Towards Temporally Synchronized Visually Indicated Sounds Through Scale-Adapted Positional Embeddings

Xinhao Mei **Gael Le Lan** **Haohe Liu** **Zhaoheng Ni**
Varun Nagaraja **Anurag Kumar** **Yangyang Shi** **Vikas Chandra**
Meta, USA

Abstract

The task of video-to-audio (V2A) generation focuses on producing audio clips that are semantically aligned and temporally synchronized with silent video inputs. Despite recent progress, achieving precise audio-visual synchronization remains a significant challenge. Existing methods often rely on onset detection models, post-ranking or contrastive audio-visual pretraining to improve synchronization, overlooking the critical role of positional embeddings. In this work, we argue that positional embeddings are key to achieve accurate synchronization. Given the strict temporal correspondence between video and audio signals, we present two key arguments: first, visual features and audio tokens should employ identical positional embeddings to enhance temporal correspondence; second, the scale difference between visual features and audio tokens introduces alignment difficulties that negatively affect cross-modal alignment. To address these issues, we propose scale-adapted positional embeddings (SAPE) which are designed to account for discrepancies in sequence lengths and scales between visual features and continuous audio tokens. Experiments on the Greatest Hits dataset show that SAPE significantly improves audio-visual synchronization, achieving a state-of-the-art onset accuracy of 65.8%.

1 Introduction

Recent advancements in video-to-audio (V2A) generation have significantly enhanced the quality and coherence of audio outputs relative to video inputs, marking substantial progress in the field (1; 2; 3). Despite these advancements, achieving precise audio-visual synchronization remains a critical challenge. Accurate alignment of audio with corresponding visual content is essential, as even slight mismatches between sound and visual events can be distinctly noticeable and diminish the overall user experience. To address this issue, researchers have proposed several approaches to improve the audio-visual synchronization in V2A systems, such as contrastive audio-visual pretraining (CAVP) (4; 5), using separate models to predict onsets as conditions (6; 7; 8), and post-ranking (9). While these approaches have shown promise, they also increase system complexity and do not completely resolve the synchronization challenges.

Video and audio are inherently sequential data, and achieving precise synchronization requires not only capturing the temporal dynamics within each modality but also accurately aligning the correspondence between them. In the era of Transformers (10), positional embeddings play a crucial role for encoding temporal information within sequences. Despite their importance, the specific role of positional embeddings in video-to-audio generation systems has been largely underexplored in existing literature. In this work, we specifically address the synchronization challenge through the lens of positional embeddings. We develop a flow matching-based V2A system that employs a non-autoregressive Transformer to generate continuous and compressed audio embeddings. The

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

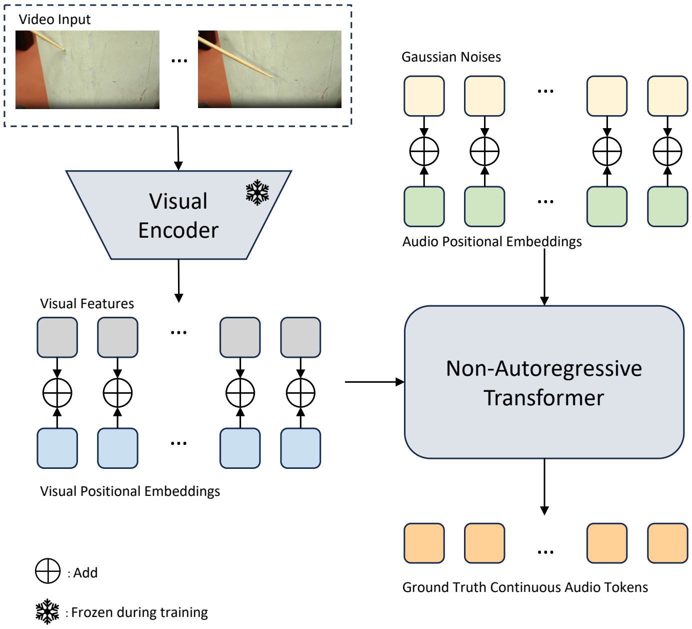
Video Input
Visual Encoder
Visual Features
Visual Positional Embeddings
Gaussian Noises
Audio Positional Embeddings
Non-Autoregressive Transformer
Ground Truth Continuous Audio Tokens
: Add
: Frozen during training

Figure 1: Overview of the flow matching based V2A system. Positional embeddings are applied to visual features and audio tokens.

Transformer attends to visual features via a cross-attention block. Given the strict correspondence between video and audio, we make two arguments: first, visual features and audio tokens should use identical positional embeddings to enhance temporal correspondence; second, the different scales of visual features and audio tokens negatively affect the correspondence. We experiment with Sinusoidal positional embeddings and Rotary Position Embedding (RoPE) (11), adapting RoPE for the cross-attention scenario. Our experiments lead to the introduction of scale-adapted positional embeddings, designed to account for discrepancies in sequence lengths and resolutions between visual features and audio tokens. Our findings underscore the critical role of positional embeddings in achieving precise audio-visual synchronization and demonstrate that our proposed scale-adapted positional embeddings significantly improve synchronization, eliminating the need for separate models, post-processing, or pretraining.

2 Proposed Method

2.1 System Overview

We adapt the flow matching-based text-to-audio generation model, MelodyFlow (12), for video-to-audio generation by replacing the text encoder with a visual encoder. The visual encoder extracts one visual embedding for each frame from the video input. Variational auto-encoder (VAE) based continuous audio codec is used to encode raw waveforms into continuous audio embeddings. A non-autoregressive Transformer generates continuous audio embeddings from Gaussian noise[1], conditioned on visual features through cross-attention blocks. These generated continuous audio embeddings are then converted back to waveforms using the decoder of the VAE audio codec. The overview of the system is illustrated in Figure 1, where audio codec is not shown for simplicity. It's important to note that different positional embeddings can be applied differently, such as RoPE being used in attention operations. In Figure1, we use Sinusoidal as an example.

Assume we sample a video of duration T seconds at a frame-per-second (FPS) rate of F, and that the visual encoder extracts one embedding per frame. Consequently, the total number of visual features will be $T \times F$. For audio, assume the corresponding clip is compressed into continuous tokens at a

[1] The flow matching model involves more complex processes, please refer to (12; 13) for a more detailed explanation.

rate of S embeddings per second using a VAE-based audio codec. Therefore, the total number of audio embeddings will be $S \times T$. In most cases, S is significantly larger than F.

2.2 Scale-Adapted Positional Embeddings

Given that both video and audio are temporally sequential data, achieving precise audio-visual synchronization depends on two critical aspects: first, effectively capturing the sequential characteristics inherent within each modality; and second, accurately capturing the temporal correspondence between video and audio. It is intuitive to employ positional embeddings to address the first objective, as they naturally facilitate the modeling of temporal relationships within each modality. Consequently, we need to use two sets of positional embeddings for this purpose: visual positional embeddings for visual features and audio positional embeddings for audio tokens.

Given the strict temporal correspondence between video and audio, we first argue that using identical positional embeddings for both visual features and audio tokens can enhance temporal alignment, enabling the Transformer model to better capture the temporal relationships between the two modalities. Specifically, the visual positional embeddings P_v and the audio positional embeddings P_a are defined as follows:

$$\begin{aligned} P_v &= \{p_1, p_2, \ldots, p_{F\times T}\} \\ P_a &= \{p_1, p_2, \ldots, p_{S\times T}\} \end{aligned} \tag{1}$$

where each p_i represents any kind of positional embedding at a position i.

However, this approach introduces a significant challenge: the resolution mismatch between visual features and audio tokens. Visual frames are typically sampled at a much lower rate than audio, leading to a difference in temporal granularity between the two modalities. As a result, even if identical positional embeddings are applied to both, the positional embeddings for higher-resolution audio tokens are not aligned correctly with those for lower-resolution visual features. For instance, p_i in P_v and p_i in P_a correspond to different time interval, which makes the model difficult to learn the correct alignment.

To address this issue, we propose the use of scale-adapted positional embeddings (SAPE), which adjusts the positional embeddings of the modality with a lower resolution (such as video) by selecting positional embeddings at intervals defined by the ratio of the higher-resolution and lower-resolution. This ensures that positional embeddings for video and audio corresponding to the same time intervals are aligned. Specifically, given the ratio between the audio frame rate and the visual frame rate, defined as $r = \frac{S}{F}$, we adjust the visual positional embeddings by selecting every r-th positional embedding to align with the audio positional embeddings. The modified visual positional embeddings P'_v are computed as:

$$P'_v = \{p_1, p_{2\times r}, \ldots, p_{F\times T\times r}\} \tag{2}$$

This approach ensures that the visual positional embeddings are temporally aligned with those of the corresponding audio embeddings, maintaining temporal correspondence between the two modalities despite their different scales. SAPE can be used in any positional embeddings and architectures requiring positional embeddings, here we experiment with Sinusoidal positional embeddings and RoPE. To apply RoPE for visual features, we adapt it for the cross-attention scenario, where queries are from the audio tokens and keys are from the visual features.

3 Experiments

3.1 Dataset

Open-domain videos often contain non-visible actions or objects emitting sounds, posing significant challenges for objectively evaluating temporal synchronization. To facilitate a reliable objective evaluation of temporal synchronization, we focus on visually indicated sounds, where sounds are directly caused by the physical interactions or actions shown in the video. We conduct experiments on the Greatest Hits dataset with de-noised audio version (14). Greatest Hits contains 977 videos of people using a drumstick to hit, scratch different objects. Given the variable duration of the videos in Greatest Hits, we segment each video into 10-second non-overlapping clips. Following the official dataset split, our training set consists of 2251 clips, while the test set includes 744 clips.

Table 1: Experimental results on Greatest Hits dataset. PE denotes positional embeddings.

Video FPS	Ratio	Visual PE	Audio PE	Onset ACC (%) ↑	Onset AP (%) ↑	FAD ↓
5	10	Sin	Sin	44.4	59.7	0.35
5	10	SAPE Sin	Sin	55.7	67.3	0.40
5	10	Sin	RoPE	31.1	54.3	**0.26**
5	10	Sin + RoPE	RoPE	47.9	60.7	0.39
5	10	Sin + SAPE RoPE	RoPE	55.7	66.5	0.40
5	10	RoPE	RoPE	32.4	53.0	0.35
5	10	SAPE RoPE	RoPE	**57.6**	**67.7**	0.43
10	5	Sin	Sin	61.1	71.4	0.36
10	5	SAPE Sin	Sin	61.5	70.3	0.37
10	5	Sin	RoPE	31.1	54.1	**0.28**
10	5	Sin + RoPE	RoPE	60.4	68.8	0.38
10	5	Sin + SAPE RoPE	RoPE	65.4	72.5	0.36
10	5	RoPE	RoPE	32.3	52.2	0.38
10	5	SAPE RoPE	RoPE	**65.8**	**72.9**	0.41

Table 2: Comparison results with baselines. Videos are 2-seconds with a FPS of 15.

Methods	Onset ACC (%) ↑	Onset AP (%) ↑	FAD ↓
CondFoleyGen (9)	26.5	60.0	6.10
SyncFusion (6)	56.9	**84.4**	5.50
Flow (ours)	**62.1**	78.2	**0.39**

3.2 Implementation Details

For the video input, we use the *ViT-B/16* version from the CLIP model (15) as the visual encoder. Each video frame is processed independently, and the CLS tokens from each frame are sequentially arranged as the visual features. The visual encoder is kept frozen during training, and a linear projection layer maps the visual features into the dimension of the Transformer model. We experiment with video frame rates of 5 and 10. For audio input, we resample all audio clips to 48 kHz. A VAE audio codec is employed to compress the waveforms into continuous embeddings with a feature dimension of 128 and a temporal rate of 50 embeddings per second. The VAE codec is pretrained on the Pond5 sound effects dataset [2]. The Transformer model has 12 layers, each with 16 attention heads and a dimensionality of 1024.

Our models are trained for 15k steps using a batch size of 256. We use the AdamW (16) optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.95$. The learning rate is set to 1e-4, with a warm-up phase spanning the first 4,000 steps. Classifier-free guidance (17) is also applied during training, where 20% of the visual features are randomly replaced with zero vectors. For inference, we use a classifier-free guidance scale of 4.0. RMSNorm (18) is employed inside Transformer layers to stabilize training.

To ensure a fair comparison with previous work, we compare our best model against SyncFusion (6) and CondFoleyGen (9). Since these baselines were trained on 2-second video clips at 15 FPS, we train an additional model under the same conditions (2-seconds and 15 video FPS). This model is trained for 30k steps with other settings same as above.

3.3 Evaluation Metrics

We follow SyncFusion (6) to evaluate the system's performance from two key aspects: audio quality using Fréchet Audio Distance (FAD) (19) and audio-visual synchronization using onset accuracy and onset average precision (14). FAD is the most popular audio quality metric used in sound effects generation works, which measures the distributional difference between generated and reference audio by comparing feature representations extracted using the VGGish model (20). For onset metrics, they first detect the onsets in both the generated and ground truth audio, and then calculate the accuracy and precision by comparing the detected onsets. The confidence of each onset is determined by the

[2] https://www.pond5.com/sound-effects/

normalized wave amplitude, and a window size of 0.1 second is used to account for small timing discrepancies. We use the same confidence interval of 0.05 second as prior works (6; 9).

3.4 Results

Table 1 presents the results across different combinations of positional embeddings and audio-visual scale ratios. We first compare the impact of positional embeddings on synchronization. When using different positional embeddings for visual features and audio tokens (Sinusoidal for visual tokens and RoPE for audio tokens), the synchronization performance is consistently the worst across both scale ratios. In contrast, using Sinusoidal positional embeddings for both visual features and audio tokens leads to significant improvements in synchronization performance, particularly when the video FPS is higher. However, using RoPE for both visual and audio tokens also leads to bad synchronization performance. We found it is important to use a separate visual positional embeddings (Sinusoidal here) when using RoPE for both visual and audio positional embeddings. These results highlights our first argument, using identical positional embeddings for visual features and audio tokens could improve the synchronization significantly.

When SAPE is used in conjunction with Sinusoidal positional embeddings at a lower frame rate, the onset accuracy is improved from 44.4% to 55.7%. This suggests that SAPE effectively compensates for the scale mismatch between visual features and audio tokens, ensuring that the positional embeddings remain aligned across modalities. Interestingly, as the FPS increases, the marginal benefit of using SAPE with Sinusoidal positional embeddings diminishes. This can be explained by the fact that the difference between consecutive Sinusoidal embeddings becomes smaller at higher frame rates, reducing the impact of scale adaptation. When SAPE is combined with RoPE, we observe the best synchronization results across both FPS settings. This underscores the flexibility of SAPE in handling various embedding schemes and demonstrates its robustness in improving synchronization performance, even without the need of separate Sinusoidal positional embeddings for the visual modality. Overall, these findings validate our arguments and show that our proposed SAPE substantially improved synchronization performance for different positional embeddings, particularly in low FPS scenarios.

For the audio quality comparison, all methods achieve low FAD scores, likely due to the strong flow matching-based V2A model and the state-of-the-art VAE audio codec. For different scale ratios, models with different visual and audio positional embeddings achieve the lowest FAD scores, while others show similar performance. However, the pretrained model used for calculating FAD scores was trained on open-domain audios. The extent to which FAD correlates with true audio quality under this specific scenario remains an open question.

Table 3.2 compares our results with CondFoleyGen (9) and SyncFusion (6), where we use SAPE RoPE for visual positional embedding and RoPE for audio positional embedding. Our model achieves the highest onset accuracy but slightly falls behind SyncFusion in terms of onset average precision. This difference could be due to a higher rate of false positives or lower waveform amplitudes, as the confidence score is calculated based on waveform amplitude. Importantly, our model with proposed SAPE demonstrates state-of-the-art synchronization performance without the need for separate models. Additionally, our FAD score is significantly lower than the baselines. Compared with 10-seconds results in Table 1, the accuracy is lower and the average precision is higher possibly due to the short duration.

4 Conclusions

Achieving precise audio-visual synchronization is a significant challenge in video-to-audio generation. In this work, we demonstrated the critical role of positional embeddings in achieving accurate synchronization. We argued that identical positional embeddings should be used in both modalities to enhance temporal correspondence. We then proposed Scale-Adapted Positional Embeddings (SAPE) to address the scale mismatch between visual and audio modalities. Our experimental results confirmed the effectiveness of SAPE in enhancing cross-modal alignment, achieving state-of-the-art synchronization performance. Future work will focus on validating the approach in more open-domain scenarios.

References

[1] Roy Sheffer and Yossi Adi, "I hear your true colors: Image guided audio generation," in *International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2023.

[2] Vladimir Iashin and Esa Rahtu, "Taming visually guided sound generation," in *British Machine Vision Conference*, 2021.

[3] Xinhao Mei, Varun Nagaraja, Gael Le Lan, Zhaoheng Ni, Ernie Chang, Yangyang Shi, and Vikas Chandra, "FoleyGen: Visually-guided audio generation," *arXiv preprint arXiv:2309.10537*, 2023.

[4] Simian Luo, Chuanhao Yan, Chenxu Hu, and Hang Zhao, "Diff-Foley: Synchronized video-to-audio synthesis with latent diffusion models," in *Advances in Neural Information Processing Systems*, A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, Eds., 2023, vol. 36, pp. 48855–48876.

[5] Yongqi Wang, Wenxiang Guo, Rongjie Huang, Jiawei Huang, Zehan Wang, Fuming You, Ruiqi Li, and Zhou Zhao, "Frieren: Efficient video-to-audio generation with rectified Flow Matching," *arXiv preprint arXiv:2406.00320*, 2024.

[6] Marco Comunità, Riccardo F Gramaccioni, Emilian Postolache, Emanuele Rodolà, Danilo Comminiello, and Joshua D Reiss, "SyncFusion: Multimodal onset-synchronized video-to-audio Foley synthesis," in *International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2024, pp. 936–940.

[7] Junwon Lee, Jaekwon Im, Dabin Kim, and Juhan Nam, "Video-Foley: Two-stage video-to-sound generation via temporal event condition for Foley sound," *arXiv preprint arXiv:2408.11915*, 2024.

[8] Yiming Zhang, Yicheng Gu, Yanhong Zeng, Zhening Xing, Yuancheng Wang, Zhizheng Wu, and Kai Chen, "FoleyCrafter: Bring silent videos to life with lifelike and synchronized sounds," *arXiv preprint arXiv:2407.01494*, 2024.

[9] Yuexi Du, Ziyang Chen, Justin Salamon, Bryan Russell, and Andrew Owens, "Conditional generation of audio from video via Foley analogies," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 2426–2436.

[10] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, 2017, pp. 5998–6008.

[11] Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu, "Roformer: Enhanced transformer with rotary position embedding," *Neurocomputing*, vol. 568, pp. 127063, 2024.

[12] Gael Le Lan, Bowen Shi, Zhaoheng Ni, Sidd Srinivasan, Anurag Kumar, Brian Ellis, David Kant, Varun Nagaraja, Ernie Chang, Wei-Ning Hsu, et al., "High fidelity text-guided music generation and editing via single-stage flow matching," *arXiv preprint arXiv:2407.03648*, 2024.

[13] Apoorv Vyas, Bowen Shi, Matthew Le, Andros Tjandra, Yi-Chiao Wu, Baishan Guo, Jiemin Zhang, Xinyue Zhang, Robert Adkins, William Ngan, et al., "Audiobox: Unified audio generation with natural language prompts," *arXiv preprint arXiv:2312.15821*, 2023.

[14] Andrew Owens, Phillip Isola, Josh McDermott, Antonio Torralba, Edward H Adelson, and William T Freeman, "Visually indicated sounds," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2405–2413.

[15] 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 *International Conference on Machine Learning*. PMLR, 2021, pp. 8748–8763.

[16] Ilya Loshchilov and Frank Hutter, "Decoupled weight decay regularization," in *International Conference on Learning Representations*, 2019.

[17] Jonathan Ho and Tim Salimans, "Classifier-free diffusion guidance," in *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021.

[18] Biao Zhang and Rico Sennrich, "Root mean square layer normalization," in *Advances in Neural Information Processing Systems*, 2019, vol. 32.

[19] Kevin Kilgour, Mauricio Zuluaga, Dominik Roblek, and Matthew Sharifi, "Fréchet audio distance: A metric for evaluating music enhancement algorithms," *arXiv preprint arXiv:1812.08466*, 2018.

[20] Shawn Hershey, Sourish Chaudhuri, Daniel PW Ellis, Jort F Gemmeke, Aren Jansen, R Channing Moore, Manoj Plakal, Devin Platt, Rif A Saurous, Bryan Seybold, et al., "CNN architectures for large-scale audio classification," in *International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2017, pp. 131–135.

[21] Yipin Zhou, Zhaowen Wang, Chen Fang, Trung Bui, and Tamara L Berg, "Visual to sound: Generating natural sound for videos in the wild," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 3550–3558.

[22] Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N. Dauphin, "Convolutional sequence to sequence learning," in *Proceedings of the 34th International Conference on Machine Learning*, 2017, p. 1243–1252.

A Related Works

A.1 Temporally-Synchronized Video-to-Audio Generation

Initial efforts in video-to-audio generation have primarily focused on enhancing semantic correspondence between generated audio and input video (21; 1; 2). More recent studies have shifted focus towards improving audio-visual synchronization. One line of work involves aligning audio and visual features in latent space through contrastive audio-visual pretraining (CAVP). For example, Diff-Foley (4) proposes to learn semantically and temporally aligned latent features, which serve as conditions in a Diffusion model. Frieren (5) regulates CAVP-pretrained visual features to the same length as audio tokens and employs channel-wise concatenation to enforce synchronization. Additionally, approaches like SyncFusion (6), FoleyCrafter (8), and Video-Foley (7) incorporate separate onset detection models. These models predict temporal onsets or timestamps from videos, which are then used as conditions to guide the audio generation. Furthermore, CondFoleyGen (9) generated multiple audio clips for a single input video and used an audio-visual synchronization model to select the most aligned clip among the generations. These methods, which rely heavily on separate models, post re-ranking, or contrastive pretraining, not only increase the overall system complexity but still do not completely resolve the synchronization challenges.

A.2 Positional Embeddings

Positional embeddings are essential in Transformer-based architectures (10) to compensate for the absence of inherent sequential structure in the attention mechanism. These embeddings enable the model to incorporate sequence or spatial order information across various types of data. While various positional embeddings have been developed (22; 10; 11), most are designed for tasks involving a single modality, such as text and vision. The impact of positional embeddings on cross-modal alignment remains largely unexplored. In this study, we experiment with both Sinusoidal (10) and Rotary Positional Embeddings (11) to assess their impact on cross-modal tasks, specifically in the context of video-to-audio generation.

Sinusoidal positional embeddings (10) is a type of absolute positional embedding, using fixed sine and cosine functions with varying wavelengths to encode the positions of tokens in a sequence. These embeddings are typically added directly to the contextual representations, providing a smooth and continuous representation of positional information. On the other hand, Rotary Positional Embeddings

(RoPE) (11) apply a rotation matrix to both the query and key vectors within the attention mechanism. In this way, RoPE encodes absolute positional information using a rotation matrix and naturally incorporates relative position dependency into the self-attention formulation, effectively unifying absolute and relative positioning methods. RoPE has shown superior performance in most NLP tasks.