Move2Hear: Active Audio-Visual Source Separation Supplementary Material

Sagnik Majumder[1] Ziad Al-Halah[1] Kristen Grauman[1,2]
[1]The University of Texas at Austin [2]Facebook AI Research
{sagnik, ziad, grauman}@cs.utexas.edu

In this supplementary material we provide additional details about:

- Video (with audio) for qualitative assessment of our task setup and agent's performance (Sec. 1), as referenced in '3D Environment and Audio-Visual Simulator' of Sec. 3 in the main paper.
- Noisy audio experiment for the *far-target* task (Sec. 2).
- Experiment to show the effect of the minimum inter-source distance on separation quality (Sec. 3), as noted in Sec. 5.2 of the main paper.
- Experiment to demonstrate the importance of the *composite policy* for *far-target* separation (Sec. 4), as mentioned in Sec. 5.2 of the main paper.
- Experiment to show how the separation quality varies across all possible type of configurations of combining the target and the distractor sources. (Sec. 5), as noted in Sec. 5.2 of the main paper.
- Experiment to show how our Move2Hear approach maintains its benefits even when using a SOTA passive audio separation backbone (Sec. 6), as noted in Sec. 5.2 of the main paper.
- Experiment to show the effect of using waveform-level audio quality metrics like SNR as the RL reward on the separation performance (Sec. 7).
- Experiment to show how audio-visual navigation with distractor sources benefits from active audio-visual source separation (Sec. 8) as mentioned in Sec. 5.2 in the main paper.
- Evaluation metric definitions for evaluating source separation quality (Sec. 9).
- Additional baseline details for reproducibility (Sec. 10).
- Implementation details (Sec. 11), as noted in 'Experimental Setup' of Sec. 5 in the main paper.

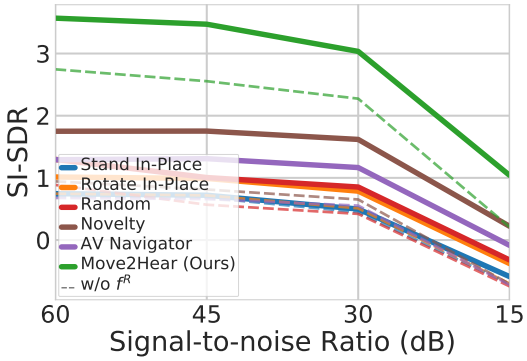


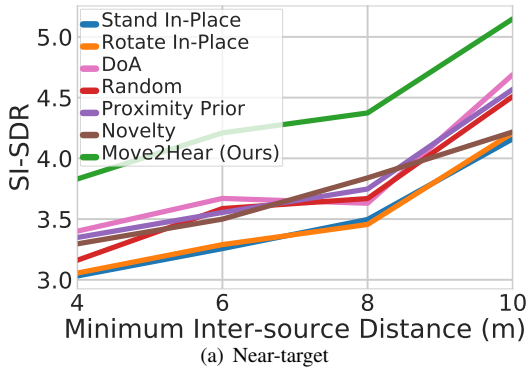
Figure 1: Models' robustness for various levels of noise in audio for *far-target*. Higher SI-SDR is better.

1. Supplementary Video

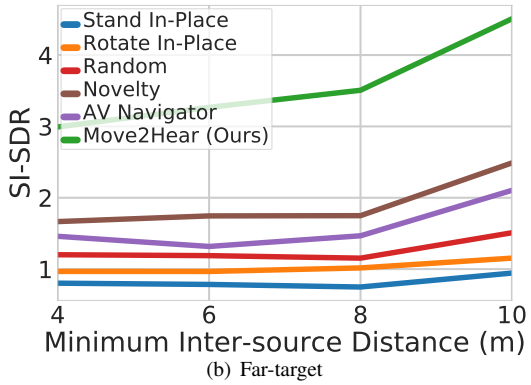
The supplementary video, available at http://vision.cs.utexas.edu/projects/move2hear, demonstrates the Active Audio-Visual Source Separation task with the SoundSpaces [3] audio simulation setup and shows the comparison between our proposed model and the baselines as well as qualitative results for both *near-target* and *far-target*. Please listen with headphones to hear the binaural audio correctly.

2. Noisy Audio for Far-Target

In the main paper submission we tested our model's robustness against standard microphone noise [23, 22] in the *near-target* task setup (see Fig. 4 in main). Here, we show the parallel experiment for the *far-target* (*heard* sounds) task setup. Fig. 1 shows the results. Our model is able to maintain its performance gain in the *far-target* setup over all other models even for very high levels of noise. In addition, we see that as in the case of *near-target*, our acoustic memory refiner module f^R again plays an important role in providing additional robustness against noisy audio; all models perform worse without it.



(a) Near-target



(b) Far-target

Figure 2: Models' robustness to various inter-source distances. Higher SI-SDR is better.

3. Minimum Inter-Source Distance

We investigate the effect of the minimum inter-source distance ('Experimental Setup' in Sec. 5 in main) on the separation performance of our model. This minimum distance is applied to all audio source pairs in every episode. Fig. 2 shows the results with *heard* sounds.

For both settings, our Move2Hear model outperforms all baselines by a significant margin. This shows that even in the challenging setting where the target and the distractor are quite close to each other (i.e., the maneuverability space around the target is reduced and the clash between the sounds of the target and the nearby distractor can be high), our model can still actively move around to effectively improve its separation quality.

4. Importance of Composite Policy

Our composite policy switches control between the navigation policy $\pi^{\mathcal{N}}$ and the quality policy $\pi^{\mathcal{Q}}$ based on the policy switch time $\mathcal{T}^{\mathcal{N}}$, where $\mathcal{T}^{\mathcal{N}}$ is selected based on performance in the validation split. Inspired by recent work in composite policy blending for complex multi-task robot learning [1, 5, 9, 19], this approach helps the agent deal with the challenges posed by the far-target task. When the agent is too far from the target audio location, the audio signal could be too weak and unreliable for $\pi^{\mathcal{Q}}$ to perform reasonably. Hence, $\pi^{\mathcal{N}}$ brings the agent to an area with a stronger signal and then passes control to $\pi^{\mathcal{Q}}$, which is expert in moving to improve $\ddot{M}^G$.

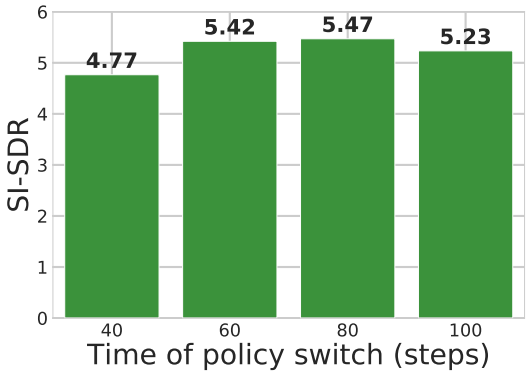


Figure 3: Effect of policy switch time in composite policy on separation performance for *far-target*.

Model	*Heard* SPL (↑)	*Heard* SR (↑)	*Unheard* SPL (↑)	*Unheard* SR (↑)
Random	3.1	6.4	3.1	6.4
Move Forward	1.1	1.1	1.1	1.1
Speaker-Target				
Gan et al. [6]	5.2	12.3	4.3	10.3
AV Navigator [3]	33.5	49.1	32.4	47.0
Move2Hear [$\pi^{\mathcal{N}}$ + *Stop*] (**Ours**)	**56.0**	**70.0**	**51.4**	**66.0**
Standard Split				
Gan et al. [6]	4.3	10.0	4.9	11.1
AV Navigator [3]	0.9	1.5	1.1	1.6
Move2Hear [$\pi^{\mathcal{N}}$ + *Stop*] (**Ours**)	**54.9**	**70.3**	**52.2**	**68.5**

Table 1: Audio-visual navigation with distractors. Higher SPL and SR are better.

Fig. 3 shows Move2Hear's separation performance on the validation data for different values of the policy switch time $\mathcal{T}^{\mathcal{N}}$ in the *far-target* and *heard* setting. Switching over from the navigation policy $\pi^{\mathcal{N}}$ to the quality improvement policy $\pi^{\mathcal{Q}}$ very early negatively affects our model's performance as it does not allow the agent to be close enough to the source for $\pi^{\mathcal{Q}}$ to make successful fine-grained movements for further improvement in separation quality. On the other hand, if there is no switching at all ($\mathcal{T}^{\mathcal{N}} = 100$) , the agent suffers from not leveraging $\pi^{\mathcal{Q}}$'s ability to take it to "sweet spots" in the vicinity of the target where the target audio can be separated even better. Overall, the composite policy is beneficial for best results, and we see the model is not overly sensitive to the switch point.

5. Separation in Different Scenarios

Separating speech (S) from a mixture of speeches is the most difficult, and extracting music (M) from background (B) is the easiest for Move2Hear and two of our strongest baselines in the *near-target* scenario, Novelty [2] and Proximity Prior. For the SI-SDR scores, refer to Table 2.

	Model	S vs. S	S vs. M	S vs. B	M vs. B
Heard	Proximity Prior	3.42	5.07	3.38	6.82
	Novelty [2]	3.56	4.84	3.85	5.83
	Move2Hear (Ours)	**4.04**	**5.50**	**4.58**	**6.95**
Unheard	Proximity Prior	2.53	2.78	2.69	3.36
	Novelty [2]	2.82	3.19	3.25	3.82
	Move2Hear (Ours)	**3.08**	**3.31**	**3.60**	**4.15**

Table 2: SI-SDR performance in different separation scenarios on *near-target*.

6. Comparison with SOTA passive separation model.

Passive audio(-visual) separation is distinct from AAViSS in that it 1) assumes access to pre-recorded audio/video, 2) has no provision for sensor motion to improve separation, and 3) doesn't extract the target latent (monaural) audio ('Passive Audio(-Visual) Source Separation' of Sec. 2 in main). Furthermore, advances in passive audio separation models are orthogonal to our contribution. We demonstrate this by replacing the audio network backbone with the passive SOTA MMDenseNet [21] On SI-SDR and heard (unheard) setting, our active model still outperforms Stand In-Place in both *near-* and *far-target* settings (Table 3).

	Near-Target		Far-target	
Model	*Heard*	*Unheard*	*Heard*	*Unheard*
Stand In-Place	7.10	4.43	3.45	1.90
Move2Hear (Ours)	**7.98**	**5.38**	**6.84**	**4.57**

Table 3: Effect of using a stronger model like MMDenseNet [21] for passive separation on SI-SDR performance .

7. SNR as RL reward

In our approach, we used the source separation error in the formulation of the agent's reward. Here, we explore an alternative option for the reward formulation by replacing the separation error with SNR (signal to noise ratio) to capture the improvement in quality of the waveform-level of the separated audio. We find that this alternative RL reward doesn't improve the separation performance. On the contrary, when using SNR as a reward with Move2Hear we see a relative degradation in SI-SDR by 5.1% and 2.2% in near-target and heard/unheard setting in comparison to the original reward formulation. Further, SNR increases training time by 2.2x due to the needed inverse-STFT calculations. Our separation reward leads to better performance and faster training.

8. Audio-visual Navigation with Distractors

While our main goal is source separation, we find that as a byproduct, our model can benefit AV navigation in the presence of cluttered sounds. Whereas existing models trained to navigate to a source are naturally confused by distractors, our $\pi^{\mathcal{N}}$ navigation policy (augmented with a *Stop* action) can successfully ignore them to more rapidly find a target source. To illustrate this, we use the *far-target* dataset and we compare our $\pi^{\mathcal{N}}$ policy to the following models in terms of navigation performance:

- Random: an agent that selects a random action at each step.
- Move Forward: an agent that always moves forward unless faced with an obstacle, then it turns right. This is a common baseline employed in the visual navigation literature [17, 3].
- AV Navigator [3]: this is the same baseline model we used in the main paper for the *far-target* task but evaluated here for navigation performance.
- Gan et al. [6]: this approach trains two supervised models using the binaural audio input, one for predicting the target location and the other for predicting a *Stop* action. During navigation, the method of Gan et al. [6] uses egocentric depth images to build an occupancy map of the environment and plans a path to the predicted location using a metric planner. We set the target location prediction frequency to every 20 steps of navigation on the basis of validation.

All models are evaluated using standard navigation metrics: success rate (SR) and success rate weighted by path length (SPL).

Table 1 shows the results. On the *Standard Split* when the target and distractor types intersect, both [3] and [6] are overwhelmed by the mixed audio and show poor navigation performance. We observe that per-step prediction for the Gan et al. model yields a very reactive navigation policy in our setup, which leads to low navigation performance. On an easier split where the target is always of speaker type and the distractors are never other speakers (*Speaker-Target*), the learned baselines fare better. However, our model outperforms all baselines by a substantial margin in both setups, showing the positive impact of using separated audio for navigation with distractor sounds.

9. Metric Definitions

Next we elaborate on the metric definitions ('Evaluation' of Sec. 5 in main).

1. **STFT distance –** The Euclidean distance between the ground-truth and predicted complex monaural spectrograms,
$$\mathcal{D}_{\{STFT\}} = ||\ddot{\boldsymbol{M}}^G - \boldsymbol{M}^G||_2.$$
2. **SI-SDR [16] –** We use a fast implementation from the nussl [13] library to measure the source-to-distortion ratio (SDR) of the predicted monaural waveforms in dB in a scale-invariant (SI) manner.

10. Baselines

We provide the following additional details about the baselines ('Baselines' of Sec. 5 in main) for reproducibility.

- **DoA:** To face the audio target, this agent starts rotating to the right from its initial pose until it finds an orientation that allows it to move to a neighboring node. Once it has moved to a neighboring node, it rotates twice to face the agent and make its first prediction.
- **Proximity Prior:** Whenever this agent tries to cross the boundary of the circle within which it is supposed to stay, it is forced to randomly choose an action from $\{TurnLeft, TurnRight\}$ by the simulation platform.
- **Novelty [2]:** this agent is rewarded on the basis of the novelty of states visited. Each valid node of the SoundSpaces [3] grids is considered to be a unique state. When an agent visits any such node, the count for that state is incremented. The novelty reward is given by:
$$r_t = \frac{1}{\sqrt{n_s}}, \tag{1}$$
where n_s is the visitation count of state s_t.
- **AV Navigator [3]:** this navigation agent uses a visual and an audio encoder for input feature representation and an actor-critic policy network for predicting actions to navigate to the target source. While the visual encoder takes RGB images as input, the audio encoder takes the mixed binaural spectrogram concatenated with the target class label as an extra channel as input. Thus, the audio input space is the same as the input space of f^B ('Binaural Audio Separator' of Sec. 4.1 in main). Following typical navigation rewards [3, 17], we reward the agent with +10 if it succeeds in reaching the target source and executing the Stop action there, plus an additional reward of +0.25 for reducing the geodesic distance to the target and an equivalent penalty for increasing it. Finally, we issue a time penalty of -0.01 per executed action to encourage efficiency.

11. Implementation Details

Next we provide further implementation details including the network architecture details.

11.1. Monaural Audio Preprocessing

For all our experiments, we sample 1-second-long monaural clips ('Experimental Setup' of Sec. 5 in main) at 16kHz and ensure that the sampled clips have a higher average power than the full audio clip that they are sampled from. This helps us prevent the sampling of a large amount of mostly silent raw audio data. The sampled waveforms are further encoded using the standard 32-bit floating point format and normalized to have the same average power of 1.2 across the whole dataset.

11.2. Audio Spectrogram

To generate an audio spectrogram, we compute the Short-Time Fourier Transform (STFT) with a Hann window of length 63.9ms, hop length of 32ms, and FFT size of 1023. This results in complex spectrograms of size $512 \times 32 \times C$, where C is the number of channels in the source audio (C is 1 for monaural and 2 for binaural audio). For all experiments, we take the magnitude of the spectrogram, compute its natural logarithm after adding 1 to all its elements for better contrast [7, 8], and reshape it to $32 \times 32 \times 16C$ by taking slices along the frequency dimension and concatenating them channel-wise to improve training speed. For all cases where the target audio class needs to be concatenated to the spectrogram channel-wise, the concatenation is carried out after slicing.

11.3. CNN Architecture Details

Binaural Audio Separator. The binaural audio separator f^B uses a U-Net style architecture [15] ('Binaural Audio Separator' of Sec. 4.1 in main). The encoder of the network has 5 convolution layers. Each convolution layer uses a kernel size of 4, a stride of 2 and a padding of 1. It is followed by a Batch Normalization [11] of $1e^{-5}$ and a leaky ReLU [14, 20] activation with a negative slope of 0.2. The number of output channels of the convolution layers are [64, 128, 256, 512, 512], respectively. The decoder consists of 5 transpose convolution layers and 1 convolution layer in the end to resize the output from the transpose convolutions to the desired spectrogram dimensions. Each transpose convolution has a kernel size of 4, a stride of 2 and a padding of 1, and is followed by a Batch Normalization [11] of $1e^{-5}$ and a ReLU activation [14, 20]. The final convolution layer uses a kernel size of 1 and a stride of 1.

Monaural Audio Predictor. The monaural audio predictor f^M uses the same architecture as f^B.

Acoustic Memory Refiner. The acoustic memory refiner f^R is a CNN network with 2 convolution layers. Both convolutions use a kernel size of 3, a stride of 1 and a padding of

1. Additionally, the first convolution is followed by a Batch Normalization [11] of $1e^{-5}$ and a ReLU activation [14, 20].

Visual Encoder. The visual encoder E^V of Move2Hear is a CNN with 3 convolution layers, where the convolution kernel sizes are [8, 4, 3], the strides are [4, 2, 1] and the number of output channels are [32, 64, 32], respectively. Each convolution layer has a ReLU activation [14, 20] function. The convolution layers of the encoder are followed by 1 fully connected layer with 512 output units. Note that the visual encoders of the AV Navigator [3] and Novelty [2] baselines share the same architecture.

Separated Binaural Encoder. Our separated binaural encoder E^B uses the same architecture as E^V, except for using a kernel size of 2 in place of 3 for the third convolution.

Policy Network. The policy network for Move2Hear, as well as for the AV Navigator [3] and Novelty [2] models, uses a one-layer bidirectional GRU [4] with 512 hidden units. The actor and the critic networks consist of one fully connected layer.

Predicted Monaural Encoder. Our predicted monoaural encoder E^M uses the same architecture as E^B.

We use the Kaiming-normal [10] weight initialization strategy for all weight initializations in the network components (f^B, f^R, f^M) of the Target Audio Separator, all feature encoders (E^V, E^B, E^M) of the Active Audio-Visual Controller, and the visual encoder of the AV Navigator [3] and Novelty [2] models.

11.4. Training Hyperparameters

We pretrain f^B and f^M by randomly sampling a maximum of 30K data samples per training scene (Eq. 6 in Sec. 4.3 in main). We optimize the loss functions $\mathcal{L}^B$ (Eq. 6 in Sec. 4.3 in main) and $\mathcal{L}^M$ (Eq. 7 in Sec. 4.3 in main) by using Adam [12] and a learning rate of $5e^{-4}$ until convergence.

To train the policies of Move2Hear, AV Navigator [3], and Novelty [2] using PPO [18] ('Training the Active Audio-Visual Controller' of Sec. 4.3 in main), we weight the action loss by 1.0 and the value loss by 0.5. For $\pi^{\mathbb{Q}}$, we use an entropy loss on the policy distribution with a coefficient of 0.01 while for all other policies, we set the coefficient to 0.2. We train all policies with Adam [12] and a learning rate of $1e^{-4}$ for a total of 38 million policy prediction steps.

References

[1] J. Barry, L. P. Kaelbling, and T. Lozano-Pérez. A hierarchical approach to manipulation with diverse actions. In *2013 IEEE International Conference on Robotics and Automation*, pages 1799–1806, 2013. 2

[2] Marc G Bellemare, Sriram Srinivasan, Georg Ostrovski, Tom Schaul, David Saxton, and Remi Munos. Unifying count-based exploration and intrinsic motivation. *arXiv preprint arXiv:1606.01868*, 2016. 3, 4, 5

[3] Changan Chen, Unnat Jain, Carl Schissler, Sebastia Vicenc Amengual Gari, Ziad Al-Halah, Vamsi Krishna Ithapu, Philip Robinson, and Kristen Grauman. SoundSpaces: Audio-visual navigation in 3D environments. In *Proceedings of the European Conference on Computer Vision (ECCV)*. Springer, 2020. 1, 2, 3, 4, 5

[4] Junyoung Chung, Kyle Kastner, Laurent Dinh, Kratarth Goel, Aaron C Courville, and Yoshua Bengio. A recurrent latent variable model for sequential data. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015. 5

[5] Coline Devin, Abhishek Gupta, Trevor Darrell, Pieter Abbeel, and Sergey Levine. Learning modular neural network policies for multi-task and multi-robot transfer. In *2017 IEEE international conference on robotics and automation (ICRA)*, pages 2169–2176. IEEE, 2017. 2

[6] Chuang Gan, Yiwei Zhang, Jiajun Wu, Boqing Gong, and Joshua B Tenenbaum. Look, listen, and act: Towards audio-visual embodied navigation. In *ICRA*, 2020. 2, 3

[7] Ruohan Gao and Kristen Grauman. 2.5d visual sound. In *CVPR*, 2019. 4

[8] Ruohan Gao and Kristen Grauman. Co-separating sounds of visual objects. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3879–3888, 2019. 4

[9] Tuomas Haarnoja, Vitchyr Pong, Aurick Zhou, Murtaza Dalal, Pieter Abbeel, and Sergey Levine. Composable deep reinforcement learning for robotic manipulation. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6244–6251. IEEE, 2018. 2

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, December 2015. 5

[11] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. PMLR, 2015. 4, 5

[12] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5

[13] Ethan Manilow, Prem Seetharaman, and Bryan Pardo. The northwestern university source separation library. Proceedings of the 19th International Society of Music Information Retrieval Conference (ISMIR 2018), Paris, France, September 23-27, 2018. 4

[14] Vinod Nair and Geoffrey E. Hinton. Rectified linear units improve restricted boltzmann machines. In *ICML*, pages 807–814, 2010. 4, 5

[15] O. Ronneberger, P.Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention (MIC-*

CAI), volume 9351 of *LNCS*, pages 234–241. Springer, 2015. (available on arXiv:1505.04597 [cs.CV]). 4

[16] J. L. Roux, S. Wisdom, H. Erdogan, and J. R. Hershey. Sdr – half-baked or well done? In *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 626–630, 2019. 4

[17] Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, Devi Parikh, and Dhruv Batra. Habitat: A Platform for Embodied AI Research. In *ICCV*, 2019. 3, 4

[18] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017. 5

[19] Zhe Su, Oliver Kroemer, Gerald E Loeb, Gaurav S Sukhatme, and Stefan Schaal. Learning manipulation graphs from demonstrations using multimodal sensory signals. In *2018 IEEE international conference on robotics and automation (ICRA)*, pages 2758–2765. IEEE, 2018. 2

[20] Yi Sun, Xiaogang Wang, and Xiaoou Tang. Deeply learned face representations are sparse, selective, and robust. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2892–2900, 2015. 4, 5

[21] Naoya Takahashi and Yuki Mitsufuji. Multi-scale multi-band densenets for audio source separation. In *2017 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pages 21–25. IEEE, 2017. 3

[22] Ryu Takeda and Kazunori Komatani. Unsupervised adaptation of deep neural networks for sound source localization using entropy minimization. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2217–2221. IEEE, 2017. 1

[23] Ryu Takeda, Yoshiki Kudo, Kazuki Takashima, Yoshifumi Kitamura, and Kazunori Komatani. Unsupervised adaptation of neural networks for discriminative sound source localization with eliminative constraint. *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3514–3518, 4 2018. 1