Continuous Autoregressive Generation with Mixture of Gaussians

Anonymous Authors[1]

Abstract

Autoregressive sequence models have traditionally relied on discrete tokenizations to leverage cross-entropy training, but this discretization introduces information loss that is costly in high-dimensional domains such as video. Utilizing higher capacity tokens enables higher quality generations, allowing one to use less tokens to represent a single image, and thus improve training and inference time. We propose a continuous-token autoregressive framework that parameterizes each step's output distribution as a mixture of Gaussians. A lightweight Mixture of Gaussians (MoG) head predicts mixture weights, means, and full covariance factors, and is trained end-to-end by minimizing the Gaussian negative log-likelihood of continuous latent tokens. We demonstrate our approach on conditional video generation from a single image, comparing against a discrete-token and a continuous "mu-only" baseline. Our model achieves the best Frechet Video Distance (FVD), and generates frames with greater temporal diversity, as measured by SSIM components, but with a modest cost to FID.

1. Introduction

Most autoregressive models are trained using discrete tokens, primarily due to the simplicity and effectiveness of cross-entropy loss. However, discrete representations tend to be more lossy than continuous ones due to information bottlenecks. This limitation becomes particularly pronounced in video generation, where even a single frame is often represented by a large number of tokens. Leveraging continuous tokens can improve image quality and reduce the number of required tokens, leading to faster generation.

Diffusion-based models benefit from continuous tokenization, offering higher fidelity representations. Yet, they come with trade-offs, such as limited ability to generate beyond the training context and, in some cases, slower inference speeds.

In this paper, we explore an alternative approach: applying continuous tokenization within a decoder-only autoregressive framework. Specifically, we introduce a method that uses a mixture of Gaussians to parameterize sampled continuous tokens. We introduce two variants of sampling from these Mixture of Gaussians and compare their generations.

2. Methods

Notation. Let $\mathbf{x}_{1:T} = \{\mathbf{x}_1, \dots, \mathbf{x}_T\}$ be a length-T sequence of continuous tokens, $\mathbf{x}_t \in \mathbb{R}^D$. A decoder-only backbone with parameters θ produces

$$\mathbf{z}_t = f_\theta(\mathbf{x}_{<t}) \in \mathbb{R}^{d_z}, \quad t = 1, \dots, T.$$

2.1. Mixture-of-Gaussians output head

From each $\mathbf{z}_t$ we predict:

$$\boldsymbol{\pi}_t = \operatorname{softmax}\big(W_\pi \mathbf{z}_t + b_\pi\big) \qquad \in \Delta^{K-1}, \tag{1}$$

$$\boldsymbol{\mu}_t = \operatorname{reshape}\big(W_\mu \mathbf{z}_t + b_\mu,\ K, D\big), \tag{2}$$

$$\boldsymbol{\ell}_t = \operatorname{reshape}\Big(W_\Sigma \mathbf{z}_t + b_\Sigma,\ K,\ \tfrac{D(D+1)}{2}\Big). \tag{3}$$

Here $\ell_{t,k} \in \mathbb{R}^{D(D+1)/2}$ is unpacked into a lower-triangular matrix

$$U_{t,k} \ = \ \operatorname{vec2tril}\big(\ell_{t,k}\big) \ \in \mathbb{R}^{D \times D},$$

and then turned into a valid Cholesky factor by applying $\operatorname{softplus}$ to the diagonal:

$$L_{t,k} = \operatorname{tril}\big(U_{t,k}\big) \ + \ \operatorname{diag}\Big(\operatorname{softplus}\big(\operatorname{diag}(U_{t,k})\big)\Big),$$

$$\Sigma_{t,k} = L_{t,k}\, L_{t,k}^\top.$$

The conditional density is therefore

$$p_\phi(\mathbf{x}_t \mid \mathbf{z}_t) = \sum_{k=1}^{K} \pi_{t,k}\, \mathcal{N}\big(\mathbf{x}_t;\, \mu_{t,k},\, \Sigma_{t,k}\big),$$

with $\phi = \{W_\pi, b_\pi, W_\mu, b_\mu, W_\Sigma, b_\Sigma\}$.

2.2. Training objective

As before, we minimise the Gaussian NLL,

$$\mathcal{L}(\theta, \phi) = -\sum_{t=1}^{T} \log p_\phi\big(\mathbf{x}_t \mid f_\theta(\mathbf{x}_{<t})\big),$$

back-propagating through both backbone and mixture head.

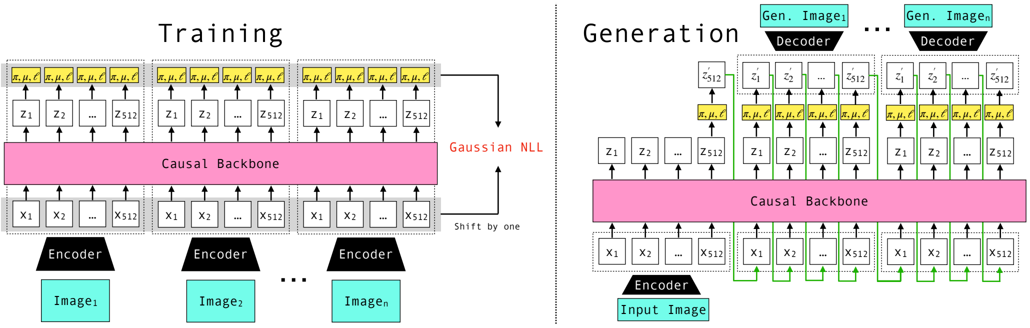

Figure 1. (left) Training Pipeline supervising video generation with Gaussian NLL. π, μ, ℓ are projections of z_t. (right) Generation with prefill on first image, then autoregressive generation for remaining tokens. $z_t^{'}$ is sampled using the Mixture of Gaussian parameters with either the weighted-average or hard-sampling variants.

2.3. Autoregressive inference

We define two variants for autoregressive inference. Both start with the generation of the mixture parameters:

1. Compute $\mathbf{z}_t = f_\theta(\widehat{\mathbf{x}}_{<t})$.
2. Form $\{\pi_{t,k}, \mu_{t,k}, L_{t,k}\}_{k=1}^{K}$ as above.

MoG (Weighted-Average) Inference. We sample a single standard-Gaussian noise vector once, then form each component's sample using that same noise, and finally take a π-weighted average. At each generation step t:

3. Sample a shared noise vector $\epsilon \sim \mathcal{N}(\mathbf{0}, I)$.
4. Compute the weighted-average sample $\widehat{\mathbf{x}}_t = \sum_{k=1}^{K} \pi_{t,k} \left(\mu_{t,k} + L_{t,k}\,\epsilon\right)$, where each $L_{t,k}$ is the lower-triangular Cholesky factor satisfying $\Sigma_{t,k} = L_{t,k}L_{t,k}^{\top}$.

MoG-Hard (Single-Component) Inference. Here we hard sample a discrete component first, then draw from that Gaussian component. At each generation step t:

3. Sample component $k \sim \mathrm{Categorical}(\pi_t)$.
4. Sample $\widehat{\mathbf{x}}_t \sim \mathcal{N}(\mu_{t,k}, L_{t,k}L_{t,k}^{\top})$.

2.4. Experimental Setup

We experiment with a conditional generation task to generate a video from a single image input. The 560M trainable parameter model is composed of a frozen Cosmos discrete or continuous image (16x16) tokenizers (NVIDIA et al., 2025) with a trainable hybrid Mamba/Transformer backbone and Mixture of Gaussians head. We train on publicly available driving videos, and evaluate on a held-out validation set. Models are trained for 600k steps on 8xH100s with batch size 1. Checkpoints are taken every 100k steps, the checkpoint with the best FVD score is taken. Input videos are resized to 25 frames of 512x256 resolution and are encoded by the Cosmos tokenizers, giving 512 tokens per image for a total context size of 12,800. Training videos are 25 frames long and the evaluation task generates 24 frames given a single frame.

3. Experiments

We compare our Mixture of Gaussians autoregressive model with weighted-average sampling (MoG) and with Hard sampling (MoG-Hard), defined in section 2.3, against a discrete baseline trained with cross entropy and to a continuous baseline that simply predicts the next continuous latent (equivalent to predicting a single Gaussian μ with $\sigma = 0$) with a MSE loss. Our Mixture of Gaussians are trained with K=2 components.

3.1. Metrics

Frechet Video Distance (FVD) (Unterthiner et al., 2019) compares the distribution of generated videos to the distribution of the validation set videos, giving a metric for video quality. We use VideoMAE (Wang et al., 2023) to calculate FVD as (Ge et al., 2024) finds the computed features less content biased compared to I3D features. Similarly, Frechet Inception Distance (FID) (Heusel et al., 2018) applied to each generated image and averaged across the temporal axis gives a metric for individual-frame image quality. We quantify the diversity of frames by calculating the Structural Similarity Index Measure (SSIM) (Wang et al., 2004) between consecutive frames, as well as the Luminance, Contrast, and Structure components of SSIM. Evaluations are performed with n=36 videos generated, with the same starting images across all the models.

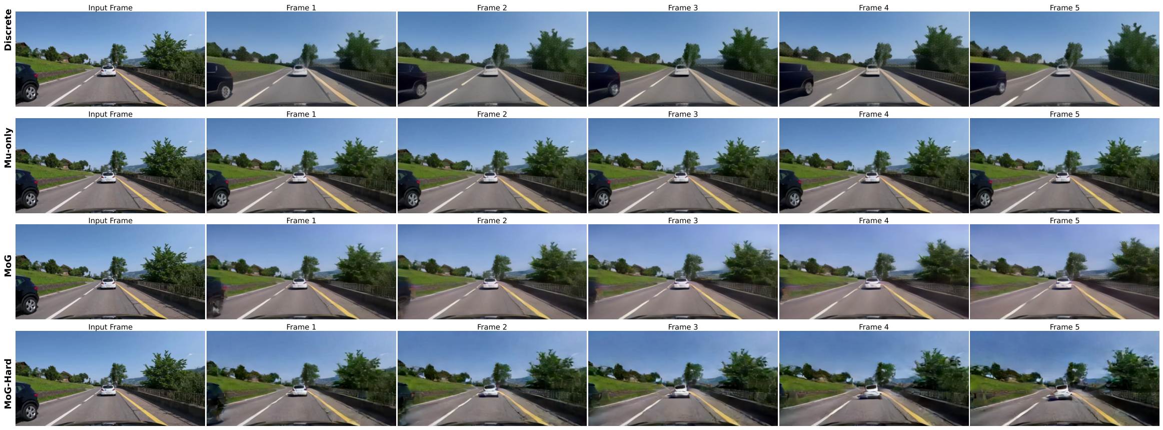

Figure 2. Video roll-outs along the x-axis (with the input frame first). Continuous generations contain more fine details than the discrete ones. The Mu-only generations have virtually no motion, the Discrete and MoG generations have evident motion, while the MoG-Hard generations have much more motion but also distortion. This sample shows that the discrete model generates less realistic videos with the black car going backwards.

3.2. Mixture of Gaussians

Visual Fidelity Consistent with intuition, we find that continuous generations capture higher fine-details from the original input frame, compared to the discrete baseline. In Figure 2, we observe qualitatively that Mu-only, MoG, and MoG-Hard generate more detailed frames by virtue of using a continuous tokenizer. Table 1 quantitatively supports this as Mu-only also achieves the highest FID. In MoG and MoG-Hard, we see a worse FID score because of compounding distortion in the long-horizon.

Motion and Temporal Diversity Despite degrading FID over frame position, MoG delivers the best FVD out of the models (Table 1), likely a result of the VideoMAE feature extractor penalizing minimal motion videos in the FVD calculation. Motionless videos and cars driving backwards seen in Figure 2 are not very realistic, so worse FVD scores for Discrete and Mu-only generations is consistent with evaluating video realism.

Furthermore, we find that the Mu-only generations have the least diversity of frames as indicated by a high SSIM score (Table 1). The Discrete and MoG models have lower SSIM scores indicating more diversity, however not to the level of the validation reference set of real videos. While the MoG model has the most variation, even more so than the reference set. To control for the luminance and contrast changes, we factor out the individual components of SSIM and find that the structure-component of SSIM follows this analysis similar to the overall SSIM aggregate metric. The MoG-Hard generations most closely approximate the frame-by-frame structural diversity of the validation reference set, however still lacks in the visual consistency.

Long-Horizon Compounding Error Our proposed Mixture of Gaussians training pipeline still faces long-horizon instability in generation, often finding itself out-of-distribution with compounding errors over the 12,800 token context window. It is possible that adding recovery trajectories and training longer may alleviate this issue, especially since the continuous domain may be more numerically sensitive and complex than the discrete domain.

Method	FVD	FID	SSIM	Lum.	Cont.	Struc.
Discrete	385	211	0.944	0.993	0.981	0.965
Mu-only	894	**205**	0.992	0.998	0.998	0.996
MoG	**324**	251	0.956	0.998	0.987	0.970
MoGHard	359	235	0.808	0.974	0.952	0.861
Val Ref.			0.862	0.989	0.964	0.892

Table 1. (n=36). FVD captures video realism via extracted feature similarity. FID captures image realism. SSIM across consecutive frames aggregates changes in Luminance, Contrast, and Structure. Mu-only has the best image realism, due to its continuous tokenizer and minimal motion. MoG has the best video realism, despite its lacking image realism in the later part of generations, but makes up for it with dynamic motion across frames.

3.3. Analysis of Gaussian Components

With these empirical benefits of the MoG approach, we now investigate how the mixture of gaussians may be representing the generative signals. In Figure 3, for the MoG model Component 2 retains a majority of the weight at around 75%, with some flunctuation across token position, while Component 1 hovers at around 25%. We observe that the Mu values are very tightly correlated between Component 1 and 2, although Component 1 has significantly higher variance from the covariance diagonal (graphs in Appendix A).

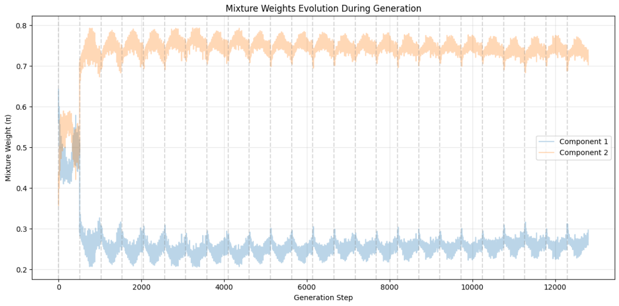

Figure 3. Mixture Weights (π) averaged across generation steps for MoG model (n=36). Dotted lines indicate boundaries between frames. Positions 0 to 512 are the prefill for the input image, and the remaining positions are the generation. Component 2 dominates the mixture weight, averaging about 0.75 of the weight while Component 1 averages about 0.25 of the weight.

Since the components do not collapse to represent the same distribution, this suggests there is some useful representational power to using mixture of Gaussians over a single Gaussian. However, Table 3 in Appendix A shows that forcing a generation to only use a single Gaussian component recovers similar or better FVD, FID, and SSIM metrics. More evaluations need to be done to determine to what extent multi-modality and diverse generation is enabled by multiple Gaussian components. Nonetheless, the Mixture of Gaussian training procedure is necessary for stable training dynamics as single Gaussian training is unstable.

4. Ablations

Mixture vs Single In ablation experiments, we find that training single gaussians are unstable in the long-context NLL setting, with loss NaNs within 50k steps for both the Multivariate and Univariate settings.

Multivariate vs Univariate Gaussians The Cosmos continuous tokenizer in this work has 16 dimensions, so we experiment whether modeling tokens in 16 dimensional space (and thus 16-dimensional Gaussians) differs from modeling the Gaussians as independent for each dimension. Ideal tokenizers may have independent channels for maximal information representation, but we find multivariate gaussian representations outperform univariate Gaussians when using the Cosmos tokenizer.

4.1. Mu-only + Fixed Sigma

We further evaluate Mu-only sampling with augmented variants using the learned mu component and naively sample with a nonzero sigma at generation time. Testing with $\sigma = \{e^{-5}, e^{-3}\}$, we find that although diversity increases with lower SSIM when forcing a fixed sigma, generations still do not have coherent motion and are largely restricted to luminance and contrast changes. Although MoG has worse FID than Mu-only with Fixed sigma, MoG improves on FVD and motion realism, demonstrating the value of learned covariance parameters (Figure 2).

Method	FVD	FID	SSIM	Lum.	Cont.	Struc.
$\sigma = 0$	894	205	0.992	0.998	0.998	0.996
$\sigma = e^{-5}$	894	**204**	0.992	0.998	0.998	0.996
$\sigma = e^{-3}$	674	214	0.984	0.996	0.996	0.992
MoG	**324**	251	0.956	0.998	0.987	0.970
MoGHard	359	235	0.808	0.974	0.952	0.861
Val Ref.			0.862	0.989	0.964	0.892

Table 2. (n=36). Increasing levels of fixed sigma with a Mu-only trained model improves FVD, but still falls short of MoG, showing the utility of learning covariance.

5. Related Works

Our work builds off of a rich history of work in both generative modeling of videos as well as explicit distribution modeling using neural networks.

Video Generation: High quality video generation often relies on diffusion-based models (Ho et al., 2022b;a). Due to the success of large scale next-token predictive LLMs, discrete-token autoregressive (AR) models using vision tokenizers has also emerged (Yan et al., 2021; Yu et al., 2023; Van Den Oord et al., 2017). Furthermore, beyond next-token prediction alone, coarse-to-fine or multi-scale generation strategies offer enhanced long-horizon coherence and sampling speed (Tian et al., 2024; Deng et al., 2024), marking a shift from diffusion to autoregression across image and video tasks. However, because vision is a very dense signal, discrete tokenization presents a significant loss of quality; thus recent approaches for AR modeling attempt to incorporate continuous latent spaces and eliminate quantization for improved fidelity and efficiency (Li et al., 2024; Agarwal et al., 2025) by combining AR with diffusion based losses, with some integrating causal and bidirectional frame modeling for speed and coherence (Deng et al., 2024).

Distribution Learning: Beyond diffusion losses for implicit modeling of distributions, a large body of work also exists for explicit parametric modeling of a distributions via maximum likelihood estimation. Classic mixture density networks parameterize predictions (Bishop, 1994), while AR models in multimodal domains predict structured distributions with better probabilistic modeling. The learned densities have been shown to be effective in modalities such as audio (Van Den Oord et al., 2016), pixels (Theis et al., 2012; Salimans et al., 2017), graphs (Errica et al., 2021), and robotic control (Amini et al., 2019).

References

Agarwal, N., Ali, A., Bala, M., Balaji, Y., Barker, E., Cai, T., Chattopadhyay, P., Chen, Y., Cui, Y., Ding, Y., et al. Cosmos world foundation model platform for physical ai. *arXiv preprint arXiv:2501.03575*, 2025.

Amini, A., Rosman, G., Karaman, S., and Rus, D. Variational end-to-end navigation and localization. In *2019 International Conference on Robotics and Automation (ICRA)*, pp. 8958–8964. IEEE, 2019.

Bishop, C. M. Mixture density networks. 1994.

Deng, H., Pan, T., Diao, H., Luo, Z., Cui, Y., Lu, H., Shan, S., Qi, Y., and Wang, X. Autoregressive video generation without vector quantization. *arXiv preprint arXiv:2412.14169*, 2024.

Errica, F., Bacciu, D., and Micheli, A. Graph mixture density networks. In *International Conference on Machine Learning*, pp. 3025–3035. PMLR, 2021.

Ge, S., Mahapatra, A., Parmar, G., Zhu, J.-Y., and Huang, J.-B. On the content bias in fréchet video distance, 2024. URL https://arxiv.org/abs/2404.12391.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. Gans trained by a two time-scale update rule converge to a local nash equilibrium, 2018. URL https://arxiv.org/abs/1706.08500.

Ho, J., Chan, W., Saharia, C., Whang, J., Gao, R., Gritsenko, A., Kingma, D. P., Poole, B., Norouzi, M., Fleet, D. J., and Salimans, T. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022a.

Ho, J., Salimans, T., Gritsenko, A., Chan, W., Norouzi, M., and Fleet, D. J. Video diffusion models. *Advances in Neural Information Processing Systems*, 35:8633–8646, 2022b.

Li, T., Tian, Y., Li, H., Deng, M., and He, K. Autoregressive image generation without vector quantization. *Advances in Neural Information Processing Systems*, 37:56424–56445, 2024.

NVIDIA, :, Agarwal, N., Ali, A., Bala, M., Balaji, Y., Barker, E., Cai, T., Chattopadhyay, P., Chen, Y., Cui, Y., Ding, Y., Dworakowski, D., Fan, J., Fenzi, M., Ferroni, F., Fidler, S., Fox, D., Ge, S., Ge, Y., Gu, J., Gururani, S., He, E., Huang, J., Huffman, J., Jannaty, P., Jin, J., Kim, S. W., Klár, G., Lam, G., Lan, S., Leal-Taixe, L., Li, A., Li, Z., Lin, C.-H., Lin, T.-Y., Ling, H., Liu, M.-Y., Liu, X., Luo, A., Ma, Q., Mao, H., Mo, K., Mousavian, A., Nah, S., Niverty, S., Page, D., Paschalidou, D., Patel, Z., Pavao, L., Ramezanali, M., Reda, F., Ren, X., Sabavat, V. R. N., Schmerling, E., Shi, S., Stefaniak, B., Tang, S., Tchapmi, L., Tredak, P., Tseng, W.-C., Varghese, J., Wang, H., Wang, H., Wang, H., Wang, T.-C., Wei, F., Wei, X., Wu, J. Z., Xu, J., Yang, W., Yen-Chen, L., Zeng, X., Zeng, Y., Zhang, J., Zhang, Q., Zhang, Y., Zhao, Q., and Zolkowski, A. Cosmos world foundation model platform for physical ai, 2025. URL https://arxiv.org/abs/2501.03575.

Salimans, T., Karpathy, A., Chen, X., and Kingma, D. P. Pixelcnn++: Improving the pixelcnn with discretized logistic mixture likelihood and other modifications. *arXiv preprint arXiv:1701.05517*, 2017.

Theis, L., Hosseini, R., and Bethge, M. Mixtures of conditional gaussian scale mixtures applied to multiscale image representations. 2012.

Tian, K., Jiang, Y., Yuan, Z., Peng, B., and Wang, L. Visual autoregressive modeling: Scalable image generation via next-scale prediction. *Advances in neural information processing systems*, 37:84839–84865, 2024.

Unterthiner, T., van Steenkiste, S., Kurach, K., Marinier, R., Michalski, M., and Gelly, S. Towards accurate generative models of video: A new metric challenges, 2019. URL https://arxiv.org/abs/1812.01717.

Van Den Oord, A., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., Kalchbrenner, N., Senior, A., Kavukcuoglu, K., et al. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*, 12, 2016.

Van Den Oord, A., Vinyals, O., et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

Wang, L., Huang, B., Zhao, Z., Tong, Z., He, Y., Wang, Y., Wang, Y., and Qiao, Y. Videomae v2: Scaling video masked autoencoders with dual masking, 2023. URL https://arxiv.org/abs/2303.16727.

Wang, Z., Bovik, A., Sheikh, H., and Simoncelli, E. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13 (4):600–612, 2004. doi: 10.1109/TIP.2003.819861.

Yan, W., Zhang, Y., Abbeel, P., and Srinivas, A. Videogpt: Video generation using vq-vae and transformers. *arXiv preprint arXiv:2104.10157*, 2021.

Yu, L., Cheng, Y., Sohn, K., Lezama, J., Zhang, H., Chang, H., Hauptmann, A. G., Yang, M.-H., Hao, Y., Essa, I., et al. Magvit: Masked generative video transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10459–10469, 2023.

A. Gaussian Components Analysis

Method	FVD	FID	SSIM	Lum.	Cont.	Struc.
MoG	324	**251**	0.956	0.998	0.987	0.970
MoG Force Component 1	**308**	254	0.951	0.997	0.985	0.967
MoG Force Component 2	322	**251**	0.956	0.998	0.987	0.910

Table 3. (n=36). Forcing Gaussian Components uncovers similar performance.

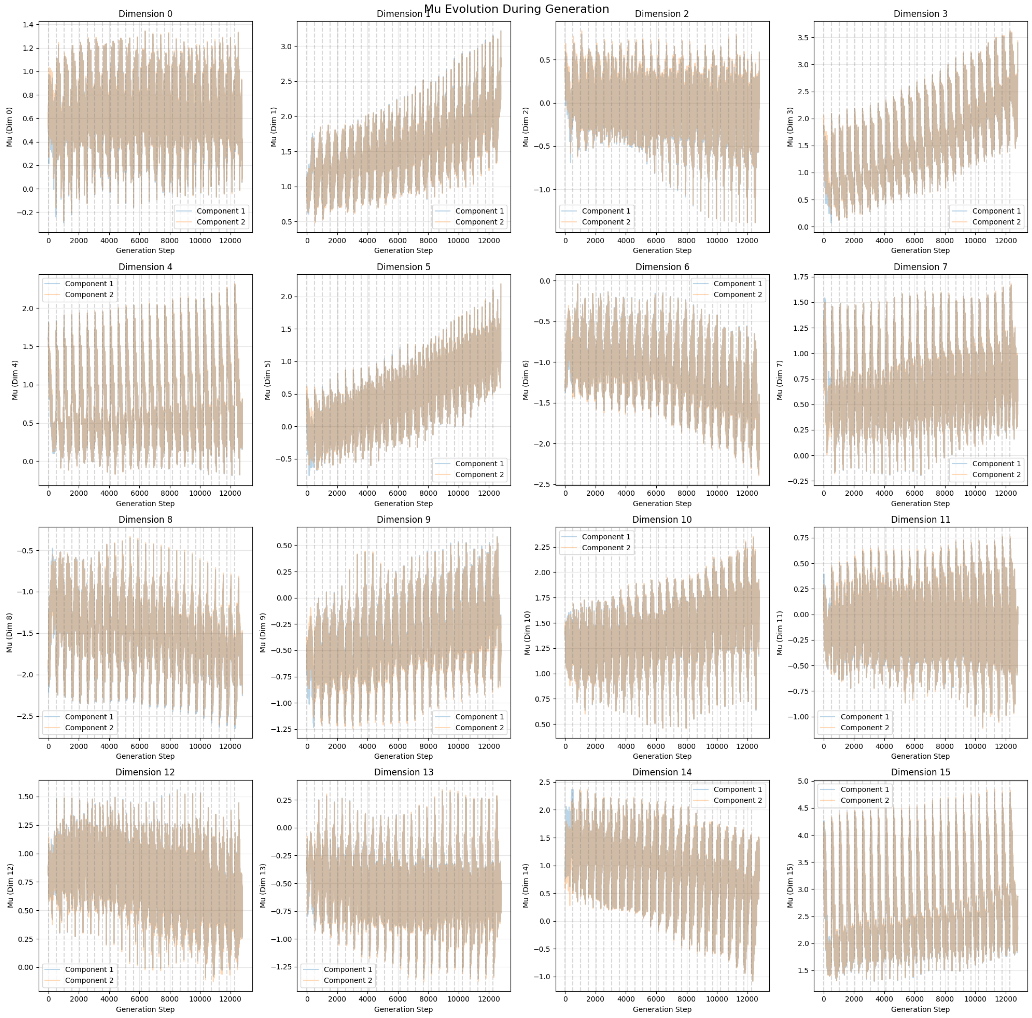

Figure 4. Mu averaged across generation steps (n=36). Dotted lines indicate boundaries between frames. Positions 0 to 512 are the prefill for the input image, and the remaining positions are the generation.

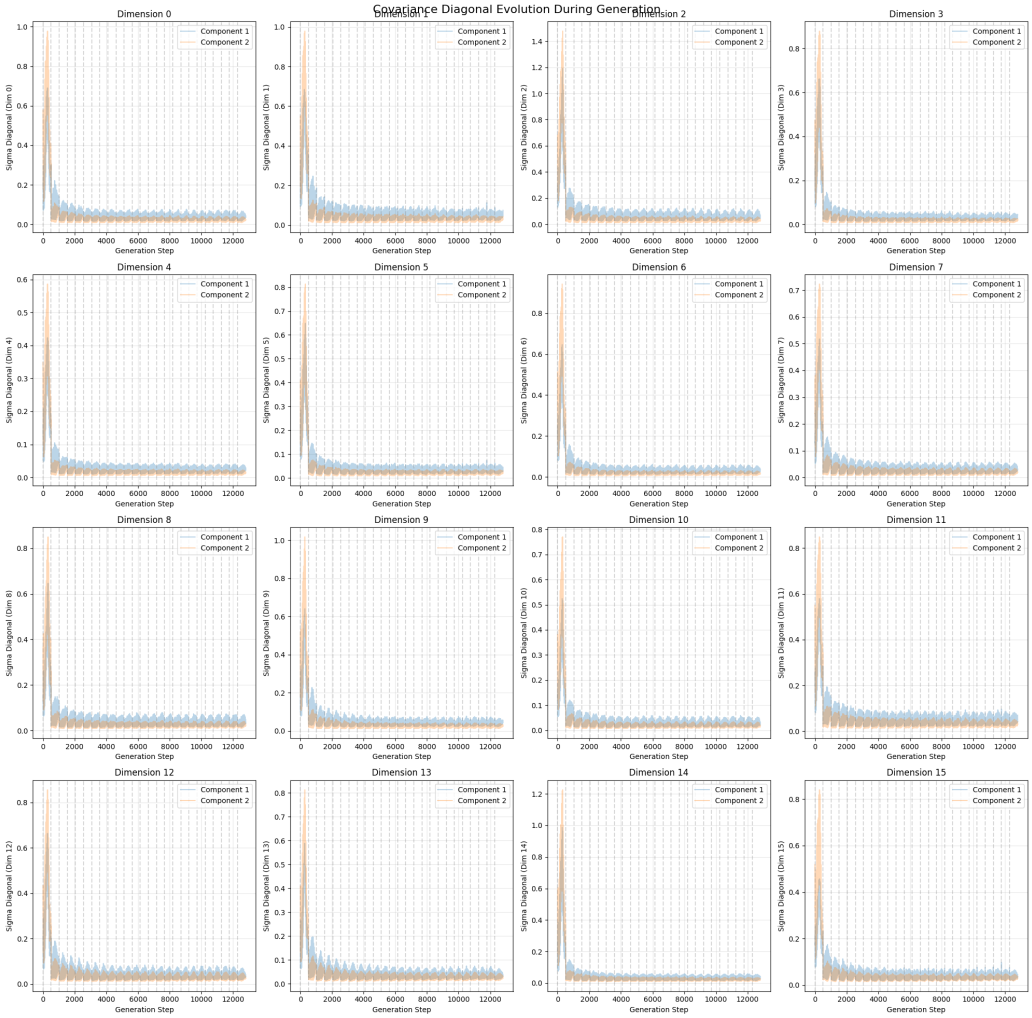

Figure 5. Mixture Weights (π) averaged across generation steps (n=36). Dotted lines indicate boundaries between frames. Positions 0 to 512 are the prefill for the input image, and the remaining positions are the generation.

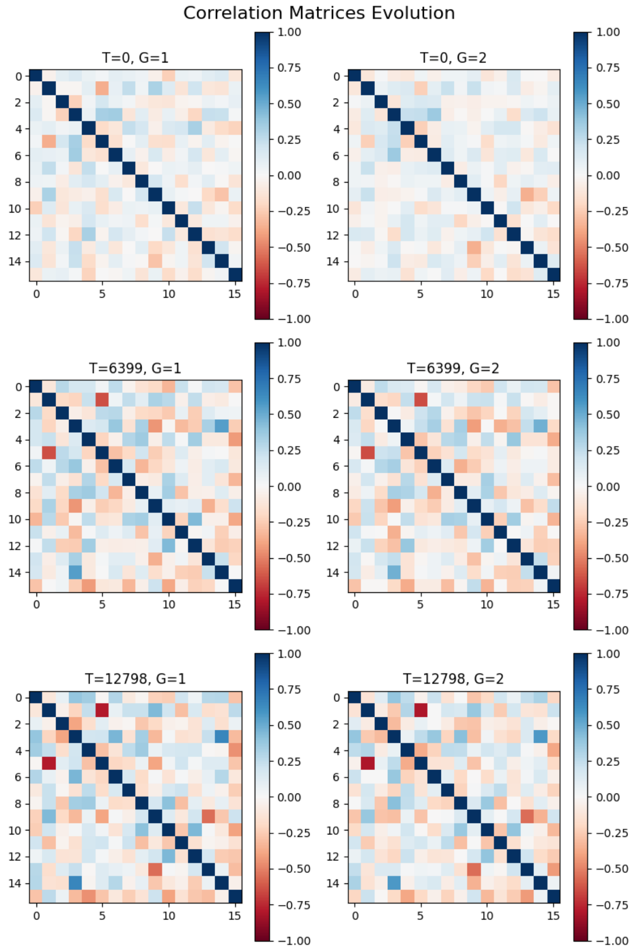

Figure 6. Mixture Weights (π) averaged across generation steps (n=36). Dotted lines indicate boundaries between frames. Positions 0 to 512 are the prefill for the input image, and the remaining positions are the generation.