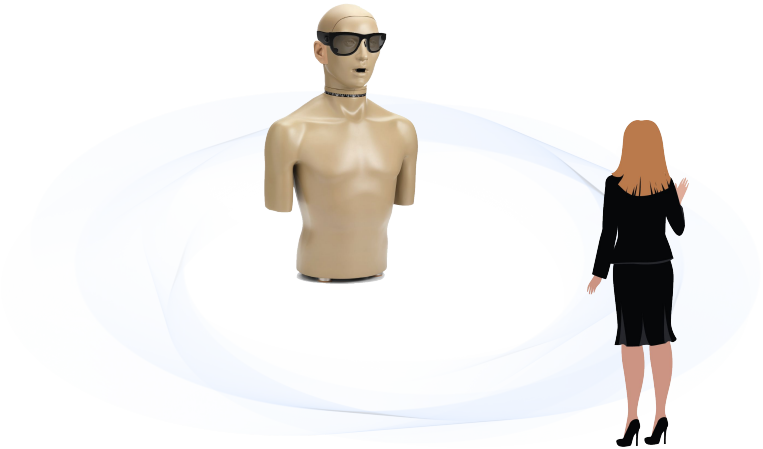
AGADIR: Towards Array-Geometry Agnostic Directional Speech Recognition

Ju Lin, Niko Moritz, Yiteng Huang,

Ruiming Xie, Ming Sun, Christian Fuegen, Frank Seide

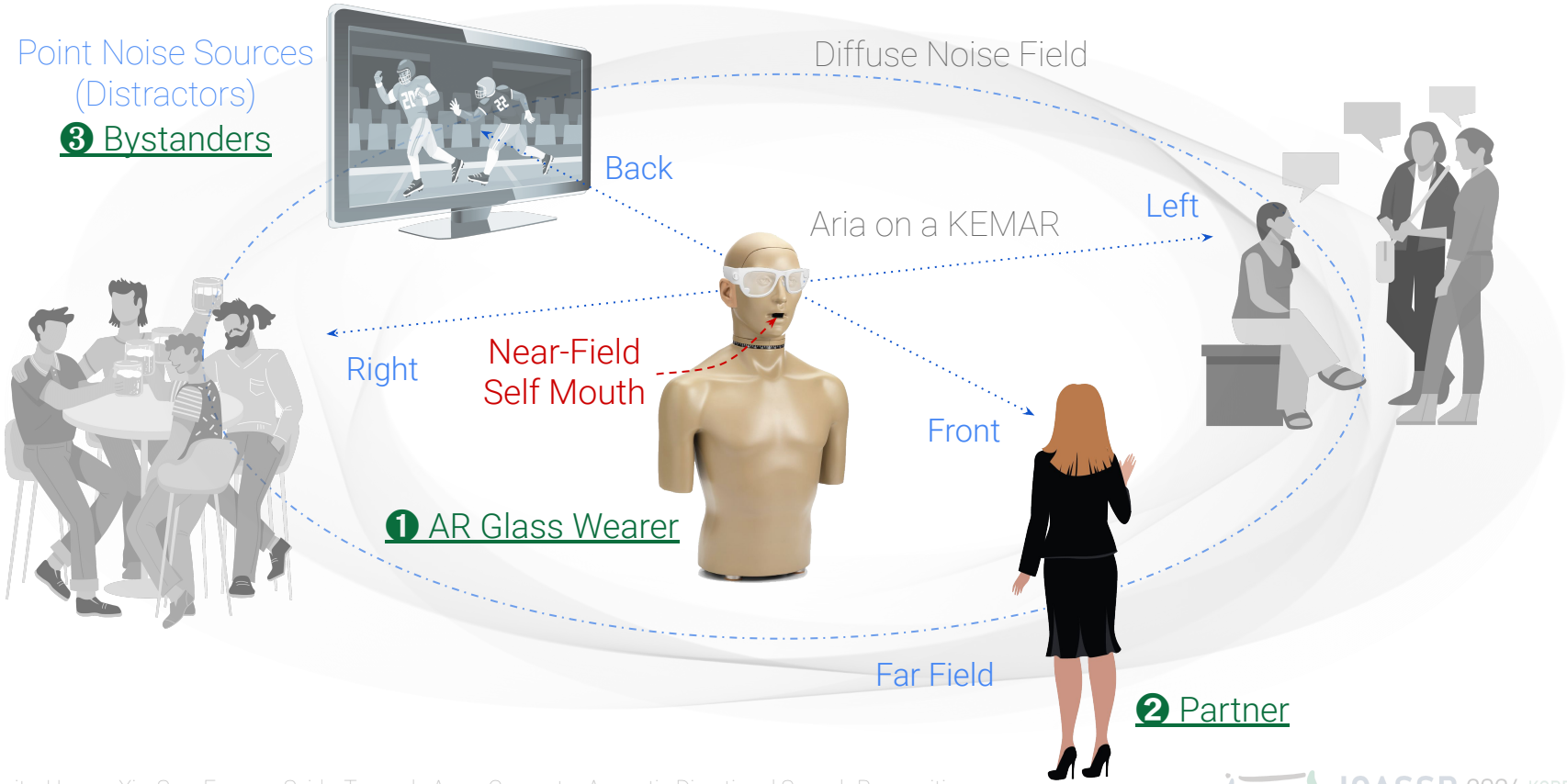
Meta



ICASSP 2024 KOREA

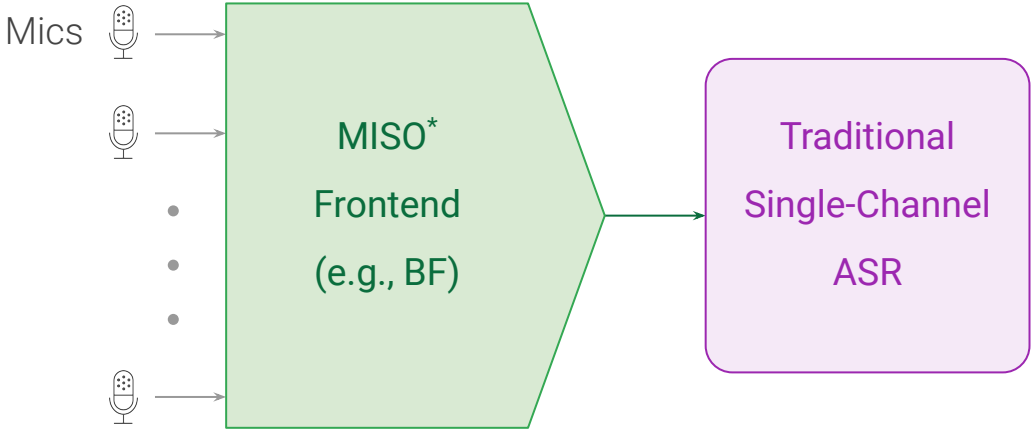
IEEE Signal Processing Society

Acoustic Scenario of Speech Recognition on AR Glasses



Three Methods for Multichannel ASR

① Traditional MISO Frontend | ② End-to-End | ③ Directional ASR

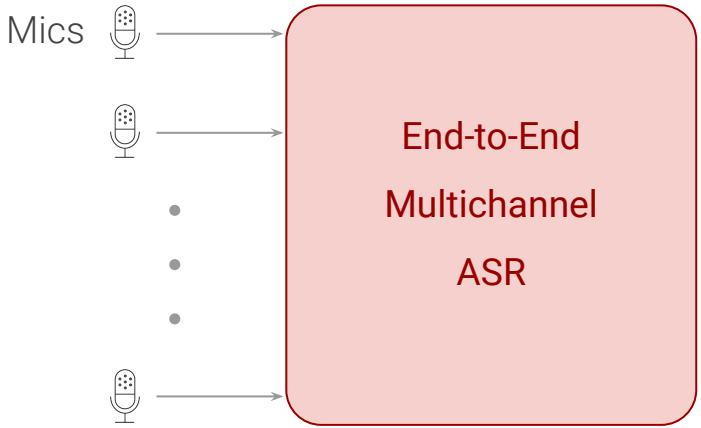


*MISO: Multiple Input Single Output

- ASR model: Array/Device agnostic.
- MISO frontend: array specific.
- Challenge: multi-talker scenario.

Three Methods for Multichannel ASR

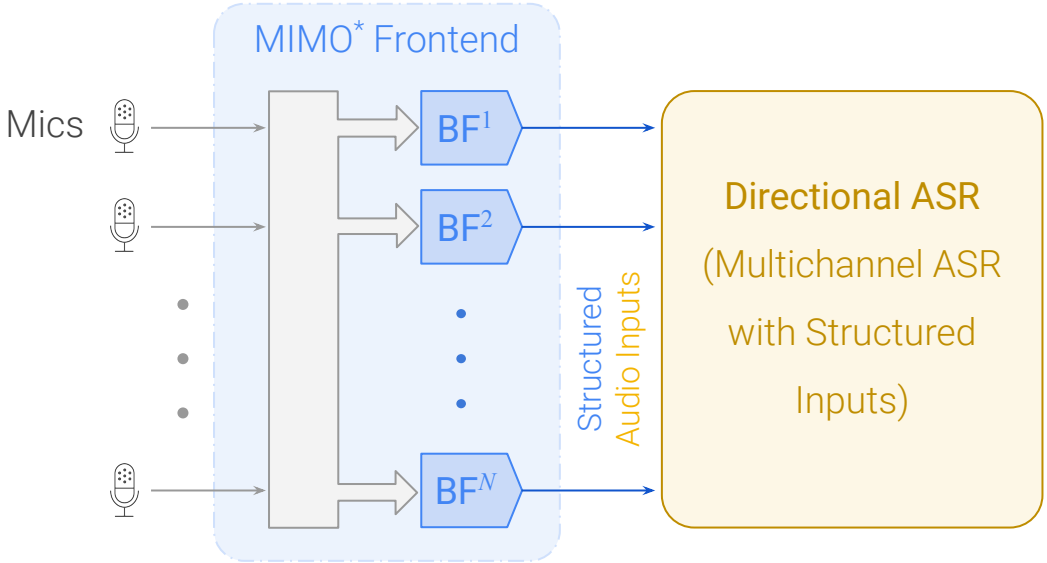
① Traditional MISO Frontend | ② End-to-End | ③ Directional ASR



- More aligned with ML: powerful.
- Lack of transparency, interpretability, and modularity.
- High cost of development and maintenance: device dependent or array specific.
- If preprocessing is conducted prior to input, it is crucial to retain cross-channel phase differences to enable end-to-end learning of the spatial sound field.

Three Methods for Multichannel ASR

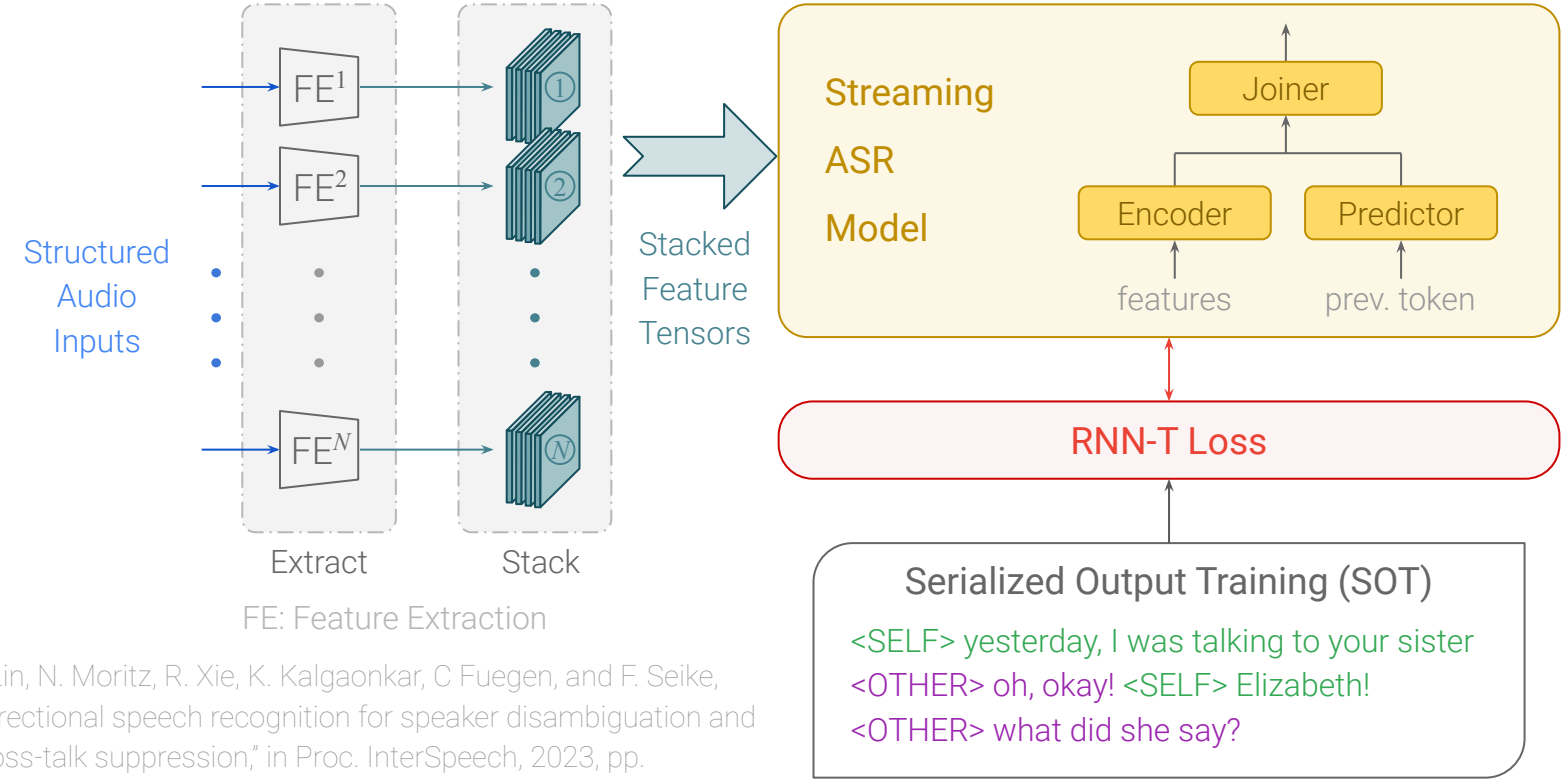
① Traditional MISO Frontend | ② End-to-End | ③ Directional ASR



*MIMO: Multiple Input Multiple Output

- The MIMO frontend is apparently array specific.
- But the structured audio inputs for Directional ASR can be independent of array geometry and microphone locations.
- After the MIMO frontend, cross-channel phase diffs can be discarded.

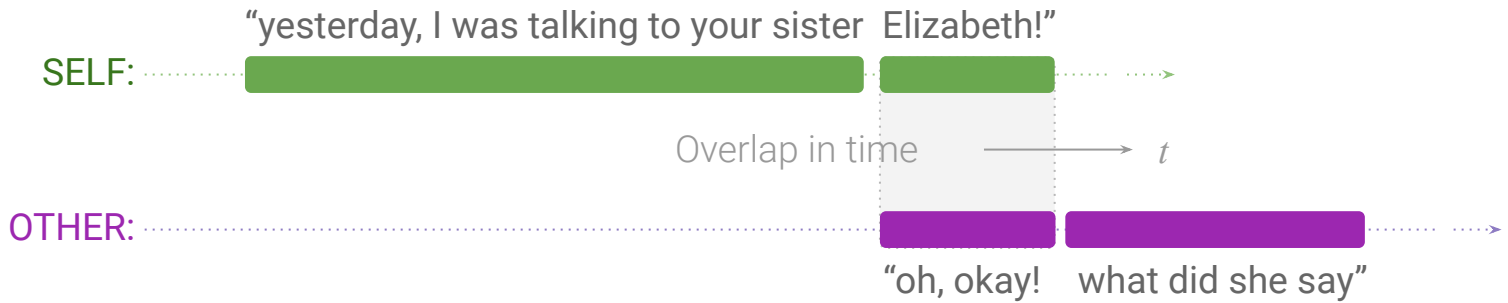
Directional ASR (D-ASR)



FE: Feature Extraction

[1] J Lin, N. Moritz, R. Xie, K. Kalgaonkar, C Fuegen, and F. Seike, "Directional speech recognition for speaker disambiguation and cross-talk suppression," in Proc. InterSpeech, 2023, pp. 3522-3526..

Serialized Output Training (SOT) with Speaker Attribution

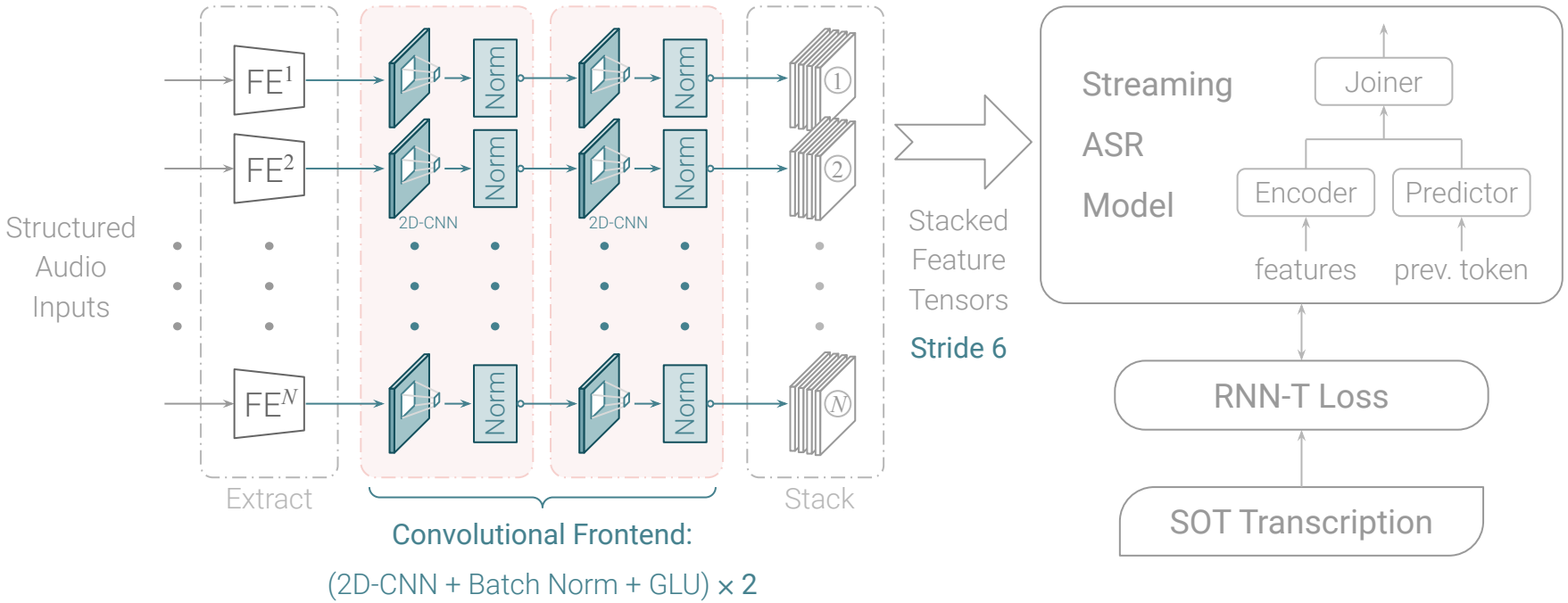


SOT: <SELF> yesterday, I was talking to your sister <OTHER> oh, okay! <SELF> Elizabeth! <OTHER> what did she say?

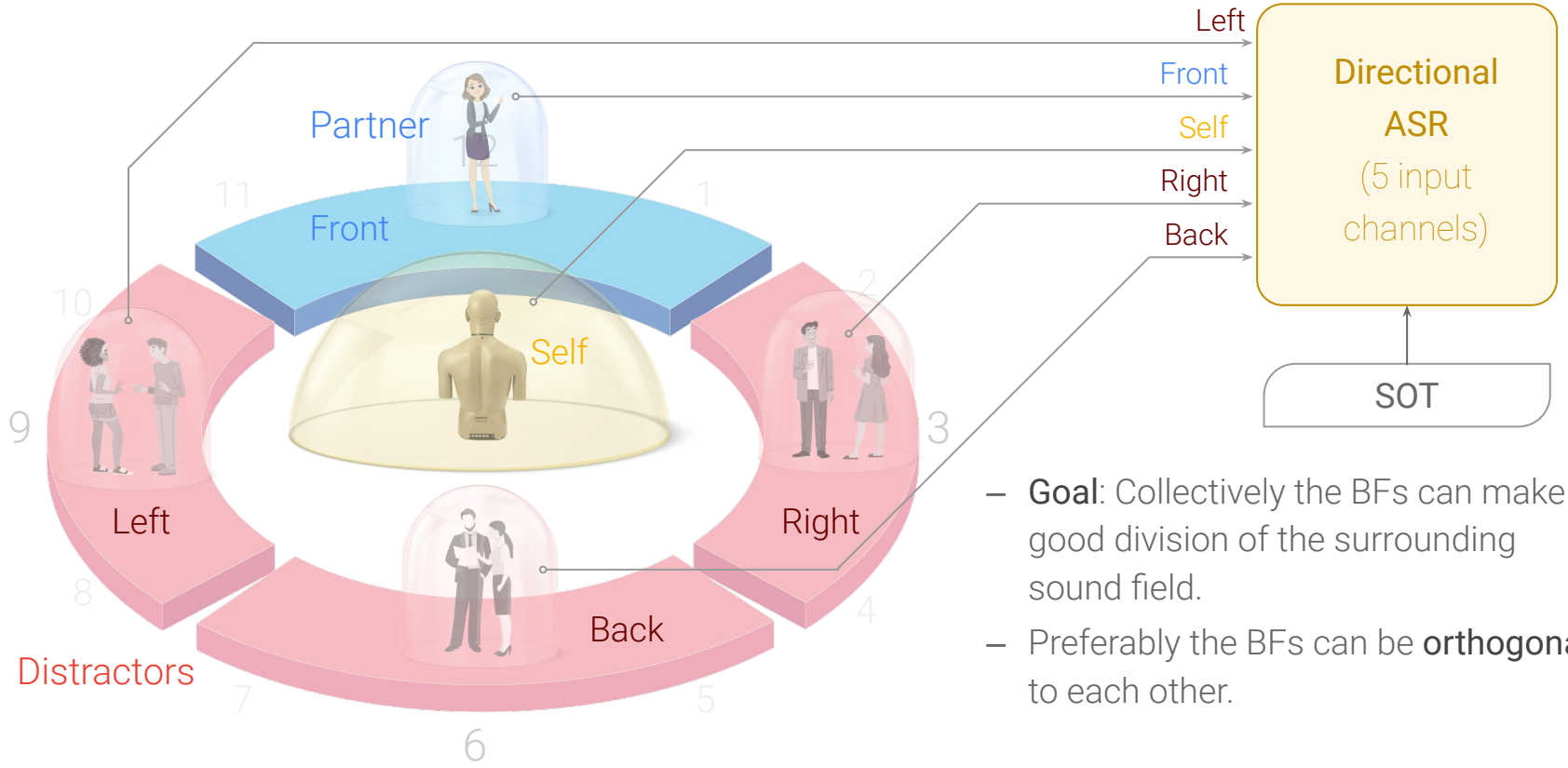
[1] N. Kanda, J. Wu, Y. Wu, X. Xiao, Z. Meng, X. Wang, Y. Gaur, Z. Chen, J. Li, and T. Yoshioka, “Streaming multi-talker ASR with token-level serialized output training,” in *Proc. InterSpeech*, 2022, pp. 3774-3778.

[2] X. Chang, N. Moritz, T. Hori, S. Watanabe, and J. Le Roux, “Extended graph temporal classification for multi-speaker end-to-end ASR,” in Proc. ICASSP, 2022, pp. 7322-7326.

Improvement #1: Convolutional Frontend

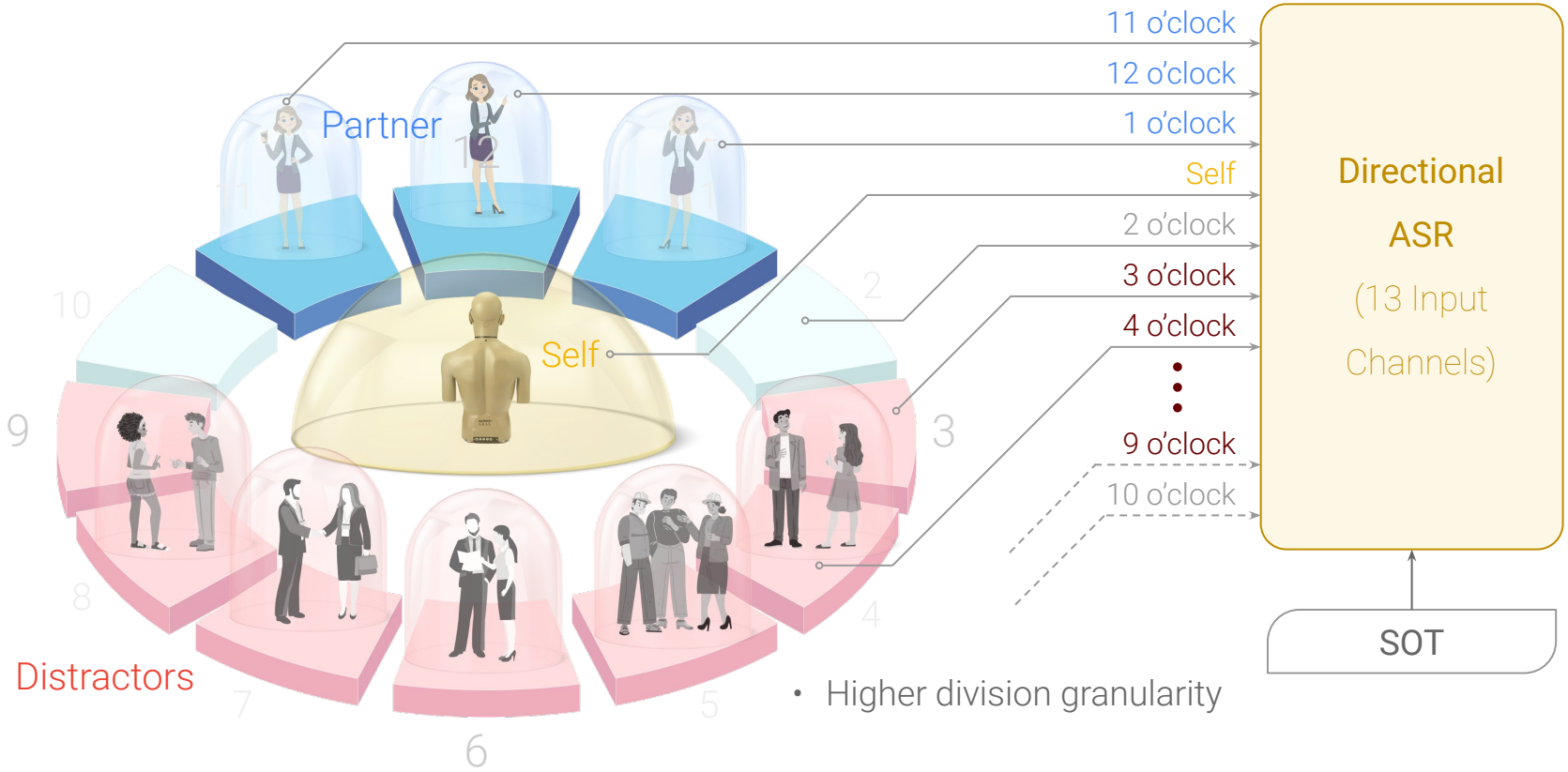


Improvement #2: Concerted Beamformers (BFs)



- **Goal**: Collectively the BFs can make a good division of the surrounding sound field.
- Preferably the BFs can be **orthogonal** to each other.

BF13: Sound Field Division with 13 Beams



- Higher division granularity

MVDR - Traditional Beamformer Design Method

- MVDR: Minimum Variance Distortionless Response (Capon Beamforming)

$$\underbrace{\min_{\mathbf{h}(j\omega)} \mathbf{h}^H(j\omega)\mathbf{\Phi}_{dd}(j\omega)\mathbf{h}(j\omega)}_{\text{MV}} \quad \text{subject to} \quad \underbrace{\mathbf{h}^H(j\omega)\mathbf{g}(j\omega) = 1.}_{\text{DR}}$$

MV: Minimize the variance of diffuse noise in the beamformer output.

DR: Subject to the constraint of no distortion on signals from the look direction.

- Using Lagrange multipliers, you can easily get:

$$\mathbf{h}_{\text{MVDR}}(j\omega) = \frac{\mathbf{\Phi}_{dd}^{-1}(j\omega)\mathbf{g}(j\omega)}{\mathbf{g}^H(j\omega)\mathbf{\Phi}_{dd}^{-1}(j\omega)\mathbf{g}(j\omega)}.$$

- A simple, elegant, and closed-form solution, but …
  a. White noise is not considered in the formulation; Have to check SWNR afterwards.
  b. No control of null directions, which may vary significantly from frequency to frequency.

NLCMV - Proposed Method

- Each BF considers the look directions of all other BFs as its "**SOFT**" null directions.
- NLCMV: **Non-Linearly Constrained Minimum Variance**

Soft control of null directions

$$\min_{\mathbf{h}(j\omega)} \mathbf{h}^H(j\omega) \left[\mathbf{\Phi}_{dd}(j\omega) + \phi_{pp}(\omega) \sum_{n=1}^{N} \alpha_{p,n} \cdot \mathbf{g}_n(j\omega) \mathbf{g}_n^H(j\omega) \right] \mathbf{h}(j\omega),$$

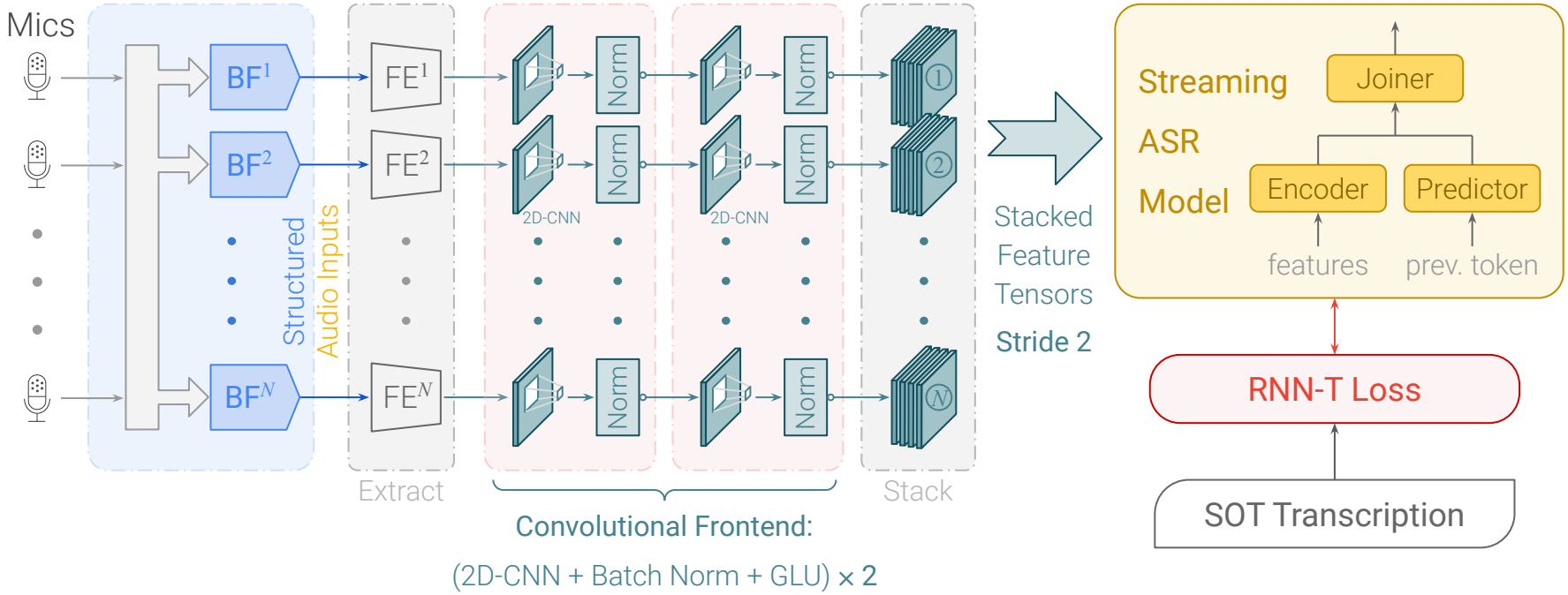
Constraint on white noise gain

$$\text{subject to} \begin{cases} \mathbf{h}^H(j\omega)\mathbf{g}(j\omega) = 1, \\ \dfrac{M}{\sum_{m=1}^{M} |G_m(j\omega)|^2} \cdot \dfrac{\left|\mathbf{h}^H(j\omega)\mathbf{g}(j\omega)\right|^2}{\left|\mathbf{h}^H(j\omega)\mathbf{h}(j\omega)\right|} \geq 1. \end{cases}$$

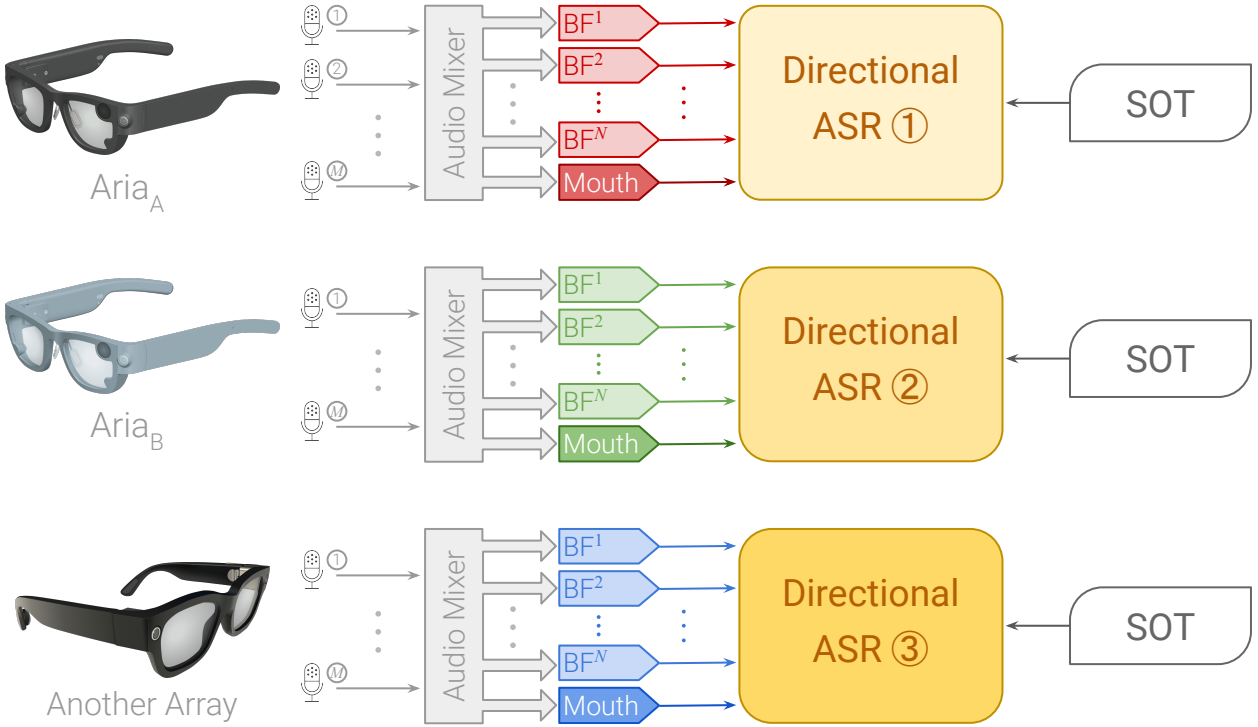
- The linear equality and nonlinear inequality constraints are simplified to the following form:

$$\begin{cases} \mathbf{h}^H(j\omega)\mathbf{g}(j\omega) = 1, \\ c(\omega) \triangleq \mathbf{h}^H(j\omega)\mathbf{\Psi}(j\omega)\mathbf{h}(j\omega) \leq 0, \end{cases} \quad \text{where,} \quad \mathbf{\Psi}(j\omega) \triangleq \mathbf{I} - \frac{M}{\sum_{m=1}^{M} |G_m(j\omega)|^2} \cdot \mathbf{g}(j\omega)\mathbf{g}^H(j\omega).$$

Improved Directional ASR – Overview

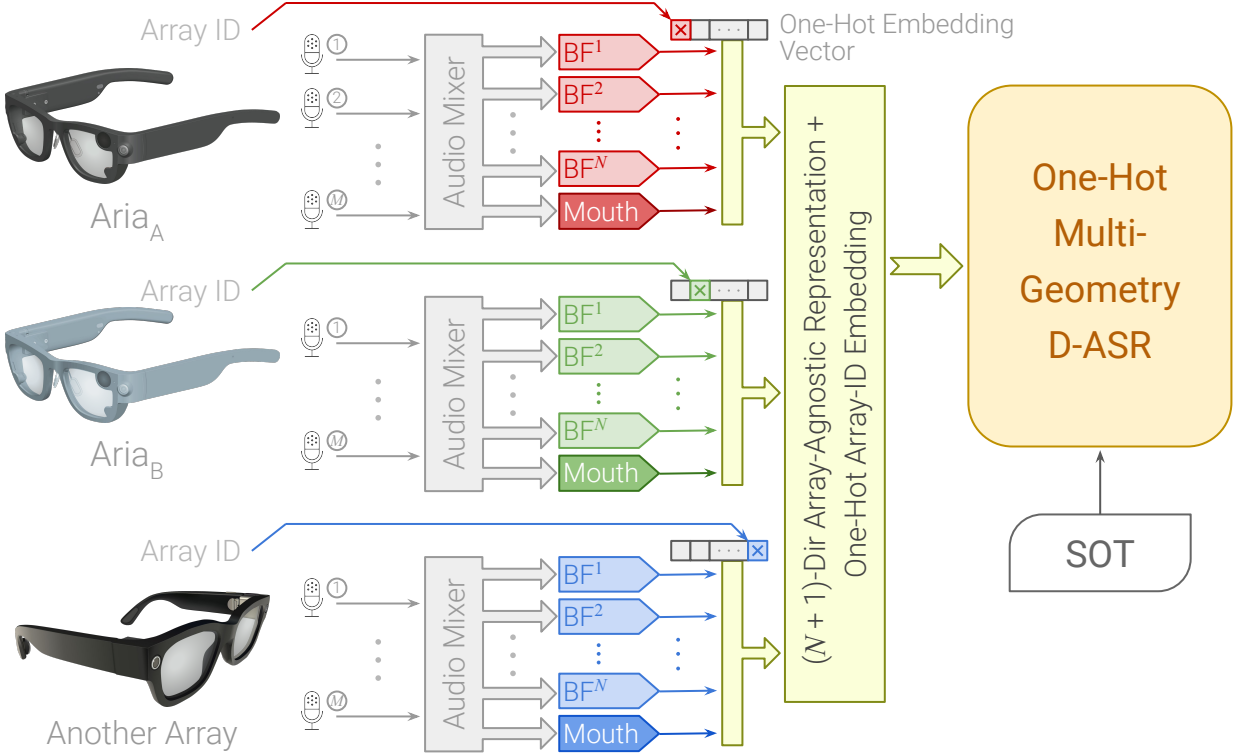


Array Specific Model Training



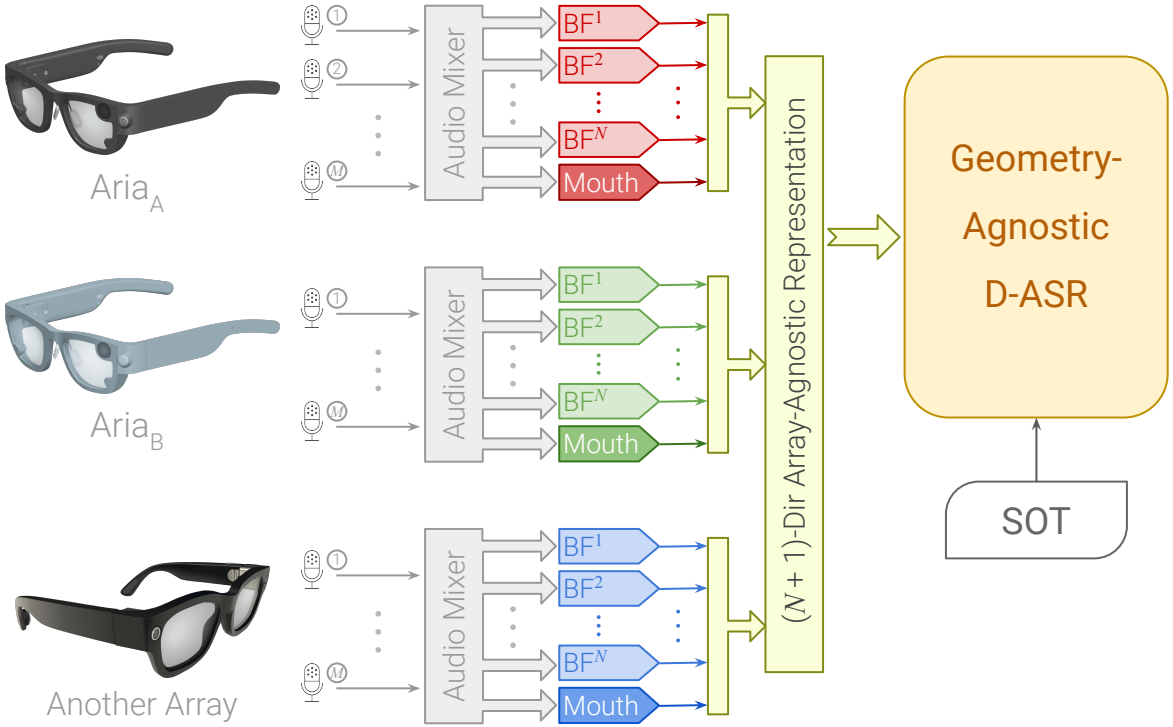
- Different ASR models are trained for different arrays/devices.
- Arrays must match between training and test.
- High cost in training and maintenance.

Towards Array Agnostic: One-Hot Multi-Geometry



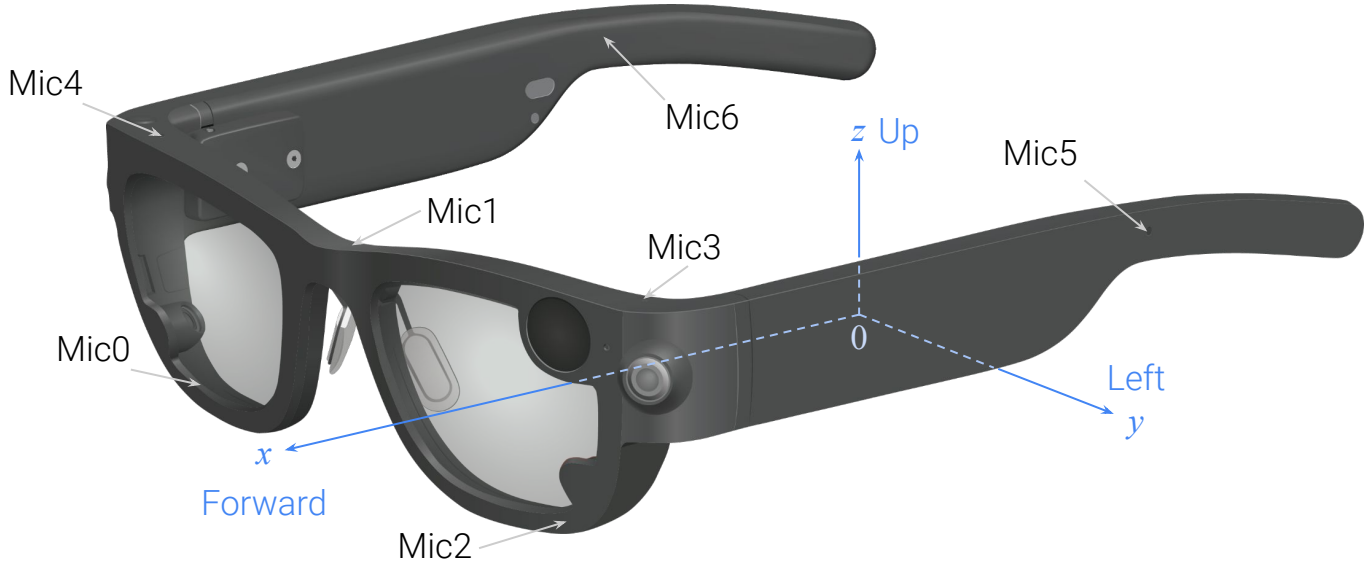
- Sound field division with (N + 1) array specific BFs yields an array agnostic representation for ASR.
- Only one D-ASR model is trained: Array-ID is encoded in a one-hot embedding vector.
- Test does NOT support previously unseen devices.

Towards Array Agnostic: Geometry-Agnostic



- Again, only one D-ASR model is trained.
- BF design is still array specific, **BUT** the D-ASR model is array agnostic.
- Training data is simulated with multiple arrays.
- Presumably robust to unseen devices & mic failure.

Project Aria Glasses (Publicly Available*)



* K. Somasundaram, et.al., "Project Aria: A new tool for egocentric multi-modal AI research," *arXiv preprint arXiv:2308.13561*, 2023.

Aria_A	Mics 2, 3, 4, 5, 6	Used in training
Aria_B	Mics 0, 3, 4, 5, 6	Not seen in training (for testing only)

Composite Prototype

Array Name	# Mics	Usage
$Comp_A$	5	Training
$Comp_B$		
$Comp_C$		
$Comp_D$		
$Comp_E$	5	Testing
$Comp_{A,4Mic}$	4	

- The prototype accommodates a substantially large number of microphones, enabling the definition of various configurations for 5-element mic arrays.

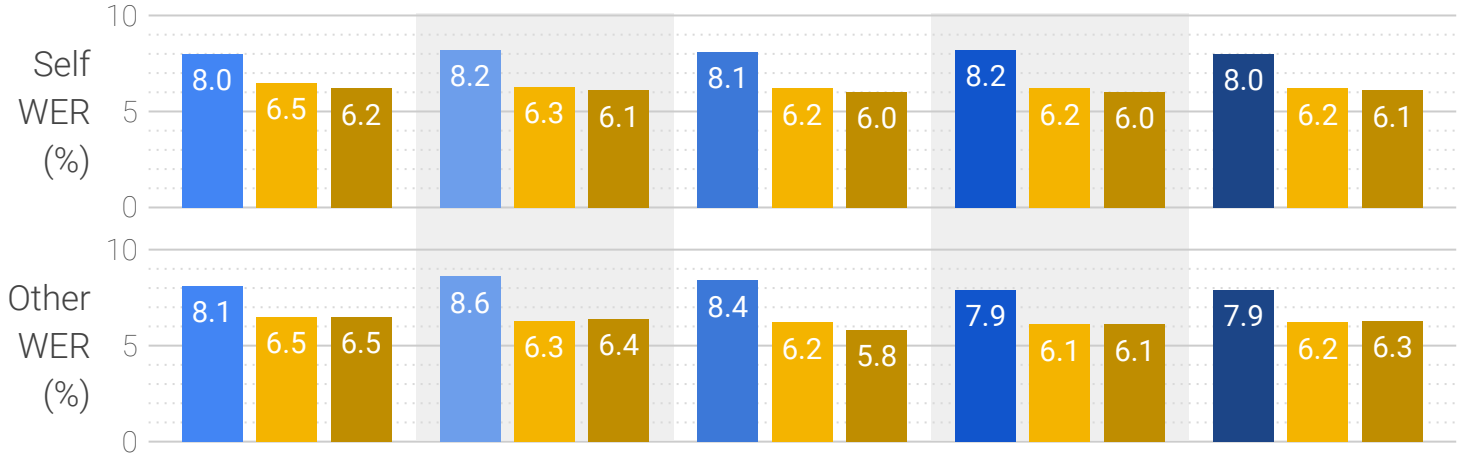
Test on Simulated Data (Matched arrays between training and testing)

① No Noise & Bystanders

② With N&B, 0% Overlap

③ With N&B, 50% Overlap

Composite Prototype



Array Specific

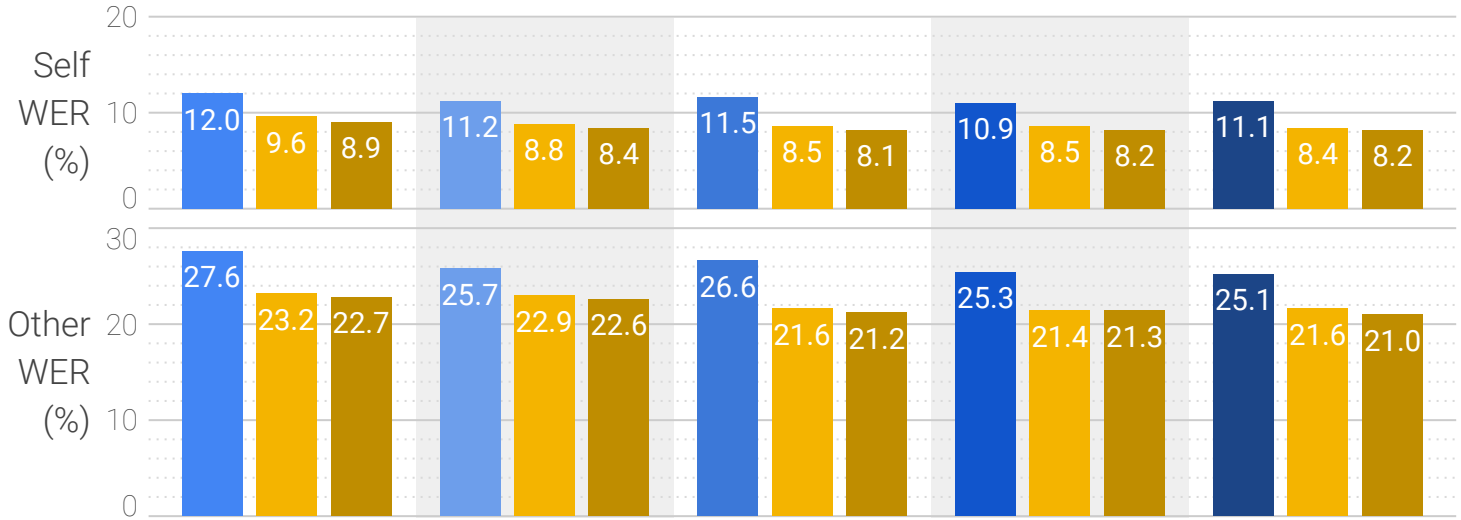
Embedding-Free Geometry-Agnostic

One-Hot Embedded Multi-Geometry

Test on Simulated Data (Matched arrays between training and testing)

① No Noise & Bystanders | **② With N&B, 0% Overlap** | ③ With N&B, 50% Overlap

Composite Prototype



Array Specific

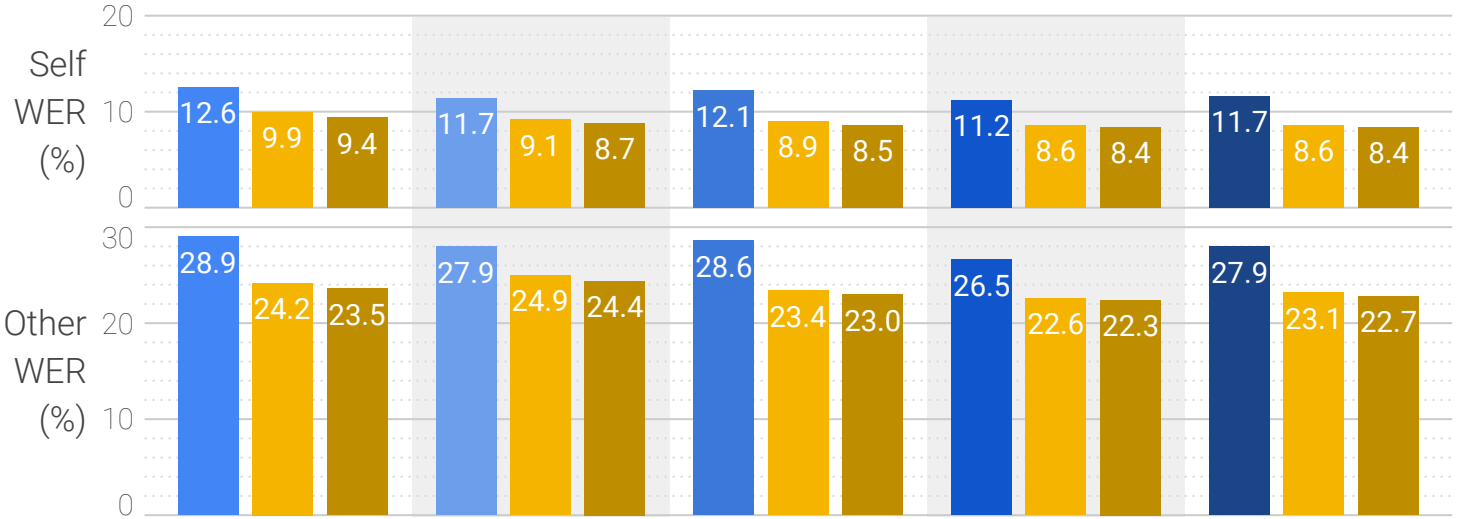
Embedding-Free Geometry-Agnostic

One-Hot Embedded Multi-Geometry

Test on Simulated Data (Matched arrays between training and testing)

① No Noise & Bystanders | ② With N&B, 0% Overlap | **③ With N&B, 50% Overlap**

Composite Prototype

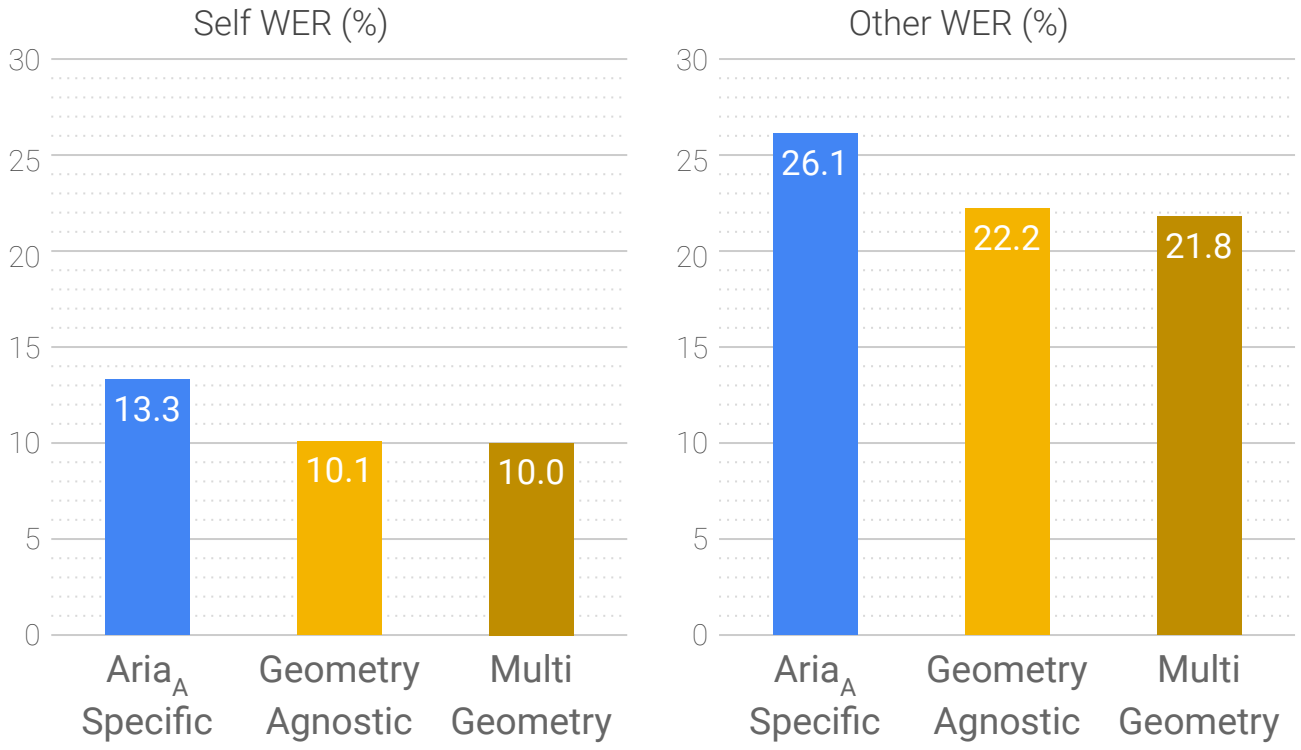


Array Specific

Embedding-Free Geometry-Agnostic

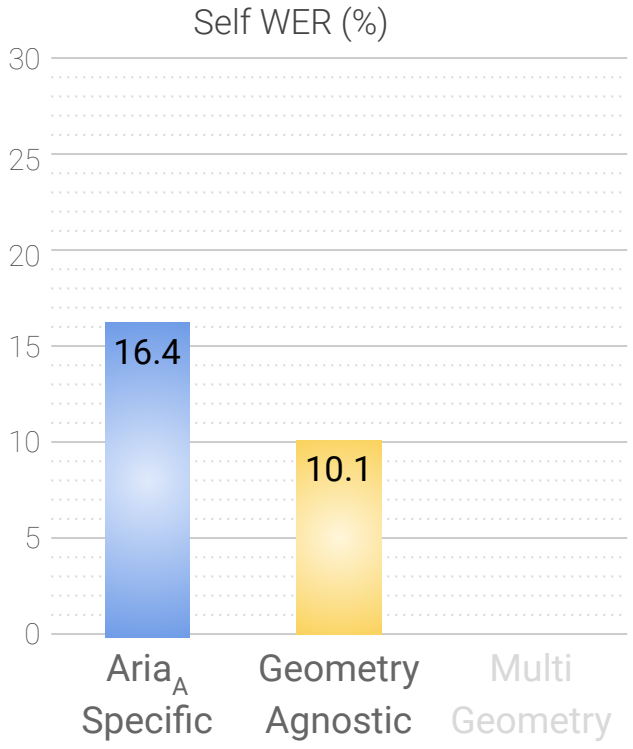
One-Hot Embedded Multi-Geometry

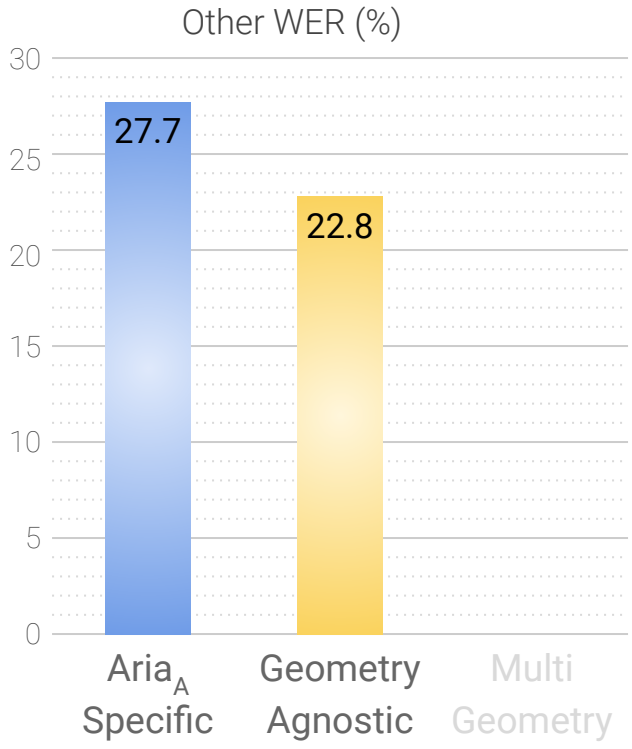
Agnosticity Test on Real Data (Recorded from Aria)



- Test data from **Aria**$_A$.
- **ALL** of the three models under test have seen data from **Aria**$_A$ during training.

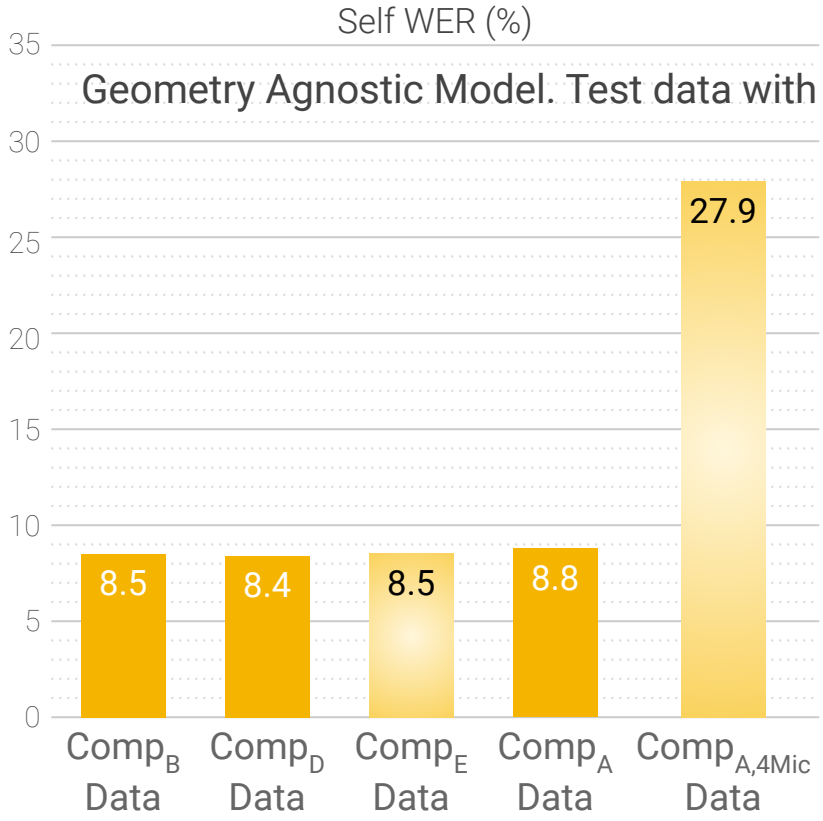
Agnosticity Stress Test on Real Data (Recorded from Aria)

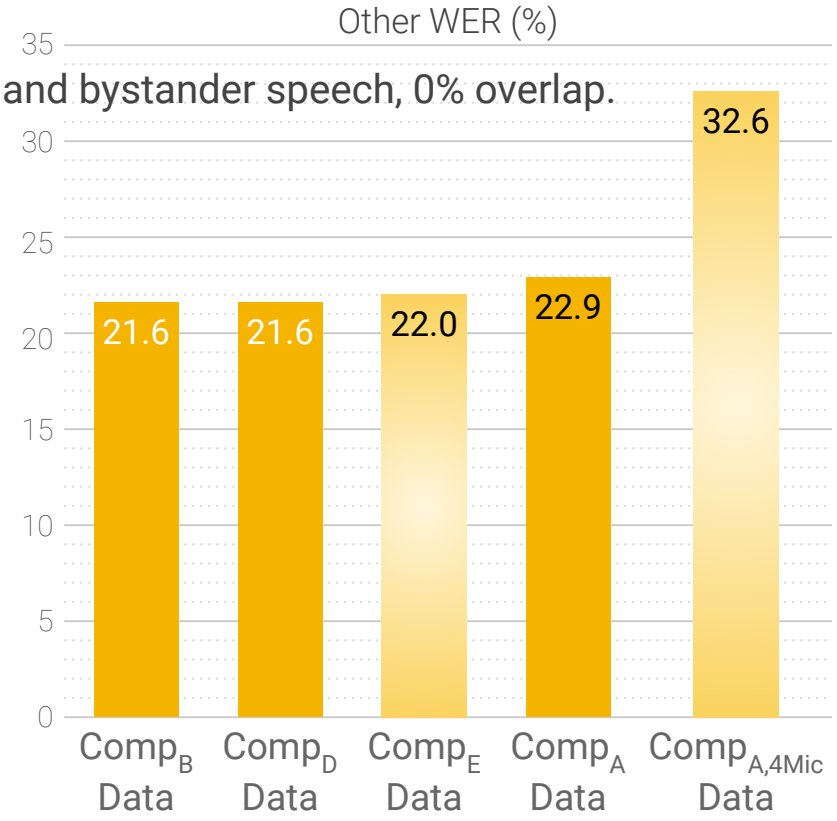




- Test data from **Aria$_B$**.
- **Note: None** of the models under test have seen data from **Aria$_B$** during training.
- The one-hot embedded multi-geometry model was NOT tested since it cannot handle unseen arrays.

Agnosticity Stress Test on Simulated Data





Summary

- Our previously proposed D-ASR models using SOT are effective, but were array specific.
- We propose to use beamformers to perform sound field division and create structured audio inputs for D-ASR models – these audio inputs are akin to array-agnostic representations.
- Two array-agnostic D-ASR models are proposed:
  embedding-free geometry-agnostic and one-hot-embedded multi-geometry.
- Comprehensive evaluations on both simulated and real-recorded test data confirmed the pursued array-agnostic ability for D-ASR.
- It is worth noting that the developed array-agnostic models outperformed array-specific baselines.

Thank You!