DepthCrafter: Generating Consistent Long Depth Sequences for Open-world Videos

Wenbo Hu[1*†]   Xiangjun Gao[2*]   Xiaoyu Li[1*†]   Sijie Zhao[1]
Xiaodong Cun[1]   Yong Zhang[1]   Long Quan[2]   Ying Shan[3,1]

[1]Tencent AI Lab   [2]The Hong Kong University of Science and Technology   [3]ARC Lab, Tencent PCG

https://depthcrafter.github.io

Figure 1. We innovate DepthCrafter, a novel video depth estimation approach, that can generate temporally consistent long depth sequences with fine-grained details for open-world videos, without requiring additional information such as camera poses or optical flow.

Abstract

Estimating video depth in open-world scenarios is challenging due to the diversity of videos in appearance, content motion, camera movement, and length. We present DepthCrafter, an innovative method for generating temporally consistent long depth sequences with intricate details for open-world videos, without requiring any supplementary information such as camera poses or optical flow. The generalization ability to open-world videos is achieved by training the video-to-depth model from a pretrained image-to-video diffusion model, through our meticulously designed three-stage training strategy. Our training approach enables the model to generate depth sequences with variable lengths at one time, up to 110 frames, and harvest both precise depth details and rich content diversity from realistic and synthetic datasets. We also propose an inference strategy that can process extremely long videos through segment-wise estimation and seamless stitching. Comprehensive evaluations on multiple datasets reveal that DepthCrafter achieves state-of-the-art performance in open-world video depth estimation under zeroshot settings. Furthermore, DepthCrafter facilitates various downstream applications, including depth-based visual effects and conditional video generation.

1. Introduction

Monocular depth estimation, serving as the bridge linking 2D observations and the 3D world, has been a long-standing fundamental problem in computer vision. It plays a crucial role in a wide range of downstream applications, *e.g.*, mixed reality, AI-generated content, autonomous driving, and robotics [14, 24, 27, 28, 40, 57, 72, 73]. The inherent ambiguity makes it challenging, as the observed information from a single view is insufficient to determine the depth of a scene uniquely.

---

* Joint first authors.   † Corresponding authors.

1

With recent advances in foundation models, we have witnessed significant progress in depth estimation from monocular images [13, 18, 32, 43, 47, 67, 68, 71]. However, all these methods are tailored for static images, without considering the temporal information in videos. Temporal inconsistency, or flickering, would be observed when directly applying them to videos, as shown in Fig. 1. Existing video depth estimation methods [35, 41, 58, 63, 77] typically try to optimize temporally consistent depth sequences from a pre-trained image depth model, with a given or learnable camera poses. Their performance is sensitive to both the proportion of dynamic content and the quality of the camera poses. Yet, videos in the open world are diverse in content, motion, camera movement, and length, making these methods hard to perform well in practice. Moreover, the required camera poses are usually non-trivial to obtain in open-world videos, particularly for long videos and videos with abundant dynamic content.

In this paper, we aim to generate temporally consistent long depth sequences with high-fidelity details for diverse open-world videos, without requiring any additional information. Observing the strong capability of diffusion models in generating various types of videos [3, 4, 6, 8, 9, 23, 64, 65], we propose *DepthCrafter*, to leverage the video diffusion model for video depth estimation, while maintaining the generalization ability to open-world videos. To train our DepthCrafter, a video-to-depth model, from a pre-trained image-to-video diffusion model, we compile paired video-depth datasets in two styles, *i.e.* realistic and synthetic, since the realistic dataset provides rich content diversity and the synthetic dataset offers precise depth details. On the aspect of temporal context, most existing video diffusion models can only produce a fixed and small number of frames at one time, *e.g.*, 25 frames in Stable Video Diffusion (SVD) [3], which, however, is usually too short for open-world video depth estimation to accurately arrange depth distributions throughout the video. To enable variable long temporal context and fuse the respective advantages of the two-styled datasets, we present a three-stage training strategy to progressively train certain layers of the diffusion model on different datasets with variable lengths. By doing so, we can adapt the video diffusion model to generate depth sequences with variable lengths at one time, up to 110 frames, and harvest both the precise depth details and rich content diversity. To further support extremely long videos, we tailor an inference strategy to process the video in overlapped segments and seamlessly stitch them together.

We extensively evaluate our DepthCrafter on diverse datasets, including indoor, outdoor, static, dynamic, realistic, and synthetic videos, under zero-shot settings. Both qualitative and quantitative results demonstrate that our DepthCrafter achieves state-of-the-art performance in open-world video depth estimation, outperforming existing meth-ods by a large margin. Besides, we demonstrate that our DepthCrafter facilitates various downstream applications, including depth-based visual effects and conditional video generation. Our contributions are summarized below:

- We innovate DepthCrafter, a novel method to generate temporally consistent long depth sequences with fine-grained details for open-world videos, outperforming existing approaches by a large margin.
- We present a three-stage training strategy to enable generating depth sequences with a long and variable temporal context, up to 110 frames. It also allows us to harvest both the precise depth details and rich content diversity from synthetic and realistic datasets.
- We design an inference strategy to segment-wisely process videos beyond 110 frames and seamlessly stitch them together, facilitating depth estimation for extremely long videos.

2. Related Work

Image depth estimation. Image depth estimation aims at predicting the depth map from a single image [1, 15, 17, 36, 38, 45, 66]. However, the generalization ability to diverse open-world scenes is hindered by the limited training data. To this end, MiDaS [49] presented the affine-invariant depth representation, enabling mixed training datasets. Depth-Anything (V2) [67, 68] followed this idea and proposed to train the model on both labeled and large-scale unlabeled images, achieving good generalization ability. Marigold [32] and follow-up works [18, 21, 42] leverage the diffusion priors to realize zero-shot transfer to unseen datasets. Besides the affine-invariant depth, another stream of methods tried to estimate the absolute metric depth [2, 5, 25, 47, 71]. All these methods are tailored for static images without considering the temporal consistency, while our work aims to generate temporally consistent long depth sequences for open-world videos.

Video depth estimation. Existing video depth estimation methods could be categorized into two classes: test-time optimization and feed-forward prediction. Test-time optimization methods [11, 35, 41, 77] involve an optimization procedure for each video during inference, which typically requires camera poses or optical flow. Their results are usually consistent, but the requirement of camera poses or optical flow would limit their applicability to open-world videos. Feed-forward prediction methods directly predict depth sequences from videos [39, 58, 60, 61, 63, 69, 70, 74], *e.g.*, DeepV2D [58] combines camera motion estimation with depth estimation, MAMO [69] leverages memory attention, and NVDS [63] introduces a plug-and-play stabilization network. However, due to the limited video depth training data, these methods often fail to address the in-the-wild videos with diverse content. By leveraging the pre-
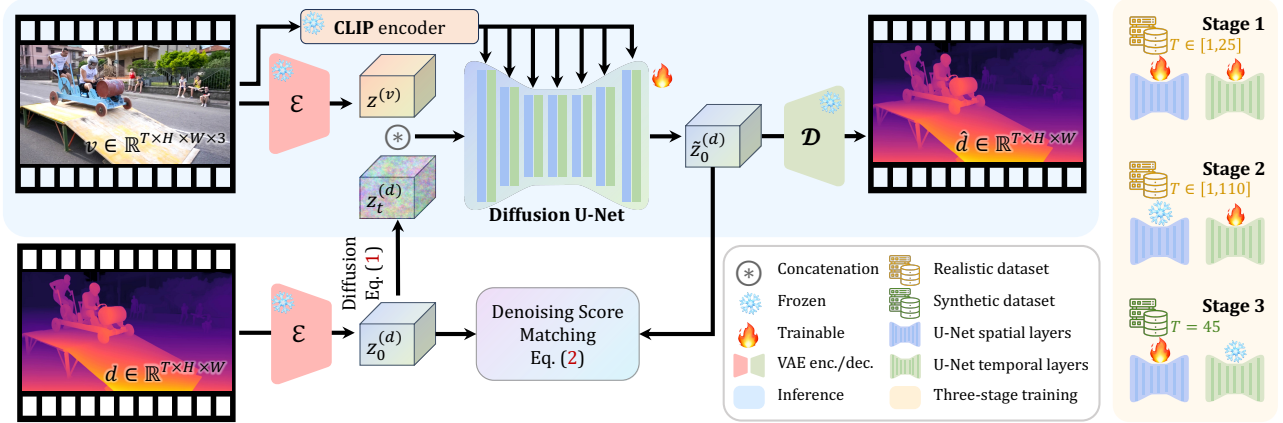
Figure 2. Overview of our *DepthCrafter*. It is a conditional diffusion model that models the distribution $p(\mathbf{d} \,|\, \mathbf{v})$ over the depth sequence $\mathbf{d}$ conditioned on the input video $\mathbf{v}$. We train the model in three stages, where the spatial or temporal layers of the diffusion model are progressively learned on our compiled realistic or synthetic datasets with variable lengths T. During inference, given an open-world video, it can generate temporally consistent long depth sequences with fine-grained details for the entire video from initialized Gaussian noise, without requiring any supplementary information, such as camera poses or optical flow.

trained video diffusion model and our designed three-stage training strategy, our method demonstrates a powerful ability for open-world video depth estimation.

Video diffusion models. Diffusion models [22, 56] have shown remarkable video generation ability [3, 4, 6, 8, 9, 23, 64, 65]. Among these methods, VDM [23] presents the first results on video generation using diffusion models, Sora [6] achieves impressive performance in this area, and SVD [3] provides the popular open-source models for image-to-video generation. Trained on a well-curated video dataset, SVD can generate high-quality videos and is used as the model prior for various video-related tasks. In this paper, we leverage the video diffusion model for high-fidelity consistent video depth estimation, such that the generalization ability to open-world videos can be maintained. Concurrent to our work, ChronoDepth [54] also explores video depth estimation with video diffusion priors. However, ChronoDepth only supports a short temporal context, *i.e.* 10 frames, which is insufficient to accurately arrange depth distributions throughout the video. In contrast, our method not only supports variable-length temporal context at one time, up to 110 frames, but also can estimate depth sequences for extremely long videos.

3. Method

Given an open-world video, $\mathbf{v} \in \mathbb{R}^{T \times H \times W \times 3}$, our goal is to estimate temporally consistent depth sequences, $\mathbf{d} \in \mathbb{R}^{T \times H \times W}$, with fine-grained details. Considering the diversity of open-world videos in appearance, content motion, camera movement, and length, the challenges to achieving our goal are threefold: 1.) a comprehensive understanding of video content for generalization ability; 2.) a long and variable temporal context to arrange the entire depth distributions accurately and keep temporal consistency; and 3.)

the ability to process extremely long videos. As shown in Fig. 2, we tackle these challenges by formulating the video depth estimation as a conditional diffusion generation problem $p(\mathbf{d} \,|\, \mathbf{v})$. We train a video-to-depth model from a pre-trained image-to-video diffusion model through a meticulously designed three-stage training strategy with compiled paired video-depth datasets, and tailor an inference strategy to process extremely long videos through segment-wise estimation and seamless stitching.

3.1. Preliminaries of Video Diffusion Models

Diffusion models [22, 56] learn the data distribution $p(\mathbf{x})$ by a forward diffusion process to gradually noise the data to a target distribution, *e.g.* the Gaussian distribution, and a reverse denoising process to iteratively recover the data from the noise by a learned denoiser. In this paper, our study is conducted based on the stale video diffusion (SVD) [3], which is a famous open-source video diffusion model. SVD adopts the EDM [31] diffusion framework. The diffusion process is achieved by adding i.i.d. σ_t^2-variance Gaussian noise to the data $\mathbf{x}_0 \sim p(\mathbf{x})$:

$$\mathbf{x}_t = \mathbf{x}_0 + \sigma_t^2 \epsilon, \quad \epsilon \sim \mathcal{N}\big(\mathbf{0}, \mathbf{I}\big), \quad (1)$$

where $\mathbf{x}_t \sim p(\mathbf{x}; \sigma_t)$ is the data with noise level σ_t. When σ_t is large enough ($\sigma_{\max}$), the distribution would be indistinguishable from the Gaussian distribution. Based on this fact, the diffusion model starts from a high-variance Gaussian noise $\epsilon \sim \mathcal{N}(\mathbf{0}, \sigma_{\max}^2 \mathbf{I})$ and gradually denoises it towards $\sigma_0 = 0$ to generate the data. The denoiser D_θ is a learnable function that tries to predict the clean data, *i.e.* $\tilde{\mathbf{x}}_0 = D_\theta\big(\mathbf{x}_t; \sigma_t\big)$. Its training objective is the denoising score matching:

$$\mathbb{E}_{\mathbf{x}_t \sim p(\mathbf{x}; \sigma_t), \sigma_t \sim p(\sigma)} \left[\lambda_{\sigma_t} \left\| D_\theta\big(\mathbf{x}_t; \sigma_t; \mathbf{c}\big) - \mathbf{x}_0 \right\|_2^2 \right], \quad (2)$$

where $p(\sigma)$ is the noise level distribution during training, $\mathbf{c}$ denotes the conditioning information, and λ_{σ_t} is the weight for the denoising loss at time t. To promote the learning, EDM adopts the preconditioning strategy [31, 53], to parameterize the denoiser D_θ as:

$$D_\theta(\mathbf{x}_t; \sigma_t; \mathbf{c}) = \\ c_{\text{skip}}(\sigma_t)\mathbf{x}_t + c_{\text{out}}(\sigma_t)F_\theta(c_{\text{in}}\mathbf{x}_t; c_{\text{noise}}(\sigma_t); \mathbf{c}), \quad (3)$$

where F_θ is implemented as a learnable U-Net [52], and c_{in}, c_{out}, c_{skip}, and c_{noise} are preconditioning functions.

3.2. Formulation with Diffusion Models

Latent space transformation. To generate high-resolution depth sequences without sacrificing computational efficiency, we adopt the framework of Latent Diffusion Models (LDMs) [51] that perform in a low-dimensional latent space. The transformation between the latent and data spaces is achieved by a Variational Autoencoder (VAE) [33], which was originally designed for encoding and decoding video frames in SVD [3]. Fortunately, we found it can be directly used for depth sequences with only a negligible reconstruction error, which is similar to the observation in Marigold [32] for image depth estimation. As shown in Fig. 2, the latent space transformation is formulated as:

$$\mathbf{z}^{(\mathbf{x})} = \mathcal{E}(\mathbf{x}), \quad \hat{\mathbf{x}} = \mathcal{D}(\mathbf{z}^{(\mathbf{x})}), \quad (4)$$

where $\mathbf{x}$ is either the video $\mathbf{v}$ or the depth sequence $\mathbf{d}$, $\mathbf{z}^{(\mathbf{x})}$ is the latent representation of the data, $\hat{\mathbf{x}}$ is the reconstructed data, $\mathcal{E}$ and $\mathcal{D}$ are encoder and decoder of the VAE, respectively. For the depth sequence, we replicate it three times to meet the 3-channel input format of the encoder in VAE and average the three channels of the decoder output to obtain the final latent of the depth sequence. Following the practice in image depth estimation [32, 49, 50, 67, 68], we adopt the relative depth, *i.e.* the affine-invariant depth, which is normalized to $[0, 1]$. But differently, our predicted depth sequence shares the same scale and shift across frames, rather than a per-frame normalization, which is crucial for maintaining temporal consistency.

Conditioning on the video. SVD is an image-to-video diffusion model that generates videos conditioned on a single image. The conditional image is fed into the U-Net in two ways, *i.e.*, concatenating its latent to the input latent, and injecting its CLIP [48] embedding to the intermediate features via cross-attention. Yet, our DepthCrafter involves the generation of depth sequences conditioned on video frames in a frame-to-frame fashion. Therefore, we adapt the conditioning mechanism to meet our video-to-depth generation task. As shown in Fig. 2, given the encoded latent of depth sequence $\mathbf{z}^{(\mathbf{d})}$ and video frames $\mathbf{z}^{(\mathbf{v})}$ from Eq. (4), we concatenate the video latent to the input noisy depth latent frame-wisely, rather than only the first frame, to condition

the denoiser for generating the depth sequence. For high-level semantic information, we embed the video frames using CLIP and then inject the embeddings in a frame-to-frame manner to the denoiser via cross-attention. Compared to the original conditioning mechanism, our adapted conditioning provides comprehensive information from the video frames to the denoiser, which guarantees the alignment between the generated depth sequences and the video content.

3.3. Training Strategy

To train our DepthCrafter, we need a large amount of high-quality paired video-depth sequences. Although there are several video depth datasets available, *e.g.*, KITTI [19], Scannet [12], VDW [62], DynamicReplica [30], and MatrixCity [37], they are either lacking high-quality depth annotations or restricted to a specific domain, *e.g.*, driving scenes, indoor scenes, or synthetic scenes.

Dataset construction. To this end, we compiled paired datasets of two styles, *i.e.* realistic and synthetic, where the realistic dataset is large-scale and diverse, and the synthetic dataset is miniature but fine-grained and accurate. The realistic dataset is constructed from a large number of binocular videos with a wide range of scene and motion diversity. We cut the videos according to scene changes, and apply the state-of-the-art video stereo matching method, *e.g.*, BiDAStereo [29], to generate temporally consistent depth sequences. Finally, we obtained $\sim$200K paired video-depth sequences with the length of $50-200$ frames. The synthetic dataset is a combination of the DynamicReplica [30] and MatrixCity [37] datasets, which contains $\sim$3K fine-grained depth annotations with a length of 150 frames.

Challenges of variable long temporal context. Different from image depth estimation which can determine the distribution of relative depth from a single frame, the video depth estimation requires a long temporal context to arrange the depth distributions accurately for the entire video and keep the temporal consistency. Besides, the model should support variable-length estimation as the length of open-world videos may vary significantly. However, existing open-source video diffusion models can only generate a fixed small number of frames at a time, *e.g.*, 25 frames in SVD [3]. It is non-trivial to adapt the pre-trained model to meet this requirement, as directly fine-tuning it with long sequences is memory-consuming, for example, modern GPUs with 40GB memory can only support the training of a 25-frame sequence in SVD.

Three-stage training. Considering both the two-style paired datasets and the long temporal context requirement, we design a three-stage training strategy to harvest the variety of video content, the precise depth details, as well as the support for long and variable sequences. As shown in Fig. 2, we train our DepthCrafter from the pre-trained SVD in three stages. We first train it on our large realistic
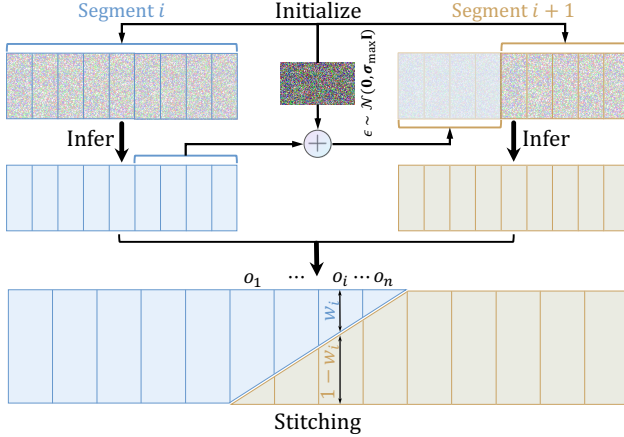
Figure 3. Inference for extremely long videos. We divide the video into overlapped segments and estimate the depth sequence for each segment with a noise initialization strategy to anchor the scale and shift of depth distributions. These depth segments are then seamlessly stitched together with a latent interpolation strategy to form the entire depth sequence. The overlapped frames and their interpolation weights are denoted as o_i, w_i and $1 - w_i$, respectively.

dataset to adapt the model to the video-to-depth generation task. The sequence length in this stage is randomly sampled from $[1, 25]$ frames, such that the model can learn to generate depth sequences with variable lengths. In the second stage, we only fine-tune the temporal layers of the model, still on our large realistic dataset, but with the sequence length randomly sampled from $[1, 110]$ frames. The reason why we only fine-tune the temporal layers is that the temporal layers are more sensitive to the sequence length while the spatial layers are already adapted to the video-to-depth generation task in the first stage, and doing so significantly reduces memory consumption compared to fine-tuning the full model. The long temporal context in this stage enables the model to precisely arrange the entire depth distributions for long and variable sequences. In the third stage, we fine-tune the spatial layers of the model on our small synthetic dataset, with a fixed sequence length of 45 frames since the model has already learned to generate depth sequences with variable lengths in the first two stages and tuning the spatial layers would not affect the temporal context. As the depth annotations in the synthetic dataset are more accurate and fine-grained, the model can learn more precise depth details in this stage. The three-stage training strategy makes our DepthCrafter capable of generating high-quality depth sequences for open-world videos with variable lengths.

3.4. Inference for Extremely Long Videos

Although the model can estimate depth sequences up to the length of 110 frames after training, it is still far from long enough for open-world videos, which can even contain hundreds or thousands of frames. To this end, we design an inference strategy to infer extremely long depth sequences in

a segment-wise manner and seamlessly stitch them together to form the entire depth sequence. As shown in Fig. 3, we first divide the video into overlapped segments, whose lengths are up to 110 frames. Then we estimate the depth sequences for each segment. Rather than purely initializing the input latent with Gaussian noise $\epsilon \sim \mathcal{N}(\mathbf{0}, \sigma_{\max}^2 \mathbf{I})$, we initialize the latent of the overlapped frames by adding noise to the denoised latent from the previous segment, to anchor the scale and shift of the depth distributions. Finally, to further ensure the temporal smoothness across segments, we craft a mortise-and-tenon style latent interpolation strategy to stitch consecutive segments together, inspired by [75]. Specifically, we interpolate the latent of the overlapped frames o_i from the two segments with the interpolation weights w_i and $1 - w_i$, respectively, where w_i is linearly decreased from 1 to 0. The final estimated depth sequence is obtained by decoding the stitched latent segments with the decoder $\mathcal{D}$ in the VAE. With the training and inference strategies, our DepthCrafter can generate temporally consistent long depth sequences for open-world videos.

4. Experiments

4.1. Implementation

We implemented our DepthCrafter based on SVD [3], using the diffusers [59] library. We train our model at the resolution of 320×640 for efficiency, but we can estimate depth sequences at any resolution, *e.g.*, 576×1024, during inference. We use the Adam optimizer [34] with a learning rate of 1×10^{-5} and a batch size of 8. The number of iterations in the three stages of training is $80K$, $40K$, and $10K$, respectively. We employed eight NVIDIA A100 GPUs for training, with a total training time of about five days. Except for specific explanations, the number of denoising steps is set to **5** during inference. Please refer to the supplementary material for more implementation details.

4.2. Evaluation

Evaluation datasets. We evaluate our model on four video datasets, a single-image dataset, as well as the DAVIS dataset [46] and in-the-wild videos for qualitative results. The evaluations are conducted under the *zero-shot* setting, where the testing datasets cover a variety of scenes, including synthetic and realistic scenes, indoor and outdoor scenes, and static and dynamic scenes, to evaluate the generalization ability of our model across various open-world scenarios. Sintel [7] is a synthetic dataset with precise depth labels, featuring dynamic scenes with diverse content and camera motion. It contains 23 sequences with a length of around 50 frames. ScanNet v2 [12] is an indoor dataset with depth maps obtained from a Kinect sensor. We extracted 90 frames at a rate of 15 frames per second from each sequence of the test set, which includes 100 RGB-D video se-

Figure 4. Qualitative comparison for open-world video depth estimation. We compare with the representative single-image depth estimation method Depth-Anything-V2 [68] and the video depth estimation method NVDS [63]. For better visualizing the temporal quality, we show the temporal profiles of each result in green boxes, by slicing the depth values along the time axis at the green line positions.

quences of various scenes. Since ScanNet v2 contains only static indoor scenes, we further introduced 5 dynamic indoor RGB-D videos with a length of 110 frames each from the Bonn [44] dataset to better evaluate the performance of our model on dynamic scenes. KITTI [19] is a street-scene outdoor dataset for autonomous driving, with sparse metric depths captured by a LiDAR sensor. We adopted the validation set, which includes 13 scenes, and extracted 13 videos from it with a length of 110 frames each. Besides, we also evaluated our model for single-image depth estimation on the NYU-v2 [55] test set, which contains 654 images.

Evaluation metrics. Following practice in affine-invariant depth estimation [14, 32, 67, 68], we align the estimated depth maps with the ground truth using a scale and shift, and calculate two metrics: AbsRel ↓ (absolute relative error: $|\hat{\mathbf{d}} - \mathbf{d}|/\mathbf{d}$) and δ_1 ↑ (percentage of $\max(\mathbf{d}/\hat{\mathbf{d}}, \hat{\mathbf{d}}/\mathbf{d}) <$ 1.25). Different from previous methods that optimize the scale and shift individually for each frame, we optimize *a*

shared scale and shift across the entire video, which is more challenging, but necessary, for video depth estimation to ensure temporal consistency.

Quantitative results. We compare our DepthCrafter with the representative methods for both single-image and video depth estimation, *i.e.* Marigold [32], Depth-Anything [67], Depth-Anything-V2 [68], NVDS [63], and ChronoDepth [54]. For Depth-Anything (V2), we use the large model variant for its best performance, while for Marigold we adopt the LCM version with an ensemble size of 5. As shown in Tab. 1, our DepthCrafter achieves state-of-the-art performance in all four video datasets, thanks to the powerful open-world video undeederstanding capability of the video diffusion models and the three-stage training strategy that leverages both realistic and synthetic datasets. For Sintel and KITTI, characterized by significant camera motion and fast-moving objects, our DepthCrafter outperforms the strong Depth-Anything (V2) model tremen-

Table 1. Zero-shot depth estimation results. We compare with the best single-image depth estimation model, Marigold [32], and Depth Anything (V2) [67, 68], as well as the representative video depth estimation models, NVDS [63] and ChronoDepth [54]. **Best** and second best results are highlighted.

Method	Sintel (∼50 frames)		Scannet (90 frames)		KITTI (110 frames)		Bonn (110 frames)		NYU-v2 (1 frame)	
	AbsRel ↓	δ_1 ↑	AbsRel ↓	δ_1 ↑	AbsRel ↓	δ_1 ↑	AbsRel ↓	δ_1 ↑	AbsRel ↓	δ_1 ↑
NVDS [63]	0.408	0.483	0.187	0.677	0.253	0.588	0.167	0.766	0.151	0.780
ChronoDepth [54]	0.587	0.486	0.159	0.783	0.167	0.759	0.100	0.911	0.073	0.941
Marigold [32]	0.532	0.515	0.166	0.769	0.149	0.796	0.091	0.931	0.070	0.946
Depth-Anything [67]	0.325	0.564	0.130	0.838	0.142	0.803	0.078	0.939	**0.042**	**0.981**
Depth-Anything-V2 [68]	0.367	0.554	0.135	0.822	0.140	0.804	0.106	0.921	0.043	0.978
DepthCrafter (Ours)	**0.270**	**0.697**	**0.123**	**0.856**	**0.104**	**0.896**	**0.071**	**0.972**	0.072	0.948

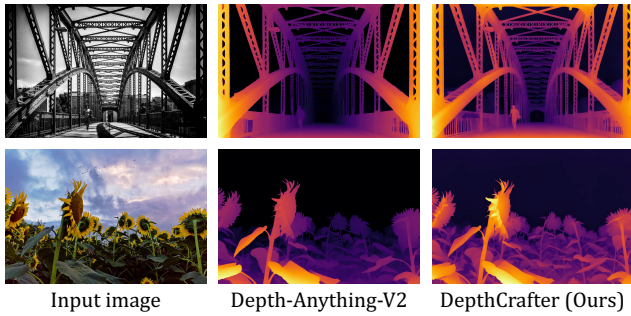Input image   Depth-Anything-V2   DepthCrafter (Ours)

Figure 5. Single-image depth estimation. We compare with the representative single-image depth estimation method Depth-Anything-V2 [68].

dously in terms of both the AbsRel and δ_1 metrics, *e.g.* $25.7\% = (0.140−0.104)/0.140$ improvement in AbsRel on KITTI. For indoor datasets like Scannet and Bonn, featuring minimal camera motion and roughly the same room scales, Depth-Anything has exhibited good performance. Nevertheless, we still have performance enhancements over it, *e.g.* $5.4\% = (0.130 − 0.123)/0.130$ improvement in AbsRel on Scannet. Note that the sequence length of these datasets varies from 50 to 110 frames, and our model can generalize well across different video lengths.

Qualitative results. To further demonstrate the effectiveness of our model, we present the qualitative results from the DAVIS dataset [46], Sora generated videos [6], and open-world videos, including human actions, animals, architectures, cartoons, and games, where the sequence length varies from 90 to 195 frames. As shown in Fig. 4, we show the temporal profiles of the estimated depth sequences in the green line position by slicing the depth values along the time axis, to better visualize the temporal consistency of the estimated depth sequences, following the practice in [26, 63]. We can observe that our DepthCrafter can produce temporally consistent depth sequences with fine-grained details across various open-world videos, while both NVDS and Depth-Anything exhibit zigzag artifacts in the temporal profiles, indicating the flickering artifacts in the estimated depth sequences. These results demonstrate the effectiveness of our DepthCrafter in generating temporally

Table 2. Ablation study on the Sintel dataset. We evaluate the performance of our model at the end of different stages of training.

	stage 1	stage 2	stage 3
AbsRel ↓	0.322	0.316	**0.270**
δ_1 ↑	0.626	0.675	**0.697**

consistent long depth sequences with high-fidelity details for open-world videos.

Single-image depth estimation. Although our model is designed for video depth estimation, it can also perform single-image depth estimation, as our DepthCrafter can estimate video depth of any length. As shown in Tab. 1, our DepthCrafter achieves competitive performance in single-image depth estimation on the NYU-v2 dataset. Since the depth labels in the NYU-v2 dataset are sparse and noisy, we also provide the qualitative results in Fig. 5 to demonstrate the effectiveness of our model in estimating depth maps from static images. We can observe that our DepthCrafter can even produce more detailed depth maps than Depth-Anything-V2, which is the existing state-of-the-art single-image depth estimation model. These results demonstrate the ability of our DepthCrafter for processing both video and single-image depth estimation tasks.

4.3. Ablation Studies

Effectiveness of the three-stage training strategy. We first ablate the effectiveness of the three-stage training strategy by evaluating the performance of our model at the end of each stage on the Sintel dataset [7], since it contains precise depth annotations on dynamic scenes. From Tab. 2, we can observe that the performance of our model consistently improves as the number of training stages increases, indicating the effectiveness of the three-stage training strategy. More ablation results on other datasets are available in the supplementary material.

Effectiveness of the inference strategy. To ablate the effectiveness of our inference strategy components, we consider these variants: **baseline**, which independently infers each segment and directly averages the overlapped frames; **+ initialization**, which contains the same initialization of

Figure 6. Ablation studies on the effectiveness of the inference strategy. We profile the estimated depth sequences of different variants on the green line position. The yellow and blue arrows point to the static and dynamic regions, respectively.

overlapped latents as our method, but without the stitching process; **+ initialization & stitching**, which is our full method. We visually compare the temporal profiles of the estimated depth sequences of these variants in Fig. 6. We can observe the overlapped jaggies in both the static regions (pointed by the yellow arrow) and the dynamic regions (pointed by the blue arrow) in temporal profiles of the "baseline" method, which indicates the flickering artifacts. The "+ initialization" method can alleviate the flickering artifacts in the static regions, but still has jaggies in the dynamic regions, while our full method can produce smooth depth sequences in both static and dynamic regions.

4.4. Applications

Our DepthCrafter can facilitate various downstream applications, *e.g.*, foreground matting, depth slicing, fog effects, and depth-conditioned video generation, by providing temporally consistent depth sequences with fine-grained details for open-world videos. We show example results of fog effects and depth-conditioned video generation in Fig. 7, while more visual effects results are available in the supplementary material. For the fog effect, we blend the fog map with the input video frames based on the depth values to simulate varying transparency levels in fog. And many recent conditioned video generation models [10, 16, 20, 76] employ depth maps as the structure conditions for video generation or editing. We adopt Control-A-Video [10] and video depth of our method as conditions to generate a video with prompts "*a rider walking through stars, artstation*". The visual effects of these applications rely heavily on the accuracy and consistency of the video depth, which demonstrates the wide applicability of our DepthCrafter in various downstream tasks.

4.5. Limitations

Although our DepthCrafter achieves state-of-the-art performance in open-world video depth estimation, there are still some limitations to be addressed in the future. First, the computation and memory consumption of our model is relatively high, which is mainly due to the large model size and the iterative denoising process in the diffusion model. As shown in Tab. 3, we report the inference speed of our model,

Figure 7. Examples of visual effectiveness that could benefit from our DepthCrafter, including adding fog effects and depth-conditioned video generation. More visual effects results are available on our website.

Table 3. Inference time per frame (ms) of our model, Depth-Anything (V2), and Marigold, with the resolution of 1024×576.

Method	Encoding	Denoising	Decoding	All
Depth-Anything (V2)	N/A	N/A	N/A	180.46
Marigold	256.40	114.53	699.36	1070.29
DepthCrafter (Ours)	51.85	160.93	253.06	465.84

Depth-Anything (V2), and Marigold, with the resolution of 1024×576 on a single NVIDIA A100 GPU. We can see that Depth-Anything (V2) is faster than our model, while our method is much faster than Marigold since it ensembles five inference results. Our model achieves the best quality without ensembling, and its inference speed of 465.84 ms per frame is acceptable for many applications. Second, our method requires around 24GB GPU memory to process a video with the resolution of 1024×576 and a segment length of 110 frames. But note that, with the segment length of 40 frames, the memory consumption can be reduced to 12GB, which is acceptable for most modern GPUs, and we can still infer videos of any length by our inference strategy under this setting. We believe that further engineering efforts, such as model distillation and quantization, can help further promote the practicality of our method.

5. Conclusion

We present DepthCrafter, a novel open-world video depth estimation method that leverages video diffusion models. It can generate temporally consistent depth sequences with fine-grained details for video width diverse content, motion, and camera movement, without requiring any additional information. It also supports videos of variable lengths, ranging from one frame (static image) to extremely long videos. This is achieved through our meticulously designed three-stage training strategy, compiled paired video-depth datasets, and an inference strategy. Extensive evaluations have demonstrated that DepthCrafter achieves state-of-the-art performance in open-world video depth estimation under zero-shot settings. It also facilitates various downstream applications, including depth-based visual effects and conditional video generation.

References

[1] Shubhra Aich, Jean Marie Uwabeza Vianney, Md Amirul Islam, and Mannat Kaur Bingbing Liu. Bidirectional attention network for monocular depth estimation. In *ICRA*, 2021. 2

[2] Shariq Farooq Bhat, Reiner Birkl, Diana Wofk, Peter Wonka, and Matthias Müller. Zoedepth: Zero-shot transfer by combining relative and metric depth. *arXiv preprint arXiv:2302.12288*, 2023. 2

[3] 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. 2, 3, 4, 5

[4] Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. In *CVPR*, 2023. 2, 3

[5] Aleksei Bochkovskii, Amaël Delaunoy, Hugo Germain, Marcel Santos, Yichao Zhou, Stephan R Richter, and Vladlen Koltun. Depth pro: Sharp monocular metric depth in less than a second. *arXiv preprint arXiv:2410.02073*, 2024. 2

[6] Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, Clarence Ng, Ricky Wang, and Aditya Ramesh. Video generation models as world simulators, 2024. 2, 3, 7

[7] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black. A naturalistic open source movie for optical flow evaluation. In *ECCV*, 2012. 5, 7, 1, 3

[8] Haoxin Chen, Menghan Xia, Yingqing He, Yong Zhang, Xiaodong Cun, Shaoshu Yang, Jinbo Xing, Yaofang Liu, Qifeng Chen, Xintao Wang, et al. Videocrafter1: Open diffusion models for high-quality video generation. *arXiv preprint arXiv:2310.19512*, 2023. 2, 3

[9] 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 *CVPR*, 2024. 2, 3

[10] Weifeng Chen, Jie Wu, Pan Xie, Hefeng Wu, Jiashi Li, Xin Xia, Xuefeng Xiao, and Liang Lin. Control-a-video: Controllable text-to-video generation with diffusion models, 2023. 8

[11] Yuhua Chen, Cordelia Schmid, and Cristian Sminchisescu. Self-supervised learning with geometric constraints in monocular video: Connecting flow, depth, and camera. In *ICCV*, 2019. 2

[12] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *CVPR*, 2017. 4, 5, 1, 3

[13] Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision transformers need registers. In *ICLR*, 2024. 2

[14] Xingshuai Dong, Matthew A Garratt, Sreenatha G Anavatti, and Hussein A Abbass. Towards real-time monocular depth estimation for robotics: A survey. *IEEE Transactions on Intelligent Transportation Systems*, 23(10):16940–16961, 2022. 1, 6

[15] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. *NeurIPS*, 27, 2014. 2

[16] Ruoyu Feng, Wenming Weng, Yanhui Wang, Yuhui Yuan, Jianmin Bao, Chong Luo, Zhibo Chen, and Baining Guo. Ccedit: Creative and controllable video editing via diffusion models. In *CVPR*, 2024. 8

[17] Huan Fu, Mingming Gong, Chaohui Wang, Kayhan Batmanghelich, and Dacheng Tao. Deep ordinal regression network for monocular depth estimation. In *CVPR*, 2018. 2

[18] Xiao Fu, Wei Yin, Mu Hu, Kaixuan Wang, Yuexin Ma, Ping Tan, Shaojie Shen, Dahua Lin, and Xiaoxiao Long. Geowizard: Unleashing the diffusion priors for 3d geometry estimation from a single image. In *ECCV*, 2024. 2

[19] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *IJRR*, 2013. 4, 6, 1, 3

[20] Yuwei Guo, Ceyuan Yang, Anyi Rao, Maneesh Agrawala, Dahua Lin, and Bo Dai. Sparsectrl: Adding sparse controls to text-to-video diffusion models. In *ECCV*, 2024. 8

[21] Jing He, Haodong Li, Wei Yin, Yixun Liang, Leheng Li, Kaiqiang Zhou, Hongbo Liu, Bingbing Liu, and Ying-Cong Chen. Lotus: Diffusion-based visual foundation model for high-quality dense prediction. *arXiv preprint arXiv:2409.18124*, 2024. 2

[22] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *NeurIPS*, 33:6840–6851, 2020. 3

[23] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *NeurIPS*, 35:8633–8646, 2022. 2, 3

[24] Fa-Ting Hong, Longhao Zhang, Li Shen, and Dan Xu. Depth-aware generative adversarial network for talking head video generation. In *CVPR*, 2022. 1

[25] Mu Hu, Wei Yin, Chi Zhang, Zhipeng Cai, Xiaoxiao Long, Hao Chen, Kaixuan Wang, Gang Yu, Chunhua Shen, and Shaojie Shen. Metric3d v2: A versatile monocular geometric foundation model for zero-shot metric depth and surface normal estimation. *IEEE TPAMI*, 46(12):10579–10596, 2024. 2

[26] Wenbo Hu, Menghan Xia, Chi-Wing Fu, and Tien-Tsin Wong. Mononizing binocular videos. *TOG (Proceedings of ACM SIGGRAPH Asia)*, 39(6):228:1–228:16, 2020. 7

[27] Wenbo Hu, Hengshuang Zhao, Li Jiang, Jiaya Jia, and Tien-Tsin Wong. Bidirectional projection network for cross dimensional scene understanding. In *CVPR*, 2021. 1

[28] Wenbo Hu, Yuling Wang, Lin Ma, Bangbang Yang, Lin Gao, Xiao Liu, and Yuewen Ma. Tri-miprf: Tri-mip representation for efficient anti-aliasing neural radiance fields. In *ICCV*, 2023. 1

[29] Junpeng Jing, Ye Mao, and Krystian Mikolajczyk. Match-stereo-videos: Bidirectional alignment for consistent dynamic stereo matching. In *ECCV*, 2024. 4

[30] Nikita Karaev, Ignacio Rocco, Benjamin Graham, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. Dy-

namicstereo: Consistent dynamic depth from stereo videos. In *CVPR*, 2023. 4

[31] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *NeurIPS*, 35:26565–26577, 2022. 3, 4, 1

[32] Bingxin Ke, Anton Obukhov, Shengyu Huang, Nando Metzger, Rodrigo Caye Daudt, and Konrad Schindler. Repurposing diffusion-based image generators for monocular depth estimation. In *CVPR*, 2024. 2, 4, 6, 7, 3

[33] DP Kingma. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 4

[34] Diederik P Kingma. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5

[35] Johannes Kopf, Xuejian Rong, and Jia-Bin Huang. Robust consistent video depth estimation. In *CVPR*, 2021. 2

[36] Jin Han Lee, Myung-Kyu Han, Dong Wook Ko, and Il Hong Suh. From big to small: Multi-scale local planar guidance for monocular depth estimation. *arXiv preprint arXiv:1907.10326*, 2019. 2

[37] Yixuan Li, Lihan Jiang, Linning Xu, Yuanbo Xiangli, Zhenzhi Wang, Dahua Lin, and Bo Dai. Matrixcity: A large-scale city dataset for city-scale neural rendering and beyond. In *ICCV*, 2023. 4

[38] Zhenyu Li, Zehui Chen, Xianming Liu, and Junjun Jiang. Depthformer: Exploiting long-range correlation and local information for accurate monocular depth estimation. *Machine Intelligence Research*, 20(6):837–854, 2023. 2

[39] Zhaoshuo Li, Wei Ye, Dilin Wang, Francis X Creighton, Russell H Taylor, Ganesh Venkatesh, and Mathias Unberath. Temporally consistent online depth estimation in dynamic scenes. In *WACV*, 2023. 2

[40] Zhengzhe Liu, Xiaojuan Qi, and Chi-Wing Fu. 3d-to-2d distillation for indoor scene parsing. In *CVPR*, 2021. 1

[41] Xuan Luo, Jia-Bin Huang, Richard Szeliski, Kevin Matzen, and Johannes Kopf. Consistent video depth estimation. *TOG (Proceedings of ACM SIGGRAPH)*, 39(4), 2020. 2

[42] Gonzalo Martin Garcia, Karim Abou Zeid, Christian Schmidt, Daan de Geus, Alexander Hermans, and Bastian Leibe. Fine-tuning image-conditional diffusion models is easier than you think. In *WACV*, 2025. 2

[43] Maxime Oquab, Timothée Darcet, Theo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Russell Howes, Po-Yao Huang, Hu Xu, Vasu Sharma, Shang-Wen Li, Wojciech Galuba, Mike Rabbat, Mido Assran, Nicolas Ballas, Gabriel Synnaeve, Ishan Misra, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision. In *TMLR*, 2024. 2

[44] E. Palazzolo, J. Behley, P. Lottes, P. Giguère, and C. Stachniss. ReFusion: 3D Reconstruction in Dynamic Environments for RGB-D Cameras Exploiting Residuals. In *IROS*, 2019. 6, 1, 2, 3

[45] Vaishakh Patil, Christos Sakaridis, Alexander Liniger, and Luc Van Gool. P3depth: Monocular depth estimation with a piecewise planarity prior. In *CVPR*, 2022. 2

[46] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, and A. Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *CVPR*, 2016. 5, 7

[47] Luigi Piccinelli, Yung-Hsu Yang, Christos Sakaridis, Mattia Segu, Siyuan Li, Luc Van Gool, and Fisher Yu. Unidepth: Universal monocular metric depth estimation. In *CVPR*, 2024. 2

[48] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 4

[49] René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE TPAMI*, 44(3):1623–1637, 2020. 2, 4, 3

[50] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *ICCV*, 2021. 4

[51] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 4

[52] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015. 4

[53] Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. In *ICLR*, 2022. 4, 1

[54] Jiahao Shao, Yuanbo Yang, Hongyu Zhou, Youmin Zhang, Yujun Shen, Matteo Poggi, and Yiyi Liao. Learning temporally consistent video depth from video diffusion priors. *arXiv preprint arXiv:2406.01493*, 2024. 3, 6, 7

[55] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *ECCV*, 2012. 6

[56] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *ICML*, 2015. 3

[57] Jiaming Sun, Yiming Xie, Linghao Chen, Xiaowei Zhou, and Hujun Bao. Neuralrecon: Real-time coherent 3d reconstruction from monocular video. In *CVPR*, 2021. 1

[58] Zachary Teed and Jia Deng. Deepv2d: Video to depth with differentiable structure from motion. In *ICLR*, 2020. 2

[59] Patrick von Platen, Suraj Patil, Anton Lozhkov, Pedro Cuenca, Nathan Lambert, Kashif Rasul, Mishig Davaadorj, Dhruv Nair, Sayak Paul, William Berman, Yiyi Xu, Steven Liu, and Thomas Wolf. Diffusers: State-of-the-art diffusion models. https://github.com/huggingface/diffusers, 2022. 5

[60] Chaoyang Wang, Simon Lucey, Federico Perazzi, and Oliver Wang. Web stereo video supervision for depth prediction from dynamic scenes. In *IEEE 3DV*, 2019. 2

[61] Yiran Wang, Zhiyu Pan, Xingyi Li, Zhiguo Cao, Ke Xian, and Jianming Zhang. Less is more: Consistent video depth estimation with masked frames modeling. In *ACM MM*, 2022. 2

[62] Yiran Wang, Min Shi, Jiaqi Li, Zihao Huang, Zhiguo Cao, Jianming Zhang, Ke Xian, and Guosheng Lin. Neural video depth stabilizer. In *ICCV*, 2023. 4

10

[63] Yiran Wang, Min Shi, Jiaqi Li, Zihao Huang, Zhiguo Cao, Jianming Zhang, Ke Xian, and Guosheng Lin. Neural video depth stabilizer. In *ICCV*, 2023. 2, 6, 7

[64] Jinbo Xing, Hanyuan Liu, Menghan Xia, Yong Zhang, Xintao Wang, Ying Shan, and Tien-Tsin Wong. Tooncrafter: Generative cartoon interpolation. *TOG (Proceedings of ACM SIGGRAPH Asia)*, 2024. 2, 3

[65] Jinbo Xing, Menghan Xia, Yong Zhang, Haoxin Chen, Wangbo Yu, Hanyuan Liu, Xintao Wang, Tien-Tsin Wong, and Ying Shan. Dynamicrafter: Animating open-domain images with video diffusion priors. In *ECCV*, 2024. 2, 3

[66] Guanglei Yang, Hao Tang, Mingli Ding, Nicu Sebe, and Elisa Ricci. Transformer-based attention networks for continuous pixel-wise prediction. In *ICCV*, 2021. 2

[67] Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything: Unleashing the power of large-scale unlabeled data. In *CVPR*, 2024. 2, 4, 6, 7, 1

[68] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *NeurIPS*, 2024. 2, 4, 6, 7, 1, 3

[69] Rajeev Yasarla, Hong Cai, Jisoo Jeong, Yunxiao Shi, Risheek Garrepalli, and Fatih Porikli. Mamo: Leveraging memory and attention for monocular video depth estimation. In *ICCV*, 2023. 2

[70] Rajeev Yasarla, Manish Kumar Singh, Hong Cai, Yunxiao Shi, Jisoo Jeong, Yinhao Zhu, Shizhong Han, Risheek Garrepalli, and Fatih Porikli. Futuredepth: Learning to predict the future improves video depth estimation. In *ECCV*, 2024. 2

[71] Wei Yin, Chi Zhang, Hao Chen, Zhipeng Cai, Gang Yu, Kaixuan Wang, Xiaozhi Chen, and Chunhua Shen. Metric3d: Towards zero-shot metric 3d prediction from a single image. In *ICCV*, 2023. 2

[72] Wangbo Yu, Jinbo Xing, Li Yuan, Wenbo Hu, Xiaoyu Li, Zhipeng Huang, Xiangjun Gao, Tien-Tsin Wong, Ying Shan, and Yonghong Tian. Viewcrafter: Taming video diffusion models for high-fidelity novel view synthesis. *arXiv preprint arXiv:2409.02048*, 2024. 1

[73] Zehao Yu, Songyou Peng, Michael Niemeyer, Torsten Sattler, and Andreas Geiger. Monosdf: Exploring monocular geometric cues for neural implicit surface reconstruction. *NeurIPS*, 2022. 1

[74] Haokui Zhang, Chunhua Shen, Ying Li, Yuanzhouhan Cao, Yu Liu, and Youliang Yan. Exploiting temporal consistency for real-time video depth estimation. In *ICCV*, 2019. 2

[75] Yuang Zhang, Jiaxi Gu, Li-Wen Wang, Han Wang, Junqi Cheng, Yuefeng Zhu, and Fangyuan Zou. Mimicmotion: High-quality human motion video generation with confidence-aware pose guidance. *arXiv preprint arXiv:2406.19680*, 2024. 5

[76] Yabo Zhang, Yuxiang Wei, Dongsheng Jiang, Xiaopeng Zhang, Wangmeng Zuo, and Qi Tian. Controlvideo: Training-free controllable text-to-video generation. In *ICLR*, 2024. 8

[77] Zhoutong Zhang, Forrester Cole, Richard Tucker, William T Freeman, and Tali Dekel. Consistent depth of moving objects in video. *TOG (Proceedings of ACM SIGGRAPH)*, 40(4):1–12, 2021. 2

DepthCrafter: Generating Consistent Long Depth Sequences for Open-world Videos

Supplementary Material

Appendix

In this supplementary material, we provide additional implementation details in Appendix A and more evaluations in Appendix B. For more visual results and details, *we highly recommend referring to the webpage:* `https://depthcrafter.github.io`, since the visual quality of the generated depth sequences can be better accessed with interactive videos. We would release the code and model to facilitate further research and applications.

A. Implementation Details

A.1. Data Preparation

Following conventional practice in depth estimation [67, 68], we represent the depth in the disparity domain. We target relative depth estimation, so we normalize the disparity values to the range of $[0, 1]$ by the maximum and minimum disparity values in the sequence. Since the training of our DepthCrafter only involves the U-Net model, with the VAE freezing, we can pre-process the latents of videos and the corresponding depth sequences in advance. This caching mechanism significantly reduces the training time and memory consumption, as the latents do not need to be re-computed during the training process, and the VAE does not need to be loaded into the memory.

A.2. Training Details

We follow the EDM-framework [31] to train our DepthCrafter. This can be mathematically formulated as Eqs. (1) to (3). The c_{in}, c_{out}, c_{skip}, and c_{noise} in Eq. (3) are the EDM preconditionining functions [31, 53]:

$$
\begin{aligned}
c_{\text{in}}(\sigma_t) &= 1/\sqrt{1 + \sigma_t^2}, \\
c_{\text{out}}(\sigma_t) &= -\sigma_t/\sqrt{1 + \sigma_t^2}, \\
c_{\text{skip}}(\sigma_t) &= 1/(1 + \sigma_t^2), \\
c_{\text{noise}}(\sigma_t) &= 0.25 \cdot \log(\sigma_t).
\end{aligned}
\tag{5}
$$

c_{in} and c_{out} are used to scale the input and output magnitudes, c_{skip} is used to modulate the skip connection, and c_{noise} is used to map the noise level σ_t into a conditioning input for the denoiser F_θ. The λ_{σ_t} in Eq. (2) effectively incurs a per-sample loss weight for balancing different noise levels, which is set as:

$$
\lambda_{\sigma_t} = 1/c_{\text{out}}(\sigma_t)^2.
\tag{6}
$$

During training, we randomly sample the noise level σ_t from a log-normal distribution:

$$
\ln(\sigma_t) \sim \mathcal{N}(0.7, 1.6^2),
\tag{7}
$$

which is following the EDM-framework [31] to target the training efforts to the relevant range.

We train our DepthCrafter on eight NVIDIA A100 GPUs with a learning rate of 10^{-5}, and a batch size of 8. We adopted the DeepSpeed ZeRO-2 strategy, gradient checkpointing, and mixed precision training to reduce memory consumption during training. We also highly optimize the U-Net structure and cache the latents to further reduce memory consumption. The first and third training stages consume around 40GB of GPU memory per device, while the second stage consumes around 80GB. The "temporal layers" mentioned in Sec. 3 are the layers performed on the time axis, such as temporal transformer and temporal resnet blocks. The remaining layers are spatial layers, such as spatial transformers and spatial resnet blocks.

A.3. Benchmark Evaluation Details

Since existing monocular depth estimation methods and benchmarks are mainly tailored for static images, we re-compile the public benchmarks to evaluate the video depth estimation methods. First, we re-format the testing datasets in the form of videos that are originally in the form of images. Specifically, for the ScanNet V2 dataset [12], we extracted the first 90 RGB-D frames from the original sensor data sequences at a rate of 15 frames per second. Besides, there are black regions in the corners of the images due to the camera calibration, which would affect the depth estimation evaluation. We carefully crop the borders of the images to remove the black regions, *i.e.* cropping 8 pixels from the top and bottom, and 11 pixels from the left and right. For the KITTI dataset [19], we extracted the first 110 frames from the original sequences without downsampling the frame rate, since the difference between consecutive frames is relatively large. And for the Bonn dataset [44], which was usually not included in the evaluation due to the small scale, but we find it is a good complement to the ScanNet dataset for indoor scenes as it contains dynamic contents while ScanNet contains only static scenes. We selected five sequences from the Bonn dataset, each with 110 frames, for evaluation. For the Sintel dataset [7], as it is a synthetic dataset, we directly used the original sequences.

For the evaluation metrics, we followed the insight from the commonly used metrics in the depth estimation, includ-
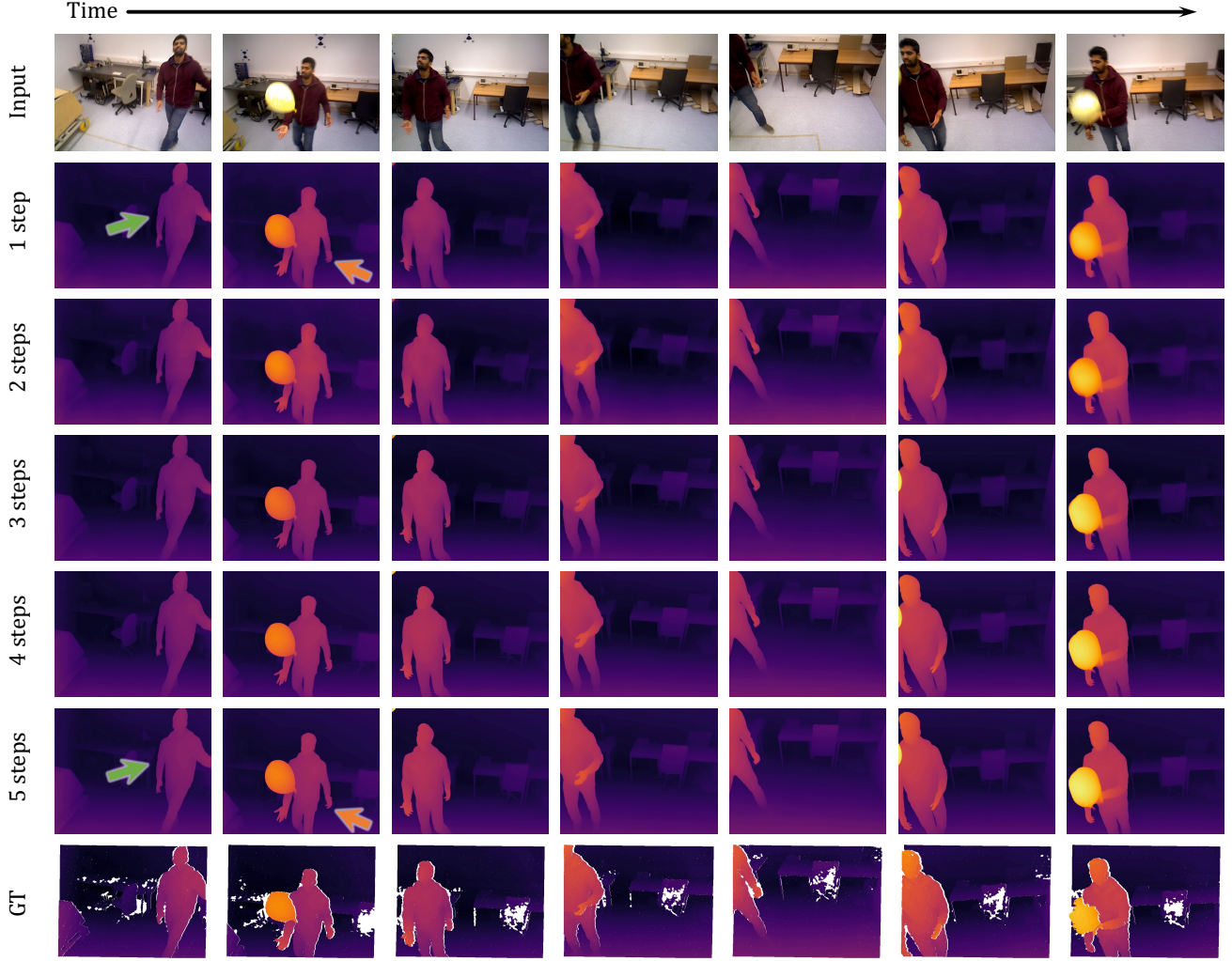
Figure S1. Effects of the number of denoising steps in our DepthCrafter. We show an example from the Bonn dataset [44], where the depth sequences are generated with different numbers of denoising steps. The green and orange arrows indicate the regions where more denoising steps can refine the depth details.

Table S1. Performance comparison of our DepthCrafter with different numbers of denoising steps. For reference, we also include the results of Marigold [32] and Depth-Anything-V2 [68]. The inference speed is measured in milliseconds per frame at the resolution of 1024×576. **Best** and <u>second best</u> results are highlighted.

Method	Steps	*ms* / frame ↓ @1024×576	Sintel (∼50 frames) AbsRel↓	δ_1↑	Scannet (90 frames) AbsRel↓	δ_1↑	KITTI (110 frames) AbsRel↓	δ_1↑	Bonn (110 frames) AbsRel↓	δ_1↑
Marigold [32]		1070.29	0.532	0.515	0.166	0.769	0.149	0.796	0.091	0.931
Depth-Anything-V2 [68]		**180.46**	0.367	0.554	0.135	0.822	0.140	0.804	0.106	0.921
DepthCrafter (Ours)	1	<u>337.10</u>	0.319	0.651	0.132	0.826	0.138	0.812	0.084	0.954
	2	369.28	0.301	0.661	0.132	0.828	0.138	0.814	0.083	0.955
	3	401.47	<u>0.273</u>	0.693	**0.123**	<u>0.854</u>	0.111	0.877	0.073	0.971
	4	433.65	0.293	**0.697**	**0.123**	**0.856**	0.107	0.888	<u>0.072</u>	0.971
	5	465.84	**0.270**	**0.697**	**0.123**	**0.856**	**0.104**	**0.896**	**0.071**	<u>0.972</u>
	6	498.03	0.299	<u>0.696</u>	<u>0.124</u>	0.851	<u>0.105</u>	<u>0.891</u>	<u>0.072</u>	**0.973**
	10	626.72	0.291	0.694	0.125	0.849	0.106	0.890	0.073	<u>0.972</u>
	25	1109.42	0.292	**0.697**	0.125	0.848	0.110	0.881	0.075	0.971

2

Table S2. Effectiveness of our three-stage training strategy. We show the performance of our DepthCrafter with different training stages on the Sintel, Scannet, KITTI, and Bonn datasets. For reference, we also include the results of Marigold [32] and Depth-Anything-V2 [68]. **Best** and <u>second best</u> results are highlighted.

Method	Training Stages	Sintel (~50 frames)		Scannet (90 frames)		KITTI (110 frames)		Bonn (110 frames)	
		AbsRel↓	δ_1↑	AbsRel↓	δ_1↑	AbsRel↓	δ_1↑	AbsRel↓	δ_1↑
Marigold [32]		0.532	0.515	0.166	0.769	0.149	0.796	0.091	0.931
Depth-Anything-V2 [68]		0.367	0.554	0.135	0.822	0.140	0.804	0.106	0.921
	1	0.322	0.626	0.170	0.721	0.174	0.724	0.103	0.917
DepthCrafter (Ours)	2	<u>0.316</u>	<u>0.675</u>	<u>0.134</u>	<u>0.826</u>	<u>0.127</u>	<u>0.844</u>	<u>0.090</u>	<u>0.935</u>
	3	**0.270**	**0.697**	**0.123**	**0.856**	**0.104**	**0.896**	**0.071**	**0.972**

ing the absolute relative error (AbsRel) and the δ_1 metric, but modified the scale and shift alignment from per-image to per-video. This is because the depth values for a video should be consistent across frames, otherwise, the depth sequences would be flickering. During evaluation, we first align the depth sequences to the ground truth by the scale and shift, using a least-square optimization. Following Mi-Das [49], we cap the maximum depth values to a certain value for different datasets, *e.g.*, 70 meters for the SinTel dataset, 80 meters for the KITTI dataset, and 10 meters for the ScanNet, Bonn, and NYUv2 datasets.

B. Additional Evaluations

B.1. Effect of Number of Denoising Steps

During inference, the number of denoising steps is a crucial hyperparameter that affects the trade-off between the inference speed and the depth estimation quality. The practice in image-to-video diffusion models [3] is to set the number of denoising steps to around 25. However, as shown in Fig. S1, we find that the number of denoising steps can be reduced significantly for video depth estimation, even one step works well. This is because the video depth estimation task is more deterministic than the video generation task. And we can see in the figure that more denoising steps would consistently improve the structure details of the generated depth sequences. In Tab. S1, we show the results of our DepthCrafter with different numbers of denoising steps. We can see that our DepthCrafter significantly outperforms existing strong baselines, such as Marigold [32] and Depth-Anything-V2 [68], even with only one denoising step. The performance of our DepthCrafter is increased with more denoising steps, but the improvement gets saturated after five steps. Thus we set the number of denoising steps to five in our experiments, which achieves a good trade-off between the inference speed and the depth estimation quality. The inference speed of our DepthCrafter with five denoising steps is 465.84 ms per 1024×576 frame, which is acceptable for many applications.

B.2. Effectiveness of Training Stages

In the main paper, we ablate the performance of our DepthCrafter with three training stages, only on the Sintel [7] dataset. To complement the evaluation, we further evaluate the effectiveness of our three-stage training strategy on all the datasets, including Sintel [7], Scannet [12], KITTI [19], and Bonn [44]. As shown in Tab. S2, we can observe that, even only with the first two stages, our DepthCrafter already outperforms the existing strong baselines, such as Marigold [32] and Depth-Anything-V2 [68]. More importantly, the performance improvement with the training stages is consistent across all the datasets. It indicates that our three-stage training strategy is effective for improving the generalization ability of our DepthCrafter to diverse open-world videos.

B.3. Effects of Classifier-Free Guidance

Classifier-free guidance (CFG) is proven to be effective in improving the details of the generated videos in video diffusion models [3, 8, 9, 64, 65]. In our DepthCrafter, we also investigate the effectiveness of CFG in video depth estimation. As shown in Fig. S2, we show an example frame from the KITTI dataset, where the results of our DepthCrafter with and without CFG are compared. We can see that the CFG can indeed improve the visual details of the generated depth sequences, especially for the fine-grained structures. However, we find that the CFG may slightly degrade the quantitative accuracy of the depth estimation, as shown in Tab. S3. This may be because the CFG is designed for improving the details of the generated videos, while the depth estimation task is more deterministic and requires more accurate predictions. Since adopting the CFG would also introduce additional computation, we do not use the CFG in our DepthCrafter for the main experiments. However, if the users are more interested in the visual details of the depth sequences, they can consider incorporating the CFG into our DepthCrafter.
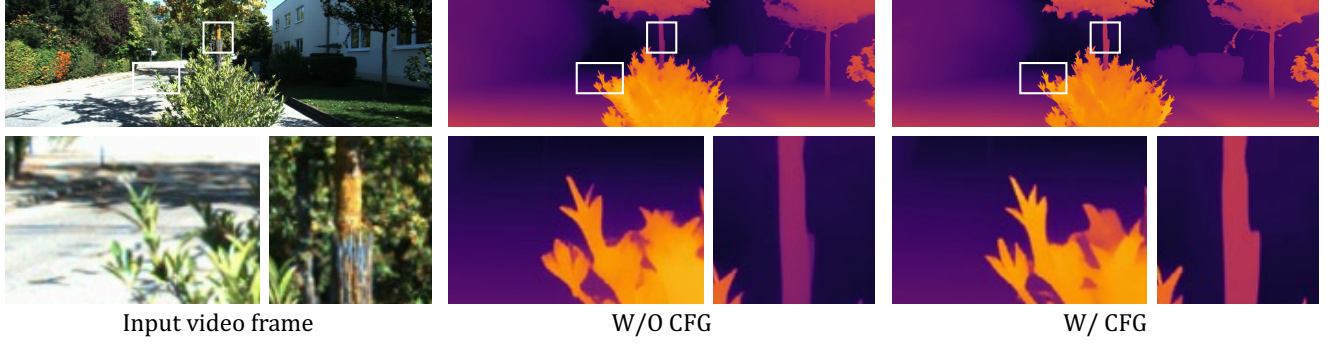
| Input video frame | W/O CFG | W/ CFG |

Figure S2. Effects of the classifier-free guidance (CFG) on the generated depth sequences. For better visualization, we blow up two regions that contain structures with fine-grained details.

Table S3. Effects of the classifier-free guidance (CFG). For reference, we also include the results of Marigold [32] and Depth-Anything-V2 [68]. **Best** and <u>second best</u> results are highlighted.

Method	Sintel (∼50 frames)		Scannet (90 frames)		KITTI (110 frames)		Bonn (110 frames)	
	AbsRel↓	δ_1↑	AbsRel↓	δ_1↑	AbsRel↓	δ_1↑	AbsRel↓	δ_1↑
Marigold [32]	0.532	0.515	0.166	0.769	0.149	0.796	0.091	<u>0.931</u>
Depth-Anything-V2 [68]	0.367	0.554	<u>0.135</u>	0.822	0.140	0.804	0.106	0.921
DepthCrafter W/O CFG	**0.270**	**0.697**	**0.123**	**0.856**	**0.104**	**0.896**	**0.071**	**0.972**
DepthCrafter W/ CFG	<u>0.315</u>	<u>0.692</u>	**0.123**	<u>0.850</u>	<u>0.108</u>	<u>0.885</u>	<u>0.076</u>	**0.972**

4