Supplementary Material for VMAs: Video-to-Music Generation via Semantic Alignment in Web Music Videos

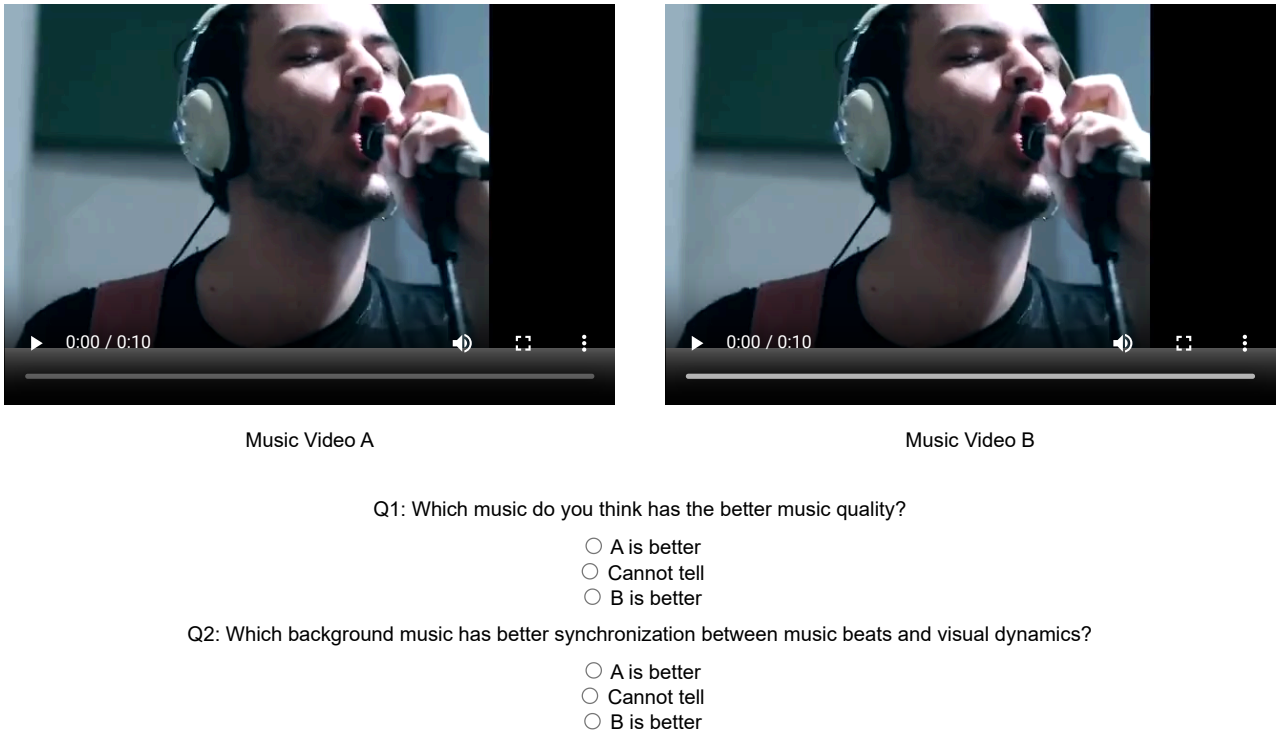

Figure 1. **Human Evaluation.** Human raters are asked to select the generated music that best aligns with a given video and the best music quality. We report the average human preference rate for each method. Note that all samples are present in a random order.

1. Appendix Overview

Our appendix consists of:

1. Implementation Details.
2. Human Evaluation Details.
3. Music Genre Analysis of DISCO-MV
4. Additional Quantitative Results.
5. Qualitative Results.
6. A Supplementary Video.

2. Implementation Details

Audio Tokenization Model and Patterns. To transform a continuous 32 kHz audio into discrete audio tokens, we leverage the pretrained EnCodec [3] with a stride of 640, resulting in a frame rate of 50 Hz and an initial hidden feature size of 64. The embeddings are quantized using a Residual Vector Quantization (RVQ) with four quantizers, each having a codebook size of 2048. As for the codebook pattern, we adopt the delay interleaving pattern [1] to translate 10 seconds of audio into 500 autoregressive steps (audio tokens).

Efficient Video Encoder. Given an input video, we ex-

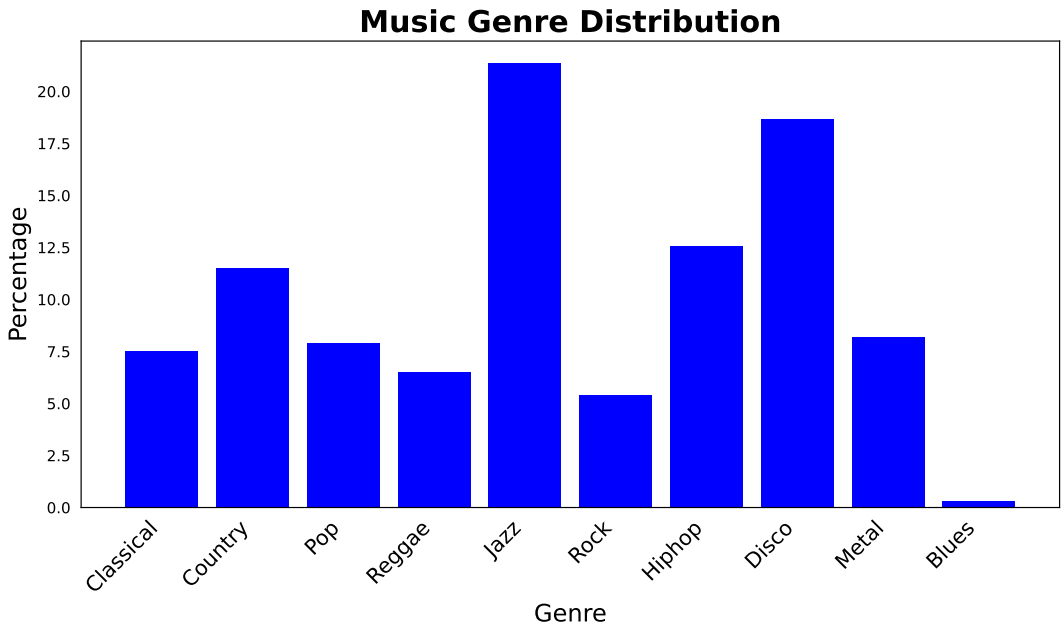

Figure 2. **Music Genre Distribution.** We present the GTZAN genres [12] for the DISCO-MV dataset. Genres are assigned to each soundtrack based on the maximum cosine similarity between its sound embedding and the corresponding genre (text) embedding.

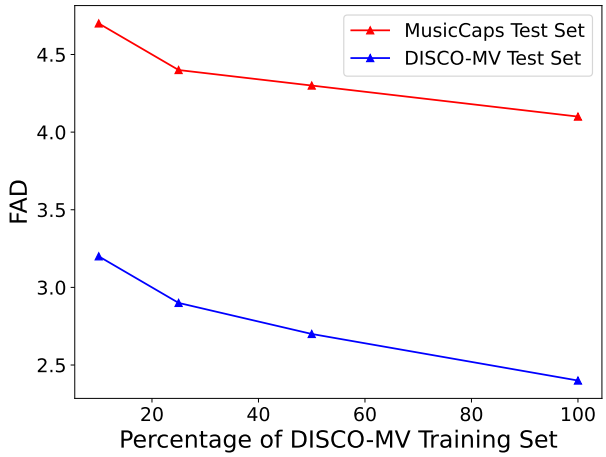

Figure 3. **Impact of the Training Data Size.** We train our method with increasing subsets of DISCO-MV and then evaluate on MusicCaps (**Red**) and DISCO-MV (**Blue**) using FAD metric (the lower the better). Our results illustrate that the size of video-music training data has a significant impact on the generated music quality. These results justify our approach of using Web videos for scaling our video-music training data.

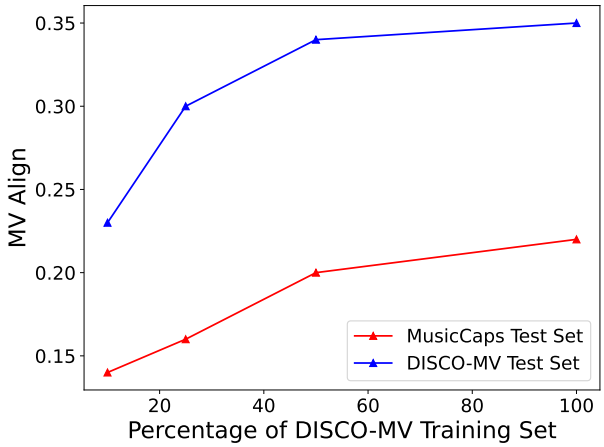

Figure 4. **Impact of the Training Data Size.** We train VMAs with increasing subsets of DISCO-MV and then report the Music-Video Alignment (MV Align) metric on MusicCaps (**Red**) and DISCO-MV (**Blue**).

tract frames at a 9.6 FPS rate, resulting in a total of 96 video frames with a resolution of 224×224 for a 10-second clip. We utilize the pretrained Hiera-Base model [10], which consists of 24 layers and performs downsampling three times through pooling. These 96 video frames are initially processed by a 3D CNN with a $[3 \times 7 \times 7]$ kernel, a $[2 \times 4 \times 4]$ stride, and a $[1 \times 3 \times 3]$ padding for video tokenization. Subsequently, we adjust the original spatial downsampling method in Hiera, namely Q-Pooling, which utilizes the same size of pooling kernels and strides.

The Q-Pooling kernel sizes for the first and second downsampling stages are increased from 2 to 4 (at the 2^{nd} layer) and from 2 to 7 (at the 5^{th} layer), respectively. For

Table 1. **Video-to-Music Retrieval Results.** We conduct a comparison of our designed efficient video encoder with existing video encoders [9, 10] for video-to-music retrieval. Our efficient video encoder is generalizable to the video-to-music retrieval task.

Method	Video Encoder	#Frames	V2M R@1 ↑	V2M R@10 ↑
MVPt [11]	CLIP [9]	16	4.2	17.3
	Hiera [10]	16	4.8	18.1
VMAs	VMAs	96	**6.3**	**26.7**

the third downsampling stage, the Q-Pooling kernel size is maintained at 2×2, implemented at the 21^{st} layer. At the 24^{th} layer, the final layer, the model only increases the channel dimension through the liner projection, yielding video representations of dimension $48 \times 1 \times 1 \times 768$.

Autoregressive Audio Decoder. We adopt the autoregressive transformer models of pretrained MusicGen-medium [1] as our autoregressive audio decoder. The decoder consists of 48 transformer layers in a feature dimension of 1536 with 24 standard causal and multi-head attention blocks.

Optimiazation. We train VMAs on 10-second video clips from DISCO-MV using the AdamW optimizer [8] with a batch size of 8 on each GPU. The training takes approximately four days using 32 NVIDIA GPUs across 4 nodes. Each node is equipped with 8 GPUs, 92 CPUs, and 1000G of memory. We utilize D-Adaptation [2] to select the overall learning rate automatically. A cosine learning rate schedule with a warmup of 4000 steps is deployed alongside an exponential moving average with a decay of 0.99. We set α in eq(3) to 0.05 and β in the main draft to 0.25.

3. Human Evaluation Details

As depicted in Figure 1, given a pair of video-music samples with the same video but different music generated by two methods, human raters are asked to choose their preferred video-music sample based on the following prompts: 1) *Which music do you think has the better music quality?* and 2) *Which background music has better synchronization between music beats and visual dynamics?"* For each question, the subjects can choose one of the two methods or the third option "Cannot tell." We collected approximately 200 subjects in the human evaluation by presenting results from a random method against VMAs. In each survey, Each conducts 10 evaluations randomly selected among 50 evaluations.

4. Music Genre Analysis of DISCO-MV

Following the setup in DISCO-10M [7], we implement zero-shot music genre classification for DISCO-MV by utilizing pretrained CLAP [13] embeddings extracted from 10-second music clips. Genre classification is conducted through genre-specific prompts ("This audio is a <genre> song") and identifying the genre via top-1 cosine similarity in a shared latent space for each music clip. In Figure 2, we report the GTZAN genre distribution [12] of our DISCO-MV dataset. Jazz and disco genres are predominant while ensuring a wide range of musical diversity. However, we note that the blues genre has few samples in DISCO-MV due to the limited number of blues music in the original DISCO-10M dataset.

5. Additional Quantitative Results

Impact of Training Data Size. Similar to our analysis in the main draft, in Figure 3 and Figure 4, we visualize our model's performance on DISCO-MV as the training data increases in the FAD and Music-Video Alignment (MV Align) metrics, respectively. We observe consistent improvement in both metrics when the DISCO-MV data size increases. Despite the Vid2MLDM model not accounting for low-level music-video beat synchronization, its performance improves with more training data. These additional results confirm that our large-scale DISCO-MV can improve music-video beat alignment as well as music quality with larger training data scales.

Video-Music Retrieval Task. To evaluate our approach's generalization capability, in Table 1, we also compare VMAs against MVPt [11] for the video-to-music retrieval tasks on DISCO-MV. Following the pipeline and size evaluation set of video-to-music retrieval framework [11], we randomly sample 2,000 videos in the DISCO-MV test set and extracted music and video representations (i.e., features in Eq. (2) of main paper) using our trained model on video-to-music generation task. We used these feature representations for the video-to-music retrieval task and measured performance using the standard Recall@1 and Recall@10 metrics. For a fair comparison, we implement MVPt [11] using CLIP [9] and Hiera [10] as the video encoders and train it under the same conditions as VMAs. These results indicate that VMAs is more accurate and generalizes better compared to MVPt (i.e., **26.7** vs. **18.1** and **17.3** R@10) on the video-to-music task.

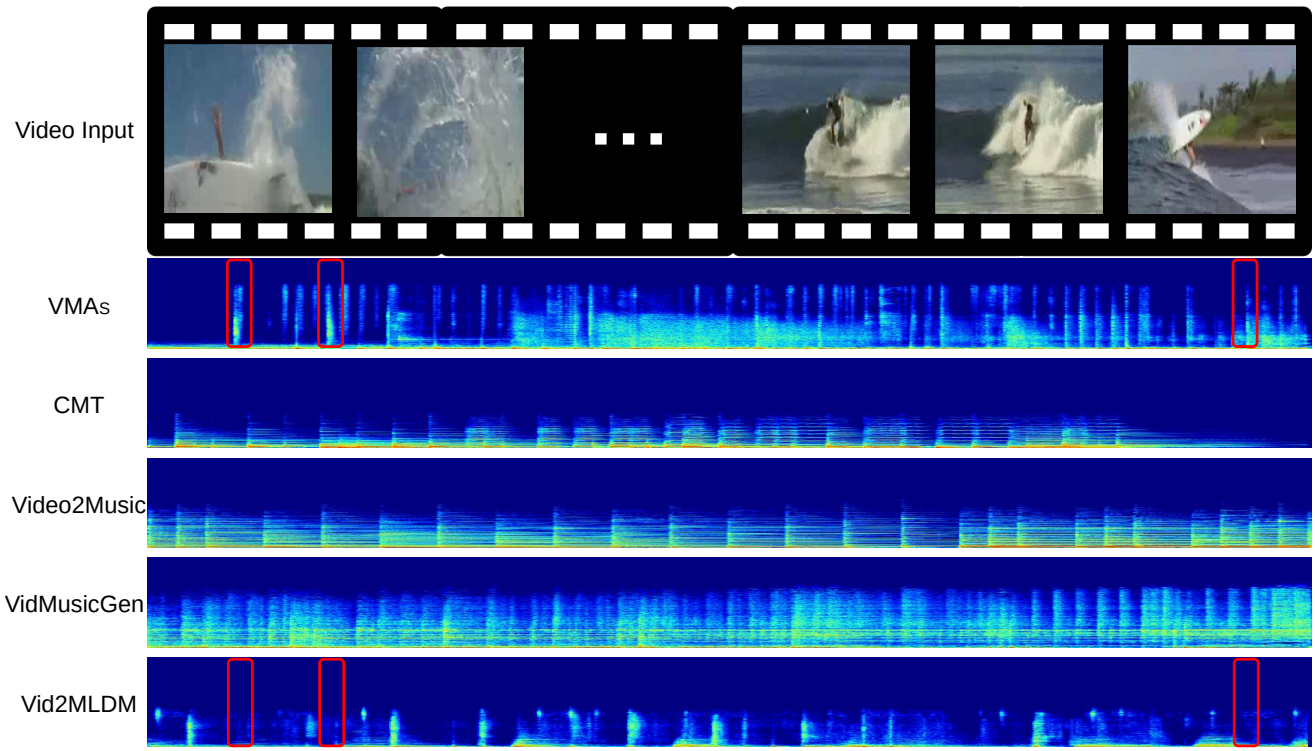

Figure 5. **Qualitative Video-to-Music Generation Results.** Here, we illustrate qualitative music generation results for a given silent video input. The generated music sample is visualized as a spectrogram. We compare the results of our model with CMT [4], Video2Music [6]), VidMusicGen [1] and Vid2MLDM [5]. We note that most prior video-to-music generation approaches produce music beats of uniform intensity. In contrast, our model generates music beats that align well with dynamic video content, i.e., significant movements when a surfer changes direction during a sharp turn in this particular example.

6. Qualitative Results

In Figure 5, we visualize our generated music results as a 2D spectrogram. We also include the results of the following video-to-music generation methods: CMT [4], Video2Music [6], VidMusicGen [1] and Vid2MLDM [5]. All results are obtained using the same video input shown at the top of the Figure.

Based on these results, we observe that symbolic music generation methods(i.e., CMT and Video2Music) often generate music with uniform music beat patterns, which is suboptimal as the music fails to match the temporal dynamics of the video content. Furthermore, the existing waveform methods (i.e., VidMusicGen and Vid2MLDM) struggle to generate music consistently synchronized with dynamic low-level video events.In comparison, the music beats generated by our model (highlighted in red boxes) have higher intensity when a surfer in the video performs a dramatic turn. This demonstrates that our model generates music that reflects the pace and magnitude of the actions occurring in the video.

References

[1] Jade Copet, Felix Kreuk, Itai Gat, Tal Remez, David Kant, Gabriel Synnaeve, Yossi Adi, and Alexandre Défossez. Simple and controllable music generation. In *NeurIPS*, 2023. 1, 3, 4

[2] Aaron Defazio and Konstantin Mishchenko. Learning-rate-free learning by d-adaptation. In *ICML*, 2023. 3

[3] Alexandre Défossez, Jade Copet, Gabriel Synnaeve, and Yossi Adi. High fidelity neural audio compression. *TMLR*, 2023. 1

[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 *ACM MM*, 2021. 4

[5] Deepanway Ghosal, Navonil Majumder, Ambuj Mehrish, and Soujanya Poria. Text-to-audio generation using instruction guided latent diffusion model. In *ACM MM*, 2023. 4

[6] Jaeyong Kang, Soujanya Poria, and Dorien Herremans. Video2music: Suitable music generation from videos using an affective multimodal transformer model. *arXiv Preprint*, 2023. 4

[7] Luca Lanzendörfer, Florian Grötschla, Emil Funke, and Roger Wattenhofer. Disco-10m: A large-scale music dataset. In *NeurIPS*, 2023. 3

[8] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv Preprint*, 2017. 3

[9] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 3

[10] Chaitanya Ryali, Yuan-Ting Hu, Daniel Bolya, Chen Wei, Haoqi Fan, Po-Yao Huang, Vaibhav Aggarwal, Arkabandhu Chowdhury, Omid Poursaeed, Judy Hoffman, et al. Hiera: A hierarchical vision transformer without the bells-and-whistles. In *ICML*, 2023. 2, 3

[11] Dídac Surís, Carl Vondrick, Bryan Russell, and Justin Salamon. It's time for artistic correspondence in music and video. In *CVPR*, 2022. 3

[12] George Tzanetakis and Perry Cook. Musical genre classification of audio signals. *TASLP*, 2002. 2, 3

[13] 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. 3