A Dynamic Stream Weight Backprop Kalman Filter for Audiovisual Speaker Tracking

ICASSP 2020

Christopher Schymura, Tsubasa Ochiai, Marc Delcroix, Keisuke Kinoshita, Tomohiro Nakatani, Shoko Araki and Dorothea Kolossa

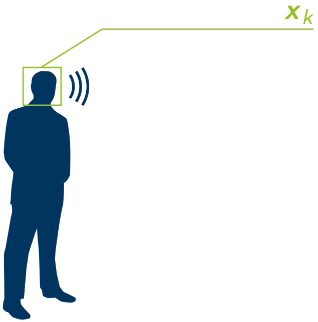
May 4-8 2020

RUHR UNIVERSITÄT BOCHUM
RUB
NTT

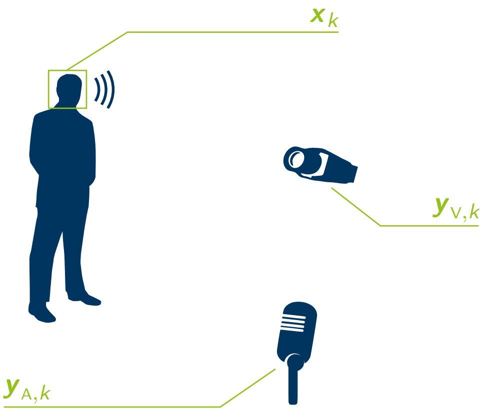
Problem statement

Problem statement

Problem statement



Problem statement

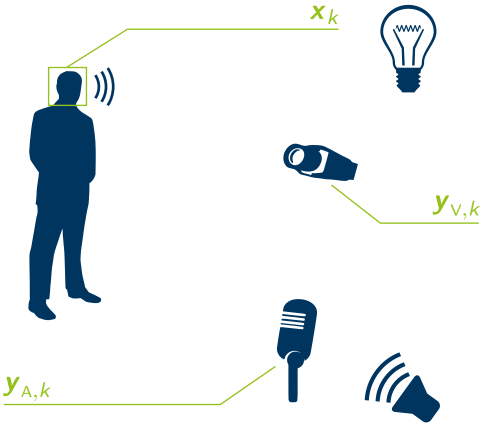


Observation functions:

$$\boldsymbol{y}_{\mathrm{A},k} = \boldsymbol{C}_{\mathrm{A}}\boldsymbol{x}_k$$

$$\boldsymbol{y}_{\mathrm{V},k} = \boldsymbol{C}_{\mathrm{V}}\boldsymbol{x}_k$$

Problem statement

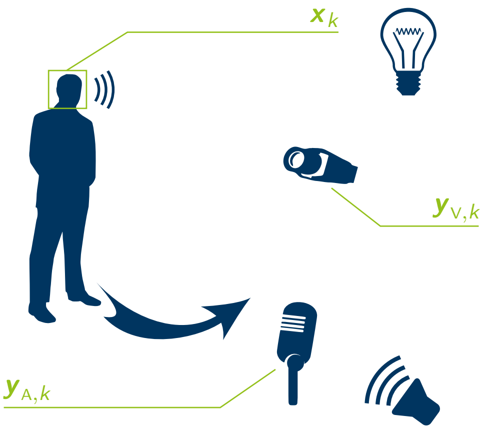


Observation functions:

$$\boldsymbol{y}_{\mathrm{A},k} = \boldsymbol{C}_{\mathrm{A}}\boldsymbol{x}_k + \boldsymbol{w}_{\mathrm{A},k}$$

$$\boldsymbol{y}_{\mathrm{V},k} = \boldsymbol{C}_{\mathrm{V}}\boldsymbol{x}_k + \boldsymbol{w}_{\mathrm{V},k}$$

Problem statement



State transition function:

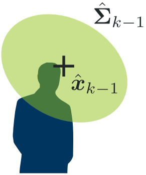
$$\boldsymbol{x}_k = \boldsymbol{A}\boldsymbol{x}_{k-1} + \boldsymbol{v}_k$$

Observation functions:

$$\boldsymbol{y}_{\mathrm{A},k} = \boldsymbol{C}_{\mathrm{A}}\boldsymbol{x}_k + \boldsymbol{w}_{\mathrm{A},k}$$

$$\boldsymbol{y}_{\mathrm{V},k} = \boldsymbol{C}_{\mathrm{V}}\boldsymbol{x}_k + \boldsymbol{w}_{\mathrm{V},k}$$

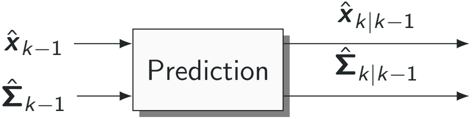
Recursive state estimation



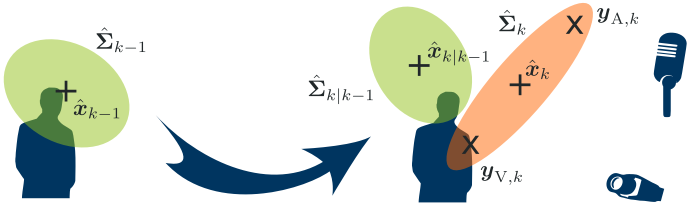
$\hat{\boldsymbol{x}}_{k-1}$

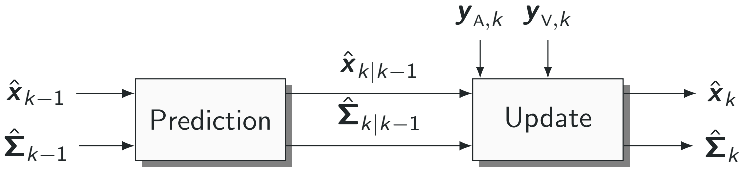
$\hat{\boldsymbol{\Sigma}}_{k-1}$

Recursive state estimation



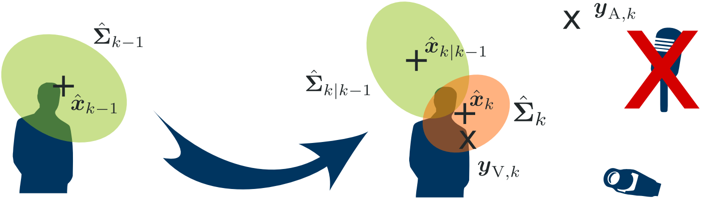
Recursive state estimation

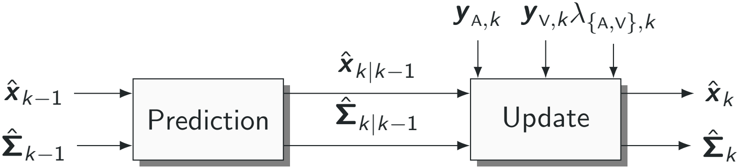




$$\underbrace{p(\mathbf{x}_k|\mathbf{Y}_{A,1:k}, \mathbf{Y}_{V,1:k})}_{\text{Posterior}} \propto \underbrace{p(\mathbf{x}_k|\mathbf{Y}_{A,1:k-1}, \mathbf{Y}_{V,1:k-1})}_{\text{Prior}} \underbrace{p(\mathbf{y}_{A,k}, \mathbf{y}_{V,k}|\mathbf{x}_k)}_{\text{Sensor model}}$$

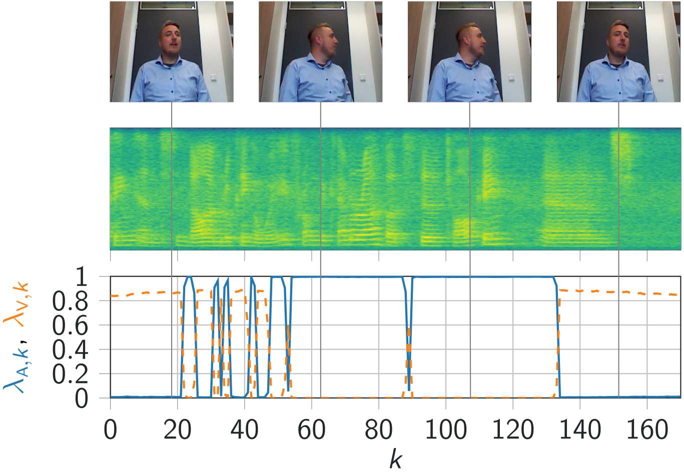
Recursive state estimation





$$\underbrace{p(\boldsymbol{x}_k|\boldsymbol{Y}_{\mathrm{A},1:k},\boldsymbol{Y}_{\mathrm{V},1:k})}_{\text{Posterior}} \propto \underbrace{p(\boldsymbol{x}_k|\boldsymbol{Y}_{\mathrm{A},1:k-1},\boldsymbol{Y}_{\mathrm{V},1:k-1})}_{\text{Prior}} \underbrace{p(\boldsymbol{y}_{\mathrm{A},k}|\boldsymbol{x}_k)^{\lambda_{\mathrm{A},k}}\, p(\boldsymbol{y}_{\mathrm{V},k}|\boldsymbol{x}_k)^{\lambda_{\mathrm{V},k}}}_{\text{Sensor model w. stream weights}}$$

Dynamic stream weights



Inference

Prediction step (identical to standard Kalman filter)

$$\hat{\boldsymbol{x}}_{k|k-1} = \boldsymbol{A}\hat{\boldsymbol{x}}_{k-1}$$
$$\hat{\boldsymbol{\Sigma}}_{k|k-1} = \boldsymbol{A}\hat{\boldsymbol{\Sigma}}_{k-1}\boldsymbol{A}^{\mathrm{T}} + \boldsymbol{Q}$$

Inference

Prediction step (identical to standard Kalman filter)

$$\hat{\boldsymbol{x}}_{k|k-1} = \boldsymbol{A}\hat{\boldsymbol{x}}_{k-1}$$
$$\hat{\boldsymbol{\Sigma}}_{k|k-1} = \boldsymbol{A}\hat{\boldsymbol{\Sigma}}_{k-1}\boldsymbol{A}^{\mathrm{T}} + \boldsymbol{Q}$$

Update step[1]

$$\begin{bmatrix} \boldsymbol{K}_{\mathrm{A},k}^{\mathrm{T}} \\ \boldsymbol{K}_{\mathrm{V},k}^{\mathrm{T}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{R}_{\mathrm{A}} + \lambda_{\mathrm{A},k}\boldsymbol{C}_{\mathrm{A},k}\hat{\boldsymbol{\Sigma}}_{k|k-1}\boldsymbol{C}_{\mathrm{A},k}^{\mathrm{T}} & \lambda_{\mathrm{V},k}\boldsymbol{C}_{\mathrm{A},k}\hat{\boldsymbol{\Sigma}}_{k|k-1}\boldsymbol{C}_{\mathrm{V},k}^{\mathrm{T}} \\ \lambda_{\mathrm{A},k}\boldsymbol{C}_{\mathrm{V},k}\hat{\boldsymbol{\Sigma}}_{k|k-1}\boldsymbol{C}_{\mathrm{A},k}^{\mathrm{T}} & \boldsymbol{R}_{\mathrm{V}} + \lambda_{\mathrm{V},k}\boldsymbol{C}_{\mathrm{V},k}\hat{\boldsymbol{\Sigma}}_{k|k-1}\boldsymbol{C}_{\mathrm{V},k}^{\mathrm{T}} \end{bmatrix}^{-1} \begin{bmatrix} \boldsymbol{C}_{\mathrm{A},k} \\ \boldsymbol{C}_{\mathrm{V},k} \end{bmatrix} \hat{\boldsymbol{\Sigma}}_{k|k-1}$$

$$\hat{\boldsymbol{x}}_k = \hat{\boldsymbol{x}}_{k|k-1} + \sum_{i \in \{\mathrm{A},\mathrm{V}\}} \lambda_{i,k}\boldsymbol{K}_{i,k}\left(\boldsymbol{y}_{i,k} - \boldsymbol{C}_i\hat{\boldsymbol{x}}_{k|k-1}\right)$$

$$\hat{\boldsymbol{\Sigma}}_k = \left(\boldsymbol{I} - \sum_{i \in \{\mathrm{A},\mathrm{V}\}} \lambda_{i,k}\boldsymbol{K}_{i,k}\boldsymbol{C}_i\right)\hat{\boldsymbol{\Sigma}}_{k|k-1}$$

[1] Christopher Schymura and Dorothea Kolossa. "Audiovisual Speaker Tracking using Nonlinear Dynamical Systems with Dynamic Stream Weights". In: IEEE/ACM Transactions on Audio, Speech, and Language Processing, 2020

DSW-KF: Benefits and remaining challenges

Kalman filter framework provides uncertainty information ...

DSW-KF: Benefits and remaining challenges

Kalman filter framework provides uncertainty information ...

Dynamic stream weights yield an additional layer of interpretability ...

DSW-KF: Benefits and remaining challenges

Kalman filter framework provides uncertainty information ...

Dynamic stream weights yield an additional layer of interpretability ...

Computationally efficient compared to Monte Carlo methods ...

DSW-KF: Benefits and remaining challenges

Kalman filter framework provides uncertainty information ...

Dynamic stream weights yield an additional layer of interpretability ...

Computationally efficient compared to Monte Carlo methods ...

Linear Gaussian assumption is a strong constraint ...

DSW-KF: Benefits and remaining challenges

Kalman filter framework provides uncertainty information ...

Dynamic stream weights yield an additional layer of interpretability ...

Computationally efficient compared to Monte Carlo methods ...

Linear Gaussian assumption is a strong constraint ...

Dynamic stream weights must be predicted from sensor data ...

DSW-KF: Benefits and remaining challenges

Kalman filter framework provides uncertainty information ...

Dynamic stream weights yield an additional layer of interpretability ...

Computationally efficient compared to Monte Carlo methods ...

Linear Gaussian assumption is a strong constraint ...

Dynamic stream weights must be predicted from sensor data ...

Dealing with high-dimensional observations is not straightforward ...

DSW-KF: Benefits and remaining challenges

Kalman filter framework provides uncertainty information ...

Dynamic stream weights yield an additional layer of interpretability ...

Computationally efficient compared to Monte Carlo methods ...

Linear Gaussian assumption is a strong constraint ...

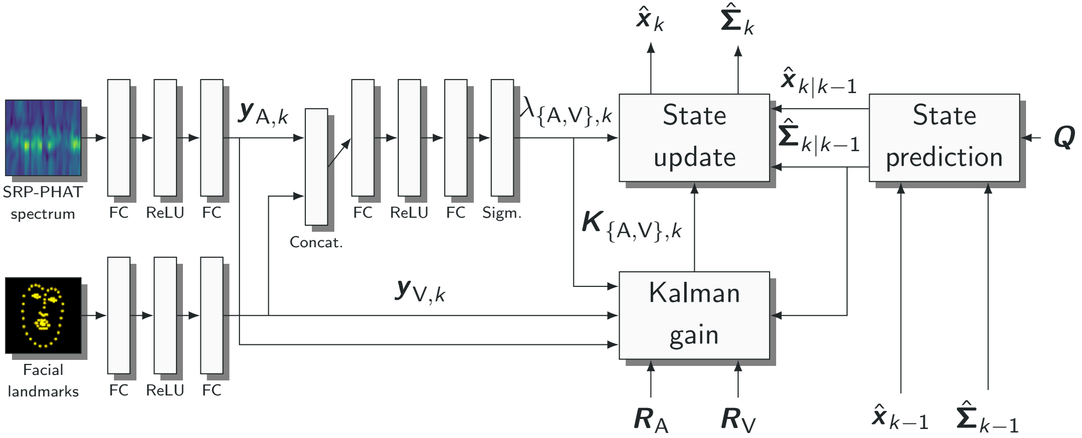
Dynamic stream weights must be predicted from sensor data ...

Dealing with high-dimensional observations is not straightforward[2] ...

[2] Tuomas Haarnoja, Anurag Ajay, Sergey Levine and Pieter Abbeel. "Backprop KF: Learning Discriminative Deterministic State Estimators". In: Advances in Neural Information Processing Systems, 2016

Proposed system

End-to-end optimization in a deep learning framework:



Proposed system

- Learning noise covariance matrices via Cholesky decomposition:

$$\boldsymbol{q} = \begin{bmatrix} q_1 \\ q_2 \\ \vdots \\ q_N \end{bmatrix} \quad \Rightarrow \quad \boldsymbol{L_Q} = \begin{bmatrix} q_1 & 0 & \cdots & 0 \\ q_2 & q_3 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ q_{N-3} & q_{N-2} & q_{N-1} & q_N \end{bmatrix} \quad \Rightarrow \quad \boldsymbol{Q} = \boldsymbol{L_Q} \boldsymbol{L_Q}^{\mathrm{T}}$$

with $\boldsymbol{Q} \in \mathbb{R}^{D \times D}$ and $N = \frac{D(D+1)}{2}$.

Proposed system

- Learning noise covariance matrices via Cholesky decomposition:

$$\boldsymbol{q} = \begin{bmatrix} q_1 \\ q_2 \\ \vdots \\ q_N \end{bmatrix} \quad \Rightarrow \quad \boldsymbol{L_Q} = \begin{bmatrix} q_1 & 0 & \cdots & 0 \\ q_2 & q_3 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ q_{N-3} & q_{N-2} & q_{N-1} & q_N \end{bmatrix} \quad \Rightarrow \quad \boldsymbol{Q} = \boldsymbol{L_Q}\boldsymbol{L_Q}^{\mathrm{T}}$$

  with $\boldsymbol{Q} \in \mathbb{R}^{D \times D}$ and $N = \frac{D(D+1)}{2}$.

- Projecting state space to direction-of-arrival in the loss function:

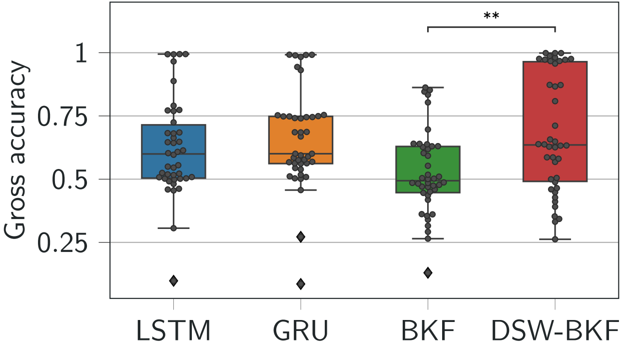
$$\mathcal{L} = \frac{1}{BK} \sum_{b=1}^{B} \sum_{k=1}^{K} \| \boldsymbol{C}_{\vartheta} \hat{\boldsymbol{x}}_k^{(b)} - \boldsymbol{\vartheta}_k^{(b)} \|_2^2$$

Evaluation

- Dataset: 70 audiovisual recordings of 7 speakers in an office environment, augmented with different acoustic noise conditions at 4 SNRs. 7-fold cross validation paradigm with 50/10/10 sequences train/val/test split.
- Training parameters:

Parameter	Description	Value
D_{z_A}	Audio feature dimension (SRP-PHAT spectrum)	481
D_{z_V}	Video feature dimension (facial landmarks)	136
D_{y_A}, D_{y_V}	Audio and video observation dimensions	4
D_x	State dimension	8
η	Learning rate	0.001
B	Batch size	128

Results

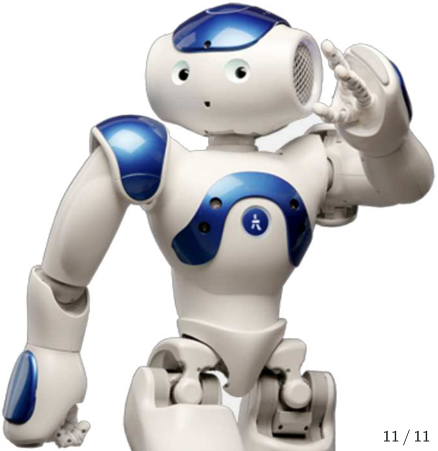


Model	Parameters
LSTM	382722
GRU	287106
BKF	21550
DSW-BKF	**42002**

Gross accuracy: Percentage of speakers detected correctly within a radius of 2° around the annotated ground-truth direction-of-arrival.

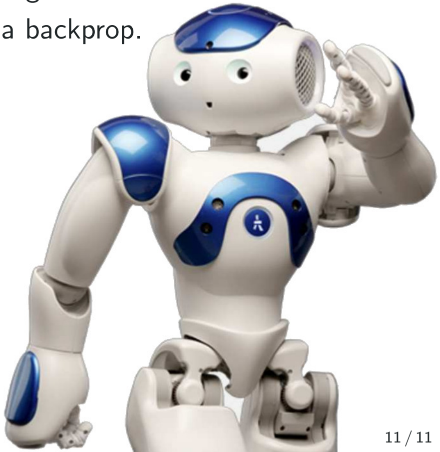
Conclusions

- Dynamic stream weights can benefit audiovisual speaker localization performance and provide an **additional level of explainability** regarding sensor reliability.



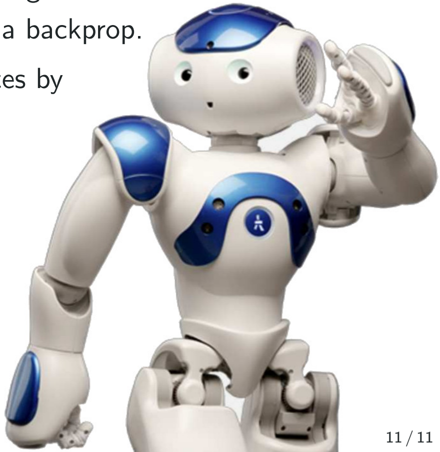
Conclusions

- Dynamic stream weights can benefit audiovisual speaker localization performance and provide an **additional level of explainability** regarding sensor reliability.
- The dynamic stream weight Kalman filter can be integrated into a **deep learning framework**, which can be trained via backprop.



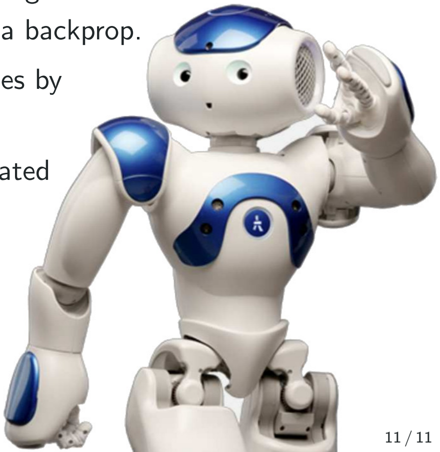
Conclusions

- Dynamic stream weights can benefit audiovisual speaker localization performance and provide an **additional level of explainability** regarding sensor reliability.
- The dynamic stream weight Kalman filter can be integrated into a **deep learning framework**, which can be trained via backprop.
- The proposed system still yields explainable estimates by providing **uncertainty information**.



Conclusions

- Dynamic stream weights can benefit audiovisual speaker localization performance and provide an **additional level of explainability** regarding sensor reliability.
- The dynamic stream weight Kalman filter can be integrated into a **deep learning framework**, which can be trained via backprop.
- The proposed system still yields explainable estimates by providing **uncertainty information**.
- **Dynamic stream weight prediction** is fully integrated into the system and can be trained jointly with the model parameters.



Conclusions

- Dynamic stream weights can benefit audiovisual speaker localization performance and provide an **additional level of explainability** regarding sensor reliability.
- The dynamic stream weight Kalman filter can be integrated into a **deep learning framework**, which can be trained via backprop.
- The proposed system still yields explainable estimates by providing **uncertainty information**.
- **Dynamic stream weight prediction** is fully integrated into the system and can be trained jointly with the model parameters.

Thank you for your attention!