TECMO: Efficient Temporal Cross-interaction Module for long synchronized Audio-Video generation

Alex Ergasti Giuseppe Tarollo Filippo Botti Tomaso Fontanini Claudio Ferrari
Massimo Bertozzi Andrea Prati
University of Parma, Department of Engineering and Architecture
Parma, Italy
{alex.ergasti, giuseppegabriele.tarollo, filippo botti, tomaso.fontanini, claudio.ferrari2, massimo.bertozzi, andrea.prati}@unipr.it

***Abstract*—Joint audio-video (AV) generation is still a significant challenge in generative AI, primarily due to four critical requirements: quality of the generated samples, seamless multimodal synchronization, with audio tracks that match the visual data and vice versa, limitless video duration, and training and inference efficiency. In this paper, we present TECMO, a novel transformer-based architecture that addresses these key challenges of AV generation. After exploring three distinct cross-modality interaction modules, we propose a novel lightweight fusion module which emerged as the most effective and computationally efficient approach for aligning audio and visual modalities. We trained our model on AIST++ and Landscape datasets measuring metrics for both image quality (FVD and KVD) as well as audio quality (FAD) and AV syncronization (CAVP). Our experimental results demonstrate that TECMO outperforms existing state-of-the-art models in multimodal AV generation tasks. Our code and checkpoints are available on https://github.com/ErgastiAlex/TECMO-FLAV.**

I. Introduction

Despite extraordinary progress achieved in the field of generative AI, the development of effective and efficient joint audio-video (AV) generation approaches still represents a significant challenge. In recent years, many works have been proposed, which yet deal with just a single modality at a time, such as, for example, video [1], [2], [3], or audio generation [4], [5], [6], [7], [8]. In addition, multimodal approaches that can shift from one modality to another have been developed as well [1], [5], [6], [9], yet the task of simultaneous AV generation remains quite unexplored due to the significant challenges it encompasses. Other than the clear technical difficulties, we identified four major critical requirements that an effective AV generation system should comply with: *(i)* audio and video quality, *(ii)* seamless multimodal synchronization, *(iii)* long video generation, *(iv)* training and inference efficiency. In other words, the duration of the generated video should not be constrained by a fixed length, while ensuring good quality without producing degenerate results and audio tracks that consistently match the dynamics of the visual data. Additionally, due to the complexity of this particular task, the computational requirements are often prohibitive and need to be reduced to ensure practical applicability. Although there have been recent proposals, they struggle to meet all the above features.

Among the few works that addressed this task, MM-Diffusion (MMD) [10] pioneered a novel fusion mechanism that allowed joint AV generation. However, such a solution, is restricted in terms of video duration, which is fixed in length during the diffusion process. Following the introduction of transformers in diffusion model architectures [11], Kim *et al.* [12] proposed a novel method to generate longer videos. In order to do so, an auto-regressive approach is needed, where the generation of a new sequence of frames is conditioned by the last generated frames of the previous sequence. Even though this approach enables the generation of videos with a variable number of frames, in the long run the quality may deteriorate because of the error accumulation that affects auto-regressive kind of strategies. A solution to this problem has been proposed by Ruhe *et al.* [3] by introducing Rolling Diffusion, a novel technique for training diffusion models for video generation that uses a sliding window denoising process. It is based on the idea that, in a sequence, the prediction of frames that are distant in the future is more uncertain with respect to temporally-closer ones. Thus, the amount of noise of the diffusion process for each frame should be proportional to the temporal distance. This approach resulted preferable to the auto-regressive one for improving the quality and consistency of longer sequences, although it does not account for the generation of synchronized audio tracks.

Finally, when developing a system able to generate multiple modalities, modeling the interaction between them is crucial. This is typically handled by attention mechanism like cross-attention and self-attention layers. The big downside of this approach is the huge amount of resources required that can limit scalability and can quickly become unsustainable.

In light of the above, in this paper we propose a novel architecture for joint AV generation, called TECMO, specifically designed to address the key challenges discussed so far. In general, our method demonstrates better performance in comparison to previous approaches in terms of generation quality, AV consistency and efficiency. It also goes a step forward by allowing the generation of high-quality long video sequences without a pre-determined length dictated by the window size. Our solution allows for the generation of paired AV sequences **without any restriction on their duration**, while maintaining high multimodal synchronization and quality. To achieve these results, we carefully designed several components in our architecture. In particular, we revisit the way audio and video are encoded so as to adapt the multimodal stream to be processed via rolling diffusion mechanism and

we propose a novel cross-modality interaction module, which, compared to self-attention, is lighter, faster and with better performances.

II. Related Work

a) Diffusion Models: Diffusion models have achieved great success in various tasks due to their versatility and scalability [5], [10], [13], [14], [15] and have presented numerous backbones for different applications [11], [16], [17]. A drawback of diffusion models though is that to generate new samples, the traditional diffusion process involves a complex backward path that requires multiple integration steps.

Initially proposed by Ho *et al.*. [18], diffusion models demonstrated that a simple iterative denoising process with a U-Net backbone could generate high-fidelity samples. Based on this, DiffWave [15] demonstrated that diffusion models could be applied effectively to the synthesis of raw audio waveforms, opening the door to their use in speech and music tasks. Later, Latent Diffusion Models (LDM) [16] introduced a scalable approach by performing the diffusion process in a compressed latent space instead of pixel space, drastically improving computational efficiency and enabling high-resolution image synthesis. This innovation became the foundation of Stable Diffusion [14].

In parallel, new backbones were proposed to replace or improve upon the traditional U-Net architecture: Diffusion Transformer (DiT) [11] demonstrated that a pure Vision Transformer operating on latent patches can scale more effectively than convolutional U-Nets, showing superior performance as model size increases, while U-ViT [17] introduced a unified ViT-based backbone that treats all elements (*i.e.*, noisy image patches, timesteps, and conditions) as tokens, and employs long skip connections to preserve fine details, offering a flexible and generalizable alternative to convolution-heavy designs.

b) Multimodal Generation: Multimodal models have the goal of learning a more general and comprehensive representation of multiple data modalities, such as video, sound, and text. In particular, for these tasks several key challenges have to be faced, such as temporal coherence for time-dependent data and modality alignment, but also more practical ones like computational performance. Typically, previous studies focused on single-modality generation [2], [3], [19] or text-to-video, text-to-audio, and other conditional generation [1], [4], [6], [20], [21], [22], which are common applications for generative models. Among these, the most successful ones are based on diffusion models, but despite their versatility, the number of applications in multimodal tasks remains limited and lacks variety, due to the complexity of such problems.

Nevertheless, an application that has gained popularity is joint audio-video generation, which recently saw a surge of different approaches based on diffusion models [10], [12], [13], [23], [24], [25]. Among these, MM-Diffusion (MMD) [10] proposed a U-Net based architecture, featuring two separate branches for audio and video, merged by a random-shift attention block, which allowed sound and visual data to influence and synchronize with each other. Upon MMD, AV-DiT [13] exploited a shared DiT-XL/2 backbone [11], pre-trained on ImageNet, by introducing trainable adapter layers for audio and video, demonstrating better alignment and generation of modality. Finally, Kim *et al.*[12] introduced a novel parametrization of the diffusion timestep for the forward process, applying a different diffusion timestep across each modality and temporal dimension. This approach, combined with their model architecture, is capable of surpassing previous work. However, the output produced is limited to 2.125 seconds, and an autoregressive mechanism is still required to produce video longer than 2.125 seconds, with the risk of error accumulation and performance degradation. Beyond architectural innovations, information bottleneck principles have been explored to improve robustness in multimodal learning. In audio-visual tasks [26], variational bottleneck methods compress modality-specific noise while preserving semantic content, and high-order formulations [27], [28] have regulated temporal correlations in spiking networks. While these approaches explicitly impose theoretic constraints, TECMO takes a complementary perspective by leveraging fine-grained audio-visual alignment through architectural design.

Our method improves upon such models by exploiting a rolling approach to generate high-quality, coherent, long-duration videos with aligned audio. Moreover, we propose a novel strategy to fuse and align audio and video during training through a lightweight cross-modality interaction block. This design also allows our model to perform video-to-audio and audio-to-video tasks, along with joint audio-video generation. Finally, since TECMO does not employ any video or audio encoders, input and output lengths are not constrained to be multiples of the encoder length, allowing training with an arbitrary number of frames and the generation of videos of unconstrained duration.

III. Methodology

Our proposed architecture, called TECMO, is a rolling rectified-flow model designed for AV generation. The two main technical innovations introduced in TECMO consist in: *(i)* the design of the Temporal Cross-interaction Module, which allows the multi-modal cross interaction to be done way more efficiently w.r.t. standard self- or cross-attention while resulting in improved quality; *(ii)* a novel strategy for handling video and audio, which operates on each video frame and audio segment independently, allowing for improved audio-video matching and synchronization. For training, we modified the rolling diffusion methodology [3] to enable training using rectified flow matching [29], [30]. In the following, we first summarize the technical details of the rolling flow matching and how we adapted its training scheme (Sect. III-A), and then introduce the architecture (Sect. III-B).

A. Background

a) Flow Matching: Flow matching [29], [30] aims to simplify the diffusion process. In particular, it defines the forward path as a linear trajectory between a sample x drawn from the data distribution and noise ϵ obtained from a Gaussian

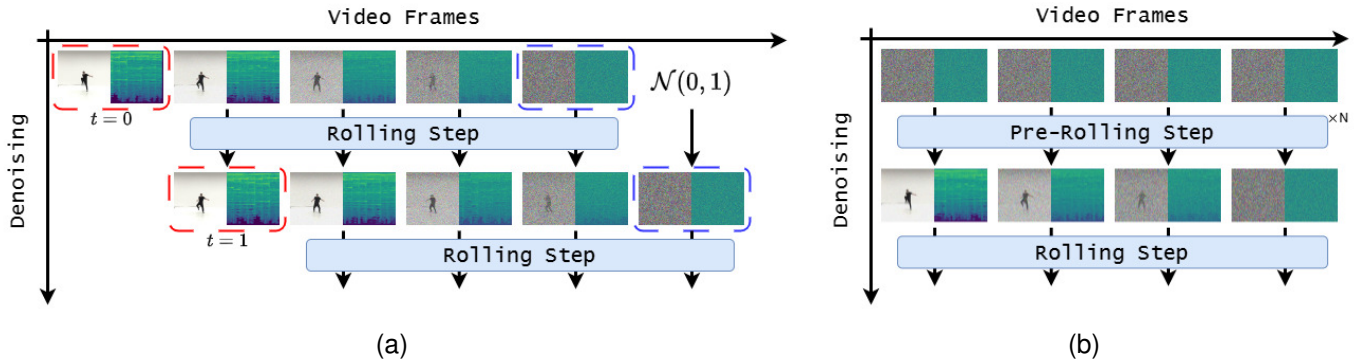

Fig. 1: a) Rolling phase: at each step, a new clean frame is produced (highlighted in red) and subsequently removed from the window. Then, a new noisy frame, (highlighted in blue), is appended to the end of the window. b) Pre-rolling phase: the frames are gradually denoised starting from a full noise configuration. The pre-rolling phase goes on for N steps, until the window is ready for the rolling phase.

distribution $\mathcal{N}(0,1)$, where the trajectory depends on a time parameter, represented by t, with t ranging from 0 to 1.

$$x_t = tx + (1-t)\epsilon \tag{1}$$

The model is then trained to predict the velocity vector ϕ, which drives a sample from noise to the data distribution. The velocity vector ϕ is defined as the derivative of the trajectory path with respect to t:

$$\phi = \frac{dx_t}{dt} = (x - \epsilon) \tag{2}$$

Hence, the model predicts the velocity vector $\hat{\phi}(x_t, t)$ taking as input both x_t and the timestep t. The final loss is:

$$\mathcal{L}_{\text{FM}}(x) = \mathbb{E}_{t\sim\mathcal{U}(0,1),\epsilon\sim\mathcal{N}(0,1)}\lambda_t||(x-\epsilon) - \hat{\phi}(x_t,t)||^2 \tag{3}$$

The loss is scaled by a weight factor λ_t. From [31], the best factor is obtained with $\lambda_t = \text{logit-normal}(t; 0, 1)$.

b) Rolling Flow Matching: Rolling Diffusion Model, introduced by Ruhe *et al.* [3], serves as a foundational framework for generating infinite video sequences without necessitating an auto-regressive approach. It employs a sliding window denoising technique, in which each frame corresponds to a distinct, but progressively higher, denoising timestep. Once the initial frame is fully denoised, the entire window shifts to allow a new noisy frame at the end of the window. Rolling Diffusion model introduces two phases: an initial pre-rolling phase handles the initial state where both modalities start as pure noise; then, the model enters a rolling phase designed for continuous generation in which the model produces content sequentially.

c) Training and Inference: We adapt the rolling diffusion and flow matching for multimodal generation as follows: given an encoded video $\mathbf{v} \in \mathbb{R}^{T\times c\times h\times w}$, obtained via a VAE encoder, and audio, represented as the mel spectrogram, $\mathbf{a} \in \mathbb{R}^{T\times F/T\times N_m}$ (see Sec. III-B.a for further details), we create the noisy versions $\hat{v}_k, \hat{a}_k$ by interpolating with Gaussian noise $\epsilon_{v,k}, \epsilon_{a,k} \sim \mathcal{N}(0,1)$ according to timestep parameter t_k for each frame $k \in [0, T-1]$:

$$\hat{v}_k = t_k v_k + (1-t_k)\epsilon_{v,k} \tag{4}$$

$$\hat{a}_k = t_k a_k + (1-t_k)\epsilon_{a,k} \tag{5}$$

The timestep parameter t_k employs two distinct scheduling mechanisms. In the pre-rolling phase (Fig. 1b), frames and mel spectrograms can be entirely noise, enabling generation from scratch. In the rolling phase (Fig. 1a), frames exhibit progressively higher noise levels across temporal positions, facilitating continuous generation. During training, hyperparameter θ controls the probability of sampling each phase:

$$\mathbf{t}(w) = \begin{cases} t^p(w) = 1 - \frac{k+w}{T}, & \theta \text{ times.} \\ t^r(w) = \text{clamp}(1 - (\frac{k}{T} + w); 0, 1), & 1-\theta \text{ times.} \end{cases} \tag{6}$$

where $w \sim \mathcal{U}(0,1)$ provides stochastic variation within each scheduling regime.

The training objective, derived from Eq. 3, combines flow matching losses for both modalities:

$$\mathcal{L}(\mathbf{v}, \mathbf{a}) = \mathcal{L}_{\mathbf{v}}(\mathbf{v}, \mathbf{t}(w), \epsilon_v) + \mathcal{L}_{\mathbf{a}}(\mathbf{a}, \mathbf{t}(w), \epsilon_a) \tag{7}$$

During inference, the reverse process begins with a sequence of fully noisy frames and initially follows the pre-rolling phase scheduler $t_k^p(w)$. In the rolling phase, the model continuously denoises all frames within the sliding window. Once the first frame reaches complete denoising, the entire window shifts forward by one position, discarding the fully denoised frame and introducing a new noisy frame at the end of the sequence. This rolling mechanism enables long audio-video generation while maintaining consistent quality throughout the sequence.

B. Architecture

Our architecture (see Fig. 2), called TECMO, is a transformer-based model designed to allow AV generation of *any length*. This is possible since we do not use any audio or video encoders, which typically process the video/audio in chunks, thus imposing the generated AV sequences to be multiples of their output sizes. We instead opted for processing video and audio frame-by-frame. As detailed in the following paragraph, this design enables the generation of videos of any length, while also improving AV synchronization.

The model processes both video and audio in two separate branches. Fusion of different distributed modalities is avoided

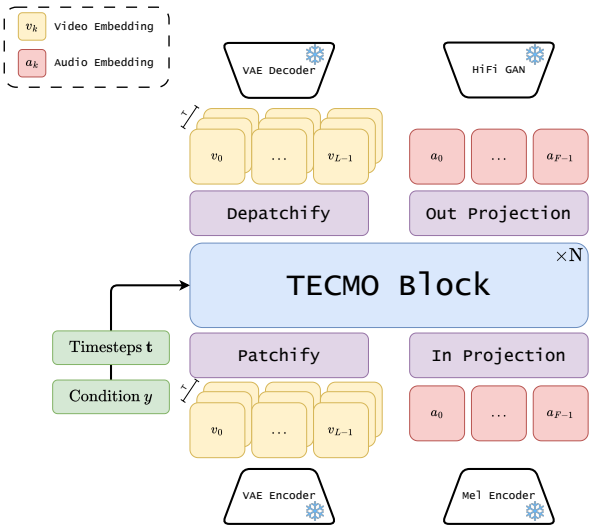

Fig. 2: An overview of our TECMO model architecture.

at early stages; they are instead combined later in the architecture. This allows for an early *intra-modality* interaction via self-attentions, before the inter-modality interaction.

a) Video and Audio encoding: To generate videos of any length, we designed our system to synthesize samples frame-by-frame. To reduce the spatial dimension of each frame of the video, our approach uses an image encoder, the same employed in single image generation [16], instead of a video encoder, avoiding any temporal compression. The output of the encoder is a video $\mathbf{v} \in \mathbb{R}^{T\times c\times h\times w}$, where T is the temporal dimension (number of frames), c represents the number of encoder channels and $h = H/f$, $w = W/f$, with H and W being the original size of the video and f being the downsampling factor. The audio $\mathbf{a} \in \mathbb{R}^{F\times N_m}$ is instead represented by its mel spectrogram obtained from the raw waveform, where F is the number of time frames and N_m the number of mel bins. Since each modality is encoded without performing any temporal compression (*i.e.*, without any video or audio encoder), a one-to-one reference between audio and video frames can be established, improving audio-video synchronization as shown in Sec IV. Each video frame can be associated with the corresponding part of the mel spectrogram, having size F/T, for a better interaction between the two modalities. This is shown in Fig. 3.

The processing occurs in parallel for both modalities. The video passes through a patchify layer, producing a feature embedding tensor $\mathbf{v} \in \mathbb{R}^{T\times L\times D}$, where L is the number of spatial patches per frame (computed as $L = \frac{h\times w}{p^2}$, with frame dimensions h, w and patch size p) and D is the hidden size. Simultaneously, the audio undergoes a linear projection to obtain $\mathbf{a} \in \mathbb{R}^{F\times D}$, where F is the number of audio segments and D matches the hidden dimension of the video, resulting in $T \times L$ video patches and F audio patches.

b) TECMO block: The main block of our architecture is divided into two parallel branches, one for the video and one for the audio. The video branch sequentially applies spatial and temporal attention. The audio branch applies only temporal attention since the audio has no spatial structure.

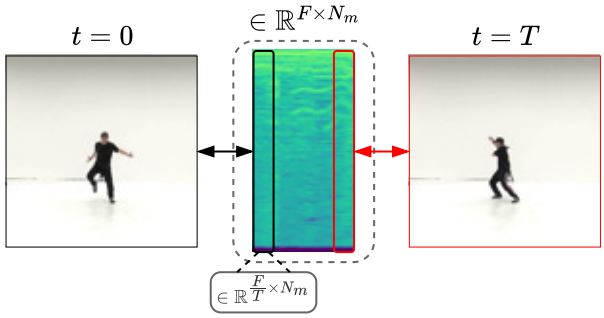

Fig. 3: Temporal alignment between video frames and mel spectrogram segments. Each video frame corresponds to a fixed-size section (F/T) of the mel spectrogram, allowing for a precise 1:1 mapping between audio and video.

Both branches incorporate time-dependent feature modulation before and after each attention layer. This is performed by a series of Temporal Scale & Shift layers that make use of adaptive layer normalization (AdaLN) [11]: each video frame k and the corresponding part of the audio are modulated by the corresponding timestep embedding obtained from t_k and, optionally, by a timestep-invariant class conditioning embedding y. After the attention layers, the cross-modality fusion is performed. To this scope, we propose and explore 3 different alternative block structures, as shown in Fig. 5.

Algorithm 1 Temporal Cross-interaction Module

Input: $\mathbf{v} \in \mathbb{R}^{B\times T\times L\times D}$, $\mathbf{a} \in \mathbb{R}^{B\times T\times \frac{F}{T}\times D}$,
$a_c, v_c \in \mathbb{R}^{B\times T\times D}$
Output: $\mathbf{v_{scale}}, \mathbf{v_{shift}}, \mathbf{v_{gate}}, \mathbf{a_{scale}}, \mathbf{a_{shift}}, \mathbf{a_{gate}}$
$\mathbf{a_{avg}} \leftarrow \text{mean}(\mathbf{a}, \text{dim} = 2)$ ▷ $\mathbf{a_{avg}} \in \mathbb{R}^{B\times T\times D}$
$\mathbf{v_{avg}} \leftarrow \text{mean}(\mathbf{v}, \text{dim} = 2)$ ▷ $\mathbf{v_{avg}} \in \mathbb{R}^{B\times T\times D}$
if v2 **then**
    $\mathbf{v_{cond}} \leftarrow \text{Lin}_a(\mathbf{a_{avg}})$
    $\mathbf{a_{cond}} \leftarrow \text{Lin}_v(\mathbf{v_{avg}})$
else if v3 **then**
    $\mathbf{v_{cond}} \leftarrow \text{Lin}_a(\mathbf{a_{avg}}) + \mathbf{v_c}$
    $\mathbf{a_{cond}} \leftarrow \text{Lin}_v(\mathbf{v_{avg}}) + \mathbf{a_c}$
end if
$[\mathbf{v_{scale}}, \mathbf{v_{shift}}, \mathbf{v_{gate}}] \leftarrow \text{Lin}_{vc}(\mathbf{a_{cond}})$
$[\mathbf{a_{scale}}, \mathbf{a_{shift}}, \mathbf{a_{gate}}] \leftarrow \text{Lin}_{ac}(\mathbf{v_{cond}})$

The first block structure (Fig. 5a) implements the cross-modal interaction through self-attention. Audio embeddings are reshaped to $\mathbb{R}^{T\times F/T\times D}$ and concatenated with video embeddings, obtaining $\mathbf{av} \in \mathbb{R}^{T\cdot(L+F/T)\times D}$. The feature map is then processed by causally masked self-attentions (Fig. 4). The self-attention mask enables custom causal interactions between video and audio modalities. Frame embeddings of index k are conditioned by all current and previous video and audio embeddings with index $\leq k$. Similarly, the audio embeddings of the index k interact causally with each other (that is, the attention mask of the audio embeddings is a lower triangular), while being conditioned by all the frame embeddings at the current index k and all previous frame and audio embeddings

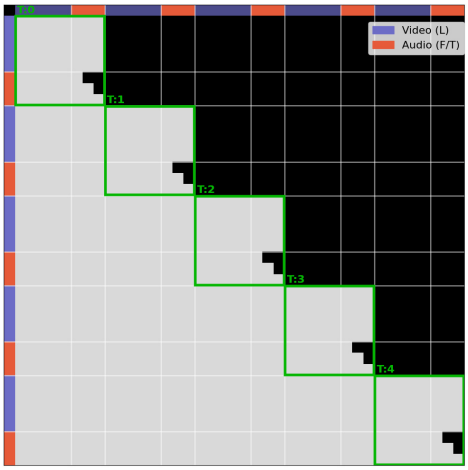

Fig. 4: Illustration of causal mask attention in block (a). Video and audio embeddings for each given frame t are highlighted with a green box. Video frame patches (highlighted in blue) can attend to themselves, all past video frames and audio, plus current audio patches. Audio patches (highlighted in red) follow causal attention, attending only to past video frames and audio, plus current video frames.

with the index $\leq k$. This asymmetric attention pattern allows to maintain strict causal constraints across both modalities for temporal consistency within a single self-attention. Then, the result is split again into the corresponding video and audio embeddings and summed to the respective branch. Finally, time-dependent feature modulation is also performed before and after the feedforward layer. Although this approach offers sophisticated cross-modal interaction, it requires much higher computational and memory costs w.r.t. the following blocks.

The second block structure (Fig. 5b) substitutes the self-attention of the first block with a lightweight modulation mechanism that starts by computing modality averages (Alg. 1). More in detail, the video features $\mathbf{v} \in \mathbb{R}^{T\times L\times D}$ are reduced to $\mathbf{v_{avg}} \in \mathbb{R}^{T\times 1\times D}$ by averaging the embeddings on the L dimension, while audio features $\mathbf{a} \in \mathbb{R}^{F\times N_m}$ are first reshaped to $\mathbb{R}^{T\times \frac{F}{T}\times D}$ and then reduced to $\mathbf{a_{avg}} \in \mathbb{R}^{T\times 1\times D}$ by averaging the embeddings on the $\frac{F}{T}$ dimension. In doing so, the spatial dimensions L and $\frac{F}{T}$ are reduced to 1. Video $\mathbf{v}$ and audio $\mathbf{a}$ features can so be conditioned with the other modality while preserving the temporal dimension T. Ultimately, this maintains the 1:1 frame-to-audio mapping that is critical for synchronization. Next, $\mathbf{v_{cond}}$ and $\mathbf{a_{cond}}$ are obtained by projecting $\mathbf{v_{avg}}$ and $\mathbf{a_{avg}}$ respectively. Then both $\mathbf{v_{cond}}$ and $\mathbf{a_{cond}}$ are further projected to obtain the scale, shift and gate parameters that will be used in the Temporal Scale & Shift layers that are placed before and after the feedforward layer of each modality branch. However, we noticed that this module struggles to propagate the timestep (and the optional condition), hindering the generation process. Hence, we introduced the third and final block.

The third and final block structure (Fig. 5c), built upon the second block, sums the timestep embedding t and the (optional) class embedding y to the modality average embedding, in order to further propagate these conditions to the feedforward layers. In detail, $\mathbf{v_{cond}}$ and $\mathbf{a_{cond}}$ are obtained as a sum of the projection of $\mathbf{a_{avg}}$ plus $\mathbf{v_c}$ and the projection of $\mathbf{v_{avg}}$ plus $\mathbf{a_c}$ respectively, where $\mathbf{v_c}$ and $\mathbf{a_c}$ are the embeddings obtained from the timestep t summed with the class embedding y for video and audio.

Importantly, since the averaging does not operate across the temporal axis, it does not reduce temporal bandwidth. Therefore, it does not suppress high-frequency synchronization signals. Indeed, high-frequency synchronization cues in audiovisual generation arise from fine-grained variations across time (i.e., changes between adjacent frames), rather than from spatial patches within a frame or individual frequency bins within a short audio window. Furthermore, it is worth noting that in all the blocks, cross-modality interaction is strategically placed after attention layers but before feedforward layers, allowing each branch to independently process information through its respective attention mechanisms before influencing the other modality.

IV. Experiments

A. Model details

Our model is composed of 12 TECMO blocks and a window size (*i.e.* T) of 10 frames. To encode each video frame we employ the stable diffusion encoder [16], which has a downsize factor $f = 8$ and with $c = 4$. For audio, we use the same vocoder as [6], with $N_m = 256$. θ is fixed to 0.2. We trained our model on 2 NVIDIA L40S for 7 days.

B. Datasets

We follow previous work [10] for evaluation. We train our model on two datasets, Landscape [10] and AIST++ [32]. Landscape contains 9 different settings (*i.e.*, explosion, fire crackling, raining, splashing water, squishing water, thunder, underwater bubbling, underwater burbling, and wind noise) providing 2.7 hours of high-quality audio-video pairs. AIST++ [32] is a subset of AIST [33] that provides 5.2 hours of video featuring paired audio and dancer movements.

C. Evaluation metrics

Current state-of-the-art methods [10], [12], [13] generate AV sequences at 64×64 pixel resolution. The only exception is [10], which additionally reports results at 256×256 resolution. For fair comparison, we evaluate TECMO against other state-of-the-art models using the resolution reported in the corresponding papers. Metrics are calculated following the standard protocol *i.e.* on 2048 generated sequences at 16 frames per video sample. For clearer visualization, all qualitative results shown in this paper are at 256×256 resolution.

Following [10], we used Fréchet Video Distance (FVD) and Kernel Video Distance (KVD) to measure video quality, employing the I3D video classifier pre-trained on Kinetics-400 [34]. To measure audio quality, we used Fréchet Audio Distance (FAD) calculated from a pre-trained AudioCLIP classifier [35]. Instead, CAVP [36] is used to assess audio video alignment.

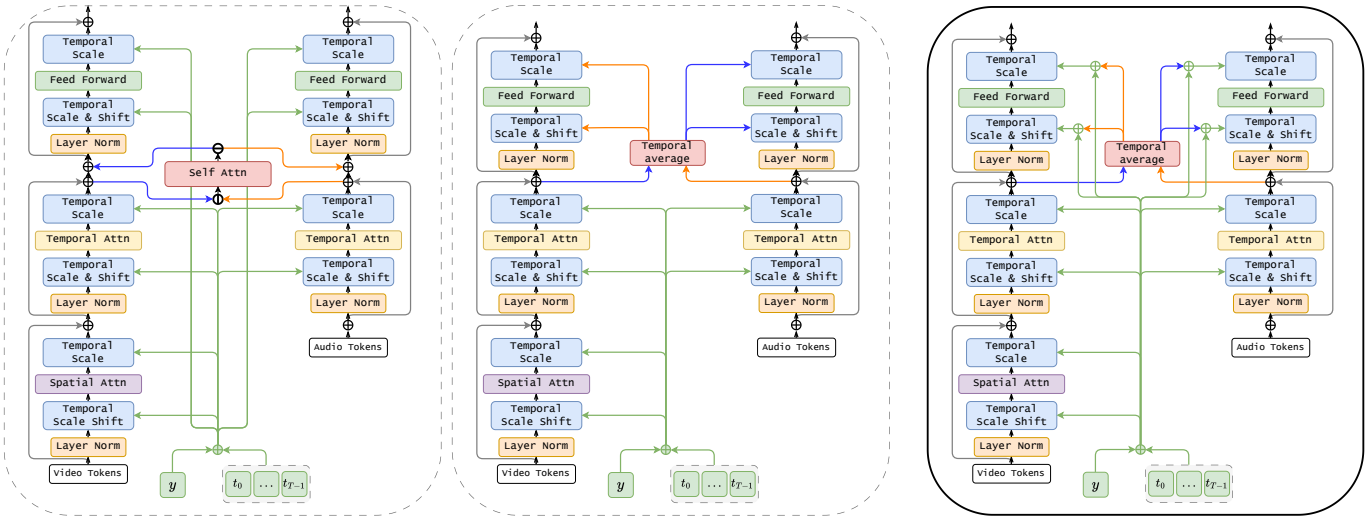

Fig. 5: a) Cross-modal interaction via self-attention, where $\oplus$ and $\ominus$ mean concatenation and split. b) Lightweight cross-modality interaction mechanism with modality average modulation. c) Our final proposed TECMO block, an enhanced lightweight mechanism incorporating timestep embedding t and optional class conditioning embedding c.

Beyond quantitative metrics, we conducted an extensive user study to both compare our best model with current state-of-the-art methods, and evaluate the different design choices. We recruited 20 participants, who were asked to rate 45 videos using a 5-point scale ranging from 1 (poor) to 5 (excellent). Each participant answered the following three questions:

- How would you rate the overall video quality?
- How would you rate the overall audio quality?
- How would you rate the audio–video synchronization?

D. Ablation Study

In Table 1.A, we present a comparison of all our proposed blocks on the AIST++ dataset and a baseline model, which does not apply any modality interaction. All metrics are calculated using 20 denoising steps. Notably, all architectures incorporating interaction modules outperform the baseline, demonstrating the importance of cross-modal interaction for this task. Among the proposed variants, block (c) achieves the best overall performance, surpassing both blocks (a) and (b). Compared to block (b), our architecture achieves a more effective propagation of the timestep embedding, leading to improved performance in the diffusion process. Additionally, compared to block (a), our proposed block is more efficient and faster, requiring fewer computational resources and less memory. Importantly, blocks (b) and (c) maintain computational efficiency comparable to the baseline with identical memory consumption, achieving superior performance without additional computational cost. In addition, in Table 1.B we also test a window size of 5 frames, 20, and 30 frames. Following this experiment, we observe that 10 frames is the optimal solution. The reason is that a smaller window (e.g., 5 frames) may not capture enough temporal context, leading to incomplete motion patterns, while a larger window can introduce redundant information and increase noise. The 10-frame window achieves best performance across all metrics while maintaining excellent efficiency, requiring only 3.94 seconds and 2.7 GB of memory, making it the most practical choice for our architecture.

E. Comparison with SOTA models

We compare our model with the current SOTA models [10], [12], [13], performing both a quantitative and qualitative analysis. For open source models, such as [10], we recompute the metrics on our hardware, and, vice versa, for closed source models ([12], [13]), we took the metrics directly from the original papers. The reason why, when possible, we choose to calculate the metrics from the SOTA models again is that, as pointed out in [37], the way images are resized and compressed can have a large impact on the common evaluation metrics of generative models. Therefore, by reproducing the same settings for all the open-source generative models we compare with, we ensure a fair comparison. Unfortunately, this cannot be done for closed-source models for obvious reasons. For the sake of a fair comparison, re-implementing the methods is not a viable solution since both *(i)* fine-grained technical details are not provided in the related papers and exactly reproducing the implementation is almost impossible and *(ii)* would likely lead to inconsistent results. Thus, we opted for reporting the numbers in the original paper.

a) Quantitative analysis.: The results of the SOTA models [10], [13] are shown in Table. II.

They show that our proposed model equipped with the enhanced AdaLN block (block (c)) with 20 steps surpasses all the SOTA models in both datasets except for the FAD score in Landscape where MMD is slightly better. Notably, thanks to the lightweight image encoder, our model has less total parameters compared to others SOTA models. We also evaluated our model with 100 and 200 steps, further improving

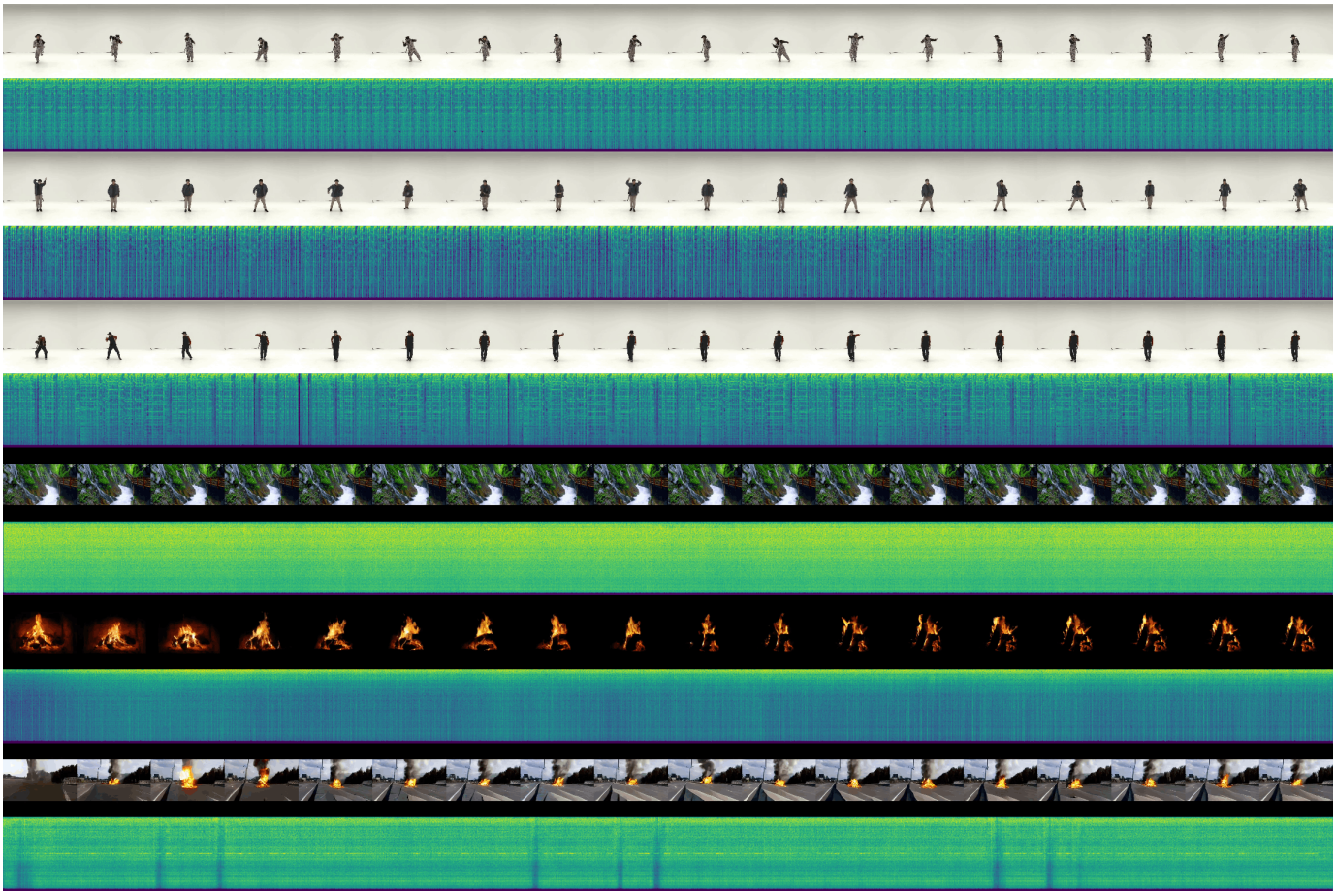

Fig. 6: 3 minutes long videos generated by our model on both AIST++ and landscape. Frames are sampled every 10s for visualization.

AIST++									
			Quantitative Metrics				Mean Opinion Score		
Model	Time (s)	Memory Usage (GB)	FVD ↓	KVD ↓	FAD ↓	CAVP ↑	Video Quality	Audio Quality	Audio-Video Sync
Baseline	3.91	2.7	81.65	17.02	10.43	0.76	2.46	2.56	2.60
TECMO-(a)	15.47	7.9	53.38	**7.33**	8.70	0.79	2.27	3.40	2.76
TECMO-(b)	3.92	2.7	51.31	9.76	**8.30**	0.82	2.71	3.60	3.16
TECMO-(c)	3.94	2.7	**50.92**	8.73	8.40	0.82	**3.51**	**4.01**	**4.12**

TABLE 1.A: Comparison between inference time for a sample of 16 frames, memory usage, quantitative metrics and mean opinion score of the user study of all the 3 proposed blocks and the baseline. Time was calculated on a NVIDIA RTX 4090 GPU.

AIST++									
				Quantitative Metrics			Mean Opinion Score		
Model	Window Size	Time (s)	Memory Usage (GB)	FVD ↓	KVD ↓	FAD ↓	Video Quality	Audio Quality	Audio-Video Sync
TECMO-(c)	5	3.36	2.5	65.42	10.41	8.46	3.16	3.67	3.51
TECMO-(c)	10	3.94	2.7	**50.92**	**8.73**	**8.40**	**3.51**	**4.01**	**4.12**
TECMO-(c)	20	5.51	3.3	77.55	12.71	8.53	2.60	3.22	3.02
TECMO-(c)	30	10.12	3.8	137.54	27.57	8.70	2.48	3.01	2.98

TABLE 1.B: Comparison of TECMO-(c) with different window size.

the quality of the generated results. Furthermore, compared to MM-Diffusion, our model achieves near-perfect audio-video alignment as measured by CAVP on both datasets, further demonstrating the effectiveness of our temporal alignment module.

Additionally, even if it is not the main objective of the proposed approach, our method can also perform Audio2Video (A2V) and Video2Audio (V2A) generation, in which original audio is used to generate a video and vice versa. Results for this task are presented in Table III. In this case we compare

			AIST++				Landscape			
Model	Params	Steps	FVD ↓	KVD ↓	FAD ↓	CAVP ↑	FVD ↓	KVD ↓	FAD ↓	CAVP ↑
GT	-	-	3.67	-0.28	8.25	0.83	7.45	-0.15	9.33	0.82
MMD [10]	426 M	25	224.14	50.52	11.41	0.79	177.29	7.63	9.73	0.80
Wang *et al.** [13]	931 M	250	68.88	21.01	10.17	-	172.69	15.41	11.17	-
TECMO	421 M	20	50.92	8.73	8.40	**0.82**	86.53	3.36	10.49	**0.82**
TECMO	421 M	100	38.93	6.58	8.35	**0.82**	85.44	3.73	**9.61**	**0.82**
TECMO	421 M	200	**38.36**	**6.15**	**8.28**	**0.82**	**80.19**	**3.30**	9.88	**0.82**

TABLE II: A comparison between our method and the current SOTA models, calculated with 2048 samples at 64×64 resolution. *Metrics of the model are taken from the paper since **neither code nor checkpoints are available.**

with AVDiT by Kim *et al.* [12] that presented results only for the A2V and V2A tasks and not for fully joint AV generation for both AIST++ and Landscape. Since neither code nor checkpoints are currently available, we took their FVD and KVD from A2V generation and FAD from V2A generation and compared them with our results for these tasks.

In order to correctly interpret the results in Table III, several factors should be considered. Kim *et al.* employ a different, closed-source audio and image/video encoders, which renders the FAD and FVD/KVD scores across the two works not directly comparable, even when evaluated on real (GT) data. A more reliable strategy is to examine the relative gap between metrics computed on generated and GT data. As reported in the table, the relative FAD gaps are very similar for both methods, indicating comparable performance. Smaller relative gaps in FVD/KVD are observed for Kim *et al.*, although these image-based metrics are known to be sensitive to additional factors such as compression and resizing, which introduces further uncertainty.

			AIST++			Landscape		
			A2V		V2A	A2V		V2A
Model	Params	Steps	FVD ↓	KVD ↓	FAD ↓	FVD ↓	KVD ↓	FAD ↓
Kim *et al.*	731 M	200	11.72 - 38.04	0.96 - 5.27	0.90 - 1.11	16.41 - 86.79	-0.25 - 4.30	0.76 - 0.78
TECMO	421 M	200	3.67 - 46.81	-0.28 - 8.72	8.25 - 8.52	7.45 - 94.47	-0.15 - 4.00	9.33 - 9.34

TABLE III: Our model compared with Kim *et al.*[12] in A2V and V2A generation. GT-Gen. metrics for A2V/V2A *Metrics of the model are taken from the paper since **neither code nor checkpoints are available.**

Finally, Table. IV reports the mean opinion scores from the user study, comparing our best model with Kim *et al.* and MM Diffusion [10]. Since the model of Kim *et al.* is closed-source and does not allow the generation of new samples, we perform the comparison using videos downloaded from their official project page. As shown in the table, our method achieves the highest mean opinion scores across all evaluation criteria.

b) High Resolution Results Analysis.: A quantitative evaluation of the 256×256 version of TECMO is reported in Table V. We report a comparison with MM Diffusion [10], as it is the only other method that provides results at this resolution. Other SOTA approaches, such as Wang *et al.* [13] and Kim *et al.* [12], cannot be retrained at higher resolutions because their code is not publicly available. Nevertheless, even under this more challenging setting, our model delivers consistent, high-quality results and outperforms MM Diffusion.

	Mean Opinion Score		
Models	Video Quality	Audio Quality	Audio-Video Sync
MMD [10]	2.91	3.58	3.40
Kim *et al.*[12]	3.25	3.51	3.69
Ours	**3.51**	**4.01**	**4.12**

TABLE IV: User study to evaluate the overall quality and the AV align between our models and the current state-of-the-art models.

c) Results on Long Video Generation.: To provide a quantitative evaluation of the video quality in the long run, we compute the metrics of Sect. IV-C on 3 minutes-long sequences, of which some examples are shown in Fig. 6, using a sliding window of 16 frames. The average results for 2048 videos are shown in Fig. 7. The figure unexpectedly reveals quite a big jump in the metrics after the first evaluation window. We argue that this is caused by the rolling diffusion design and, particularly, when switching from the pre-rolling to the full rolling phase. Indeed, during the pre-rolling phase, each frame $k < T$ is conditioned only on the previous $k-1$ frames in the sequence, due to temporal attention. This makes the generative process easier since the model needs to take into account less conditioning information. On the contrary, in the full rolling phase, at each diffusion step a fully denoised frame exits the window, while a new one (pure noise) gets in, meaning that each new frame will be conditioned by all the previous frames in the window, equal to $T-1$. Overall, this seems to lead to a slight shift in the generated data distribution, even though no perceivable changes are observed

To validate this hypothesis, we calculated the distance between features extracted from the first window (16 frames) of the sequence, and all the other windows. To extract the features we used the same network used to calculate the FVD [34]. The idea is that, if the above conjecture holds, then we will observe a strong initial drift in the encoded features. From the results (purple line in Fig. 7), we indeed observe a large drift after the first chunk. After that though, the drift increment becomes minimal, suggesting that after the transitory phase from pre-rolling to full rolling, the features get stabilized. The same behavior is observed for all the other metrics. In particular, FVD initially increases across the first several windows before plateauing. In contrast, KVD and FAD

		AIST			Landscape		
Model	Steps	FVD ↓	KVD ↓	FAD ↓	FVD ↓	KVD ↓	FAD ↓
MM Diffusion [10]	25	162.84	30.76	11.90	192.93	9.89	9.85
TECMO-(c)	20	**62.05**	**10.00**	**8.39**	**136.00**	**6.66**	10.64
TECMO-(c)	200	**59.14**	**9.34**	**8.34**	**129.14**	**6.33**	**9.80**

TABLE V: Comparison against MM Diffusion at 256×256 resolution on both AIST++ and Landscape

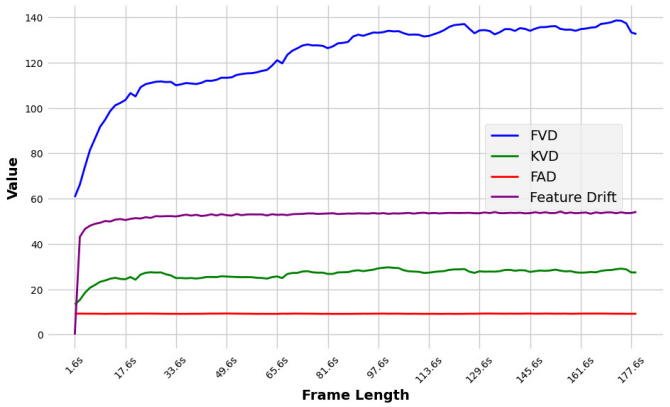

Fig. 7: AV metrics and feature drift calculated on 3 minutes long generated videos using a sliding window of 16 frames.

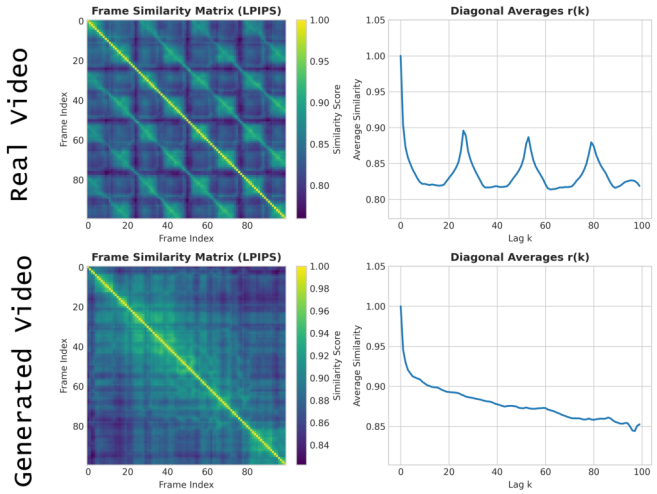

Fig. 8: Frame similarity matrix of real vs. generated videos from the AIST++ dataset, illustrating the presence of loops in real videos and their absence in generated content.

exhibit greater stability throughout the video duration, with KVD showing only a slight upward trend and FAD remaining largely constant. The increase in FVD can be attributed to imperceptible noise accumulation that does not affect human perception, whereas KVD demonstrates greater robustness to this phenomenon.

d) Detecting Looping Sequences in Videos.: One may wonder whether the generated long videos are just a looped repetition of shorter sequences which would hinder our claim of long AV generation. To clearly show this is not the case and our model effectively learned varying motion dynamics, we designed a specific test. The idea is that if a video contains looped sub-sequences, then similar frames will appear at periodic timesteps. To verify that, we compute a pairwise frame-to-frame similarity using the Learned Perceptual Image Patch Similarity (LPIPS) metric [38] on both real and generated videos from the AIST++ dataset, which features complex motion dynamics. From these, we build a pairwise similarity matrix where the (i, j) entry measures the LPIPS between the frames i and j. To quantify looping behavior, we then compute the average of off-diagonal similarity values, denoted as $r(k)$, where k represents the frame offset or the time lag. Peaks in $r(k)$ values denote a looping sequence of k frames *i.e.* a high similarity peak in $r(k)$ means that a general frame i closely resembles the frame $(i + k)$, indicating a repetitive pattern of period k.

Fig. 8 illustrates the process for two videos, real and generated. On the left are the two similarity matrices and on the right the $r(k)$ calculated from them. It is evident from the similarity matrix that the real video presents a peculiar diagonal pattern, which is a clear hint of a loop. This can be verified by observing the AIST++ videos which depict people performing repetitive dance moves. The real video presents 3 peaks in $r(k)$, denoting a looping sequence with a length of $\sim$ 23 frames. On the contrary, the analyzed generated video does not show such behavior.

For the sake of completeness, we analyze the whole training split of AIST++, which contains 980 videos, with an average duration of 10 seconds. Among them, we identified 692 videos containing looping sequences. To do so, first we compute the Fourier transform of $r(k)$. Then, in frequency domain, we can identify dominant frequency components and selecting the videos for which the largest component (indicating a loop) is above a fixed threshold. In contrast, when looking at 980 synthetic videos of the same length generated by our model, only 264 exhibited looping behavior. This demonstrates that our model can generate extended video sequences without repetitive loops.

F. Limitations

A limitation of our method is represented by objects that remain occluded for periods exceeding our temporal window size. This cause them to possibly fail to reappear correctly when they become visible again. For example, scene elements like rocks or terrain features can disappear permanently after temporary occlusion rather than reappearing as expected. This occurs because our sliding window approach, while enabling longer generation than existing methods, cannot maintain indefinite memory of occluded content. Note that other methods cannot be affected by this new problem as they generate only

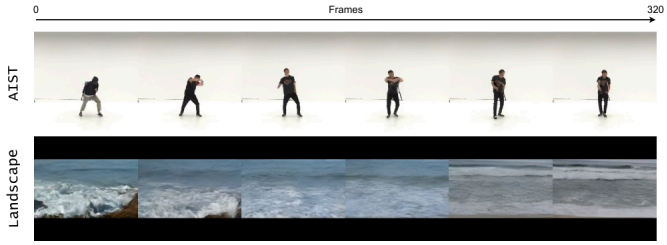

Fig. 9: An example of long-term consistency loss on AIST++ and Landscape

very short (1.6 s) videos. Examples of this limitation are reported in Fig. 9.

V. Conclusion

In this paper, we propose TECMO, a novel architecture based on flow matching that takes advantage of a rolling diffusion approach to generate coherent and infinite video samples with synchronized audio tracks and unfixed duration. Moreover, we designed a novel temporal fusion module that enables efficient AV generation, while only requiring a limited amount of frames during training. Through a series of experiments and ablation studies on two common datasets, we demonstrate, both qualitatively and quantitatively, our improvements over state-of-the-art methods.

Despite the substantial contribution of the proposed architecture w.r.t. the quality and length of the generated AV sequences, when dealing with such a task, a series of technical challenges remain to be faced. In particular, when trained on AIST++, our model occasionally struggles to generate complex limb movements. Furthermore, when generating long videos, it may sometimes fail to preserve time consistency when some visual elements are obscured beyond the model processing window. In AIST++, this happens in scenarios with partial occlusion or unconventional movements, such as intricate dance sequences. For example, a dancer's arm could cover the logo on their t-shirt for several frames and therefore the model will forget about it. On the other side, in Landscape, in sea scenarios, the water could obscure for an extended duration visual elements that will be then forgotten by the model. This reveals areas for future algorithmic refinement.

VI. Acknowledgments

This work was funded by "Partenariato FAIR (Future Artificial Intelligence Research) - PE00000013, CUP J33C22002830006" funded by the European Union - NextGenerationEU through the italian MUR within NRRP, project DL-MIG.

References

[1] W. Weng, R. Feng, Y. Wang, Q. Dai, C. Wang, D. Yin, Z. Zhao, K. Qiu, J. Bao, Y. Yuan et al., "Art-v: Auto-regressive text-to-video generation with diffusion models," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 7395–7405.

[2] D. Weissenborn, O. Täckström, and J. Uszkoreit, "Scaling autoregressive video models," in 8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020. OpenReview.net, 2020. [Online]. Available: https://openreview.net/forum?id=rJgsskrFwH

[3] D. Ruhe, J. Heek, T. Salimans, and E. Hoogeboom, "Rolling diffusion models," in Proceedings of the 41st International Conference on Machine Learning, ser. ICML'24. JMLR.org, 2024, pp. 42 818–42 835.

[4] Z. Evans, J. D. Parker, C. Carr, Z. Zukowski, J. Taylor, and J. Pons, "Stable audio open," in ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2025, pp. 1–5.

[5] H. Liu, Z. Chen, Y. Yuan, X. Mei, X. Liu, D. Mandic, W. Wang, and M. D. Plumbley, "Audioldm: Text-to-audio generation with latent diffusion models," in International Conference on Machine Learning. PMLR, 2023, pp. 21 450–21 474.

[6] J. Xue, Y. Deng, Y. Gao, and Y. Li, "Auffusion: Leveraging the power of diffusion and large language models for text-to-audio generation," IEEE/ACM Transactions on Audio, Speech, and Language Processing, 2024.

[7] J. Hai, Y. Xu, H. Zhang, C. Li, H. Wang, M. Elhilali, and D. Yu, "Ezaudio: Enhancing text-to-audio generation with efficient diffusion transformer," CoRR, 2024.

[8] Z. Fei, M. Fan, C. Yu, and J. Huang, "FLUX that plays music," CoRR, vol. abs/2409.00587, 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2409.00587

[9] S. Lee, C. Kong, D. Jeon, and N. Kwak, "Aadiff: Audio-aligned video synthesis with text-to-image diffusion," CoRR, vol. abs/2305.04001, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2305.04001

[10] L. Ruan, Y. Ma, H. Yang, H. He, B. Liu, J. Fu, N. J. Yuan, Q. Jin, and B. Guo, "MM-Diffusion: Learning multi-modal diffusion models for joint audio and video generation," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 10 219–10 228.

[11] W. Peebles and S. Xie, "Scalable diffusion models with transformers," in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 4195–4205.

[12] G. Kim, A. Martinez, Y.-C. Su, B. Jou, J. Lezama, A. Gupta, L. Yu, L. Jiang, A. Jansen, J. Walker et al., "A versatile diffusion transformer with mixture of noise levels for audiovisual generation," Advances in Neural Information Processing Systems, vol. 37, pp. 11 837–11 865, 2024.

[13] K. Wang, S. Deng, J. Shi, D. Hatzinakos, and Y. Tian, "Av-dit: Efficient audio-visual diffusion transformer for joint audio and video generation," in Audio Imagination: NeurIPS 2024 Workshop AI-Driven Speech, Music, and Sound Generation.

[14] A. Blattmann, T. Dockhorn, S. Kulal, D. Mendelevitch, M. Kilian, D. Lorenz, Y. Levi, Z. English, V. Voleti, A. Letts et al., "Stable video diffusion: Scaling latent video diffusion models to large datasets," CoRR, 2023.

[15] Z. Kong, W. Ping, J. Huang, K. Zhao, and B. Catanzaro, "Diffwave: A versatile diffusion model for audio synthesis," in International Conference on Learning Representations.

[16] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2022, pp. 10 684–10 695.

[17] F. Bao, S. Nie, K. Xue, Y. Cao, C. Li, H. Su, and J. Zhu, "All are worth words: A vit backbone for diffusion models," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2023, pp. 22 669–22 679.

[18] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," Advances in neural information processing systems, vol. 33, pp. 6840–6851, 2020.

[19] Y. Gong, Y.-A. Chung, and J. Glass, "Ast: Audio spectrogram transformer," Interspeech 2021, 2021.

[20] S. Ge, T. Hayes, H. Yang, X. Yin, G. Pang, D. Jacobs, J.-B. Huang, and D. Parikh, "Long video generation with time-agnostic vqgan and time-sensitive transformer," in European Conference on Computer Vision. Springer, 2022, pp. 102–118.

[21] D. Ghosal, N. Majumder, A. Mehrish, and S. Poria, "Text-to-audio generation using instruction guided latent diffusion model," in Proceedings of the 31st ACM International Conference on Multimedia, 2023, pp. 3590–3598.

[22] Z. Liu, S. Wang, S. Inoue, Q. Bai, and H. Li, "Autoregressive diffusion transformer for text-to-speech synthesis," CoRR, vol. abs/2406.05551, 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2406.05551

[23] Y. Xing, Y. He, Z. Tian, X. Wang, and Q. Chen, "Seeing and hearing: Open-domain visual-audio generation with diffusion latent aligners," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 7151–7161.

[24] R. Yang, H. Gamper, and S. Braun, "Cmmd: Contrastive multi-modal diffusion for video-audio conditional modeling," in European Conference on Computer Vision. Springer, 2024, pp. 214–226.
[25] Z. Tang, Z. Yang, C. Zhu, M. Zeng, and M. Bansal, "Any-to-any generation via composable diffusion," Advances in Neural Information Processing Systems, vol. 36, pp. 16 083–16 099, 2023.
[26] Y. Li, Y. Luo, and B. Du, "Audio-visual generalized zero-shot learning based on variational information bottleneck," in 2023 IEEE International Conference on Multimedia and Expo (ICME), 2023, pp. 450–455.
[27] S. Yang and B. Chen, "Effective surrogate gradient learning with high-order information bottleneck for spike-based machine intelligence," IEEE Transactions on Neural Networks and Learning Systems, vol. 36, no. 1, pp. 1734–1748, 2025.
[28] S. Yang, B. Linares-Barranco, Y. Wu, and B. Chen, "Self-supervised high-order information bottleneck learning of spiking neural network for robust event-based optical flow estimation," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 47, no. 4, pp. 2280–2297, 2025.
[29] Y. Lipman, R. T. Q. Chen, H. Ben-Hamu, M. Nickel, and M. Le, "Flow matching for generative modeling," in The Eleventh International Conference on Learning Representations, 2023. [Online]. Available: https://openreview.net/forum?id=PqvMRDCJT9t
[30] X. Liu, C. Gong et al., "Flow straight and fast: Learning to generate and transfer data with rectified flow," in The Eleventh International Conference on Learning Representations.
[31] P. Esser, S. Kulal, A. Blattmann, R. Entezari, J. Müller, H. Saini, Y. Levi, D. Lorenz, A. Sauer, F. Boesel et al., "Scaling rectified flow transformers for high-resolution image synthesis," in Forty-first International Conference on Machine Learning, 2024.
[32] R. Li, S. Yang, D. A. Ross, and A. Kanazawa, "Ai choreographer: Music conditioned 3d dance generation with aist++," in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 13 401–13 412.
[33] S. Tsuchida, S. Fukayama, M. Hamasaki, and M. Goto, "Aist dance video database: Multi-genre, multi-dancer, and multi-camera database for dance information processing." in ISMIR, vol. 1, no. 5, 2019, p. 6.
[34] J. Carreira and A. Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset," in proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 6299–6308.
[35] A. Guzhov, F. Raue, J. Hees, and A. Dengel, "Audioclip: Extending clip to image, text and audio," in ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2022, pp. 976–980.
[36] S. Luo, C. Yan, C. Hu, and H. Zhao, "Diff-foley: Synchronized video-to-audio synthesis with latent diffusion models," Advances in Neural Information Processing Systems, vol. 36, pp. 48 855–48 876, 2023.
[37] G. Parmar, R. Zhang, and J.-Y. Zhu, "On aliased resizing and surprising subtleties in gan evaluation," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 11 410–11 420.
[38] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018. Computer Vision Foundation / IEEE Computer Society, 2018, pp. 586–595. [Online]. Available: http://openaccess.thecvf.com/content_cvpr_2018/html/Zhang_The_Unreasonable_Effectiveness_CVPR_2018_paper.html

Alex Ergasti earned a bachelor's degree in Information Engineering from Politecnico di Milano in 2021 and a master's degree in ICT from the University of Parma in 2023. He currently is a PhD student at IMPLab under the supervision of Prof. Massimo Bertozzi, focusing his research on multimodal deep learning algorithms. He has authored several papers on these topics.

Giuseppe Tarollo earned the master's degree in Computer Science from the University of Parma in 2024. Later, he worked at IMPLab laboratory in the same university as a research assistant. His work focused on deep learning and generative models for audio-video generation.

Filippo Botti Received the B.Sc. and M.Sc. degrees (both cum laude) in Computer Engineering from the University of Parma in 2021 and 2023, respectively. He is currently a Ph.D. student in Information Technology at the University of Parma, working under the supervision of Prof. Andrea Prati. His research focuses on efficient deep learning architectures and style transfer, and he has authored several papers in these areas.

Tomaso Fontanini received the Ph.D. degree in information engineering from the University of Parma, in 2021, under the supervision of Prof. Andrea Prati. In 2020, he was a Visiting Student with the VisLAB Laboratory, University of California at Riverside (UCR), under the supervision of Prof. Bir Bhanu. He is currently an Assistant Professor with the Department of Engineering and Architecture, University of Parma. His current research interests include image generation and manipulation of facial attributes and styles in an unsupervised setting. In the past, he worked on applying meta-learning techniques to generative adversarial networks and image retrieval. He has co-authored several publications about these topics in journals and international conferences.

Claudio Ferrari is currently tenure-track assistant professor at the Department of Information Engineering of the University of Florence. He received the M.Sc. degree (cum laude) in Computer Engineering, and the Ph.D. degree in Information Engineering from the University of Florence in 2014 and 2018, respectively. In 2014, he spent 6 months as visiting research scholar at the IRIS Computer Vision Lab of the University of Southern California (USC). His research activity is focused on computer vision for human understanding and generation, 3D/4D face and body modeling and generation, and multi-modal generative models. On the topics, he authored and co-authored more than 60 publications in international journals and conference proceedings. He currently also serves as Associate Editor for IEEE Transactions on Circuits and Systems for Video Technology.

Massimo Bertozzi received the Dr.Eng. degree in Electronic Engineering from the University of Parma discussing a Master Thesis about the implementation of Simulation of Petri Nets on the CM-2 Massive Parallel Architecture. From November 1994 to October 1997 he was a PhD student in information technology at the Dipartimento di Ingegneria dell'Informazione of the Università di Parma where he chaired the local IEEE student branch. Since November 1997 he has held a permanent position and he is now associate professor at the Department of Engineering and architecture. His research interests mainly focus on the application of image processing to real-time systems and specifically to autonomous driving, on the optimization of machine code at assembly level, and on parallel and distributed computing, use of CNN for autonomous driving and industrial inspection. From 2015 until 2020 he worked at VisLab on the development of SW and System Integration for the Ambarella CVflow chip architecture. Since 2020 he is also focusing his research interests to CNN in cooperation within the IMP Lab research group.

Andrea Prati (Senior Member, IEEE) received the Graduate degree in computer engineering and the Ph.D. degree from the University of Modena and Reggio Emilia, in 1998 and 2002, respectively. He was an Assistant Professor, from 2005 to 2011, and has been an Associate Professor with the Iuav University of Venice, since 2015. In December 2015, he moved to the University of Parma and got promoted to full professorship, in 2019. He is currently the Head of the IMP Laboratory, research group. His research interests include computer vision and image processing, deep learning, and generative models. He is the author of nine book chapters, more than 40 articles in international refereed journals, and more than 100 papers in proceedings of international conferences and workshops. To date, his H-index on Google Scholar is 46, with a total of 10 780 citations. On Scopus, his H-index is 33, with a total of 5937 citations. He is a fellow of IAPR and a member of CVPL.