Scaling Sequence-to-Sequence Generative Neural Rendering

Shikun Liu[1,*], **Kam Woh Ng**[1,*], **Wonbong Jang**[1], **Jiadong Guo**[1], **Junlin Han**[1], **Haozhe Liu**[1], **Yiannis Douratsos**[1], **Juan C. Pérez**[1], **Zijian Zhou**[1], **Chi Phung**[1], **Tao Xiang**[1], **Juan-Manuel Pérez-Rúa**[1]

[1]Meta AI
*Core Contributors

We present Kaleido, a family of generative models designed for photorealistic, unified object- and scene-level neural rendering. Kaleido operates on the principle that 3D can be regarded as a specialised sub-domain of video, expressed purely as a sequence-to-sequence image synthesis task. Through a systemic study of scaling sequence-to-sequence generative neural rendering, we introduce key architectural innovations that enable our model to: i) perform generative view synthesis without explicit 3D representations; ii) generate any number of 6-DoF target views conditioned on any number of reference views via a masked autoregressive framework; and iii) seamlessly unify 3D and video modelling within a single decoder-only rectified flow transformer. Within this unified framework, Kaleido leverages large-scale video data for pre-training, which significantly improves spatial consistency and reduces reliance on scarce, camera-labelled 3D datasets — all without any architectural modifications. Kaleido sets a new state-of-the-art on a range of view synthesis benchmarks. Its zero-shot performance substantially outperforms other generative methods in few-view settings, and, for the first time, matches the quality of per-scene optimisation methods in many-view settings.

Date: October 5, 2025
Correspondence: shikun@meta.com (model design, training), kamwohng@meta.com (data rendering, inference)
Blogpost: https://shikun.io/projects/kaleido

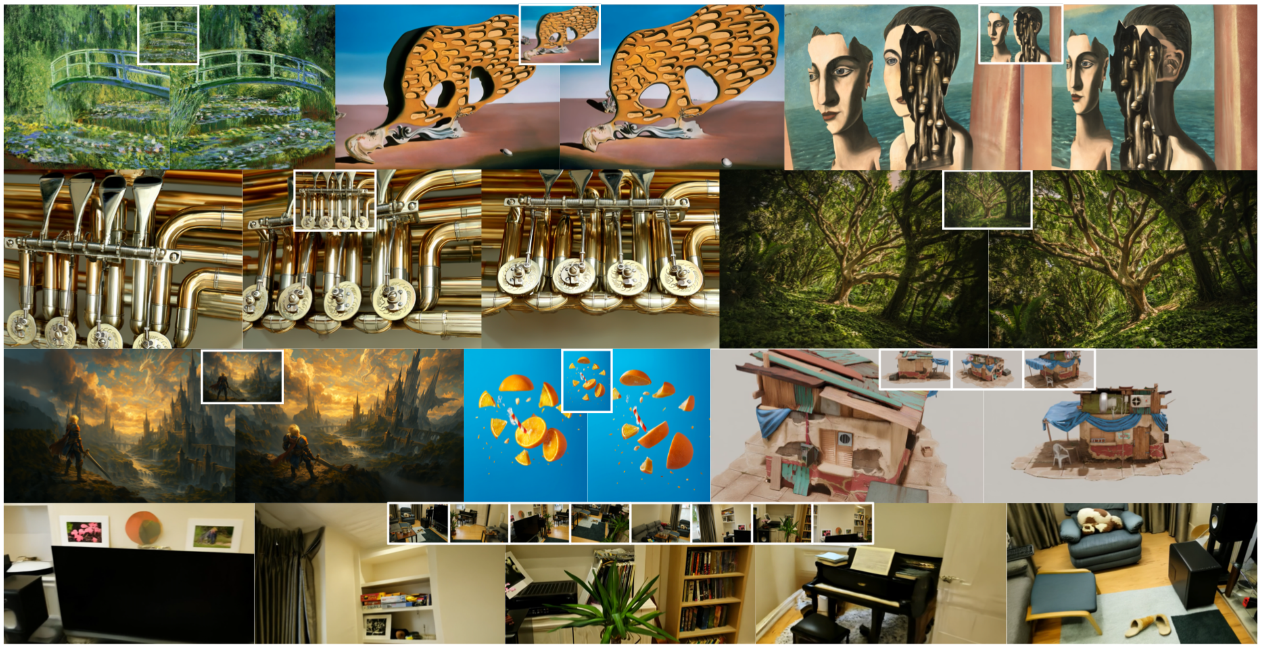


Figure 1 *Kaleido* is a generative rendering engine that can synthesise any number of photorealistic novel views across diverse artistic styles from any number of reference images (white boxes) with arbitrary 6-DoF camera poses.

1 Introduction

Rendering and view synthesis are foundational to 3D computer vision and graphics, driving applications across virtual reality, cinematic effects, robotics and autonomous driving. By allowing a scene to be rendered from arbitrary viewpoints based on a limited set of reference views, view synthesis mimics the adaptability of human vision — the ability to construct and reconstruct a coherent 3D understanding of our surroundings.

While deep learning, fuelled by massive datasets and scalable architecture designs, has achieved remarkable success in language modelling and 2D vision, its progress in 3D vision for *general-purpose rendering* has been comparatively slow. We argue this stems from two persistent and interconnected bottlenecks:

1. *A Fragmented Landscape of 3D Representations.* 3D vision lacks a consensus on the *right 3D representation*, with methods spanning explicit structures like voxels (Wu et al., 2015) and point clouds (Qi et al., 2017; Guo et al., 2020) to implicit ones like neural fields (Mildenhall et al., 2020; Xie et al., 2022). This fragmentation has prevented the focused, collective effort required to scale a powerful architecture for any single representation, as development remains divided across incompatible data formats.
2. *The High Cost of 3D Data.* 3D datasets are scarce and difficult to obtain primarily because their creation is guided by the principle of *strict 3D consistency*. Achieving this level of precision requires either hand-crafting 3D synthetic object meshes (Deitke et al., 2023b,a) or employing bundle adjustment and global alignments (Hartley and Zisserman, 2003) for slow multi-view camera labelling, making the data acquisition process slow, costly, and fundamentally difficult to scale.

As a direct consequence of these challenges, the 3D vision community has yet to converge on a scalable paradigm for 3D modelling. The combination of fragmented research efforts and restrictive data requirements has prevented the kind of focused, large-scale investment that enabled the dramatic architectural scaling and performance gains seen in language and 2D vision.

We believe these limitations, taken together, point to a fundamental oversight:

3D perception is not a geometric problem, but a form of visual common sense.

The human ability to perceive 3D structure emerges from extensive observation of the world, not from maintaining a precise 3D model in the mind. For example, humans can interpret 3D geometry in optical illusions (*e.g.* in M.C. Escher's impossible structures and the Ponzo illusion), without having a physically accurate 3D representation or even a correct sense of depth. Accordingly, we argue that an ideal rendering system should not aim to *explicitly model perfect geometric consistency*, but instead to learn an *implicit representation* by capturing the statistical patterns of the extensive visual experience of the world.

Building on this insight, we introduce *Kaleido*, a scalable architecture for generative neural rendering. We design Kaleido as a type of *spatial generative model* that does not encode any explicit 3D structures. Instead, Kaleido inherits spatial perception and visual common sense directly from large-scale video data, purely in a data-driven way, similar to how modern large language models acquire textual common sense from large-scale corpora without relying on explicit linguistic rules. This leads to our central hypothesis, inspired by the success of domain-specific fine-tuning in pre-trained language models (e.g., coding in Roziere et al. (2023); Anil et al. (2023); Chen et al. (2021)), we believe that a powerful general-purpose rendering model can be created by treating *3D as a specialised sub-domain of video*. To put it simply,

$$
\begin{array}{lll}
 & \textit{We observed} & \textit{large-scale corporus data} \rightarrow \textit{structured code data} = \textit{a general-purpose coding model} \\
\Longrightarrow & \textit{We hypothesise} & \textit{large-scale video data} \rightarrow \textit{structured 3D data} = \textit{a general-purpose rendering model}
\end{array}
$$

To realise this hypothesis, we reformulate rendering as a sequence-to-sequence problem, specifically as a pose-conditioned, image-to-image synthesis task. We first establish *a unified, geometrically consistent representation of space and time* as the core of our model design. This is achieved with a positional encoding design that extends the original Rotary Positional Encoding (RoPE) (Su et al., 2021) to parametrise all 2D, 3D, and temporal positions *relatively*, within the dot-product self-attention of a transformer (Vaswani et al., 2017). This foundational design enables Kaleido to learn rich world representations from large-scale, unstructured video data and then perform efficient transfer learning with much smaller-scale, structured multi-view 3D data, all within the same model *without any task-specific architectural changes*.

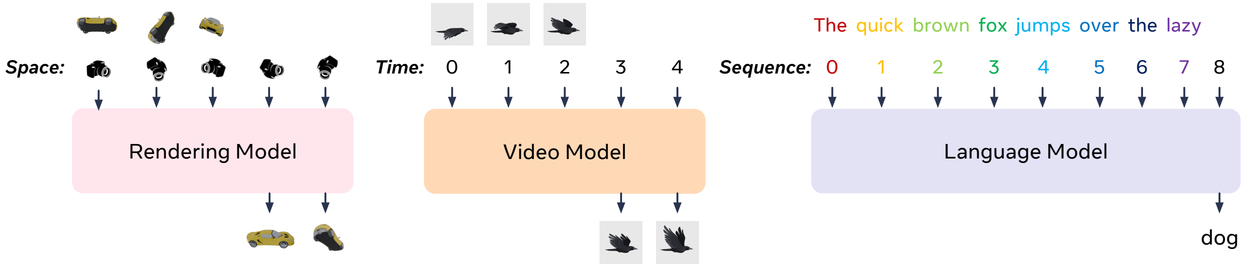

The quick brown fox jumps over the lazy
Space:
Time: 0 1 2 3 4
Sequence: 0 1 2 3 4 5 6 7 8
Rendering Model
Video Model
Language Model
dog

Figure 2 Rendering as Sequence-to-Sequence Image Modelling. We propose that neural rendering can be framed as a sequence-to-sequence task, unifying its design with language and video generation. In this formulation, a transformer (Vaswani et al., 2017) learns to generate image tokens conditioned on their spatial positions, similar to how language models condition on token positions in a sequence, and video models condition on temporal positions across frames.

Building on this unified representation, Kaleido naturally benefits from scalable architectures and powerful generative techniques developed for language and vision. Specifically, Kaleido adopts a scalable transformer architecture inspired by Diffusion Transformer (DiT) (Peebles and Xie, 2023) and Llama-3 (Dubey et al., 2024), performing generative modelling via a rectified flow objective (Liu et al., 2022; Esser et al., 2024) within a masked autoregressive framework (Li et al., 2024b; Fan et al., 2025; Liu et al., 2025a).

Finally, we identify that rectified flow SNR samplers commonly used for text-to-image/video generation are suboptimal for the precise pose conditioning required in rendering. We therefore introduce an improved, noise-biased sampling strategy and other key architectural adjustments to ensure stable and efficient scaling. Through extensive systemic studies, we validate these designs and highlight our primary contributions:

1. We introduce the *Kaleido family of Spatial Generative Models* (*SGMs*), which can perform unified object- and scene-level view synthesis from any number of reference views to any number of target views with full 6-DoF camera control. This is enabled by the following designs:
   (a) A simple decoder-only rectified flow transformer that considers generative rendering as a sequence-to-sequence task.
   (b) A unified positional encoding design that seamlessly processes both 3D and video data within a single, unchanged architecture.
   (c) An effective scaling recipe for both model size and resolution, supported with a tailored SNR sampler and solutions for training instability.
2. Kaleido generates high-resolution images (up to 1024px) across diverse aspect ratios, achieving state-of-the-art results on numerous view synthesis and 3D reconstruction benchmarks. Most notably, in many-view settings, Kaleido is the first zero-shot generative model to match the rendering quality of per-scene optimisation methods like Instant-NGP (Müller et al., 2022).

2 Related Work

From 2D to 3D and Camera Parameters. Reconstructing 3D geometry and camera parameters from 2D images is a foundational problem in computer vision. Classical approaches like Structure from Motion (SfM) (Hartley and Zisserman, 2003; Schönberger et al., 2016) and Simultaneous Localisation and Mapping (SLAM) (Davison et al., 2007; Mur-Artal et al., 2015; Izadi et al., 2011) have been highly successful, but they are limited by their need to optimise each scene from scratch and their struggles with non-overlapping views. More recently, learning-based methods have emerged to address these limitations. Models like DUSt3R (Wang et al., 2024) and VGGT (Wang et al., 2025) introduce feed-forward pointmap regression, enabling end-to-end 3D reconstruction that generalises across scenes. While these methods represent a significant step forward, their reliance on direct geometric regression means they cannot effectively infer content in occluded regions. Notably, Kaleido's fully generative design allows it to predict plausible, spatially consistent content for occluded regions, a key advantage over both classical and modern regression-based techniques.

Multi-View Stereo, Neural Rendering, and Novel View Synthesis. Traditional Multi-view Stereo (MVS) (Furukawa et al., 2015; Schönberger et al., 2016) reconstructs 3D surfaces by triangulating features across multiple viewpoints. This principle was revolutionised by Neural Radiance Fields (NeRF) (Mildenhall et al., 2020), which uses volume rendering and coordinate MLPs to achieve photorealistic novel view synthesis. A plethora of follow-up works have focused on improving the speed and quality of this per-scene optimisation paradigm (Müller et al., 2022; Fridovich-Keil et al., 2022; Chen et al., 2022; Kerbl et al., 2023). However, these methods require numerous, dense input views. To handle synthesis from only a few views, a learned prior is necessary. Early works in this area used category-specific priors and pre-trained image features (Sitzmann et al., 2019; Yu et al., 2021; Jang and Agapito, 2021), but a performance gap remained compared to scene-specific methods with dense views. More recently, feed-forward transformer-based models have emerged (Hong et al., 2023; Jang and Agapito, 2024; Jin et al., 2024), which can directly predict 3D primitives or render novel views from limited inputs. However, as deterministic models, they still fundamentally struggle with the inherently probabilistic nature of inferring large, occluded regions.

Generative 3D Modelling and View Synthesis Generative 3D modelling has rapidly evolved from synthesising isolated objects to composing entire, complex scenes. Pioneering text-to-3D works like Shap-E (Jun and Nichol, 2023) and Score Distillation Sampling (SDS) based methods like DreamFusion (Poole et al., 2022) laid the groundwork for single-object synthesis, inspiring a wave of research focused on high-fidelity object generation (Liang et al., 2024; Tang et al., 2024; Wang et al., 2023b; Shi et al., 2024). More recently, the frontier has expanded to scene generation, with approaches ranging from procedural construction (Sun et al., 2023; Raistrick et al., 2023) to direct compositional scene optimisation (Li et al., 2024a). A common thread in many of these works is the reliance on SDS to refine an explicit 3D representation.

Generative view synthesis models (Liu et al., 2023b,a,d; Shi et al., 2023) have emerged alongside this trend, but often face their own limitations. These methods typically struggle with multi-view consistency, are designed for a fixed number of reference (often one) and target views, and frequently rely on the same complex, two-stage SDS pipelines to enforce geometric coherence. Conversely, Kaleido's sequence-to-sequence design naturally handles an arbitrary number of both reference and target views, allowing it to generate spatially consistent views directly without requiring any post-processing or optimisation stages like SDS.

Sequence-to-Sequence Generative View Synthesis. Our work formulates generative view synthesis as a sequence-to-sequence modelling problem, built upon a pure transformer architecture. A critical challenge when applying transformers to this domain is effectively encoding camera positions. Recent advancements have introduced RoPE-style encodings (Su et al., 2021) to parameterise 6-DoF camera extrinsics, with notable examples including CaPE (Kong et al., 2024), GTA (Miyato et al., 2024), and also camera intrinsics in a more recent work (Li et al., 2025). Kaleido builds directly on this direction, leveraging a GTA-based framework to create a unified representation for both multi-view 3D poses and temporal video positions.

While other sequence-to-sequence methods like CAT3D (Gao et al., 2025), EscherNet (Kong et al., 2024) and SEVA (Zhou et al., 2025) have shown impressive results, their foundations lie in text-to-image latent diffusion models that use U-Net backbones. This reliance on a convolutional architecture is known to scale less effectively than pure transformers. Furthermore, these models often require 3D-specific learnable components, such as Plücker ray encodings for camera poses and a separate vision encoder for reference views. In contrast, Kaleido adheres to a pure transformer design from first principles, which results in a simpler, cleaner design that unifies 3D and video modelling, without any 3D-specific architectural modifications.

Generative Video and World Models Kaleido's methodology is deeply connected to recent advancements in generative video and the emerging paradigm of world models. The field of video generation has seen milestone progress with models like OpenAI's Sora (Brooks et al., 2024) and DeepMind's Veo (Deepmind, 2024), which have set a new standard for generative realism and temporal consistency. This progress is largely driven by a dominant technical stack combining diffusion or rectified flow models with transformer architectures, a foundation shared by many other generative video models (Blattmann et al., 2023; Chen et al., 2024b,a; Yang et al., 2025). While Kaleido is built upon this same foundation and is pre-trained on large-scale video data, its goal is not to be a standalone video generator. Instead, it leverages video pre-training specifically to build a robust world representation for high-fidelity generative rendering.

The increasing capabilities of video generation have also positioned it as a stepping stone towards building world models — systems that learn an internal model of the world to simulate physical interactions and predict future states. This trajectory is evident in models that focus on controllability and interactivity. For instance, the Navigation World Model (Bar et al., 2025) predicts future observations to facilitate planning, while frameworks like WonderWorld (Yu et al., 2025a) and GameFactory (Yu et al., 2025b) generate explorable 3D environments. Most notably, the Genie series (Bruce et al., 2024; Deepmind, 2025) creates interactive environments with persistent spatial memory and real-time promptable world events, marking a significant advance toward truly immersive and dynamic virtual worlds.

Kaleido contributes to this broader pursuit of world modelling from a different perspective. Instead of focusing on temporal dynamics or agent-based interactivity, Kaleido approaches world modelling through the lens of neural rendering, prioritising *spatial consistency* and *generation flexibility*. This unique approach allows it to operate across a spectrum of realities: with many reference views, it produces a grounded reality through faithful reconstruction; while with few views, it creates a generated reality with plausible unseen details. This unique capability to seamlessly transition between reconstruction and creative generation marks a distinct and intriguing path toward creating truly versatile and navigable virtual worlds.

3 Kaleido: Scaling Rectified Flow Transformers for Generative Rendering

3.1 Background and Notations

Kaleido considers rendering and video generation within a unified sequence-to-sequence framework. The goal is to estimate the conditional distribution of a set of target views given a set of reference views:

$$\mathcal{X}^T \sim p(\mathcal{X}^T | \mathcal{X}^R, \mathcal{P}^R, \mathcal{P}^T) \tag{1}$$

Here, the conditioning set consists of N reference views $\mathcal{X}^R = \{x^R_{i=1:N}\}$ and their corresponding positions $\mathcal{P}^R = \{P^R_{i=1:N}\}$. The target set consists of M target views $\mathcal{X}^T = \{x^T_{j=1:M}\}$ and their positions $\mathcal{P}^T = \{P^T_{j=1:M}\}$.

The positions P are defined flexibly depending on their data modality. For 3D data, each $P \in SE(3)$ represents a 6-DoF camera pose. For video data, each $P \in \mathbb{N}$ represents a temporal position (*i.e.,* a frame index).

This *"any-to-any view prediction"* can be seen as a form of *"next set-of-tokens prediction"*, which is elegantly handled by a masked auto-regressive framework (Li et al., 2024b). A key advantage of this approach is its flexibility: the number of reference views, N, and target views, M, can be arbitrary during both training and inference. This allows for various inference strategies, such as generating all target views at once or generating long sequences autoregressively by treating previously generated frames as new reference views. For training efficiency with batched optimisations, within each iteration, we sample a fixed total of V views, and choose N reference and M target views such that $N + M = V$.

Kaleido is a latent rectified flow model (Rombach et al., 2022; Ma et al., 2024; Esser et al., 2024) that operates on spatially compressed image tokens. We first use a pre-trained VAE (Kingma and Welling, 2014) (with an 8×8 compression rate and 16 latent channels) to encode all reference and target images into a latent space: $\{\mathcal{Z}^R, \mathcal{Z}^T\} = \mathcal{E}(\{\mathcal{X}^R, \mathcal{X}^T\})$.

Following the rectified flow formulation (Liu et al., 2023c; Lipman et al., 2023), we then construct a linear interpolation path between each target latent $z^T \in \mathcal{Z}^T$ (from the data distribution p_0) and a standard normal noise latent $\epsilon \sim \mathcal{N}(0, I)$ (from the noise distribution p_1):

$$\mathcal{Z}^T_t = (1-t)z^T + t\epsilon, \quad \text{where } t \in [0,1] \text{ and } \forall z^T \in \mathcal{Z}^T. \tag{2}$$

A vision transformer (ViT) (Dosovitskiy et al., 2020) then processes the combined sequence of clean reference latents $\mathcal{Z}^R$ and noised target latents $\mathcal{Z}^T_t$. We tokenise the latents using a patch size of 2×2 (for a combined spatial compression of 16×16), which we found provides an optimal trade-off between generation quality and inference speed. Kaleido is trained with a standard noise-prediction objective (Kingma and Gao, 2023), applied only to the target latents $\mathcal{Z}^T_t$, to estimate a velocity field between p_0 and p_1, conditioned as defined in Eq. 1. To analyse the scalability, we present three model variants: **Kaleido-Small**, **Kaleido-Medium**, and **Kaleido-Large (Kaleido)**, with detailed architectures and training strategies introduced next.

3.2 Kaleido Architecture Details and Training Strategies

In this section, we present a comprehensive ablation study of the Kaleido architecture design and its training strategies. Our goal is to identify the key design decisions that address the unique scaling challenges of sequence-to-sequence generative neural rendering. Our main findings are summarised in Fig. 3, with full quantitative results and explorations of alternative designs detailed in Appendix A. An overview of the final Kaleido architecture is shown in Fig. 4.

To provide a holistic view of our design process, we conducted a series of controlled experiments. For efficiency, we used our **Kaleido-Small** for all ablations, allowing for rapid iteration. We trained each experimental configuration on two distinct datasets: **Objaverse** (Deitke et al., 2023b), which contains synthetic objects with ground-truth camera poses, and **uCO3D** (Liu et al., 2025b), which contains real-world objects with noisy, estimated camera poses. Each experiment is trained for 100K optimisation steps on 8× H100 GPUs. We perform a greedy search over key design choices, organised by the four primary objectives introduced next.

3.2.1 Designing Kaledio's Design Spaces

We begin by exploring Kaleido's architectural design spaces and training strategies. Our Kaleido-Small's starting point is a vanilla DiT-L/SiT-L architecture (Peebles and Xie, 2023; Ma et al., 2024) within a rectified flow framework, whose scaling properties have been well established in image and video generation (Esser et al., 2024; Polyak et al., 2024; Chen et al., 2025).

(i) *Improved Architecture Design with Llama 3.* We first incorporate recent architectural advances from state-of-the-art sequence-to-sequence language models like Llama-3 (Dubey et al., 2024). Specifically, we replace the standard GLU activations in our transformer's feed-forward layers with **SwiGLU** (Shazeer, 2020) and swap multi-head attention (MHA) for the more efficient **grouped-query attention (GQA)** (Ainslie et al., 2023). These simple modifications yield consistent performance gains across our experiments without increasing computational overhead.

(ii) *Unified Positional Encoding for Space and Time* One of the critical design decisions in Kaleido is a unified positional encoding that seamlessly represents 2D, 3D, and temporal positions within a single, consistent design. Specifically, we introduce a parameter-free encoding scheme that extends the principles of **RoPE-style relative encodings** (Su et al., 2021) and **Geometric Transformation Attention (GTA)** (Miyato et al., 2024), which we adapt and generalise to create a unified representation for space and time. This design allows Kaleido to process both multi-view 3D and video data without any architectural modifications.

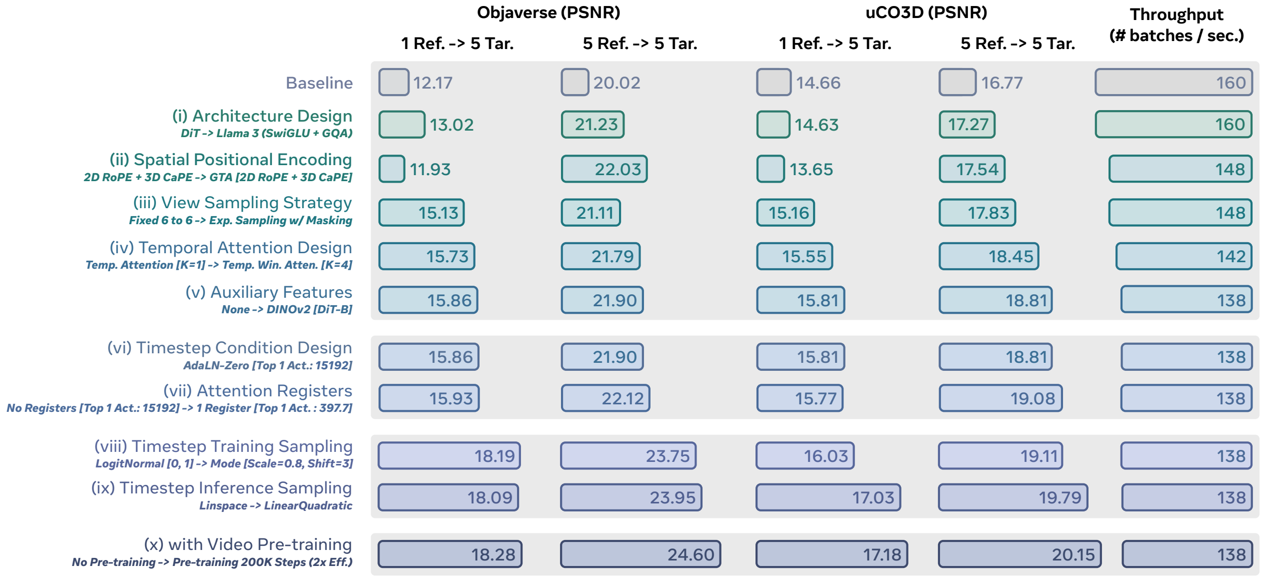
We represent different positions as follows: In 2D image positions, pixel coordinates are mapped to a pair of angles (θ_h, θ_w), representing an element in $SO(2) \times SO(2)$, where $\theta_{h,w} \in [0, 2\pi)$ distributed uniformly from the top-left to the bottom-right patches; In temporal positions, frame indices are similarly mapped to a single angle $\theta_t \in SO(2)$, with values interpolated linearly from the start to the end of a clip. In 3D camera poses, 6-DoF camera extrinsics $c = \left[\begin{smallmatrix}\mathbf{R} & \mathbf{t} \\ 0 & 1\end{smallmatrix}\right]$ (with rotation $\mathbf{R}$ and translation $\mathbf{t}$) are represented as an element in $SE(3)$, following the design in CaPE (Kong et al., 2024).

This allows us to define a unified geometric attribute g for each image token, depending on its data modality:

$$\text{For 3D data:} \quad g := (\theta_h, \theta_w, c) \in SO(2) \times SO(2) \times SE(3) \tag{3}$$

$$\text{For video data:} \quad g := (\theta_h, \theta_w, \theta_t) \in SO(2) \times SO(2) \times SO(2). \tag{4}$$

Within the GTA framework, these components are used to construct a block-diagonal transformation matrix P_g that is applied to each token's feature vector $v \in \mathbb{R}^d$. The construction of P_g varies for different attention blocks, allocating the feature dimension d as follows: In Spatial Attention, we apply only the 2D position embeddings (θ_h, θ_w), which are expanded into $d/4$ distinct frequency bands, with dimensions allocated to image height and width components based on a 1:1 ratio; In Temporal/3D Attention, the 2D embeddings (θ_h, θ_w) are expanded into $d/8$ frequency bands. For video data, the temporal embedding θ_t is expanded into $d/4$ frequency bands. For 3D data, the pose embedding c is repeated to fill the remaining dimensions. The total dimensions are allocated to image height, width, and temporal/3D components based on a 1:1:2 ratio.



	Objaverse (PSNR)		uCO3D (PSNR)		Throughput (# batches / sec.)
	1 Ref. -> 5 Tar.	5 Ref. -> 5 Tar.	1 Ref. -> 5 Tar.	5 Ref. -> 5 Tar.	
Baseline	12.17	20.02	14.66	16.77	160
(i) Architecture Design *DiT -> Llama 3 (SwiGLU + GQA)*	13.02	21.23	14.63	17.27	160
(ii) Spatial Positional Encoding *2D RoPE + 3D CaPE -> GTA [2D RoPE + 3D CaPE]*	11.93	22.03	13.65	17.54	148
(iii) View Sampling Strategy *Fixed 6 to 6 -> Exp. Sampling w/ Masking*	15.13	21.11	15.16	17.83	148
(iv) Temporal Attention Design *Temp. Attention [K=1] -> Temp. Win. Atten. [K=4]*	15.73	21.79	15.55	18.45	142
(v) Auxiliary Features *None -> DINOv2 [DiT-B]*	15.86	21.90	15.81	18.81	138
(vi) Timestep Condition Design *AdaLN-Zero [Top 1 Act.: 15192]*	15.86	21.90	15.81	18.81	138
(vii) Attention Registers *No Registers [Top 1 Act.: 15192] -> 1 Register [Top 1 Act. : 397.7]*	15.93	22.12	15.77	19.08	138
(viii) Timestep Training Sampling *LogitNormal [0, 1] -> Mode [Scale=0.8, Shift=3]*	18.19	23.75	16.03	19.11	138
(ix) Timestep Inference Sampling *Linspace -> LinearQuadratic*	18.09	23.95	17.03	19.79	138
(x) with Video Pre-training *No Pre-training -> Pre-training 200K Steps (2x Eff.)*	18.28	24.60	17.18	20.15	138

Figure 3 Kaleido Design Ablations. We extensively ablate various architectural designs and training strategies to explore effective scaling strategies for generative neural rendering. Each ablation experiment was conducted with Kaleido-Small, trained for 100K steps in total, on a mixture of Objaverse and uCO3D sampled randomly. We report PSNR and training throughput for each configuration and evaluate performance in two settings: 1 target view conditioned on 5 reference views, and 5 target views conditioned on 5 reference views. We broadly split our designs into four categories: the Kaleido architecture design spaces (i–v); scaling stability techniques to handle large activations (vi–vii); training and inference timestep sampling strategies (viii–ix); and the role of video pre-training (x). The arrow ($\rightarrow$) indicates the progression from our initial baseline design to our final, optimised design choice.

Finally, we normalise the camera translation element $\mathbf{t}$ such that its maximum norm across all views in a given scene is 1. This ensures all positional transformations remain within a *bounded* range, which we found is crucial for stable training to handle different scene scales.

Our ablations confirm that this unified design outperforms (more significantly in multi-view settings) both a simpler baseline (2D RoPE + 3D CaPE without value-transformation) and the Plücker embeddings used in other leading sequence-to-sequence rendering models (Gao et al., 2025; Zhou et al., 2025; Jin et al., 2024).

(iii) *Principled View Sampling Improves Generalisation.* The strategy for sampling reference and target views is an *often-overlooked design aspect* of sequence-to-sequence rendering models, which are typically trained with a fixed number of reference and target views (Jin et al., 2024; Kong et al., 2024; Gao et al., 2025). While such fixed-view training can generalise to other view configurations, we found that performance can be further improved by adopting a more principled sampling strategy.

Our key insight is that *the rendering task becomes easier and more constrained as the number of reference views increases*. Therefore, the training process should place more emphasis on the challenging, less-constrained scenarios involving fewer reference views. To achieve this, we designed a sampling distribution $\pi(n)$ where its probability density halves as the number of reference views n increases, *i.e.*, $\pi(n+1) = \frac{1}{2}\pi(n), n \geq 0$. We design an **exponential distribution** which elegantly provides this property:

$$\pi(n) = \lambda e^{-\lambda n}, \quad \text{where } \lambda = \ln(2) \tag{5}$$

In each training step, we sample the number of reference views N from this distribution and set the remaining $M = V - N$ as target views. By combining this with random attention masking, our model is exposed to all possible combinations of $(n, m), n \in [1, N], m \in [1, M]$ view pairs, such that $n + m \in [2, V]$.

Our experiments show this sampling strategy offers the best trade-off between single- and multi-view conditioning. It significantly outperforms both fixed-view sampling and uniform sampling, which tend to degrade single-view performance by over-emphasising multi-view settings.

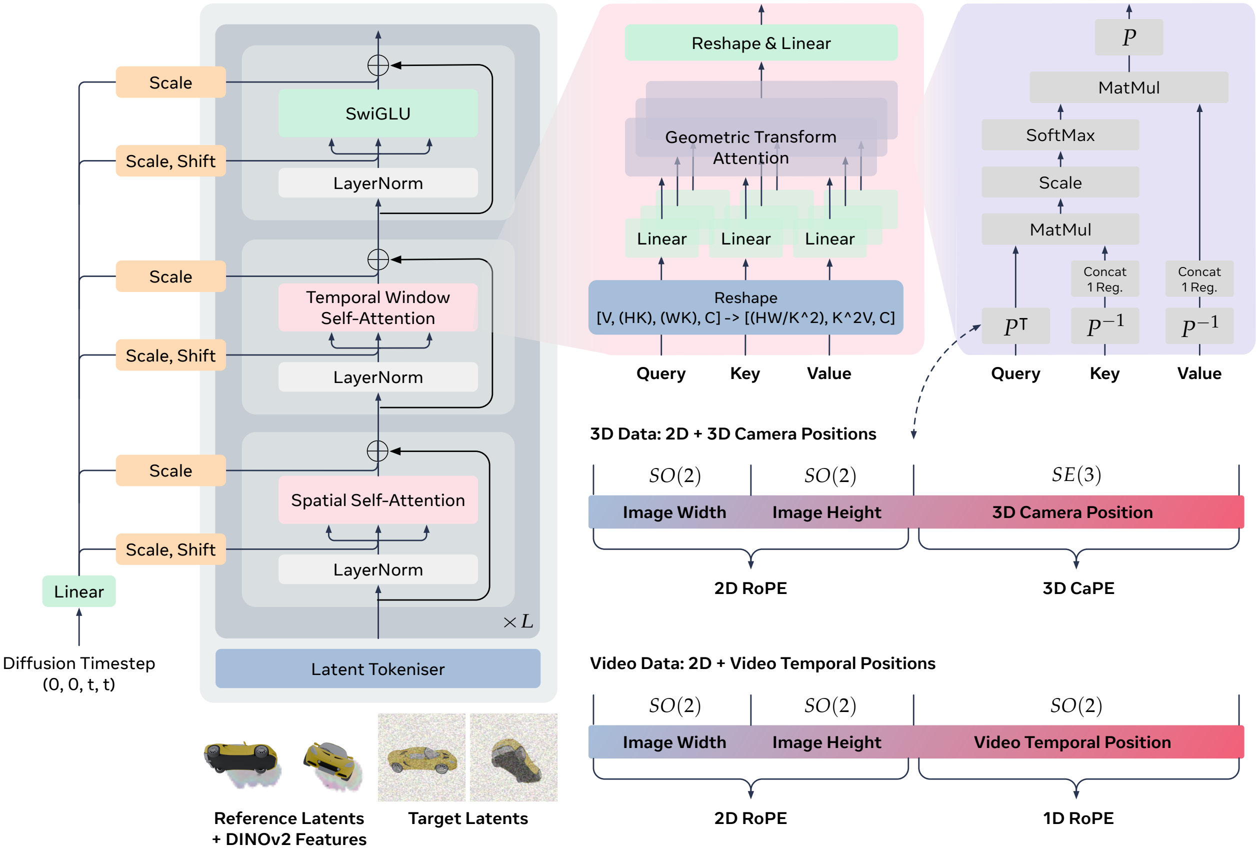


Figure 4 Kaleido Architecture Design Details. Kaleido is designed with a simple and scalable decoder-only transformer. It processes a sequence of tokens with clean reference latents (concatenated with their DINOv2 features) and noised target latents. During training, a single timestep t is sampled per scene and integrated into the network via AdaIN layers, similar to DiT (Peebles and Xie, 2023). The core of the model consists of repeating blocks of spatial self-attention (for within-frame interactions) followed by temporal window attention (for cross-frame interactions), and a SwiGLU feed-forward layer. Within each attention block, we encode a unified positional encoding design based on Geometric Transformation Attention (GTA) (Miyato et al., 2024), which consistently represents all 2D, 3D, and temporal positions. This enables the same architecture to be trained on both video and multi-view 3D data without architectural changes.

(iv) *Expanded Perception Field with Window Attention* Our baseline model processes a token sequence of shape $V \times H \times W$ (a sequence of V image latents with height H and width W) using a standard factorised attention mechanism: Spatial Attention (within-frame) followed by Temporal Attention (cross-frame). This approach has a computational complexity of $O(H^2W^2) + O(V^2)$, which is efficient but limits cross-view interactions to a single token at each spatial location. To improve this, we redesign the Temporal Attention layer by expanding its receptive field. Instead of attending to a single token across frames, each query token now attends to a local $K \times K$ window around the corresponding spatial location in all other frames. This **windowed cross-view attention** design significantly improves feature exchange between views while maintaining computational efficiency. The complexity only increases by a small, constant factor from $O(V^2)$ to $O(V^2K^4)$, which is far more scalable than the full attention's complexity of $O(V^2H^2W^2)$ cost, given that $K \ll H, W$.

Our experiments show that this design consistently boosts performance with larger window sizes. In practice, we use a window size of $K = 4$ for our Small and Medium models and $K = 8$ for our Large model.

(v) *Integration of Auxiliary Visual Features* We study the integration of auxiliary visual features from pre-trained networks to enhance 3D perception. Our findings show that features from **DINOv2** (Oquab et al., 2024) further improve Kaleido's depth estimation on in-the-wild images, leading to more accurate renderings. These pre-trained semantic features performed similarly to, and sometimes slightly better than, pre-computed depth or surface normals, which encode explicit scene geometry built on top of the same DINOv2 model.

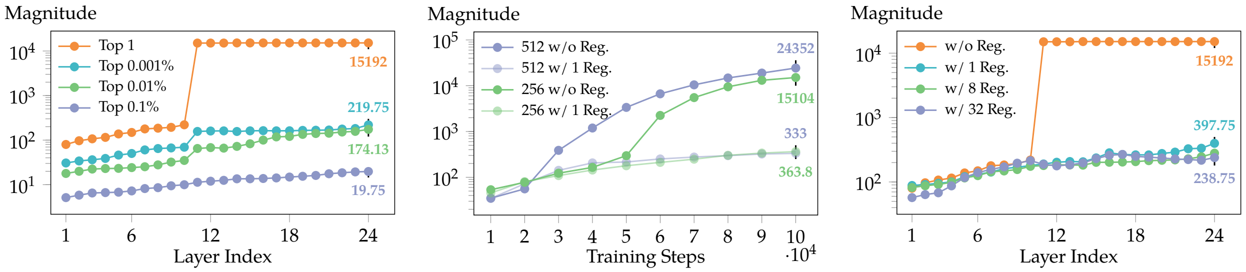
We also observed that larger DINOv2 models provide additional, albeit marginal, performance gains. Based on this trade-off, we pair the feature extractor with our model size: we use DINOv2 with ViT-B backbone for Kaleido-Small and Medium, and DINOv2 with ViT-L backbone for our Large model.

3.2.2 Massive Activations in Rectified Flow Transformers

During our initial scaling experiments with Kaleido, we consistently observed severe instability in training convergence on high-resolution images. A deeper analysis revealed that this instability arises from massive activations emerging within the transformer layers.

While massive activations have been studied extensively in autoregressive language models (Sun et al., 2024) (where they are sometimes called “attention sinks” (Xiao et al., 2024; Gu et al., 2025)) and in visual representation learning (Darcet et al., 2024), their behaviour within diffusion or rectified flow models remains largely unexplored. In other contexts, they are known to act as *attention biases* or *global information aggregators*. In this section, we provide the first empirical analysis of this issue in the context of rectified flow transformers, comparing our findings to those observed in LLMs (Sun et al., 2024) and ViTs (Darcet et al., 2024) (illustrated in Fig. 5). Our key observations are:

1. Similar to the observations in language models, massive activations are very sparse in numbers, with only a few tokens exhibiting this behaviour (Fig. 5a).
2. The magnitude of these activations grows with model depth. We observe a sudden jump at a middle layer, after which the magnitude remains constantly high until the final layer. Unlike in language models, these activations do not diminish towards the end of the model depth.
3. The activation magnitudes in rectified flow transformers are significantly higher than those reported in other domains. While language transformers report values around 1K-2K and vision transformers around 200, our activations can reach as high as 15K for 256px resolution and 24K for 512px resolution (Fig. 5b). These values seem to positively correlate with the number of training tokens and they continue to grow during training, directly causing precision overflow in our fp16 mixed-precision training.



(a) Massive activations are very few in numbers.

(b) The activations grow faster and larger with more tokens.

(c) The use of learnable registers effectively reduces massive activations.

Figure 5 Visual Analysis of Massive Activations in a Rectified Flow Transformer. We provide an empirical analysis of massive activations emerging during training. (a) Visualisation of activation magnitudes across model layers at 100K training steps, showing they are sparse but grow suddenly at a middle layer. (b) The maximum activation magnitude (measured at the final layer) grows over training time and correlates positively with image resolution (and thus, number of tokens). (c) The same training configuration as (a), but with learnable register tokens applied, demonstrating a significant and consistent reduction in activation magnitudes.

We also found that these massive activations emerge most prominently when training on a mixture of synthetic and real-world data. This supports the hypothesis in ViTs that they act as *global information aggregators*, perhaps in our context to reconcile the *different rendering logic* required for synthetic scenes (*e.g.*, maintaining clean/solid colour backgrounds) and real scenes (*e.g.*, generating semantically consistent textures).

To resolve this instability, we adopted the solution from Sun et al. (2024), appending learnable "register" tokens to the keys and values in each attention layer. This simple design proved highly effective, consistently reducing activation magnitudes to a stable level ($\sim$300) for both low and high-resolution training, as shown in Fig. 5b and 5c. Additionally, our **ablation (vii)** shows that having 1 register token is already optimal, adding more provides no benefit and can even degrade performance.

In **ablation (vi)**, we also explored alternative solutions inspired by recent studies (Bozic et al., 2021; Sun et al., 2025; Tang et al., 2025), such as removing the scaling factor from the timestep conditioning or removing timestep conditioning entirely. While these modifications slightly reduced activation magnitudes, they were far less effective than using register tokens and consistently resulted in lower overall performance.

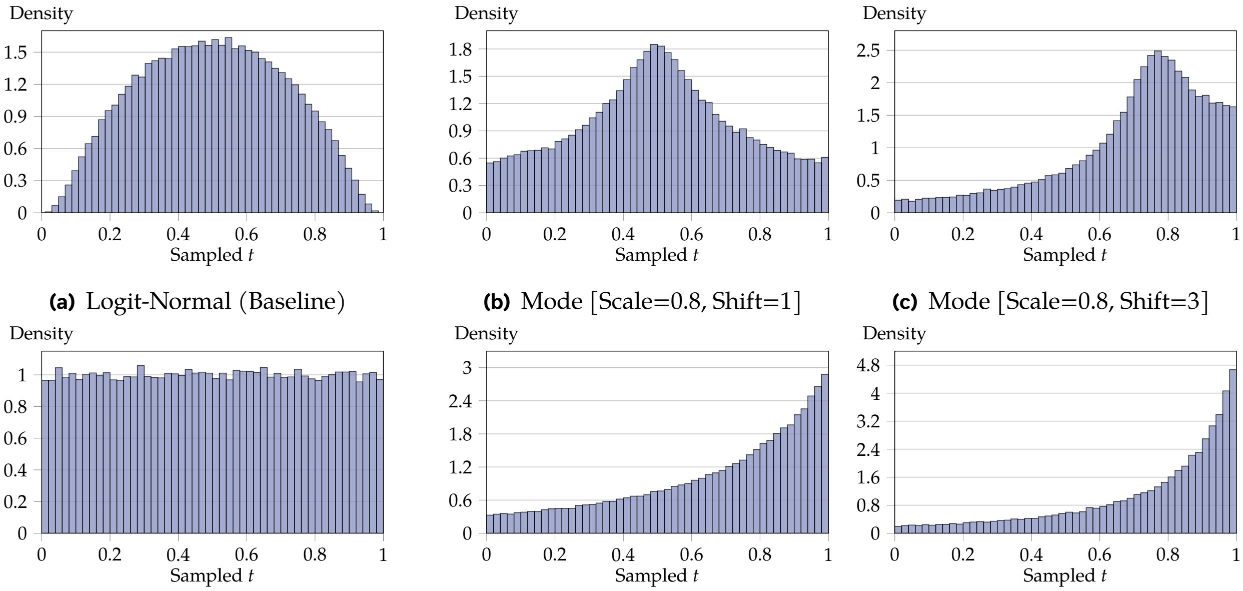
Finally, we empirically observed that massive activations are a pervasive phenomenon. They appear across all our model sizes, persist in both diffusion and rectified flow frameworks for image and video generation, and are independent of the inference timestep or input data. While our register token solution effectively stabilises training, we believe we have only scratched the surface of understanding this issue in visual generative models. A deeper analysis could inspire more stable and efficient architectures, which we consider an important direction for future work.

3.2.3 Tailored Rectified Flow SNR Samplers for Generative Neural Rendering

Prior rectified flow models for text-to-image/video generation (Esser et al., 2024; Polyak et al., 2024) typically use logit-normal sampling (Atchison and Shen, 1980) to focus on intermediate timesteps $t \in [0, 1]$. However, we found this approach to be suboptimal for rendering tasks, as generation quality becomes highly sensitive to the exact inference timesteps chosen, especially near the start ($t \approx 1$) and end ($t \approx 0$) of the trajectory.

We hypothesise this discrepancy arises from a fundamental difference between generation tasks: Text-conditioned image and video synthesis explores a *vast, unconstrained solution space*, whereas image-to-3D rendering is a *highly constrained problem*, as the output must be spatially consistent with the provided reference images. We argue this insight implies that rendering models should focus more heavily on the *early, high-noise timesteps* where the initial scene structure is formed.

This motivated our exploration of alternative SNR samplers. To test our hypothesis, we ablate three base distributions previously explored in SD3 (Esser et al., 2024): **Uniform**, **Logit-Normal**, and **Mode**, and apply a **modulation function**: $m(t, \sigma) = \sigma \cdot t / (1 + (\sigma - 1) \cdot t)$ to skew them towards the noise end of the trajectory (using a shifting factor $\sigma > 1$). The resulting probability densities are visualised in Fig. 6.



(a) Logit-Normal (Baseline)

(b) Mode [Scale=0.8, Shift=1]

(c) Mode [Scale=0.8, Shift=3]

(d) Uniform [Shift=1]

(e) Uniform [Shift=3]

(f) Uniform [Shift=5]

Figure 6 Probability Densities of Different Timestep Sampler. We visualise the PDFs for the samplers evaluated in our ablation study, where $t = 1$ is the noise end and $t = 0$ is the data end. (a) Logit-Normal: The standard baseline, which concentrates probability mass on the middle of the timestep range and has diminished density near the endpoints; (b) Mode: Similar to Logit-Normal, but maintains a positive density at the endpoints. (d) Uniform: A standard uniform distribution, which samples all timesteps with equal probability; (c), (e), and (f) show the Mode and Uniform distributions shifted towards the high-noise end of the interval to study the effect of noise-biased sampling.

In **ablation (viii)**, our results clearly show that distributions shifted towards the noise end outperform the unshifted baselines Logit-Normal and Mode samplers by a large margin, especially in multi-view settings. This validates our hypothesis that emphasising the early, high-noise timesteps is crucial for view synthesis, which operates in a highly constrained solution space. While a uniform distribution with a shift factor of 3 performs marginally better than a shift factor of 5, indicating performance was near saturation, we chose **shifted Mode sampling** as our final design. This approach provides a superior balance between the critical early timesteps and the intermediate steps compared to a heavily shifted uniform distribution.

In **ablation (ix)**, we align our inference process with our noise-biased training strategy by adopting a **linear-quadratic sampling** schedule. This schedule samples the first half of the timesteps linearly and the second half quadratically, effectively concentrating more computation on the initial, high-noise part of the trajectory. We found that this design combination, using a noise-biased sampler for both training and inference, delivered the single most significant performance improvement across all of our Kaleido design ablations.

3.2.4 Video Pre-training Improves 3D Efficiency

Finally, in **ablation (x)**, we validated our core motivation of treating 3D as a specialised sub-domain of video. The results confirm that pre-training on video data significantly improves the efficiency of subsequent 3D fine-tuning. Specifically, we observed that pre-training on large-scale video data for 100K and 200K steps resulted in 1.3x and 2.0x improvements in 3D training efficiency, respectively. This demonstrates that a more capable video foundation model directly translates to faster convergence on view synthesis tasks, successfully concluding our Kaleido design ablations.

3.3 Frame Interpolation as Zero-shot Spatial Upsampler

While many video generation models use a VAE with temporal compression to reduce memory usage, this method is incompatible with multi-view 3D datasets, which are typically sparsely captured and lack temporal consistency. Consequently, Kaleido must rely on a standard image-based VAE. This presents a practical challenge at inference time. When Kaleido renders a dense, video-like sequence from a continuous camera trajectory, generating every frame of such a sequence solely by Kaleido alone would be computationally expensive and memory-intensive.

To address this, we train a separate, lightweight frame interpolation model using our video data. This model's role is to efficiently generate the intermediate frames between the sparse keyframes rendered by Kaleido. For this task, we adapt the FiLM architecture (Reda et al., 2022), a deterministic, convolutional model designed for fast prediction. This two-stage approach, sparse generation by Kaleido followed by deterministic interpolation by FiLM, mitigates the high memory cost of dense rendering. It effectively emulates the decoding stage of a temporal VAE, allowing us to produce smooth, high frame-rate video sequences efficiently.

4 Experiments

We designed three variations of our model: Small, Medium, and Large, with increasing parameter counts to demonstrate the scalability of our architecture. The design choices for each model are summarised in Table 1. Hereafter, we refer to our largest model simply as *Kaleido* and the entire collection as the *Kaleido family*.

	Layers	Hidden Size	Query Heads	KV Heads	Window Size	Aux. Encoder	Total Params.
Kaleido-Small	24	1024	16	4	4	DINOv2-B (86M)	653M
Kaleido-Medium	32	1280	20	5	4	DINOv2-B (86M)	1.2B
Kaleido	40	1792	28	7	8	DINOv2-L (300M)	3.1B

Table 1 Kaleido Family Architecture Details. We detail the key hyper-parameters for our three model variants: the number of layers, hidden embedding size, number of query and key/value heads, the window size used in temporal attention, the choice of auxiliary DINOv2 encoder, and the total parameter count.

4.1 Training Configurations and Evaluation Strategy

Training Datasets The Kaleido family is trained on a diverse mixture of object-level and scene-level datasets. For object-level data, we use **ShutterStock 3D**, our licensed collection of synthetic 3D meshes, which we render with object-centric camera poses under varied lighting conditions; and **uCO3D** (Liu et al., 2025b), which includes real-world objects with estimated poses. For scene-level data, we combine several datasets: **RealEstate10K** (Zhou et al., 2018), which features indoor room scenes; **DL3DV** (Ling et al., 2024), which features both indoor and outdoor scenes; and a filtered subset of **ShutterStock Video**. This licensed video subset is curated to include only static scenes, and then labelled with a pose estimator VGGT (Wang et al., 2025).

In summary, our 3D fine-tuning dataset consists of approximately 1.5M object sequences and 2M scene sequences. For the initial video pre-training stage, we leverage the full, unfiltered Shutterstock Video dataset, comprising 34M video clips.

Training Strategy Our training process follows a two-stage, progressive-resolution curriculum. First, we pre-train Kaleido exclusively on video data at a fixed 256px resolution. We then fine-tune on our combined multi-view 3D datasets, progressively increasing the resolution from 256px to 512px, and finally to 1024px. In our 1024px fine-tuning, we introduce multi-aspect-ratio training (including 1:1, 4:5, 5:4, 16:9, and 9:16) to enable flexible resolution generation. Detailed training hyper-parameters can be found in Appendix B.

Evaluation Strategy We evaluate Kaleido on both view synthesis (Section 4.2) and 3D reconstruction benchmarks (Section 4.3). All evaluation datasets were held out and not used during model training. For all experiments, we report the zero-shot performance of Kaleido without any per-dataset fine-tuning. Unless otherwise specified, all generations use a classifier-free guidance scale of 1.5. To ensure a fair comparison with prior work, the frame interpolation model is not used in these evaluations.

4.2 Results on Novel View Synthesis

Compared to Generative NVS Methods We first evaluate Kaleido's zero-shot performance on standard novel view synthesis (NVS) benchmarks. For object-level NVS, we compare against leading object-specific generative NVS methods: **SV3D** (Voleti et al., 2024) and **EscherNet** (Kong et al., 2024) on three synthetic object-level datasets: **OmniObject3D (OO3D)** (Wu et al., 2023), the 30-object subset of **GSO (GSO-30)** (Downs et al., 2022), and the multi-object **RTMV** dataset (Tremblay et al., 2022).

For scene-level NVS, we compare against **SEVA** (Zhou et al., 2025), the current state-of-the-art general-purpose generative NVS model, on three scene-level datasets: **LLFF** (Mildenhall et al., 2019), **Mip-NeRF 360** (Barron et al., 2022), and **Tanks and Temples** (Knapitsch et al., 2017) datasets. To ensure a fair comparison, we match our model's resolution to the baselines, using our 256px checkpoint against EscherNet (evaluated at 256px) and our 512px checkpoint against SV3D and SEVA (both evaluated at 576px). For single-view evaluations on scene-level datasets, we address scale ambiguity by sweeping camera translations along each and all axes (from 0.1 to 2.0) and reporting the best result, following the protocol of SEVA.

	OO3D	**GSO-30**					**RTMV**					**LLFF**		**Mip-NeRF 360**			**Tanks and Temples**			
# Ref. Views	1	1	2	3	5	10	1	2	3	5	10	1	3	1	3	6	1	3	6	9
Eval. Data Type	Object	Object					Multi-Object					Scene		Scene			Scene			
Eval. Resolution	512	256					256					512		512			512			
Eval. Tar. Views	20	15					10					5		27			35			
SoTA Model	SV3D	EscherNet					EscherNet					SEVA		SEVA			SEVA			
Results (PSNR↑)	19.28	20.24	22.91	24.09	25.09	25.90	10.56	12.66	13.59	14.52	15.55	14.03	19.48	12.93	15.78	16.70	11.28	12.65	13.80	14.72
Kaleido-Small	19.77	18.58	23.73	26.20	29.11	31.66	13.57	17.18	18.41	19.97	21.75	14.57	19.30	12.75	15.81	17.07	11.40	13.13	14.13	15.20
Kaleido-Medium	20.78	20.32	25.78	28.01	30.74	32.94	13.78	18.07	19.41	21.09	22.73	14.86	20.40	14.17	16.47	17.80	11.36	13.04	14.43	15.47
Kaleido	21.06	20.94	26.31	28.89	31.37	33.74	14.66	18.48	19.69	21.13	23.04	15.34	20.71	13.74	16.78	18.03	11.79	13.20	14.61	15.88

Table 2 Zero-shot PSNR Performance with Generative Methods. Kaleido achieves state-of-the-art NVS performance across all object- and scene-level benchmarks, with particularly dominant results in many-view settings. Notably, our Kaleido-Small model consistently matches or outperforms all baselines despite having significantly fewer model parameters.

In Table 2, our PSNR results demonstrate that even our smallest model, Kaleido-Small (0.6B), performs on par with or surpasses all baselines across both object and scene-level benchmarks. This is particularly noteworthy given its efficiency, as it uses less than half the model parameters of SEVA (1.5B) and SV3D (2.3B).

Furthermore, Kaleido exhibits strong positive scaling, with performance consistently improving as model size increases. Our largest Kaleido model decisively outperforms all competing methods across every dataset, often by a remarkable margin. The benefits of scaling are most pronounced in multi-view settings, where Kaleido achieves an incredible +7.8 dB PSNR improvement on GSO-30 (10 views) over EscherNet; and +1.3 dB PSNR improvement on LLFF (3 views) over SEVA. However, we note that both Kaleido and SEVA struggle on the Tanks and Temples dataset ($<$ 16 PSNR with 9 views), which features unbounded scenes with extreme viewpoint changes, highlighting a clear direction for future improvements.

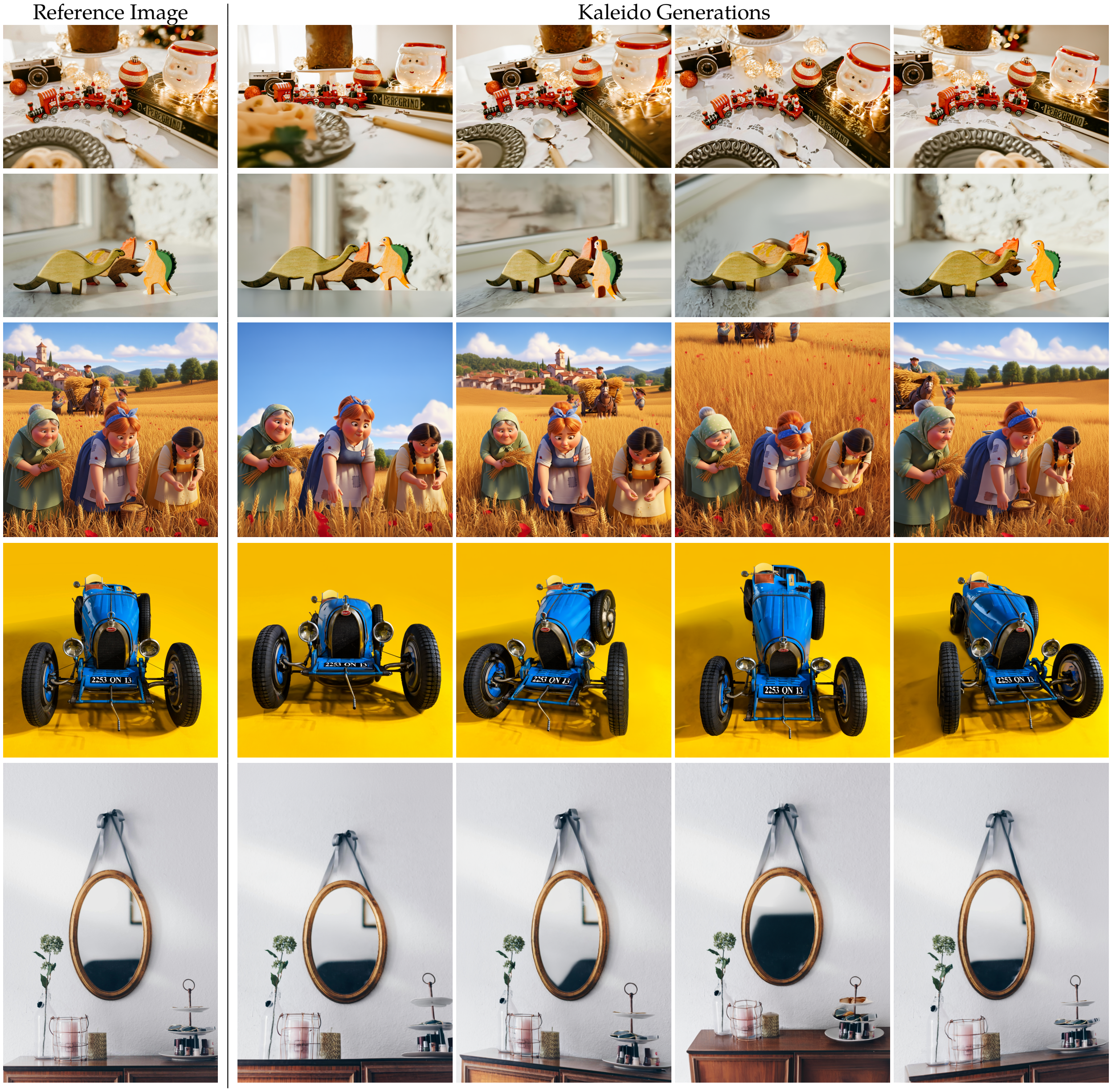
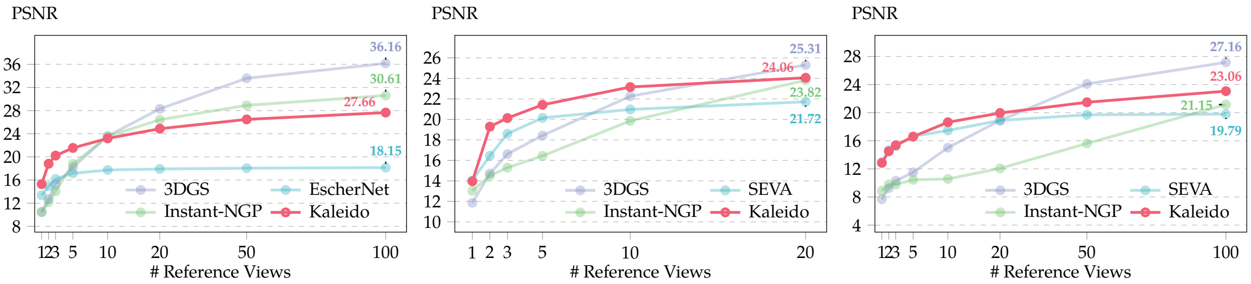


Figure 7 In-the-Wild Single-View Rendering with Kaleido. We showcase Kaleido's zero-shot generative capabilities on challenging in-the-wild images. From a single input view (first column), Kaleido generates a sequence of photorealistic novel views along a circular, object-centric camera trajectory. The examples feature complex scenes with diverse objects and structures, demonstrating Kaleido's remarkable generalisation and high-fidelity rendering quality.

A full breakdown of SSIM and LPIPS metrics is provided in Appendix C. For additional qualitative results, we present high-resolution (1024px) single-view conditioned generations on in-the-wild images in Fig. 7.

Compared to Per-Scene Optimisation Methods Next, we evaluate the upper bound of Kaleido's rendering precision when provided with many reference views. For this analysis, we compare against two state-of-the-art scene-specific optimisation methods: **Instant-NGP** (Müller et al., 2022) and **3D Gaussian Splatting (3DGS)** (Kerbl et al., 2023). As these methods are optimised per-scene, they represent a strong performance ceiling[1]. We also include our generative NVS baselines that can accept a flexible number of reference views: **EscherNet**, evaluated on the **NeRF-Synthetic** dataset (Mildenhall et al., 2020) with 256px resolution; and **SEVA**, evaluated on the **LLFF** and **Mip-NeRF 360** datasets (Mildenhall et al., 2019; Barron et al., 2022) with 512px resolution.



(a) NeRF Synthetic [256 Res.] **(b)** LLFF [512 Res.] **(c)** Mip-NeRF 360 [512 Res.]

Figure 8 PSNR Performance with Per-Scene Optimisation Methods. Kaleido's performance scales consistently with more reference views, demonstrating strong zero-shot generalisation despite being trained on 12 fixed total frames. It significantly outperforms other generative NVS baselines, with the performance gap widening as more views are provided. Notably, when given all available reference views, Kaleido surpasses Instant-NGP on both scene-level datasets.

In Fig. 8, we can observe that Kaleido and the other generative baselines (EscherNet, SEVA) initially outperform the scene-specific methods when given fewer than 10 reference views. However, a key difference emerges as more views are added: the performance of the other generative baselines quickly plateaus, while Kaleido shares the same positive scaling trend as the per-scene optimisation methods, with its performance continuing to improve. This superior zero-shot view generalisation, despite being trained on only 12 fixed total views, highlights the advantages of Kaleido's design and creates a widening performance gap over other generative models.

When all available reference views are used, Kaleido's performance on the NeRF-Synthetic dataset is nearly on par with Instant-NGP. On the more complex LLFF and Mip-NeRF 360 scene datasets, Kaleido surpasses Instant-NGP, marking the first time a zero-shot generative model has matched the quality of a state-of-the-art, per-scene optimisation method. Qualitative comparisons are provided in Fig. 9.

Given that per-scene methods can be (very) sensitive to camera coordinate systems and sometimes fail to converge, Kaleido's robust, data-driven performance highlights the immense potential of zero-shot solutions for general-purpose rendering.

4.3 Results on 3D Reconstruction

Given Kaleido's precise multi-view rendering capabilities, high-quality 3D reconstruction can be achieved by applying an off-the-shelf reconstruction framework to its generated views. In this section, we evaluate this capability on the **GSO-30** dataset. We compare Kaleido against a diverse set of generative models designed specifically for image-to-3D tasks. These include methods for direct 3D generation like **Point-E** (point clouds) (Nichol et al., 2022) and **Shape-E** (NeRFs) (Jun and Nichol, 2023); optimisation-based methods like **DreamGaussian** (Tang et al., 2024); and view-synthesis-based methods like **One-2-3-45** (Liu et al., 2023a) and **SyncDreamer** (Liu et al., 2023d).

[1] For both Instant-NGP and 3DGS, we follow the best practices validated in NVS benchmarking pipelines (Kulhanek and Sattler, 2024), applying hand-tuned pose transformations for each scene and in each dataset to obtain the optimal performance.

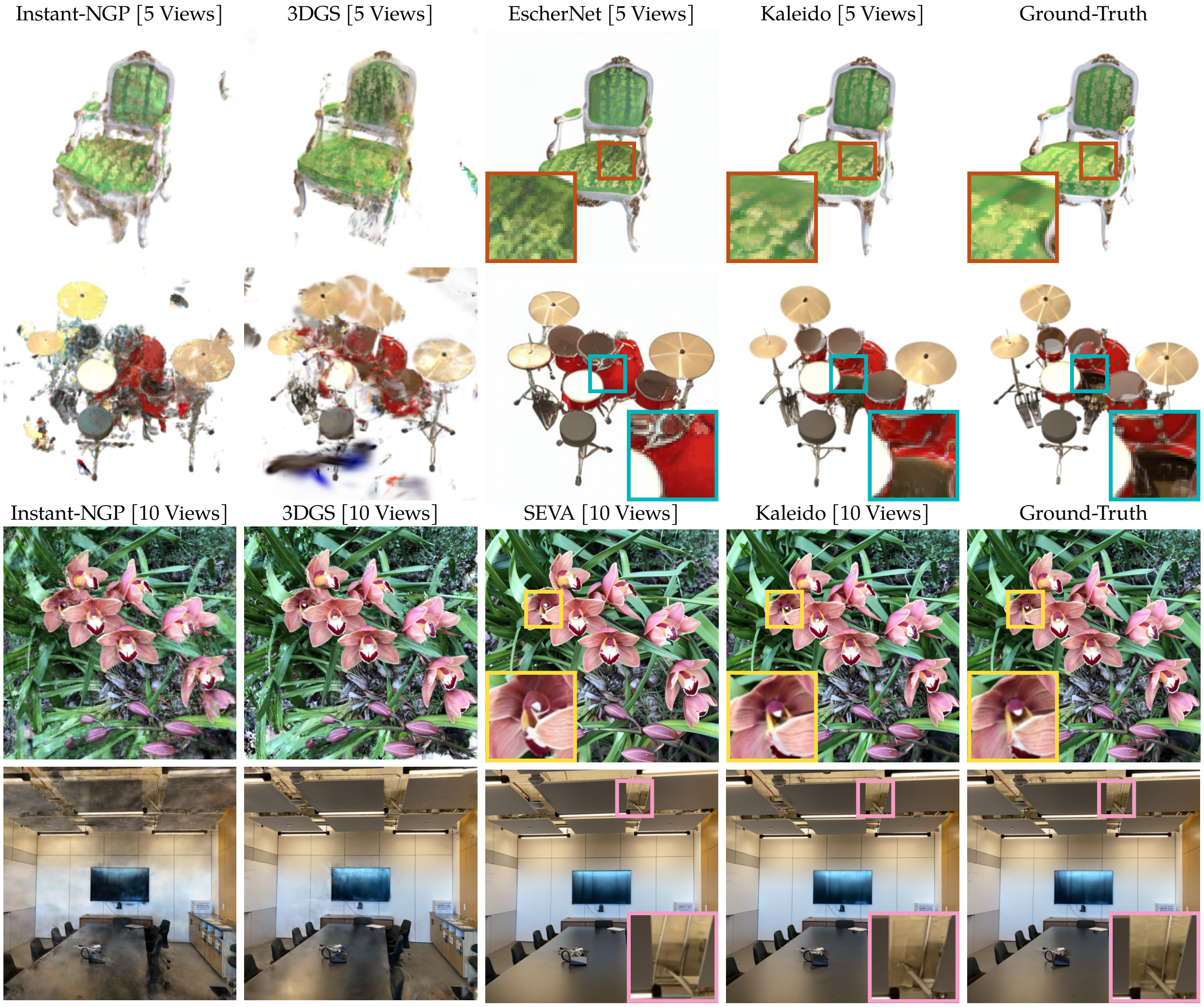


Figure 9 Qualitative Comparison on NeRF-Synthetic (256px, top) and LLFF (512px, bottom). With more reference views, Kaleido demonstrates superior rendering precision compared to other generative baselines, with more accurate texture details and pose alignment. Furthermore, it avoids the representation-based artefacts sometimes present in the per-scene optimisation methods, highlighting the robustness of its learned, data-driven prior.

Following the evaluation protocol of SyncDreamer and EscherNet, we perform reconstruction by first using Kaleido to generate a set of views from pre-defined, object-centric camera poses, and then fitting a surface with an off-the-shelf surface reconstruction framework. Specifically, we adopt the camera setup from EscherNet, rendering 36 views by varying the azimuth from 0° to 360° (in 30° increments) at three fixed elevations (-30°, 0°, 30°). These generated views then serve as input for the NeuS2 reconstruction (Wang et al., 2023a). For a fair comparison, all baseline methods and Kaleido are evaluated at 256px resolution. We also provide results for Kaleido at 1024px resolution to showcase its high-resolution generation capabilities.

In Table 3, Kaleido again achieves state-of-the-art performance in 3D reconstruction, significantly outperforming direct image-to-3D models, our NeuS baseline, and EscherNet.[2] The results highlight Kaleido's remarkable rendering efficiency and precision. With just 2 reference views, our model has surpassed the reconstruction quality that EscherNet achieves with 10 views. This superiority is more evident qualitatively in Fig. 10. Given the same 256px resolution, Kaleido's generated meshes are significantly better. At 1024px resolution, the reconstructed textures are incredibly detailed and sharp, appearing close to the ground truth and suggesting exciting new applications for high-fidelity, few-shot 3D reconstruction.

[2] We also attempted to evaluate NeuS2 with the same limited input views, but the reconstruction failed to converge for most objects.

	1 View		2 Views		3 Views		5 Views		10 Views	
	CD↓	VIoU↑	CD↓	VIoU↑	CD↓	VIoU↑	CD↓	VIoU↑	CD↓	VIoU↑
Point-E	0.0447	0.2503	–	–	–	–	–	–	–	–
Shape-E	0.0448	0.3762	–	–	–	–	–	–	–	–
One-2-3-45	0.0667	0.4016	–	–	–	–	–	–	–	–
DreamGaussian	0.0459	0.4531	–	–	–	–	–	–	–	–
SyncDreamer	0.0400	0.5220	–	–	–	–	–	–	–	–
NeuS	–	–	–	–	0.0366	0.5352	0.0245	0.6742	0.0195	0.7264
EscherNet	0.0314	0.5974	0.0215	0.6868	0.0190	0.7189	0.0175	0.7423	0.0167	0.7478
Kaleido	0.0214	0.6800	0.0120	0.7785	0.0113	0.7960	0.0104	0.8082	0.0100	0.8118
Kaleido [1024 Res.]	0.0183	0.7006	0.0118	0.7851	0.0104	0.8053	0.0091	0.8290	0.0086	0.8418

Table 3 3D Reconstruction Performance on GSO-30. We measure reconstruction quality using Chamfer Distance (CD, lower is better) and Volumetric IoU (VIoU, higher is better). Kaleido clearly surpasses EscherNet by a large margin, demonstrating 5x greater view efficiency: Kaleido achieves a better reconstruction quality with just two views than EscherNet does with ten. The quality is further improved when using higher-resolution renderings from Kaleido.

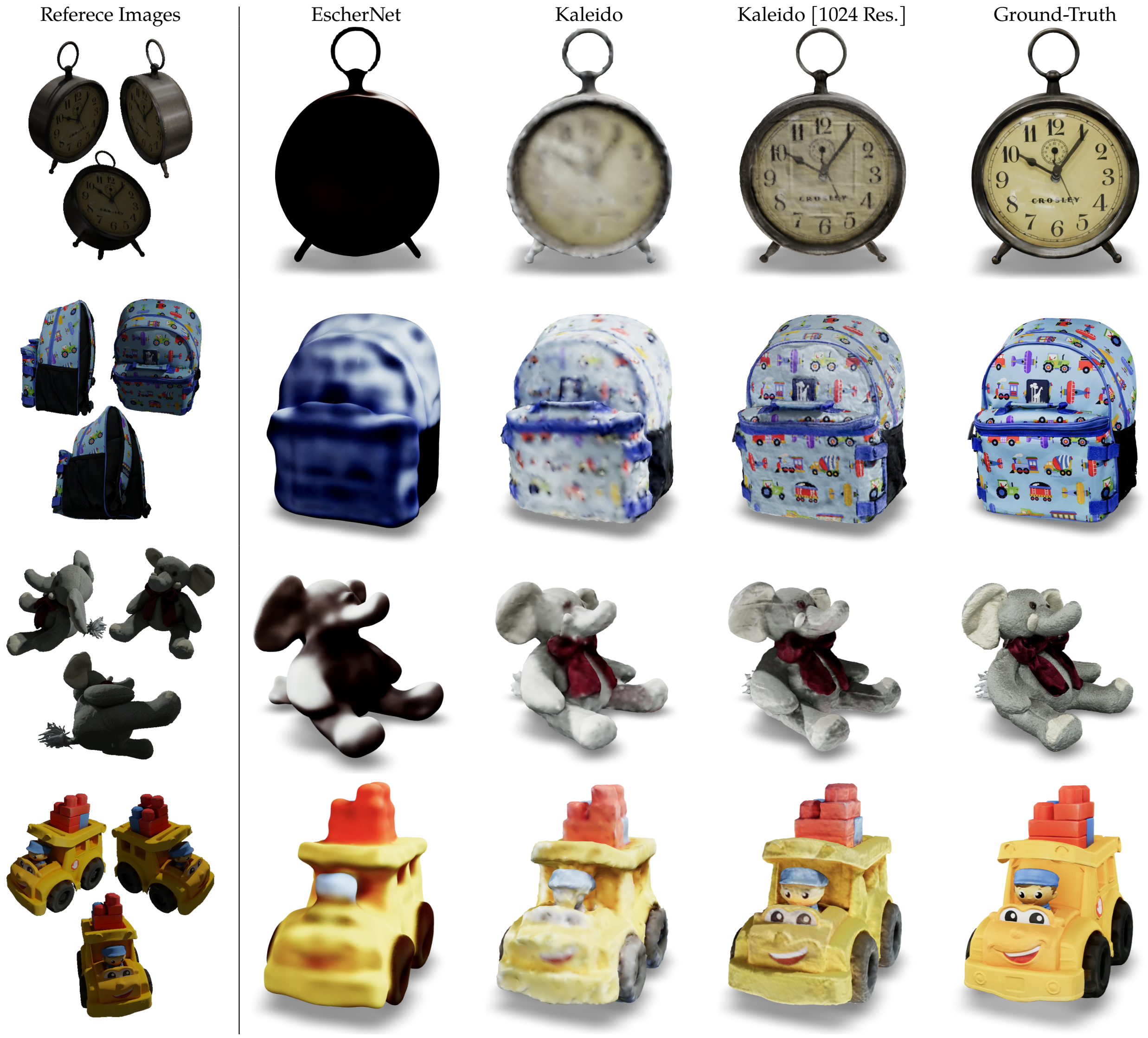


Figure 10 Visualisation of 3D Reconstructions with 3 Reference Views. Kaleido's precise renderings enable high-fidelity 3D mesh reconstruction using NeuS2. When leveraging 1024px renderings, the resulting meshes exhibit incredibly detailed textures, accurately capturing fine features like the numbers on the clock and the intricate patterns on the backpack.

5 Conclusions, Limitations and Future Works

In this paper, we introduced Kaleido, a new family of generative models that redefines neural rendering as a pure sequence-to-sequence problem, unifying 3D and video modelling. Through extensive ablations, we progressively modernised the architecture and training strategies, resulting in a model with exceptional rendering precision and spatial consistency.

Kaleido exhibits strong scaling properties and achieves state-of-the-art performance across a wide range of view synthesis and 3D reconstruction benchmarks. Most notably, it is the first generative rendering model to match the quality of per-scene optimisation methods in a zero-shot setting, representing a significant step towards a universal, general-purpose rendering engine.

Despite its strong performance, Kaleido has several limitations that open exciting avenues for future research:

Texture Flickering and Sticking. In certain challenging scenarios, we observe two main types of visual artefacts in Kaleido's generations. Texture flickering can occur in scenes with high-frequency details (e.g., the LLFF Fern scene), particularly at lower resolutions or when conditioned on very few reference views, i.e. 1 view. We also occasionally observe texture sticking, where the generated sequence exhibits a non-continuous jump between frames. Improving spatial consistency in these most challenging settings remains an important direction for future work.

Fixed Camera Intrinsics. Kaleido currently does not model camera intrinsics, which prevents it from generating effects like dolly-zooms, a capability present in models like SEVA (Zhou et al., 2025). Future work could explore incorporating intrinsic parameterisation, potentially through another form of RoPE-based positional encoding designs (Li et al., 2025), to allow for more flexible camera control.

Degraded Generations with Large Viewpoint Changes. While Kaleido often maintains excellent spatial consistency, its generated views can sometimes lack semantic plausibility when the viewpoint change is extreme. This suggests that while video pre-training builds a strong geometric foundation, it may not provide the diverse semantic knowledge required for high-fidelity single-image realism. Integrating priors from large-scale text-to-image/video models could be a promising direction to address this limitation.

Towards Faster Rendering. Kaleido's generation time scales with the number of input views, and it is far from real-time. To fully bridge the gap with efficient and fast scene-specific methods like 3D Gaussian Splatting, future work will focus on improving inference speed through techniques like step distillation or architectural optimisations.

Towards 4D Generation. Our unified positional encoding for space and time provides a natural foundation for true 4D generation. A promising future direction is to extend Kaleido to precisely control scenes across both space and time, enabling generative modelling of dynamic, four-dimensional worlds.

Acknowledgement

We would like to thank Jianyuan Wang for his support on VGGT and for his insightful suggestions on our early paper draft; and to Takeru Miyato for his helpful discussions on GTA. We are also grateful to Tom Monnier for his support with the uCO3D and ShutterStock3D datasets, and to Zaiwei Zhang and Marc Mapeke for their support with the monocular depth and normal model benchmarking. Finally, we extend our gratitude to Mengmeng Xu for his assistance with training data preparation and filtering.

References

Joshua Ainslie, James Lee-Thorp, Michiel de Jong, Yury Zemlyanskiy, Federico Lebron, and Sumit Sanghai. Gqa: Training generalized multi-query transformer models from multi-head checkpoints. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2023.

Rohan Anil, Andrew M Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, et al. Palm 2 technical report. *arXiv preprint arXiv:2305.10403*, 2023.

Jhon Atchison and Sheng M Shen. Logistic-normal distributions: Some properties and uses. *Biometrika*, 1980.

Amir Bar, Gaoyue Zhou, Danny Tran, Trevor Darrell, and Yann LeCun. Navigation world models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023.

Aljaz Bozic, Pablo Palafox, Justus Thies, Angela Dai, and Matthias Nießner. Transformerfusion: Monocular rgb scene reconstruction using transformers. *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, et al. Video generation models as world simulators. *OpenAI Blog*, 1(8):1, 2024.

Jake Bruce, Michael D Dennis, Ashley Edwards, Jack Parker-Holder, Yuge Shi, Edward Hughes, Matthew Lai, Aditi Mavalankar, Richie Steigerwald, Chris Apps, et al. Genie: Generative interactive environments. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2024.

Anpei Chen, Zexiang Xu, Andreas Geiger, Jingyi Yu, and Hao Su. Tensorf: Tensorial radiance fields. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022.

Haoxin Chen, Yong Zhang, Xiaodong Cun, Menghan Xia, Xintao Wang, Chao Weng, and Ying Shan. Videocrafter2: Overcoming data limitations for high-quality video diffusion models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024a.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde De Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021.

Shoufa Chen, Mengmeng Xu, Jiawei Ren, Yuren Cong, Sen He, Yanping Xie, Animesh Sinha, Ping Luo, Tao Xiang, and Juan-Manuel Perez-Rua. Gentron: Diffusion transformers for image and video generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024b.

Shoufa Chen, Chongjian Ge, Yuqi Zhang, Yida Zhang, Fengda Zhu, Hao Yang, Hongxiang Hao, Hui Wu, Zhichao Lai, Yifei Hu, et al. Goku: Flow based video generative foundation models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision transformers need registers. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2024.

Andrew J Davison, Ian D Reid, Nicholas D Molton, and Olivier Stasse. Monoslam: Real-time single camera slam. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2007.

Deepmind. Veo. https://deepmind.google/models/veo/, 2024.

Deepmind. Genie3. https://deepmind.google/discover/blog/genie-3-a-new-frontier-for-world-models/, 2025.

Matt Deitke, Ruoshi Liu, Matthew Wallingford, Huong Ngo, Oscar Michel, Aditya Kusupati, Alan Fan, Christian Laforte, Vikram Voleti, Samir Yitzhak Gadre, et al. Objaverse-xl: A universe of 10m+ 3d objects. *Advances in Neural Information Processing Systems (NeurIPS)*, 2023a.

Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023b.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2020.

Laura Downs, Anthony Francis, Nate Koenig, Brandon Kinman, Ryan Hickman, Krista Reymann, Thomas B McHugh, and Vincent Vanhoucke. Google scanned objects: A high-quality dataset of 3d scanned household items. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2022.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv e-prints*, 2024.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2024.

Lijie Fan, Tianhong Li, Siyang Qin, Yuanzhen Li, Chen Sun, Michael Rubinstein, Deqing Sun, Kaiming He, and Yonglong Tian. Fluid: Scaling autoregressive text-to-image generative models with continuous tokens. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2025.

Sara Fridovich-Keil, Alex Yu, Matthew Tancik, Qinhong Chen, Benjamin Recht, and Angjoo Kanazawa. Plenoxels: Radiance fields without neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Yasutaka Furukawa, Carlos Hernández, et al. Multi-view stereo: A tutorial. *Foundations and Trends in Computer Graphics and Vision*, 2015.

Ruiqi Gao, Aleksander Holynski, Philipp Henzler, Arthur Brussee, Ricardo Martin Brualla, Pratul Srinivasan, Jonathan Barron, and Ben Poole. Cat3d: Create anything in 3d with multi-view diffusion models. *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

Xiangming Gu, Tianyu Pang, Chao Du, Qian Liu, Fengzhuo Zhang, Cunxiao Du, Ye Wang, and Min Lin. When attention sink emerges in language models: An empirical view. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2025.

Yulan Guo, Hanyun Wang, Qingyong Hu, Hao Liu, Li Liu, and Mohammed Bennamoun. Deep learning for 3d point clouds: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2020.

Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge University Press, 2003.

Yicong Hong, Kai Zhang, Jiuxiang Gu, Sai Bi, Yang Zhou, Difan Liu, Feng Liu, Kalyan Sunkavalli, Trung Bui, and Hao Tan. Lrm: Large reconstruction model for single image to 3d. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2023.

Shahram Izadi, David Kim, Otmar Hilliges, David Molyneaux, Richard Newcombe, Pushmeet Kohli, Jamie Shotton, Steve Hodges, Dustin Freeman, Andrew Davison, et al. Kinectfusion: real-time 3d reconstruction and interaction using a moving depth camera. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, 2011.

Wonbong Jang and Lourdes Agapito. Codenerf: Disentangled neural radiance fields for object categories. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2021.

Wonbong Jang and Lourdes Agapito. Nvist: In the wild new view synthesis from a single image with transformers. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Haian Jin, Hanwen Jiang, Hao Tan, Kai Zhang, Sai Bi, Tianyuan Zhang, Fujun Luan, Noah Snavely, and Zexiang Xu. Lvsm: A large view synthesis model with minimal 3d inductive bias. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2024.

Heewoo Jun and Alex Nichol. Shap-e: Generating conditional 3d implicit functions. *arXiv preprint arXiv:2305.02463*, 2023.

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics (TOG)*, 2023.

Diederik Kingma and Ruiqi Gao. Understanding diffusion objectives as the elbo with simple data augmentation. *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2014.

Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM Transactions on Graphics (TOG)*, 2017.

Xin Kong, Shikun Liu, Xiaoyang Lyu, Marwan Taher, Xiaojuan Qi, and Andrew J Davison. Eschernet: A generative model for scalable view synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Jonas Kulhanek and Torsten Sattler. Nerfbaselines: Consistent and reproducible evaluation of novel view synthesis methods. *arXiv preprint arXiv:2406.17345*, 2024.

Haoran Li, Haolin Shi, Wenli Zhang, Wenjun Wu, Yong Liao, Lin Wang, Lik-hang Lee, and Peng Yuan Zhou. Dreamscene: 3d gaussian-based text-to-3d scene generation via formation pattern sampling. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2024a.

Ruilong Li, Brent Yi, Junchen Liu, Hang Gao, Yi Ma, and Angjoo Kanazawa. Cameras as relative positional encoding. *arXiv preprint arXiv:2507.10496*, 2025.

Tianhong Li, Yonglong Tian, He Li, Mingyang Deng, and Kaiming He. Autoregressive image generation without vector quantization. *Advances in Neural Information Processing Systems (NeurIPS)*, 2024b.

Yixun Liang, Xin Yang, Jiantao Lin, Haodong Li, Xiaogang Xu, and Yingcong Chen. Luciddreamer: Towards high-fidelity text-to-3d generation via interval score matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Lu Ling, Yichen Sheng, Zhi Tu, Wentian Zhao, Cheng Xin, Kun Wan, Lantao Yu, Qianyu Guo, Zixun Yu, Yawen Lu, et al. Dl3dv-10k: A large-scale scene dataset for deep learning-based 3d vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2023.

Haozhe Liu, Shikun Liu, Zijian Zhou, Mengmeng Xu, Yanping Xie, Xiao Han, Juan Camilo Perez, Ding Liu, Kumara Kahatapitiya, Menglin Jia, et al. Mardini: Masked auto-regressive diffusion for video generation at scale. *Transactions on Machine Laerning Research (TMLR)*, 2025a.

Minghua Liu, Chao Xu, Haian Jin, Linghao Chen, Zexiang Xu, Hao Su, et al. One-2-3-45: Any single image to 3d mesh in 45 seconds without per-shape optimization. *Advances in Neural Information Processing Systems (NeurIPS)*, 2023a.

Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3d object. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2023b.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*, 2022.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2023c.

Xingchen Liu, Piyush Tayal, Jianyuan Wang, Jesus Zarzar, Tom Monnier, Konstantinos Tertikas, Jiali Duan, Antoine Toisoul, Jason Y Zhang, Natalia Neverova, et al. Uncommon objects in 3d. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025b.

Yuan Liu, Cheng Lin, Zijiao Zeng, Xiaoxiao Long, Lingjie Liu, Taku Komura, and Wenping Wang. Syncdreamer: Generating multiview-consistent images from a single-view image. *arXiv preprint arXiv:2309.03453*, 2023d.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2019.

Nanye Ma, Mark Goldstein, Michael S Albergo, Nicholas M Boffi, Eric Vanden-Eijnden, and Saining Xie. Sit: Exploring flow and diffusion-based generative models with scalable interpolant transformers. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2024.

Ben Mildenhall, Pratul P Srinivasan, Rodrigo Ortiz-Cayon, Nima Khademi Kalantari, Ravi Ramamoorthi, Ren Ng, and Abhishek Kar. Local light field fusion: Practical view synthesis with prescriptive sampling guidelines. *ACM Transactions on Graphics (TOG)*, 2019.

Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

Takeru Miyato, Bernhard Jaeger, Max Welling, and Andreas Geiger. Gta: A geometry-aware attention mechanism for multi-view transformers. In *International Conference on Learning Representations (ICLR)*, 2024.

Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Transactions on Graphics (TOG)*, 2022.

Raul Mur-Artal, Jose Maria Martinez Montiel, and Juan D Tardos. Orb-slam: A versatile and accurate monocular slam system. *IEEE Transactions on Robotics and Automation*, 2015.

Alex Nichol, Heewoo Jun, Prafulla Dhariwal, Pamela Mishkin, and Mark Chen. Point-e: A system for generating 3d point clouds from complex prompts. *arXiv preprint arXiv:2212.08751*, 2022.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *Transactions on Machine Laerning Research (TMLR)*, 2024.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2023.

Adam Polyak, Amit Zohar, Andrew Brown, Andros Tjandra, Animesh Sinha, Ann Lee, Apoorv Vyas, Bowen Shi, Chih-Yao Ma, Ching-Yao Chuang, et al. Movie gen: A cast of media foundation models. *arXiv preprint arXiv:2410.13720*, 2024.

Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*, 2022.

Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Alexander Raistrick, Lahav Lipson, Zeyu Ma, Lingjie Mei, Mingzhe Wang, Yiming Zuo, Karhan Kayan, Hongyu Wen, Beining Han, Yihan Wang, et al. Infinite photorealistic worlds using procedural generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

Fitsum Reda, Janne Kontkanen, Eric Tabellion, Deqing Sun, Caroline Pantofaru, and Brian Curless. Film: Frame interpolation for large motion. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Baptiste Roziere, Jonas Gehring, Fabian Gloeckle, Sten Sootla, Itai Gat, Xiaoqing Ellen Tan, Yossi Adi, Jingyu Liu, Romain Sauvestre, Tal Remez, et al. Code llama: Open foundation models for code. *arXiv preprint arXiv:2308.12950*, 2023.

Johannes Lutz Schönberger, Enliang Zheng, Marc Pollefeys, and Jan-Michael Frahm. Pixelwise view selection for unstructured multi-view stereo. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2016.

Noam Shazeer. Glu variants improve transformer. *arXiv preprint arXiv:2002.05202*, 2020.

Ruoxi Shi, Hansheng Chen, Zhuoyang Zhang, Minghua Liu, Chao Xu, Xinyue Wei, Linghao Chen, Chong Zeng, and Hao Su. Zero123++: a single image to consistent multi-view diffusion base model. *arXiv preprint arXiv:2310.15110*, 2023.

Yichun Shi, Peng Wang, Jianglong Ye, Mai Long, Kejie Li, and Xiao Yang. Mvdream: Multi-view diffusion for 3d generation. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2024.

Vincent Sitzmann, Michael Zollhöfer, and Gordon Wetzstein. Scene representation networks: Continuous 3d-structure-aware neural scene representations. *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Jianlin Su, Yu Lu, Shengfeng Pan, Ahmed Murtadha, Bo Wen, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 2021.

Chunyi Sun, Junlin Han, Weijian Deng, Xinlong Wang, Zishan Qin, and Stephen Gould. 3d-gpt: Procedural 3d modeling with large language models. In *Proceedings of the International Conference on 3D Vision (3DV)*, 2023.

Mingjie Sun, Xinlei Chen, J Zico Kolter, and Zhuang Liu. Massive activations in large language models. In *Conference on Language Modeling (CoLM)*, 2024.

Qiao Sun, Zhicheng Jiang, Hanhong Zhao, and Kaiming He. Is noise conditioning necessary for denoising generative models? In *Proceedings of the International Conference on Machine Learning (ICML)*, 2025.

Bingda Tang, Boyang Zheng, Sayak Paul, and Saining Xie. Exploring the deep fusion of large language models and diffusion transformers for text-to-image synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

Jiaxiang Tang, Jiawei Ren, Hang Zhou, Ziwei Liu, and Gang Zeng. Dreamgaussian: Generative gaussian splatting for efficient 3d content creation. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2024.

Jonathan Tremblay, Moustafa Meshry, Alex Evans, Jan Kautz, Alexander Keller, Sameh Khamis, Thomas Müller, Charles Loop, Nathan Morrical, Koki Nagano, et al. Rtmv: A ray-traced multi-view synthetic dataset for novel view synthesis. *arXiv preprint arXiv:2205.07058*, 2022.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

Vikram Voleti, Chun-Han Yao, Mark Boss, Adam Letts, David Pankratz, Dmitry Tochilkin, Christian Laforte, Robin Rombach, and Varun Jampani. Sv3d: Novel multi-view synthesis and 3d generation from a single image using latent video diffusion. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2024.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Yiming Wang, Qin Han, Marc Habermann, Kostas Daniilidis, Christian Theobalt, and Lingjie Liu. Neus2: Fast learning of neural implicit surfaces for multi-view reconstruction. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2023a.

Zhengyi Wang, Cheng Lu, Yikai Wang, Fan Bao, Chongxuan Li, Hang Su, and Jun Zhu. Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation. *Advances in Neural Information Processing Systems (NeurIPS)*, 2023b.

Tong Wu, Jiarui Zhang, Xiao Fu, Yuxin Wang, Jiawei Ren, Liang Pan, Wayne Wu, Lei Yang, Jiaqi Wang, Chen Qian, et al. Omniobject3d: Large-vocabulary 3d object dataset for realistic perception, reconstruction and generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2024.

Yiheng Xie, Towaki Takikawa, Shunsuke Saito, Or Litany, Shiqin Yan, Numair Khan, Federico Tombari, James Tompkin, Vincent Sitzmann, and Srinath Sridhar. Neural fields in visual computing and beyond. In *Computer Graphics Forum*, 2022.

Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2025.

Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelnerf: Neural radiance fields from one or few images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

Hong-Xing Yu, Haoyi Duan, Charles Herrmann, William T Freeman, and Jiajun Wu. Wonderworld: Interactive 3d scene generation from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025a.

Jiwen Yu, Yiran Qin, Xintao Wang, Pengfei Wan, Di Zhang, and Xihui Liu. Gamefactory: Creating new games with generative interactive videos, 2025b.

Jensen Zhou, Hang Gao, Vikram Voleti, Aaryaman Vasishta, Chun-Han Yao, Mark Boss, Philip Torr, Christian Rupprecht, and Varun Jampani. Stable virtual camera: Generative view synthesis with diffusion models. *Proceedings of the International Conference on Computer Vision (ICCV)*, 2025.

Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. Stereo magnification: learning view synthesis using multiplane images. *ACM Transactions on Graphics (TOG)*, 2018.

A Kaleido Design Ablative Quantitative Results

Table 4 presents the full quantitative results supporting the ablation study in Fig. 3, including all alternative design decisions we explored. To provide a comprehensive comparison, we report performance across three settings: one-to-five, five-to-one, and five-to-five reference-to-target views.

	Objaverse			uCO3D			Training
	1 → 5	5 → 1	5 → 5	1 → 5	5 → 1	5 → 5	Throughput
Objaverse - Single Baseline	14.56	19.53	21.23	-	-	-	-
uCO3D - Single Baseline	-	-	-	14.89	19.19	16.97	-
Objaverse + uCO3D - Joint Baseline	12.17	19.10	20.02	14.66	18.71	16.77	160
(i) Architecture Design [Vanilla DiT]							
DiT + Llama3 (SwiGLU + GQA)	13.02	20.18	21.23	14.63	19.31	17.27	160
(ii) Spatial Positional Encoding [2D RoPE + 3D CaPE]							
RoPE + Plucker	12.18	20.00	20.17	14.75	19.43	17.61	160
GTA [2D RoPE + 3D CaPE]	11.93	21.05	22.03	13.65	20.84	17.54	148
(iii) View Sampling Strategies [Fixed 6->6]							
Uniform Sampling w/o Masking	12.18	22.22	21.65	14.22	21.36	18.34	150
Uniform Sampling w/ Masking	14.51	21.00	20.22	14.58	20.37	17.60	150
Exponential Sampling w/ Masking	15.13	21.41	21.11	15.16	20.25	17.83	148
(iv) Temporal Attention Design [Temporal Attention (K=1)]							
Full Attention	14.54	22.62	22.40	15.00	20.75	18.28	58
Temporal Window Attention (K = 2)	15.45	21.60	21.04	15.40	20.43	18.15	146
Temporal Window Attention (K = 4)	15.67	22.25	21.79	15.55	20.81	18.45	142
Temporal Window Attention (K = 8)	15.73	22.60	22.54	15.85	21.39	19.15	103
(v) Auxiliary Features [None]							
DiNOv2 [DiT-B]	15.86	22.28	21.90	15.81	21.09	18.81	138
DiNOv2 [DiT-L]	16.34	22.65	22.43	15.94	21.24	18.75	135
MetaDepth [DiT-L]	15.82	22.28	22.22	15.76	21.26	18.65	135
MetaNormals [DiT-L]	15.77	22.32	22.11	15.59	21.20	18.85	135
(vi) Timestep Conditioning Design [AdaLN-Zero, Top 1 Act.: 15192]							
Shift Only [Top 1 Act.: 6820]	16.07	21.56	21.51	15.97	20.56	18.28	144
No Timestep [Top 1 Act.: 4312.]	16.26	20.85	21.00	16.15	20.38	18.38	148
(vii) Attention Registers [No Registers, Top 1 Act.: 15192]							
1 Register [Top 1 Act.: 397.75]	15.93	22.27	22.12	15.77	21.02	19.03	138
8 Registers [Top 1 Act.: 279.75]	15.26	22.04	21.94	15.72	20.75	18.55	138
32 Registers [Top 1 Act.: 238.75]	15.07	21.88	21.54	15.66	20.59	18.27	138
(viii) Timestep Sampling Training Strategy [LogitNorm [0,1]]							
Uniform [Shift = 1]	17.58	22.01	21.96	15.90	20.49	18.57	138
Uniform [Shift = 3]	18.27	23.64	23.28	16.39	21.74	19.21	138
Uniform [Shift = 5]	18.43	23.38	23.06	15.90	21.42	18.94	138
Mode [Scale = 0.8]	17.39	22.06	22.08	15.99	20.75	18.68	138
Mode [Scale = 0.8, Shift = 3]	18.19	24.06	23.75	16.03	21.76	19.11	138
(ix) Timestep Sampling Inference Sampling [Linspace [1, 999]]							
Trailing [1, 980]	17.95	23.66	23.51	16.70	21.83	19.43	138
LinearQuadratic [1, 999]	18.09	23.87	23.95	17.03	22.15	19.79	138
(x) with Video Pre-training [No video Pre-training]							
Video Pre-training 100K Steps (1.3x Eff.)	18.16	24.22	24.30	17.11	22.23	20.10	138
Video Pre-training 200K Steps (2x Eff.)	18.28	24.55	24.60	17.18	22.43	20.15	138

Table 4 Quantitative Results for Kaleido Design Ablations. We report the complete quantitative results (PSNR, higher is better) corresponding to the ablation study in Fig. 3. Performance is evaluated in one-to-five, five-to-one, and five-to-five reference-to-target view settings. Our final design choice for each component is marked in red.

B Additional Details of Kaleido Training Strategies

All Kaleido model variants are trained using the same datasets detailed in Sec. 4.1, with the AdamW optimiser (Loshchilov and Hutter, 2019) and a weight decay of 0.01. In each training iteration, we randomly sample a total of 12 frames per sequence. These frames are then partitioned into reference and target views according to the view sampling strategy in Sec. 3.2.1.

The learning rate is chosen based on the training stage. For the initial video pre-training and the first stage of 3D fine-tuning (both at 256px resolution), we apply a learning rate of 10^{-4}. For the subsequent high-resolution 3D fine-tuning stages (512px and 1024px resolution), we decrease the learning rate to 10^{-5}. To train our larger models at high resolutions, we incorporate FSDP sharding and activation checkpointing.

Across all stages, we use `fp16` mixed-precision training, as we find it crucial for stable training convergence; while `bf16` consistently leads to unstable training. Our largest Kaleido model is trained for two weeks on 512 NVIDIA H100 GPUs. Additional hyper-parameters are listed in Table 5.

	Stage 1 (Video data) [256 × 256]		**Stage 2 (3D data)** [256 × 256]		**Stage 3 (3D data)** [512 × 512]		**Stage 4 (3D data)** [1024 mixed AR]	
	Batch Size	# Steps	Batch Size	# Steps	Batch Size	# Steps	Batch Size	# Steps
Kaleido-Small	1024	700K	1024	300K	256	100K	256	100K
Kaleido-Medium	1024	700K	1024	300K	256	100K	256	100K
Kaleido	2048	700K	2048	500K	256	100K	256	100K

Table 5 Kaleido Training Pipeline. Kaleido's training follows a multi-stage curriculum. The model is first pre-trained on a large-scale video dataset and is then fine-tuned on combined multi-view 3D datasets, with the image resolution progressively increased from 256px up to 1024px. In the final stage, we sample images with mixed aspect ratios to enable flexible resolution generation. Larger batch sizes are used for our largest Kaleido model to validate scaling laws.

C Additional Results for Few-shot View Synthesis

We provide additional quantitative metrics for our few-view NVS benchmarks. Consistent with the PSNR results presented in Table 2, Kaleido achieves state-of-the-art SSIM and LPIPS scores across all object- and scene-level datasets, confirming its superior generative rendering capabilities.

	OO3D	**GSO-30**					**RTMV**					**LLFF**		**Mip-NeRF 360**			**Tanks and Temples**			
# Ref. Views	1	1	2	3	5	10	1	2	3	5	10	1	3	1	3	6	1	3	6	9
Eval. Data Type	Object	Object					Multi-Object					Scene		Scene			Scene			
Eval. Resolution	512	256					256					512		512			512			
Eval. Tar. Views	20	15					10					5		27			35			
SoTA Model	SV3D	EscherNet					EscherNet					SEVA		SEVA			SEVA			
Results (LPIPS↓)	0.158	0.095	0.064	0.052	0.043	0.036	0.410	0.301	0.258	0.222	0.185	0.389	0.181	0.573	0.364	0.319	0.571	0.463	0.387	0.328
Kaleido-Small	0.144	0.123	0.061	0.043	0.029	0.019	0.332	0.204	0.166	0.130	0.095	0.323	0.152	0.528	0.376	0.318	0.549	0.449	0.385	0.328
Kaleido-Medium	0.126	0.094	0.048	0.034	0.023	0.015	0.329	0.181	0.145	0.109	0.080	0.315	0.127	0.473	0.347	0.290	0.508	0.437	0.359	0.302
Kaleido	0.121	0.086	0.044	0.030	0.021	0.013	0.289	0.171	0.137	0.105	0.074	0.301	0.123	0.530	0.344	0.286	0.541	0.465	0.363	0.288
Results (SSIM↑)	0.850	0.884	0.908	0.918	0.927	0.935	0.518	0.585	0.611	0.633	0.657	0.384	0.602	0.282	0.377	0.395	0.342	0.385	0.427	0.452
Kaleido-Small	0.873	0.867	0.919	0.938	0.954	0.969	0.584	0.670	0.703	0.746	0.800	0.341	0.574	0.221	0.313	0.362	0.313	0.359	0.403	0.444
Kaleido-Medium	0.880	0.885	0.933	0.948	0.963	0.975	0.591	0.697	0.731	0.778	0.827	0.359	0.645	0.271	0.347	0.410	0.351	0.359	0.419	0.459
Kaleido	0.884	0.895	0.938	0.954	0.966	0.978	0.610	0.704	0.738	0.781	0.836	0.375	0.659	0.248	0.361	0.433	0.333	0.368	0.429	0.479

Table 6 Zero-shot SSIM/LPIPS Performance with Generative Methods. Kaleido achieves state-of-the-art performance across all object- and scene-level benchmarks, with SSIM and LPIPS metrics consistent with the superior PSNR performance reported in Table 2.