Video-to-Music Generation for Film Production: A Dataset and Framework

Haven Kim[1*] **Leduo Chen**[1*] **Bill Wang**[1] **Hao-Wen Dong**[2] **Julian McAuley**[1]
[1]University of California San Diego
[2] University of Michigan

Abstract

Despite growing interest in video-to-music generation systems, their application in film production remains limited, primarily due to the lack of large-scale datasets containing aligned pairs of movie clips and soundtracks. Although prior work has attempted to construct such a dataset [12], this comprises only 36.5 hours of data, which is insufficient for training robust models. In this paper, we present **Open Screen Soundtrack Library Version 2**, a novel dataset comprising pairs of video clips from films and their corresponding soundtracks, curated with a novel methodology that automatically identifies and extracts soundtrack segments from video clips. The dataset consists of 552.70 hours and 76,408 video clips sourced from both public domain movies as well as commercial ones from a publicly available dataset [1]. Our objective evaluation results show the usefulness of our dataset for building a soundtrack generation model for film production.

1 Introduction

Video-to-music generation systems have gained increasing attention in both the audio and symbolic domains. However, their application in film production remains limited, as most prior work has focused on use cases such as music videos [29, 23, 16], advertisements [24], or user-generated content [30]. While some studies have utilized trailer data for video-to-music generation [24], trailer music possesses unique characteristics that are significantly different from main movie soundtracks.

To date, we have identified only two studies that specifically address video-to-music generation for film production—one in the symbolic domain [27] and the other in the audio domain [12]. Despite the success of the former, the symbolic-domain approach requires expert knowledge to convert symbolic outputs into usable soundtracks, making it impractical for film producers. The latter leveraged video information to generate soundtracks directly; however, it relied heavily on textual information due to the limited size of its dataset, which is only about 36.5 hours.

To overcome these limitations, we constructed a large-scale dataset consisting of aligned movie clips and corresponding soundtracks, totaling 552.70 hours and 76,408 video clips. Using a novel methodology, we identified and extracted soundtrack segments from video content. As demonstrated by our experimental results, this dataset enables successful training video-to-music generation models tailored specifically for film production applications.

2 Related Work

Music-Video Datasets A number of datasets have been developed for video-to-music generation tasks in the audio domain. These datasets span various types of video content, including music videos [7, 29, 16, 17, 24], musical performance recordings [15], and user-generated content [7]. The

[*]Equal contribution.

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: AI for Music.

Table 1: Comparison of video-music datasets available as of August 2025.

Dataset	Self-Hosted	Video Content	Length (Hours)
HIMV-200K [7]	✗	Music Video, User-Generated Video	-
URMP [15]	✗	Music Performance	33.5
TikTok [28]	✗	Dance Video	1.5
SymMV [29]	✗	Music Video	76.5
MuVi-Sync [8]	✗	Music Video	-
BGM909 [17]	✗	Music Video	-
VidMuse [24]	✗	Music Video, Advertisements, Trailer	18k
OSSL [12]	✓	Films	36.5
OSSL-v2 (ours)	✓ and ✗ (partial)	Films	522.7

dataset most closely related to our work is the Open Screen Soundtrack Library, which comprises music–movie clip pairs sourced from public domain films [12]. In contrast, our dataset is significantly larger in scale and draws from both public domain and commercial films.

Video-Conditioned Music Generation Several studies have explored music generation conditioned on video input. One of the early and influential works in this area employed human pose features extracted by pre-trained models to generate music for video clips depicting individuals playing musical instruments [6]. More recent approaches utilize either handcrafted features [4] or embeddings from pre-trained video encoders [29, 24, 23, 30, 12] to build video-to-music generation systems. Our method illustrated in this paper adopts this latter strategy, leveraging learned video representations to guide the music generation process.

3 Dataset Construction

Our music-movie clip dataset, Open Screen Soundtrack Library Version 2 (OSSL-v2) is constructed from two types of movie data. The former comprises 1,886 public domain films downloaded from YouTube [2], and the latter is derived from a publicly available movie dataset, the Condensed Movies Dataset [1]. Our dataset construction process consists of two main components: source separation and event detection.

In the first step, we apply an open-source separation model [22] in order to extract music from each movie clip's audio track. This model offers a high-quality processing option that requires three times longer than the default option. We select the high-quality option for audio source separation because our objective is to create a music-movie clip dataset with the highest possible quality.

In the second step, we employ an event detection model to estimate the probability distribution of event types in source-separated musical tracks. This step is essential because the source separation model, even when using a high-quality option, is not perfect; source-separated music often contained non-musical events. To address this, we use an open-source automatic event detection model [13], from which we identify 157 out of 527 categories as musical events (e.g., "trance music"). We define the music probability as the sum of probabilities for the 157 musical events, and the non-music probability as the sum of probabilities for the 370 non-musical events, to source-separated music. We extract segments where the music probability exceeds the non-music probability for at least 10 consecutive seconds. However, this fails to filter out cases where both musical and non-musical events are prominent (e.g., music probability of 0.8 and non-music probability of 0.7). Therefore, we apply an additional filter to exclude cases where the non-music probability exceeded 0.05.

This approach yields a total of 76,408 video clips with source-separated soundtracks (processed using the high-quality option) averaging 26.04 seconds in length, along with rich metadata such as genres, release year, and title. Detailed dataset statistics are presented in Table 2.

[2]This part is self-hosted, meaning that readers do not need to undergo a separate download process such as web scraping.

Table 2: Statistics of the OSSL-v2 Dataset. To obtain commercial movie clips, we used a list of YouTube IDs from the Condensed Movies Dataset [1] and scraped the corresponding clips from the web.

	Public Domain	Commercial [1]	Total
Number of Clips	35,705	40,703	76,408
Number of Unique Films	1,886	2,633	4,519
Average Length (seconds)	28.77	23.65	26.04
Total Length(hours)	285.31	267.39	552.70

4 Experimental Setups

4.1 Comparative Models

The goal of our experiment is to validate the potential of our large-scale film-soundtrack dataset, OSSL-v2, for the video-to-music generation task. We use MMAudio [3], a state-of-the-art model in video-foley sound generation, as our baseline architecture. We compare this model with three different models that are built upon this architecture.

Pretrained MMAudio-S-16kHz (Baseline): We employ the pretrained MMAudio model, which is trained on VGGSound [2] (approximately 500 hours of audio-visual data), AudioCaps [11] (approximately 128 hours), and WavCaps [19] (approximately 7,600 hours). While MMAudio was not specifically trained for music generation, its audio-text training data contains substantial music content, enabling the model to capture associations between visual inputs and musical features, thus retaining the capacity for video-to-music generation.

Fine-tuning without Text Modality: Since our OSSL-v2 dataset does not contain paired text prompts. We leverage MMAudio's capability to handle missing modalities by setting all the text features as learnable empty tokens during fine-tuning, allowing the model to focus primarily on the video and music modalities.

Training from Scratch with Visually-Grounded Text Features: We address the missing text modality by using CLAP [26]-derived audio embeddings to simulate corresponding text embeddings, following approaches validated in prior work [23]. This model is trained from scratch using these visually-grounded text features. Since MMAudio has demonstrated that the text modality serves as an anchor to connect multiple modalities and enhance overall model performance, we adopt this method to address the absence of text modality in our dataset.

Fine-tuning with Visually-Grounded Text Features: We combine pretrained weights with visually-grounded text features by adding a mapping layer that transforms CLAP features to match CLIP text feature dimensions used in MMAudio. This preserves learned representations while adapting text modality handling to our dataset.

4.2 Evaluation Metrics

Following prior works [21, 12], we evaluate the quality of generated music using a suite of objective metrics.

To assess distributional properties, we extract CLAP [26] embeddings from generated samples and a 5K high-quality commercial soundtracks that are not included in the OSSL-v2 dataset, and compute Fréchet Audio Distance (FAD)[10]. Following [20], we report Precision, Recall, Density, and Coverage based on CLAP embeddings. Precision measures the proportion of generated samples close to real data (fidelity), while Recall reflects how well real samples are recovered by generation (diversity), both using k-NN overlap. Density quantifies the concentration of generated samples around real data, and Coverage estimates how much of the real data manifold is covered.

For paired fidelity, we report CLAP Similarity and KL Divergence. The former is cosine similarity between CLAP embeddings; the latter compares PaSST [14] label distributions to capture semantic differences.

Finally, we assess temporal alignment in terms of dynamics by using our novel metric, Dynamics Distance (DD), to capture smooth variations in loudness and energy of music that are often overlooked

Table 3: Comprehensive evaluation results. See Section 4.2 for the choice of metrics.

Method	FAD ↓	CLAP Sim. ↑	KL Div. ↓	DD ↓	Precision ↑	Recall ↑	Density ↑	Coverage ↑
Pretrained MMAudio	80.90	44.88	2.19 ± 1.49	1.38 ± 0.23	28.63	0.73	3.02	3.23
Fine-tune w/o Text	51.25	67.36	0.68 ± 1.01	1.10 ± 0.43	**37.30**	4.50	5.26	**9.23**
Training from Scratch	61.72	66.55	**0.61 ± 0.87**	1.06 ± 0.51	34.13	1.91	4.82	5.60
Fine-tune w/ Text	**50.80**	**68.36**	0.65 ± 0.98	**1.03 ± 0.49**	35.63	**10.73**	**5.47**	9.20

by existing evaluation methods. Dynamics Distance is calculated as follows. We extract energy contours using short-time Fourier transform (STFT), apply Savitzky–Golay smoothing for temporal coherence, as suggested in [25] and compute the root-mean-square (RMS) error between normalized dynamics curves to quantify alignment quality.

4.3 Details

We partition the OSSL-v2 dataset into training, validation, and test sets with an 8:1:1 ratio, ensuring uniform distribution of different film genres across all subsets. This results in 138,783 video clips for training data, 17,352 clips for validation data, and 17,347 clips for final evaluation as our test set, where the clips are 8 seconds long. We provide more training details in Appendix B.

5 Results and analysis

Table 3 shows the performance of the different methods on the test set.

The experimental results demonstrate that all methods trained with the OSSL-v2 significantly outperform the pretrained MMAudio baseline across content quality metrics, indicating that domain-specific training enables models to generate music content better aligned with film soundtrack characteristics.

Fine-tuning with visually-grounded text features performs optimally on most metrics, particularly excelling in Recall where it substantially surpasses other approaches, suggesting that text modality serves as an effective multimodal anchor for enhancing generation diversity. In contrast, while fine-tuning without text demonstrates strong performance in Precision and Coverage, its lower Recall reflects that without text guidance, the model tends toward more conservative generation with limited coverage scope.

From a training strategy perspective, fine-tuning methods consistently outperform training from scratch, indicating that pretrained weights contain valuable general audio-visual associative knowledge for film scoring tasks, and that additional text-audio datasets help models better capture semantic relationships. However, all methods show limited improvement in Dynamics Distance (DD) metrics, revealing fundamental challenges in temporal dynamic alignment with current approaches.

6 Conclusion

In this paper, we introduced the OSSL-v2 dataset, a large-scale dataset of paired movie clips and their corresponding soundtracks, constructed using a novel methodology that automatically identifies and extracts soundtrack segments from video clips. We believe this dataset will facilitate the training of video-to-music generation systems with applications in film production. Although our focus was specifically on film clips, we also want to emphasize the broad generalizability of our methodology, which is also applicable to other types of video content, such as vlogs, highlighting its potential for constructing diverse music–video datasets.

In addition, we demonstrate that our dataset enhanced video-to-music generation capabilities, with all trained models substantially outperforming general-purpose baselines. However, persistent challenges in temporal dynamics alignment reveal that current synchronization mechanisms designed for discrete audio events are fundamentally mismatched with music generation's continuous temporal requirements, and we regard this as an important direction for future exploration.

References

[1] Max Bain, Arsha Nagrani, Andrew Brown, and Andrew Zisserman. Condensed movies: Story based retrieval with contextual embeddings. *arXiv preprint arXiv:2005.04208*, 2020.

[2] Honglie Chen, Weidi Xie, Andrea Vedaldi, and Andrew Zisserman. *Vggsound: A Large-Scale Audio-Visual Dataset*, volume 14, page 15. IEEE, 5 2020.

[3] Ho Kei Cheng, Masato Ishii, Akio Hayakawa, Takashi Shibuya, Alexander Schwing, and Yuki Mitsufuji. Mmaudio: Taming multimodal joint training for high-quality video-to-audio synthesis. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 28901–28911, 2025.

[4] Shangzhe Di, Zeren Jiang, Si Liu, Zhaokai Wang, Leyan Zhu, Zexin He, Hongming Liu, and Shuicheng Yan. Video background music generation with controllable music transformer. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 2037–2045, 2021.

[5] P Diederik, Jimmy Kingma, and Ba. Adam: A method for stochastic optimization. *ICLR*, 18, 2015.

[6] Chuang Gan, Deng Huang, Peihao Chen, Joshua B Tenenbaum, and Antonio Torralba. Foley music: Learning to generate music from videos. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI 16*, pages 758–775. Springer, 2020.

[7] Sungeun Hong, Woobin Im, and Hyun S. Yang. Content-based video-music retrieval using soft intra-modal structure constraint, 2017.

[8] Jaeyong Kang, Soujanya Poria, and Dorien Herremans. Video2music: Suitable music generation from videos using an affective multimodal transformer model. *Expert Systems with Applications*, 249:123640, September 2024.

[9] Tero Karras, Miika Aittala, Jaakko Lehtinen, Janne Hellsten, Timo Aila, and Samuli Laine. *Analyzing and improving the training dynamics of diffusion models*, volume 15, page 19. 2024.

[10] Kevin Kilgour, Mauricio Zuluaga, Dominik Roblek, and Matthew Sharifi. Fr\'echet audio distance: A metric for evaluating music enhancement algorithms. *arXiv preprint arXiv:1812.08466*, 2018.

[11] Chris Dongjoo Kim, Byeongchang Kim, Hyunmin Lee, and Gunhee Kim. *Audiocaps: Generating captions for audios in the wild*, page 15. 2019.

[12] Haven Kim, Zachary Novack, Weihan Xu, Julian McAuley, and Hao-Wen Dong. Video-guided text-to-music generation using public domain movie collections. *arXiv preprint arXiv:2506.12573*, 2025.

[13] Qiuqiang Kong, Yin Cao, Turab Iqbal, Yuxuan Wang, Wenwu Wang, and Mark D Plumbley. Panns: Large-scale pretrained audio neural networks for audio pattern recognition. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:2880–2894, 2020.

[14] Khaled Koutini, Jan Schlüter, Hamid Eghbal-Zadeh, and Gerhard Widmer. Efficient training of audio transformers with patchout. *arXiv preprint arXiv:2110.05069*, 2021.

[15] Bochen Li, Xinzhao Liu, Karthik Dinesh, Zhiyao Duan, and Gaurav Sharma. Creating a multitrack classical music performance dataset for multimodal music analysis: Challenges, insights, and applications. *IEEE Transactions on Multimedia*, 21(2):522–535, February 2019.

[16] Ruiqi Li, Siqi Zheng, Xize Cheng, Ziang Zhang, Shengpeng Ji, and Zhou Zhao. Muvi: Video-to-music generation with semantic alignment and rhythmic synchronization. *arXiv preprint arXiv:2410.12957*, 2024.

[17] Sizhe Li, Yiming Qin, Minghang Zheng, Xin Jin, and Yang Liu. Diff-bgm: A diffusion model for video background music generation, 2024.

[18] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *ICLR*, 18, 2019.

[19] Xinhao Mei, Chutong Meng, Haohe Liu, Qiuqiang Kong, Tom Ko, Chengqi Zhao, Mark Plumbley, Yuexian Zou, and Wenwu Wang. Wavcaps: A chatgpt-assisted weakly-labelled audio captioning dataset for audio-language multimodal research. *TASLP*, page 15, 2024.

[20] Muhammad Ferjad Naeem, Seong Joon Oh, Youngjung Uh, Yunjey Choi, and Jaejun Yoo. Reliable fidelity and diversity metrics for generative models. In *International conference on machine learning*, pages 7176–7185. PMLR, 2020.

[21] Zachary Novack, Ge Zhu, Jonah Casebeer, Julian McAuley, Taylor Berg-Kirkpatrick, and Nicholas J Bryan. Presto! distilling steps and layers for accelerating music generation. *arXiv preprint arXiv:2410.05167*, 2024.

[22] Roman Solovyev, Alexander Stempkovskiy, and Tatiana Habruseva. Benchmarks and leaderboards for sound demixing tasks, 2023.

[23] Kun Su, Judith Yue Li, Qingqing Huang, Dima Kuzmin, Joonseok Lee, Chris Donahue, Fei Sha, Aren Jansen, Yu Wang, Mauro Verzetti, et al. V2meow: Meowing to the visual beat via video-to-music generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 4952–4960, 2024.

[24] Zeyue Tian, Zhaoyang Liu, Ruibin Yuan, Jiahao Pan, Qifeng Liu, Xu Tan, Qifeng Chen, Wei Xue, and Yike Guo. Vidmuse: A simple video-to-music generation framework with long-short-term modeling. *arXiv preprint arXiv:2406.04321*, 2024.

[25] Shih-Lun Wu, Chris Donahue, Shinji Watanabe, and Nicholas J. Bryan. Music controlnet: Multiple time-varying controls for music generation, 2023.

[26] Yusong Wu, Ke Chen, Tianyu Zhang, Yuchen Hui, Taylor Berg-Kirkpatrick, and Shlomo Dubnov. Large-scale contrastive language-audio pretraining with feature fusion and keyword-to-caption augmentation. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2023.

[27] Zhifeng Xie, Qile He, Youjia Zhu, Qiwei He, and Mengtian Li. Filmcomposer: Llm-driven music production for silent film clips, 2025.

[28] Ye Zhu, Kyle Olszewski, Yu Wu, Panos Achlioptas, Menglei Chai, Yan Yan, and Sergey Tulyakov. Quantized gan for complex music generation from dance videos, 2022.

[29] Le Zhuo, Zhaokai Wang, Baisen Wang, Yue Liao, Chenxi Bao, Stanley Peng, Songhao Han, Aixi Zhang, Fei Fang, and Si Liu. Video background music generation: Dataset, method and evaluation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15637–15647, 2023.

[30] Heda Zuo, Weitao You, Junxian Wu, Shihong Ren, Pei Chen, Mingxu Zhou, Yujia Lu, and Lingyun Sun. Gvmgen: A general video-to-music generation model with hierarchical attentions. *arXiv preprint arXiv:2501.09972*, 2025.

A List of Music-related Categories from PANNs

PANNs classifies events into 527 categories. Among them, we identified the following 157 as music-related categories:

- Singing
- Choir
- Yodeling
- Chant
- Mantra
- Male singing
- Female singing
- Child singing
- Synthetic singing
- Rapping
- Humming
- Music
- Musical instrument
- Plucked string instrument
- Guitar
- Electric guitar
- Bass guitar
- Acoustic guitar

- Steel guitar, slide guitar
- Tapping (guitar technique)
- Strum
- Banjo
- Sitar
- Mandolin
- Zither
- Ukulele
- Keyboard (musical)
- Piano
- Electric piano
- Organ
- Electronic organ
- Hammond organ
- Synthesizer
- Sampler
- Harpsichord
- Percussion
- Drum kit
- Drum machine
- Drum
- Snare drum
- Rimshot
- Drum roll
- Bass drum
- Timpani
- Tabla
- Cymbal
- Hi-hat
- Wood block
- Tambourine
- Rattle (instrument)
- Maraca
- Gong
- Tubular bells
- Mallet percussion
- Marimba, xylophone
- Glockenspiel
- Vibraphone
- Steelpan
- Orchestra
- Brass instrument
- French horn
- Trumpet
- Trombone
- Bowed string instrument
- String section
- Violin, fiddle
- Pizzicato
- Cello
- Double bass
- Wind instrument, woodwind instrument
- Flute
- Saxophone
- Clarinet
- Harp
- Bell
- Church bell
- Jingle bell
- Bicycle bell
- Tuning fork
- Chime
- Wind chime
- Change ringing (campanology)
- Harmonica
- Accordion
- Bagpipes
- Didgeridoo
- Shofar
- Theremin
- Singing bowl
- Scratching (performance technique)
- Pop music
- Hip hop music
- Beatboxing
- Rock music
- Heavy metal
- Punk rock
- Grunge
- Progressive rock
- Rock and roll
- Psychedelic rock
- Rhythm and blues
- Soul music
- Reggae
- Country
- Swing music
- Bluegrass
- Funk
- Folk music
- Middle Eastern music
- Jazz
- Disco
- Classical music
- Opera
- Electronic music
- House music
- Techno
- Dubstep
- Drum and bass
- Electronica
- Electronic dance music
- Ambient music
- Trance music
- Music of Latin America
- Salsa music
- Flamenco
- Blues
- Music for children
- New-age music
- Vocal music
- A capella
- Music of Africa
- Afrobeat
- Christian music
- Gospel music
- Music of Asia
- Carnatic music
- Music of Bollywood
- Ska
- Traditional music
- Independent music
- Song
- Background music
- Theme music
- Jingle (music)
- Soundtrack music
- Lullaby

- Video game music
- Christmas music
- Dance music
- Wedding music
- Happy music
- Funny music
- Sad music
- Tender music
- Exciting music
- Angry music
- Scary music

B Training Details

B.1 Basic Training Configuration

Our training process largely follows the original MMAudio settings with identical optimizer configurations and learning rate scheduling strategies. Specifically, we use a base learning rate of 1×10^{-4} with a linear warmup schedule of 1K steps to train our models for 300K iterations. We employ the AdamW optimizer [5, 18] with $\beta_1 = 0.9$ and $\beta_2 = 0.95$. We notice occasional training collapse (to NaN) if the default $\beta_2 = 0.999$ was used instead. The learning rate is reduced to 1×10^{-5} after 80% of training steps, and further to 1×10^{-6} after 90% of training steps. We use post-hoc EMA [9] with a relative width $\sigma_{rel} = 0.05$ for all models. For efficiency, we use bf16 mixed precision training in all runs. All audio latents and visual embeddings are precomputed offline and loaded during training.

B.2 Fine-tuning Specific Adjustments

For fine-tuning experiments, we make targeted adjustments to training parameters to accommodate domain-specific data training requirements:

Learning Rate: Fine-tuning learning rate is set to 1/10 of the original rate (1×10^{-5}), following standard practice for fine-tuning tasks to preserve pretrained knowledge while adapting to new domain data.

Warmup and Scheduling: Linear warmup steps are adjusted to 500, with learning rate schedule milestones correspondingly adjusted to 60% and 80% of total iterations.

Validation and Saving Intervals: Validation interval, weight saving interval, and checkpoint saving interval are all adjusted to more frequent intervals (every 2K-5K steps) for better monitoring of the fine-tuning process.

Early Stopping: We implement early stopping with patience of 3-5 epochs and minimum delta of 0.001 to prevent overfitting, without changing the maximum iteration limit.

B.3 CLAP-to-CLIP Projection Architecture

To address the missing text modality in our OSSL-v2 dataset, we implement a learnable projection network that transforms CLAP audio embeddings into CLIP-compatible text features. This design choice is motivated by the fundamental architectural differences between CLAP and CLIP embeddings: CLAP produces single 512-dimensional global audio representations, while CLIP text features consist of 77 tokens each with 1024 dimensions to capture fine-grained textual semantics.

Our projection architecture addresses both dimensional and sequential mismatches through a four-stage transformation pipeline. The first stage performs feature projection from the 512-dimensional CLAP space to the 1024-dimensional CLIP space through an intermediate 768-dimensional hidden layer with LayerNorm and GELU activation for stable training. The second stage expands the single global feature into 77 sequential tokens via a linear transformation followed by normalization, effectively distributing the global audio semantics across the expected text sequence length. The third stage introduces learnable positional embeddings to provide sequential structure that mimics natural language token positioning. Finally, the fourth stage applies a refinement projection with dropout regularization to produce the final CLIP-compatible representations.

Mathematically, the complete transformation can be expressed as:

$$\text{CLAPtoCLIP}(x) = \text{Proj}_{\text{final}}\left(\text{Reshape}(\text{Expand}(\text{Proj}_{\text{feat}}(x))) + \mathbf{P}\right) \tag{1}$$

where $x \in \mathbb{R}^{B\times 1\times 512}$ represents the input CLAP features, $\mathbf{P} \in \mathbb{R}^{1\times 77\times 1024}$ denotes the learnable positional embeddings, and the output spans $\mathbb{R}^{B\times 77\times 1024}$ to match CLIP text feature dimensions.

This projection network enables seamless integration of audio-derived semantic features into the existing multimodal architecture while maintaining compatibility with the original CLIP text feature space. The learnable nature of all transformation components allows the network to adapt the audio-text semantic mapping specifically for film soundtrack generation tasks.

B.4 Modality Handling Strategies

Fine-tuning without Text Modality: This approach leverages MMAudio's inherent capability to handle missing modalities through its masking mechanism. During training, we replace all text features with learnable empty tokens ($\emptyset_t$), effectively removing the text branch from the multimodal attention computation while preserving the architectural integrity. This strategy forces the model to establish direct semantic correspondences between visual and musical modalities without relying on textual intermediaries. The model learns to map visual features directly to musical semantics, potentially capturing more nuanced audio-visual relationships that might be lost in text-mediated training. However, this approach sacrifices the semantic richness that text modality typically provides, which may limit the model's ability to understand abstract musical concepts or emotional associations.

Training from Scratch with Visually-Grounded Text Features: Recognizing that text modality serves as a crucial semantic anchor in multimodal learning, we implement a simulation strategy using CLAP-derived audio embeddings to generate pseudo-text features. This approach is motivated by the observation that CLAP models, trained on large-scale audio-text pairs, learn to encode semantic information about audio content in ways that partially overlap with text-based semantic representations. We extract CLAP embeddings from the audio component of each video-music pair and transform them through our projection network to create CLIP-compatible text features. This visually-grounded text modality provides semantic grounding while maintaining the three-way interaction between video, audio, and text that has proven effective in MMAudio's architecture. The complete model is trained from scratch using these visually-grounded text features, allowing all components to co-adapt to the film-specific domain while preserving the multimodal learning benefits.

Fine-tuning with Visually-Grounded Text Features: This hybrid approach combines the representational power of pretrained weights with the domain-specific adaptation enabled by visually-grounded text features. We begin with MMAudio's pretrained parameters, which encode rich audio-visual correspondences learned from general video data, and integrate our CLAP-to-CLIP projection network to handle the text modality. During fine-tuning, the projection network learns to map audio-derived semantic features into the CLIP text space while the pretrained multimodal transformer adapts to film-specific video-music relationships. This strategy preserves the general audio-visual knowledge encoded in the pretrained model while introducing film-specific semantic understanding through the visually-grounded text features. The projection network effectively serves as a domain adaptation layer that translates film soundtrack semantics into the text feature space that the pretrained model expects, enabling efficient knowledge transfer while maintaining architectural consistency.

B.5 Training Resources

All experiments are conducted on NVIDIA A100 GPUs. Training from scratch requires 24 hours on two A100 GPUs, while fine-tuning experiments complete within 8-12 hours on a single A100 GPU. Table 4 summarizes the computational resources used for each experimental configuration.

Table 4: Training resources for different experimental configurations.

Configuration	GPUs	Training Hours	Total GPU-Hours
Training from Scratch	2 A100	24	48
Fine-tuning w/o Text	1 A100	8	8
Fine-tuning w/ Text	1 A100	12	12