DeepMMAudio: Leveraging Depth Features in Multimodal Video-to-Audio Synthesis

1st Hugo Bachér *bacher@kth.se*
2nd Pontus Berglund *pobergl@kth.se*
3rd Rikard Löwbeer *lowbeer@kth.se*
4th Samer Jameel *sbsja@kth.se*

KTH Royal Institute of Technology
Stockholm, Sweden

***Abstract*—Recent advances in multimodal diffusion models have enabled high-fidelity audio generation from silent video. Yet, state-of-the-art systems such as MMAudio lack explicit depth awareness, often producing sounds that contradict spatial cues in the visual scene. This work introduces DeepMMAudio, an extension of MMAudio that incorporates monocular depth features to improve audio-visual alignment. Depth maps are extracted using MiDaS DPT-Hybrid and encoded with CLIP, then concatenated with visual embeddings and projected back into the original latent space for joint training. Using the VGGSound dataset and AV-Benchmark metrics, including Fréchet Distance, ImageBind similarity, and DeSync, we compare the baseline and depth-augmented models. Although Fréchet Distance degrades slightly, the depth-enhanced model achieves improved semantic alignment and temporal synchronization, suggesting stronger spatial awareness. Qualitative results further indicate fewer spatial inconsistencies, such as the correction of erroneous approaches or retreats by moving objects. Although training time increases by 35% and convergence is slower, the results indicate that integrating depth cues can enhance cross-modal coherence in video-to-audio synthesis. Future work should explore optimized depth encoders, higher-quality datasets, and fully converged training to assess the complete potential of depth-aware audio generation.**

I. Introduction

The current state of the art in high-fidelity, synchronized audio generation from silent video is MMAudio [1], which employs a joint training framework based on multimodal Diffusion Transformers (DiTs) [2]. Replacing traditional convolutional U-Net architectures [3] with transformer-based models operating on latent image patches, MMAudio improves scalability and performance. By incorporating both video and text inputs with Synchformer [4], the model effectively generates audio that is semantically and temporally aligned with visual cues.

Despite these advancements, MMAudio processes visual inputs primarily as two-dimensional feature maps, ignoring the acoustic properties in three-dimensional space. In reality, depth, distance, and object movement fundamentally influence auditory perception; for example, a car approaching the camera sounds distinct from one moving parallel to it. Because MMAudio lacks explicit depth awareness, it can generate audio that contradicts the visual spatial dynamics. This study was directly motivated by such an artifact, in which a horse running parallel to the camera had sound generated as if it were running towards the camera.

To bridge the gap between visual distance and realistic audio, this paper investigates how to improve audio generation by explicitly incorporating depth features. Monocular Depth Estimation (MiDaS) [5] with the Dense Prediction Transformer (DPT) [6] is employed to extract inverse depth maps, which are encoded alongside visual and text features via Contrastive Language-Image Pre-Training (CLIP) [7]. Following training on the audio-visual VGGSound dataset [8] and other annotated audio datasets, evaluating the performance using ImageBind [9] for semantic alignment across modalities (including depth) and the Patchout Fast Spectrogram Transformer (PaSST) [10] for acoustic quality assessment.

Thanks to Carl Thomé, Parham Fazelzadeh Hashemi, and Matt Destephe at Epidemic Sound, Stockholm, Sweden.

II. Related Work

Recent work by StereoSync [11] incorporates depth maps into video-to-audio generation to facilitate mono-to-stereo conversion. By employing RollingDepth [12] to extract depth features and EVA-CLIP [13] for encoding, the model achieves superior spatial and temporal alignment. Although our primary focus is not spatial audio separation, these findings validate that explicit depth cues contribute significantly to the perceived realism of synthesized audio.

III. Problem Description

The current state-of-the-art video-to-audio generation model, MMAudio, lacks explicitly modeled depth-perception features [1]. As a result, it struggles to consistently generate audio that matches the visual depth, size, and trajectory of objects, leading to some generated sounds with unrealistic directionality and motion.

IV. Methodology

The implementation and evaluation of new depth map features in MMAudio consisted of data preparation, model modification, training setup, and evaluation.

A. Data preparation

The model was trained using five different datasets: VGGSound, BBCSound, AudioSet_SL, and Clotho. VGGSound, the primary audio-visual dataset, contains short, annotated

videos with corresponding audio [8]. The remaining three datasets are annotated audio-only collections [14]–[16].

For model training, all five datasets were used to expose the model to a variety of different audio-visual and audio-only samples. However, for testing, only VGGSound was used because it is the only dataset with video-to-sound pairings, which are used to compare the generated audio against the original ground truth.

A cleaning process was applied to the datasets before training. During the gathering and unpacking of the archives, several issues were found. For the VGGSound dataset, one of the 20 archives was corrupted. Upon feature extraction of the remaining files, over 500 were found to contain no audio. Similarly, some of the files in the AudioSet_SL dataset were also corrupted. All of the identified corrupted or silent files were excluded from both the training and testing sets.

Depth maps were extracted from the VGGSound videos using the MiDaS DPT-Hybrid model, yielding a single-channel, monochrome inverse-depth map. To ensure compatibility with subsequent network architectures that expect a 3-channel (RGB) input, this single-channel output was stacked across all 3 channels, resulting in a 3-channel video.

B. Model Modification

The model's projection layer from the CLIP embeddings of the visual features $\mathbf{F_v}$ = (1024d, 8fps) was modified, as seen in Figure 1, by concatenating the CLIP embeddings of the depth maps generated from MiDaS $\mathbf{F_d}$ = (1024d, 8fps), after the CLIP embeddings of the visual features, resulting in $\mathbf{F_c}$ = (2048d, 8fps). The decision to use CLIP embeddings for the generated depth maps was driven by MMAudio's existing use of them for visual feature embedding.

The concatenated feature is then projected from $\mathbf{F_c}$ = (2048d, 8fps) back to the original vector size of (1024d, 8fps) using a single fully connected linear layer **Linear**(2048d, 1024d). This enables the network to consistently combine information from the standard RGB video and depth maps without altering the rest of the architecture.

C. Training setup

Google Cloud Platform virtual machines equipped with 1 NVIDIA A100 80GB GPU, 16-core Intel Cascade Lake processors, and 170GB of RAM each were used to run all experiments.

The model was trained with a batch size of 448, for 300,000 iterations on the gathered datasets, once with and once without depth features. To ensure a clear comparison of the base model and the contribution of depth features.

D. Evaluation

Model performance was evaluated using the AV-Benchmark framework and the following metrics: Fréchet Distance (FD), ImageBind, and DeSync. These metrics assess the fidelity of the generated audio. The Fréchet Distance was computed on PaSST and PANNs embeddings to determine how well the generated audio matched the ground truth audio. The PaSST embedding was favored since it performed better on the HEAR benchmark on the ESC50 dataset [17]. Cross-modal semantic alignment was evaluated using the ImageBind similarity score. Finally, DeSync quantifies the temporal synchronization between the audio and video visual data.

In addition to the quantitative metric scores, a qualitative evaluation was conducted by reviewing the generated audio on a subset of the data. This manual inspection focused on assessing audio quality and verifying its semantic and temporal coherence with the video visuals.

V. Results

The original results from the MMAudio paper, along with those of the retrained base and depth models, are presented in Table I. The model with added depth features has 3,000,000 more parameters than the base model. Both retrained models performed better than the original MMAudio paper on Fréchet Distance but worse on ImageBind and DeSync scores. When comparing only the retrained models, the depth model achieves better ImageBind and DeSync scores while receiving slightly worse Fréchet Distance.

TABLE I: Video-to-audio results on the VGGSound test dataset with the Small 44.1kHz models.

Model	Params	FD_{PaSST} ↓	ImageBind ↑	DeSync ↓
$MMAudio_{Original}$	157M	65.25	**32.27**	**0.444**
$MMAudio_{Retrain}$	157M	**58.03**	31.82	0.477
DeepMMAudio	160M	59.28	32.22	0.461

Qualitative analysis of a subset of generated samples reveals that the model maintains high acoustic fidelity and perceptual plausibility. While general performance remains comparable to the baseline, specific instances demonstrate the benefit of depth integration. For example, in the video of a galloping horse, which initially motivated this study, the generated audio exhibits improved spatial accuracy. Unlike the baseline, the modified model correctly aligns sound intensity with the subject's distance, thereby eliminating the auditory artifact in which the horse appears to approach the camera when moving parallel to it.

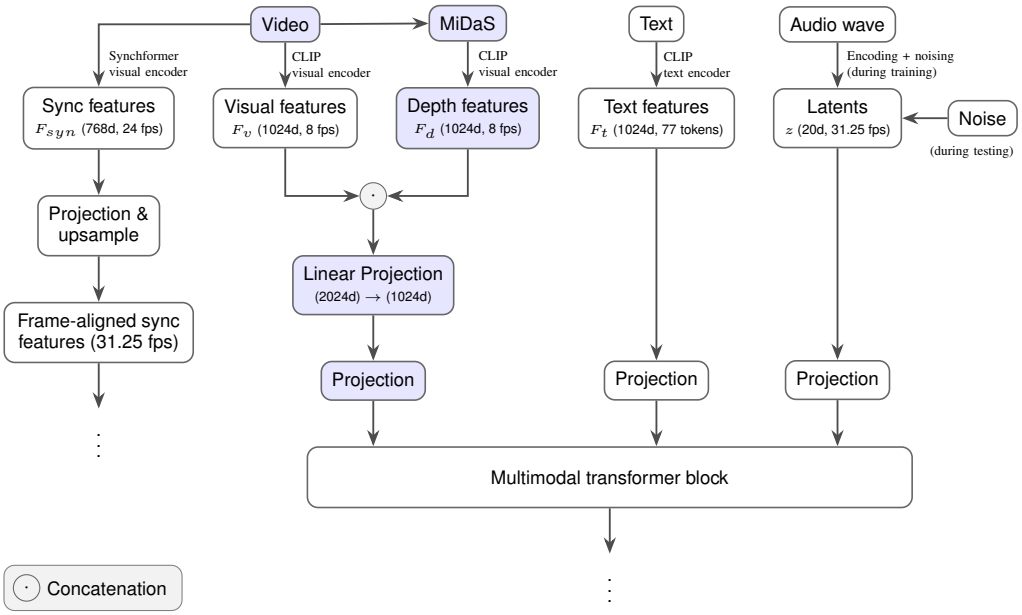
The depth model took 35% longer to train, requiring 156 hours, 40 hours more than the baseline architecture's 116 hours. This excludes the additional time required for feature extraction, which also took significantly longer for extracting the reverse depth maps.

The validation loss trajectories for both models are found in Figures 2 and 3. These plots show distinct convergence behaviors. The baseline model converged relatively quickly, reaching a plateau around 250,000 iterations. In contrast, the depth model had a steeper learning curve, with the loss continuing to decrease even when training was stopped at 300,000 iterations.

VI. Discussion

Verification of the baseline first establishes the integrity of the replication. This is followed by an interpretation of the

Fig. 1: Overview of the DeepMMAudio flow-prediction network with depth features modification. The blue regions indicate the modified components; the remainder is identical to the original MMAudio model [1].



divergence in convergence behaviors and metric scores, effectively highlighting the trade-off between geometric awareness and acoustic precision, and by an outline of the study's validity and future work.

A. Baseline Verification

Comparing the metric scores in Table I, the retrained MMAudio model achieves scores similar to those reported in the original paper, indicating that the retraining of the baseline model was successful. A possible reason for the differences in scores, and specifically for better results for the baseline model on the Fréchet Distance and worse results on ImageBind and DeSync, is the absence of the AudioCaps dataset in the retrain version. It could also be due to the different batch size used in the original paper (512) versus ours (448).

B. Impact of Depth Features

Among the newly trained models, the depth model achieves higher scores on both ImageBind and DeSync, indicating that depth features enhance the model's ability to interpret visual input. These improvements likely stem from the model's increased awareness of scene geometry, which helps contextualize motion and spatial structure—factors that are important for synchronized audiovisual generation. This aligns with results from the related work presented in Section II.

Contrarily, the Fréchet Distance for both the PaSST and PANNs embeddings shows a slight decline. One possible explanation is that integrating an additional depth modality may lead the model to allocate more of its representational capacity toward spatial and geometric structure, leaving slightly less emphasis on the features needed for fine-grained acoustic classification and spectrogram-level fidelity. In this view, adding depth could introduce a mild trade-off, where the model balances more information sources and, as a result, becomes somewhat less aligned with the ground-truth audio patterns emphasized by the Fréchet Distance.

C. Training Convergence

Although both architectures were trained for the same number of iterations, as shown in Figures 2 and 3, their convergence behaviors diverged significantly. The baseline's early saturation suggests it reached capacity around 250,000 iterations, at which point early stopping could have been employed. In contrast, the depth model's sustained improvement suggests it was navigating a more complex optimization landscape, requiring longer training to integrate and fully leverage the additional depth features. While time and budget constraints prevented further training, the depth model's consistently lower loss, even before convergence, demonstrates that the added depth features yielded better predictive potential.

D. Validity

The validity of our results is supported by the controlled experimental setup, in which the baseline MMAudio model was retrained under identical conditions to those used for the depth model. However, as noted in the discussion of training convergence, the depth model had not fully converged by the end of the 300,000 iterations. This suggests that the reported results likely underestimate the architecture's full potential when depth maps are used.

E. Future Work

Several factors affect the results of this study. The older MiDaS DPT depth model used to generate inverse depth maps,

and the use of CLIP for depth embeddings, are examples that could be examined in future work to identify the best available combination. Finding better depth map models and other ways to encode the depth map information. It would also be interesting to see the full effect of the depth model by training it until the validation loss converges.

Future work could also investigate how the depth features propagate through the model, whether they affect the final result, and whether the model learns to ignore them. Additionally, improving the video-to-audio dataset is necessary. Examining VGGSound videos revealed many instances of low quality and questionable content. Among the videos examined, the computer-generated annotations were also sometimes irrelevant.

F. Summary

In summary, adding depth features improves scores on ImageBind and DeSync, indicating an enhanced ability to reason about cross-modal correspondence. However, the improvement is minimal, and human perception of the generated audio remains essentially unchanged, while training time increases significantly. The question of whether depth features provide a practical benefit relative to their computational cost remains open to further exploration.

VII. Opposing Group

Our opponent group does not work directly on the same thing as we do. We are working with videos, audio, and text in a large multimodal model, while they are working with training an LLM on text data to generate music playlists. Therefore, we have not been able to apply each other's findings directly. However, during the project, both groups have been working at the Epidemic Sound office, and we have mainly discussed dataset access and model evaluation with the other group.

Gathering data and getting access to good datasets have been problematic for both groups. We have public datasets, but the data is missing, corrupted, or cannot be downloaded, while the other group had to request data, which would take longer to wait for than to make a dataset themselves.

Evaluation of the final models is also different. Our model can be tested by generating audio, qualitatively evaluating the outputs by watching and listening to them, and assessing whether the audio is a good fit for the visuals. We also have a few other evaluation models that give output scores, which we can use for a quantitative evaluation. The opposing group currently does not have any way to do a quantitative evaluation, their only way is to visually look at the playlist and determine if they are good or not.

References

[1] H. K. Cheng, M. Ishii, A. Hayakawa, T. Shibuya, A. Schwing, and Y. Mitsufuji, "MMAudio: Taming Multimodal Joint Training for High-Quality Video-to-Audio Synthesis," 2025. [Online]. Available: https://arxiv.org/abs/2412.15322

[2] W. Peebles and S. Xie, "Scalable Diffusion Models with Transformers," 2023. [Online]. Available: https://arxiv.org/abs/2212.09748

[3] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," 2015. [Online]. Available: https://arxiv.org/abs/1505.04597

[4] V. Iashin, W. Xie, E. Rahtu, and A. Zisserman, "Synchformer: Efficient Synchronization from Sparse Cues," 2024. [Online]. Available: https://arxiv.org/abs/2401.16423

[5] R. Ranftl, K. Lasinger, D. Hafner, K. Schindler, and V. Koltun, "Towards Robust Monocular Depth Estimation: Mixing Datasets for Zero-shot Cross-dataset Transfer," 2020. [Online]. Available: https://arxiv.org/abs/1907.01341

[6] R. Ranftl, A. Bochkovskiy, and V. Koltun, "Vision transformers for dense prediction," 2021. [Online]. Available: https://arxiv.org/abs/2103.13413

[7] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," 2021. [Online]. Available: https://arxiv.org/abs/2103.00020

[8] H. Chen, W. Xie, A. Vedaldi, and A. Zisserman, "VGGSound: A Large-scale Audio-Visual Dataset," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2020. [Online]. Available: https://arxiv.org/abs/2004.14368

[9] R. Girdhar, A. El-Nouby, Z. Liu, M. Singh, K. V. Alwala, A. Joulin, and I. Misra, "Imagebind: One embedding space to bind them all," 2023. [Online]. Available: https://arxiv.org/abs/2305.05665

[10] K. Koutini, J. Schlüter, H. Eghbal-zadeh, and G. Widmer, "Efficient training of audio transformers with patchout," in *Interspeech 2022*. ISCA, Sep. 2022. [Online]. Available: https://arxiv.org/abs/2110.05069

[11] C. Marinoni, R. F. Gramaccioni, K. Shimada, T. Shibuya, Y. Mitsufuji, and D. Comminiello, "Stereosync: Spatially-aware stereo audio generation from video," 2025. [Online]. Available: https://arxiv.org/abs/2510.05828

[12] B. Ke, D. Narnhofer, S. Huang, L. Ke, T. Peters, K. Fragkiadaki, A. Obukhov, and K. Schindler, "Video depth without video models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. [Online]. Available: https://arxiv.org/abs/2411.19189

[13] Q. Sun, Y. Fang, L. Wu, X. Wang, and Y. Cao, "Eva-clip: Improved training techniques for clip at scale," 2023. [Online]. Available: https://arxiv.org/abs/2303.15389

[14] British Broadcasting Corporation, "BBCSound," 2025. [Online]. Available: https://sound-effects.bbcrewind.co.uk/

[15] J. F. Gemmeke, D. P. W. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter, "Audio set: An ontology and human-labeled dataset for audio events," in *Proc. IEEE ICASSP 2017*, New Orleans, LA, 2017, pp. 776–780. [Online]. Available: http://doi.org/10.1109/ICASSP.2017.7952261

[16] K. Drossos, S. Lipping, and T. Virtanen, "Clotho: An audio captioning dataset," 2019. [Online]. Available: https://arxiv.org/abs/1910.09387

[17] J. Turian, J. Shier, H. R. Khan, B. Raj, B. W. Schuller, C. J. Steinmetz, C. Malloy, G. Tzanetakis, G. Velarde, K. McNally, M. Henry, N. Pinto, C. Noufi, C. Clough, D. Herremans, E. Fonseca, J. Engel, J. Salamon, P. Esling, P. Manocha, S. Watanabe, Z. Jin, and Y. Bisk, "Hear: Holistic evaluation of audio representations," 2022. [Online]. Available: https://arxiv.org/abs/2203.03022

Appendix A
Training Graphs

Figures 2 and 3 shows the validation loss throughout the training over 300,000 iterations. The validation loss is calculated on the VGGSound dataset's validation split.

Appendix B
Full results

The scores for Fréchet Distance and KL-Divergence on the PaSST, PANNs, and VGG embeddings, with ImageBind and DeSync, can be found in Table II for model comparison. Table IV shows the project timeline and Table IV the week by week diary.

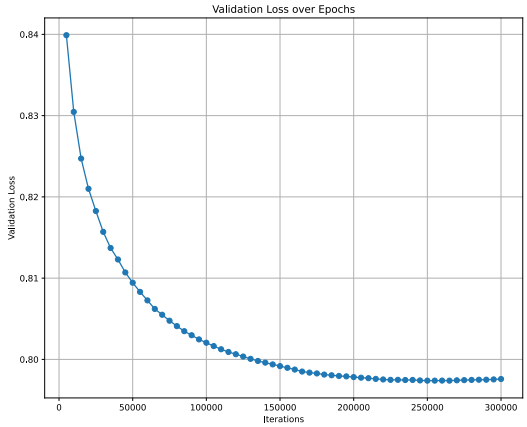


Fig. 2: Validation loss over iterations for the training of the retrained baseline MMAudio model.

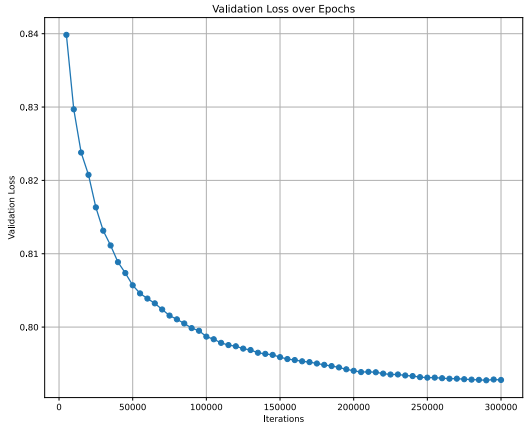


Fig. 3: Validation loss over iterations for the training of the new DeepMMAudio depth model.

TABLE II: All measured Video-to-audio results on the VGGSound test set with the Small 44.1kHz models.

Model	Params	FD_{PaSST} ↓	FD_{PANNs} ↓	FD_{VGG} ↓	KL_{PaSST} ↓	KL_{PANNs} ↓	ImageBind ↑	DeSync ↓
$MMAudio_{Original}$	157M	65.25	5.55	1.66	1.44	1.67	**32.27**	**0.444**
$MMAudio_{Retrain}$	157M	**58.03**	**4.59**	**1.30**	**1.43**	**1.66**	31.83	0.477
DeepMMAudio	160M	59.28	5.03	1.51	**1.43**	**1.66**	32.22	0.461

TABLE III: Project Timeline and Key Milestones: Video-to-SFX Generation

	Focus Area	**Key Activities & Milestones**	**Collaboration**
Phase 1: Inception & Setup (Sept 9 – Sept 29)	Scoping, Literature Review, and Infrastructure	• Defined project scope to add depth and optical flow understanding to existing models. • Conducted literature review and tested initial tools, including RAFT and MiDaS, for feature extraction. • Established Google Cloud Platform (GCP) environment and successfully installed MMAudio on an Nvidia L4 instance. • Verified feasibility by running demo training and testing audio generation on reversed videos.	The work was split so that each group member researched different aspects, including how to extract depth features, how MMAudio works, and other relevant topics. Each group member also tried to set up environments to run the models, either locally or on GCP.
Phase 2: Methodology & Architecture (Oct 6 – Oct 27)	Model Design and Pipeline Modification	• Designed architecture: Video $\rightarrow$ MiDaS $\rightarrow$ CLIP $\rightarrow$ Concatenation with CLIP visual output. • Modified the forward pass and model flow to accept extra depth features via concatenation. • Confirmed MiDaS implementation and output sizes ($3 \times 384 \times 384$) match normal video inputs. • Enabled pulling updates from the repository to the GCP VM using Skypilot for syncing the working directory.	• Pontus and Rikard modified the model, changing the forward pass and flow to incorporate depth features. • Hugo and Samer incorporated the MiDaS model in the pipeline. • Each group member was granted access to Epidemic Sound's GCP and configured it so that all of us could access the machines.
Phase 3: Implementation & Training (Nov 3 – Nov 17)	Data Processing and Model Training	• Downloaded the datasets, and moved data to a GCS bucket. • Developed and debugged training scripts, removing corrupted data and fixing configuration errors. • Initiated baseline and depth training scripts and set up batch prediction pipelines. • Executed feature extraction scripts on the prepared data.	• All members browsed the internet to locate ZIP files to download. • Feature extraction and model training for both the base model and the depth model were distributed across different machines, with each group member responsible for specific components.
Phase 4: Evaluation & Results (Nov 24 – Dec 8)	Benchmarking, Comparison, and Reporting	• Completed full feature extraction of depth features and trained the depth and base model for 300,000 steps. • Performed batch predictions and evaluations on both baseline and depth checkpoints. • Consolidated results showing the new model ("DeepMMAudio") outperformed the retrained baseline in some aspects. • Documented final metrics.	• Hugo and Samer wrote the report and made the presentation. • Pontus and Rikard each used one machine to run the evaluation scripts. One machine evaluated the base model, and the other evaluated the depth model.

TABLE IV: Project Work Diary and Timeline

Week of	**Activities and Technical Milestones**
Sep 9	Project initialization and scope definition. Began literature review and established the goal of adding depth and optical flow understanding to the existing MMAudio model.
Sep 15 and Sep 22	Verified MMAudio inference locally. Successfully tested feature extraction using RAFT and MiDaS on custom videos. Began setup of Google Cloud Platform (GCP) environment and Deep Learning VMs.
Sep 29	Configured GCP instance (Nvidia L4). Conducted initial experiments generating audio for reversed videos to test temporal sensitivity. Formulated the primary Research Question regarding the value of depth maps in video-to-audio synthesis.
Oct 6	Designed the architecture modification plan: Extract depth via MiDaS, encode via CLIP, concatenate with visual embeddings, and project using a linear layer. Analyzed data representations for the new depth features.
Oct 20	Implemented the modified forward pass to accept extra features. Successfully tested the flow with concatenation on random depth features. Delegated core tasks: Training (Pontus), Evaluation (Rikard), and Feature Extraction (Hugo).
Oct 27	Confirmed MiDaS implementation aligns with video dimensions ($3 \times 384 \times 384$). Solved repository synchronization issues on GCP. Added toggles for summation vs. concatenation of features.
Nov 3	Debugging phase. Attempted full test training loops with fewer epochs. Encountered and resolved scripting errors preventing full training completion.
Nov 10	Large-scale data acquisition. Downloaded VGGSound dataset, fixed configuration errors, and removed corrupted data/files from the training set.
Nov 17	Data preprocessing pipeline. Subsampled VGGSound and moved data to GCS buckets. Started full feature extraction (Depth) and initiated the training scripts for both the Baseline and Contribution (DeepMMAudio) models.
Nov 24	Training and Validation. Batch predicted MP4s from baseline and contribution checkpoints. Performed full feature extraction of depth features. Initial evaluation results indicated the new model outperformed the retrained baseline.
Dec 1	Final Training and Analysis. Models reached 300,000 training steps. Collected final evaluation scores.
Dec 8	Project Closure. Code cleanup, organization of example predictions, and preparation of the final report and presentation. Presented at the Epidemic Sound office.