\ 2. \tag{11}$$

In Eq. (11), the measurement noise components, $\mathbf{n}_i[k]$, are assumed to be Gaussian and mutually independent.
Ideally, the final state estimate after fusing all individual nonlinear estimates should be identical to the centralized state estimate. Due to the nonlinear equations, a general answer to this problem appears difficult, and, at least to the knowledge of the authors, no solution to has been presented so far.

3. MULTI-STATE MULTI-SENSOR ESTIMATION

Figure 3 shows a block diagram of the multi-state multi-sensor estimator. This configuration is composed of two decentralized Kalman filters. Each decentralized Kalman filter comprises an audio and a video node both recursively computing position estimates. The two decentralized Kalman filters differ in their underlying state-space models, and each fusion center computes a global estimate. The final, "universal", estimate follows as a weighted av-

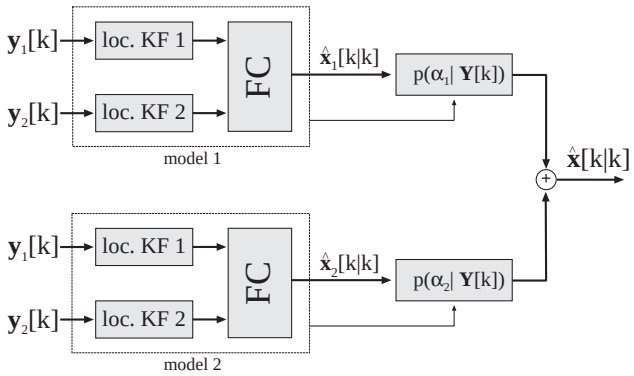

Figure 3: Block diagram of a multi-state multi-sensor estimator.

erage of the two global estimates. The weighting may be thought of as a mechanism prefering that joint a posteriori estimate $\hat{\mathbf{x}}_i[k|k]$, $i = 1,\ 2$, whose underlying state space model better matches the current object motion. To sort out unreliable estimates due to malfunctioning sensors, a consistency check is performed in the two fusion centers as well. Note that the configuration shown in Fig. 3 could be based on more than two state-space models.

4. SYSTEM MODELS

4.1. State Models

To track a real object using a Kalman filter, a suitable motion model is needed. Since it is difficult to accurately describe complex object maneuvers, we use a linear model as a first approximation instead. Assuming constant object speed and a Cartesian coordinate system, our state-space equation can be expressed as

$$\underbrace{\begin{bmatrix} x_x[k+1] \\ v_x[k+1] \\ x_y[k+1] \\ v_y[k+1] \end{bmatrix}}_{\mathbf{x}[k+1]} = \underbrace{\left[\begin{array}{cc|cc} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ \hline 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{array}\right]}_{\mathbf{A}[k]} \underbrace{\begin{bmatrix} x_x[k] \\ v_x[k] \\ x_y[k] \\ v_y[k] \end{bmatrix}}_{\mathbf{x}[k]} + \mathbf{v}[k], \tag{12}$$

where T is the sampling interval and x_x, x_y, v_x, v_y, are the horizontal and vertical components of the object position and velocity.

4.2. Measurement Models

A Kalman filter requires a model for the measurement channel. In this paper, we consider object localization using audio and video data. Audio object localization is based on a steered beamformer [3], while video object localization relies on skin color detection [4].

Both algorithms use the same measurement model

$$\underbrace{\begin{bmatrix} y_x[k] \\ y_y[k] \end{bmatrix}}_{\mathbf{y}[k]} = \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}}_{\mathbf{C}[k]} \underbrace{\begin{bmatrix} x_x[k] \\ v_x[k] \\ x_y[k] \\ v_y[k] \end{bmatrix}}_{\mathbf{x}[k]} + \mathbf{n}[k], \qquad (13)$$

but the audio and video measurement noise covariance matrices, $\mathbf{R}_{nn}^{(i)}[k]$, $i = 1, 2$, differ. In case of audio object localization which takes place in a polar coordinate system, the covariance matrix $\mathbf{R}_{nn}^{(1)}[k]$ is a diagonal matrix depending on the object distance. Although no longer diagonal, the video covariance matrix $\mathbf{R}_{nn}^{(2)}[k]$ also depends on the object distance from the focal plane.

5. EXPERIMENTAL RESULTS

For the experiment, the multi-state multi-sensor configuration shown in Fig. 3 was used to estimate the position of a whistling model railway moving along an oval track. The local measurements were obtained using a steered beamformer [3] and a skin color detector [4]. The audio estimator operates in a polar coordinate system to compute object positions $\mathbf{x}_1[k]$. The video estimates, $\mathbf{x}_2[k]$, on the other hand, are expressed in Cartesian coordinates. The joint estimate, $\hat{\mathbf{x}}[k|k]$, however, is again expressed in polar coordinates. To recursively combine the outputs of the two local Kalman filters, the video position estimates are transformed into polar coordinates. As a consequence, the measurement equations for the associated video Kalman filters become nonlinear, and the recursive video estimators have to be implemented as extended Kalman filters.

Two state-space models were implemented as shown in Fig. 3. The first one implies constant speed of the object and is given through Eq. (12). It is set up with the appropriate covariance matrix $\mathbf{R}_{vv}[k]$. The second model is based upon the same state-space model, only the entries of the associated covariance matrix are multiplied with a factor 100. Due to the increased noise level in the state-space equation, the Kalman filter using the second motion model relies more heavily on the measurements. Thus, it provides better results in situations where the first state model no longer fits.

Figure 4 shows the results of a simulation where the railway was moving with constant speed. Note that the multi-state multi-sensor estimator provides a reliable estimate of the object position at all time.

To quantify the position errors at the audio and video position estimators, we introduce the Euclidean distance between the true position $\mathbf{x}[k]$ and its associated a posteriori audio/video estimate $\hat{\mathbf{x}}_i[k|k]$ at time k

$$d_i[k] = ||\mathbf{x}[k] - \hat{\mathbf{x}}_i[k|k]|| \,. \qquad (14)$$

Similarly, we measure the Euclidean distance between $\mathbf{x}[k]$ and the "universal" a posteriori estimate $\hat{\mathbf{x}}[k|k]$

$$d[k] = ||\mathbf{x}[k] - \hat{\mathbf{x}}[k|k]|| \,. \qquad (15)$$

Their variances are ${\sigma_{d_1}}^2 = 1.8 \cdot 10^{-3}\mathrm{m}^2$, ${\sigma_{d_2}}^2 = 2.4 \cdot 10^{-4}\mathrm{m}^2$, and ${\sigma_d}^2 = 1.5 \cdot 10^{-4}\mathrm{m}^2$. We see that the audio position estimates are rather unreliable compared to what can be achieved with the video system. Yet the use of a multi-state multi-sensor estimator combining both modalities yields joint position estimates which

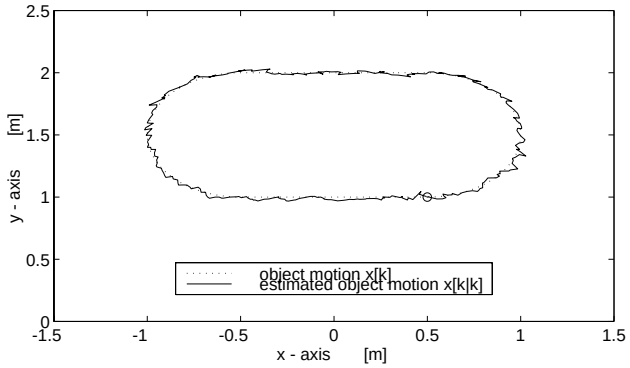

Figure 4: Simulation of a railway moving on an oval track

are almost 40% more accurate than the video position estimates on average. Another benefit of a multi-sensor system is its increased robustness. Even if one sensor fails, a sensible global position estimate may still be computed. To this end, another consistency check at the fusion center is to be performed.

6. DISCUSSION AND CONCLUSIONS

We showed how to apply a decentralized Kalman filter to the problem of dynamic object localization using separate audio and video sensors. Each sensor recursively computes a local position estimate. Both estimates are then fused using a decentralized Kalman filter. To take into account multiple motion models, we introduced a multi-state multi-sensor configuration of the Kalman filter. Our simulation results showed that the multi-state multi-sensor estimator yields position estimates which are almost 40% more accurate than what can be obtained with the best single (video) sensor. We thus conclude, that single sensor estimations can be successfully improved by a second modality, even when the second sensor estimates are of inferior quality.

Acknowledgements. This work is part of the ongoing Sonderforschungsbereich (SFB) No. 603 being carried out at the University of Erlangen-Nürnberg. It is supported by the Deutsche Forschungsgemeinschaft (DFG).

7. REFERENCES

[1] Robert Grover Brown and Patrick Y.C. Hwang. *Introduction to random signals and applied Kalman filtering*. Wiley, 1997.

[2] H. R. Hashemipour, S. Roy, and A. J. Laub. Decentralized structures for parallel Kalman filtering. *IEEE Transactions on Automatic Control*, 33(1):88–93, 1988.

[3] N. Strobel, T. Meier, and R. Rabenstein. Speaker localization using steered filtered-and-sum beamformers. In *Proc. Erlangen Workshop on Vision, Modeling, and Visualization*, Erlangen, 1999.

[4] R. J. Qian, M. I. Sezan, and K. E. Matthews. A robust real-time face tracking algorithm. In *Proc. IEEE Int. Conference on Image Processing*, volume 1, pp. 131–135, Chicago, 1998.

[5] B. S. Rao, H. F. Durrant-Whyte, and J. A. Sheen. A fully decentralized multi-sensor system for tracking and surveillance. *Int. Journal of Robotics Research*, 12(1):20–44, 1993.