Learning Dynamic Stream Weights for Multimodal Dynamical System Models

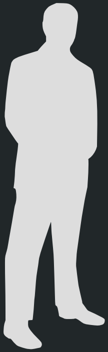
68. Sitzung der ITG-Fachgruppe "Signalverarbeitung und maschinelles Lernen"

Christopher Schymura and Dorothea Kolossa

September 27th, 2019

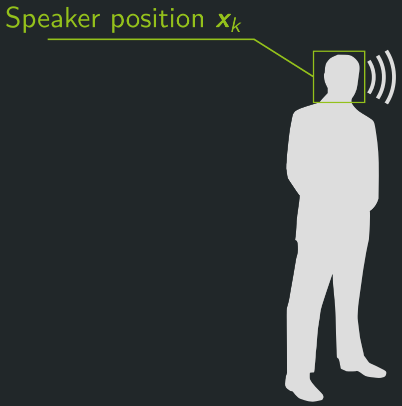
RUHR
UNIVERSITÄT
BOCHUM

Audiovisual speaker tracking

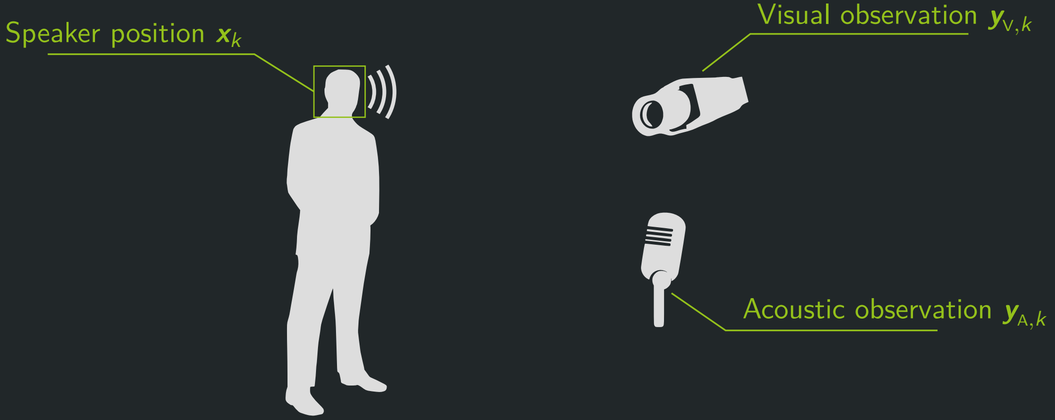


Audiovisual speaker tracking

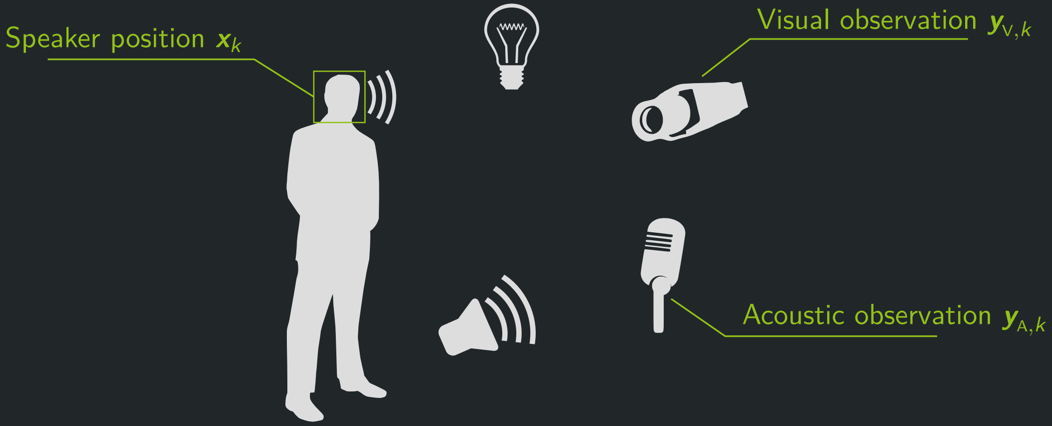
Audiovisual speaker tracking



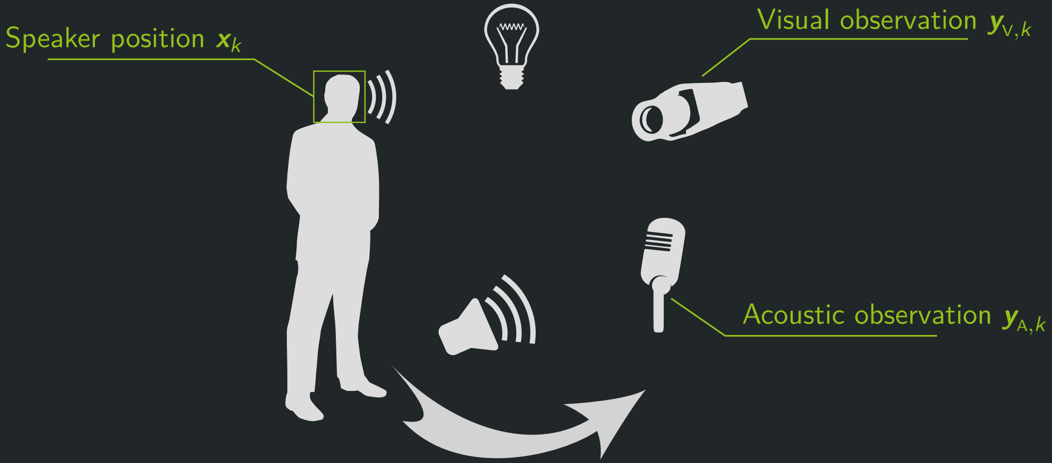
Audiovisual speaker tracking



Audiovisual speaker tracking



Audiovisual speaker tracking

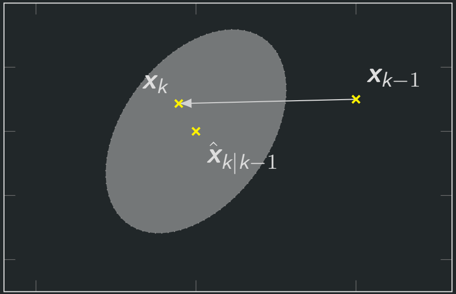


Audiovisual speaker tracking

Prediction step

System dynamics:

$$\boldsymbol{x}_k = f(\boldsymbol{x}_{k-1}) + \boldsymbol{v}_k, \quad \boldsymbol{v}_k = \mathcal{N}(0,\ \boldsymbol{Q})$$

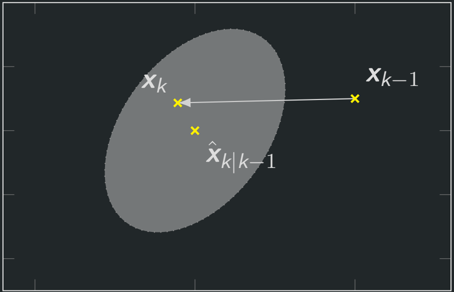


Audiovisual speaker tracking

Prediction step

System dynamics:

$$\mathbf{x}_k = f(\mathbf{x}_{k-1}) + \mathbf{v}_k, \quad \mathbf{v}_k = \mathcal{N}(\mathbf{0},\ \mathbf{Q})$$



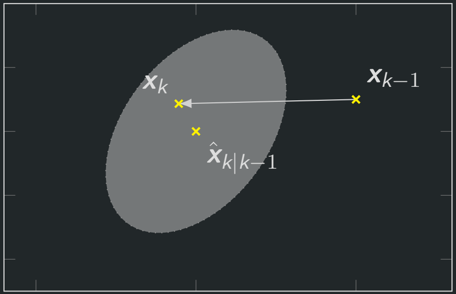
$$p(\mathbf{x}_k \,|\, \mathbf{Y}_{\mathrm{A},k-1},\ \mathbf{Y}_{\mathrm{V},k-1}) = \int p(\mathbf{x}_k \,|\, \mathbf{x}_{k-1})\ p(\mathbf{x}_{k-1} \,|\, \mathbf{Y}_{\mathrm{A},k-1},\ \mathbf{Y}_{\mathrm{V},k-1})\ d\mathbf{x}_{k-1}$$

Audiovisual speaker tracking

Prediction step

System dynamics:

$$\boldsymbol{x}_k = f(\boldsymbol{x}_{k-1}) + \boldsymbol{v}_k, \quad \boldsymbol{v}_k = \mathcal{N}(\mathbf{0},\ \boldsymbol{Q})$$



$$p(\boldsymbol{x}_k \mid \boldsymbol{Y}_{\mathrm{A},k-1},\ \boldsymbol{Y}_{\mathrm{V},k-1}) = \int \underbrace{p(\boldsymbol{x}_k \mid \boldsymbol{x}_{k-1})}_{\text{Dynamic model}} \underbrace{p(\boldsymbol{x}_{k-1} \mid \boldsymbol{Y}_{\mathrm{A},k-1},\ \boldsymbol{Y}_{\mathrm{V},k-1})}_{\text{Prior}} \, d\boldsymbol{x}_{k-1}$$

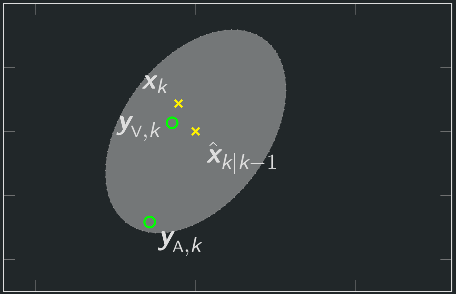
Audiovisual speaker tracking

Observation

Observation model:

$$\boldsymbol{y}_k = \begin{bmatrix} \boldsymbol{y}_{\mathrm{A},k} & \boldsymbol{y}_{\mathrm{V},k} \end{bmatrix}^{\mathsf{T}} = h\left(\boldsymbol{x}_k\right) + \boldsymbol{w}_k$$

$$\boldsymbol{w}_k = \mathcal{N}(0,\ \boldsymbol{R}), \quad \boldsymbol{R} = \begin{bmatrix} \boldsymbol{R}_{\mathrm{AA}} & \boldsymbol{R}_{\mathrm{AV}} \\ \boldsymbol{R}_{\mathrm{VA}} & \boldsymbol{R}_{\mathrm{VV}} \end{bmatrix}$$



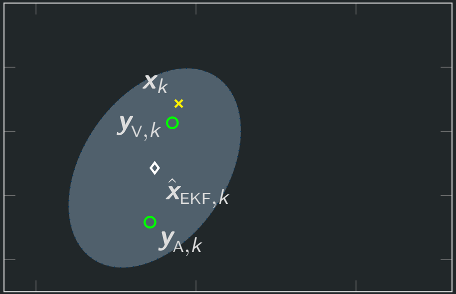
Audiovisual speaker tracking

Update step (standard Kalman filter)

Observation model:

$$\boldsymbol{y}_k = \begin{bmatrix} \boldsymbol{y}_{\mathrm{A},k} & \boldsymbol{y}_{\mathrm{V},k} \end{bmatrix}^{\mathrm{T}} = h\left(\boldsymbol{x}_k\right) + \boldsymbol{w}_k$$

$$\boldsymbol{w}_k = \mathcal{N}(0,\ \boldsymbol{R}), \quad \boldsymbol{R} = \begin{bmatrix} \boldsymbol{R}_{\mathrm{AA}} & \boldsymbol{R}_{\mathrm{AV}} \\ \boldsymbol{R}_{\mathrm{VA}} & \boldsymbol{R}_{\mathrm{VV}} \end{bmatrix}$$



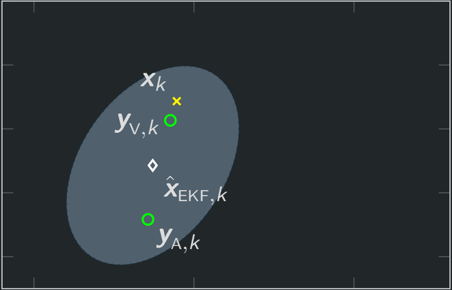
Audiovisual speaker tracking

Update step (standard Kalman filter)

Observation model:

$$\boldsymbol{y}_k = \begin{bmatrix} \boldsymbol{y}_{\mathrm{A},k} & \boldsymbol{y}_{\mathrm{V},k} \end{bmatrix}^{\mathrm{T}} = h\left(\boldsymbol{x}_k\right) + \boldsymbol{w}_k$$

$$\boldsymbol{w}_k = \mathcal{N}(0,\ \boldsymbol{R}), \qquad \boldsymbol{R} = \begin{bmatrix} \boldsymbol{R}_{\mathrm{AA}} & \boldsymbol{R}_{\mathrm{AV}} \\ \boldsymbol{R}_{\mathrm{VA}} & \boldsymbol{R}_{\mathrm{VV}} \end{bmatrix}$$



$$p(\boldsymbol{x}_k \,|\, \boldsymbol{Y}_{\mathrm{A},k},\ \boldsymbol{Y}_{\mathrm{V},k}) \propto p(\boldsymbol{x}_k \,|\, \boldsymbol{Y}_{\mathrm{A},k-1},\ \boldsymbol{Y}_{\mathrm{V},k-1})\, p(\boldsymbol{y}_{\mathrm{A},k},\ \boldsymbol{y}_{\mathrm{V},k} \,|\, \boldsymbol{x}_k)$$

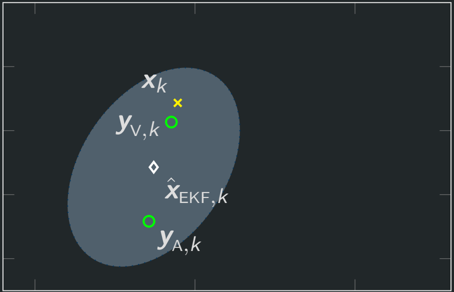
Audiovisual speaker tracking

Update step (standard Kalman filter)

Observation model:

$$\boldsymbol{y}_k = \begin{bmatrix} \boldsymbol{y}_{\mathrm{A},k} & \boldsymbol{y}_{\mathrm{V},k} \end{bmatrix}^{\mathsf{T}} = h\left(\boldsymbol{x}_k\right) + \boldsymbol{w}_k$$

$$\boldsymbol{w}_k = \mathcal{N}(0,\ \boldsymbol{R}), \quad \boldsymbol{R} = \begin{bmatrix} \boldsymbol{R}_{\mathrm{AA}} & \boldsymbol{R}_{\mathrm{AV}} \\ \boldsymbol{R}_{\mathrm{VA}} & \boldsymbol{R}_{\mathrm{VV}} \end{bmatrix}$$



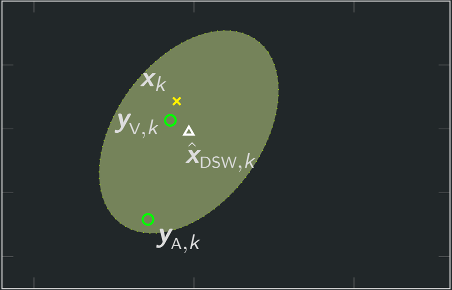
$$p(\boldsymbol{x}_k \mid \boldsymbol{Y}_{\mathrm{A},k},\ \boldsymbol{Y}_{\mathrm{V},k}) \propto p(\boldsymbol{x}_k \mid \boldsymbol{Y}_{\mathrm{A},k-1},\ \boldsymbol{Y}_{\mathrm{V},k-1})\ \underbrace{p(\boldsymbol{y}_{\mathrm{A},k},\ \boldsymbol{y}_{\mathrm{V},k} \mid \boldsymbol{x}_k)}_{\text{Sensor model}}$$

Audiovisual speaker tracking

Update step (Kalman filter with dynamic stream weights[1])

Observation model:

$$\boldsymbol{y}_{\mathrm{A},k} = h_{\mathrm{A}}(\boldsymbol{x}_k) + \boldsymbol{w}_{\mathrm{A},k}, \quad \boldsymbol{w}_{\mathrm{A},k} = \mathcal{N}(0,\ \boldsymbol{R}_{\mathrm{AA}})$$
$$\boldsymbol{y}_{\mathrm{V},k} = h_{\mathrm{V}}(\boldsymbol{x}_k) + \boldsymbol{w}_{\mathrm{V},k}, \quad \boldsymbol{w}_{\mathrm{V},k} = \mathcal{N}(0,\ \boldsymbol{R}_{\mathrm{VV}})$$



[1] C. Schymura et al.: *Extending linear dynamical systems with dynamic stream weights for audiovisual speaker localization*, IWAENC, 2018

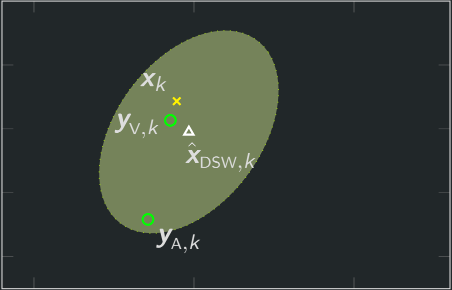
Audiovisual speaker tracking

Update step (Kalman filter with dynamic stream weights[1])

Observation model:

$$\boldsymbol{y}_{\mathrm{A},k} = h_{\mathrm{A}}(\boldsymbol{x}_k) + \boldsymbol{w}_{\mathrm{A},k}, \quad \boldsymbol{w}_{\mathrm{A},k} = \mathcal{N}(0,\ \boldsymbol{R}_{\mathrm{AA}})$$

$$\boldsymbol{y}_{\mathrm{V},k} = h_{\mathrm{V}}(\boldsymbol{x}_k) + \boldsymbol{w}_{\mathrm{V},k}, \quad \boldsymbol{w}_{\mathrm{V},k} = \mathcal{N}(0,\ \boldsymbol{R}_{\mathrm{VV}})$$



$$p(\boldsymbol{x}_k \mid \boldsymbol{Y}_{\mathrm{A},k},\ \boldsymbol{Y}_{\mathrm{V},k}) \propto p(\boldsymbol{x}_k \mid \boldsymbol{Y}_{\mathrm{A},k-1},\ \boldsymbol{Y}_{\mathrm{V},k-1})\ \underbrace{p(\boldsymbol{y}_{\mathrm{A},k} \mid \boldsymbol{x}_k)^{\lambda_k}}_{\text{Acoustic model}}\ \underbrace{p(\boldsymbol{y}_{\mathrm{V},k} \mid \boldsymbol{x}_k)^{1-\lambda_k}}_{\text{Visual model}}$$

[1] C. Schymura et al.: *Extending linear dynamical systems with dynamic stream weights for audiovisual speaker localization*, IWAENC, 2018

Inference

Extended Kalman filter approach: first-order Taylor series expansion

$$f(\boldsymbol{x}_{k-1}) \approx f(\hat{\boldsymbol{x}}_{k-1}) + \boldsymbol{F}(\hat{\boldsymbol{x}}_{k-1})(\boldsymbol{x}_{k-1} - \hat{\boldsymbol{x}}_{k-1})$$

Inference

Extended Kalman filter approach: first-order Taylor series expansion

$$f(\boldsymbol{x}_{k-1}) \approx f(\hat{\boldsymbol{x}}_{k-1}) + \boldsymbol{F}(\hat{\boldsymbol{x}}_{k-1})(\boldsymbol{x}_{k-1} - \hat{\boldsymbol{x}}_{k-1})$$

$$\Rightarrow \qquad p(\boldsymbol{x}_k \mid \boldsymbol{x}_{k-1}) = \mathcal{N}\Big(\boldsymbol{x}_k \mid f(\hat{\boldsymbol{x}}_{k-1}) + \boldsymbol{F}(\hat{\boldsymbol{x}}_{k-1})(\boldsymbol{x}_{k-1} - \hat{\boldsymbol{x}}_{k-1}),\ \boldsymbol{Q}\Big)$$

$$\Rightarrow \qquad p(\boldsymbol{x}_k \mid \boldsymbol{Y}_{\mathrm{A},k-1},\ \boldsymbol{Y}_{\mathrm{V},k-1}) = \mathcal{N}\Big(\boldsymbol{x}_k \mid \hat{\boldsymbol{x}}_{k-1},\ \hat{\Sigma}_{k-1}\Big)$$

Inference

Extended Kalman filter approach: first-order Taylor series expansion

$$f(\boldsymbol{x}_{k-1}) \approx f(\hat{\boldsymbol{x}}_{k-1}) + \boldsymbol{F}(\hat{\boldsymbol{x}}_{k-1})(\boldsymbol{x}_{k-1} - \hat{\boldsymbol{x}}_{k-1})$$

$$\Rightarrow \qquad p(\boldsymbol{x}_k \,|\, \boldsymbol{x}_{k-1}) = \mathcal{N}\Big(\boldsymbol{x}_k \,|\, f(\hat{\boldsymbol{x}}_{k-1}) + \boldsymbol{F}(\hat{\boldsymbol{x}}_{k-1})(\boldsymbol{x}_{k-1} - \hat{\boldsymbol{x}}_{k-1}),\, \boldsymbol{Q}\Big)$$

$$\Rightarrow \qquad p(\boldsymbol{x}_k \,|\, \boldsymbol{Y}_{\mathrm{A},k-1},\, \boldsymbol{Y}_{\mathrm{V},k-1}) = \mathcal{N}\Big(\boldsymbol{x}_k \,|\, \hat{\boldsymbol{x}}_{k-1},\, \hat{\boldsymbol{\Sigma}}_{k-1}\Big)$$

Prediction step (identical to standard EKF)

$$\hat{\boldsymbol{x}}_{k|k-1} = f(\hat{\boldsymbol{x}}_{k-1})$$

$$\hat{\boldsymbol{\Sigma}}_{k|k-1} = \boldsymbol{F}_{k-1}\hat{\boldsymbol{\Sigma}}_{k-1}\boldsymbol{F}_{k-1}^{\mathsf{T}} + \boldsymbol{Q}, \qquad \boldsymbol{F}_{k-1} \equiv \boldsymbol{F}(\hat{\boldsymbol{x}}_{k-1}) = \frac{\partial f(\boldsymbol{x}_{k-1})}{\partial \boldsymbol{x}_{k-1}}\Big|_{\boldsymbol{x}_{k-1}=\hat{\boldsymbol{x}}_{k-1}}$$

Inference

Extended Kalman filter approach: first-order Taylor series expansion

$$h_{\{\mathrm{A,V}\}}(\boldsymbol{x}_k) \approx h_{\{\mathrm{A,V}\}}(\hat{\boldsymbol{x}}_k) + \boldsymbol{H}_{\{\mathrm{A,V}\},k}(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k), \quad \boldsymbol{H}_{\{\mathrm{A,V}\},k} \equiv \left.\frac{\partial h_{\{\mathrm{A,V}\}}(\boldsymbol{x}_k)}{\partial \boldsymbol{x}_k}\right|_{\boldsymbol{x}_k=\hat{\boldsymbol{x}}_k}$$

Inference

Extended Kalman filter approach: first-order Taylor series expansion

$$h_{\{\mathrm{A,V}\}}(\boldsymbol{x}_k) \approx h_{\{\mathrm{A,V}\}}(\hat{\boldsymbol{x}}_k) + \boldsymbol{H}_{\{\mathrm{A,V}\},k}(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k), \quad \boldsymbol{H}_{\{\mathrm{A,V}\},k} \equiv \frac{\partial h_{\{\mathrm{A,V}\}}(\boldsymbol{x}_k)}{\partial \boldsymbol{x}_k}\bigg|_{\boldsymbol{x}_k=\hat{\boldsymbol{x}}_k}$$

$$\Rightarrow \qquad p(\boldsymbol{y}_{\{\mathrm{A,V}\},k} \,|\, \boldsymbol{x}_k) = \mathcal{N}\Big(\boldsymbol{y}_{\{\mathrm{A,V}\},k}, \,|\, h_{\{\mathrm{A,V}\}}(\hat{\boldsymbol{x}}_k) + \boldsymbol{H}_{\{\mathrm{A,V}\},k})(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k), \, \boldsymbol{R}_{\{\mathrm{A,V}\}}\Big)$$

Inference

Extended Kalman filter approach: first-order Taylor series expansion

$$h_{\{A,V\}}(\boldsymbol{x}_k) \approx h_{\{A,V\}}(\hat{\boldsymbol{x}}_k) + \boldsymbol{H}_{\{A,V\},k}(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k), \quad \boldsymbol{H}_{\{A,V\},k} \equiv \left.\frac{\partial h_{\{A,V\}}(\boldsymbol{x}_k)}{\partial \boldsymbol{x}_k}\right|_{\boldsymbol{x}_k = \hat{\boldsymbol{x}}_k}$$

$$\Rightarrow \qquad p(\boldsymbol{y}_{\{A,V\},k} \,|\, \boldsymbol{x}_k) = \mathcal{N}\left(\boldsymbol{y}_{\{A,V\},k}; \,|\, h_{\{A,V\}}(\hat{\boldsymbol{x}}_k) + \boldsymbol{H}_{\{A,V\},k})(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k), \, \boldsymbol{R}_{\{A,V\}}\right)$$

Update step

$$\begin{bmatrix} \boldsymbol{K}_{A,k}^{T} \\ \boldsymbol{K}_{V,k}^{T} \end{bmatrix} = \begin{bmatrix} \boldsymbol{R}_A + \lambda_k \boldsymbol{H}_{A,k} \hat{\boldsymbol{\Sigma}}_{k|k-1} \boldsymbol{H}_{A,k}^{T} & (1-\lambda_k) \boldsymbol{H}_{A,k} \hat{\boldsymbol{\Sigma}}_{k|k-1} \boldsymbol{H}_{V,k}^{T} \\ \lambda_k \boldsymbol{H}_{V,k} \hat{\boldsymbol{\Sigma}}_{k|k-1} \boldsymbol{H}_{A,k}^{T} & \boldsymbol{R}_V + (1-\lambda_k) \boldsymbol{H}_{V,k} \hat{\boldsymbol{\Sigma}}_{k|k-1} \boldsymbol{H}_{V,k}^{T} \end{bmatrix}^{-1} \begin{bmatrix} \boldsymbol{H}_{A,k} \\ \boldsymbol{H}_{V,k} \end{bmatrix} \hat{\boldsymbol{\Sigma}}_{k|k-1}$$

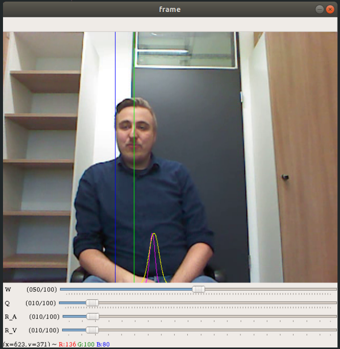
$$\hat{\boldsymbol{x}}_k = \hat{\boldsymbol{x}}_{k|k-1} + \lambda_k \boldsymbol{K}_{A,k}\left(\boldsymbol{y}_{A,k} - h_A(\hat{\boldsymbol{x}}_k)\right) + (1-\lambda_k)\boldsymbol{K}_{V,k}\left(\boldsymbol{y}_{V,k} - h_V(\hat{\boldsymbol{x}}_k)\right)$$

$$\hat{\boldsymbol{\Sigma}}_{k|k-1} = \left(\boldsymbol{I} - \lambda_k \boldsymbol{K}_{A,k}\boldsymbol{H}_{A,k} - (1-\lambda_k)\boldsymbol{K}_{V,k}\boldsymbol{H}_{V,k}\right)\hat{\boldsymbol{\Sigma}}_{k|k-1}$$

Evaluation I

Experimental setup

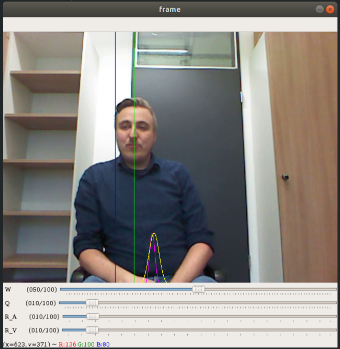
- KAVTraC audiovisual dataset, recorded in an office room at RUB using the Kinect sensor (7 speakers, $T_{60} \approx 350\,\mathrm{ms}$, 35 min. duration).



Evaluation I

Experimental setup

- KAVTraC audiovisual dataset, recorded in an office room at RUB using the Kinect sensor (7 speakers, $T_{60} \approx 350\,\mathrm{ms}$, 35 min. duration).
- Constant velocity linear dynamics model and nonlinear rotating vector observation models.
- DSW-EKF uses Dirichlet-prior oracle DSWs[2].

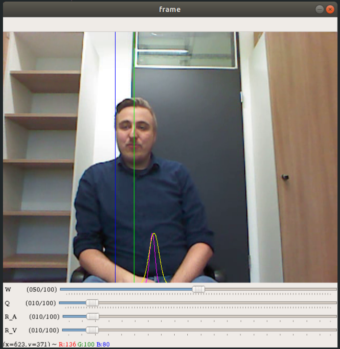


[2] C. Schymura et al.: *Audiovisual speaker tracking using nonlinear dynamical systems with dynamic stream weights*, arXiv, 2019

Evaluation I

Experimental setup

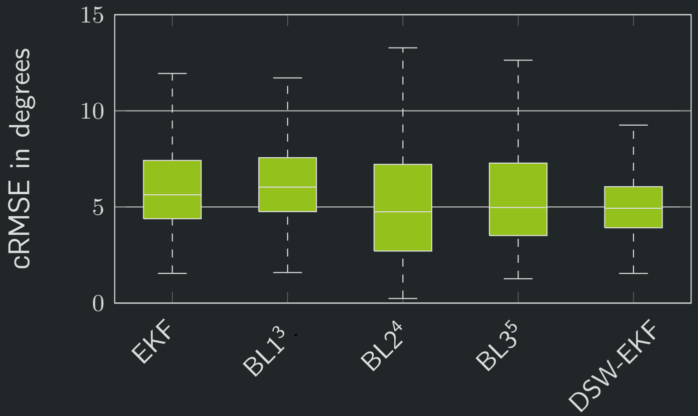
- KAVTraC audiovisual dataset, recorded in an office room at RUB using the Kinect sensor (7 speakers, $T_{60} \approx 350\,\mathrm{ms}$, 35 min. duration).
- Constant velocity linear dynamics model and nonlinear rotating vector observation models.
- DSW-EKF uses Dirichlet-prior oracle DSWs[2].
- Four baseline systems: standard EKF, one KF-based and two particle filter-based systems.
- Leave-one-out cross-validation paradigm.



[2] C. Schymura et al.: *Audiovisual speaker tracking using nonlinear dynamical systems with dynamic stream weights*, arXiv, 2019

Evaluation I

Results



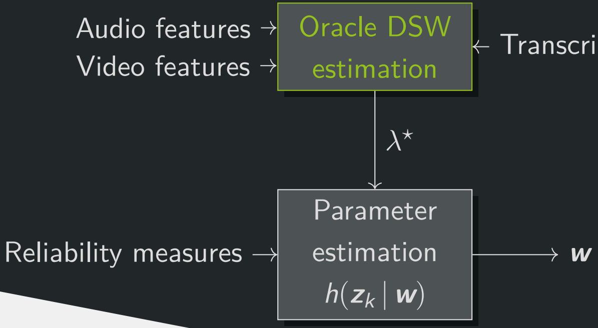
[3] T. Gehrig et al.: *Kalman filters for audio-video source localization*, WASPAA, 2005

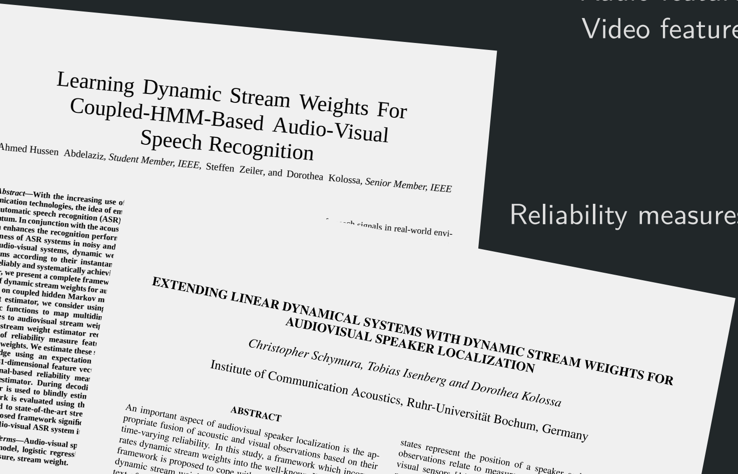
[4] S. Gerlach et al.: *2D audio-visual localization in home environments using a particle filter*, ITG Symp., 2012

[5] X. Qian et al.: *3D audio-visual speaker tracking with an adaptive particle filter*, ICASSP, 2017

Learning dynamic stream weights

Standard approach: Supervised training with oracle dynamic stream weights





Learning Dynamic Stream Weights For Coupled-HMM-Based Audio-Visual Speech Recognition

Ahmed Hussen Abdelaziz, Student Member, IEEE, Steffen Zeiler, and Dorothea Kolossa, Senior Member, IEEE

EXTENDING LINEAR DYNAMICAL SYSTEMS WITH DYNAMIC STREAM WEIGHTS FOR AUDIOVISUAL SPEAKER LOCALIZATION

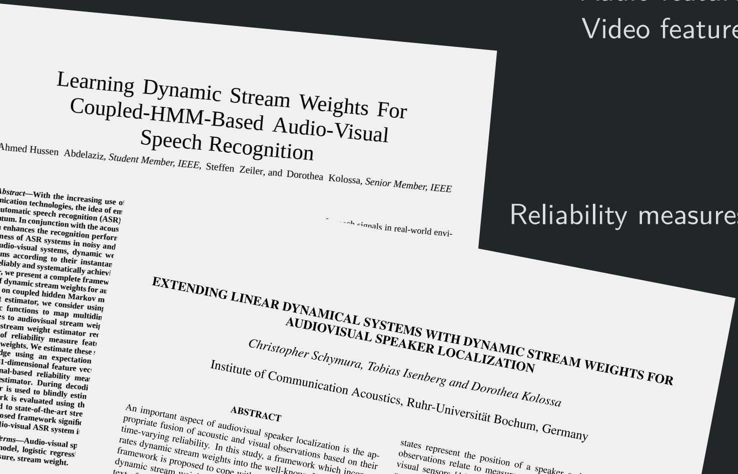
Christopher Schymura, Tobias Isenberg and Dorothea Kolossa

Institute of Communication Acoustics, Ruhr-Universität Bochum, Germany

ABSTRACT

Learning dynamic stream weights

Standard approach: Supervised training with oracle dynamic stream weights


Learning Dynamic Stream Weights For Coupled-HMM-Based Audio-Visual Speech Recognition

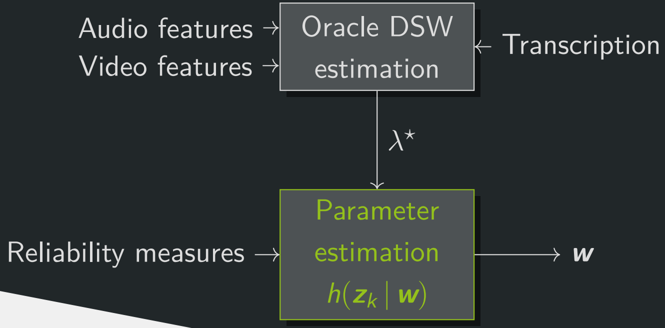
Ahmed Hussen Abdelaziz, *Student Member, IEEE*, Steffen Zeiler, and Dorothea Kolossa, *Senior Member, IEEE*

EXTENDING LINEAR DYNAMICAL SYSTEMS WITH DYNAMIC STREAM WEIGHTS FOR AUDIOVISUAL SPEAKER LOCALIZATION

Christopher Schymura, Tobias Isenberg and Dorothea Kolossa

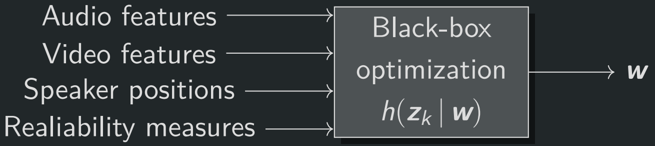
Institute of Communication Acoustics, Ruhr-Universität Bochum, Germany

ABSTRACT

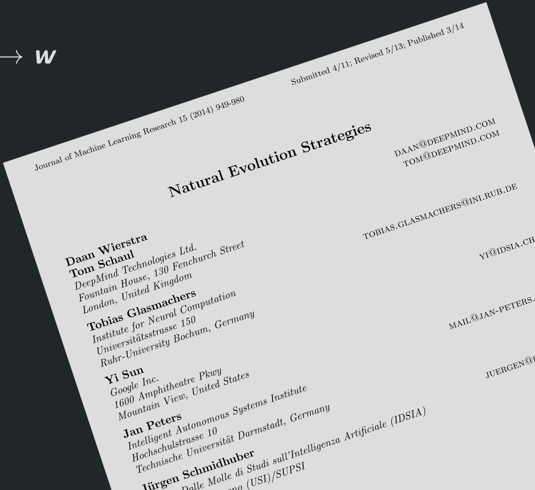


Learning dynamic stream weights

Proposed approach: Training with natural evolution strategies



- No oracle information required.
- Flexible choice of loss/fitness function.


Journal of Machine Learning Research 15 (2014) 949-980 Submitted 4/11; Revised 5/13; Published 3/14

Natural Evolution Strategies

Daan Wierstra DAAN@DEEPMIND.COM
Tom Schaul TOM@DEEPMIND.COM
DeepMind Technologies Ltd.
Fountain House, 130 Fenchurch Street
London, United Kingdom

Tobias Glasmachers TOBIAS.GLASMACHERS@INI.RUB.DE
Institute for Neural Computation
Universitätsstrasse 150
Ruhr-University Bochum, Germany

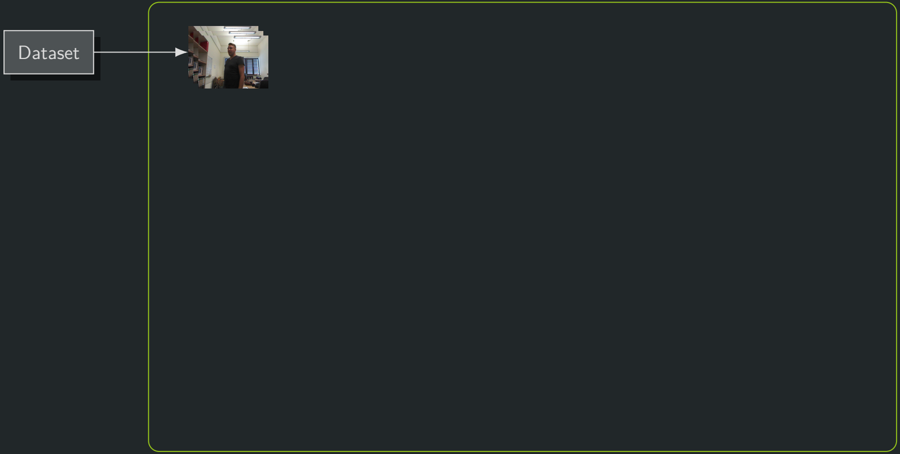
Yi Sun YI@IDSIA.CH
Google Inc.
1600 Amphitheatre Pkwy
Mountain View, United States

Jan Peters MAIL@JAN-PETERS.
Intelligent Autonomous Systems Institute
Hochschulstrasse 10
Technische Universität Darmstadt, Germany

Jürgen Schmidhuber JUERGEN@
Dalle Molle di Studi sull'Intelligenza Artificiale (IDSIA)
(USI)/SUPSI

Learning dynamic stream weights

Training procedure[6]



[6] C. Schymura et al.: *Learning dynamic stream weights for linear dynamical systems using natural evolution strategies*, ICASSP, 2019

Learning dynamic stream weights

Training procedure[6]

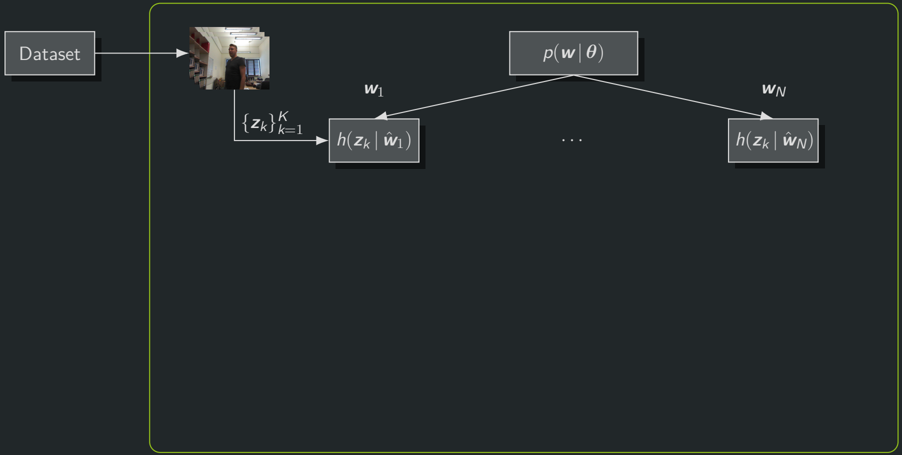
Dataset

$p(\boldsymbol{w} \mid \boldsymbol{\theta})$

[6] C. Schymura et al.: *Learning dynamic stream weights for linear dynamical systems using natural evolution strategies*, ICASSP, 2019

Learning dynamic stream weights

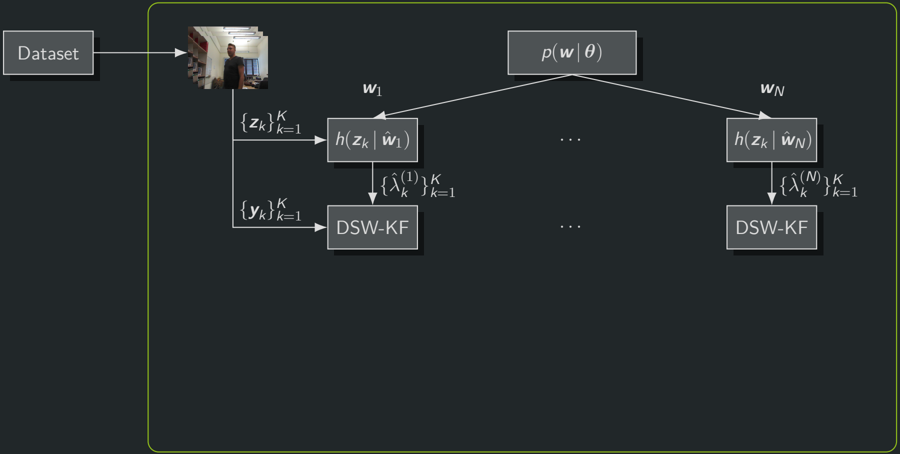
Training procedure[6]



[6] C. Schymura et al.: *Learning dynamic stream weights for linear dynamical systems using natural evolution strategies*, ICASSP, 2019

Learning dynamic stream weights

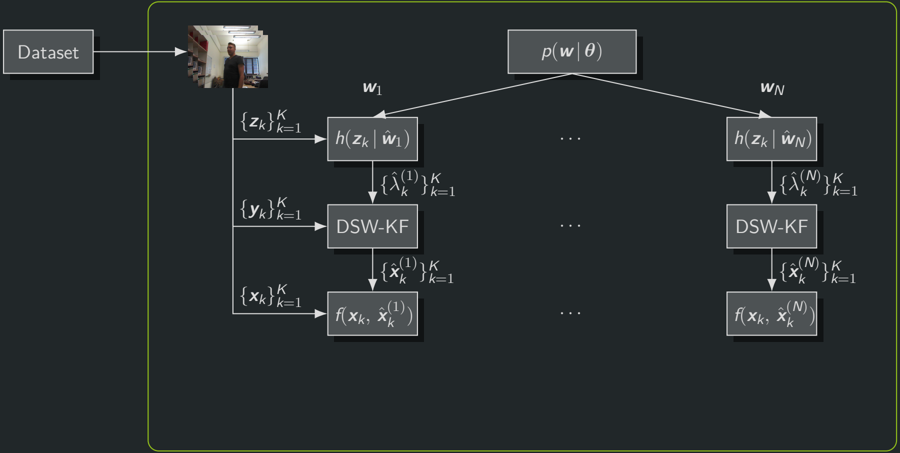
Training procedure[6]



[6] C. Schymura et al.: *Learning dynamic stream weights for linear dynamical systems using natural evolution strategies*, ICASSP, 2019

Learning dynamic stream weights

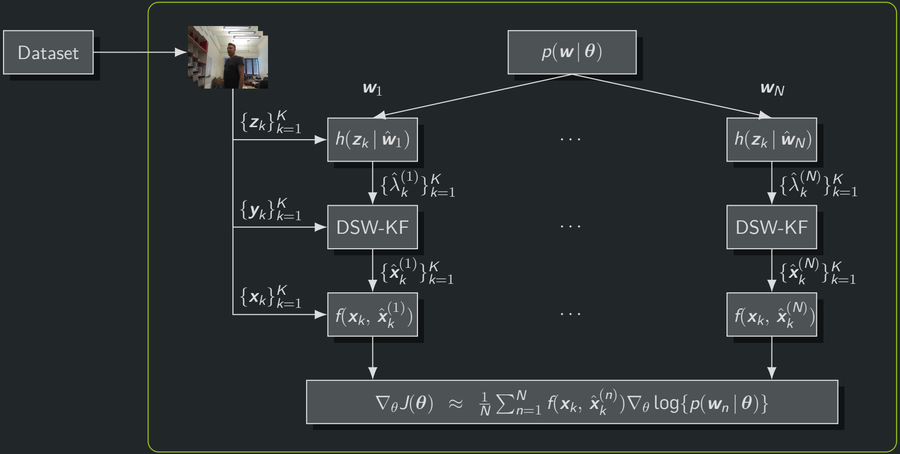
Training procedure[6]



[6] C. Schymura et al.: *Learning dynamic stream weights for linear dynamical systems using natural evolution strategies*, ICASSP, 2019

Learning dynamic stream weights

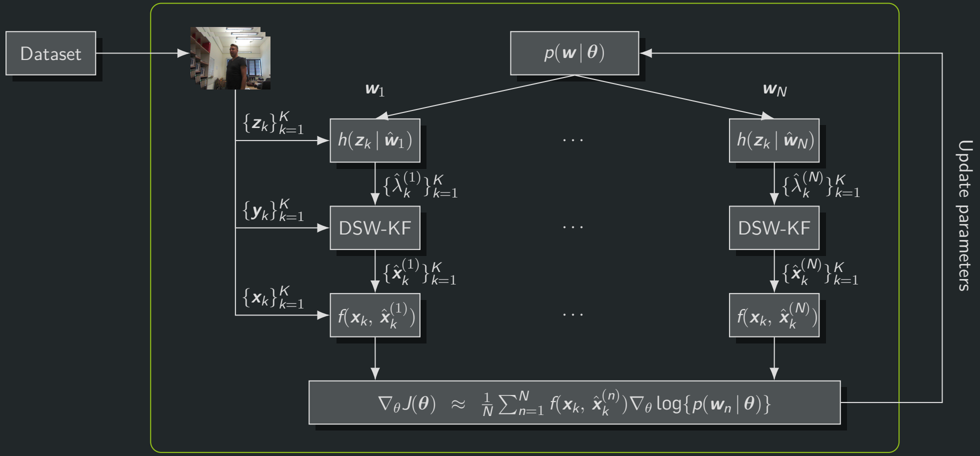
Training procedure[6]



[6] C. Schymura et al.: *Learning dynamic stream weights for linear dynamical systems using natural evolution strategies*, ICASSP, 2019

Learning dynamic stream weights

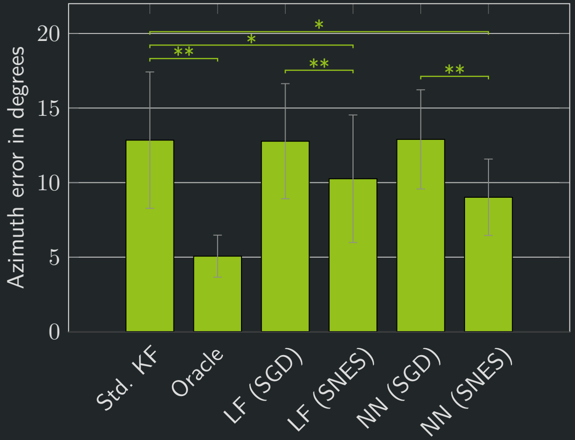
Training procedure[6]



[6] C. Schymura et al.: *Learning dynamic stream weights for linear dynamical systems using natural evolution strategies*, ICASSP, 2019

Evaluation II

Results



Conclusions and outlook

- DSW-based audiovisual speaker tracking frameworks can be extended to cope with nonlinear systems.

Conclusions and outlook

- DSW-based audiovisual speaker tracking frameworks can be extended to cope with nonlinear systems.
- A DSW-based audiovisual speaker tracking system can benefit from black-box optimization approaches.

Conclusions and outlook

- DSW-based audiovisual speaker tracking frameworks can be extended to cope with nonlinear systems.
- A DSW-based audiovisual speaker tracking system can benefit from black-box optimization approaches.
- Ideas for future work:

Conclusions and outlook

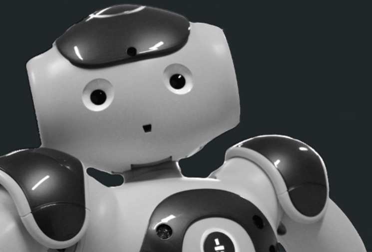
- DSW-based audiovisual speaker tracking frameworks can be extended to cope with nonlinear systems.
- A DSW-based audiovisual speaker tracking system can benefit from black-box optimization approaches.
- Ideas for future work:
  - Joint optimization of model and DSW estimation in a deep learning framework.

Conclusions and outlook

- DSW-based audiovisual speaker tracking frameworks can be extended to cope with nonlinear systems.
- A DSW-based audiovisual speaker tracking system can benefit from black-box optimization approaches.
- Ideas for future work:
  - Joint optimization of model and DSW estimation in a deep learning framework.
  - Extension to multi-speaker scenarios.

Conclusions and outlook

- DSW-based audiovisual speaker tracking frameworks can be extended to cope with nonlinear systems.
- A DSW-based audiovisual speaker tracking system can benefit from black-box optimization approaches.
- Ideas for future work:
  - Joint optimization of model and DSW estimation in a deep learning framework.
  - Extension to multi-speaker scenarios.



Thank you for your attention!