VDOT: Efficient Unified Video Creation via Optimal Transport Distillation

Yutong Wang[1] Haiyu Zhang[3,2] Tianfan Xue[4,2] Yu Qiao[2]
Yaohui Wang[2] Chang Xu[1*] Xinyuan Chen[2*]
[1]USYD [2]Shanghai AI Laboratory [3]BUAA [4]CUHK
Project Page: https://vdot-page.github.io/

Abstract

The rapid development of generative models has significantly advanced image and video applications. Among these, video creation, aimed at generating videos under various conditions, has gained substantial attention. However, existing video creation models either focus solely on a few specific conditions or suffer from excessively long generation times due to complex model inference, making them impractical for real-world applications. To mitigate these issues, we propose an efficient unified video creation model, named `VDOT`*. Concretely, we model the training process with the distribution matching distillation (DMD) paradigm. Instead of using the Kullback-Leibler (KL) minimization, we additionally employ a novel computational optimal transport (OT) technique to optimize the discrepancy between the real and fake score distributions. The OT distance inherently imposes geometric constraints, mitigating potential zero-forcing or gradient collapse issues that may arise during KL-based distillation within the few-step generation scenario, and thus, enhances the efficiency and stability of the distillation process. Further, we integrate a discriminator to enable the model to perceive real video data, thereby enhancing the quality of generated videos. To support training unified video creation models, we propose a fully automated pipeline for video data annotation and filtering that accommodates multiple video creation tasks. Meanwhile, we curate a unified testing benchmark,* `UVCBench`*, to standardize evaluation. Experiments demonstrate that our 4-step* `VDOT` *outperforms or matches other baselines with 100 denoising steps.*

1. Introduction

Recent years have witnessed the remarkable development of AI-generated content (AIGC). Particularly in image and video generation, the impressive capability to generate high-quality perceptual data has attracted interest from both academia and industry. The ability to synthesize and edit media under diverse conditions has significantly broadened the scope of application for generative models, facilitating their integration across a wide range of domains.

Despite progress, most existing generative models remain task-specific, with models tailored to narrowly defined objectives. While unified image generation and editing have seen notable breakthroughs [43, 47, 58, 61, 62], attempts at unified video creation remain comparatively limited [26, 40, 65]. Recently, VACE [26] reformulates various conditioning signals into unified frame and mask representations and introduces additional adapters for context processing. Similarly, UNIC [65] proposed a unified token-based framework that encodes all inputs into three token categories, combined with native attention and task-aware rotary position embeddings (RoPE) to differentiate tasks. Although these methods achieve impressive visual fidelity, their architectural complexity and large parameter budgets result in substantial inference latency, limiting their practicality in real-world deployments.

To address the above challenges, we propose a novel distillation framework for unified video creation based on computational optimal transport (OT) techniques. Specifically, we formulate the distillation within the distribution-matching distillation paradigm. Instead of solely relying on the conventional reverse KL divergence between the teacher and student score distributions, we incorporate an OT-based discrepancy that enforces geometrically meaningful alignment. This constraint regularizes the transport direction and effectively mitigates the collapse problem that often arises in few-step distillation. Furthermore, we employ an adversarial discriminator that leverages real videos to improve fidelity and counteract undesirable biases inherited from the foundation-scale video models. We adopt an alternating optimization scheme to jointly update the generator and critics, yielding a few-step unified video creator with improved visual quality and efficiency.

Due to the scarcity of large-scale open-source datasets and evaluations for unified video creation, we additionally construct a comprehensive multi-task training dataset and

*Corresponding author.

evaluation benchmark, termed UVCBench. The construction pipeline is fully automated, including 4K-resolution video collection, dense captioning via vision-language models, task-aware data filtering, and candidate ranking, ensuring high-quality and diverse samples that support unified conditioning. To enable standardized and scalable evaluation, UVCBench supports 18 generation tasks, each with 20 representative test cases spanning a broad range of video types. We hope that this automated data construction pipeline and unified benchmark will advance future research in this area.

In summary, the contributions of this work are two-fold:

- We propose VDOT, an efficient unified video creation framework based on optimal-transport distillation. The OT regularizer provides a geometric constraint to distribution matching, improving training stability and efficiency. To the best of our knowledge, this is the first application of OT within distribution-matching distillation.
- We develop a fully automated multi-task data construction pipeline and curate a comprehensive benchmark, UVCBench. Experiments on UVCBench demonstrate that our unified video creator achieves superior performance on both objective metrics and human evaluations while maintaining few-step inference.

2. Related Works

2.1. Visual Creation and Editing

The rapid advancement of image [7, 16, 31, 44] and video [20, 30, 36, 52, 57] generation models has significantly impacted advertising, film production, e-commerce, and interactive entertainment [41, 54]. To meet diverse and personalized application needs, numerous methods for precise control and editing have emerged [47, 69]. Most of these approaches produce high-quality visuals conditioned on pose, depth, optical flow, or reference images. Conditional image generation is commonly enabled via ControlNet [70] or T2I-Adapter [39], while OmniGen [62] and OmniControl [47] extend to multi-task image editing and generation. By contrast, most video editing systems remain specialized, such as animating characters [13, 22], canvas outpainting [14], and colorization [68]. Recent efforts toward all-in-one video creation and editing, such as VACE [26] and UNIC [65], show promise but suffer from complex architectures and large parameter counts, leading to long processing times. In this paper, we build on VACE and distill it into a few-step generator; moreover, by integrating a discriminator, we suppress undesirable artifacts and biases present in the base model.

2.2. Visual Distillation

In visual distillation, several influential approaches have recently emerged, including Progressive Distillation [45], Consistency Distillation [19, 29, 53], Score Distillation [28, 59], Rectified Flow [32, 33, 63], and Adversarial Distillation [17, 34, 46]. These methods typically train a student generator to follow the teacher's ODE-defined sampling trajectory using substantially fewer steps. Distribution matching distillation (DMD) [66, 67] acts in another way. It extends the score distillation by minimizing the expectation over t of approximate KL divergences between the teacher's diffused data distribution and the student's diffused output distribution. Self-Forcing [23] extends the DMD paradigm to the video by using a DMD objective and a denoising loss to distill a few-step video generator. However, in the few-step regime, relying solely on DMD loss can lead to zero-forcing and gradient collapse, resulting in unstable training and susceptibility to model-seeking problems. To address these issues, we propose a novel distribution matching objective that imposes geometric constraints on distribution alignment, enhancing training stability and accelerating convergence.

2.3. Optimal Transport in Computer Vision

The computational optimal transport (OT) techniques [42] have become increasingly popular tools in computer vision for aligning probability distributions. These methods have been successfully applied to point cloud registration [6], image generation [15, 48], as well as video understanding [35, 55, 56] and generation [1]. Many challenging optimization problems can be cast as minimizing the OT distance. In high-dimensional generative modeling, minimizing the Wasserstein distance between data and model distributions motivates the Wasserstein autoencoders [49], while maximizing the Kantorovich dual yields the Wasserstein GAN (WGAN) [4]. Beyond generative modeling, the detector YOLOX adopts Optimal Transport Alignment (OTA) for label assignment, exploiting OT's strong matching behavior [18]. These successes motivate an OT-based objective for distribution matching for few-step video distillation.

3. Preliminary

3.1. Multimodal Inputs and Video Condition Unit

Existing video editing and creation tasks differ in objectives and input forms, yet all can be represented through four basic modalities: **text**, **image**, **video**, and **mask**. All creation tasks can be categorized into five classes: **Text-to-Video Generation (T2V)**, **Reference-to-Video Generation (R2V)**, **Video-to-Video Generation (V2V)**, **Masked Video-to-Video Generation (MV2V)**, and **composite tasks**. Following VACE [26], we introduce the Video Condition Unit (VCU):

$$V = [T; F; M], \tag{1}$$

where T is a text prompt, $F = \{u_1, \ldots, u_n\}$ denotes context frames, and $M = \{m_1, \ldots, m_n\}$ consists of aligned

Table 1. The representation of frames (Fs) and masks (Ms) under the four basic tasks [26].

Tasks	Frames (Fs) & Masks (Ms)
T2V	$F = \{0_{h\times w}\} \times n$ $M = \{1_{h\times w}\} \times n$
R2V	$F = \{r_1, r_2, ..., r_l\} + \{0_{h\times w}\} \times n$ $M = \{0_{h\times w}\} \times l + \{1_{h\times w}\} \times n$
V2V	$F = \{u_1, u_2, ..., u_n\}$ $M = \{1_{h\times w}\} \times n$
MV2V	$F = \{u_1, u_2, ..., u_n\}$ $M = \{m_1, m_2, ..., m_n\}$

binary masks. Each frame $u_i \in [-1, 1]^{3\times h\times w}$ and mask $m_i \in \{0, 1\}^{h\times w}$ indicate editable regions, in which "1"s and "0"s symbolize where to edit or not. As illustrated in Table 1, by adjusting the frames and masks, the VCU generalizes to all video tasks.

3.2. Wasserstein Discrepancy

Given two distributions $\boldsymbol{A} \in \mathbb{R}^{I\times D}$ and $\boldsymbol{B} \in \mathbb{R}^{J\times D}$, an effective way to measure their difference is through the sample-based Wasserstein discrepancy [12], defined as:

$$\begin{aligned}\mathbb{W}_2(\boldsymbol{A}, \boldsymbol{B}) &= \min_{\boldsymbol{T}\in\Pi(\boldsymbol{u},\boldsymbol{\mu})} \mathbb{E}_{(\boldsymbol{a},\boldsymbol{b})\sim\boldsymbol{T}}[d(\boldsymbol{a}, \boldsymbol{b})] \\ &= \min_{\boldsymbol{T}\in\Pi(\boldsymbol{u},\boldsymbol{\mu})} \langle \boldsymbol{D}_{ab},\, \boldsymbol{T}\rangle,\end{aligned} \tag{2}$$

where $\boldsymbol{D}_{ab} = [d(\boldsymbol{a}_i, \boldsymbol{b}_j)] \in \mathbb{R}^{I\times J}$ is a distance matrix, with each element $d(\boldsymbol{a}_i, \boldsymbol{b}_j)$ representing the distance between the i-th sample from $\boldsymbol{A}$ and the j-th sample from $\boldsymbol{B}$. For the Wasserstein distance, we typically apply the Euclidean distance matrix. The set $\Pi(\boldsymbol{u}, \boldsymbol{\mu}) = \{\boldsymbol{T} \geq 0 \mid \boldsymbol{T}\mathbf{1}_J = \boldsymbol{u}, \boldsymbol{T}^T\mathbf{1}_I = \boldsymbol{\mu}\}$ represents the set of doubly-stochastic matrices, where the marginals must lie on the Simplex, i.e., $\boldsymbol{u} \in \Delta^{I-1}$ and $\boldsymbol{\mu} \in \Delta^{J-1}$. Generally, we set the marginals to be uniform, i.e., $\boldsymbol{u} = \frac{1}{I}\mathbf{1}_I$ and $\boldsymbol{\mu} = \frac{1}{J}\mathbf{1}_J$. The optimal transport matrix corresponding to $\mathbb{W}_2(\boldsymbol{A}, \boldsymbol{B})$, denoted as $\boldsymbol{T}^* = [\boldsymbol{t}^*_{ij}]$, is the optimal joint distribution of the samples and targets that minimizes the expectation of the distance.

3.3. Distribution Matching Distillation

Distribution Matching Distillation (DMD) [66, 67] is a method for distilling pretrained diffusion models $\boldsymbol{F}_\psi$ into efficient one-step or multi-step generators $\boldsymbol{G}_\theta$ by minimizing the reverse Kullback-Leibler (KL) divergence between the teacher (real) distribution $\boldsymbol{p}_{\text{real}}$ and the student (fake) distribution $\boldsymbol{p}_{\text{fake}}$ generated by the model. The reverse KL divergence is given by:

$$\mathbb{D}_{\text{KL}}(\boldsymbol{p}_{\text{fake}} \| \boldsymbol{p}_{\text{real}}) = \int \boldsymbol{p}_{\text{fake}}(\boldsymbol{x}) \log \frac{\boldsymbol{p}_{\text{fake}}(\boldsymbol{x})}{\boldsymbol{p}_{\text{real}}(\boldsymbol{x})}\, d\boldsymbol{x}, \tag{3}$$

This divergence quantifies the information lost when $\boldsymbol{p}_{\text{fake}}$ is used to approximate $\boldsymbol{p}_{\text{real}}$, and minimizing it aligns the generated distribution with the real data distribution. The gradient of the DMD objective with respect to the generator parameters θ is given by:

$$\nabla_\theta \mathcal{L}_{\text{DMD}} = \mathbb{E}_{\boldsymbol{z},t',t,\boldsymbol{x}_t}\left[(\boldsymbol{s}_{\text{real}}(\boldsymbol{x}_t) - \boldsymbol{s}_{\text{fake}}(\boldsymbol{x}_t))\frac{d\boldsymbol{G}_\theta(\boldsymbol{z}, t')}{d\theta}\right], \tag{4}$$

where $\boldsymbol{z} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$ is a random latent variable. $t' \sim \mathcal{U}(0, T)$ is randomly selected from the generator schedule, and $\boldsymbol{x}_t$ is the noisy sample obtained by diffusing the generator output $\hat{\boldsymbol{x}}_0$, $\boldsymbol{x}_t = \boldsymbol{q}(\boldsymbol{x}_t|\hat{\boldsymbol{x}}_0)$. The real score $\boldsymbol{s}_{\text{real}}(\boldsymbol{x}_t)$ and the fake score $\boldsymbol{s}_{\text{fake}}(\boldsymbol{x}_t)$ are the gradients of the log probabilities of the real and fake distributions, respectively:

$$\boldsymbol{s}_{\{\text{real / fake}\}}(\boldsymbol{x}_t) = \nabla_{\boldsymbol{x}_t} \log \boldsymbol{p}_{\{\text{real / fake}\}}(\boldsymbol{x}_t). \tag{5}$$

The final DMD loss is computed as:

$$\mathcal{L}_{\text{DMD}}(\theta) = \mathbb{E}_{\boldsymbol{z},t,\boldsymbol{x}_t}\left[\|\hat{\boldsymbol{x}}_0 - \text{sg}(\hat{\boldsymbol{x}}_0 - \nabla_{\text{KL}}(\boldsymbol{x}_t, t))\|_2^2\right], \tag{6}$$

where $\text{sg}(\cdot)$ denotes the stop-gradient operation. In practice, the gradient ∇_{KL} can be approximated by the minus of score functions, $\nabla_x \mathbb{D}_{\text{KL}}(\boldsymbol{p}_{\text{fake}} \| \boldsymbol{p}_{\text{real}}) \approx \boldsymbol{s}_{\text{fake}}(x) - \boldsymbol{s}_{\text{real}}(x)$.

4. Methodology

4.1. Overview

VDOT is designed as an efficient unified video creator that accepts text, images, videos, and masks as inputs and produces a task-compliant output video with few denoising steps. We adopt the pretrained VACE-Wan2.1-14B [26] as the base generator, a state-of-the-art all-in-one model for video creation and editing. VACE comprises frozen Wan DiT blocks and newly introduced VACE DiT blocks; the latter process contextual information derived from conditional inputs. A Video Condition Unit (VCU) converts heterogeneous conditioning signals into a unified representation. Inputs are then preprocessed and tokenized into video, text, and context tokens. Wan blocks process the video tokens, while VACE blocks handle the context tokens; the outputs of the VACE blocks are fused into the corresponding layers of the Wan backbone. Further architectural details can be found in the VACE paper [26]. For brevity, we omit the explicit notation for text and context tokens in the following sections.

To enable few-step video generation, we propose a computational optimal-transport (OT)–based step-distillation framework. Specifically, we cast training of the few-step generator $\boldsymbol{G}_\theta$ within the Distribution-Matching Distillation (DMD) paradigm, employing two score networks, $\boldsymbol{F}_\psi$ and $\boldsymbol{F}_\phi$, that estimate the teacher (real) and student (fake) score distribution, respectively. Instead of relying solely on KL

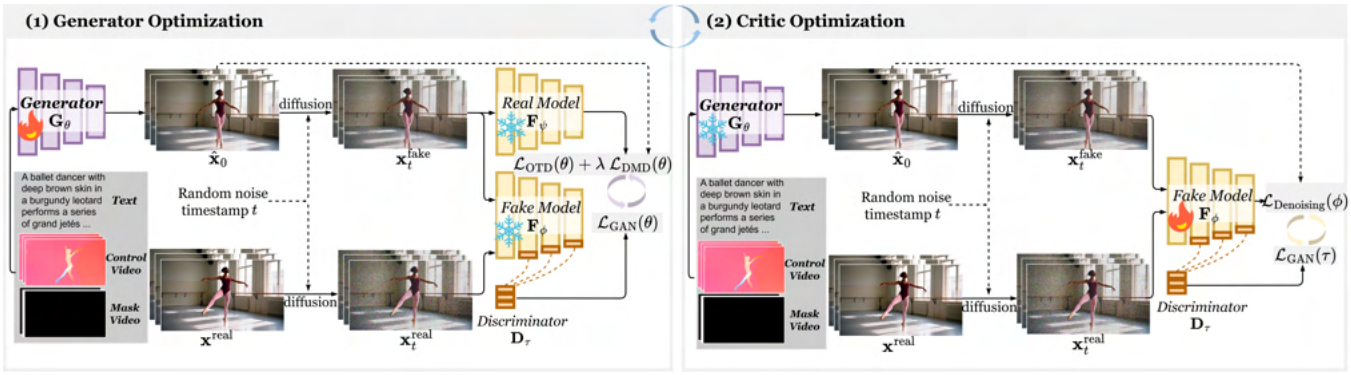

Figure 1. **The training pipeline of VDOT.** By integrating the OTD loss, DMD loss, and GAN loss, we distill the teacher model into a few-step unified video creator. At each step, we alternately train the generator and the critics, while between steps, we alternate between distribution matching and adversarial objectives.

minimization, we incorporate an OT discrepancy to constrain the distribution matching geometrically. We further augment the fake model with an adversarial discriminator to correct score-approximation errors and mitigate biases inherited from the base model by leveraging real video data. Within each step, we alternately train the generator and the critics, while between steps, we alternate between distribution matching objectives and adversarial objectives. An overview of the framework is illustrated in Figure 1.

4.2. Optimal Transport Distillation

The objective of DMD is to minimize the reverse KL divergence, $\mathbb{D}_{\text{KL}}(\boldsymbol{p}_{\text{fake}} \| \boldsymbol{p}_{\text{real}})$. The fake score function enables backpropagation, allowing the generator $\boldsymbol{G}_\theta$ to update and move the student distribution closer to the teacher distribution. However, a common issue that arises is model seeking.

As shown in Equation (3), reverse KL tends to overemphasize regions of the target distribution where the model has high probability, often ignoring areas of the target distribution with lower probability. This leads the model to seek and overfit specific regions. In the few-step generation scenario, this issue is exacerbated. For example, when training a few-step generator, the difference between the real and fake score distributions is initially large. Without directional guidance, model training is prone to encountering zero-forcing or gradient collapse, ultimately failing to capture the diversity of the target distribution and generalizing poorly across the entire distribution, as shown in Figure 2.

- For any point in $\{x \mid \boldsymbol{p}_{\text{fake}}(x) \to 0 \text{ and } \boldsymbol{p}_{\text{real}}(x) > 0\}$, the integrated $\mathbb{D}_{\text{KL}} \to 0$. This causes the models to neglect regions where $\boldsymbol{p}_{\text{real}}(x) > 0$, preventing those regions from updating. This is the so-called zero-forcing problem [34]. As a result, the student distribution fails to fully cover the teacher distribution.
- For any point in $\{x \mid \boldsymbol{p}_{\text{fake}}(x) > 0 \text{ and } \boldsymbol{p}_{\text{real}}(x) \to 0\}$, the integrated $\mathbb{D}_{\text{KL}} \to +\infty$, leading to gradient collapse or training instability.

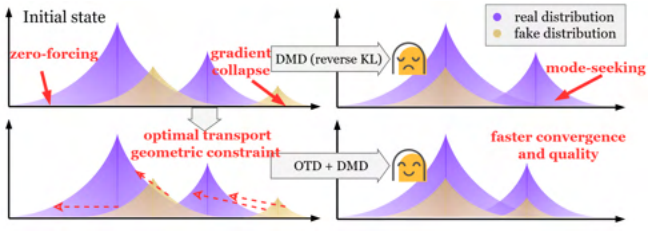

Figure 2. Illustration of potential problems caused by reverse KL divergence and the strength of optimal transport constraint.

The work in ADP [34] addresses this issue by pretraining the fake model via adversarial optimization, thereby alleviating the mode-seeking problem. However, this paradigm requires collecting numerous ODE pairs from the offline teacher model and generating noisy samples through interpolation, which is both costly and labor-intensive.

In this paper, we introduce a computational optimal transport discrepancy as a geometric constraint to aid the optimization of the generator $\boldsymbol{G}_\theta$. The OT discrepancy calculates the minimum transport cost between two distributions and the corresponding optimal transport plan, establishing a one-to-one correspondence. Concretely, for two score distributions $\boldsymbol{p}_{\text{fake}} = [a_i] \in \mathbb{R}^{I \times D}$ and $\boldsymbol{p}_{\text{real}} = [b_j] \in \mathbb{R}^{J \times D}$, we apply the entropic optimal transport (EOT) discrepancy:

$$\mathbb{W}_2^\epsilon(\boldsymbol{p}_{\text{fake}}, \boldsymbol{p}_{\text{real}}) = \min_{\boldsymbol{T} \in \Pi(\boldsymbol{u}, \boldsymbol{\mu})} \langle \boldsymbol{D}, \boldsymbol{T} \rangle + \epsilon \underbrace{\langle \boldsymbol{T}, \log \boldsymbol{T} \rangle}_{\text{Entropy Term}}, \tag{7}$$

where ϵ controls the intensity of the entropy term. The EOT problem can be solved efficiently using the Sinkhorn algorithm [12] with complexity $\mathcal{O}(IJ)$.

According to the envelope theorems [38], the derivative of the objective function with respect to $\boldsymbol{D}$ is the optimal transport plan $\boldsymbol{T}^*$, $\frac{\partial \mathbb{W}_2^\epsilon}{\partial \boldsymbol{D}} = \boldsymbol{T}^*$. When the distance matrix is defined by Euclidean distance, $\boldsymbol{d}(\boldsymbol{a}_i, \boldsymbol{b}_j) = \frac{1}{2}\|\boldsymbol{a}_i - \boldsymbol{b}_j\|^2$,

the gradient of the objective with respect to any $\boldsymbol{a}_i$ is:

$$\nabla_{\boldsymbol{a}_i} \mathbb{W}_2^{\epsilon} = \sum_j \boldsymbol{T}_{ij}^* \nabla_{\boldsymbol{a}_i} \frac{1}{2} \|\boldsymbol{a}_i - \boldsymbol{b}_j\|^2 = \sum_j \boldsymbol{T}_{ij}^* (\boldsymbol{a}_i - \boldsymbol{b}_j), \tag{8}$$

We can then compute the gradient of the objective with respect to the noisy sample $\boldsymbol{x}_t$:

$$\nabla_{\text{OT}}(\boldsymbol{x}_t, t) = \nabla_{\boldsymbol{x}_t} \mathbb{W}_2^{\epsilon} = \sum_{ij} \frac{\partial}{\partial \boldsymbol{x}_t} \left(\boldsymbol{T}_{ij}^* (a_i - b_j) \right). \tag{9}$$

In practice, we compute the gradient $\nabla_{\text{OT}}(\boldsymbol{x}_t, t)$ through `torch.autograd`. Finally, the optimal transport distillation (OTD) loss is computed in the same manner as the DMD loss:

$$\mathcal{L}_{\text{OTD}}(\theta) = \mathbb{E}_{\boldsymbol{z}, t, \boldsymbol{x}_t} \left[\|\hat{\boldsymbol{x}}_0 - \text{sg}(\hat{\boldsymbol{x}}_0 - \nabla_{\text{OT}}(\boldsymbol{x}_t, t))\|_2^2 \right]. \tag{10}$$

4.3. Generative Adversarial Networks

The framework described above does not employ the Teacher-Forcing paradigm [27, 71], which typically requires real video data as denoising conditions. Instead, following the Self-Forcing approach [23], it uses previously denoised frames to denoise the current frame, thus maintaining consistency between training and testing. However, relying solely on distribution matching objectives, without access to real data, leads to approximation errors in the real score function $\boldsymbol{F}_\psi$ [66], which manifest as artifacts in video textures and details. Moreover, without real data inputs, the output quality of the generator $\boldsymbol{G}_\theta$ is constrained by the teacher model, while also learning some of the teacher's undesirable prototypes, as discussed further in the Appendix.

To address this limitation, we introduce real data and a discriminator to correct the score functions by incorporating the generative adversarial networks (GANs) objective. Specifically, we select three blocks, blocks 23, 31, and 39, from the denoised blocks of the fake score function $\boldsymbol{F}_\phi$. We introduce three learnable registration tokens that interact with the corresponding blocks through cross-attention. The resulting outputs are concatenated along the channel dimension and passed through a linear layer-based classifier that outputs classification logits. We denote the involved registration tokens, cross-attention blocks, and classifier as the discriminator $\boldsymbol{D}_\tau$, with parameter τ. Given a real video corresponding to the input prompt, we first encode it into the same latent space using a pre-trained VAE, denoted as $\boldsymbol{x}^{\text{real}}$. Then, using the randomly sampled timestamp from the scheduler, we add noise to $\boldsymbol{x}^{\text{real}}$ and $\hat{\boldsymbol{x}}_0$, yielding $\boldsymbol{x}_t^{\text{real}}$ and $\boldsymbol{x}_t^{\text{fake}}$, respectively. A relative GAN loss is used to calibrate the score functions:

$$\mathcal{L}_{\text{GAN}}(\theta) = \mathop{\mathbb{E}}_{\boldsymbol{z}, t} \left[-\left(\boldsymbol{D}_\tau \left(\boldsymbol{x}_t^{\text{fake}}, t \right) - \boldsymbol{D}_\tau \left(\boldsymbol{x}_t^{\text{real}}, t \right) \right) \right], \tag{11}$$

$$\mathcal{L}_{\text{GAN}}(\tau) = \mathop{\mathbb{E}}_{\boldsymbol{x}_t^{\text{fake}}, t} \left[-\left(\boldsymbol{D}_\tau \left(\boldsymbol{x}_t^{\text{real}}, t \right) - \boldsymbol{D}_\tau \left(\boldsymbol{x}_t^{\text{fake}}, t \right) \right) \right]. \tag{12}$$

4.4. Model Learning

Our training employs an alternating strategy to optimize the generator and critics. The parameters of the real score $\boldsymbol{F}_\psi$ remain frozen throughout training. At each step, we first freeze the fake model $\boldsymbol{F}_\phi$ and train the generator $\boldsymbol{G}_\theta$ using the distribution matching objective $\mathcal{L}_{\text{OTD}}(\theta) + \lambda \mathcal{L}_{\text{DMD}}(\theta)$ or adversarial objective $\mathcal{L}_{\text{GAN}}(\theta)$. Then, we freeze the generator and train both the fake model $\boldsymbol{F}_\phi$ and the discriminator $\boldsymbol{D}_\tau$ using either the diffusion denoising objective $\mathcal{L}_{\text{Denoising}}(\phi)$ or the adversarial objective $\mathcal{L}_{\text{GAN}}(\tau)$.

Algorithm 1: Training Algorithm of `VDOT`.

Input: dataset $\mathcal{D}$, pretrained VACE $\boldsymbol{F}_{\text{pretrain}}$, pretrained few-step Wan $\boldsymbol{W}_{\text{pretrain}}$, Generator $\boldsymbol{G}_\theta$, Fake model $\boldsymbol{F}_\phi$, Real model $\boldsymbol{F}_\psi$, Discriminator $\boldsymbol{D}_\tau$, learning rates η_1, η_2

Init : init $\boldsymbol{G}_\theta$, $\boldsymbol{F}_\phi$ and $\boldsymbol{F}_\psi$ with $\boldsymbol{F}_{\text{pretrain}}$, init $\boldsymbol{G}_\theta$ with $\boldsymbol{W}_{\text{pretrain}}$.

for *step* $\leftarrow 0$ ***to*** *max_step* **do**
- ▷ **Update Generator $\boldsymbol{G}_\theta$**:
- Sample $\boldsymbol{z} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$, $\hat{\boldsymbol{x}}_0 \leftarrow \boldsymbol{G}_\theta(\boldsymbol{z})$
- Sample timestamp t, $\boldsymbol{x}_t \leftarrow \text{add\_noise}(\hat{\boldsymbol{x}}_0, t)$
- if step$\%2 = 0$:
  - Compute $\mathcal{L}_\theta = \mathcal{L}_{\text{OTD}}(\theta) + \lambda \mathcal{L}_{\text{DMD}}(\theta)$ via (10) and (6)
- else:
  - Compute $\mathcal{L}_\theta = \mathcal{L}_{\text{GAN}}(\theta)$ via (11)
- Update $\theta \leftarrow \theta - \eta_1 \nabla_\theta \mathcal{L}_\theta$
- ▷ **Update Fake model $\boldsymbol{F}_\phi$ and Discriminator $\boldsymbol{D}_\tau$**:
- Sample $\boldsymbol{z} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$, $\hat{\boldsymbol{x}}_0 \leftarrow \boldsymbol{G}_\theta(\boldsymbol{z})$, $x^{\text{real}} \sim \mathcal{D}$
- Sample timestamp t, $\boldsymbol{x}_t^{\text{fake}} / \boldsymbol{x}_t^{\text{real}} \leftarrow \text{add\_noise}(\hat{\boldsymbol{x}}_0 / \boldsymbol{x}^{\text{real}}, t)$
- if step$\%2 = 0$:
  - Compute diffusion denoising loss $\mathcal{L}_{\text{Denoising}}(\phi)$
  - Update $\phi \leftarrow \phi - \eta_2 \nabla_\phi \mathcal{L}_{\text{Denoising}}$
- else:
  - Compute $\mathcal{L}_\tau = \mathcal{L}_{\text{GAN}}(\tau)$ via (12)
  - Update $\tau \leftarrow \tau - \eta_2 \nabla_\tau \mathcal{L}_\tau$

end

5. Dataset

5.1. Dataset Construction

We construct the training dataset fully automatically. We generate 250,000 4K videos from Artgrid [5] spanning diverse content types, and rescale them to 832×480 for training. For each video, we use InternVL [10] to generate a caption, which serves both as a text prompt for generation and as a criterion for filtering. As an example, for the *pose*-conditioned data pipeline, we first exclude videos without human subjects by applying Qwen3 [64] to the video captions. We then sample batches of videos and their captions from this filtered pool, derive pose videos from the selected videos, and compute a score defined as the weighted average of inter-frame keypoint distances. Finally, we sort videos by this score in descending order and retain the top-ranked pose videos and their captions as training data for the pose task. We apply analogous task-specific procedures for the remaining tasks, yielding a unified multi-task video dataset suitable for training across all creation settings. Per-task training set sizes are reported in the Appendix.

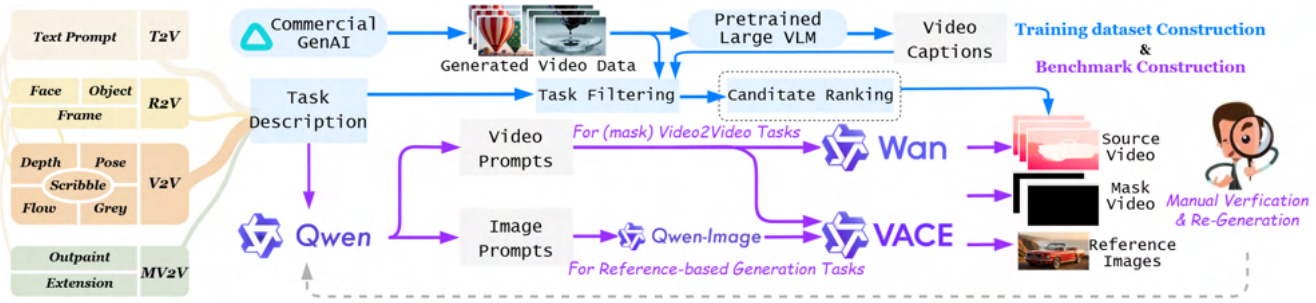

Figure 3. **Construction pipeline of Training dataset (the blue arrow) and `UVCBench` (the purple arrow).**

5.2. UVCBench

Video generation has advanced rapidly. VBench [24] and VBench++ [25] provide established benchmarks for text-to-video and image-to-video tasks. However, a comprehensive and standardized benchmark for the arising video creation tasks is still lacking. Recently, VACE proposed the VACE-Benchmark with 240 high-quality videos across 12 creation tasks, but it lacks evaluation samples for composite tasks. To address this gap and advance this field, we curate `UVCBench`, a new open-source comprehensive benchmark for unified video creation. `UVCBench` covers 18 tasks—8 single-condition and 10 composite-condition settings. Single-condition tasks are defined by the input conditioning signal, including pose, depth, optical flow, greyscale, scribble, reference image (face or object), canvas outpainting, and temporal extension (first, last, or random clip). Composite tasks combine either a reference image or a first-frame image with a second, video-based condition (pose, depth, optical flow, greyscale, or scribble).

We construct the benchmark via an automated pipeline and generate 20 videos per task. As shown in Figure 3, we first elaborate task descriptions and prompt Qwen3-Max [64] to produce candidate video prompts; for reference-based tasks, this yields paired image–video prompts. For V2V and MV2V tasks, we synthesize high-resolution exemplar videos using Wan2.1-14B. For R2V and composite tasks, we first create reference images with Qwen-Image [60] from image prompts, then use VACE-14B to generate exemplar videos conditioned on the reference images and video prompts. After exemplar generation, an annotator derives the corresponding source inputs and mask videos required by each task. Finally, the authors manually verify the results and regenerate failure cases.

6. Experiments

6.1. Experimental Setup

Implementation Details. `VDOT` is trained based on Wan2.1-VACE-14B [26]. The training consists of two stages. *Stage 1* follows the Self-Forcing pipeline [23] to distill Wan2.1–T2V–14B [52] into a few-step generator. We train using only video captions from our Artgrid dataset for 1,500 steps. The learning rates for the generator and the critic are 2×10^{-6} and 4×10^{-7}, respectively. The TTUR ratio is 5. *Stage 2* initializes the generator from Wan2.1–VACE–14B, further pre-initialized with the Stage-1 few-step Wan2.1–T2V–14B weights. This stage uses multi-task video data comprising 8 single-task and 10 composite-task settings and trains for 1,200 steps. The learning rates for the generator and the critic are 1×10^{-6} and 4×10^{-7}, respectively. The TTUR ratio is 5. Both stages use Adam optimizer [2]. All experiments are conducted on 4 NVIDIA H200 GPUs with a batch size of 1 per GPU, and we adopt gradient checkpointing with a size of 4 to optimize memory usage during training.

Baselines. We evaluate `VDOT` against VACE [26], the only open-source all-in-one video creation model, and further include task-specific online and offline methods for comparison. (1) *Video-to-video (V2V)*, under *depth* conditioning—Control-A-Video [9] and ControlVideo [72]; under *pose* conditioning—ControlVideo and Follow-Your-Pose [37]; under *scribble* conditioning—ControlVideo; under *optical-flow* conditioning—FLATTEN [11]. (2) *MV2V*: for *outpainting*—Follow-Your-Canvas [8]. (3) *Reference-based video generation (R2V)*: online systems Keling-1.6 [3] and Vidu-2.0 [50].

Evaluation. For a comprehensive benchmarking, we use our proposed `UVCBench` for evaluation. We use VBench [24] to assess video quality and consistency of generated results across six dimensions: Aesthetic Quality, Background Consistency, Dynamic Degree, Imaging Quality, Motion Smoothness, and Subject Consistency. In addition to automatic scoring, we conduct a user study for subjective evaluation. Following VACE, annotators rate Prompt Following, Temporal Consistency, and Overall Video Quality. Scores are reported on a 1–5 Likert scale.

6.2. Quantitative and Qualitative Comparisons

Table 2 presents comprehensive quantitative comparisons across multiple video creation tasks on `UVCBench`, cover-

Table 2. **Quantitative comparison for various methods on `UVCBench`**. We bold the **best results** and underline the second-best results.

Type	Method	Base Model	#NFE↓	Video Quality & Video Consistency							User Study			
				Aesthetic Quality	Background Consistency	Dynamic Degree	Imaging Quality	Motion Smoothness	Subject Consistency	Normalized Average	Prompt Following	Temporal Consistency	Video Quality	Average
Depth	Control-A-Video [9]	SD-1.5	100	57.22%	92.89%	20.00%	66.95%	98.06%	91.85%	71.16%	2.04	1.85	1.12	1.67
	ControlVideo [72]	SD-1.5	100	**64.50%**	97.99%	5.00%	**71.76%**	98.10%	97.41%	72.46%	2.84	2.78	2.31	2.64
	VACE [26]	LTX-0.9B	80	58.62%	**98.18%**	30.00%	70.14%	99.29%	97.81%	75.67%	2.89	2.82	2.24	2.65
	VACE [26]	Wan-14B	100	64.36%	97.63%	35.00%	70.29%	99.17%	97.58%	77.34%	4.33	**4.41**	**4.65**	**4.46**
	`VDOT` (Ours)	Wan-14B	4	64.28%	**98.18%**	**40.00%**	71.24%	**99.35%**	**97.96%**	**78.50%**	**4.51**	4.27	4.60	**4.46**
Pose	ControlVideo [72]	SD-1.5	100	63.32%	**96.82%**	10.00%	69.88%	98.50%	94.45%	72.16%	2.79	2.65	2.18	2.54
	Follow-Your-Pose [37]	SD-1.4	50	50.36%	88.61%	40.00%	68.73%	91.78%	77.81%	69.55%	1.63	1.42	1.05	1.37
	VACE [26]	LTX-0.9B	80	59.72%	95.67%	45.00%	69.12%	98.10%	**94.80%**	77.06%	2.71	2.72	2.27	2.57
	VACE [26]	Wan-14B	100	**64.99%**	95.48%	55.00%	69.15%	98.64%	94.07%	79.56%	4.44	**4.50**	**4.36**	4.43
	`VDOT` (Ours)	Wan-14B	4	63.50%	95.88%	**60.00%**	**70.65%**	**98.69%**	94.53%	**80.54%**	**4.75**	4.37	4.28	**4.47**
Flow	FLATTEN [11]	SD-2.1	100	**63.86%**	**96.68%**	60.00%	53.28%	96.35%	93.39%	77.26%	3.05	2.89	3.31	3.08
	VACE [26]	LTX-0.9B	80	55.47%	95.97%	65.00%	57.51%	97.62%	92.90%	77.41%	2.56	2.43	2.93	2.64
	VACE [26]	Wan-14B	100	62.78%	95.78%	**75.00%**	58.01%	97.77%	92.75%	**80.35%**	4.39	4.40	4.55	4.45
	`VDOT` (Ours)	Wan-14B	4	62.78%	96.36%	70.00%	**59.22%**	**98.84%**	**93.93%**	80.18%	**4.45**	**4.52**	**4.57**	**4.51**
Scribble	ControlVideo [72]	SD-1.5	100	54.79%	96.19%	5.00%	64.30%	98.53%	94.64%	68.91%	3.78	2.27	2.39	2.81
	VACE [26]	LTX-0.9B	80	44.69%	96.88%	40.00%	61.06%	**99.17%**	93.51%	72.55%	3.99	3.71	3.46	3.72
	VACE [26]	Wan-14B	100	**55.70%**	96.91%	**55.00%**	58.69%	98.99%	94.51%	76.63%	**4.67**	4.67	**4.82**	**4.72**
	`VDOT` (Ours)	Wan-14B	4	54.24%	**97.13%**	50.00%	**65.19%**	98.90%	**95.20%**	**76.77%**	4.62	**4.74**	4.78	4.71
Grey	VACE [26]	LTX-0.9B	80	60.64%	98.09%	5.00%	58.29%	**99.37%**	97.14%	69.75%	4.30	4.07	3.75	4.04
	VACE [26]	Wan-14B	100	**63.76%**	**98.11%**	**15.00%**	**60.90%**	99.30%	97.56%	**72.44%**	4.65	**4.79**	4.81	4.75
	`VDOT` (Ours)	Wan-14B	4	62.13%	**98.11%**	**15.00%**	60.60%	99.34%	**97.62%**	72.13%	**4.70**	4.75	**4.84**	**4.76**
Outpaint	Follow-Your-Canvas [8]	SD-2.1	80	51.98%	97.13%	25.00%	66.95%	98.92%	95.55%	72.59%	3.75	3.89	3.51	3.72
	VACE [26]	LTX-0.9B	80	58.78%	98.07%	25.00%	70.74%	**99.20%**	97.35%	74.86%	4.20	3.44	3.77	3.80
	VACE [26]	Wan-14B	100	62.66%	98.13%	25.00%	70.56%	99.18%	**97.41%**	75.49%	**4.61**	4.63	4.47	4.57
	`VDOT` (Ours)	Wan-14B	4	**63.18%**	**98.22%**	**30.00%**	**71.38%**	99.16%	97.34%	**76.54%**	4.52	**4.70**	**4.63**	**4.62**
Extension	VACE [26]	LTX-0.9B	80	51.54%	**96.48%**	20.00%	62.22%	**99.21%**	**94.67%**	70.68%	3.50	2.85	3.14	3.16
	VACE [26]	Wan-14B	100	57.52%	95.46%	55.00%	63.36%	98.99%	92.27%	77.10%	**4.55**	**4.54**	**4.48**	**4.52**
	`VDOT` (Ours)	Wan-14B	4	**58.93%**	94.18%	**75.00%**	**65.11%**	**99.21%**	91.09%	**80.53%**	4.32	4.40	4.35	4.36
R2V	Keling1.6 [3]	-	-	63.61%	93.85%	**90.00%**	64.46%	**98.95%**	90.11%	**83.50%**	**4.82**	**4.78**	**4.85**	**4.82**
	Vidu2.0 [50]	-	-	60.79%	93.22%	85.00%	54.26%	97.54%	87.42%	79.71%	4.52	4.33	4.50	4.45
	VACE [26]	LTX-0.9B	80	50.18%	91.31%	50.00%	55.88%	98.84%	85.49%	71.95%	2.36	2.03	3.11	2.50
	VACE [26]	Wan-14B	100	67.39%	**95.39%**	80.00%	62.82%	97.96%	**91.70%**	82.54%	4.65	4.71	4.63	4.66
	`VDOT` (Ours)	Wan-14B	4	**69.70%**	94.34%	70.00%	**65.93%**	98.15%	89.88%	81.32%	4.62	4.65	4.66	4.64

Table 3. **Ablation Study on `UVCBench`**. Base Model is VACE-Wan2.1-14B. AccWanInit refers to initializing the Wan blocks in the VACE model using a few-step Wan model.

	DMD	OTD	GAN	AccWanInit	Depth	Pose	Flow	Scribble	Grey	Outpaint	Extention	R2V
(1)	✗	✗	✗	✓	77.82%	80.07%	79.71%	75.17%	71.69%	75.38%	76.01%	80.17%
(2)	✓	✗	✗	✗	76.89%	78.34%	79.12%	75.79%	71.70%	74.21%	77.33%	76.45%
(3)	✓	✓	✗	✓	78.24%	80.14%	79.95%	76.90%	71.59%	76.44%	79.01%	77.66%
(4)	✓	✗	✓	✓	78.05%	**80.79%**	80.15%	76.58%	71.98%	76.40%	78.88%	78.40%
(5)	✓	✓	✓	✗	77.15%	79.83%	79.16%	**77.19%**	**72.21%**	75.57%	79.76%	77.00%
`VDOT`	✓	✓	✓	✓	**78.50%**	80.54%	**80.18%**	76.77%	72.13%	**76.54%**	**80.53%**	**81.32%**

ing both objective video metrics and subjective user preference. In terms of video quality and temporal consistency, `VDOT` delivers competitive or superior performance while requiring drastically fewer inference steps. In particular,

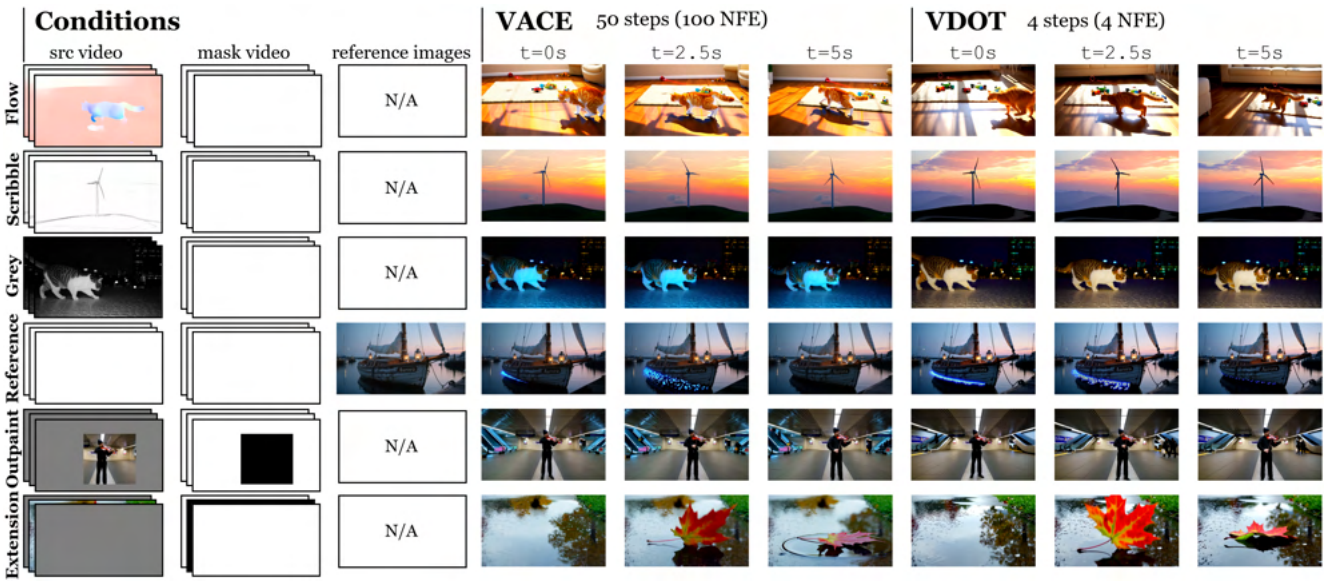

Figure 4. **Qualitative comparison between generated videos.** VDOT can generate comparable visual fidelity with 4 denoising steps compared with VACE-Wan2.1-14B with 50 denoising steps. Refer to the Appendix for more visualizations. Zoom in for more details.

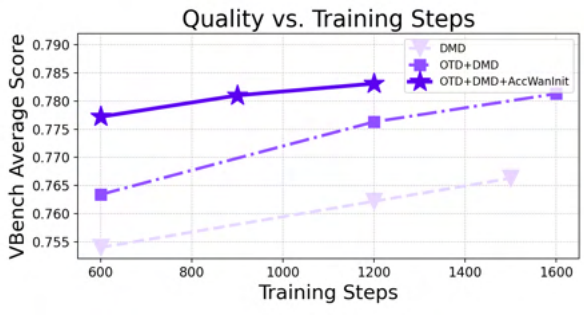

Figure 5. **Training efficiency comparison.**

for *Imaging Quality*, VDOT attains the best or second-best scores across all tasks. These results highlight the efficiency and generality of VDOT. We also include a user study comparison to assess *Prompt Following*, *Temporal Consistency*, and *Overall Video Quality* across all creation tasks. We establish a website and invite 20 volunteers to provide ratings. For each metric, scores are averaged across raters. In summary, VDOT achieves encouraging performance compared with the open-source baselines or commercial products in average user preference, validating the effectiveness of our proposed framework.

Figure 4 presents qualitative comparisons between VACE and VDOT on five tasks—*flow*, *scribble*, *grey*, *outpainting*, and *temporal extension*. VDOT achieves comparable visual appearance and structural consistency to VACE while using far fewer inference steps. More quantitative results and visualizations about the composite tasks can be found in the Appendix.

6.3. Ablation Study

Table 3 presents a comprehensive ablation study evaluating the contributions of each component in VDOT using VACE-Wan2.1-14B as the base model. All variants (except row 1) are trained for 1,200 steps. Row (1) is the baseline, where we replace only the Wan blocks in VACE [26] with a distilled Wan. Row (2) corresponds to the Self-Forcing [23] paradigm and employs only the DMD loss. Removing GAN objectives (row 3) or OTD objective (row 4) leads to consistent degradation across most metrics, confirming their necessary roles in video distillation. Row (5) indicates that AccWanInit supplies a stronger initialization, substantially improving training efficiency. As shown in Figure 5, we also plot a quality versus training steps graph to demonstrate the training efficiency gains afforded by OTD and AccWanInit.

7. Conclusion

In this work, we propose VDOT, an efficient unified video creation model. We introduce a novel computational optimal transport–based distribution matching objective which, together with DMD and GAN losses, enables few-step distillation of video models with improved visual quality and enhanced both training and inference efficiency. To support multi-task video creation, we develop a fully automated pipeline for training data annotation and filtering, and curate a unified benchmark, UVCBench, to enable fair and comprehensive evaluation. Experiments on UVCBench show that our method achieves encouraging performance in unified video creation using only few denoising steps.

References

[1] Dinesh Acharya, Zhiwu Huang, Danda Pani Paudel, and Luc Van Gool. Towards high resolution video generation with progressive growing of sliced wasserstein gans. *arXiv preprint arXiv:1810.02419*, 2018. 2

[2] Kingma DP Ba J Adam et al. A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 1412(6), 2014. 6

[3] KLING AI. Kling ai. *https://klingai.com/*, 2025. 6, 7

[4] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International conference on machine learning*, pages 214–223. PMLR, 2017. 2

[5] Artgrid. Artgrid. *https://artgrid.io/*, 2025. 5

[6] Nicolas Bonneel and David Coeurjolly. Spot: sliced partial optimal transport. *ACM Transactions on Graphics (TOG)*, 38(4):1–13, 2019. 2

[7] Junsong Chen, Jincheng Yu, Chongjian Ge, Lewei Yao, Enze Xie, Yue Wu, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, et al. Pixart-alpha: Fast training of diffusion transformer for photorealistic text-to-image synthesis. *arXiv preprint arXiv:2310.00426*, 2023. 2

[8] Qihua Chen, Yue Ma, Hongfa Wang, Junkun Yuan, Wenzhe Zhao, Qi Tian, Hongmei Wang, Shaobo Min, Qifeng Chen, and Wei Liu. Follow-your-canvas: Higher-resolution video outpainting with extensive content generation. *arXiv preprint arXiv:2409.01055*, 2024. 6, 7

[9] Weifeng Chen, Yatai Ji, Jie Wu, Hefeng Wu, Pan Xie, Jiashi Li, Xin Xia, Xuefeng Xiao, and Liang Lin. Control-a-video: Controllable text-to-video generation with diffusion models. *arXiv e-prints*, pages arXiv–2305, 2023. 6, 7

[10] Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 24185–24198, 2024. 5

[11] Yuren Cong, Mengmeng Xu, Christian Simon, Shoufa Chen, Jiawei Ren, Yanping Xie, Juan-Manuel Perez-Rua, Bodo Rosenhahn, Tao Xiang, and Sen He. Flatten: optical flow-guided attention for consistent text-to-video editing. *arXiv preprint arXiv:2310.05922*, 2023. 6, 7

[12] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26, 2013. 3, 4

[13] Zuozhuo Dai, Zhenghao Zhang, Yao Yao, Bingxue Qiu, Siyu Zhu, Long Qin, and Weizhi Wang. Animateanything: Fine-grained open domain image animation with motion guidance, 2023. 2

[14] Loïc Dehan, Wiebe Van Ranst, Patrick Vandewalle, and Toon Goedemé. Complete and temporally consistent video outpainting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 687–695, 2022. 2

[15] Yonatan Dukler, Wuchen Li, Alex Lin, and Guido Montúfar. Wasserstein of wasserstein loss for learning generative models. In *International conference on machine learning*, pages 1716–1725. PMLR, 2019. 2

[16] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*, 2024. 2

[17] Xingtong Ge, Xin Zhang, Tongda Xu, Yi Zhang, Xinjie Zhang, Yan Wang, and Jun Zhang. Senseflow: Scaling distribution matching for flow-based text-to-image distillation. *arXiv preprint arXiv:2506.00523*, 2025. 2

[18] Zheng Ge, Songtao Liu, Feng Wang, Zeming Li, and Jian Sun. Yolox: Exceeding yolo series in 2021. *arXiv preprint arXiv:2107.08430*, 2021. 2

[19] Zhengyang Geng, Ashwini Pokle, William Luo, Justin Lin, and J Zico Kolter. Consistency models made easy. *arXiv preprint arXiv:2406.14548*, 2024. 2

[20] 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. 2

[21] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 1

[22] Li Hu. Animate anyone: Consistent and controllable image-to-video synthesis for character animation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8153–8163, 2024. 2

[23] Xun Huang, Zhengqi Li, Guande He, Mingyuan Zhou, and Eli Shechtman. Self forcing: Bridging the train-test gap in autoregressive video diffusion. *arXiv preprint arXiv:2506.08009*, 2025. 2, 5, 6, 8

[24] Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, et al. Vbench: Comprehensive benchmark suite for video generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21807–21818, 2024. 6

[25] Ziqi Huang, Fan Zhang, Xiaojie Xu, Yinan He, Jiashuo Yu, Ziyue Dong, Qianli Ma, Nattapol Chanpaisit, Chenyang Si, Yuming Jiang, et al. Vbench++: Comprehensive and versatile benchmark suite for video generative models. *arXiv preprint arXiv:2411.13503*, 2024. 6

[26] Zeyinzi Jiang, Zhen Han, Chaojie Mao, Jingfeng Zhang, Yulin Pan, and Yu Liu. Vace: All-in-one video creation and editing. *arXiv preprint arXiv:2503.07598*, 2025. 1, 2, 3, 6, 7, 8

[27] Yang Jin, Zhicheng Sun, Ningyuan Li, Kun Xu, Hao Jiang, Nan Zhuang, Quzhe Huang, Yang Song, Yadong MU, and Zhouchen Lin. Pyramidal flow matching for efficient video generative modeling. In *The Thirteenth International Conference on Learning Representations*, 2025. 5

[28] Oren Katzir, Or Patashnik, Daniel Cohen-Or, and Dani Lischinski. Noise-free score distillation. *arXiv preprint arXiv:2310.17590*, 2023. 2

[29] Dongjun Kim, Chieh-Hsin Lai, Wei-Hsiang Liao, Naoki Murata, Yuhta Takida, Toshimitsu Uesaka, Yutong He, Yuki Mitsufuji, and Stefano Ermon. Consistency trajectory models: Learning probability flow ode trajectory of diffusion. *arXiv preprint arXiv:2310.02279*, 2023. 2

[30] Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, et al. Hunyuanvideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024. 2

[31] Zhimin Li, Jianwei Zhang, Qin Lin, Jiangfeng Xiong, Yanxin Long, Xinchi Deng, Yingfang Zhang, Xingchao Liu, Minbin Huang, Zedong Xiao, et al. Hunyuan-dit: A powerful multi-resolution diffusion transformer with fine-grained chinese understanding. *arXiv preprint arXiv:2405.08748*, 2024. 2

[32] Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*, 2022. 2

[33] Xingchao Liu, Xiwen Zhang, Jianzhu Ma, Jian Peng, et al. Instaflow: One step is enough for high-quality diffusion-based text-to-image generation. In *The Twelfth International Conference on Learning Representations*, 2023. 2

[34] Yanzuo Lu, Yuxi Ren, Xin Xia, Shanchuan Lin, Xing Wang, Xuefeng Xiao, Andy J Ma, Xiaohua Xie, and Jian-Huang Lai. Adversarial distribution matching for diffusion distillation towards efficient image and video synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16818–16829, 2025. 2, 4

[35] Dixin Luo, Yutong Wang, Angxiao Yue, and Hongteng Xu. Weakly-supervised temporal action alignment driven by unbalanced spectral fused gromov-wasserstein distance. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 728–739, 2022. 2

[36] Xin Ma, Yaohui Wang, Gengyun Jia, Xinyuan Chen, Ziwei Liu, Yuan-Fang Li, Cunjian Chen, and Yu Qiao. Latte: Latent diffusion transformer for video generation. *arXiv preprint arXiv:2401.03048*, 2024. 2

[37] Yue Ma, Yingqing He, Xiaodong Cun, Xintao Wang, Siran Chen, Xiu Li, and Qifeng Chen. Follow your pose: Pose-guided text-to-video generation using pose-free videos. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 4117–4125, 2024. 6, 7

[38] Paul Milgrom and Ilya Segal. Envelope theorems for arbitrary choice sets. *Econometrica*, 70(2):583–601, 2002. 4

[39] Chong Mou, Xintao Wang, Liangbin Xie, Yanze Wu, Jian Zhang, Zhongang Qi, and Ying Shan. T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. In *Proceedings of the AAAI conference on artificial intelligence*, pages 4296–4304, 2024. 2

[40] Chong Mou, Qichao Sun, Yanze Wu, Pengze Zhang, Xinghui Li, Fulong Ye, Songtao Zhao, and Qian He. Instructx: Towards unified visual editing with mllm guidance. *arXiv preprint arXiv:2510.08485*, 2025. 1

[41] Xingang Pan, Ayush Tewari, Thomas Leimkühler, Lingjie Liu, Abhimitra Meka, and Christian Theobalt. Drag your gan: Interactive point-based manipulation on the generative image manifold. In *ACM SIGGRAPH 2023 conference proceedings*, pages 1–11, 2023. 2

[42] Gabriel Peyré, Marco Cuturi, et al. Computational optimal transport: With applications to data science. *Foundations and Trends® in Machine Learning*, 11(5-6):355–607, 2019. 2, 1

[43] Qi Qin, Le Zhuo, Yi Xin, Ruoyi Du, Zhen Li, Bin Fu, Yiting Lu, Jiakang Yuan, Xinyue Li, Dongyang Liu, et al. Lumina-image 2.0: A unified and efficient image generative framework. *arXiv preprint arXiv:2503.21758*, 2025. 1

[44] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494, 2022. 2

[45] Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. *arXiv preprint arXiv:2202.00512*, 2022. 2

[46] Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. Adversarial diffusion distillation. In *European Conference on Computer Vision*, pages 87–103. Springer, 2024. 2

[47] Zhenxiong Tan, Songhua Liu, Xingyi Yang, Qiaochu Xue, and Xinchao Wang. Ominicontrol: Minimal and universal control for diffusion transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14940–14950, 2025. 1, 2

[48] Guillaume Tartavel, Gabriel Peyré, and Yann Gousseau. Wasserstein loss for image synthesis and restoration. *SIAM Journal on Imaging Sciences*, 9(4):1726–1755, 2016. 2

[49] Ilya Tolstikhin, Olivier Bousquet, Sylvain Gelly, and Bernhard Schoelkopf. Wasserstein auto-encoders. *arXiv preprint arXiv:1711.01558*, 2017. 2

[50] Vidu. Vidu. *https://vidu.cn/*, 2025. 6, 7

[51] Pascal Vincent. A connection between score matching and denoising autoencoders. *Neural computation*, 23(7):1661–1674, 2011. 1

[52] 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. 2, 6

[53] Fu-Yun Wang, Zhaoyang Huang, Alexander Bergman, Dazhong Shen, Peng Gao, Michael Lingelbach, Keqiang Sun, Weikang Bian, Guanglu Song, Yu Liu, et al. Phased consistency models. *Advances in neural information processing systems*, 37:83951–84009, 2024. 2

[54] Qixun Wang, Xu Bai, Haofan Wang, Zekui Qin, Anthony Chen, Huaxia Li, Xu Tang, and Yao Hu. Instantid: Zero-shot identity-preserving generation in seconds. *arXiv preprint arXiv:2401.07519*, 2024. 2

[55] Yutong Wang, Hongteng Xu, and Dixin Luo. Self-supervised video summarization guided by semantic inverse optimal transport. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 6611–6622, 2023. 2

[56] Yutong Wang, Sidan Zhu, Hongteng Xu, and Dixin Luo. An inverse partial optimal transport framework for music-guided trailer generation. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 9739–9748, 2024. 2

[57] Yaohui Wang, Xinyuan Chen, Xin Ma, Shangchen Zhou, Ziqi Huang, Yi Wang, Ceyuan Yang, Yinan He, Jiashuo Yu, Peiqing Yang, et al. Lavie: High-quality video generation with cascaded latent diffusion models. *International Journal of Computer Vision*, 133(5):3059–3078, 2025. 2

[58] Zhenyu Wang, Aoxue Li, Zhenguo Li, and Xihui Liu. Genartist: Multimodal llm as an agent for unified image generation and editing. *Advances in Neural Information Processing Systems*, 37:128374–128395, 2024. 1

[59] Min Wei, Jingkai Zhou, Junyao Sun, and Xuesong Zhang. Adversarial score distillation: When score distillation meets gan. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8131–8141, 2024. 2

[60] Chenfei Wu, Jiahao Li, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kun Yan, Sheng ming Yin, Shuai Bai, Xiao Xu, Yilei Chen, Yuxiang Chen, Zecheng Tang, Zekai Zhang, Zhengyi Wang, An Yang, Bowen Yu, Chen Cheng, Dayiheng Liu, Deqing Li, Hang Zhang, Hao Meng, Hu Wei, Jingyuan Ni, Kai Chen, Kuan Cao, Liang Peng, Lin Qu, Minggang Wu, Peng Wang, Shuting Yu, Tingkun Wen, Wensen Feng, Xiaoxiao Xu, Yi Wang, Yichang Zhang, Yongqiang Zhu, Yujia Wu, Yuxuan Cai, and Zenan Liu. Qwen-image technical report, 2025. 6

[61] Bin Xia, Yuechen Zhang, Jingyao Li, Chengyao Wang, Yitong Wang, Xinglong Wu, Bei Yu, and Jiaya Jia. Dreamomni: Unified image generation and editing. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 28533–28543, 2025. 1

[62] Shitao Xiao, Yueze Wang, Junjie Zhou, Huaying Yuan, Xingrun Xing, Ruiran Yan, Chaofan Li, Shuting Wang, Tiejun Huang, and Zheng Liu. Omnigen: Unified image generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 13294–13304, 2025. 1, 2

[63] Hanshu Yan, Xingchao Liu, Jiachun Pan, Jun Hao Liew, Qiang Liu, and Jiashi Feng. Perflow: Piecewise rectified flow as universal plug-and-play accelerator. *Advances in Neural Information Processing Systems*, 37:78630–78652, 2024. 2

[64] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025. 5, 6

[65] Zixuan Ye, Xuanhua He, Quande Liu, Qiulin Wang, Xintao Wang, Pengfei Wan, Di Zhang, Kun Gai, Qifeng Chen, and Wenhan Luo. Unic: Unified in-context video editing. *arXiv preprint arXiv:2506.04216*, 2025. 1, 2

[66] Tianwei Yin, Michaël Gharbi, Taesung Park, Richard Zhang, Eli Shechtman, Fredo Durand, and William T Freeman. Improved distribution matching distillation for fast image synthesis. In *NeurIPS*, 2024. 2, 3, 5

[67] Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Fredo Durand, William T Freeman, and Taesung Park. One-step diffusion with distribution matching distillation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6613–6623, 2024. 2, 3

[68] Bo Zhang, Mingming He, Jing Liao, Pedro V Sander, Lu Yuan, Amine Bermak, and Dong Chen. Deep exemplar-based video colorization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8052–8061, 2019. 2

[69] Kai Zhang, Lingbo Mo, Wenhu Chen, Huan Sun, and Yu Su. Magicbrush: A manually annotated dataset for instruction-guided image editing. *Advances in Neural Information Processing Systems*, 36:31428–31449, 2023. 2

[70] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3836–3847, 2023. 2

[71] Tianyuan Zhang, Sai Bi, Yicong Hong, Kai Zhang, Fujun Luan, Songlin Yang, Kalyan Sunkavalli, William T Freeman, and Hao Tan. Test-time training done right. *arXiv preprint arXiv:2505.23884*, 2025. 5

[72] Yabo Zhang, Yuxiang Wei, Dongsheng Jiang, Xiaopeng Zhang, Wangmeng Zuo, and Qi Tian. Controlvideo: Training-free controllable text-to-video generation. *arXiv preprint arXiv:2305.13077*, 2023. 6, 7

VDOT: Efficient Unified Video Creation via Optimal Transport Distillation

Supplementary Material

Table 4. **Overview of the training dataset composition.**

Task		`Input`	#Examples
V2V	*Depth*	`txt+video`	$\mathcal{O}(6)$K
	Pose	`txt+video`	$\mathcal{O}(12)$K
	Grey	`txt+video`	$\mathcal{O}(6)$K
	Scribble	`txt+video`	$\mathcal{O}(6)$K
	Flow	`txt+video`	$\mathcal{O}(6)$K
MV2V	*Outpaint*	`txt+video+mask`	$\mathcal{O}(12)$K
	Extension	`txt+video+mask`	$\mathcal{O}(12)$K
R2V	*Reference*	`txt+image`	$\mathcal{O}(15)$K
Composite Tasks		`txt+video+mask+image`	$\mathcal{O}(20)$K

8. Implementation Details

8.1. Dataset Composition

Table 4 details the composition of our training dataset across different tasks. For the V2V tasks, we curate datasets for five distinct condition signals: *depth*, *pose*, *greyscale*, *scribble*, and *flow*. Each is formulated as a conditional video-to-video generation task containing approximately 6-12K samples. The MV2V tasks extend this framework by incorporating mask inputs to support *outpainting* and *temporal extension*, with roughly 12K samples per task. For the R2V task, we include 15K examples where generation is conditioned jointly on text prompts and reference images. Finally, for the composite tasks, we implement dual control signals: (1) a first-frame or reference image, and (2) one of the five V2V modalities. This results in 10 distinct subtasks, each with 2K samples, totaling 20K composite training examples.

8.2. Diffusion denoising objective

When updating the fake model $\boldsymbol{F}_\phi$, we adopt the standard diffusion denoising loss [21, 51]:

$$\mathcal{L}_{\text{Denoising}}(\phi) = \|\boldsymbol{F}_\phi(\boldsymbol{x}_t^{\text{fake}}, t) - \hat{\boldsymbol{x}}_0\|_2^2, \tag{13}$$

where $\boldsymbol{x}_t^{\text{fake}}$ represents the noisy sample obtained by adding noise to the generator output $\hat{\boldsymbol{x}}_0$.

8.3. Sinkhorn algorithm

As illustrated in Algorithm 2, we solve the entropic optimal transport problem in Equation (7) using the log-domain Sinkhorn algorithm [42]. Unlike the standard Sinkhorn iterations, the log-domain formulation stabilizes the dual updates by performing all computations in logarithmic space, thereby preventing underflow of the Gibbs kernel and ensuring numerically reliable convergence even with small regularization ϵ.

Algorithm 2: Log-domain Sinkhorn Algorithm for Entropic Optimal Transport (Equation (7))

Input : Cost matrix $\boldsymbol{D} \in \mathbb{R}^{I\times J}$;
Marginals $\boldsymbol{u} \in \mathbb{R}_+^I$, $\boldsymbol{\mu} \in \mathbb{R}_+^J$;
Regularization parameter $\epsilon > 0$;
Tolerance `tol` and iterations `max_iter`.
Output: Optimal transport plan $\boldsymbol{T}^* \in \mathbb{R}_+^{I\times J}$.

```
// Precompute log-kernel
log K ← −D/ε
// Initialize dual variables
f^(0) ← 0_I,   g^(0) ← 0_J
for t = 1 to max_iter do
    f_old ← f^(t−1),   g_old ← g^(t−1)
    // Update f
    for i = 1 to I do
        s ← log Σ_{j=1}^{J} exp(log K_ij + g_j^(t−1))
        f_i^(t) ← log u_i − s
    end
    // Update g
    for j = 1 to J do
        s ← log Σ_{i=1}^{I} exp(log K_ij + f_i^(t))
        g_j^(t) ← log μ_j − s
    end
    // Stopping criterion
    if ‖f^(t) − f_old‖_1 + ‖g^(t) − g_old‖_1 < tol then
        break
    end
end
// Recover optimal transport plan
for i = 1 to I do
    for j = 1 to J do
        log T*_ij ← f_i^(t) + log K_ij + g_j^(t)
        T*_ij ← exp(log T*_ij)
    end
end
return T*
```

9. Quantitative and Qualitative Analysis

Composite tasks. Table 5 presents the quantitative evaluation of composite tasks on the `UVCBench`. Across all settings, our proposed `VDOT` consistently matches or outperforms VACE in terms of both video quality and temporal consistency, while requiring only 4 NFEs. Notably, `VDOT` achieves the highest normalized average score in 6 out of 10

Table 5. **Quantitative evaluations of composite tasks on UVCBench**. We compare the automated score metrics of our VDOT with VACE on the dimensions of video quality and video consistency.

Condition1	Condition2	Method	Base Model	#NFE↓	Video Quality & Video Consistency						
					Aesthetic Quality	Background Consistency	Dynamic Degree	Imaging Quality	Motion Smoothness	Subject Consistency	Normalized Average
FirstFrame	Depth	VACE [26]	LTX-0.9B	80	64.78%	**96.90%**	40.00%	66.19%	98.96%	94.49%	76.89%
		VACE [26]	Wan-14B	100	**68.28%**	96.74%	40.00%	68.43%	98.92%	94.38%	77.79%
		VDOT (Ours)	Wan-14B	4	67.34%	96.67%	**45.00%**	**70.09%**	**98.97%**	**94.64%**	**78.78%**
	Pose	VACE [26]	LTX-0.9B	80	64.75%	**96.29%**	15.00%	63.90%	**99.13%**	94.63%	72.28%
		VACE [26]	Wan-14B	100	**66.66%**	95.91%	**40.00%**	66.99%	98.94%	**94.70%**	**77.19%**
		VDOT (Ours)	Wan-14B	4	66.26%	96.20%	30.00%	**68.63%**	99.08%	94.22%	75.73%
	Flow	VACE [26]	LTX-0.9B	80	54.38%	**95.19%**	70.00%	52.16%	**98.21%**	91.05%	76.83%
		VACE [26]	Wan-14B	100	**59.80%**	94.66%	**85.00%**	53.41%	97.80%	**91.62%**	80.38%
		VDOT (Ours)	Wan-14B	4	59.27%	95.01%	**85.00%**	**55.75%**	97.92%	91.58%	**80.75%**
	Scribble	VACE [26]	LTX-0.9B	80	55.16%	96.57%	35.00%	66.52%	**99.05%**	94.36%	74.44%
		VACE [26]	Wan-14B	100	**58.26%**	**96.78%**	**50.00%**	66.91%	98.91%	94.89%	77.63%
		VDOT (Ours)	Wan-14B	4	57.54%	96.64%	**50.00%**	**68.23%**	98.97%	**94.94%**	**77.72%**
	Grey	VACE [26]	LTX-0.9B	80	66.45%	97.97%	10.00%	63.60%	**99.34%**	97.85%	72.54%
		VACE [26]	Wan-14B	100	68.87%	**98.25%**	15.00%	**66.11%**	99.30%	98.02%	74.25%
		VDOT (Ours)	Wan-14B	4	**69.08%**	98.01%	**20.00%**	65.92%	99.30%	**98.08%**	**75.07%**
Reference	Depth	VACE [26]	LTX-0.9B	80	52.92%	87.63%	60.00%	63.22%	**98.82%**	77.83%	73.40%
		VACE [26]	Wan-14B	100	**70.07%**	96.53%	**65.00%**	67.61%	98.39%	94.97%	**82.09%**
		VDOT (Ours)	Wan-14B	4	69.49%	**96.64%**	**65.00%**	**67.85%**	98.44%	**95.05%**	82.08%
	Pose	VACE [26]	LTX-0.9B	80	49.89%	85.24%	65.00%	52.74%	**98.43%**	71.46%	70.46%
		VACE [26]	Wan-14B	100	**71.05%**	95.38%	**75.00%**	67.72%	97.77%	93.40%	83.38%
		VDOT (Ours)	Wan-14B	4	70.81%	**95.97%**	**75.00%**	**67.81%**	98.19%	**94.24%**	**83.67%**
	Flow	VACE [26]	LTX-0.9B	80	52.33%	83.37%	85.00%	58.50%	**98.36%**	66.36%	73.98%
		VACE [26]	Wan-14B	100	**65.92%**	96.30%	**90.00%**	**66.74%**	97.71%	93.01%	**84.94%**
		VDOT (Ours)	Wan-14B	4	65.31%	**96.67%**	80.00%	66.54%	98.00%	**93.78%**	83.38%
	Scribble	VACE [26]	LTX-0.9B	80	51.64%	89.42%	50.00%	56.96%	**98.70%**	73.25%	69.99%
		VACE [26]	Wan-14B	100	**65.57%**	96.43%	**75.00%**	**66.20%**	98.19%	93.18%	**82.42%**
		VDOT (Ours)	Wan-14B	4	64.10%	**96.78%**	**75.00%**	65.35%	98.30%	**93.41%**	82.16%
	Grey	VACE [26]	LTX-0.9B	80	58.04%	95.46%	25.00%	60.91%	**99.06%**	95.68%	72.35%
		VACE [26]	Wan-14B	100	**72.00%**	97.60%	**30.00%**	64.43%	98.75%	97.45%	76.70%
		VDOT (Ours)	Wan-14B	4	71.98%	**97.66%**	**30.00%**	**64.52%**	98.81%	**97.48%**	**76.74%**

settings, showing substantial gains in *Imaging Quality* and *Subject Consistency*. In conditions where structure guidance is crucial (e.g., *Depth*, *Scribble*, *Grey*), VDOT delivers robust improvements over the VACE baselines, demonstrating superior temporal stability and perceptual fidelity. Figure 7 provides a qualitative comparison between VDOT and VACE. We visualize frames at uniformly spaced intervals (indices 1, 21, 41, 61, and 81). In the *firstframe&depth* setting, VACE suffers from inconsistent background coloration (e.g., in the sky), whereas VDOT maintains strong spatiotemporal consistency. In the *Reference&pose* setting, also known as the animate anyone setting, both VACE and VDOT adhere well to the input pose skeleton; however, minor discrepancies appear in the generated effects. Specifically, our generated effects align more closely with the motion, adhering to developmental patterns.

Results of VACE-Benchmark. Quantitative evaluations on the VACE-Benchmark demonstrate that our proposed VDOT achieves a superior balance between generation quality and computational efficiency compared to the VACE baselines. As shown in Table 6, our method consistently attains comparable or higher scores across diverse tasks, such as *pose*, *depth*, and *flow*, particularly in dimensions of *Imaging Quality* and *Subject Consistency*. The substantial reduction in inference steps, combined with top-tier performance across the majority of tasks, highlights our method's ability to maintain high video fidelity and temporal stability with minimal inference latency.

Denoising steps. Our training adheres to a four-step paradigm based on Self-Forcing [23]. Evaluations on UVCBench reveal the impact of step counts on generation quality. Table 7 shows that increasing denoising steps consistently boosts performance, particularly for *pose* and *extension* tasks. This can be attributed to the coarse-to-fine nature of the generation process: low-frequency structure is determined early, while fine-grained details require

Table 6. **Quantitative evaluations on VACE-Benchmark**. We compare the automated score metrics of our VDOT with VACE on the dimensions of video quality and video consistency.

Type	Method	Base Model	#NFE↓	Video Quality & Video Consistency						
				Aesthetic Quality	Background Consistency	Dynamic Degree	Imaging Quality	Motion Smoothness	Subject Consistency	Normalized Average
Depth	VACE [26]	LTX-0.9B	80	55.73%	96.06%	60.00%	67.53%	**98.92%**	93.90%	78.69%
	VACE [26]	Wan-14B	100	62.34%	**96.10%**	**65.00%**	68.85%	98.57%	94.25%	80.85%
	VDOT (Ours)	Wan-14B	4	**62.46%**	95.81%	**65.00%**	**69.69%**	98.39%	**94.53%**	**80.98%**
Pose	VACE [26]	LTX-0.9B	80	59.97%	94.80%	**85.00%**	66.85%	**98.93%**	**94.91%**	83.41%
	VACE [26]	Wan-14B	100	66.69%	**95.32%**	75.00%	68.48%	98.56%	94.19%	83.04%
	VDOT (Ours)	Wan-14B	4	**66.71%**	94.99%	80.00%	**70.40%**	98.45%	94.26%	**84.13%**
Flow	VACE [26]	LTX-0.9B	80	55.36%	95.99%	**75.00%**	64.30%	**99.07%**	94.14%	80.64%
	VACE [26]	Wan-14B	100	**60.44%**	**96.09%**	**75.00%**	66.99%	98.48%	**94.71%**	81.95%
	VDOT (Ours)	Wan-14B	4	60.24%	95.65%	**75.00%**	**70.02%**	98.42%	94.26%	**82.27%**
Scribble	VACE [26]	LTX-0.9B	80	53.81%	96.57%	45.00%	66.05%	**99.21%**	95.23%	75.98%
	VACE [26]	Wan-14B	100	**59.44%**	96.49%	**50.00%**	67.01%	98.46%	95.44%	77.80%
	VDOT (Ours)	Wan-14B	4	57.99%	**96.75%**	**50.00%**	**69.16%**	98.60%	**95.79%**	**78.05%**
Grey	VACE [26]	LTX-0.9B	80	58.63%	**95.65%**	55.00%	59.93%	**98.89%**	92.39%	76.75%
	VACE [26]	Wan-14B	100	**62.87%**	95.46%	60.00%	66.35%	98.40%	92.91%	79.33%
	VDOT (Ours)	Wan-14B	4	62.25%	95.64%	**65.00%**	**66.74%**	98.40%	**92.88%**	**80.15%**
Extension	VACE [26]	LTX-0.9B	80	57.82%	**96.06%**	35.00%	68.15%	**99.34%**	**94.06%**	75.07%
	VACE [26]	Wan-14B	100	**61.34%**	95.05%	**60.00%**	69.55%	98.75%	92.83%	**79.59%**
	VDOT (Ours)	Wan-14B	4	61.25%	95.94%	45.00%	**70.01%**	99.04%	93.86%	77.52%
Outpaint	VACE [26]	LTX-0.9B	80	56.98%	96.47%	45.00%	**69.39%**	**99.17%**	**95.50%**	77.08%
	VACE [26]	Wan-14B	100	58.53%	96.83%	40.00%	68.55%	99.03%	95.34%	76.38%
	VDOT (Ours)	Wan-14B	4	**58.86%**	**96.84%**	**50.00%**	69.18%	99.00%	95.48%	**78.23%**
R2V	VACE [26]	LTX-0.9B	80	62.57%	97.85%	17.50%	**72.32%**	**99.47%**	97.91%	74.60%
	VACE [26]	Wan-14B	100	**65.99%**	**98.77%**	15.00%	71.19%	99.38%	**98.67%**	74.83%
	VDOT (Ours)	Wan-14B	4	65.47%	98.58%	**20.00%**	71.94%	99.32%	98.48%	**75.63%**
T2V	VACE [26]	LTX-0.9B	80	58.08%	**97.94%**	40.00%	**71.16%**	**99.06%**	**97.71%**	77.33%
	VACE [26]	Wan-14B	100	63.49%	97.63%	**45.00%**	69.49%	98.75%	96.96%	**78.55%**
	VDOT (Ours)	Wan-14B	4	**64.37%**	97.75%	35.00%	70.65%	99.02%	97.30%	77.35%

Table 7. **Quantitative comparison of denosing steps of VDOT on UVCBench**.

Denoising Steps	Depth	Pose	Flow	Scribble	Grey	Outpaint	Extension	R2V
2 steps	77.82%	79.65%	78.08%	76.82%	71.99%	75.86%	74.83%	79.18%
3 steps	78.42%	80.21%	79.09%	76.95%	72.05%	75.70%	78.51%	80.80%
4 steps	78.50%	80.54%	80.18%	76.77%	72.13%	76.54%	80.53%	81.32%

more steps to resolve. Thus, tasks providing strong structural constraints (e.g., *scribble*, *grey*) saturate quickly, with the 2-step model performing similarly to the 4-step model. In contrast, under-constrained tasks like *pose* and *extension* benefit substantially from additional steps. Visualizations in Figure 6 validate this observation; for instance, in the pose-to-video task, the 4-step model generates superior fine-grained details (e.g., hair, hands, and background elements) compared to the 2-step output.

Generation resolution. While our model is capable of inference at any resolution, it was trained specifically on 832×480 videos. Table 8 quantifies the impact of increasing resolution to 1280×720, showing a marked decline in performance for most tasks. This degradation arises from a domain gap: the model lacks priors for high-resolution details. Forced to extrapolate low-resolution features to a larger pixel space, the model generates outputs that suffer from blurred details and noise artifacts.

Generalizability to untrained tasks. Beyond the 18 supervised tasks listed in Table 4, our model demonstrates

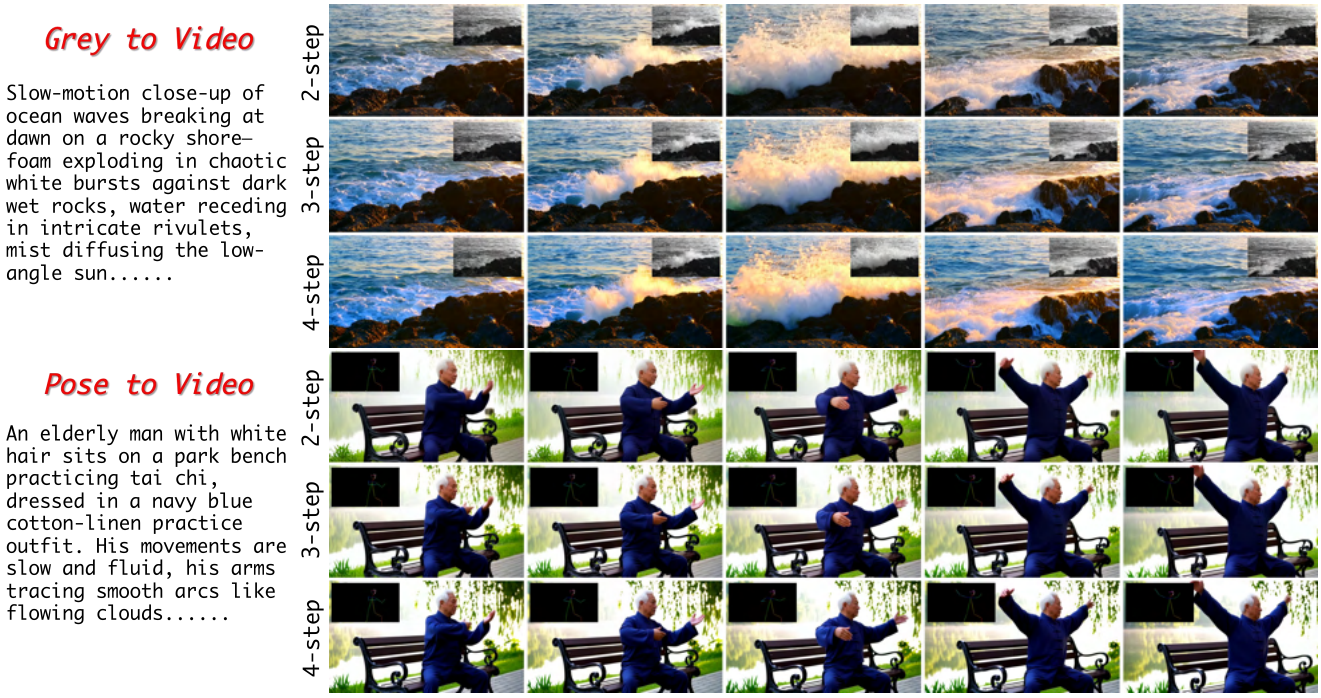

Figure 6. **Qualitative comparison of different denoising steps of VDOT.**

Table 8. **Quantitative comparison of generation resolution of VDOT on UVCBench.**

Resolution	Depth	Pose	Flow	Scribble	Grey	Outpaint	Extension	R2V
832×480	78.50%	80.54%	80.18%	76.77%	72.13%	76.54%	80.53%	81.32%
1280×720	78.15%	80.34%	79.44%	76.99%	71.88%	75.81%	77.98%	79.15%

strong potential for few-step generalization. Even without specific training, the model can generate high-quality videos for novel tasks using just 4 denoising steps. Qualitative results in Figure 8 and Figure 9 highlight this versatility, showcasing successful applications in video inpainting, swap anything, character animation, character replacement, and video try-on.

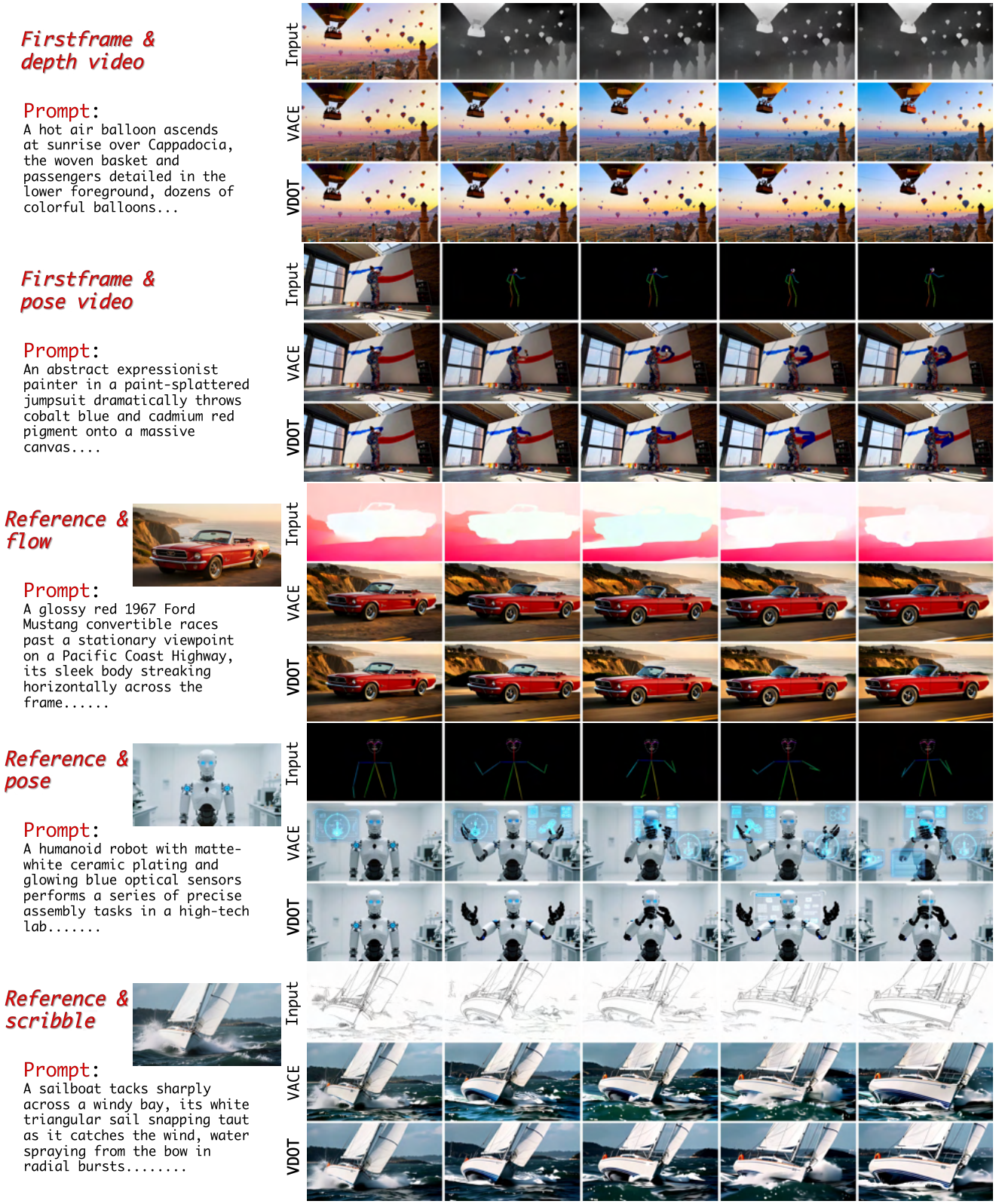

Figure 7. **Qualitative comparison of composite tasks.**

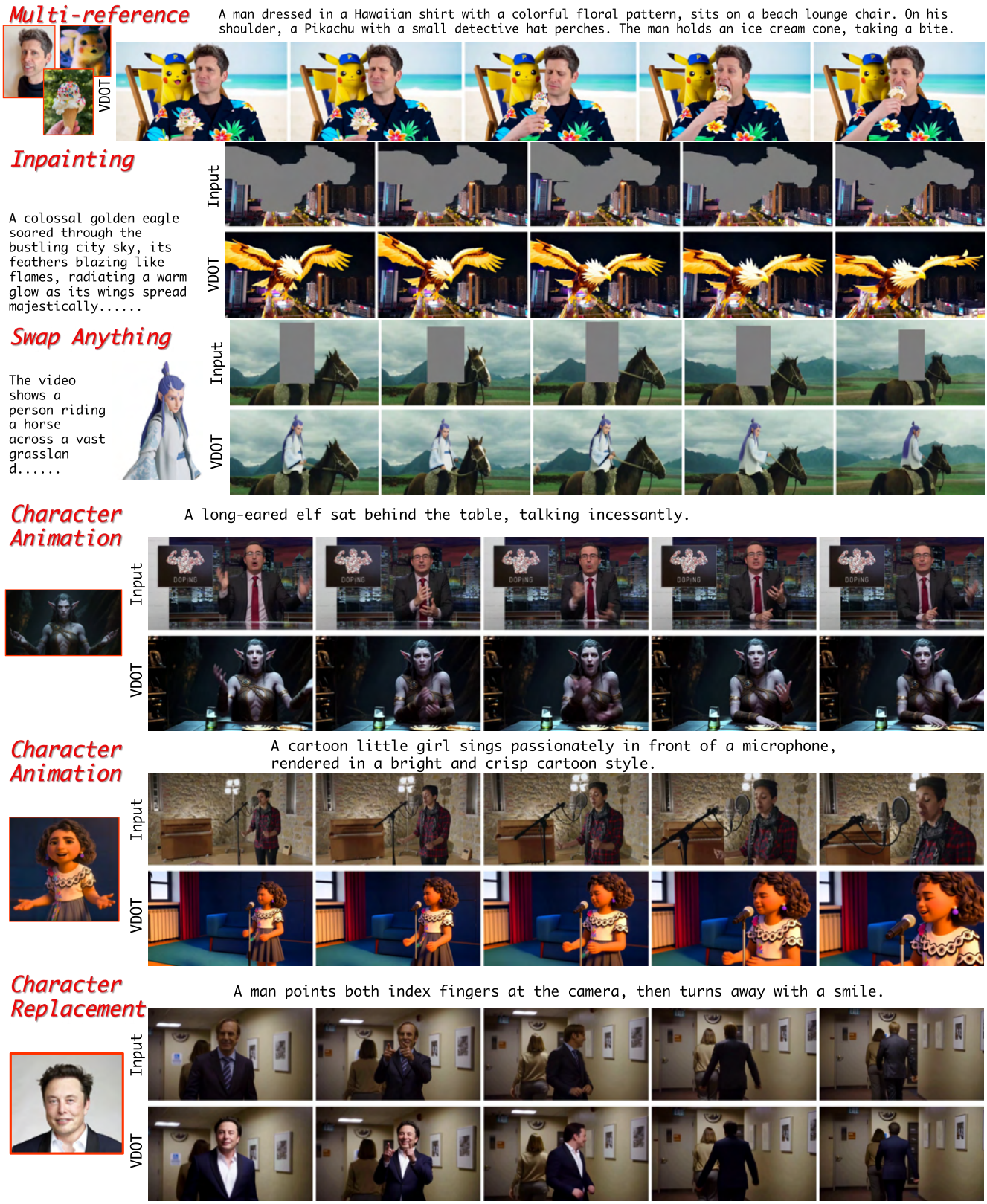

Figure 8. **Visualization of VDOT on untrained tasks.**

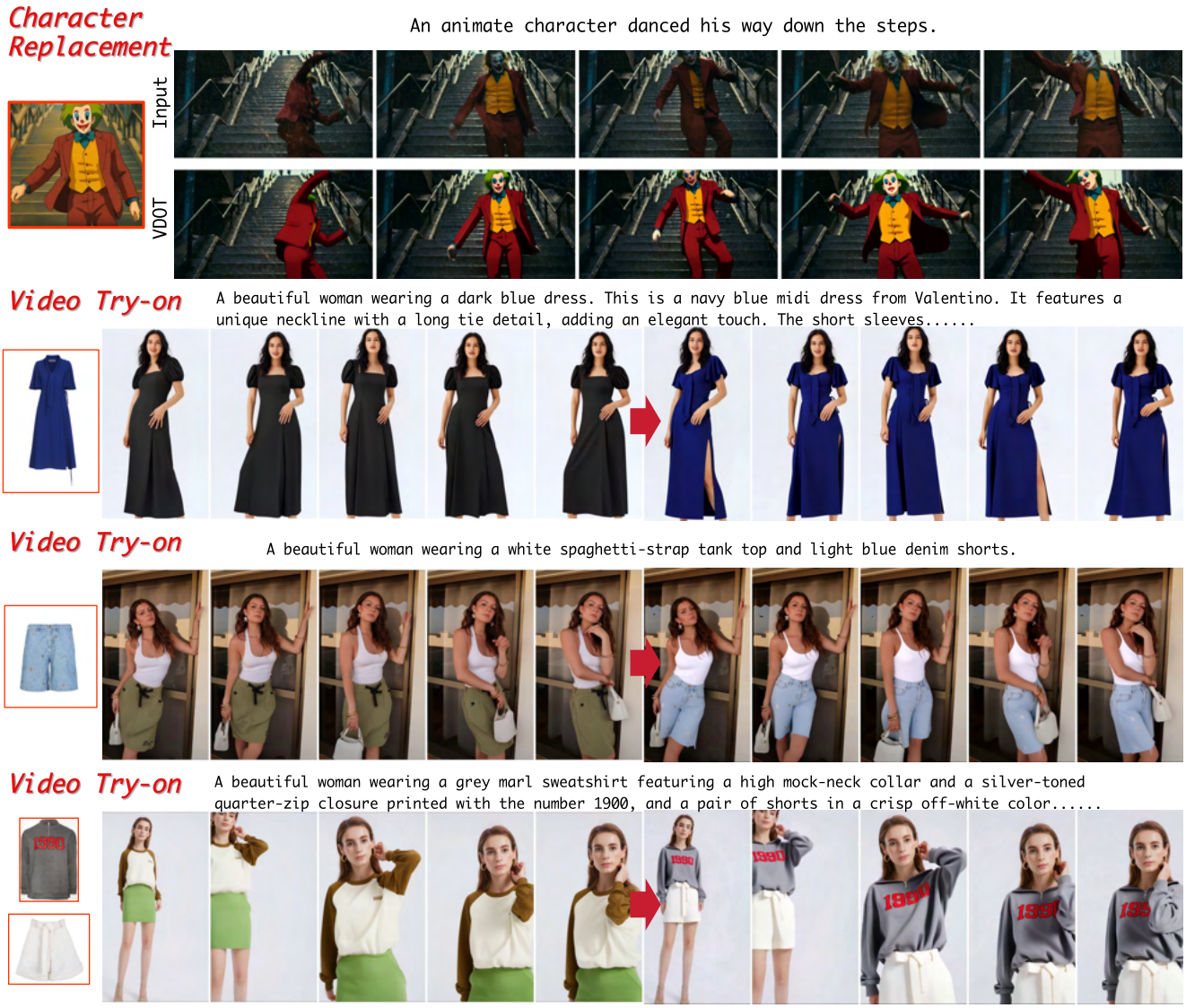

Figure 9. **Visualization of VDOT on untrained tasks.**