

EasyOmnimatte: Taming Pretrained Inpainting Diffusion Models for End-to-End Video Layered Decomposition

Yihan Hu¹

Xuelin Chen²

Xiaodong Cun^{1,†}

¹ GVC Lab, Great Bay University

² Adobe Research

<https://github.com/GVCLab/EasyOmnimatte>

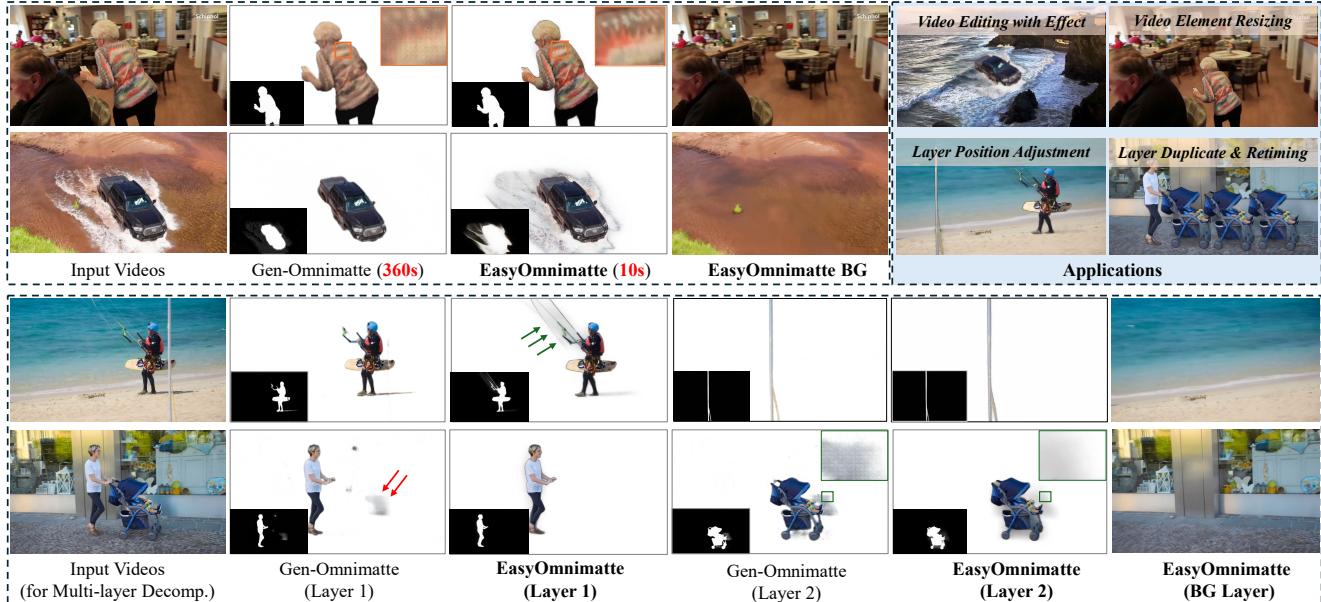


Figure 1. **EasyOmnimatte** is the first, end-to-end video omnimatte method, eliminating the need for slow, multi-stage optimization-based pipelines (e.g., Gen-Omnimatte [15]). EasyOmnimatte produces high-fidelity alpha mattes that accurately capture the associated effects, *all within just seconds*, representing a significant gain over existing state-of-the-art methods in both quality and efficiency.

Abstract

Existing video omnimatte methods typically rely on slow, multi-stage, or inference-time optimization pipelines that fail to fully exploit powerful generative priors, producing suboptimal decompositions. Our key insight is that, if a video inpainting model can be finetuned to remove the foreground-associated effects [15], then it must be inherently capable of perceiving these effects, and hence can also be finetuned for the complementary task: foreground layer decomposition with associated effects. However, although naively finetuning the inpainting model with LoRA applied to all blocks can produce high-quality alpha mattes, it fails to capture associated effects. Our systematic analysis reveals this arises because effect-related cues are primarily encoded in specific DiT blocks and

become suppressed when LoRA is applied across all blocks. To address this, we introduce EasyOmnimatte, the first unified, end-to-end video omnimatte method. Concretely, we finetune a pretrained video inpainting diffusion model to learn dual complementary experts while keeping its original weights intact: an Effect Expert, where LoRA is applied only to effect-sensitive DiT blocks to capture the coarse structure of the foreground and associated effects, and a fully LoRA-finetuned Quality Expert learns to refine the alpha matte. During sampling, Effect Expert is used for denoising at early, high-noise steps, while Quality Expert takes over at later, low-noise steps. This design eliminates the need for two full diffusion passes, significantly reducing computational cost without compromising output quality. Ablation studies validate the effectiveness of this Dual-Expert strategy. Experiments demonstrate that EasyOmnimatte sets a new

state-of-the-art for video omnimatte and enables various downstream tasks, significantly outperforming baselines in both quality and efficiency.

1. Introduction

Decomposing videos into layered representations is a fundamental problem in computer vision, with significant applications in video editing and visual effects. Effective decomposition enables a variety of video editing tasks, including object removal, background replacement, and other creative workflows. However, extracting accurate alpha mattes for target foregrounds is labor-intensive, especially for complex effects arising from object-environment interactions such as shadows, reflections, generated smoke, or splashes. Apparently, this task is highly difficult to achieve using conventional methods. To address this, the pioneering work [27] has formulated the *Omnimatte* task, which seeks to separate a foreground object and all its associated visual effects into a single coherent layer, and trains a UNet model on the input video in a self-supervised manner. Subsequently, there has been a surge of follow-up omnimatte methods improving it from various perspectives. However, these follow-ups typically rely on slow, multi-stage, or inference-time optimization pipelines that fail to fully exploit powerful generative priors, producing suboptimal decompositions.

In this work, we present the *first unified, end-to-end* framework for video omnimatte generation, addressing the aforementioned limitations. Our key insight is that, if a video inpainting model can be finetuned to remove the foreground-associated visual effects presented with the original, unmasked video [15, 43, 50, 51], which is a task complementary to video omnimatte, then *the model must inherently possess the ability to perceive these effects*. Consequently, the same model can be finetuned to include such effects in the foreground layer prediction. To this end, we explore employing lightweight Low-Rank Adaptation (LoRA) [10] modules to finetune the inpainting model on a synthetic matting dataset, enabling it to directly predict the foreground layer in a single, efficient stage. Nonetheless, our initial experiments with this straightforward setup revealed a critical and unexpected challenge: while the finetuned model effectively produced high-quality alpha mattes for the foreground objects, it failed to consistently perceive and capture associated effects such as shadows and reflections — even when trained with ground-truth targets that include them.

Intrigued by this phenomenon, we conduct a systematic analysis of the model’s internal mechanisms. Inspired by ObjectDrop [43] and the analysis in Gen-Omnimatte [15], we examine how different blocks of the inpainting model respond to subtle “effect” cues. Our structural analysis of the inpainting model reveals that the perception of these effects is concentrated within a specific subset of blocks

in the DiT architecture. Applying LoRA to all DiT blocks tends to overemphasize the main alpha prediction task and inadvertently suppresses the weaker, effect-related signals, manifested as missing effects in the omnimatting results. With these findings, our key idea is to finetune a pretrained video inpainting diffusion model to learn two complementary experts while keeping its original weights frozen: an *Effect Expert*, where LoRA is applied only to effect-sensitive intermediate stages to capture the *coarse*, coarse structure of the foreground and its associated effects, and a *Quality Expert*, where LoRA is applied to all DiT blocks, learns to add *fine* details to the alpha matte. Notably, during diffusion sampling, the Effect Expert model is only employed at early, high-noise stages to generate coarse, effect-aware omnimatte predictions, while the Quality Expert model refines the alpha matte only at later, low-noise stages. Rather than running the two finetuned models sequentially for full diffusion sampling, this alternating strategy achieves high-quality results with roughly half the computational cost. Experiments confirm that this classic *coarse-to-fine* strategy, which leverages both the specialized and full LoRA adaptations, is crucial for producing alpha mattes that are accurate and faithfully preserve associated effects. Our evaluation demonstrates that EasyOmnimatte sets a new state-of-the-art (SOTA) video omnimatte. Importantly, the proposed method maintains an end-to-end, feed-forward decomposition of the foreground layers, ensuring both effectiveness and efficiency.

In summary, our main contributions are as follows:

- We introduce the *first end-to-end video omnimatte* method, which directly adapts a pre-trained video inpainting model to capture associated visual effects, eliminating the need for slow, multi-stage optimization-based pipelines.
- We perform a structural analysis revealing that the standard inpainting pretraining objective inherently conflicts with the preservation of foreground effects, strongly motivating our key designs.
- We finetune a pretrained video inpainting diffusion model to learn two specialized experts. These complementary experts operate in tandem across the high- and low-noise denoising stages, producing high-fidelity alpha mattes that accurately capture the associated effects, all without additional computational overhead.

2. Related Work

Foreground Isolation and Matting. The most straightforward way to extract the foreground layer from a video is via matting [30, 38]. Converting the foreground into a transparent (RGBA) format enables a wide range of video editing tasks, making video matting a fundamental technique for such applications. In recent years, deep learning-based matting methods for both images [11, 42, 44, 47, 49] and videos [12, 21, 35, 45] have achieved significant improvements, enhancing the accuracy of alpha matte prediction

while reducing the need for extensive guidance information. These methods, often trained on fine-grained labeled datasets, can automatically learn image features to predict high-quality alpha mattes. However, limitations in the diversity of training data [20, 31, 35, 44] and specific training strategies [12, 13, 21] often limit their performance to certain foreground categories, most notably humans. Furthermore, they tend to fail to capture foreground-associated effects, such as shadows or reflections. While some methods [20, 33] leverage a known background to partially address these limitations, they impose strict requirements, such as a static scene or a pre-captured clean background plate.

Video Matting with Associated Effects. A series of methods, exemplified by Omnimatte [26, 27], enables the decomposition of a video into foreground and background layers, with the foreground layer encompassing associated visual effects. These approaches aim to completely separate the foreground layer from the video, leveraging motion cues through flow-based techniques [1, 3, 39] to achieve this effect. Subsequent works have further enhanced the capabilities of Omnimatte by incorporating deep image priors [7, 28], extending planar homography constraints through non-rigid warping [26, 48] or 3D scene representations [19, 34]. However, a significant drawback is the reliance on restrictive motion assumptions, which can result in severely degraded performance when such assumptions no longer hold.

Video Inpainting for Object Removal. Both background matting and Omnimatte-like optimization methods rely on precise background guidance. On the other end, the rapid progress of visual generative models [2, 5, 6, 8, 9, 14, 24, 36, 46] has greatly advanced video inpainting techniques, enabling accurate background layer prediction. Early video inpainting methods focused on natural content completion by exploiting spatio-temporal cues from adjacent frames [4, 23, 37, 54] or propagating optical flow [17, 52] to reduce hallucinations and artifacts within masked regions. More recent approaches [16, 55] directly adapt and fine-tune large video generative models for efficient object removal. Nonetheless, these methods cannot eliminate the associated effects of an object, often leaving residual artifacts inside the inpainted background. In image object removal, several recent methods [43, 50, 51] have specifically addressed the challenge of removing both objects and their associated effects. The key idea behind these approaches is to condition the generative model on the original, unmasked image, enabling it to learn the correlations between objects and their corresponding effects. Built upon this success, Generative Omnimatte [15] extended it to video generation models, successfully training a video removal model capable of cleanly removing objects along with their associated visual effects.

While recent methods can estimate the background by removing objects, using this pseudo-background for foreground separation still remains challenging. On one hand,

per-frame techniques, such as background matting, often exhibit temporal inconsistencies like flickering and struggle to capture foreground-associated effects. On the other hand, two-stage approaches like Generative Omnimatte are not only computationally expensive due to test-time optimization but also prone to error propagation, as inaccuracies in background generation can corrupt the final foreground decomposition. In this work, we propose the first unified framework that simultaneously predicts the background layer and separates the foreground along with its associated effects, all within a single end-to-end model.

3. Method

We tackle the video omnimatte task, which seeks to decompose an input video $\mathbf{V} \in \mathbb{R}^{N \times H \times W \times 3}$ into a foreground layer $\mathbf{F} \in \mathbb{R}^{N \times H \times W \times 3}$, its corresponding alpha matte $\alpha \in [0, 1]^{N \times H \times W}$, and a restored background layer $\mathbf{B} \in \mathbb{R}^{N \times H \times W \times 3}$. These layers must adhere to the standard alpha compositing equation: $\mathbf{V} = \alpha \odot \mathbf{F} + (1 - \alpha) \odot \mathbf{B}$, where \odot denotes element-wise multiplication. In contrast to multi-stage pipelines [15, 27] that predict \mathbf{B} and $\{\alpha, \mathbf{F}\}$ sequentially, we formulate this as a direct, end-to-end prediction problem with a single model \mathcal{G} . This model, with trainable parameters Θ , is trained to map the input video directly and simultaneously to the complete set of decomposition layers. This unified transformation is expressed as:

$$(\mathbf{F}, \alpha, \mathbf{B}) = \mathcal{G}(\mathbf{V}, \mathbf{M}, c; \Theta), \quad (1)$$

where \mathbf{M} indicates the per-frame coarse foreground masks (*i.e.*, without the effects), and c the inpainting condition.

As shown in Fig. 2, to mitigate the data sparsity issue, our core idea is to repurpose the powerful generative prior of a pre-trained video inpainting model in Sec. 3.1. However, we observe that a naive fine-tuning of such a model struggles to capture subtle but crucial foreground effects (*e.g.*, shadows, reflections). To understand this limitation, we first conduct a rigorous structural analysis (Sec. 3.2) of the inpainting model, revealing an innate functional conflict in its later blocks that actively suppresses effect-related signals. Guided by this key observation, we propose a novel Dual Experts Sampling (Sec. 3.3) that resolves this conflict.

3.1. Repurpose Inpainting Diffusion for Omnimatte

Video inpainting models inherently encode a strong separation prior, allowing them to *remove* foregrounds and their associated effects [15]. We leverage this prior differently: by adapting the model to directly predict the foreground’s alpha matte, we achieve a unified, end-to-end Omnimatte pipeline. As Figure 2 shows, to achieve omnimatte prediction in a feed-forward way without compromising the model’s powerful priors, we introduce *Branch DiT Blocks*, constructed by adding trainable LoRA modules [10] to the original DiT blocks, to recover the alpha mate with additional tokens.

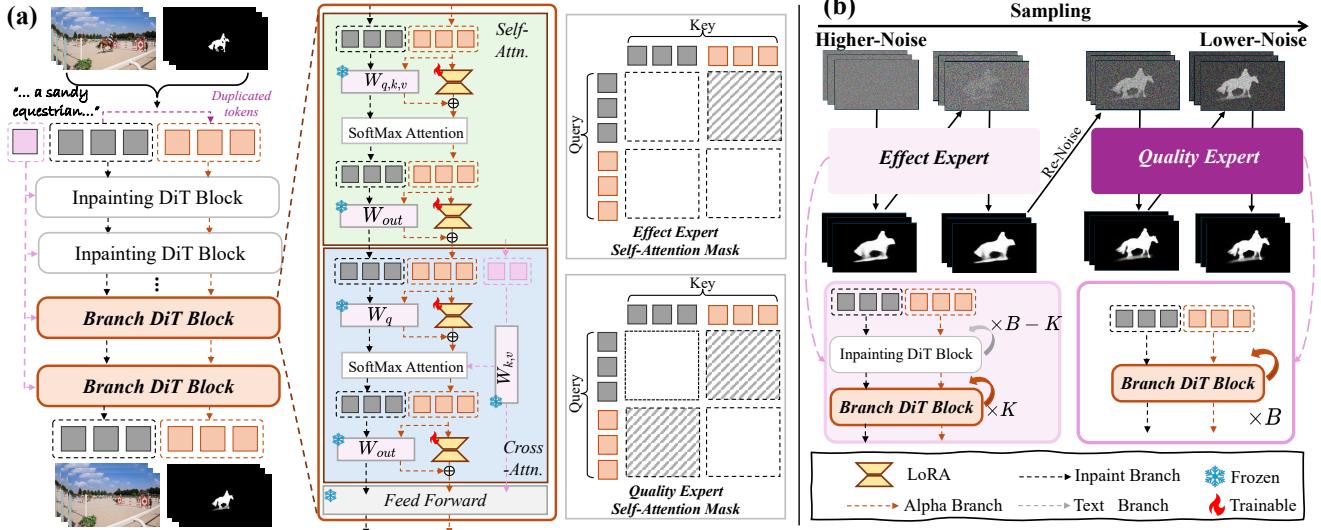


Figure 2. **EasyOmnimatte.** a) We branch out LoRA-finetuned blocks from the original inpainting DiT blocks to jointly predict the alpha matte, alongside the pretrained model. In each Branch DiT Block, LoRA are applied only to the duplicated set of input tokens, leaving the original inpainting branch unaffected. b) During sampling, the *Effect Expert* model is only employed at early, high-noise stages to generate coarse, effect-aware omnimatte predictions, while the *Quality Expert* model refines the alpha matte only at later, low-noise stages. This alternating strategy achieves high-quality results with greatly reduced compute cost, compared to individually sampling.

The baseline inpainting model is conditioned on the complete video frames \mathbf{F} , per-frame foreground masks \mathbf{M} provided by modern segmentors [32], and a brief environmental text description c for background guidance. \mathbf{F} and \mathbf{M} are incorporated into the noise latents of the inpainting DiT through channel concatenation and projection to obtain the input visual token. Our method directly inherits this complete input scheme with additional tokens. Specifically, we first duplicate the input visual tokens as well as the rotary positional embeddings (RoPE) following [40], then token-wisely concatenate them, creating two parallel sets: the original tokens destined for the inpainting task and a copied set for alpha matte prediction. These lightweight LoRA are applied *exclusively* to the copied set of tokens. Consequently, the original, frozen backbone of the inpainting model operates on the untainted original tokens, proceeding with its background prediction task, while the LoRA-equipped branch processes the copied tokens, effectively redirecting their output from background prediction toward alpha matte estimation.

We choose to predict the alpha matte $\hat{\alpha}$ instead of the foreground layer $\hat{\mathbf{F}}$ [53] since the alpha matte is well-defined across the entire frame (*i.e.*, it has a value of 0 in pure background regions), providing a more stable and well-posed prediction target. In contrast, the foreground pixels are only defined where the alpha is greater than zero.

Since we froze the original video inpainting model, the original input of the video inpainting model predicts the background $\hat{\mathbf{B}}$, whereas the alpha matte $\hat{\alpha}$ is generated by the additional tokens, as shown in Fig. 2. We then analytically recover the foreground layer $\hat{\mathbf{F}}$ by rearranging the standard compositing equation. For each frame f , the foreground \mathbf{F}_f

is computed as:

$$\hat{\mathbf{F}}_f = \frac{\text{clip}(\mathbf{I}_f - (1 - \hat{\alpha}_f) \cdot \hat{\mathbf{B}}_f)}{\hat{\alpha}_f + \epsilon}, \quad (2)$$

where \mathbf{I}_f is the original frame, $\hat{\mathbf{B}}_f$ is the inpainted background, and ϵ is a small constant to prevent division by zero. The clip function ensures that the resulting pixel values remain within the valid $[0, 1]$ range.

3.2. Block-Wise Analysis of Effects Association

Following the above structure, we observe that even when employing all *Branch DiT* blocks, video inpainting models still fail to effectively capture and preserve the associated effects. Given the efficacy of inpainting models in erasing an object as well as its attendant effects that are not specified, we infer an underlying sequential process of first perceiving and then eliminating these phenomena. To validate this hypothesis, we perform a block-wise analysis to probe the mechanisms for effect-perception and processing at different depths of the network.

We calculate the *Effects Association Score* [15, 43] $s(p)$ for each foreground pixel p , based on a block's self-attention map \mathbf{W} . This score represents the fraction of a pixel's attention that is directed towards the effects region \mathbf{M}^e :

$$s(p) = \frac{\sum_{y \in \mathbf{M}^e} \mathbf{W}_{p,y}}{\sum_{x \in I} \mathbf{W}_{p,x}}, \quad (3)$$

so for frame f , we can obtain the effect-related attentive map of the b -th DiT blocks, denoted as $\mathbf{S}_{f,b}$.

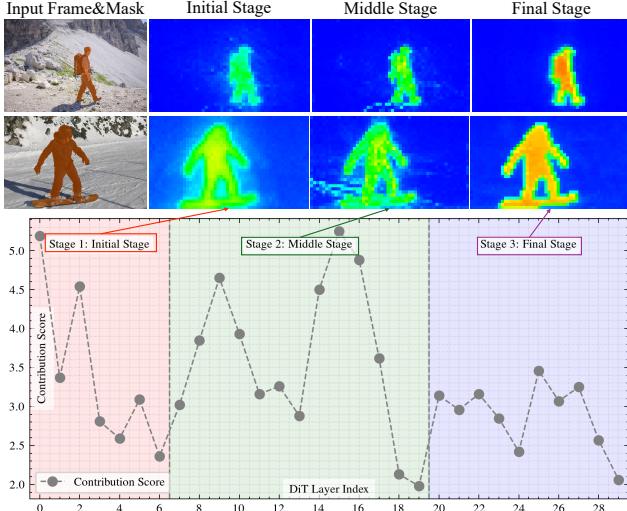


Figure 3. **Our block-wise Analysis.** **Bottom:** Contribution scores per layer reveal the three inpainting stages. **Top:** Averaged attention maps (S_b) for each stage show that the middle stage is most sensitive to effects (e.g., shadows), which are then suppressed in the final stage, corroborating our perception-then-elimination hypothesis.

To quantify how different attention blocks in a DiT perceive associated effects, we compute a normalized, block-wise contribution score. Specifically, for each frame f , we extract the ground-truth effect mask M_f^e by taking the intersection of a binarized alpha matte and the complement of a dilated foreground mask, isolating the effect region from the object itself. Supplementary presents the full derivation.

The block-wise contribution score C_b for a video is the sum of activations within the effect mask over all N frames:

$$C_b = \frac{\sum_{f=1}^N \sum_p (S_{f,b} \odot M_f^e)}{\sum_{b=1}^B \sum_{f=1}^N \sum_p (S_{f,b} \odot M_f^e)}. \quad (4)$$

By averaging these scores over the full dataset, we obtain an architectural sensitivity profile that highlights which blocks are most responsive to associated effects (Fig. 3, bottom).

Based on the block-wise contribution scores, which we plot in Fig. 3, we partition the inpainting model into three distinct stages using the curve's troughs as clear delineators. The averaged S_b maps for each stage, also visualized in Fig. 3 on real-world videos, elucidate their functional roles. The initial stage, characterized by broad receptive fields, primarily encodes contextual scene information. Blocks in the middle stage most strongly capture the spatial structures of foreground-associated effects. Conversely, in the final stage, effect-related features are actively suppressed. This observation validates our initial hypothesis on the inpainting mechanism.

3.3. Dual Experts Sampling

Based on the above findings, we involve a novel dual expert sampling strategy by *Effect Expert* and *Quality Expert*

sequentially to obtain the final omnimatte as shown in Fig. 2 (b). Below, we provide the details for each part.

Effect Expert is a model specialized in fine-tuning the original video inpainting model to capture the effects. Instead of applying LoRA to all DiT blocks, *Effect Expert* has only LoRA fine-tuning blocks exclusively on the final stages of the inpainting DiT as shown in Fig. 2. The motivation lies in the finding from Fig. 3, which is to prevent the LoRA modules trained for alpha prediction from interfering with the inpainting model's inherent ability to perceive and remove associated effects. Formally, denote the DiT blocks in the inpaintint model as $\Theta = \{L_1, L_2, \dots, L_B\}$, our Effect Expert model \mathcal{G}^E now has trainable weights:

$$\Theta_E := \{\mathcal{B}(L_b) \mid b \in (B - K, B]\}, \quad (5)$$

where \mathcal{B} is the branch-out operation adding LoRA to the last K DiT block. To train the *Effect Expert*, we modify the self-attention mask so that the inpainting query tokens do not attend to the effect-matting tokens. This ensures that the inpainting branch remains unaffected, while its intermediate representations provide stable guidance for the effect-capturing branch (see Fig. 2, top-middle).

Quality Expert. As shown in Fig. 7 (d), while the Effect Expert model significantly improves the preservation of effects, it comes at the cost of reduced precision in the overall matte shape compared to the single, full *Quality Expert* with trainable weights Θ_O :

$$\Theta_O := \{\mathcal{B}(L_b) \mid b \in [1, B]\}, \quad (6)$$

To train *Quality Expert*, we mask out the attention between the inpainting tokens and the matting tokens (see Fig. 2, bottom-middle). This design not only accelerates attention computation but also yields improved matting quality.

Sampling. We further introduce a *Dual Expert Sampling Strategy* to achieve both high-fidelity effect prediction and accurate matte generation, and high computational efficiency. Our approach is inspired by the progressive nature of diffusion and flow-matching models, which generate content by gradually reducing noise over a series of timesteps, from high noise to low noise, to capture the target effects at early diffusion stages when the major content forms and refine details at later stages. As illustrated in Fig. 2 (b), we control the applied model of the diffusion timestep by the threshold τ . Let \mathcal{G}^E and \mathcal{G}^Q denote the trained *Effect Expert* model and *Quality Expert* model, respectively, the sampling progress at a timestep t is determined as follows:

$$\mathcal{G} = \begin{cases} \mathcal{G}_E & \text{if } t > \tau, \\ \mathcal{G}_Q & \text{if } t \leq \tau, \end{cases} \quad (7)$$

where τ is set to 0.5 in our experiments suggested by Fig. 8. These complementary experts operate in tandem across the

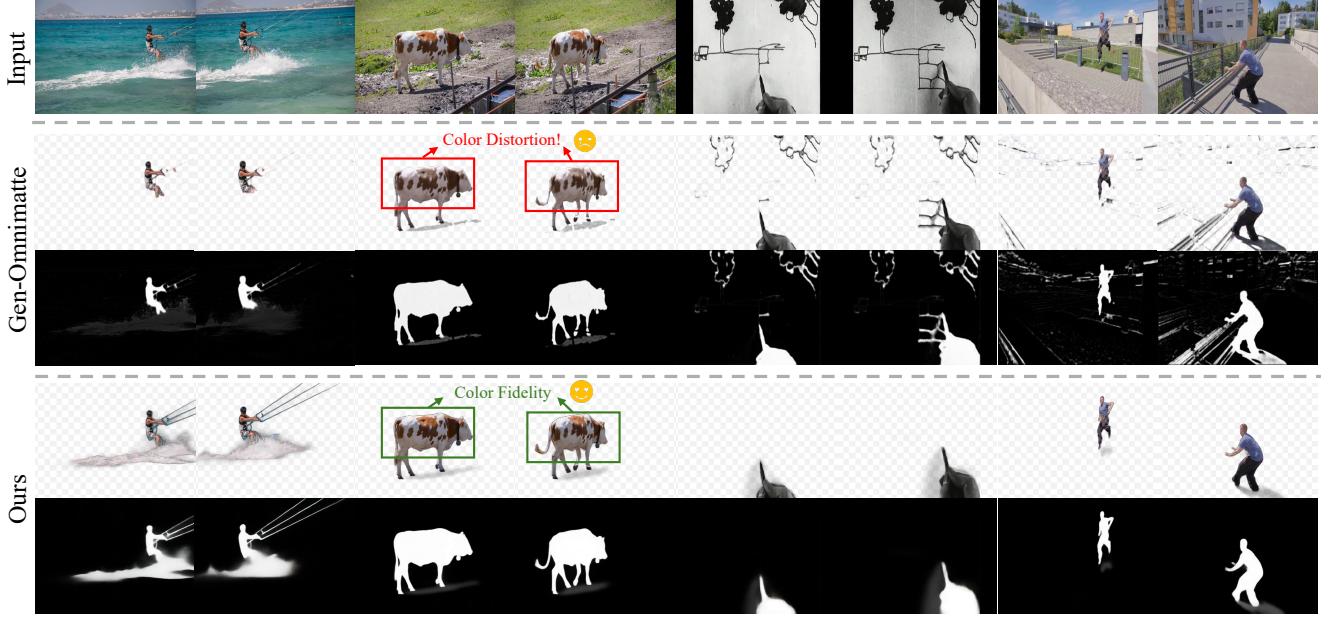


Figure 4. Qualitative comparison with optimization-based omnimatte method [15]. Please zoom in for a better view.

high- and low-noise denoising stages, producing high-fidelity alpha mattes that accurately capture the associated effects, all without additional computational overhead.

4. Experiments

Our training data is synthetically generated by compositing foreground videos with high-quality alpha matte annotation [20] over background videos. The background videos are sourced from large-scale, high-resolution video datasets [18], each accompanied by a descriptive caption. During composition, we apply a set of dynamic augmentations, including random resizing, rotation, and translation of the foreground, to enhance the diversity of the training samples. The foreground mask M required by the inpainting model is derived directly from the ground-truth alpha matte α_{gt} . To simulate foreground effects, we generate pseudo-shadows by applying a series of transformations (e.g., shearing, blurring, and color adjustments) to the foreground matte.

For our experiments, we initialize our approach with a video inpainting model provided by Lee et al. [15], which is built upon the open-source video generation framework from Wan et al. [36]. We then fine-tune this model to adapt it for the Omnimatte task. Our fine-tuning strategy incorporates a dual-expert architecture using LoRA, where the ranks for the Effect Expert and the Quality Expert are set to 128 and 64, respectively. For the dual-expert sampling strategy, we employ a default threshold of $\tau = 0.5$. The entire model is trained end-to-end for 8,000 iterations using the AdamW optimizer [25] with a learning rate of 1×10^{-3} . The training is conducted on two H100 GPUs.

4.1. Comparison with State-of-the-Art Methods

Optimization-based Omnimatte Prediction. After generating the background with the same removal model we use, we employ the Omnimatte Optimization method from Gen-Omnimatte [15] to decompose the foreground. Following the official recommendations, we set the number of optimization steps to 4000 and enabled detail transfer [26]. Note that this is a test-time optimization method similar to the previous approaches [19, 27], requiring several minutes of optimization for each foreground layer, which is over an order of magnitude slower than our approach.

Learning-based Matting Prediction. We establish two categories of learning-based matting baselines. First, for background-matting methods, we use the output of our removal model to guide BGMv2 [20]. Second, for mask-guided methods, we generate an initial mask for the foreground and its effects by integrating the segmentation model [32] with SSIS-v2 [41]. This mask is then used to prompt the video matting model [45] for prediction. While contemporary learning-based methods are considerably more efficient than Gen-Omnimatte, they typically fail to preserve the object’s associated effects, and even with auxiliary post-processing, the results are often suboptimal.

Qualitative Comparison. Fig. 4 presents our comparison against the optimization-based method. Gen-Omnimatte [15] suffers from color distortion, leading to poor foreground decomposition. Furthermore, it fails in scenarios with rapid background changes, resulting in significant background bleeding into the foreground layer. In contrast, our method overcomes these shortcomings while drastically reducing the inference time from several minutes to under 10 seconds. Fig. 5 shows the comparison against dedicated mat-

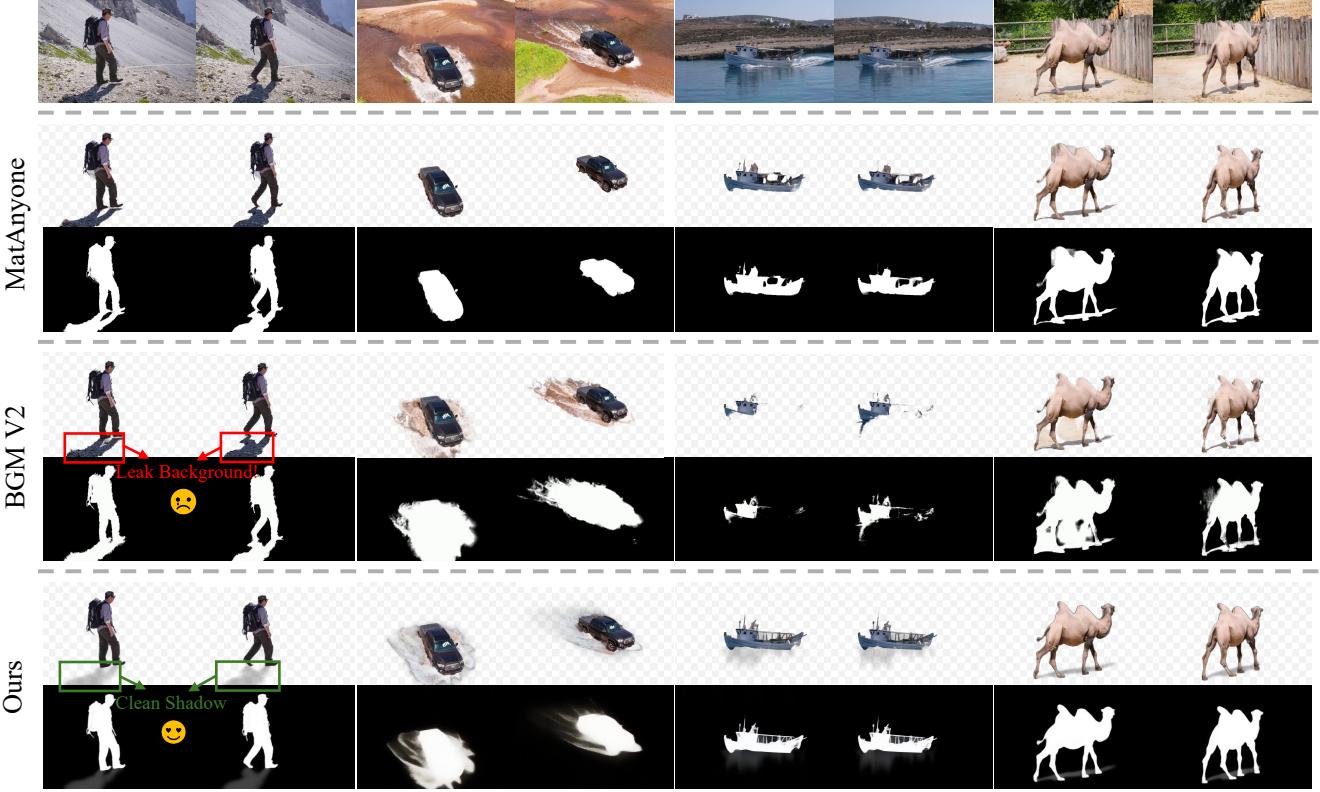


Figure 5. Qualitative comparisons with learning-based matting method [20, 45]. Please zoom in for a better view.

ting methods. When capturing associated foreground effects, both matting baselines tend to predict excessively high alpha values in the effect regions. This results in the extracted foreground being contaminated with colors from the original background. Moreover, baselines exhibit poor generalization to non-human categories due to the missing of the generative priors, leading to significant artifacts and low-quality mattes.

Quantitative Comparison and Human Evaluation. To objectively evaluate the performance of EasyOmnimatte in the absence of labeled test data, we design two quantitative experiments and incorporate a human evaluation study. The quantitative experiments include calculating a series of perceptual losses of the recomposed video against the original, as well as batch compositing onto numerous backgrounds and calculating the effect on the background video distribution. These experiments are based on the viewpoint that high-quality layered results can prevent the loss of original video information and allow for more harmonious synthesis into new backgrounds. More details can be found in the supplementary. We also conduct the human evaluation to assess the layering effect based on three criteria: foreground integrity, effect harmony, and temporal consistency. In our experiments, 28 users participated with 20 videos to study, resulting in 1,680 opinions.

Table 1. **Quantitative comparison with baselines.** Notice that the BGM V2 and MatAnyone are not specifically for associated effects. † indicates cooperation with the shadow detection method [41].

Method	PSNR (\uparrow)	SSIM (\uparrow)	WE (\downarrow)	FVD (\downarrow)
BGM V2 [20]	26.61	78.78	101.04	168.31
MatAnyone† [45]	26.12	78.68	100.46	146.44
Gen-Omnimatte [15]	24.35	69.36	101.33	<u>116.32</u>
EasyOmnimatte(Ours)	<u>26.23</u>	78.83	100.94	105.48

Table 2. **User Studies.** Human evaluation of different video generation methods across multiple aspects. Scores range from 0 to 5, with higher scores indicating better performance. Bold values represent the best performance within each metric. † indicates cooperation with the shadow detection method [41].

Method	Overall Score \uparrow	Foreground Integrity \uparrow	Effect Harmony \uparrow	Temporal Consistency \uparrow
BGM-V2	2.26	1.78	2.96	2.04
MatAnyone†	2.82	2.83	2.60	3.02
Gen-Omnimatte	2.85	2.45	3.36	2.74
EasyOmnimatte(Ours)	4.08	4.07	3.97	4.21

4.2. Ablation Study

Effectiveness of different attention masking strategies. Figure 6 shows the results under different self-attention mask configurations. As shown in (a), allowing unrestricted communication between the branch tokens during inference leads to a corrupted background prediction, which in turn degrades

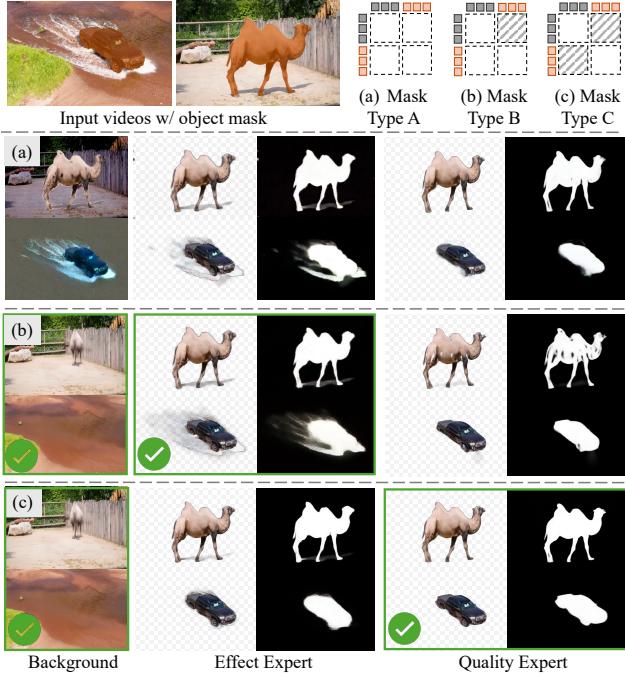


Figure 6. **Ablation of Attention Masking.** Training without the correct self-attention mask configuration renders both the Effect Expert and the Matting Expert ineffective. We use Mask type B and Mask type C for Effect Expert Quality Expert, respectively.

the final foreground decomposition. Conversely, as seen in (b), preserving only the alpha-to-background attention path enhances the effect expert to perceive object-associated effects. However, this masking setup fails to resolve the poor quality of the alpha matte predicted by the Effect Expert. We attribute this issue to the masking strategy itself, rather than to the partial application of LoRA on only a subset of the Effect Expert’s blocks. This is evidenced by the Quality Expert, which, despite being fine-tuned with LoRA across all blocks, also yields unsatisfactory results under the identical mask configuration. Therefore, our final strategy is to train the branches in complete isolation and then use the mask in (c) during inference. With this masking constraint, training the Quality Expert is equivalent to fine-tuning the inpainting model. Without being influenced by the frozen inpainting branch, the Quality Expert can quickly fit the matting data and form high-quality alpha mattes. However, the trade-off is that the Quality Expert remains incapable of perceiving effects, further highlighting the expert sample strategy.

Ablation of Branch DiT Placement. We experiment with modifying the Branch DiT Block to different inpainting stages. Our finding is consistent with the conclusion from our layer-wise analysis: applying LoRA to the later layers, which are responsible for suppressing effects, yields the most significant improvement in effect preservation.

Validation of the Dual Expert Strategy. We compare our full method against two primary variants: (d) using only the Effect Expert for the entire generation process, and (e) using

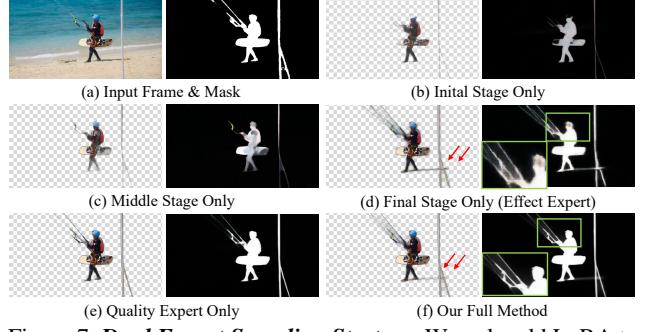


Figure 7. **Dual Expert Sampling Strategy.** We only add LoRA to the final stage to keep the prediction of the effects, and our full dual expert strategy further improves the matting quality.



Figure 8. Effect of varying **threshold** τ . By default, we set $\tau = 0.5$ for a balanced outcome, but users can decrease it for higher matte precision or increase it for more prominent associated effects.

only the Quality Expert. The results show that the Quality Expert achieves high precision on the main foreground but fails on effects, while the Effect Expert is strong on effects but sacrifices boundary precision. Our full model, which combines them via the hybrid diffusion sampling strategy, combines both of the advantages.

Effect of different timestep threshold τ . To further investigate the impact of the τ in our Dual Experts Sampling strategy, we perform a quantitative analysis on a synthetic validation set. This set is generated by augmenting the test set of VideoMatte240K [20] with the pseudo-shadow augmentation method. We use the Mean Squared Error (MSE) metric to separately assess the quality of the predicted foreground subject and the captured effects. The results of this analysis are visualized in Figure 8.

5. Conclusion

We introduce EasyOmnimatte, a novel, end-to-end framework for video omnimatte, that is capable of capture complex foreground-associated effects with ease. Our key idea is to repurpose a powerful video inpainting model that has been proofed to be capable of perceiving the associated effects. Nonetheless, a trivial setup that directly apply LoRA to all DiT blocks fails to capture the effects. We analyze the source of this failure and, consequently, propose the Dual Expert matting strategy. This strategy successfully captures the associated effects while simultaneously producing high-quality

alpha matting. Our work establishes a new and highly effective paradigm for adapting large-scale generative models to complex, detailed video decomposition tasks.

Limitations and Future Work. While our method demonstrates superior video omnimatte performance, it is intrinsically linked to the capabilities of the base inpainting model. For future work, we will explore the application of our block-wise analysis and adaptation framework to more advanced and larger-scale generative models, pushing the boundaries of video omnimatte.

References

[1] Edward H Adelson. Layered representations for vision and video. In *Proceedings IEEE Workshop on Representation of Visual Scenes (In Conjunction with ICCV'95)*, pages 3–9. IEEE, 1995. 3

[2] 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. 3

[3] Gabriel J Brostow and Irfan A Essa. Motion based decomposing of video. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*, volume 1, pages 8–13. IEEE, 1999. 3

[4] Ya-Liang Chang, Zhe Yu Liu, Kuan-Ying Lee, and Winston Hsu. Free-form video inpainting with 3d gated convolution and temporal patchgan. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9066–9075, 2019. 3

[5] 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/CVF Conference on Computer Vision and Pattern Recognition*, pages 7310–7320, 2024. 3

[6] Rohit Girdhar, Mannat Singh, Andrew Brown, Quentin Duval, Samaneh Azadi, Sai Saketh Rambhatla, Akbar Shah, Xi Yin, Devi Parikh, and Ishan Misra. Emu video: Factorizing text-to-video generation by explicit image conditioning. *arXiv preprint arXiv:2311.10709*, 2023. 3

[7] Zeqi Gu, Wenqi Xian, Noah Snavely, and Abe Davis. Factormatte: Redefining video matting for re-composition tasks. *ACM Transactions on Graphics (TOG)*, 42(4):1–14, 2023. 3

[8] Yoav HaCohen, Nisan Chiprut, Benny Brazowski, Daniel Shalem, Dudu Moshe, Eitan Richardson, Eran Levin, Guy Shiran, Nir Zabari, Ori Gordon, et al. Ltx-video: Realtime video latent diffusion. *arXiv preprint arXiv:2501.00103*, 2024. 3

[9] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022. 3

[10] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022. 2, 3, 12

[11] Yihan Hu, Yiheng Lin, Wei Wang, Yao Zhao, Yunchao Wei, and Humphrey Shi. Diffusion for natural image matting. In *European Conference on Computer Vision*, pages 181–199. Springer, 2024. 2

[12] Chuong Huynh, Seoung Wug Oh, Abhinav Shrivastava, and Joon-Young Lee. Maggie: Masked guided gradual human instance matting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3870–3879, 2024. 2, 3

[13] Zhanghan Ke, Jiayu Sun, Kaican Li, Qiong Yan, and Rynson WH Lau. Modnet: Real-time trimap-free portrait matting via objective decomposition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 1140–1147, 2022. 3

[14] Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, et al. Hunyuuanvideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024. 3

[15] Yao-Chih Lee, Erika Lu, Sarah Rumbley, Michal Geyer, Jia-Bin Huang, Tali Dekel, and Forrester Cole. Generative omnimatte: Learning to decompose video into layers. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 12522–12532, 2025. 1, 2, 3, 4, 6, 7, 12

[16] Xiaowen Li, Haolan Xue, Peiran Ren, and Liefeng Bo. Difueraser: A diffusion model for video inpainting. *arXiv preprint arXiv:2501.10018*, 2025. 3

[17] Zhen Li, Cheng-Ze Lu, Jianhua Qin, Chun-Le Guo, and Ming-Ming Cheng. Towards an end-to-end framework for flow-guided video inpainting. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 17562–17571, 2022. 3

[18] Bin Lin, Yunyang Ge, Xinhua Cheng, Zongjian Li, Bin Zhu, Shaodong Wang, Xianyi He, Yang Ye, Shenghai Yuan, Lihuan Chen, et al. Open-sora plan: Open-source large video generation model. *arXiv preprint arXiv:2412.00131*, 2024. 6

[19] Geng Lin, Chen Gao, Jia-Bin Huang, Changil Kim, Yipeng Wang, Matthias Zwicker, and Ayush Saraf. Omnimatterf: Robust omnimatte with 3d background modeling. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 23471–23480, 2023. 3, 6

[20] Shanchuan Lin, Andrey Ryabtsev, Soumyadip Sengupta, Brian L Curless, Steven M Seitz, and Ira Kemelmacher-Shlizerman. Real-time high-resolution background matting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8762–8771, 2021. 3, 6, 7, 8, 12

[21] Shanchuan Lin, Linjie Yang, Imran Saleemi, and Soumyadip Sengupta. Robust high-resolution video matting with temporal guidance. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 238–247, 2022. 2, 3, 12

[22] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022. 12

[23] Rui Liu, Hanming Deng, Yangyi Huang, Xiaoyu Shi, Lewei Lu, Wenxiu Sun, Xiaogang Wang, Jifeng Dai, and Hongsheng Li. Fuseformer: Fusing fine-grained information in transformers for video inpainting. In *Proceedings of the IEEE/CVF*

international conference on computer vision, pages 14040–14049, 2021. 3

[24] Yixin Liu, Kai Zhang, Yuan Li, Zhiling Yan, Chujie Gao, Ruoxi Chen, Zhengqing Yuan, Yue Huang, Hanchi Sun, Jianfeng Gao, et al. Sora: A review on background, technology, limitations, and opportunities of large vision models. *arXiv preprint arXiv:2402.17177*, 2024. 3

[25] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 6

[26] Erika Lu, Forrester Cole, Tali Dekel, Weidi Xie, Andrew Zisserman, David Salesin, William T Freeman, and Michael Rubinstein. Layered neural rendering for retiming people in video. *arXiv preprint arXiv:2009.07833*, 2020. 3, 6

[27] Erika Lu, Forrester Cole, Tali Dekel, Andrew Zisserman, William T Freeman, and Michael Rubinstein. Omnimatte: Associating objects and their effects in video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4507–4515, 2021. 2, 3, 6

[28] Erika Lu, Forrester Cole, Weidi Xie, Tali Dekel, Bill Freeman, Andrew Zisserman, and Michael Rubinstein. Associating objects and their effects in video through coordination games. *Advances in Neural Information Processing Systems*, 35:27049–27060, 2022. 3

[29] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4195–4205, 2023. 12

[30] Thomas Porter and Tom Duff. Compositing digital images. In *Proceedings of the 11th annual conference on Computer graphics and interactive techniques*, pages 253–259, 1984. 2

[31] Yu Qiao, Yuhao Liu, Xin Yang, Dongsheng Zhou, Mingliang Xu, Qiang Zhang, and Xiaopeng Wei. Attention-guided hierarchical structure aggregation for image matting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13676–13685, 2020. 3

[32] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädel, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024. 4, 6

[33] Soumyadip Sengupta, Vivek Jayaram, Brian Curless, Steven M Seitz, and Ira Kemelmacher-Shlizerman. Background matting: The world is your green screen. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2291–2300, 2020. 3

[34] Mohammed Suhail, Erika Lu, Zhengqi Li, Noah Snavely, Leonid Sigal, and Forrester Cole. Omnimatte3d: Associating objects and their effects in unconstrained monocular video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 630–639, 2023. 3

[35] Yanan Sun, Guanzhi Wang, Qiao Gu, Chi-Keung Tang, and Yu-Wing Tai. Deep video matting via spatio-temporal alignment and aggregation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6975–6984, 2021. 2, 3

[36] Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025. 3, 6, 12

[37] Chuan Wang, Haibin Huang, Xiaoguang Han, and Jue Wang. Video inpainting by jointly learning temporal structure and spatial details. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 5232–5239, 2019. 3

[38] Jue Wang, Michael F Cohen, et al. Image and video matting: a survey. *Foundations and Trends® in Computer Graphics and Vision*, 3(2):97–175, 2008. 2

[39] John YA Wang and Edward H Adelson. Representing moving images with layers. *IEEE transactions on image processing*, 3(5):625–638, 1994. 3

[40] Luozhou Wang, Yijun Li, Zhifei Chen, Jui-Hsien Wang, Zhifei Zhang, He Zhang, Zhe Lin, and Ying-Cong Chen. Transpixeler: Advancing text-to-video generation with transparency. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 18229–18239, 2025. 4

[41] Tianyu Wang, Xiaowei Hu, Pheng-Ann Heng, and Chi-Wing Fu. Instance shadow detection with a single-stage detector. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–14, 2022. 6, 7

[42] Zhixiang Wang, Baiang Li, Jian Wang, Yu-Lun Liu, Jinwei Gu, Yung-Yu Chuang, and Shin’ichi Satoh. Matting by generation. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–11, 2024. 2

[43] Daniel Winter, Matan Cohen, Shlomi Fruchter, Yael Pritch, Alex Rav-Acha, and Yedid Hoshen. Objectdrop: Bootstrapping counterfactuals for photorealistic object removal and insertion. In *European Conference on Computer Vision*, pages 112–129. Springer, 2024. 2, 3, 4

[44] Ning Xu, Brian Price, Scott Cohen, and Thomas Huang. Deep image matting. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2970–2979, 2017. 2, 3

[45] Peiqing Yang, Shangchen Zhou, Jixin Zhao, Qingyi Tao, and Chen Change Loy. Matanyone: Stable video matting with consistent memory propagation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 7299–7308, 2025. 2, 6, 7

[46] 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. *arXiv preprint arXiv:2408.06072*, 2024. 3

[47] Jingfeng Yao, Xinggang Wang, Shusheng Yang, and Baoyuan Wang. Vitmatte: Boosting image matting with pre-trained plain vision transformers. *Information Fusion*, 103:102091, 2024. 2

[48] Vickie Ye, Zhengqi Li, Richard Tucker, Angjoo Kanazawa, and Noah Snavely. Deformable sprites for unsupervised video decomposition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2657–2666, 2022. 3

[49] Qihang Yu, Jianming Zhang, He Zhang, Yilin Wang, Zhe Lin, Ning Xu, Yutong Bai, and Alan Yuille. Mask guided matting via progressive refinement network. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1154–1163, 2021. 2

- [50] Xin Yu, Tianyu Wang, Soo Ye Kim, Paul Guerrero, Xi Chen, Qing Liu, Zhe Lin, and Xiaojuan Qi. Objectmover: Generative object movement with video prior. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 17682–17691, 2025. [2](#), [3](#)
- [51] Yongsheng Yu, Ziyun Zeng, Haitian Zheng, and Jiebo Luo. Omnipaint: Mastering object-oriented editing via disentangled insertion-removal inpainting. *arXiv preprint arXiv:2503.08677*, 2025. [2](#), [3](#)
- [52] Kaidong Zhang, Jingjing Fu, and Dong Liu. Flow-guided transformer for video inpainting. In *European conference on computer vision*, pages 74–90. Springer, 2022. [3](#)
- [53] Lvmin Zhang and Maneesh Agrawala. Transparent image layer diffusion using latent transparency. *arXiv preprint arXiv:2402.17113*, 2024. [4](#)
- [54] Shangchen Zhou, Chongyi Li, Kelvin CK Chan, and Chen Change Loy. Propainter: Improving propagation and transformer for video inpainting. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10477–10486, 2023. [3](#)
- [55] Bojia Zi, Weixuan Peng, Xianbiao Qi, Jianan Wang, Shihao Zhao, Rong Xiao, and Kam-Fai Wong. Minimax-remover: Taming bad noise helps video object removal. *arXiv preprint arXiv:2505.24873*, 2025. [3](#)

This document provides supplementary details for our main paper. We begin with the preliminaries of our method (Sec. A), followed by more details on the training data production (Sec. B), specific settings for our Block-wise Analysis (Sec. C), and the full experimental evaluation protocol (Sec. D). Furthermore, we present additional ablation studies (Sec. E) and an analysis of failure cases (Sec. F). Full video comparisons are available on our project page.

A. Preliminaries

WAN2.1 Model Structure. Our work is built upon WAN2.1 [36], a video diffusion model constructed as a stack of customized Diffusion Transformer (DiT) blocks [29]+. Each block is composed of a visual self-attention layer for modeling spatio-temporal relationships within the video, and a cross-attention layer to incorporate textual conditioning, although text prompts are not used in our specific application.

To adapt this architecture for the video inpainting task, we follow a similar input formulation to Gen-Omnimatte [15]. The process is as follows: first, the input video V is encoded into a latent representation \mathbf{z}_v using a pre-trained Variational Autoencoder (VAE). This video latent is then concatenated along the channel dimension with the downsampled frame-wise binary mask M and a noise latent of the same spatial dimensions. The concatenated tensor is then passed through a linear projection layer to compress its channel dimension, forming the final sequence of visual tokens. These tokens are then duplicated and concatenated to serve as the input to the DiT model for the diffusion process.

LoRA and Training Objectives. To efficiently fine-tune the model for the Omnimatte task, we integrate Low-Rank Adaptation (LoRA) [10] into the model’s self-attention layers. For a pre-trained weight matrix $W_0 \in \mathbb{R}^{d \times k}$, the update is represented by a low-rank decomposition $W_0 + \Delta W = W_0 + BA$, where $B \in \mathbb{R}^{d \times r}$ and $A \in \mathbb{R}^{r \times k}$ are trainable matrices with a low rank $r \ll \min(d, k)$. The forward pass is modified as:

$$h = W_0 x + B A x. \quad (8)$$

In our framework, the Branch DiT applies this LoRA computation selectively to the copied tokens that are designated to learn the alpha matte, leaving the original tokens to be processed by the frozen, pre-trained weights.

Our model is trained with direct supervision on the alpha matte prediction. We employ the standard flow matching objective [22], which trains the model to predict the noise ϵ added to a clean latent \mathbf{z}_0 at timestep t . The loss function is defined as:

$$\mathcal{L}_{\text{FM}} = \mathbb{E}_{t, \mathbf{z}_0, \mathbf{z}_1} [\|(\mathbf{z}_1 - \mathbf{z}_0) - \mathbf{v}_\theta(\mathbf{z}_t, t)\|^2]. \quad (9)$$

where \mathbf{z}_0 is the VAE-encoded ground-truth alpha matte, \mathbf{z}_1 is the sampled Gaussian noise, t is the diffusion timestep, and \mathbf{v}_θ is our network (with trainable LoRA parameters) that predicts the velocity from the noisy latent \mathbf{z}_t .

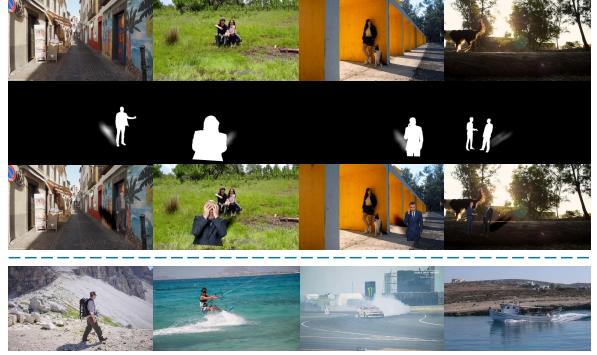


Figure 9. Generalization from simple synthetic data to complex real-world scenes. Our method is trained exclusively on synthetic data (top) featuring only basic shadow effects. Despite this, it successfully decomposes challenging in-the-wild videos (bottom) with a variety of unseen effects, including reflections and smoke. This highlights the model’s ability to bridge a substantial domain gap by learning a true separation principle.

B. Training Data

While the sources and types of our input data are outlined in the main paper, this section provides a detailed description of our data augmentation process and presents visual examples of our training data.

Data Augmentation. We employ a temporally coherent video augmentation pipeline adapted from [20, 21] to improve model robustness and generalization. This involves simulating camera motion by first embedding the sequence within a larger canvas using asymmetric padding (sampled from a uniform distribution $\mathcal{U}(0.3, 0.5)$ of the target width), followed by a smooth affine transformation. Rather than per-frame randomization, we interpolate between two affine states (A and B) using an easing function. To create pronounced lateral movement, the horizontal translations of A and B are set to be of opposite sign with magnitudes ranging from 15-30% of the image width, while rotation ($\pm 5^\circ$), scale (0.95-1.05), and shear ($\pm 3^\circ$) are varied subtly.

A crucial component of our data synthesis is the addition of realistic associated effects. We focus on simulating shadows, as they are one of the most common and challenging effects to separate. For a given foreground alpha matte α , we generate a shadow matte α_s by applying strong vertical compression (to 10-30% of original height), significant horizontal shear ($30^\circ - 60^\circ$), semi-transparent rendering (30-70% opacity) and a final blur. This shadow matte is then used to darken the corresponding region on the background video before the final foreground is composited on top. This process forces the model to learn to identify and separate regions that are visually part of the foreground layer but are not captured by the original object mask.

Demonstration and Discussion. In Fig. 9, we provide a visual comparison between our synthetic training data and

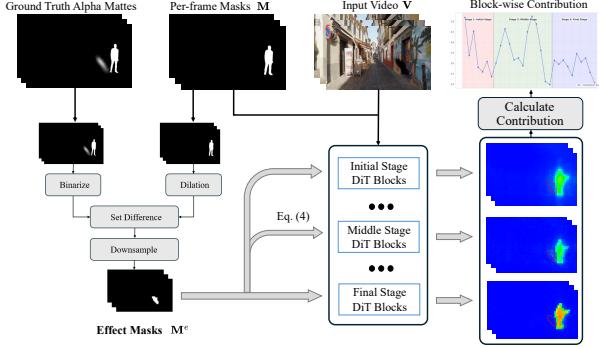


Figure 10. Pipeline of our Block-wise Contribution Analysis Method. The ground-truth alpha matte enables us to precisely localize the regions containing effects and subsequently compute the contribution score.

real-world videos from our test set. The top row showcases examples of our generated training frames, including the final composited video and the corresponding ground-truth alpha matte, which includes both the object and its simulated shadow. The bottom row presents frames from real videos, where the effects (e.g., natural shadows, reflections) are significantly more complex and subtle.

As is visually evident, a substantial *domain gap* exists between our synthetic training data and the real-world test scenarios. Our training set, despite the augmentations, features clean-cut objects and simplified, programmatic shadows. In contrast, real videos contain complex lighting, soft and intricate effects, and various image artifacts. Despite this gap, our model demonstrates strong performance on these real-world examples, successfully isolating both the foreground object and its nuanced, naturally occurring effects. This robust generalization capability suggests that our end-to-end training approach has enabled the model to learn the fundamental, underlying logic of foreground-effect separation, rather than merely memorizing the specific characteristics of our synthetic dataset.

C. Analysis Details

Our block-wise analysis, presented in the main paper, was conducted on a test set of 1000 synthetic videos generated using the same pipeline as our training data but with held-out foreground and background clips. To obtain the effect mask M^e used in the analysis, we isolate the rendered shadow region from our synthetic data generation process. This provides a ground-truth spatial map of where the associated effect is located, allowing us to quantitatively measure each block's sensitivity to effect-related features. The whole pipeline is illustrated in Fig. 10.

D. Evaluation Protocols

Video Reconstruction Quality. This protocol assesses the fidelity of the decomposition. We take a set of P videos to carry out this experiment. The predicted foreground layer \hat{F} is composited back onto the predicted background layer using the predicted alpha matte $\hat{\alpha}$. The quality is measured by comparing this reconstructed video with the original input video V . A high-quality decomposition should allow for a near-perfect reconstruction.

We use three standard metrics: Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index Measure (SSIM), and Warping Loss to calculate temporal misalignments.

Background Composition Plausibility. This protocol evaluates how well the separated foreground layer can be composited onto novel, unseen backgrounds. We take a set of P predicted foreground layers and compose them with a set of Q different background videos. The key is that the separated foreground layer must be clean and free of artifacts from its original background to ensure a seamless new composition.

We use the Fréchet Video Distance (FVD) to quantitatively measure the quality of the newly generated videos. We compute the FVD between the set of our composed videos and the set of original background videos. A lower FVD score indicates that the distribution of the composed videos is closer to that of real videos, suggesting a higher-quality and more plausible decomposition.

We present cases in Fig. 11 to demonstrate the effectiveness of our method in the above experiments, where P and Q are 40 and 200, respectively. Our method demonstrates superiority in preserving the completeness and fine-grained details of the foreground. More critically, it possesses the capability to capture clean and intact associated effects, free from background artifacts.

Human Evaluation. To complement our quantitative metrics with a qualitative assessment of perceptual quality, we conducted a comprehensive user study. We recruited [Number of Participants, e.g., 30] participants with backgrounds in computer graphics and vision. The study was designed to compare our method against 4 leading baselines on a set of 20 challenging video sequences featuring a variety of objects and associated effects.

For each video sequence, participants were presented with a side-by-side comparison of the results from all methods, displayed in a randomized order to prevent bias. Participants were first briefed on the video decomposition task, and then they were shown how a clean separation is crucial for plausible compositing onto novel backgrounds.

Participants were asked to rate the quality of each method's output based on three predefined criteria, which were carefully explained to them beforehand:

1. **Foreground Integrity:** This criterion assesses the completeness and color fidelity of the main object, emphasizing

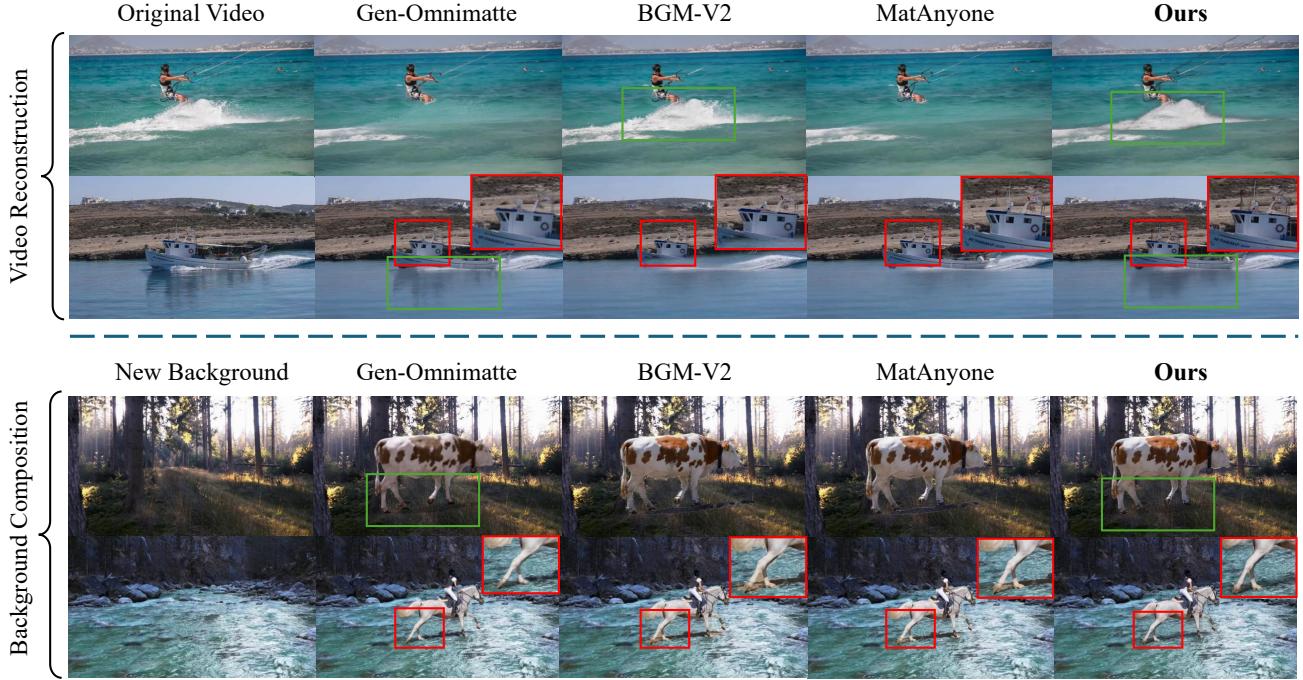


Figure 11. **Visualization of the Video Reconstruction and Background Composition.** The excellent performance of our method in both experiments stems directly from its superior ability to separate the foreground from its effects with high fidelity. We highlight correctly preserved effects in **green** boxes and magnify challenging details in **red** to demonstrate this capability. Zoom in for a better view.

ing the absence of background color bleeding.

2. **Effect Harmony:** This focuses on the integrity of the secondary effects, evaluating both their completeness and the plausibility of their rendered transparency.
3. **Temporal Consistency:** This evaluates the temporal consistency and aesthetic quality of the final decomposition, penalizing artifacts such as temporal flickering, jagged boundaries, or other visual instabilities.

For each criterion, participants provided a score on a 6-point scale, ranging from *0 (very poor)* to *5 (excellent)*. The final scores for each method were then averaged across all participants and video sequences to compute a **Overall Score** for each method aspect.

E. More Ablation Studies

In this section, we investigate the importance of the inpainting training of our base model. Specifically, we conduct an ablation study by fine-tuning the general-purpose conditional video generation model, WAN2.1 Fun, instead of our specialized video inpainting model. The primary goal is to determine whether the inherent knowledge of foreground removal is a prerequisite for successfully training a foreground decomposition model with our proposed strategy. The WAN2.1 Fun model shares the same input scheme with



Figure 12. **Importance of Inpainting Pre-training.** Fine-tuning a general video model (WAN Fun) instead of an inpainting model leads to performance degradation. The Quality Expert (top) exhibits severe color bleeding artifacts in the alpha matte. The Effect Expert (bottom) fails entirely to capture shadows. This ablation confirms that an inpainting foundation is critical, especially for capturing associated effects.

our inpainting base model but is trained for general conditional generation, not explicitly for object removal.

As shown in Fig. 12, we present the results of this ablation by training the WAN2.1 Fun model with our two expert modules:

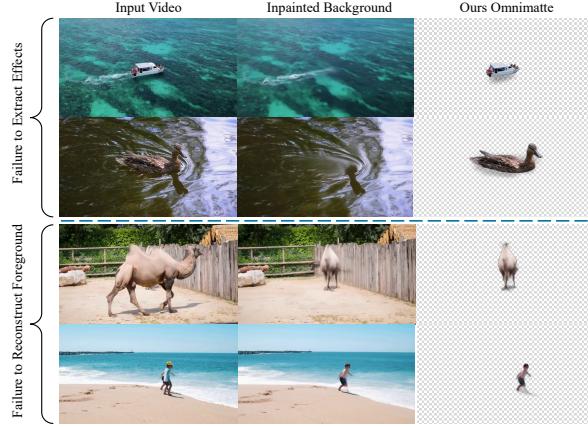


Figure 13. **Failure Cases.** Since our method is built upon an inpainting model, it naturally fails when the underlying inpainting model itself fails.

- **Training as a Quality Expert:** When fine-tuned to predict the alpha matte, we observe that the model can successfully learn to transfer to this new domain and generate a coarse alpha matte. However, the output suffers from significant artifacts, most notably a tendency to leak colors from the original video directly into the alpha matte, corrupting its purity. We hypothesize that this issue could potentially be alleviated with a much larger training dataset and extended training steps, but it highlights a fundamental difficulty for a general model to learn this task.
- **Training as an Effect Expert:** When trained to capture associated effects, the model completely fails to acquire this capability. The fine-tuned model does not learn to identify or isolate effects like shadows or reflections, indicating that this skill does not emerge naturally from a general video generation prior.

These experiments collectively demonstrate that under our current training strategy, fine-tuning a general generative model can roughly predict a primary alpha matte but is incapable of cleanly separating its associated effects. This deficit serves as strong evidence that the pre-training objective of *object removal* is decisively important.

F. Failure Cases

Our method’s performance is inherently dependent on the capabilities of the underlying inpainting model. As illustrated in Fig. 13, we observe two typical failure modes induced by this limitation:

- **Failure to Extract Effects:** If the inpainting model fails to perceive an associated effect, it will treat that region as part of the true background. Consequently, our method cannot separate this effect.
- **Failure to Reconstruct Foreground:** In cases where the inpainting model struggles to inpaint the object that is

heavily occluded, its internal features for that region may become corrupted. This can lead to our method producing only a distorted foreground layer, as the decomposition is derived from these flawed features.

We believe that these limitations will be mitigated as the underlying video inpainting models become more powerful in their removal and completion capabilities.