Supplementary Material for Sound Localization from Motion: Jointly Learning Sound Direction and Camera Rotation

A.1. Camera pose from audio prompting

We illustrate our prompting idea in Fig. 9. To create our audio prompts, we simulate 181 binaural RIRs at different angles from $[-90°, 90°]$ without reverberation using SoundSpaces [15] and render with audio signals from LibriSpeech [69]. We use the sound with an angle of $0°$ as the input prompt $\mathbf{a}_s$ (the source view audio) and mix it into mono audio as the input at the target viewpoint. We calculate the interaural intensity difference (IID) cues for the audio prompts $\mathbf{a}_i$ and generated audio $\hat{a}_t$. We use L1 distance between IID cues to find the nearest neighbors:

$$\arg\min_{\mathbf{A}_i} \left| \log_{10} \frac{\hat{\mathbf{A}}_t^L}{\hat{\mathbf{A}}_t^R} - \log_{10} \frac{\mathbf{A}_i^L}{\mathbf{A}_i^R} \right|, \tag{11}$$

where $\mathbf{A}_i = \text{STFT}(\mathbf{a}_i)$. We use ground truth annotations of sound directions from the nearest prompts to predict the camera rotation angles. We first obtain rotation prediction votes from 1024 audio prompts and use a RANSAC-like mode estimation [26, 17] to get the final prediction.

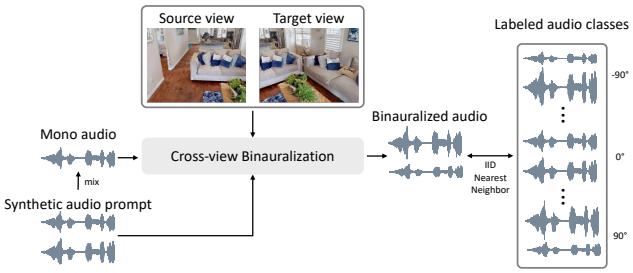

Figure 9: **Estimating camera pose from audio prompting.** We estimate camera rotation by providing our cross-view binauralization model with synthetically generated audio prompts. Given the sound that it predicts, we infer the camera angle. We do this by finding the nearest neighbor (using IID cues) to a database of synthetic sounds, each paired with their corresponding angle.

A.2. Additional experimental results

Evaluating pretext task. We also evaluate the performance of our model on the pretext task, which involves binauralizing sound at a novel microphone pose using sound from a different viewpoint and visual cues from both views as references. We use the STFT distance between the predicted and ground-truth spectrogram to measure the audio reconstruction performance. As the results are shown in Tab. 7, our model that incorporates both visual and audio features as input performs the best and is comparable to the model that receives ground truth rotation angles as inputs. This suggests that our model effectively uses the spatial information in both visual and audio signals to solve binauralization tasks, and encourages the network to learn useful representations. Moreover, the results show that training with more viewpoints improves the performance of the pretext task.

	Model	Input features $\mathcal{V}$	Input features $\mathcal{A}$	STFT distance ↓
Mono2Binaural	Random	✓	✓	0.368
	Ours (2 views)	✓		0.206
			✓	0.161
		✓	✓	0.130
	Ours–GTRot		✓	0.131
	Ours (3 views)	✓	✓	**0.125**

Table 7: **Reconstruct performance of cross-view binauralization pretext task.** We report the STFT distance performance of variants of our models with different input features on HM3D-SS dataset with LibriSpeech samples [69]. $\mathcal{V}$ and $\mathcal{A}$ mean visual and audio features, respectively.

Experiment on FreeMusic. We report the performance of downstream tasks with learned representations on the HM3D-SS dataset with FreeMusic [20] samples in Tab. 8. We outperform baselines and learn a useful representation.

	Model	Audio Loc. Acc (%) ↑	Camera Rot. Acc (%) ↑
FreeMusic	Random feature	6.0	4.7
	ImageNet [38]+Random	–	56.3
	RotNCE [27]	46.3	–
	AVSA [63]	66.5	6.7
	Ours–L2R (3 views)	**72.0**	76.5
	Ours (2 views)	67.5	76.2
	Ours (3 views)	67.5	**81.1**
	Supervised	77.1	95.8

Table 8: **Downstream task performance on HM3D-SS dataset with FreeMusic [20] samples.** We report linear probe performance on the audio localization and camera rotation downstream tasks.

SLfM without pretraining. We further demonstrate the important role of the features learned from our cross-view binaural pretext task by training our SLfM model with random features. We show results in Tab. 9. We can see that the models perform better using our feature representations, which emphasizes the significance of our pretext task. Our SLfM model finetuned from random features achieves accurate predictions, highlighting that our proposed method successfully leverages the geometrically consistent changes between visual and audio signals.

Model	Init. feature	Audio angle MAE (°) ↓	Camera angle MAE (°) ↓
Ours	Random (freeze)	36.51	29.26
Ours	Random (finetune)	3.92	1.32
Ours	M2B (freeze)	3.17	0.77
Ours	M2B (finetune)	**2.77**	**0.76**

Table 9: **SLfM results with different features.** We evaluate our SLfM models trained with different feature initialization on HM3D-SS.

A.3. Ablation study

Audio prediction network. We study how audio prediction architectures will influence representation learning from our proposed pretext task. We adapt the U-Net architecture with cross-attention modules for conditional feature inputs [79, 88] and compare the pretext and downstream performance with U-Net [30] we used for our main experiments. We train our models on the HM3D-SS dataset with a single sound source presented in the scenes and use LibriSpeech signals [69]. We report results in Tab. 10. Interestingly, we found that ATTN U-Net can reconstruct better sounds for the pretext task while it does not learn the features as well as the 2.5D U-Net [30]. We hypothesize that a more complex network may transfer the representation learning inside of the prediction networks rather than the feature extractors.

Model	Pretext ↓	Downstream Acc (%) ↑	
	STFT Dist.	AudLoc.	CamRot.
ATTN U-Net [79, 88]	**0.128**	68.0	75.3
2.5D U-Net [30]	0.130	**74.5**	**80.0**

Table 10: **Audio prediction model ablation study.** We evaluate both pretext and downstream performance on the HM3D-SS with LibriSpeech samples [69].

Robustness to reverberation. We also evaluate our representation under the influence of reverberation. We report linear probe performance on downstream tasks with average reverberation $\text{RT}_{60} \in \{0.1, 0.2, 0.3, 0.4, 0.5\}$. As shown in Fig. 10, the results indicate a decrease in downstream performances as the level of reverberation increases, where audio becomes more challenging during both training and testing.

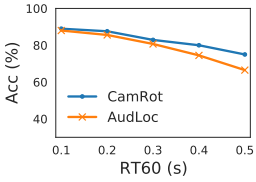

Figure 10: **Robustness to reverberation.** We study the effect of reverberation on our pretext model. Chance performance is 1.5%.

Weights of geometric loss. We assign appropriate weights for the geometric loss (Eq. (6)) to avoid it from dominating the optimization. In our approach, we search λ from 1 to 10 during training, and we select models weights using a metric by calculating $1/(100 \cdot \mathcal{L}_{\text{geo}} + \mathcal{L}_{\text{binaural}} + \mathcal{L}_{\text{sym}})$ during validation. We show the search experiment in Fig. 11. The performance is relatively stable when $\lambda \in [3, 8]$. We select $\lambda = 5$ or 3 in the main paper, please see Appendix A.4 for details.

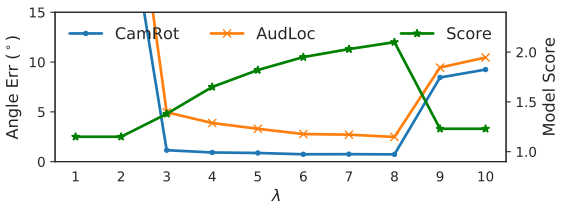

Figure 11: **Hyparameter search.** We experiment with λ from 1 to 10 and monitor the scores of models.

A.4. Implementation details

SLfM model. We use separate multi-layer perceptrons g_v and g_a (*i.e.*, FC ($512 \to 256$)–ReLU–FC ($256 \to 1$) layers) to predict scalar rotation and sound angles.

Hyperparameters. For all experiments, we re-sample the audio to 16kHz and use 2.55s audio for the binauralization task. For pretext training, we use the AdamW optimizer [46, 53] with a learning rate of 10^{-4}, a cosine decay learning rate scheduler, a batch size of 96, and early stopping. During downstream tasks, we change the learning rate to 10^{-3} for linear probing experiments. To train our self-supervised pose estimation model, we set the weights λ of geometric loss to be 5 and weights of binaural and symmetric losses to be 1. For more complex scenarios (Sec. 4.5), we set the weights λ as 3 to avoid the geometric loss from dominating.

IID cues. We describe our implementation of predicting sound on the left or right using IID cues in detail here: we first compute the magnitude spectrogram $|\mathbf{A}|$ from the binaural waveform $\mathbf{a}$ and sum the magnitude over the frequency axis. Next, we calculate the log ratio between the left and right channels for each time frame. After this, we take the sign of log ratios and convert them into either +1 or -1. We sum over the votes and take the sign of it for final outputs.

Dataset. Due to the fact that SoundSpaces 2.0 [15] does not support material configuration for HM3D [76] at the current time, we obtain binaural RIRs with different reverberation levels by scaling the indirect RIRs and add them up with direct RIRs. We render binaural sounds with random audio samples as augmentation during training.