

# Video Numerical Continuation for Differentiable Inverse Rendering

HAOLIN LU, University of California San Diego, USA

ISHIT MEHTA, University of California San Diego, USA

HAODONG LI, University of California San Diego, USA

YU FU, University of California San Diego, USA

TZU-MAO LI, University of California San Diego, USA

RAVI RAMAMOORTHY, University of California San Diego, USA

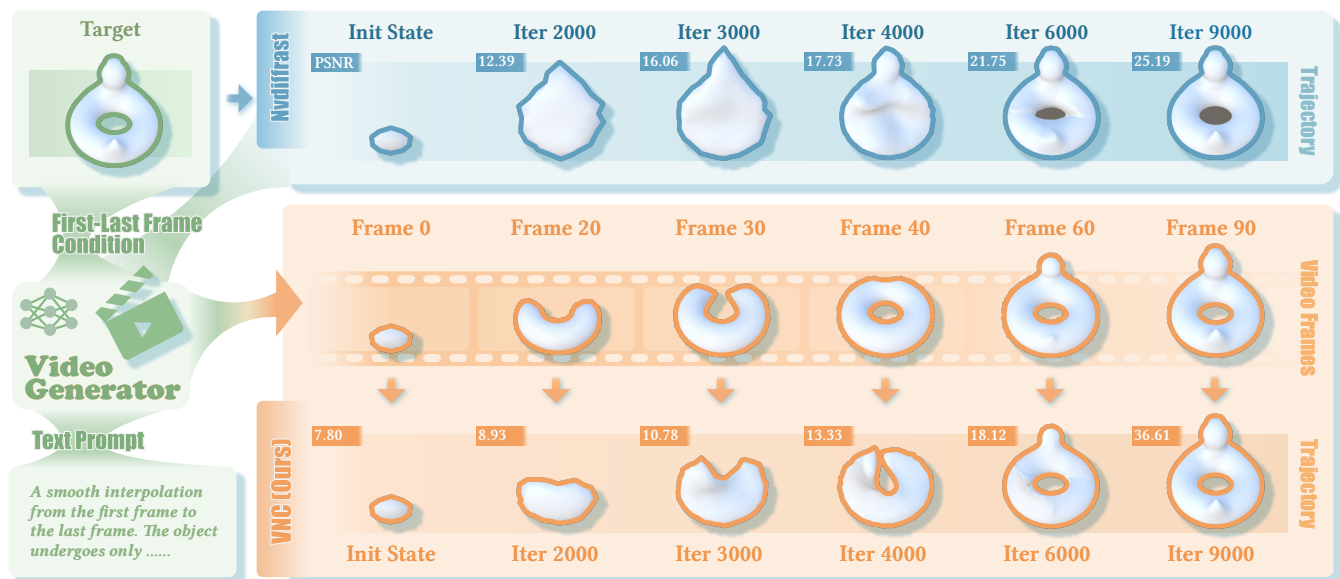


Fig. 1. VIDEO NUMERICAL CONTINUATION. **Top-left:** Given a target image of Bob [Crane 2014]\*, we aim to recover its geometry starting from a small polyhedron using a differentiable renderer Nvdiffrast [Laine et al. 2020]. **Blue block:** The baseline [Nicolet et al. 2021] optimizes the mean squared error between the current render and the target image. It struggles to model the hole and instead produces flipped mesh geometry to approximate the shading. Such suboptimal recovery is a common issue in inverse rendering across a wide range of settings, largely arising from non-convex loss landscapes and gradient sparsity [Fischer and Ritschel 2023; Mehta et al. 2025; Zhang et al. 2025]. **Green block:** Instead, we use a video generative model conditioned on the initial render and target image as first-last frames, along with a text prompt, to generate a sequence depicting a smooth transition between them. **Orange block:** We use each frame in the generated sequence as the target sequentially to guide the progressive reconstruction of the hole and yielding a correct topology within the same number of iterations. Our idea is broadly applicable beyond geometry optimization. We report Peak Signal-to-Noise Ratio values with respect to the target image at the top-left of each state. Although our method starts with lower PSNR (as it does not directly optimize against the final target), it ultimately achieves higher final quality. *“The longest way round is the shortest way home.”<sup>†</sup> — James Joyce*

While differentiable rendering enables gradient-based scene parameter recovery, it is frequently hampered by high non-convexity and poor local minima. We present a practical approach that circumvents these challenges

Authors’ Contact Information: Haolin Lu, hal128@ucsd.edu, University of California San Diego, USA; Ishit Mehta, ibmehta@ucsd.edu, University of California San Diego, USA; Haodong Li, hal211@ucsd.edu, University of California San Diego, USA; Yu Fu, yuf044@ucsd.edu, University of California San Diego, USA; Tzu-Mao Li, tzli@ucsd.edu, University of California San Diego, USA; Ravi Ramamoorthy, ravir@cs.ucsd.edu, University of California San Diego, USA.



This work is licensed under a Creative Commons Attribution 4.0 International License. SA Conference Papers ’26, Kuala Lumpur, Malaysia  
© 2026 Copyright held by the owner/author(s).  
ACM ISBN 979-8-4007-2842-6/2026/12  
<https://doi.org/10.1145/3829340.3842219>

by leveraging priors from foundation video models. Given initial and target images, we synthesize a smooth video trajectory and optimize scene parameters sequentially across its frames. This creates a gradually evolving objective that guides parameters along a tractable path, allowing the optimizer to escape local minima that stall standard methods. Conceptually, our approach acts as a form of numerical continuation, progressively deforming the objective to ensure robust convergence. We demonstrate its effectiveness across diverse inverse rendering tasks – including 6D pose estimation, lighting and PBR material recovery, geometry optimization, and skeleton optimization – successfully achieving convergence to the global optimum, from poor initializations where standard methods struggle.

CCS Concepts: • **Computing methodologies** → **Rendering**.

Additional Key Words and Phrases: inverse rendering, differentiable rendering, non-convex optimization, numerical continuation, video generation

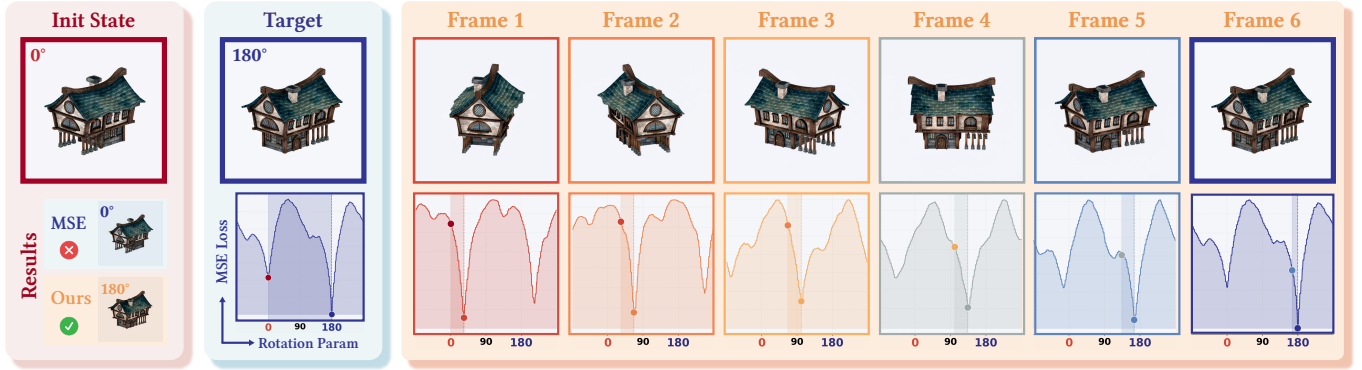


Fig. 2. ROTATION OPTIMIZATION. **First column:** We optimize the rotation of a house around the up-axis using a differentiable renderer [Laine et al. 2020], with image Mean Square Error as the objective. Optimization with MSE against the target image becomes trapped at the initial state, while our method converges to the correct rotation. **Blue block:** This occurs because the **initial** ( $0^\circ$ ) and **target** ( $180^\circ$ ) states exhibit strong visual similarity due to partial symmetry, making the initial state a local minimum, as shown in the loss landscape. Progress toward the global minimum would initially increase the loss, preventing standard gradient descent from escaping this minimum. **Orange block:** Instead of directly optimizing against the target image, we sequentially use intermediate frames from the generated video as targets. <sup>1</sup>This gradually reshapes the loss landscape so that each intermediate objective admits a nearby local minimum reachable by gradient descent from the current state. These minima progressively move toward the global minimum, enabling global convergence. For each step, we visualize the parameter updates under idealized gradient descent: the left marker denotes the parameter value before the update, while the right marker denotes the updated value. Over time, the trajectory does converge to the global minimum after all 6 frames.

#### ACM Reference Format:

Haolin Lu, Ishit Mehta, Haodong Li, Yu Fu, Tzu-Mao Li, and Ravi Ramamoorthi. 2026. Video Numerical Continuation for Differentiable Inverse Rendering. In *SIGGRAPH Asia 2026 Conference Papers (SA Conference Papers '26)*, December 01–04, 2026, Kuala Lumpur, Malaysia. ACM, New York, NY, USA, 11 pages. <https://doi.org/10.1145/3829340.3842219>

## 1 INTRODUCTION

Differentiable rendering is a powerful tool for recovering scene parameters from reference images via gradient-based optimization. However, the underlying inverse problem is usually highly non-convex, making optimization sensitive to initialization and prone to poor local minima [Fischer and Ritschel 2023; Mehta et al. 2025; Zhang et al. 2025]. To address this, we use a foundation video model to generate a video trajectory from the initial state to the target image, and sequentially optimize over its frames to guide the parameters to evolve smoothly toward the global optimum.

As a concrete example, consider optimizing a building’s yaw angle in Fig. 2. Due to the object’s near symmetry, the initial state forms a local minimum: small rotations in either direction increase the image MSE, causing standard inverse rendering methods [Laine et al. 2020] to remain trapped at the starting point. Instead, we synthesize a video trajectory depicting a smooth rotation from the initial state to the target. By optimizing sequentially over these intermediate frames, our rotation parameter tracks the rotation and escapes the local minimum, ultimately converging to the final reference image.

The visualized loss landscapes illustrate how our method works: starting with a target that is close to the initial state creates a local minimum that is easily accessible via gradient descent. As the target frames gradually change, this minimum shifts toward the

global optimum while remaining sufficiently close to make optimization tractable. The concept of guiding optimization with a smoothly evolving loss can be seen as a variant of numerical continuation [Allgower and Georg 2012].

In summary, our contributions are as follows:

- We propose a practical method for mitigating local minima in inverse rendering using foundation video models in a numerical continuation method.
- We show the effectiveness of our method across a range of tasks, including 6D and body pose estimation, lighting and material recovery, and mesh optimization. We also provide an analysis of the strengths and limitations of our method.
- We extend our method to challenging scenarios, such as material-lighting ambiguities and multi-view settings.

An open-source implementation of our method is available at <https://github.com/suikasibyl/vnc>.

## 2 BACKGROUND AND RELATED WORK

*Differentiable inverse rendering.* Given one or more reference images  $I_{\text{ref}}$ , inverse rendering aims to recover scene parameters  $\theta$  via a differentiable renderer  $I(\theta)$  by minimizing a loss  $\mathcal{L}$ :

$$\theta^* = \arg \min_{\theta} \mathcal{L}(I(\theta), I_{\text{ref}}). \quad (1)$$

Extensive research has focused on developing efficient differentiable renderers, based on either rasterization [de La Gorce et al. 2011; Laine et al. 2020; Liu et al. 2019; Loper and Black 2014; Pidhorskyi et al. 2024] or path tracing [Bangaru et al. 2020; Li et al. 2018; Loubet et al. 2019; Nimier-David et al. 2019; Zhang et al. 2020, 2023]. However, a key practical challenge is the non-convexity of the objective

<sup>\*</sup>Black backgrounds are removed here; full images are shown in the supplementary.

<sup>†</sup>Strictly speaking, our method is not taking the “longest” way around; the point is that the seemingly shortest path (directly minimizing the target) is not necessarily the best.



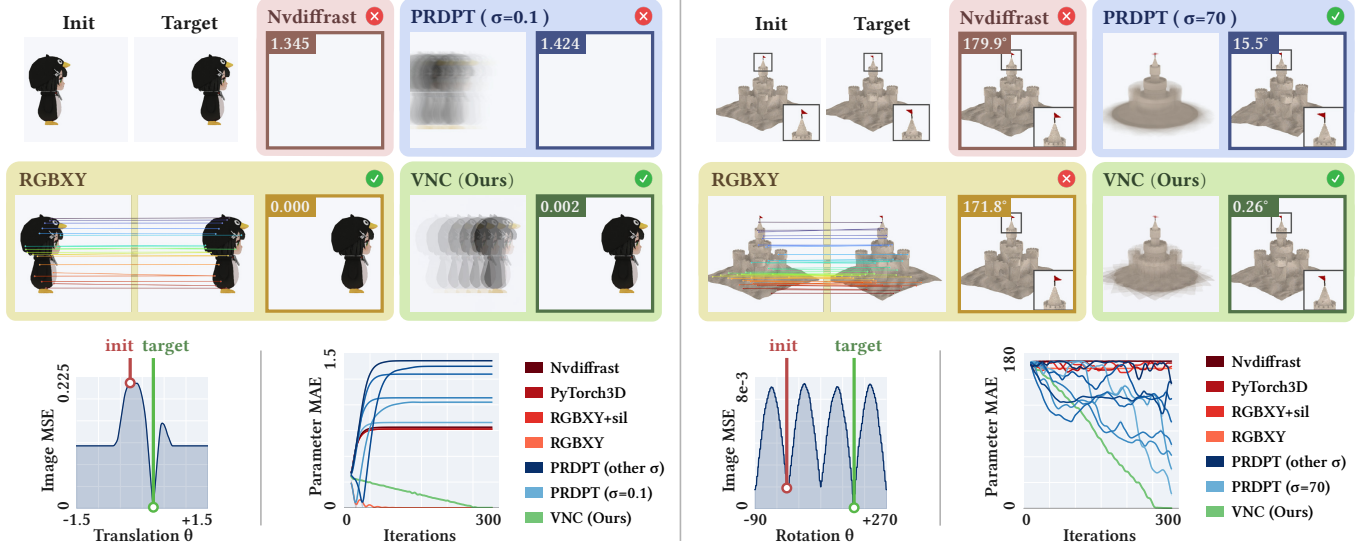


Fig. 3. INVERSE RENDERING WITH LOCAL MINIMA. Non-convexity is common in inverse rendering, as illustrated by two 1D examples. In PENGUINGIRL (left), optimizing the x-translation yields a local minimum when moving far to the left (out of view), since the image MSE also decreases. Moving right toward the target produces a flat plateau until the objects overlap, and both effects hinder optimization. SANDCASTLE (right), which exhibits approximate fourfold symmetry, has four local minima in yaw optimization.<sup>2</sup> We compare four methods under identical number of iterations and learning rates, reporting parameter mean absolute error (MAE) in the top-left labels. Nvdiffrastr [Laine et al. 2020] gets trapped in local minima due to its naive objective. PRDPT [Fischer and Ritschel 2023] smooths the loss, which can alleviate local minima and plateaus but still leads to unstable or suboptimal convergence. RGBXY [Xing et al. 2022] leverages pixel correspondences and works well for translation, but provides little improvement in the SANDCASTLE case. In contrast, our method generates a video trajectory (shown as overlays) and optimizes along it using Nvdiffrastr, escaping local minima and converging to the correct solution in both cases.

in Eq. (1), as shown in Fig. 3, which cannot be resolved simply by improving gradient estimation.

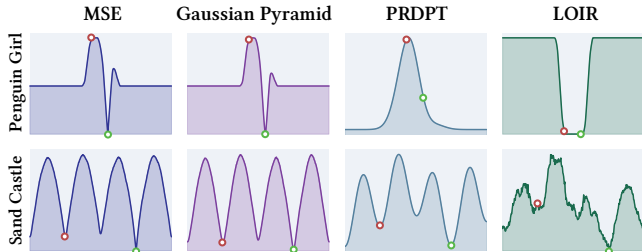


Fig. 4. LOSS SMOOTHING AND RESHAPING. We visualize the loss landscapes for the examples in Fig. 3 under different objectives, with **initial** and **target** parameters indicated by red and green markers, respectively. Gaussian pyramid and PRDPT filter the loss landscape in different ways, but may still preserve non-convex structures (second row) and can even blur out the global minimum (first row, third column). LOIR reshapes the first loss landscapes pretty well, but only makes the second landscape noisier.

**Variational optimization.** One workaround is variational optimization [Rechenberg and Eigen 1973; Spall 1992; Staines and Barber 2012], which replaces the original loss with a smoothed objective obtained by averaging over a parameter distribution:

$$\min_{\theta} \mathcal{L}(\mathbf{I}(\theta), \mathbf{I}_{\text{ref}}) \leq \min_{\theta'} \mathbb{E}_{\theta \sim p(\theta | \theta', \sigma)} [\mathcal{L}(\mathbf{I}(\theta), \mathbf{I}_{\text{ref}})], \quad (2)$$

<sup>1</sup>For better visualization, we uniformly pick six frames from the generated video here. In practice, all frames can be densely used for an even smoother evolution of the objective.

where the distribution  $p(\theta | \theta', \sigma)$  is typically Gaussian with mean  $\theta'$  and variance  $\sigma$ . This expectation defines a locally smoothed objective over  $\theta'$ , helping to escape local minima and plateaus. While this idea integrates well with inverse rendering [Deliot et al. 2024; Fischer and Ritschel 2023; Lidec et al. 2021], it has some limitations: gradient estimation is usually more expensive, the variance increases with problem dimensionality. Importantly, variational optimization does not necessarily improve global convergence as shown in Fig. 3.

**RGBXY derivatives.** Xing et al. [2022] introduced an alternative approach that mitigates the issue of convexity, by establishing correspondences between pixels in image space and directly minimizing pixel distances. While very effective when the correspondences are easy to identify, it struggles in more general settings where correspondences are ambiguous: e.g., under large rotations, significant lighting or shape variations, or when only weak texture or geometric cues are present. In comparison, our method is more generally applicable. Moreover, RGBXY relies on a rasterizer to track Lagrangian gradients and thus find it difficult to handle global illumination [Worchel and Alexa 2023] and volumetric effects like subsurface scattering.

**Gaussian Pyramid and LOIR.** Loper and Black [2014] use a Gaussian pyramid to smooth the loss landscape by filtering both renderings and targets in image space, while PRDPT [Fischer and Ritschel 2023] does that in parameter space. Mehta et al. [2025] further extend

<sup>2</sup>The entire castle undergoes a rigid rotation; we visualize only the top red flag in the insets, as its motion is the most visually apparent.

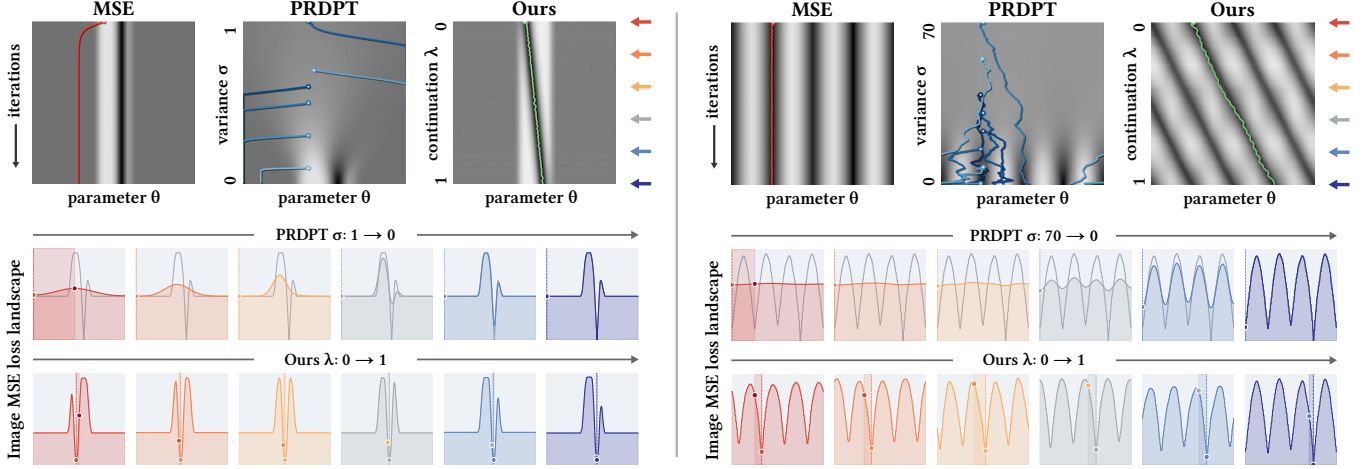


Fig. 5. **LOSS LANDSCAPE EVOLUTION.** We visualize how the loss landscape evolves during optimization for PENGUINGIRL (left) and SANDCASTLE (right), introduced in Fig. 3. **First row:** The figures show how the loss landscape evolves during optimization. The horizontal axis is the parameter  $\theta$ , and the vertical axis represents optimization progress (top: early iterations, bottom: later iterations). Each horizontal slice corresponds to the loss landscape at a given stage of optimization. The bright/dark regions indicate higher/lower loss values. The colored curve traces how the parameter  $\theta$  evolves over time during optimization. Direct optimization with MSE (first column) has a constant loss landscape, so the parameter just gets trapped in local minima. PRDPT [Fischer and Ritschel 2023] (second column) starts with a heavily smoothed landscape that gradually transitions back to the original objective. While this can help escape initial minima, it does not reliably guide the optimization toward the correct solution and often converges to different minima depending on the choice of  $\sigma$  (visualized by different blue trajectories). In contrast, our method forms a continuous trajectory (the dark tunnel) from the initial  $\theta$  to the target  $\theta^*$ , enabling stable convergence. **The second and third rows** show slices of the loss landscape at different stages. For PRDPT, the Gaussian-filtered loss landscape do not necessarily guide the optimization towards global minimum. In contrast, our method progressively transports the parameters toward the reference by gradually shifting the relevant minimum.

intensity matching to histogram matching to reshape the landscape. As shown in Fig. 4, these methods can soften the objective, but break down in more challenging cases, e.g. SANDCASTLE.

**Video models.** Recent video generation models [Blattmann et al. 2023; Brooks et al. 2024; Gao et al. 2025; HaCohen et al. 2026; Team et al. 2025; Wan et al. 2025; Wiedemer et al. 2025] can produce high-quality, temporally consistent sequences conditioned on text, images, or sparse frames. In particular, we use first-last-frame-to-video (FLF2V), which conditions generation on the initial and target frames to interpolate between them. This allows us to use the video model as a zero-shot reasoner [Liang et al. 2025; Tong et al. 2025; Wiedemer et al. 2025] to infer a plausible transformation trajectory, which is then used to guide optimization.

### 3 METHOD

We first formulate our approach as a form of numerical continuation, and later analyze why and when it works. The main challenge in Eq. (1) is its non-convexity, which makes optimization unreliable. We instead define a family of objectives  $\mathcal{L}(\theta, \lambda)$  where  $\theta \in \mathbb{R}^n$  and  $\lambda \in [0, 1]$ , and consider the corresponding minimizers  $\theta^*(\lambda)$ :

$$\theta^*(\lambda) = \arg \min_{\theta} \mathcal{L}(\theta, \lambda). \quad (3)$$

Here,  $\lambda = 0$  represents a simplified, more tractable version of the problem, while  $\lambda = 1$  recovers the original target objective. By incrementally increasing  $\lambda$  and using the solution from the preceding step to initialize the next optimization, we trace a continuous

trajectory of parameters,  $\theta^*(\lambda)$ , toward the final optimum. This design maintains tractability at each stage by guiding the optimization through a sequence of smoother sub-problems. Then, we define video numerical continuation, denoted as  $\mathcal{L}^{\text{vnc}}$ , as a specific instance of this kind of continuation objective family.

**First-last-frame-to-video generation.** For simplicity, consider a single-view inverse rendering problem with reference image  $\mathbf{I}_{\text{ref}}$ . We first render the initial image  $\mathbf{I}_{\text{init}}$ , and use the image pair  $(\mathbf{I}_{\text{init}}, \mathbf{I}_{\text{ref}})$  as first-last-frame conditioning to generate a video:

$$\{\mathbf{F}_0, \mathbf{F}_1, \dots, \mathbf{F}_N\} = \mathcal{G}(\mathbf{I}_{\text{init}}, \mathbf{I}_{\text{ref}}, T), \quad (4)$$

where  $\mathbf{F}_i$  are the generated frames,  $\mathcal{G}$  is a foundation video model that preserves the specified first and last frames, i.e.,  $\mathbf{F}_0 = \mathbf{I}_{\text{init}}$  and  $\mathbf{F}_N = \mathbf{I}_{\text{ref}}$ , and  $T$  is a task-specific text prompt describing the intermediate transition. Kling-V3-Omni [Team et al. 2025] is used in all experiments, as it is more stable for our tasks.

**Optimization with video numerical continuation.** With the generated video, we define our video numerical continuation loss as

$$\mathcal{L}^{\text{vnc}}(\theta, \lambda) = \mathcal{L}(\mathbf{I}(\theta), \mathbf{I}^*(\lambda)), \quad (5)$$

where  $\mathcal{L}$  can be any image loss, as defined in Eq. (1), and  $\mathbf{I}^*(\lambda)$  is the intermediate target image provided by the video trajectory:

$$\mathbf{I}^*(\lambda) = \begin{cases} \mathbf{I}_{\text{init}}, & \lambda = 0, \\ \mathbf{F}_{\lfloor \lambda N \rfloor}, & \lambda \in (0, 1), \\ \mathbf{I}_{\text{ref}}, & \lambda = 1. \end{cases} \quad (6)$$

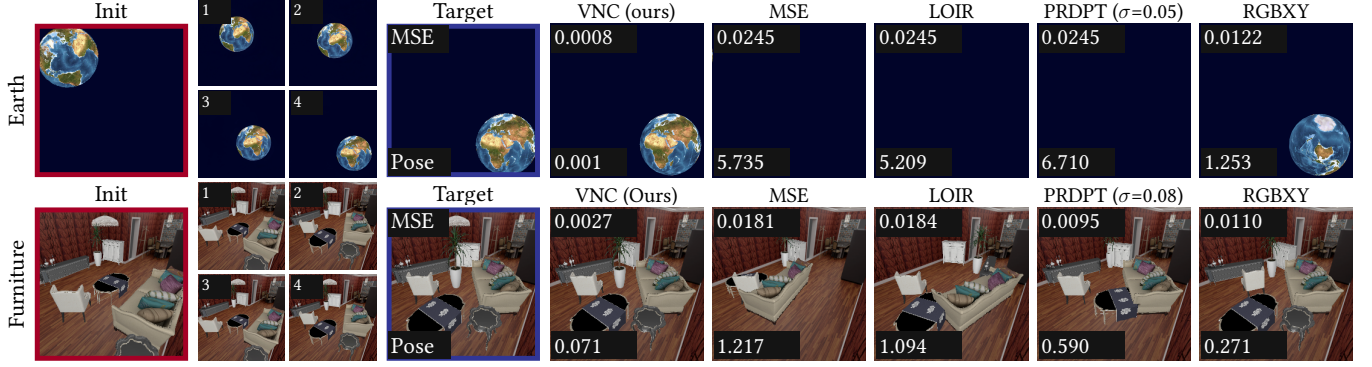


Fig. 6. 6D POSE OPTIMIZATION. We optimize the full 6D pose of a single Earth (top row) and the  $xz$  position and rotation of all furniture (bottom row). Four generated frames are shown in the second column. While both RGBXY and our method can stably recover positional alignment, only our method correctly estimates the EARTH’s rotation. In the FURNITURE case, our method fails to rotate the white cabinet in the back, while RGBXY fails to rotate the chair.

We linearly increase  $\lambda$  from 0 to 1 over the first  $\gamma$  fraction of iterations, then keep it at 1. This schedule ensures smooth evolution in the early stages and stable convergence in the final stage.

*Relation to variational optimization.* Variational optimization methods [Fischer and Ritschel 2023] typically anneal  $\sigma$  (often linearly) over iterations to facilitate convergence. This can also be interpreted as a form of numerical continuation, with objective

$$\mathcal{L}^{\text{vo}}(\theta', \lambda) = \int \mathcal{L}(\mathbf{I}(\theta), \mathbf{I}_{\text{ref}}) \cdot p(\theta | \theta', \sigma(\lambda)) d\theta, \quad (7)$$

where the variance  $\sigma$  is now parameterized by  $\lambda$ . Thus, we conduct a comparison between variational methods and ours in Fig. 5.

The key observation here is that the choice of objective  $\mathcal{L}(\theta, \lambda)$  is critical in numerical continuation. In practice,  $\mathcal{L}^{\text{vnc}}(\theta, \lambda)$  tends to be more reliable than  $\mathcal{L}^{\text{vo}}(\theta, \lambda)$  for reaching better optima. While  $\mathcal{L}^{\text{vo}}$  smooths local minima, it can also attenuate the global optimum and introduce new minima, making convergence sensitive to the choice of initial  $\theta$  and  $\sigma$ . In contrast,  $\mathcal{L}^{\text{vnc}}$  induces a smooth basin-like trajectory between the initial and target  $\theta$ , which is much easier to follow. This phenomenon can be interpreted as follows:

- (1) When the video generation works ideally, the generated frames implicitly define a trajectory of parameter states  $\theta^*(\lambda)$  over  $\lambda$ , describing the underlying dynamics. And ideally, each frame corresponds to the exact rendering of that state, i.e.,  $\mathbf{F}_i = \mathbf{I}(\theta^*(i/N))$ .
- (2) Given Eq. (5),  $\mathcal{L}^{\text{vnc}}(\theta^*(\lambda), \lambda) = 0$  for all  $\lambda$ , defining a continuous zero-set curve in the joint  $(\theta, \lambda)$  space. With  $\theta^*(0) = \theta^{\text{init}}$  and  $\theta^*(1) = \theta^*$ , the initial and target states lie at the endpoints of this curve. This trajectory explains what we observe in Fig. 5.
- (3) If the optimization follows the trajectory  $\theta^*(\lambda)$ , which is generally tractable since it only requires to keep moving to a nearby local optimum, the parameters  $\theta$  can naturally converge to  $\theta^*$ .

In practice, the ideal setting rarely holds, yet the approach remains empirically robust. Significant deviations from these assumptions can lead to failure cases, which we detail more in Section 5.1.

## 4 RESULTS AND APPLICATIONS

While the previous discussion focuses on single-parameter pose estimation examples for clarity, our approach generalizes to a wide

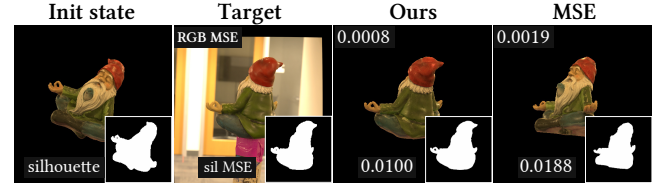


Fig. 7. REAL PHOTO POSE ESTIMATION. We optimize the 6D pose of GNOME (Stanford-ORB [Kuang et al. 2023]) from a masked real image, using NvdiffrMC [Hasselgren et al. 2022]. We optimize pose under RGB and silhouette MSE. The silhouette video is obtained by binarizing non-zero pixels from the generated RGB video. The baseline converges to incorrect poses with similar-looking masks, while our method recovers the correct pose.

range of inverse rendering tasks and is naturally compatible with different types of differentiable rendering algorithms.

We specify the text prompts for each task using `LABELS`, with the full prompts provided in the supplementary material. We avoid leaking any problem-specific parameters in the prompts; for example, in SANDCASTLE (Fig. 3), we do not specify a rotation of  $180^\circ$ , but instead use a generic instruction such as “only rotate”.

Our implementation uses Nvdiffrast [Laine et al. 2020] and PyTorch3D [Ravi et al. 2020] for rasterization, and Mitsuba 3 [Jakob et al. 2022] for path tracing. All experiments are conducted on an RTX 5090 GPU. Additional results, like video sequences, optimization procedure, and timing, are provided in the supplemental.

We compare our method against simple MSE, RGBXY [Xing et al. 2022], PRDPT [Fischer and Ritschel 2023], and LOIR [Mehta et al. 2025]. However, our approach is not mutually exclusive with these methods and can be complementary. In practice, we can simply use the generated video frames as target images while computing image losses under their respective formulations. The benefits of such combinations are discussed in Section 5.1.

### 4.1 Transformation optimization

*Single object pose estimation.* PENGUINGIRL and SANDCASTLE in Fig. 3 use `TRANSLATION` and `ROTATION` prompts, respectively.



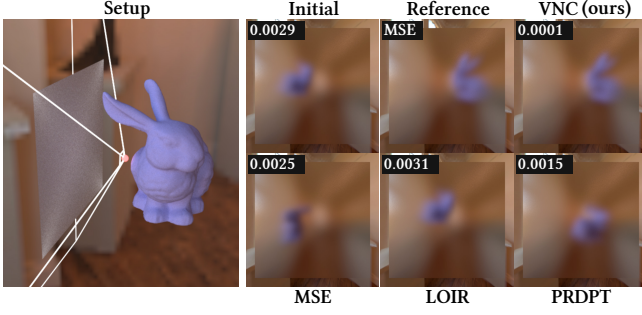


Fig. 8. INFERRING POSE FROM GLOBAL ILLUMINATION. We optimize the yaw rotation and  $xy$  position of BUNNY using only its reflection on a glossy surface ( $\alpha = 0.1$ ), with projective sampling [Zhang et al. 2023]. Since the initial and reference states also hardly overlap in reflection, baseline methods struggle in this setting, while our method succeeds under the video guidance.

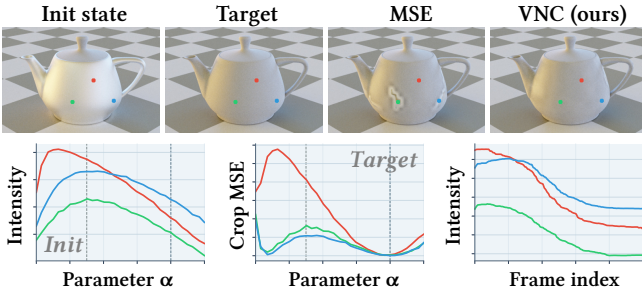


Fig. 9. ROUGHNESS TEXTURE OPTIMIZATION. We optimize a  $64^2$  texture for the alpha parameter of the teapot’s rough conductor material. Under environment lighting, the intensity is not monotonic with respect to  $\alpha$ . We visualize the color variation over  $\alpha$  for three  $5 \times 5$  crops marked in red, green, and blue. In the plots, the left vertical dashed line indicates the initial roughness, and the right indicates the target. The red shading point behaves monotonically over this range, so simple MSE converges. In contrast, the green and blue points have nearby local minima on the left, leading to brighter artifacts in those regions. With video guidance, our method overcomes this, as the green and blue points also increase in intensity in the early frames.

These prompts indicate only translation and rotation changes<sup>3</sup>. We present a full 6D pose optimization in Fig. 6 using the **RIGID** prompt, and show a real photo example in Fig. 7. More pose optimization results are provided in the supplementary material.

*Multi-object pose estimation.* Our method naturally extends to multiple objects, as shown in the second row of Fig. 6, where the video is also generated using the **RIGID** prompt. Since our approach operates only in image space, it scales well regardless of the number of objects or parameters involved. In contrast, variational optimization typically suffers from increased variance as the dimensionality of the parameter space grows.

*Pose estimation with global illumination.* Fig. 8 demonstrates that our method extends naturally to global illumination problems and a physically-based differentiable renderer [Jakob et al. 2022].

<sup>3</sup>This is reasonable, since this information is already known when setting up the optimizer, as baselines also optimize only translation and rotation parameters.

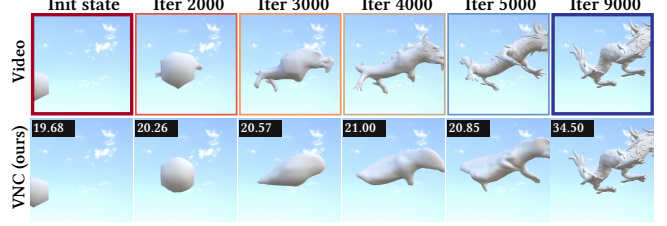


Fig. 10. GEOMETRY OPTIMIZATION. We optimize a DRAGON shape starting from an off-centered polyhedron. When the initial mesh does not overlap with the target, geometry optimization may move the shape far left out of view, similar to the pose estimation scenarios; this is the behavior observed with simple MSE optimization (not shown). In contrast, our method follows a generated morphing video and successfully recovers the final geometry.

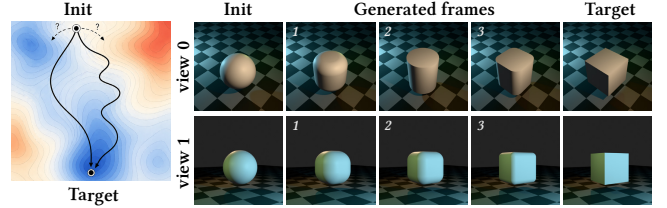


Fig. 11. MULTIVIEW CONSISTENCY. Consider morphing a sphere into a cube with two views. We generate the video for each view independently, so there is no guarantee that they represent a consistent progression. For example, view 0 may morph from a sphere to a cylinder and then to a cube, while view 1 transitions directly to a cube (these sequences are manually created in Blender with deliberate inconsistencies for illustration purposes). As a result, corresponding frames (e.g., frame 2) may not represent the same geometry across views. From a loss landscape perspective, this implies that two different parameter trajectories are formed. Although both connect the start and target points, their differing objectives can confuse the optimizer.

## 4.2 Material and light optimization

In Fig. 9, we show that our method can help overcome local minima in physically-based material recovery, such as roughness texture estimation. In Fig. 12, we present two examples of light optimization, where the light position is inferred from highlights and shadows. Our method accurately recovers the parameters across tasks.

## 4.3 Geometry optimization

*Single-view geometry optimization.* We show how our approach can be applied to geometry recovery, building on large steps inverse rendering [Nicolet et al. 2021]. As shown in Fig. 10, when the initialization is poor, the baseline falls into local minima by moving the geometry out of view, similar to the pose estimation case. In contrast, our method follows the generated transition (via the **MORPH** prompt) and successfully tracks the correct solution. Fig. 1 and Fig. 16 show two additional cases where video guidance enables topology to evolve in a feasible manner, using the prompts **BOBMORPH** and **ROPEMORPH**.

*Multi-view geometry optimization.* While we assume single-view optimization in previous experiments, it is generally insufficient for robust 3D geometry recovery, where multi-view supervision is often required. This poses an additional challenge for our method.



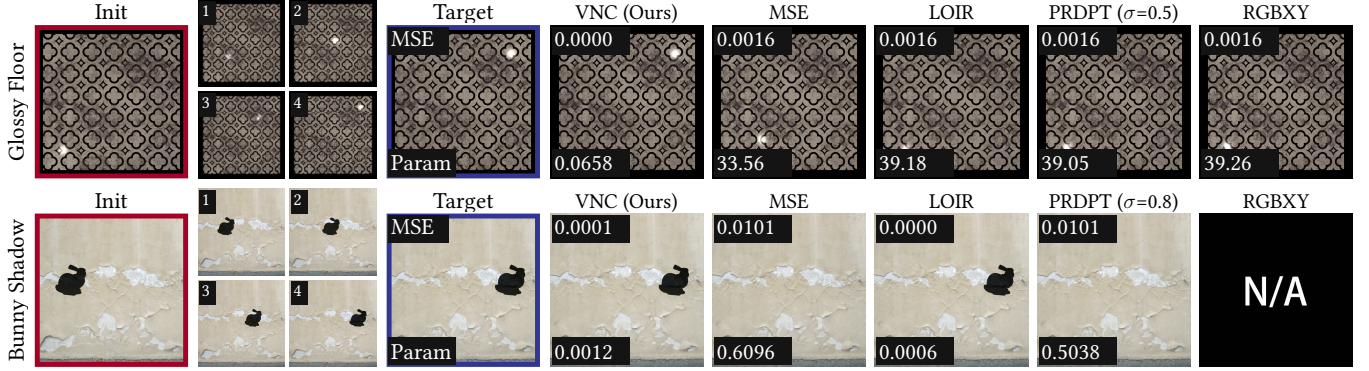


Fig. 12. **LIGHT POSE OPTIMIZATION.** We optimize point light positions in two examples. **Top row:** a glossy floor rendered with rasterization and Phong shading (shininess = 5000), where the light blob corresponds to the specular highlight of a point light. Other methods fail to optimize this case, whereas our method succeeds by following the smooth motion of the highlight. **Second row:** we optimize the light position from the shadow of a bunny cast onto a wall, with the bunny fixed behind the camera. RGBXY, which is limited to rasterization, cannot handle this scenario, while LOIR and our method succeed.

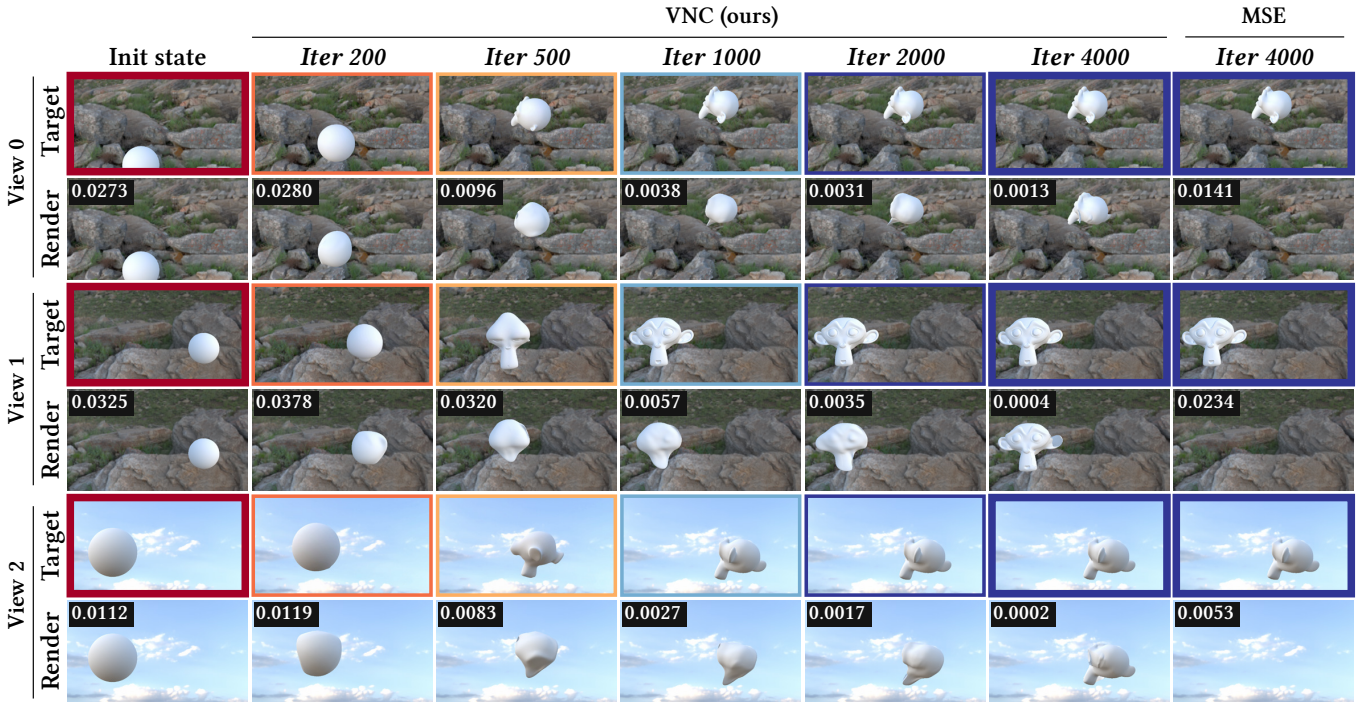


Fig. 13. **MULTIVIEW GEOMETRY OPTIMIZATION.** We optimize SUZANNE using large steps [Nicolet et al. 2021] with three-view observations. The initial shape (first column) is an off-centered sphere with no overlap with the target SUZANNE, causing MSE alone (last column) to diverge by moving the shape out of view. We generate one video per view independently and use the corresponding frames as targets during optimization (rows 1, 3, 5), while the optimized shapes across iterations are shown in rows 2, 4, and 6. Note that the three videos do not exhibit consistent progression across views. For example, at iteration 500, the ear is not yet visible in View 1 and the head appears triangular, while in View 0 the ear has already emerged but the head remains spherical. Nonetheless, this is sufficient to move the off-centered shape toward alignment with the target within the first 1000 iterations, followed by refinement over the next 3000 iterations. This requires a bit hyperparameter tuning though; here we use  $\gamma = 0.2$ , which deemphasizes inconsistent shape details and prioritizes overall translation.

Fig. 11 illustrates the key concern: although our approach can be extended to multi-view settings by generating frame sequences independently for each view, there is no guarantee that they correspond to the same trajectory  $\mathbf{I}^*(\lambda)$  across views. As a result, intermediate objectives may conflict across views, hindering the optimization.

However, we show two examples demonstrating that our method is tolerant to certain inconsistencies and still effective in overcoming local minima. Fig. 14 shows optimization of a B-spline curve using two shadow observations with projective sampling [Zhang et al. 2023]. While optimization with simple MSE also works, it often

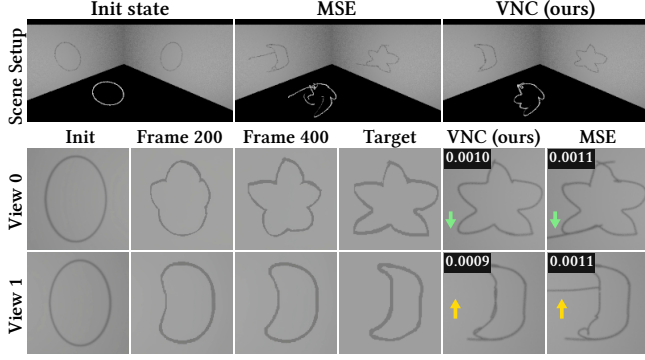


Fig. 14. **TWO-VIEW CURVE OPTIMIZATION.** **First row:** The scene consists of a B-spline curve casting two shadow projections onto two walls under a single light source. The curve is initialized as a ring, and its shape is optimized using MSE and our VNC. **Second and third row:** Optimization is performed under two-view supervision, where each view observes one shadow. For each view, we generate a video and jointly optimize using the corresponding frames. Our method achieves more stable convergence than MSE alone.

produces stretched and twisted shapes. In contrast, our method follows the independently generated objective transitions for each shadow, resulting in better solutions, since it effectively acts as a regularizer by constraining the transformation to progress smoothly.

Fig. 13 shows 3-view optimization of SUZANNE starting from an off-centered sphere with no initial overlap. We jointly optimize the geometry and global translation of the mesh. Although the per-view trajectories are not synchronized or fully consistent, they still help pull the shape out of local minima and lead to good convergence.

**Joint geometry–material optimization.** Joint recovery of material and geometry has been a vexing problem in inverse rendering. It introduces a specific type of local minimum arising from the inherent ambiguity between shading and geometry: a reference color can be explained either by surface geometry or by material properties.

In Fig. 15, we study such a joint optimization of object roughness and geometry using Nvdiffrast. We compute the shading using precomputed spherical harmonics coefficients [Ramamoorthi and Hanrahan 2001] with a von Mises–Fisher frequency filter controlled by a roughness parameter. A generated video in which shape and roughness change jointly (Video A) does not sufficiently resolve the underlying ambiguity. Instead, we adopt a prompt that generates a video to first adjust material properties and then performs shape morphing, effectively decomposing the problem into (1) simple roughness optimization and (2) simple shape optimization<sup>4</sup>, each with far less ambiguity. This results in significantly improved geometry reconstruction, as shown in the last row.

## 5 DISCUSSION AND FUTURE WORK

### 5.1 Limitations of our approach

Given the broad range of applications of our approach, now we systematically analyze its failure cases to highlight potential limitations and motivate future work. Building on the discussion in Section 3,

<sup>4</sup>This decomposition is conceptual only, we still perform a single optimization **without** staged parameter freezing, the only change is using the sequence as targets as before.

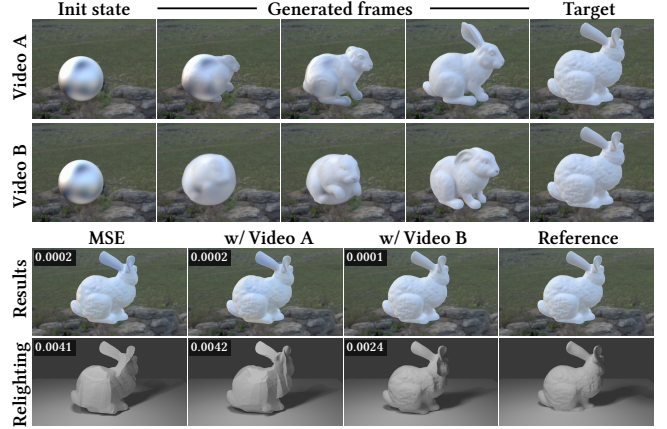


Fig. 15. **JOINT GEOMETRY-MATERIAL OPTIMIZATION.** We jointly optimize the shape and roughness (assumed constant over the mesh) from a single view. **First two rows:** We visualize two generated sequences with the same first and last frames but different text prompts. **Video A** uses the `JOINTMORPH` prompt, where shape and roughness change jointly, while **Video B** uses `MATTHENMORPH`, which first changes the roughness and then performs shape morphing. **Last two rows:** We compare optimization results with simple MSE and our method using the two videos. Although all three approaches partially match the reference view, our method with Video B yields significantly better geometry, which is more evident in relit renderings.

the failure cases of our method can be categorized as violations of its underlying assumptions, including:

- (1) The video may imply an incorrect trajectory,  $\theta^*(\lambda)$  – for example, one that includes states not representable under the current parameterization (e.g., changing shape in a pose-only optimization), or that evolves in a way that does not lead to the target state correctly (e.g., rotating  $270^\circ$  when  $180^\circ$  is required).
- (2) Even if the trajectory is plausible, the frames may not accurately render the implied states, leading to incorrect gradients.
- (3) Even if all frames are valid, the optimizer may still fail to follow.

We now discuss common scenarios in which these issues may arise:

**Emergence of unobserved content.** Although modern video models are quite powerful, a common cause of (1) arises when the initial and target renders do not sufficiently constrain the intermediate trajectory. In such cases, the generative model must hallucinate missing content, which may be inconsistent with the true reference. For example, in Fig. 17, only Eurasia and the Americas are visible from the first-last frames, while Oceania is hallucinated and does not match the true texture, thereby hindering optimization.

**Multiview reference targets.** Another common cause of (1) arises in multi-view settings, as discussed in Fig. 11. While we show that, with careful choosing of the video ratio  $\gamma$  and learning rate, even inconsistent videos can still be beneficial, having a consistent multi-view video model [Li et al. 2024; Wu et al. 2024] that can condition on multi-first–last frames is a promising direction for future work.

**Low and high dynamic range.** For (2), although generative models are often considered to struggle with reproducing accurate rendering, conditioning on first–last frames allows them to handle diverse

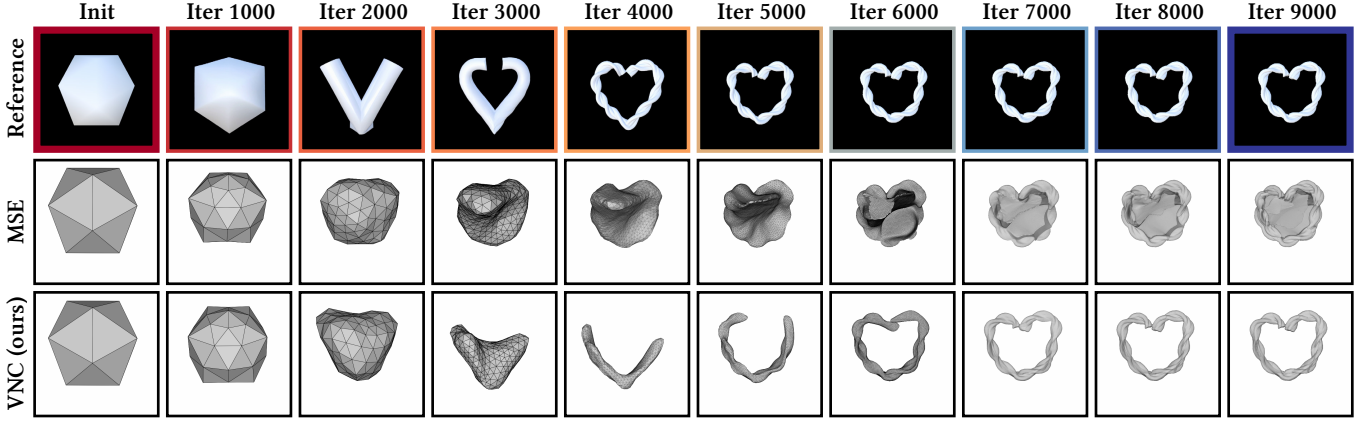


Fig. 16. GUIDING TOPOLOGICAL CORRECTION. We optimize the geometry of a heart formed by a rope, initialized from a low-poly polyhedron that fully covers the target shape. Unlike Fig. 1, the rope here only connects at its two ends and therefore shares the same topology as the initial shape. However, even in this case, standard inverse rendering methods fail to obtain meaningful gradients for the interior hole [Mehta et al. 2023; Sharp and Crane 2018]. Instead, we use a dedicated prompt, [\(ROPEMORPH\)](#), to generate a video in which the shape first transforms into a V-shape, which is easy to optimize, and then gradually closes into the final heart shape through another simple transition. Under this guidance, our method successfully recovers the rope structure as intended.

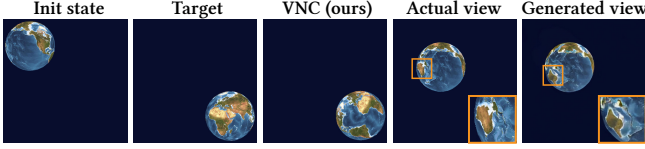


Fig. 17. EMERGENCE OF UNOBSERVED CONTENT. If we optimize the same Earth as in Fig. 6 but start from a larger rotation offset, our method may converge to an incorrect rotation. This occurs because here regions like Oceania are not visible in either the initial or target render. The video generator therefore hallucinates these unseen regions with inconsistent shapes relative to the reference, which prevents the MSE loss from correctly aligning the rotation.

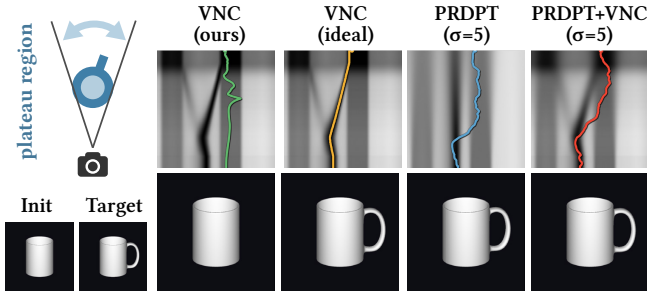


Fig. 18. UNRESOLVED PLATEAU. When optimizing the mug’s rotation with the handle initially occluded, video guidance alone cannot provide a useful gradient. Although a valid parameter path exists (yellow curve), the plateau region (gray band) prevents the optimization from following it, causing it to get stuck (green curve). Combining our approach with PRDPT smooths the plateau in early stages and enables escape to the correct path (red curve).

shading effects fairly well, as demonstrated in Figs. 9 and 15. However, a practical limitation is that current video models typically support only low dynamic range sequences, while many shading effects can easily exceed this range. Potential solutions include tone mapping, multiple exposures, and the development of native high dynamic range video models [Korem et al. 2026].

*Unstable gradient.* One cause of (3) is instability in the gradient itself. For example, in Fig. 19, we observe the skeleton optimization with MSE loss is unstable and can diverge even when the target is relatively close. Our method alone does not resolve this inherent instability. Instead, we find that RGBXY can provide a stable gradient when the target is close, and combining it with our method enables more reliable optimization in more complex scenarios.

*Unresolved loss plateau.* Another cause of (3) is that our continuation over frames may still leave unresolved loss plateaus. As shown in Fig. 18, infinitesimal perturbations of the mug do not change the rendered image, resulting in zero gradients by definition. Combining our approach with PRDPT helps overcome this issue.

## 5.2 Influence of video foundation models

We also study the instability caused by the inherent stochasticity and limited prompt adherence of current video models, although this class of limitations is expected to diminish as the models continue to improve. More results are provided in the supplementary material.

*Sampling randomness.* Diffusion-based video generation is inherently stochastic, so repeated rollouts can produce different videos. We study the effect of this randomness by generating 10 videos with the same prompt using the Kling API for the SANDCASTLE example (Fig. 3). Our method converges in 8/10 trials (rotation error  $\leq 2^\circ$ ). This task is particularly challenging, as it relies on video priors to infer a  $180^\circ$  rotation; thus, baseline methods rarely converge across random seeds and hyperparameter settings as shown in Table 1.

*Prompt sensitivity.* We also investigate the effect of prompt variation on the 6D pose estimation task EARTH (Fig. 15). We use ChatGPT to generate 10 variants of the rigid-body motion prompt, covering simpler or more descriptive wording, English and simplified Chinese, and prompts with or without negative prompting. Using the first generated video and a single inverse-rendering run for each prompt, our method converges in 9/10 trials. In comparison, we



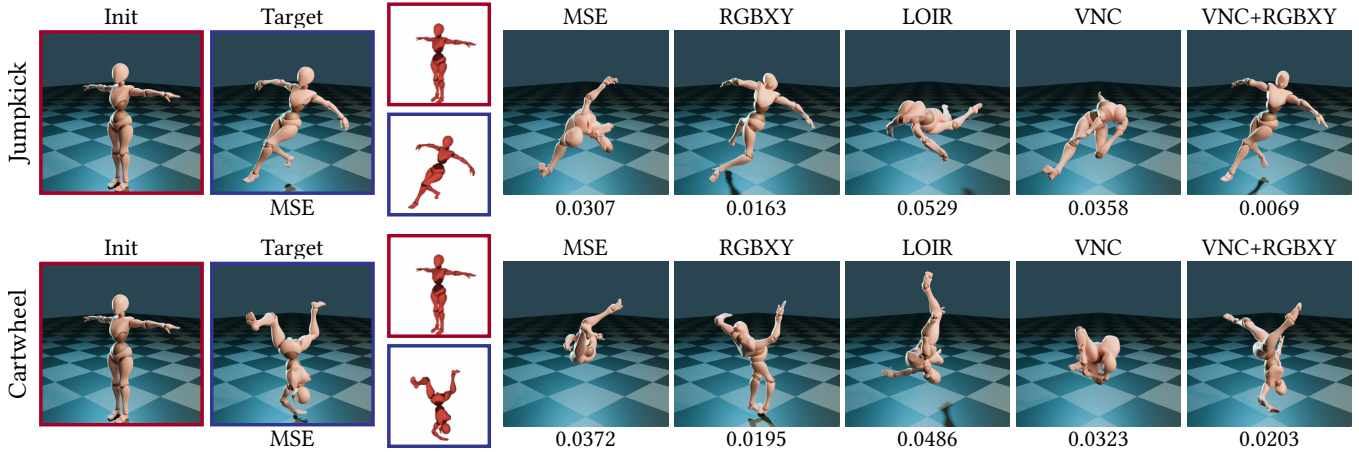


Fig. 19. SKELETAL POSE ESTIMATION. We optimize articulated skeletal pose parameters (axis-angle joint rotations) and a global rotation to match a single-view reference image from a breakdance sequence. The optimization and MSE are computed using simple rasterization (third column), while path-traced renderings are shown only for visualization. MSE loss with Nvdiffrast provides unstable gradients for this challenging task. While RGBXY is more stable in simple cases, it fails on complex targets – for example, in JUMPICK, the legs cross, and in CARTWHEEL, the character flips upside down, confusing correspondence matching. Our method with MSE loss alone is also unstable, but when combined with RGBXY, it helps find correct correspondences and better handles both cases.

Table 1. CONVERGENCE RATES OVER MULTIPLE RUNS. We repeat video generation 10 times with the same input conditions for our method and evaluate the baselines across different random seeds and hyperparameter settings. Our method successfully recovers the rotation parameter for SANDCASTLE in most trials, whereas the baseline methods rarely converge.

|           | MSE  | LoIR | RGBXY | PRDPT<br>( $\sigma = 1$ ) | PRDPT<br>(other $\sigma$ ) | VNC<br>(Ours) |
|-----------|------|------|-------|---------------------------|----------------------------|---------------|
| Converged | 0/10 | 1/10 | 0/10  | 3/10                      | 0/70                       | <b>8/10</b>   |

repeat each baseline with 10 random seeds, and none converges in any trial. More details are provided in the supplementary material.

*Random initialization.* To evaluate robustness to initialization, we test 10 random light-pose initializations on GLOSSYFLOOR (Fig. 12). Our method converges in 9/10 trials, compared with 3/10 for MSE. While a good initialization can help MSE avoid local minima, our method may also benefit from better initializations, as the corresponding videos are easier to generate and less prone to errors. More importantly, searching for good initializations exhaustively becomes prohibitive in higher-dimensional parameter spaces, making our approach’s greater tolerance to initialization particularly valuable.

*Foundation models.* We investigate how different video foundation models affect our approach, as shown in Table 2.<sup>5</sup> Although weaker models such as WAN2.1-FLF2V-14B exhibit poorer prompt adherence and overall generation quality, they still improve over the baselines on the evaluated tasks.

*Timing.* By default, our results use videos generated via the Kling API, whose response time ranges from 90.78 s to 175.93 s, with an average of 135.81 s over 43 API calls. A detailed runtime comparison is provided in the supplementary material.

<sup>5</sup>Due to API budget limitations, some models are evaluated over 5 runs instead of 10.

Table 2. CONVERGENCE RATES USING DIFFERENT FOUNDATION MODELS. We repeat the experiments using different video foundation models. KLING 3.0 OMNI and SEEDANCE 2.0 achieve the highest convergence rates, while WAN 2.1 and WAN 2.7 perform worse, likely due to their weaker overall quality. Nevertheless, our approach consistently outperforms the MSE baseline. While the baseline fails to converge in every trial, our method can still achieve convergence in some cases even with weaker video models.

|             | KLING        | SEEDANCE2.0 | WAN2.7 | WAN2.1 | MSE  |
|-------------|--------------|-------------|--------|--------|------|
| SANDCASTLE  | <b>8/10</b>  | 3/5         | 0/5    | 2/10   | 0/10 |
| GLOSSYFLOOR | <b>10/10</b> | 5/5         | 3/5    | 3/10   | 0/10 |

## 6 CONCLUSION

We present video numerical continuation, a simple yet effective approach for guiding inverse rendering toward global minima. Owing to its simplicity, it is naturally compatible with a wide range of inverse rendering algorithms and applications; the only modification required is the reference image. We analyze its underlying assumptions and explain why it works, and demonstrate its versatility across diverse applications. We also discuss potential limitations, but anticipate broad applicability, especially with further advances in foundation models and integration with other image-space losses.

## Acknowledgments

This paper was supported in part by NSF grants 2238839, 2341952, 2402583, 2100237, 2120019, a UCSD Fellowship, gifts from Qualcomm, Google Adobe, Activision, and Meshy, the Ronald L. Graham Chair and the UC San Diego Center for Visual Computing. Scene assets are courtesy of Keenan Crane, Ziyi Zhang, Sketchfab users patrakeevasveta, Serafim123, Maki, ReedMan, LightSwitch, bogdan-zloy20280, and Sh0ma.



## References

- Eugene L Allgower and Kurt Georg. 2012. *Numerical continuation methods: an introduction*. Springer Science & Business Media.
- Sai Praveen Bangaru, Tzu-Mao Li, and Frédo Durand. 2020. Unbiased Warped-Area Sampling for Differentiable Rendering. *ACM Trans. Graph. (Proc. SIGGRAPH Asia)* 39, 6 (2020), 245:1–245:18.
- Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. 2023. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127* (2023).
- Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, Clarence Ng, Ricky Wang, and Aditya Ramesh. 2024. Video generation models as world simulators. (2024). <https://openai.com/research/video-generation-models-as-world-simulators>
- Keenan Crane. 2014. Keenan’s 3D Model Repository. <https://www.cs.cmu.edu/~kmc Crane/Projects/ModelRepository/>. Accessed: 2026-05-04.
- Martin de La Gorce, David J Fleet, and Nikos Paragios. 2011. Model-based 3D hand pose estimation from monocular video. *IEEE Trans. Pattern Anal. Mach. Intell.* 33, 9 (2011), 1793–1805.
- Thomas Deliot, Eric Heitz, and Laurent Belcour. 2024. Transforming a Non-Differentiable Rasterizer into a Differentiable One with Stochastic Gradient Estimation. *Proc. ACM Comput. Graph. Interact. Tech.* 7, 1, Article 3 (May 2024), 16 pages. doi:10.1145/3651298
- Michael Fischer and Tobias Ritschel. 2023. Plateau-Reduced Differentiable Path Tracing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4285–4294.
- Yu Gao, Haoyuan Guo, Tuyen Hoang, Weilin Huang, Lu Jiang, Fangyuan Kong, Huixia Li, Jiashi Li, Liang Li, Xiaojie Li, et al. 2025. Seedance 1.0: Exploring the boundaries of video generation models. *arXiv preprint arXiv:2506.09113* (2025).
- Yoav HaCohen, Benny Brazowski, Nisan Chiprut, Yaki Bitterman, Andrew Kvochko, Avishai Berkowitz, Daniel Shalem, Daphna Lifschitz, Dudu Moshe, Eitan Porat, et al. 2026. LTX-2: Efficient Joint Audio-Visual Foundation Model. *arXiv preprint arXiv:2601.03233* (2026).
- Jon Hasselgren, Nikolai Hofmann, and Jacob Munkberg. 2022. Shape, Light, and Material Decomposition from Images using Monte Carlo Rendering and Denoising. *arXiv:2206.03380* (2022).
- Wenzel Jakob, Sébastien Speierer, Nicolas Roussel, Merlin Nimier-David, Delio Vicini, Tizian Zeltner, Baptiste Nicolet, Miguel Crespo, Vincent Leroy, and Ziyi Zhang. 2022. *Mitsuba 3 renderer*. <https://mitsuba-renderer.org>.
- Naomi Ken Korem, Mohamed Oumoumad, Harel Cain, Matan Ben Yosef, Urska Jerelcic, Ofir Bibi, Yaron Inger, Or Patashnik, and Daniel Cohen-Or. 2026. HDR Video Generation via Latent Alignment with Logarithmic Encoding. *arXiv:2604.11788* [cs.CV] <https://arxiv.org/abs/2604.11788>
- Zhengfei Kuang, Yunzhi Zhang, Hong-Xing Yu, Samir Agarwala, Shangzhe Wu, Jiajun Wu, et al. 2023. Stanford-ORB: a real-world 3D object inverse rendering benchmark. *Advances in Neural Information Processing Systems Datasets and Benchmarks Track*.
- Samuli Laine, Janne Hellsten, Tero Karras, Yeongho Seol, Jaakko Lehtinen, and Timo Aila. 2020. Modular Primitives for High-Performance Differentiable Rendering. *ACM Transactions on Graphics* 39, 6 (2020).
- Bing Li, Cheng Zheng, Wenxuan Zhu, Jinjie Mai, Biao Zhang, Peter Wonka, and Bernard Ghanem. 2024. Vivid-ZOO: Multi-View Video Generation with Diffusion Model. *arXiv:2406.08659*
- Tzu-Mao Li, Miika Aittala, Frédo Durand, and Jaakko Lehtinen. 2018. Differentiable Monte Carlo Ray Tracing through Edge Sampling. *ACM Trans. Graph. (Proc. SIGGRAPH Asia)* 37, 6 (2018), 222:1–222:11.
- Junbang Liang, Pavel Tokmakov, Ruoshi Liu, Sruthi Sudhakar, Paarth Shah, Rares Ambrus, and Carl Vondrick. 2025. Video generators are robot policies. *arXiv preprint arXiv:2508.00795* (2025).
- Quentin Le Lidec, Ivan Laptev, Cordelia Schmid, and Justin Carpentier. 2021. Differentiable rendering with perturbed optimizers. In *Proceedings of the 35th International Conference on Neural Information Processing Systems (NIPS ’21)*. Curran Associates Inc., Red Hook, NY, USA, Article 1560, 12 pages.
- Shichen Liu, Tianye Li, Weikai Chen, and Hao Li. 2019. Soft Rasterizer: A Differentiable Renderer for Image-based 3D Reasoning. *International Conference on Computer Vision* (2019).
- Matthew M. Loper and Michael J. Black. 2014. OpenDR: An Approximate Differentiable Renderer. In *European Conference on Computer Vision*, Vol. 8695. ACM, 154–169.
- Guillaume Loubet, Nicolas Holzschuch, and Wenzel Jakob. 2019. Reparameterizing discontinuous integrands for differentiable rendering. *ACM Trans. Graph. (Proc. SIGGRAPH Asia)* 38, 6 (2019), 228.
- Isht Mehta, Manmohan Chandraker, and Ravi Ramamoorthi. 2023. A Theory of Topological Derivatives for Inverse Rendering of Geometry. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 419–429.
- Isht Mehta, Manmohan Chandraker, and Ravi Ramamoorthi. 2025. Locally Orderless Images for Optimization in Differentiable Rendering. In *IEEE Conference on Computer Vision and Pattern Recognition*.
- Baptiste Nicolet, Alec Jacobson, and Wenzel Jakob. 2021. Large Steps in Inverse Rendering of Geometry. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 40, 6 (Dec. 2021). doi:10.1145/3478513.3480501
- Merlin Nimier-David, Delio Vicini, Tizian Zeltner, and Wenzel Jakob. 2019. Mitsuba 2: A retargetable forward and inverse renderer. *ACM Trans. Graph. (Proc. SIGGRAPH Asia)* 38, 6 (2019), 1–17.
- Stanislav Pidhorskyi, Tomas Simon, Gabriel Schwartz, He Wen, Yaser Sheikh, and Jason Saragih. 2024. Rasterized Edge Gradients: Handling Discontinuities Differentiably. *arXiv:2405.02508* [cs.CV] <https://arxiv.org/abs/2405.02508>
- Ravi Ramamoorthi and Pat Hanrahan. 2001. An Efficient Representation for Irradiance Environment Maps. In *SIGGRAPH*. ACM Press/Addison-Wesley Publishing Co., 497–500.
- Nikhila Ravi, Jeremy Reizenstein, David Novotny, Taylor Gordon, Wan-Yen Lo, Justin Johnson, and Georgia Gkioxari. 2020. Accelerating 3D Deep Learning with Py-Torch3D. *arXiv:2007.08501* [cs.CV] <https://arxiv.org/abs/2007.08501>
- Ingo Rechenberg and Manfred Eigen. 1973. *Evolutionstrategie: Optimierung technischer Systeme nach Prinzipien der biologischen Evolution*. Frommann-Holzboog Verlag.
- Nicholas Sharp and Keenan Crane. 2018. Variational Surface Cutting. *ACM Trans. Graph. (Proc. SIGGRAPH)* 37, 4 (2018), 156:1–156:13.
- James C Spall. 1992. Multivariate stochastic approximation using a simultaneous perturbation gradient approximation. *IEEE transactions on automatic control* 37, 3 (1992), 332–341.
- Joe Staines and David Barber. 2012. Variational optimization. *arXiv preprint arXiv:1212.4507* (2012).
- Kling Team, Jialu Chen, Yuanzheng Ci, Xiangyu Du, Zipeng Feng, Kun Gai, Sainan Guo, Feng Han, Jingbin He, Kang He, Xiao Hu, Xiaohua Hu, Boyuan Jiang, Fangyuan Kong, Hang Li, Jie Li, Qingyu Li, Shen Li, Xiaohan Li, Yan Li, Jiajun Liang, Borui Liao, Yiqiao Liao, Weihong Lin, Quande Liu, Xiaokun Liu, Yilun Liu, Yuliang Liu, Shun Lu, Hangyu Mao, Yunyao Mao, Haodong Ouyang, Wenyu Qin, Wanqi Shi, Xiaoyu Shi, Lianghao Su, Haozhi Sun, Peiqin Sun, Pengfei Wan, Chao Wang, Chenyu Wang, Meng Wang, Qulin Wang, Runqi Wang, Xintao Wang, Xuebo Wang, Zekun Wang, Min Wei, Tiancheng Wen, Guohao Wu, Xiaoshi Wu, Zhenhua Wu, Da Xie, Yinglong Xiong, Yulong Xu, Sile Yang, Zikang Yang, Weicai Ye, Ziyang Yuan, Shenglong Zhang, Shuaiyu Zhang, Yuanxing Zhang, Yufan Zhang, Wenzheng Zhao, Ruiliang Zhou, Yan Zhou, Guosheng Zhu, and Yongjie Zhu. 2025. Kling-Omni Technical Report. *arXiv:2512.16776* [cs.CV] <https://arxiv.org/abs/2512.16776>
- Jingqi Tong, Yurong Mou, Hangcheng Li, Mingzhe Li, Yongzhao Yang, Ming Zhang, Qiguang Chen, Tianyi Liang, Xiaomeng Hu, Yining Zheng, et al. 2025. Thinking with video: Video generation as a promising multimodal reasoning paradigm. *arXiv preprint arXiv:2511.04570* (2025).
- Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwei Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, Lianghua Huang, Mengyang Feng, Ningyi Zhang, Pandeng Li, Pingyu Wu, Ruihang Chu, Ruili Feng, Shiwei Zhang, Siyang Sun, Tao Fang, Tianxing Wang, Tianyi Gui, Tingyu Weng, Tong Shen, Wei Lin, Wei Wang, Wei Wang, Wenmeng Zhou, Wente Wang, Wenting Shen, Wenyuan Yu, Xianzhong Shi, Xiaoming Huang, Xin Xu, Yan Kou, Yangyu Lv, Yifei Li, Yijing Liu, Yiming Wang, Yingya Zhang, Yitong Huang, Yong Li, You Wu, Yu Liu, Yulin Pan, Yun Zheng, Yuntao Hong, Yupeng Shi, Yutong Feng, Zeyinzi Jiang, Zhen Han, Zhi-Fan Wu, and Ziyu Liu. 2025. Wan: Open and Advanced Large-Scale Video Generative Models. *arXiv:2503.20314* [cs.CV] <https://arxiv.org/abs/2503.20314>
- Thaddäus Wiedemer, Yuxuan Li, Paul Vicol, Shixiang Shane Gu, Nick Matarese, Kevin Swersky, Been Kim, Priyank Jaini, and Robert Geirhos. 2025. Video models are zero-shot learners and reasoners. *arXiv:2509.20328* [cs.LG] <https://arxiv.org/abs/2509.20328>
- Markus Worchel and Marc Alexa. 2023. Differentiable Shadow Mapping for Efficient Inverse Graphics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 142–153.
- Rundi Wu, Ruiqi Gao, Ben Poole, Alex Trevithick, Changxi Zheng, Jonathan T. Barron, and Aleksander Holynski. 2024. CAT4D: Create Anything in 4D with Multi-View Video Diffusion Models. *arXiv:2411.18613* (2024).
- Jiankai Xing, Fujun Luan, Ling-Qi Yan, Xuejun Hu, Houde Qian, and Kun Xu. 2022. Differentiable Rendering using RGBXY Derivatives and Optimal Transport. *ACM Trans. Graph.* 41, 6, Article 189 (dec 2022), 13 pages. doi:10.1145/3550454.3555479
- Cheng Zhang, Bailey Miller, Kai Yan, Ioannis Gkioulekas, and Shuang Zhao. 2020. Path-Space Differentiable Rendering. *ACM Trans. Graph. (Proc. SIGGRAPH)* 39, 6 (2020), 143:1–143:19.
- Ziyi Zhang, Nicolas Roussel, and Wenzel Jakob. 2023. Projective Sampling for Differentiable Rendering of Geometry. *Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 42, 6 (Dec. 2023). doi:10.1145/3618385
- Ziyi Zhang, Nicolas Roussel, and Wenzel Jakob. 2025. Many-Worlds Inverse Rendering. *ACM Trans. Graph.* (Sept. 2025). doi:10.1145/3767318