ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 979-8-3503-4485-1/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICASSP48485.2024.10447950

GPT-4 DRIVEN CINEMATIC MUSIC GENERATION THROUGH TEXT PROCESSING

*Muhammad Taimoor Haseeb**, *Ahmad Hammoudeh**, *Gus Xia*

Music X Lab, MBZUAI

ABSTRACT

This paper presents Herrmann-1[1], a multimodal framework to generate background music tailored to movie scenes, by integrating state-of-the-art vision, language, music, and speech processing models. Our pipeline begins by extracting visual and speech information from a movie scene, performing emotional analysis on it, and converting these into descriptive texts. Then, GPT-4 translates these high-level descriptions into low-level music conditions. Finally, these text-based music conditions guide a text-to-music model to generate music that resonates with input movie scenes. Comprehensive objective and subjective evaluations attest to the high synthesis quality, congruence, and superiority of our pipeline.

***Index Terms*—** Music For Movie Scenes, Cross-Modal Generation, Content-Based Music Generation

1. INTRODUCTION

Cinema harnesses the power of music to evoke emotions and enhance storytelling, where the right background music can be the difference between a fleeting moment and a memorable scene. The importance of relevant background music is also becoming increasingly evident in user-generated content across digital platforms. The crux of the challenge is to select background music that complements the narrative, intensifies emotions, and captivates the audience. However, this selection process predominantly remains a laborious manual task and often lacks uniqueness, with creators having to sift through countless tracks, finding music that fits the mood and tone, while also being distinctive. This is where AI can offer a faster, cost-effective, and personalized solution.

Though machine learning has ventured into the domain of automated music generation for films, its scope has been limited. This is partly because generating fitting music pieces from visual scenes is a difficult problem. Existing models neglect spoken dialogues and emotional subtext, focus mainly on low-level music synchronization issues, and restrict music representation to MIDI. In contrast, our multimodal design adopts a holistic approach by integrating both speech and visual aspects of diverse scenes, performs sentiment analysis to identify the nuanced emotional undertone of key-frames, and leverages raw audio to enrich output music quality. Our contributions are as follows:

- We present a multimodal method, combining state-of-the-art models, for generating music compositions tailored to movie scenes.
- Comprehensive objective and subjective assessments are conducted to demonstrate our method's effectiveness.
- Our method enhances transparency and interpretability, and simplifies troubleshooting by generating a textual output at each stage in our pipeline.

We name our methodology Herrmann-1, paying homage to the legendary film composer Bernard Herrmann.

2. RELATED WORK

With this research space largely uncharted, only a handful of studies are available, each with its unique attributes and limitations. V2Meow [1] is a multi-stage autoregressive model that surpasses prior systems in visual-audio alignment. The model neglects any dialogue in the video. Also, while it allows for style control via text prompts, producing such low-level descriptions of music might necessitate a nuanced understanding of compositions. Similarly, V-MusProd [2] utilizes music priors and video-music relations and uses three progressive transformer stages to generate full-length background music for general videos. Despite its merits, the emphasis is on video content, ignoring dialogues, and the MIDI format does not capture the depth of raw audio. Built on a transformer-based architecture, CMT [3] uses rhythmic relations between videos and background music and offers genre and instrument controls. Yang et al. [4] proposed a dual linear transformers-based model that neglects sentiment and restricts music representation to MIDI. Some of these models heavily rely on video features like color changes for music generation, which can impact their performance with black-and-white videos. Specialized music generation models for performance-based videos also exist, but their domains do not align with movie scenes and are thus tangential.

* The first two authors contributed equally.

[1] Samples available at: audiomatic-research.github.io/herrmann-1/.

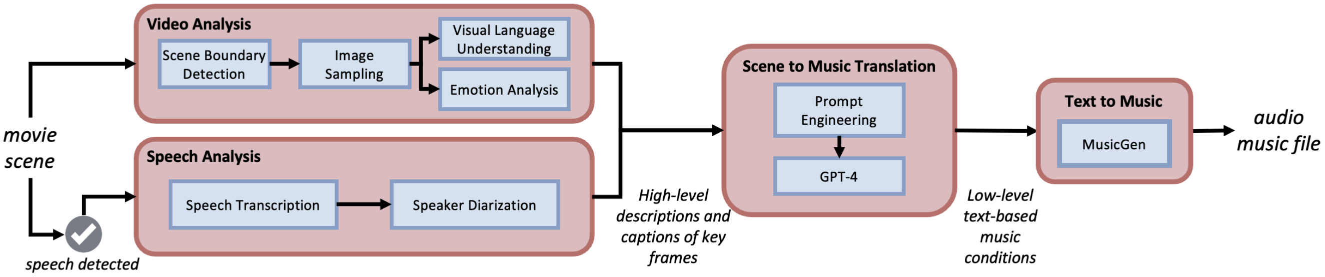

Fig. 1. Our background music generation pipeline takes a movie scene as input and generates a tailored audio music file.

3. METHODOLOGY

Herrmann-1 integrates a suite of publicly available, state-of-the-art models to understand movie scenes and subsequently generate fitting background music. First, we leverage advancements in vision and speech models to extract visual information and spoken dialogue from a scene. In our pipeline, audio and video streams run concurrently. Emotion analysis is also performed on key frames. These high-level insights are then translated into low-level text-based music conditions using a Large Language Model (LLM). Finally, these prompts are fed into a text-to-music model to generate relevant music. While we primarily opted for state-of-the-art models, alternatives were always explored for each pipeline component to ensure the best performance on our test set. Fig. 1 outlines our methodology. Subsequent sections dive deeper into individual components of our pipeline.

3.1. Video Analysis

Scene Boundary Detection & Image Sampling: Scene boundary detection involves identifying and temporally localizing sub-scene boundaries, which are points where significant changes in a scene occur. In our pipeline, we ensure that images are sampled from every sub-scene, providing a comprehensive representation regardless of its length. The literature presents various approaches for this task, including pixel-based methods, histograms, Otsu's method, edge-based techniques, etc. [5].

We employ PySceneDetect[2] for efficient scene boundary detection, leveraging its capacity to detect HSV color space shifts for scene transitions and utilize ffmpeg for scene cuts. After identifying the scenes, our frame sampling algorithm functions as follows: for scenarios with fewer than n detected scenes, a minimum of n images are sampled across them; when there are n or more scenes, one image is sampled from each scene. Given the high processing demands of the downstream multimodal image captioning and sentiment detection stages, over-sampling at this stage could adversely affect subsequent performance. We find n = 15 to be an optimal balance between capturing sufficient scene detail and maintaining a manageable computational load. Our tests confirm the superiority of our sampling method over uniform sampling.

Visual Language Understanding: Visual Language Understanding decodes static photos and videos using natural language. Traditionally, vision-text tasks involved captioning, retrieval, and question-answering. Both video-text and image-text models utilize text and visual encoders. Encoder-decoder models generate outputs post multimodal fusion, while encoder-only models directly harness cross-modal representations. Recent innovations like CLIP [6] and ALIGN [7], arising from large-scale Visual Language Pre-training (VLP), have spurred advanced multimodal models like SimVLM and Florence.

In assessing both video and image captioning models, we observed varied outcomes: PDVC [8] and BMT [9] yielded inconsistent video captions, VidSUM[3] is subtitle dependent, and DSNet's [10] video summarization had mixed outcomes. Among image captioning models, GIT [11], BLIP [12], and CLIP [6] presented challenges in capturing scene context. BLIP2 [13] and CoCa [14] showed promising results in producing accurate and descriptive captions. BLIP2 was selected for its performance and scalability on our test set.

Emotion Analysis: Discerning a scene's emotional undertone is pivotal for generating fitting background music. From classic techniques like KNN and SVM using wavelet energy features, emotion analysis has advanced to modern methods like DNN, CNN, R-CNN, and Fast R-CNN [15]. OpenAI's CLIP has proved very effective for emotion recognition within images, with robust performance on the CIFAR100 dataset [16]. Informed by the emotional categories outlined in Cowen et al. [17], we utilize CLIP, fine-tuned on emotion detection, in a zero-shot setting.

3.2. Speech Analysis

Speech Transcription: Transcription converts scene dialogues into text. This is essential for our study as it enables the analysis of verbal content in videos, enhancing our understanding of scenes. An intriguing development in this domain

[2] https://pyscenedetect.readthedocs.io/

[3] https://github.com/OpenGenus/vidsum

is Whisper2 [18], which uses an encoder-decoder Transformer that trains directly on raw transcripts, eliminating the need for complex data standardization steps. It can accommodate diverse audio inputs including foreign language dialogues in scenes. In our pipeline, we employ Whisper2 for speech transcription.

Speaker Diarization: It follows transcription, ensuring accurate attribution of the text to specific speakers. This process augments our main goal: generating contextually appropriate background music based on speaker dynamics in videos. Noteworthy toolkits include PyAnnote [19], based on YouTube recordings; NVIDIA NeMo, trained on telephone conversations; Simple Diarizer, with a more basic framework; SpeechBrain, a PyTorch-based AI toolkit; Kaldi, catering to researchers; UIS-RNN, Google's diarization model; FunASR, a PyTorch-based open-source toolkit; VBx, an x-vector extractor for diarization. Commercial options also exist, like Google Recorder App, Amazon Transcribe, and IBM Watson Speech To Text API.

We chose PyAnnote due to its efficiency, ease of integration, and adaptability. PyAnnote offers lightweight architecture and a hybrid speaker diarization method, meeting our accuracy and computational efficiency demands. Its open-source nature also guarantees ongoing access.

3.3. Text Based Scene to Music Translation

Prompt Engineering is the process of crafting a contextually rich, informative prompt that can guide a model's output in the desired direction. Sections 3.1 and 3.2 outline our approach to Vision Language Understanding, Emotion Analysis, and Audio Transcription and Diarization to generate high-level descriptions of the input movie scene. Through prompt engineering, a LLM *translates* these high-level descriptions into text-based, low-level music conditions.

High-Level Video Descriptions: Our input prompt is designed to encapsulate key frame captions, speech transcriptions, and emotional nuances. This provides LLM with a high-level description of the video's context through text. For instance, in the famous *I'm Flying* scene from Titanic, where Rose and Jack share a defining, passionate moment at the ship's bow, the high-level video descriptions are:

Image captions: 1) A man and woman standing on the deck of a boat at sunset. ... 19) A woman looking into the eyes of a man. 20) A man and a woman are looking at each other. 21) A man and a woman standing next to each other. ...

Audio transcriptions: ... Speaker 00: Give me your hand. Speaker 00: Now close your eyes. Speaker 00: Go on. Speaker 00: Step up. Speaker 00: Now hold on to the railing.

Video sentiments: Romantic (100%)

Text Based Low-Level Music Conditions: LLM translates the above high-level video descriptions into low-level output that describes in detail what the relevant background music for the scene should sound like. For example, the high-level descriptions from our Titanic scene generated the following music conditions: *Ethereal orchestral piece with soft piano undertones, interlaced with melancholic violin solos, reminiscent of historical romance. Very gentle harp plucks accentuate moments of intimacy, crescendos mirroring the ebb and flow of the sea. ... Capturing the essence of timeless romance and the breathtaking moments of connection between two souls. Subtle choral harmonies emerge, giving depth and warmth, encapsulating the nostalgia and longing found in epic tales of love.* These text-based music conditions then serve as an input prompt for the subsequent text-to-music model. Sample prompts and generated music can be found at: https://audiomatic-research.github.io/herrmann-1/.

We use OpenAI's GPT-4 [20] in our research, renowned for its advanced language comprehension and generation.

3.4. Text-to-Music Generation

Text-to-music generation transforms rich textual outputs from GPT-4 into music via specialized text-to-music models. Recent research in this domain has been vibrant; notable models include ERNIE-Music [21], MusicLM [22], Noise2Music [23], MuseCoco [24], and MusicGen [25].

We selected MusicGen for our pipeline. It transposes textual inputs into music tokens, offering robust control over the generated output music. It stands out for conditional music generation and its single-stage transformer design, which avoids the complexities of multi-model cascading. Its capability to produce high-quality audio music, conditioned on textual features, aligns with our objective to mirror video content's emotion and narrative. The public availability of the model supports further fine-tuning of the model.

4. EVALUATION

Dataset: To evaluate the efficacy of our model, we curated an internal test set comprising movie scenes from YouTube.

4.1. Objective Metrics

Kullback-Leiber Divergence (KLD): Due to a many-to-many relationship between input videos and generated music, a direct waveform comparison between original and generated music isn't sensible. Instead, we use KLD, similar to the approach by Copet et al. [25]. Mathematically, KLD between two probability distributions P and Q is given by:

$$D_{KL}(P||Q) = \sum_x P(x) \log \frac{P(x)}{Q(x)} \tag{1}$$

We employ a LEAF classifier [26] pre-trained on AudioSet to obtain class predictions for both sets of music. Table 1 presents the KLD scores for our method and CMT's method using their publicly available code-base[4]. A lower KLD score

[4] https://github.com/wzk1015/video-bgm-generation/tree/main

for our model suggests a higher similarity between the distributions of original music and music generated by our model.

Fréchet Audio Distance (FAD): FAD scores serve as an indicator of audio quality and resonate with how humans perceive sound [27]. The FAD score between two multivariate Gaussians $\mathcal{N}_b(\mu_b, \Sigma_b)$ and $\mathcal{N}_e(\mu_e, \Sigma_e)$ is given by:

$$\mathbf{F}(\mathcal{N}_b, \mathcal{N}_e) = ||\mu_b - \mu_e||^2 + tr(\Sigma_b + \Sigma_e - 2\sqrt{\Sigma_b \Sigma_e}) \quad (2)$$

Models with low FAD scores tend to produce more credible audio outputs. To calculate FAD scores, we leveraged the widely-used VGGish[5] [28] audio embeddings. As the MusicCaps dataset isn't tailored for cinematic music, calculating FAD scores against it wasn't directly applicable. To address this, we collected a dedicated, 15-hour-long, high-quality, cinematic music dataset. The FAD scores for our pipeline, the original tracks, and the music generated by CMT are shown in Table 1. A lower FAD score for our model indicates the capability of our method to better capture the auditory characteristics of movie music.

Method	KLD Score ↓	FAD Score ↓
Original Tracks	0.00	2.78
Our Method	0.73	4.85
CMT [3]	1.93	13.58

Table 1. Summary of results for objective evaluation.

4.2. Subjective Assessment

Study Design: Besides objective measurements, we also conducted a subjective survey to evaluate the performance of our music generation pipeline. Similar to the method followed by Yang et al. [29], each subject reviewed three versions of five movie scenes: one with the original soundtrack; another with our model-generated music; and a third with a random track from a mixed dataset of human and machine-produced music serving as a *baseline*. All scenes had the same duration (30 seconds). A total of 120 subjects participated in the survey, including 10 professional musicians. The order in which scenes appeared was randomized to prevent bias, and participants were unaware of the music's source. The subjects were asked to rate each sample on a 5-point scale from 1 (very low) to 5 (very high) according to the following criteria:

- **Relevance:** Perceived appropriateness of the background music to the video's contents.
- **Quality:** Assessment of overall music quality, including detection of any musical anomalies.

Results and Discussion: Fig 2 presents the results of our qualitative evaluation. The y-axis represents average ratings and the error bars indicate the Standard Deviation computed using within-subject ANOVA [30]. Our model outperformed the random music baseline in terms of relevance and quality, with statistical significance ($p<0.05$). Our method's relevance ratings were comparable to the original compositions.

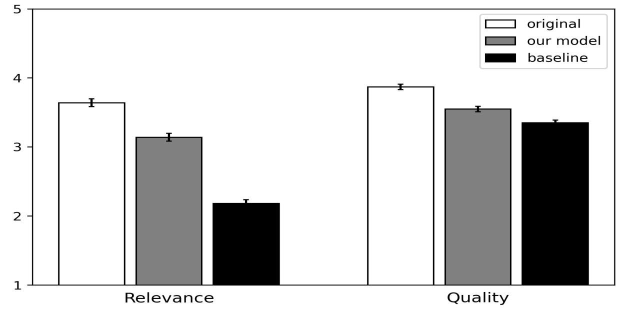

Fig. 2. Subjective evaluation results computed using within-subject ANOVA.

5. CONCLUSION

We contribute a novel pipeline that assembles publicly available, state-of-the-art models to generate background music that complements the emotional undertone and narrative of movie scenes. It extracts visual and speech information from scenes, performs emotional analysis, and translates these insights into descriptive texts. Leveraging GPT-4, these descriptions are then translated into low-level music conditions. By mapping various modalities to text before music generation, we create an abstraction layer, offering insights into how different elements influence the music's mood and style. Finally, the text-to-music model generates music using these text conditions. Experimental results showed our model is successful in generating music that closely mirrors the original scene's music relevance and quality.

We see this study as a significant step in multimodal background music generation for movie scenes, offering an innovative alternative to how filmmakers and content creators approach background music selection. That said, upstream inaccuracies in the pipeline may propagate to subsequent stages, impacting generated music's relevance (e.g. incorrect emotions recognition, brief or sporadic speaker contributions missed by diarization, etc.) The system's intricate architecture could present computational and optimization hurdles. AI-generated music might resemble copyrighted tracks used in training, raising concerns about diversity, ethics, and legality. Nonetheless, the rapid advancements in related fields hold the promise of further refining each element of our pipeline in the future. Future directions may include the inclusion of non-speech audio elements (e.g. lion roaring), performance optimization, or adaptability across varied content types.

6. ACKNOWLEDGMENTS

We would like to thank Prof. Jin Huang and the Xinfuzhuhua Group at ECNU for supporting this project.

[5] tfhub.dev/google/vggish/1

7. REFERENCES

[1] K. Su, J.Y. Li, Q. Huang, D. Kuzmin, J. Lee, C. Donahue, F. Sha, A. Jansen, Y. Wang, M. Verzetti, et al., "V2meow: Meowing to the visual beat via music generation," *arXiv preprint arXiv:2305.06594*, 2023.

[2] L. Zhuo, Z. Wang, B. Wang, Y. Liao, S. Peng, C. Bao, M. Lu, X. Li, and S. Liu, "Video background music generation: Dataset, method and evaluation," *arXiv preprint arXiv:2211.11248*, 2022.

[3] S. Di, Z. Jiang, S. Liu, Z. Wang, L. Zhu, Z. He, H. Liu, and S. Yan, "Video background music generation with controllable music transformer," 2021, ACM.

[4] X. Yang, Y. Yu, and X. Wu, "Double linear transformer for background music generation from videos," *Applied Sciences*, vol. 12, pp. 5050, 05 2022.

[5] B. Reddy and A. Jadhav, "Comparison of scene change detection algorithms for videos," in *IEEE ACCT*, 2015, pp. 84–89.

[6] 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.

[7] C. Jia, Y. Yang, Y. Xia, Y-T. Chen, Z. Parekh, H. Pham, Q.V. Le, Y. Sung, Z. Li, and T. Duerig, "Scaling up visual and vision-language representation learning with noisy text supervision," 2021.

[8] T. Wang, R. Zhang, Z. Lu, F. Zheng, R. Cheng, and P. Luo, "End-to-end dense video captioning with parallel decoding," 2021.

[9] V. Iashin and E. Rahtu, "A better use of audio-visual cues: Dense video captioning with bi-modal transformer," 2020.

[10] W. Zhu, J. Lu, J. Li, and J. Zhou, "Dsnet: A flexible detect-to-summarize network for video summarization," *IEEE Transactions on Image Processing*, vol. 30, pp. 948–962, 2021.

[11] J. Wang, Z. Yang, X. Hu, L. Li, K. Lin, Z. Gan, Z. Liu, C. Liu, and L. Wang, "Git: A generative image-to-text transformer for vision and language," 2022.

[12] J. Li, D. Li, C. Xiong, and S. Hoi, "Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation," 2022.

[13] J. Li, D. Li, S. Savarese, and S. Hoi, "Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models," 2023.

[14] J. Yu, Z. Wang, V. Vasudevan, L. Yeung, M. Seyedhosseini, and Y. Wu, "Coca: Contrastive captioners are image-text foundation models," 2022.

[15] N. Mittal, D. Sharma, and M. L. Joshi, "Image sentiment analysis using deep learning," in *IEEE/WIC/ACM International Conference on Web Intelligence*, 2018, pp. 684–687.

[16] A. Bondielli and L.C. Passaro, "Leveraging clip for image emotion recognition," in *CEUR WORKSHOP PROCEEDINGS*, 2021, vol. 3015.

[17] A.S. Cowen, X. Fang, D. Sauter, and D. Keltner, "What music makes us feel: At least 13 dimensions organize subjective experiences associated with music across different cultures," *Proceedings of the National Academy of Sciences*, vol. 117, no. 4, pp. 1924–1934, 2020.

[18] A. Radford, J.W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, "Robust speech recognition via large-scale weak supervision," 2022.

[19] H. Bredin, R. Yin, J.M. Coria, G. Gelly, P. Korshunov, M. Lavechin, D. Fustes, H. Titeux, W. Bouaziz, and M.P. Gill, "pyannote.audio: neural building blocks for speaker diarization," in *ICASSP*, 2020.

[20] OpenAI, "Gpt-4 technical report," 2023.

[21] P. Zhu, C. Pang, S. Wang, Y. Chai, Y. Sun, H. Tian, and H. Wu, "Ernie-music: Text-to-waveform music generation with diffusion models," 2023.

[22] A. Agostinelli, T. Denk, Z. Borsos, J. Engel, M. Verzetti, A. Caillon, Q. Huang, A. Jansen, A. Roberts, M. Tagliasacchi, M. Sharifi, N. Zeghidour, and C. Frank, "Musiclm: Generating music from text," 2023.

[23] Q. Huang, D. Park, T. Wang, T. Denk, A. Ly, N. Chen, Z. Zhang, Z. Zhang, J. Yu, C. Frank, J. Engel, Q. Le, W. Chan, Z. Chen, and W. Han, "Noise2music: Text-conditioned music generation with diffusion models," 2023.

[24] P. Lu, X. Xu, C. Kang, B. Yu, C. Xing, X. Tan, and J. Bian, "Musecoco: Generating symbolic music from text," 2023.

[25] J. Copet, F. Kreuk, I. Gat, T. Remez, D. Kant, G. Synnaeve, Y. Adi, and A. Défossez, "Simple and controllable music generation," 2023.

[26] N. Zeghidour, O. Teboul, F. de Chaumont Quitry, and M. Tagliasacchi, "Leaf: A learnable frontend for audio classification," 2021.

[27] K. Kilgour, M. Zuluaga, D. Roblek, and M. Sharifi, "Fr'echet audio distance: A metric for evaluating music enhancement algorithms," 2019.

[28] S. Hershey, S. Chaudhuri, D.P.W. Ellis, J.F. Gemmeke, A. Jansen, C. Moore, M. Plakal, D. Platt, R.A. Saurous, B. Seybold, M. Slaney, R. Weiss, and K. Wilson, "Cnn architectures for large-scale audio classification," in *ICASSP*, 2017.

[29] R. Yang, D. Wang, Z. Wang, T. Chen, J. Jiang, and G. Xia, "Deep music analogy via latent representation disentanglement," 2019.

[30] H. Scheffe, *The Analysis of Variance*, vol. 72, John Wiley & Sons, 1999.