Fine-Tuning Image-Conditional Diffusion Models is Easier than You Think

Gonzalo Martin Garcia[1] Karim Abou Zeid[1] Christian Schmidt[1]
Daan de Geus[1,2] Alexander Hermans[1] Bastian Leibe[1]

[1]RWTH Aachen University [2]Eindhoven University of Technology

vision.rwth-aachen.de/diffusion-e2e-ft

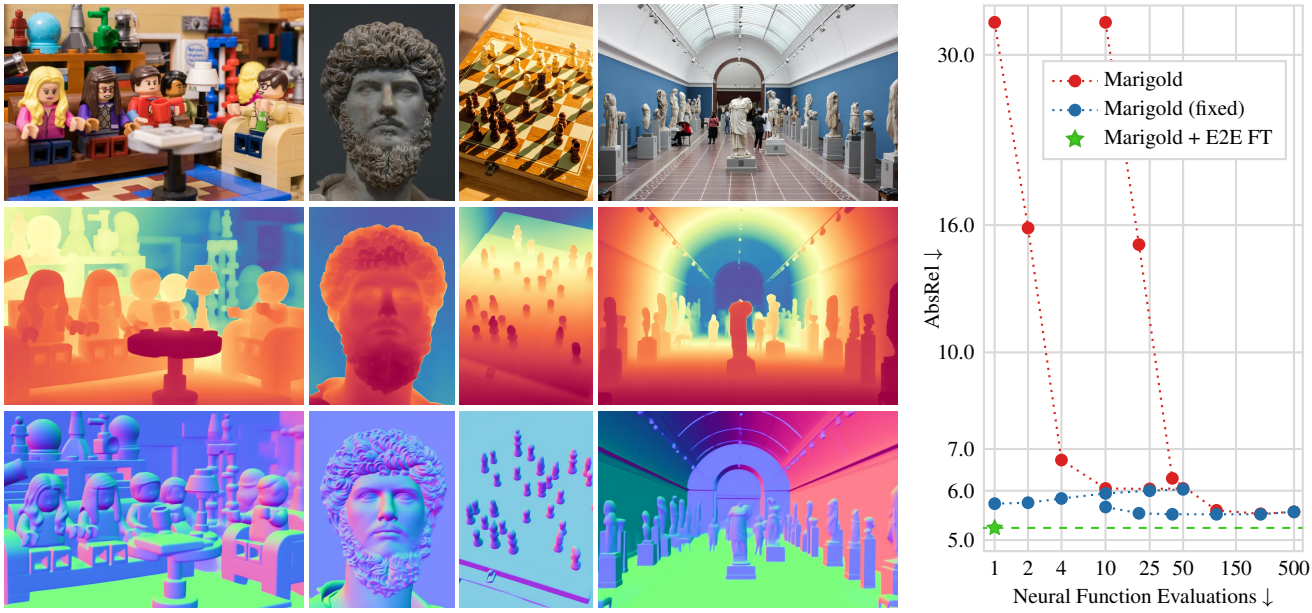


Figure 1. **Repurposing diffusion models for geometry estimation is as simple as end-to-end fine-tuning.** Left: Depth and normal predictions of our method on in-the-wild images. Right: A simple fix for the DDIM scheduler enables single-step inference for recent diffusion-based depth estimators; and simple end-to-end fine-tuning outperforms more complex diffusion baselines in speed and accuracy.

Abstract

Recent work showed that large diffusion models can be reused as highly precise monocular depth estimators by casting depth estimation as an image-conditional image generation task. While the proposed model achieved state-of-the-art results, high computational demands due to multi-step inference limited its use in many scenarios. In this paper, we show that the perceived inefficiency was caused by a flaw in the inference pipeline that has so far gone unnoticed. The fixed model performs comparably to the best previously reported configuration while being more than 200× faster. To optimize for downstream task performance, we perform end-to-end fine-tuning on top of the single-step model with task-specific losses and get a deterministic model that outperforms all other diffusion-based depth and normal estimation models on common zero-shot benchmarks. We surprisingly find that this fine-tuning protocol also works directly on Stable Diffusion and achieves comparable performance to current state-of-the-art diffusion-based depth and normal estimation models, calling into question some of the conclusions drawn from prior works.

1. Introduction

Monocular depth estimation has long been used in many downstream tasks, such as image and video editing, scene reconstruction, novel view synthesis, and robotic navigation. Since the task is inherently ill-posed due to the scale-

distance ambiguity, learning-based methods need to incorporate strong semantic priors in order to perform well. For this reason, recent work has proposed to adapt large diffusion models [39] for monocular depth estimation by casting depth prediction as a conditional image generation task [24]. The resulting models show good task performance and exhibit remarkably high levels of details. However, the consensus in the community is that they tend to be slow [14, 16, 24], since they need to perform many evaluations of a large neural network during inference.

In this paper, we argue that, contrary to common belief, inference of conditional latent diffusion models such as Marigold [24] and follow-up work [14] should be able to yield reasonable predictions with a single inference step. We investigate the behavior of Marigold and find that its dismal performance in the few-step regime is due to a critical flaw in the inference pipeline. While this bug has already been reported in the general diffusion model literature [28], we demonstrate that it is particularly critical in the scope of image-conditional methods such as Marigold. In particular, our results indicate that *existing works have probably drawn wrong conclusions due to flawed inference results*.

With a small correction to the inference pipeline, Marigold-like models [14, 24] obtain single-step performance that is comparable to multi-step, ensembled inference, while **being more than 200× faster**. In fact, this bug-fix makes diffusion-based depth estimators speed-wise comparable to state-of-the-art discriminative depth estimation models, opening up exciting avenues for further improvements. First, a single-step model allows efficient task-specific end-to-end fine-tuning since there is no need to backpropagate through multiple network invocations. Second, advanced techniques such as self-training with pseudo-labels [50, 51], which have been proven to be effective for discriminative models, can now be efficiently applied to diffusion-pretrained models as well.

We fine-tune Marigold end-to-end into a deterministic affine-invariant depth estimator for monocular images using a scale and shift invariant loss function [37]. To our surprise, this model outperforms the best configurations of Marigold. We repeat this experiment with the task of surface normal estimation and find similar results: end-to-end fine-tuning with a task-specific loss outperforms more complicated architectures which were trained on more data.

Following Occam's Razor, we find that even the simplest baseline, direct fine-tuning of Stable Diffusion (SD) [39] into a deterministic feed-forward model, outperforms Marigold and other diffusion-based depth- and normal estimation methods. These findings contradict some conclusions that have been drawn in earlier works. First, diffusion-based depth and normal estimation methods do not need to be slow. Second, casting depth estimation as conditional image generation is not more effective than simple end-to-end fine-tuning. But in line with existing intuitions, we find a small dataset of high-quality (synthetic) labeled data sufficient for good performance.

In summary, our contributions are: (1) we analyze the behavior of Marigold and similar diffusion-based geometry estimation models and find a critical flaw in their inference pipeline, (2) we fix this critical flaw, enabling high-precision single-step inference and boosting efficiency of these models by more than 200×, and (3) we show that simple task-specific fine-tuning of diffusion models is sufficient for good performance in depth and normal estimation.

We demonstrate the effectiveness of our approach on common zero-shot benchmarks. Our deterministic one-step model outperforms other diffusion-based depth- and normal estimation methods, and achieves results comparable to state-of-the-art methods for affine-invariant depth prediction and surface normal estimation.

2. Related Work

Monocular Depth Estimation. Monocular depth estimation models predict a pixel-wise depth map of a scene from a single image. The most comprehensive representation is *metric depth*, which requires modeling the focal length to account for different cameras, introducing additional uncertainty [4, 21, 34, 53].

An alternative to metric depth is *affine-invariant depth*, which is equivalent to metric depth up to an unknown global scale and shift of the scene. This is the representation of choice for a wide range of monocular depth estimation methods [10, 22, 36, 37, 50, 51, 55]. Unlike metric depth, affine-invariant depth is independent of the focal length and is easier to regress in unfamiliar scenes, where no object can serve as a metric reference. However, it still preserves the distance ratios between objects. Metric depth can be recovered from affine-invariant depth by anchoring it with sparse, known depth values or by explicitly estimating the missing scale and shift [54].

To generalize to "in-the-wild" unseen scenes, depth estimation methods must handle a wide variety of environments. Methods designed for such applications are typically evaluated in a zero-shot setting, where the model is tested on unseen datasets without fine-tuning. Early zero-shot depth estimation methods already focused on generalization primarily through (at the time) large training datasets [27, 52]. MiDaS [37] achieved significant improvements by leveraging a combination of multiple datasets and a high-capacity backbone.

The transition from CNNs to ViTs [8] in DPT [36] and Omnidata [10] marked another key advancement in the field. Recent methods such as Depth Anything [50, 51] and Metric3D [21, 53] have followed the success of MiDaS by utilizing the high-capacity ViT-g DINOv2 [33] back-

bone and training on vast datasets—62M and 16M samples, respectively—one to two orders of magnitude larger than previous work. Notably, Depth Anything retains a simple DPT architecture, but combining a DINOv2 initialization and a large training dataset enables it to generalize to in-the-wild scenarios. Metric3D additionally utilizes the focal length to boost performance, which however limits its training data to scenes with known focal length.

Monocular Normal Estimation. Surface normal estimation involves predicting the orientation of surfaces in a scene from an image, resulting in a 3D vector representing the surface's orientation for each pixel. Hoiem *et al*. [19,20] were among the pioneers to introduce learning-based approaches for surface normal estimation. Since then, deep learning methods have become dominant, with many notable contributions [1–3, 10, 11, 22, 47]. Among these, Omnidata [10] stands out for training a UNet-based model on a large-scale dataset of 12M images captured in diverse environments. Its successor, Omnidata v2 [22], advanced the field by transitioning to a ViT-based architecture with a DPT head, incorporating sophisticated 3D-aware data augmentations to enhance generalization. In contrast, DSINE [2] adopts a more data-efficient approach that focuses on introducing new inductive biases to enhance performance. Lastly, Metric3D v2 [21], mentioned earlier, is also capable of predicting surface normals in addition to depth.

Diffusion Models for Geometry Estimation. Several recent generative text-to-3D methods [30, 35] explicitly produce multi-view depth and normal maps. However, these methods focus on isolated single-object scenarios, making them unsuitable for complex, in-the-wild environments.

Other approaches have utilized diffusion models for scene-level depth estimation [9, 41, 42, 56]. Among these, VPD [56] leverages Stable Diffusion [39] as both an image and a text feature extractor, incorporating a depth regression head that utilizes multi-scale image feature maps alongside text-to-image cross-attention maps. However, these models have not demonstrated robust generalization.

More recently, Marigold [24] fine-tuned SD to transition from generating realistic images conditioned on text to producing detailed and precise depth maps conditioned on images. The core idea behind this approach is that SD's ability to model realistic images also provides strong geometric and semantic priors, essential for accurate depth estimation. Key to Marigold's success is its exclusive training on high-quality synthetic datasets with perfect ground truth and a smooth transition from text-conditioned images to image-conditioned depth in the latent space, preserving the model's generalization capability.

Marigold has inspired several follow-up works. DiffCalib [17], for example, extends Marigold by jointly predicting depth and camera intrinsics through the addition of an incident map [57], which is denoised alongside the depth map. GeoWizard [14] jointly predicts both depth and surface normals through two parallel UNet evaluations, incorporating cross-attention between the two branches. However, a common drawback of these models is the high computational cost during inference, driven by the requirement for an iterative denoising process.

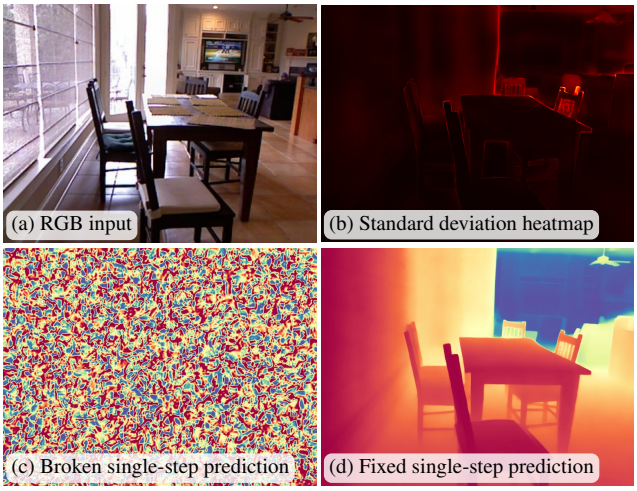


Figure 2. **Marigold output visualizations.** (a) The RGB input; (b) the pixel-wise standard deviation of Marigold's depth map output during 50-step DDIM inference; and Marigold's depth map prediction (c) before and (d) after the fixing the inference pipeline.

Addressing this limitation, DepthFM [16] combines Marigold's core ideas with Flow Matching [29] to reduce the number of denoising steps while maintaining high output quality. Additionally, the authors of Marigold now provide an LCM-distilled [32] version that allows for single-step evaluation, albeit at a reduced quality.

In our work, we observe that Marigold and follow-up methods, aside from DepthFM and Marigold LCM, which are designed for few-step prediction, suffer from a flawed implementation [28] of the DDIM [45] inference pipeline that prevents them from functioning effectively in the few-step regime. Furthermore, although the denoising diffusion fine-tuning objective used by Marigold and follow-up works for depth and normals estimation has shown effectiveness, we find it to be neither a key factor for good results nor clearly superior to task-specific end-to-end fine-tuning.

3. Image-Conditional Latent Diffusion Models

In this section, we review conditional latent diffusion models and how Marigold [24] leverages them for depth estimation. We also argue why single-step inference should produce sensible predictions for depth estimation, and explain why this has not been the case in practice so far.

3.1. Latent Diffusion Models

Denoising Diffusion Probabilistic Models (DDPM) learn a mapping from some simple, known noise distribution p_T

to the data distribution p_0 by reversing a stochastic forward process $p_t,\ t = 1, \ldots, T$, which repeatedly adds a small amount of Gaussian noise [18]. The variance β_t of the noise added in each step is chosen to be small, so that the reverse process can be approximated as Gaussians. Additionally, the number of steps T is set sufficiently large such that the terminal distribution p_T can be approximated as a Gaussian as well. Using $\alpha_t = 1 - \beta_t$ and $\bar{\alpha}_t = \prod_{\tau=1}^{t} \alpha_\tau$, the forward process can be written as $\mathbf{x}_t = \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1 - \bar{\alpha}_t}\boldsymbol{\epsilon}$ given a data sample $\mathbf{x}_0$ and $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. For the reverse process, a neural network is trained to gradually remove noise from its inputs to predict $\mathbf{x}_{t-1}$ given $\mathbf{x}_t$ [18].

Inference starts with noise $\mathbf{x}_T$, which is repeatedly denoised by passing it through the diffusion model and thus following the reverse diffusion process. A popular alternative inference scheme are *Denoising Diffusion Implicit Models (DDIM)* [45], which formulate a non-Markovian diffusion process that leads to the same training objective as DDPMs, but allows for inference in just a few steps.

Latent Diffusion Models (LDMs) [39] operate in the latent space of another model, *e.g.*, a *Variational Autoencoder (VAE)* [25]. The VAE consists of an encoder $\mathcal{E}$ and a decoder $\mathcal{D}$ and is trained independently of the LDM. The goal of the VAE encoder is to compress inputs into lower-dimensional latent codes, and for the decoder to faithfully reconstruct the input: $\mathcal{D}(\mathcal{E}(\mathbf{x})) \approx \mathbf{x}$. LDMs tend to be easier to train due to the reduced dimensionality and improved smoothness of the VAE's latent space.

In conditional diffusion models, both forward and reverse processes are conditioned on some additional input $\mathbf{c}$, such as a text description or an additional image input. The network is then trained to denoise the input given $\mathbf{c}$.

It has been shown that the optimal prediction at the final timestep T is the mean of the data distribution [23]. For unconditional and text-conditional image generation, the data distribution has multiple modes, and the mean therefore corresponds to a blurry image dominated by the average color and average scene composition. For a text-conditional model, the mean of the image distribution conditioned on the input text will be a blurry resemblance of the average image fitting the description. But for image-conditional generation, in particular depth and normal estimation, we expect the conditional distribution to be approximately unimodal, since an image usually corresponds to a single depth map. The optimal single-step prediction at T should therefore be close to the ground-truth depth/normal map.

3.2. Marigold

Marigold casts depth estimation as a conditional latent diffusion process, allowing it to build upon large pretrained diffusion models such as Stable Diffusion [24]. Marigold is conditioned on images, so following the above argument, we expect its single-step prediction to be a sensible, but blurry depth map.

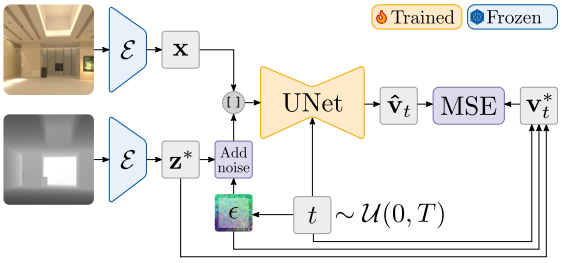


Figure 3. **Marigold [24] diffusion training for conditional depth map generation.** Marigold starts with a pretrained Stable Diffusion v2 model [39], which is fine-tuned for image-conditional generation of depth maps or surface normal maps.

Training. Fig. 3 shows an overview of Marigold's training procedure. Marigold adapts the SD v2 [39] UNet architecture for conditional depth map generation, using $\mathbf{v}$-*parametrization* [40] during training. The training objective is formulated in the latent space of the frozen SD VAE. The GT depth map $\mathbf{d}^*$ is replicated along the channel dimension to conform to the 3-channel inputs of the VAE and encoded as $\mathbf{z}^* = \mathcal{E}(\mathbf{d}^*)$. Similarly, the RGB latent is $\mathbf{x} = \mathcal{E}(\mathbf{I}_{\text{RGB}})$. During training, noise is added only to the depth latent to get $\mathbf{z}_t = \sqrt{\bar{\alpha}_t}\mathbf{z}^* + \sqrt{1 - \bar{\alpha}_t}\boldsymbol{\epsilon}$. The UNet receives the concatenated latents and the timestep t as inputs and predicts $\hat{\mathbf{v}}_t = \hat{\mathbf{v}}_\theta([\mathbf{z}_t, \mathbf{x}], t)$. The first convolutional layer is duplicated to accomodate the larger number of input channels [24].

The optimization target $\mathbf{v}_t^*$ at timestep t is then a linear combination of the sampled noise $\boldsymbol{\epsilon}$ and the GT depth map latent $\mathbf{z}^*$ such that $\mathbf{v}_t^* = \sqrt{\bar{\alpha}_t}\boldsymbol{\epsilon} - \sqrt{1 - \bar{\alpha}_t}\mathbf{z}^*$. The model is optimized with a squared error objective comparing the model prediction $\hat{\mathbf{v}}_t$ with $\mathbf{v}_t^*$. With the chosen noise schedule, $t = 1$ corresponds to little noise in the input, and the model is forced to predict the noise; nearer to $t = T$, the input is mostly noise and the model should predict a denoised image. Additionally, the authors observed significantly improved depth estimation performance by training with annealed multi-resolution noise [48] and sampling with isotropic Gaussian noise [24].

Fixing Single-Step Inference. We now turn to analyze the behavior of Marigold during inference. First, we show the pixel-wise standard deviation across steps for Marigold's default 50-step inference in Fig. 2b. We observe very small differences for almost all pixels, indicating that the model changes predictions very little during inference; in particular, this means that the first prediction of the 50-step schedule already corresponds closely to the final output. While we would expect that the single-step output should be similar to the first step of the 50-step schedule, as shown in Fig. 2c it rather corresponds to pure noise.

We find that this discrepancy is caused by a flaw in

the inference scheduler implementation used by Marigold and some derivative works [14, 24]. The flaw causes the model to receive an inconsistent pairing of timestep and noise, leading to nonsensical predictions. In particular, for a single-step prediction, the model receives a timestep encoding that indicates an almost perfect depth map whereas the actual input is pure noise. In other words, the model receives significantly more noise than it expects, and forwards the noise almost unchanged.

Fixing the flaw is simple: we need to align the timestep with the noise level. To do this, we can use the `trailing` setting as proposed in recent work [28] for image-generative models. However, we emphasize that while this setting provided only slight improvements for image generation [28], it is crucial for single-step inference in models such as Marigold. In Fig. 2c and Fig. 2d, we compare the outputs of the same model, using the flawed and the fixed inference schedule, respectively. Clearly, the fixed inference process produces sensible predictions, while the original does not.

4. End-to-End Fine-Tuning of LDMs

While diffusion-based depth estimation models show good overall performance and accurate details, they also exhibit artifacts such as blurred or over-sharpened outputs; see Fig. 6 for qualitative results. This could be due to the diffusion training objective, which does not guarantee that models are trained for the desired downstream task, but for the surrogate denoising task. To fix this, we directly fine-tune the diffusion model in an end-to-end manner. Note that end-to-end fine-tuning of a diffusion model without sensible single-step predictions would require backpropagation through multiple network invocations, which is computationally infeasible for models with hundreds of millions of parameters. This further shows the importance of fixing the inference pipeline as described in the previous section.

We continue to train the modified UNet used in the diffusion training stage. However, we do not sample the timestep t anymore and instead fix $t = T$ in order to always train the model for single-step prediction. Additionally, we replace the noise with the mean of the noise distribution, *i.e.*, zero, and only forward the RGB latent through the model. During the diffusion training, $t = T$ corresponded to $\mathbf{v}_T^* = \sqrt{\bar{\alpha}_T}\boldsymbol{\epsilon} - \sqrt{1-\bar{\alpha}_T}\mathbf{z}^*$ according to the $\mathbf{v}$-parameterization [40]. With $\bar{\alpha}_T \approx 0$, the model is trained to convert pure noise into a clean prediction $\mathbf{z}^*$, effectively performing single-step prediction. The output of the UNet can be converted into a latent depth map prediction using $\hat{\mathbf{z}}_0 = \sqrt{\bar{\alpha}_t}\mathbf{z}_t - \sqrt{1-\bar{\alpha}_t}\hat{\mathbf{v}}_\theta([\mathbf{z}_t, \mathbf{x}], t)$, which is decoded using the frozen VAE decoder and compared to the ground-truth depth map. Note the difference between this fine-tuning approach and Marigold's diffusion fine-tuning objective: Marigold trains to match the *latents* of the GT depth maps using an MSE loss; instead, we optimize to predict good *decoded depth maps*. Our resulting feedforward model is deterministic and we train it end-to-end using a task-specific loss. We show a comparison of this model to the previous inference strategy in Fig. 4.

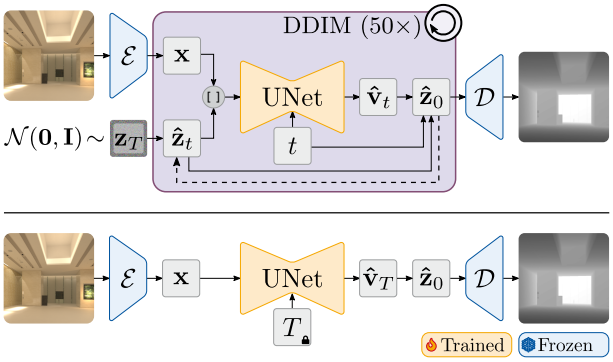


Figure 4. **Inference procedures** of Marigold (top) and our proposed simplification (bottom), which is deterministic and uses RGB latents *without noise*. Note that the timestep is fixed to T for the simplified model.

For monocular depth estimation, we use an affine-invariant loss function [37] which is invariant to global scale and shift of the depth map. In particular, we perform least-squares fitting between the ground-truth depth $\mathbf{d}^*$ and the predicted depth map $\mathbf{d}$ to estimate the scale and shift values s and t. The aligned prediction is then given as $\hat{\mathbf{d}} = s\mathbf{d} + t$, and the loss function is defined as

$$\mathcal{L}_{\mathrm{D}} = \frac{1}{HW}\sum_{i,j}\left|d^*_{i,j} - \hat{d}_{i,j}\right|, \tag{1}$$

where (i, j) denotes the pixel coordinates, and H and W are the height and width of the image, respectively.

For surface normal estimation, we use a loss based on the angle between the ground truth and predicted normals:

$$\mathcal{L}_{\mathrm{N}} = \frac{1}{HW}\sum_{i,j}\arccos\left(\frac{n^*_{i,j}\cdot\hat{n}_{i,j}}{\|n^*_{i,j}\|\|\hat{n}_{i,j}\|}\right), \tag{2}$$

where $n^*_{i,j}$ is the ground-truth normal at pixel (i, j), and $\hat{n}_{i,j}$ is the predicted normal.

5. Experimental Setup

Training Datasets. To allow for direct comparison with Marigold [24], we use the same training datasets: Hypersim [38], which consists of photorealistic indoor scenes, and Virtual KITTI 2 [6], which covers driving scenarios. Both datasets are fully synthetic and provide high-quality ground-truth annotations.

Evaluation Datasets. We evaluate the fine-tuned models on commonly used benchmarks for monocular depth estimation. NYUv2 [44] and ScanNet [7] provide RGB-D data of indoor environments captured with Kinect cam-

Table 1. **Main depth estimation results.** Distilling Marigold with LCM is worse than simply applying the fix without any retraining. Our task-specific training further boosts the model. We use Marigold's official evaluation pipeline to evaluate all diffusion depth models. Inference time is for an NVIDIA RTX 4090 GPU at 576×768 resolution for a single image.

Method	Steps	Ensemble	Inference time	NYUv2 [44]		KITTI [15]		ETH3D [43]		ScanNet [7]		DIODE [46]	
				AbsRel↓	δ1↑	AbsRel↓	δ1↑	AbsRel↓	δ1↑	AbsRel↓	δ1↑	AbsRel↓	δ1↑
Marigold [24]	50	10	24 s	5.5	96.4	9.9	91.6	6.5	**96.0**	6.4	95.1	30.8	77.3
Marigold [24]	50	1	3.1 s	6.0	95.9	10.5	90.4	7.1	95.1	6.9	94.5	31.0	77.2
Marigold LCM	4	5	1.8 s	6.2	95.6	9.9	91.7	6.9	95.5	7.0	94.5	30.9	77.6
Marigold LCM	1	1	**121** ms	6.5	95.4	10.7	89.9	7.5	94.5	7.6	93.8	31.5	76.3
Marigold + DDIM fix	1	1	**121** ms	5.7	96.2	10.8	89.6	6.9	95.5	6.6	95.2	31.1	76.8
Marigold + E2E FT	1	1	**121** ms	**5.2**	**96.6**	**9.6**	91.9	**6.2**	95.9	**5.8**	96.2	**30.2**	**77.9**
Stable Diffusion [39] + E2E FT	1	1	**121** ms	5.4	96.5	**9.6**	**92.1**	6.4	95.9	**5.8**	**96.5**	30.3	77.6

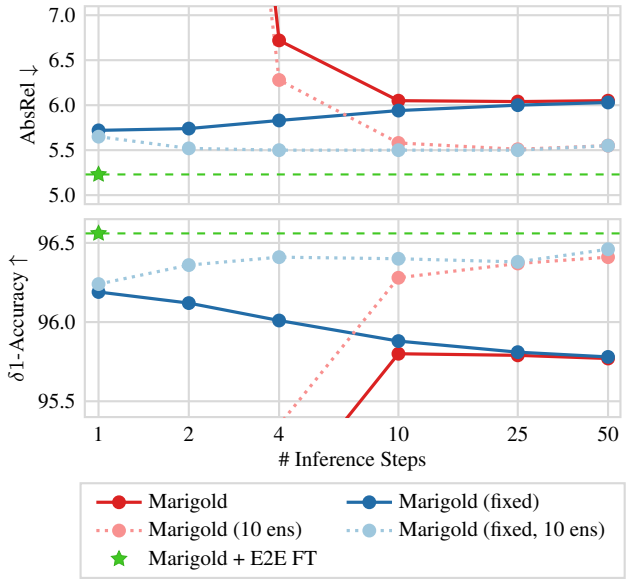


Figure 5. **Depth estimation results for different numbers of inference steps.** The fixed inference strategy outperforms the original Marigold in all cases, but especially for a single step. The impact of ensembling (*10 ens*) is still noticeable when performing multi-step inference. The deterministic end-to-end fine-tuned baseline outperforms all other versions of Marigold.

eras. ETH3D [43] and DIODE [46] consist of both indoor and outdoor scenes, derived from LiDAR sensors. KITTI [15] contains outdoor driving scenes captured by vehicle-mounted cameras and LiDAR sensors. For surface normal estimation, we evaluate on NYUv2, ScanNet, and additionally on iBims-1 [26], a high-quality indoor RGB-D dataset, as well as Sintel [5], a synthetic outdoor dataset.

Evaluation Protocol. All evaluations are conducted in the zero-shot setting. We evaluate affine-invariant depth predictions using the standard approach, which involves the same scale and shift optimization between the predicted depth and the ground truth as in the loss computation [37]. We report the mean absolute relative error (AbsRel), defined as the average relative difference between the ground-truth depth and the aligned predicted depth at each pixel, as well as the $\delta 1$ accuracy, which is the percentage of pixels where

Table 2. **Main normal estimation results.** Marigold for normal estimation is trained and evaluated by us.

Method	NYUv2 [44]		ScanNet [7]		iBims-1 [26]		Sintel [5]	
	Mean↓	11.25°↑	Mean↓	11.25°↑	Mean↓	11.25°↑	Mean↓	11.25°↑
Marigold (50, 10) [24]	18.8	55.9	17.7	58.8	18.4	64.3	39.1	14.9
Marigold + DDIM fix	17.4	56.5	16.8	57.6	18.1	62.9	37.1	15.7
Marigold + E2E FT	**16.2**	**61.4**	**14.7**	66.0	**15.8**	**69.9**	**33.5**	21.5
SD [39] + E2E FT	16.5	60.4	**14.7**	**66.1**	16.1	69.7	**33.5**	**22.3**

the ratio of the aligned predicted depth to the ground truth (and its inverse) is less than 1.25.

For surface normal predictions, we report the commonly used mean angular error (Mean) between the ground-truth normal vectors and the predictions, as well as the percentage of pixels with an angular error below 11.25 degrees.

Implementation Details. For depth estimation, we use the official Marigold checkpoint, whereas for normal estimation, we train a model with the same training setup as Marigold's depth estimation, encoding normal maps as 3D vectors in the color channels.

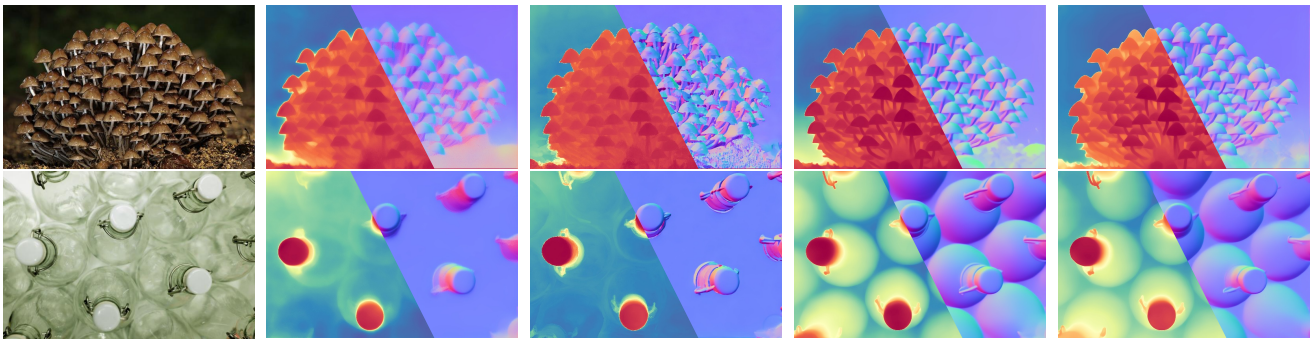
Unless noted otherwise, we follow Marigold's hyperparameters. We train all models for 20K iterations using the AdamW optimizer [31] with a base learning rate of 3×10^{-5} and an exponential learning rate decay after a 100-step warm-up. The batch size is set to 2, with gradient accumulation over 16 steps for an effective batch size of 32. This is a deliberate strategy to allow for mixing of images with different aspect ratios and resolutions. We use a specific mix of indoor and outdoor scenes from both Hypersim [38] (90%) and Virtual KITTI 2 [6] (10%), which was beneficial to the model's performance. Fine-tuning takes approximately 3 days on a single Nvidia H100 GPU.

6. Experimental Evaluation

Comparison with Marigold. As is evident from Fig. 5, the fixed DDIM scheduler reveals that Marigold's [24] multi-step denoising is not actually working: instead of improving the depth map with more denoising steps, the performance actually gets *worse*. This is because repeatedly denoising sharpens the depth map, but also accumulates errors because the model expects noised ground truth latents

Table 3. **Fixed DDIM scheduler and end-to-end fine-tuning (E2E FT) for GeoWizard [14].** We use the official code and model weights to re-evaluate the method on all datasets. Inference time is for a single 576×768-pixel image, evaluated on an NVIDIA RTX 4090 GPU. We obtain significant speed-ups, improving results. GeoWizard's original results include additional post-processing steps, such as smoothing.

Method	Steps	Ensemble	Inference time	NYUv2 [44]		ScanNet [7]		iBims-1 [26]		Sintel [5]	
				Mean↓	11.25°↑	Mean↓	11.25°↑	Mean↓	11.25°↑	Mean↓	11.25°↑
GeoWizard [14] (ECCV 24)	50	10	72 s	17.0	56.5	15.4	61.6	13.0	65.3	—	—
↳ reproduced by us	50	10	72 s	19.1	49.5	17.3	53.7	19.5	61.6	40.4	13.2
GeoWizard + DDIM fix	1	1	**254** ms	17.0	54.1	15.5	59.3	18.3	62.5	35.9	15.6
GeoWizard + E2E FT	1	1	**254** ms	**16.1**	**60.7**	**15.3**	**63.6**	**16.2**	**69.4**	**33.4**	**22.4**



RGB | Marigold (1, 1) | Marigold (50, 10) | Marigold + E2E FT | GeoWizard + E2E FT

Figure 6. **Qualitative results** of *fixed* single-step/ensembled multi-step Marigold, and deterministic end-to-end fine-tuned models based on Marigold and GeoWizard. Multi-step results show noise-like artifacts. The predictions of the end-to-end fine-tuned models tend to be more sharp and accurate. Additional qualitative results can be found in the supplementary materials.

Table 4. **Deterministic or Probabilistic.** The effect of different types of noise for task-specific fine-tuning for depth estimation.

Noise	NYUv2 [44]		KITTI [15]		ETH3D [43]		ScanNet [7]		DIODE [46]	
	AbsRel↓	$\delta1$↑	AbsRel↓	$\delta1$↑	AbsRel↓	$\delta1$↑	AbsRel↓	$\delta1$↑	AbsRel↓	$\delta1$↑
Marigold [24] fine-tuning										
Gaussian	5.3	96.4	9.9	91.4	6.3	95.9	5.9	96.0	30.5	77.3
Pyramid	5.4	96.5	9.9	91.0	6.3	**96.0**	6.0	95.9	**30.1**	77.7
Zeros	**5.2**	**96.6**	**9.6**	**91.9**	**6.2**	95.9	**5.8**	**96.2**	30.2	**77.9**
Stable Diffusion [39] fine-tuning										
Gaussian	5.8	96.1	9.8	91.5	6.6	95.5	6.0	96.1	30.7	77.2
Zeros	**5.4**	**96.5**	**9.6**	**92.1**	**6.4**	**95.9**	**5.8**	**96.5**	**30.3**	**77.6**

instead of its own predictions. This behavior was previously masked by the large error due to the broken DDIM scheduler. Note, however, that the fixed model performs strictly better than before, for any given number of steps. Ensembling does still provide noticeable benefits when using at least two inference steps. For single-step inference, the predictions tend to be highly correlated, in which case ensembling does not lead to significant improvements.

We compare vanilla Marigold and a variant distilled into a Latent Consistency Model (LCM) [32] for few-step inference with our single-step variants in Tab. 1. Using 4 steps and ensemble size 5, Marigold LCM performs on par with the best setting of vanilla Marigold (50 steps, 10 ensemble) on outdoor data (KITTI [15], DIODE [46]), but slightly worse for the indoor datasets. Using a single step only, performance drops even more strongly to 6.5 AbsRel on NYUv2 [44], compared to the previous best 5.5 AbsRel at 50 steps with an ensemble of 10 predictions. In contrast, we see that vanilla Marigold with the fixed DDIM scheduler reaches 5.7 AbsRel in a single step and without ensembling. Notably, this is better than the 6.0 AbsRel of 50-step Marigold *with the same model weights*. Moreover, we show that further end-to-end fine-tuning of Marigold leads to a substantial improvement of −0.5 AbsRel on NYUv2, surpassing all previous settings of Marigold, in a single step and without ensembling. Finally, directly fine-tuning Stable Diffusion [39] instead of the Marigold-pretrained model leads to comparable results. Tab. 2 paints the same overall picture for surface normal estimation; fixed single-step Marigold outperforms the vanilla multi-step, ensembled Marigold, and end-to-end fine-tuning yields even better results, even when applied to Stable Diffusion directly.

Deterministic or Probabilistic. We perform an ablation on the type of noise used during fixed-timestep fine-tuning; the results are shown in Tab. 4. "Gaussian" and "Pyramid" refer to the standard normal and multi-resolution noise commonly employed in diffusion training and used in Marigold, respectively. "Zeros" describes our default setting, *i.e.*, no noise. We find that using constant zeros performs slightly better than the alternatives, although the method seems to be fairly robust to the actual choice of noise.

Comparison with GeoWizard. GeoWizard [14] jointly predicts depth and surface normals, thus we also jointly fine-tune the model end-to-end for both tasks. Tab. 3 shows substantial improvements for surface normal estimation. In particular, the fine-tuned model performs substantially bet-

Table 5. **Comparison to state-of-the-art depth estimation methods.** †Metric3D v2 [21] was trained on ScanNet, so zero-shot evaluation on this dataset is not possible. We gray out results that were not reproducible with the released code and models.

Method	Training samples	NYUv2 [44]		KITTI [15]		ETH3D [43]		ScanNet [7]		DIODE [46]	
		AbsRel↓	$\delta1$↑	AbsRel↓	$\delta1$↑	AbsRel↓	$\delta1$↑	AbsRel↓	$\delta1$↑	AbsRel↓	$\delta1$↑
MiDaS [37] (TPAMI '22)	2M	11.1	88.5	23.6	63.0	18.4	75.2	12.1	84.6	33.2	71.5
LeReS [54] (CVPR '21)	354K	9.0	91.6	14.9	78.4	17.1	77.7	9.1	91.7	27.1	76.6
Omnidata v1 [10] (ICCV '21)	12.2M	7.4	94.5	14.9	83.5	16.6	77.8	7.5	93.6	33.9	74.2
HDN [55] (NeurIPS '22)	300K	6.9	94.8	11.5	86.7	12.1	83.3	8.0	93.9	24.6	78.0
DPT [36] (ICCV '21)	1.39M	9.8	90.3	10.0	90.1	7.8	94.6	8.2	93.4	18.2	75.8
Depth Anything [50] (CVPR '24)	62M	**4.3**	**98.1**	7.6	94.7	12.7	88.2	—	—	6.6	95.2
Depth Anything v2 [51] (arXiv '24)	62M	4.4	97.9	7.5	94.8	13.2	86.2	—	—	**6.5**	**95.4**
Metric3D [53] (ICCV '23)	8M	5.0	96.6	5.8	97.0	6.4	96.5	**7.4**	**94.1**	22.4	80.5
Metric3D v2 [21] (TPAMI '24)	16M	**4.3**	**98.1**	**4.4**	**98.2**	**4.2**	**98.3**	—†	—†	13.6	89.5
Marigold [24] (CVPR '24)	74K	5.5	96.4	9.9	91.6	6.5	96.0	6.4	95.1	30.8	77.3
GeoWizard [14] (ECCV '24)	278K	5.2	96.6	9.7	92.1	6.4	96.1	6.1	95.3	29.7	79.2
↳ reproduced by us	278K	5.7	96.2	14.4	82.0	7.5	94.3	6.1	95.8	31.4	77.1
DepthFM [16] (arXiv '24)	63K	6.5	95.6	8.3	93.4	—	—	—	—	22.5	80.0
↳ reproduced by us	63K	6.9	95.4	11.4	88.1	6.5	**96.2**	8.1	92.5	**25.0**	**78.3**
Marigold + E2E FT	74K	**5.2**	**96.6**	**9.6**	91.9	**6.2**	95.9	**5.8**	96.2	30.2	77.9
Stable Diffusion [39] + E2E FT	74K	5.4	96.5	**9.6**	**92.1**	6.4	95.9	**5.8**	**96.5**	30.3	77.6

Table 6. **Comparison to state-of-the-art normal estimation methods.** †Metric3D v2 [21] was trained on ScanNet, so zero-shot evaluation on this dataset is not possible. We gray out results that were not reproducible with the released code and models.

Method	Training samples	NYUv2 [44]		ScanNet [7]		iBims-1 [26]		Sintel [5]	
		Mean↓	11.25°↑	Mean↓	11.25°↑	Mean↓	11.25°↑	Mean↓	11.25°↑
Omnidata v1 [10] (ICCV '21)	12.2M	23.1	45.8	22.9	47.4	19.0	62.1	41.5	11.4
Omnidata v2 [22] (CVPR '22)	12.2M	17.2	55.5	**16.2**	60.2	18.2	63.9	40.5	14.7
DSINE [2] (CVPR '24)	161K	16.4	59.6	**16.2**	**61.0**	**17.1**	67.4	**34.9**	**21.5**
Metric3D v2 [21] (TPAMI '24)	16M	**13.3**	**66.4**	—†	—†	19.6	**69.7**	—	—
Marigold [24] (CVPR '24)	74K	18.8	55.9	17.7	58.8	18.4	64.3	39.1	14.9
GeoWizard [14] (ECCV '24)	278K	17.0	56.5	15.4	61.6	13.0	65.3	—	—
↳ reproduced by us	278K	19.1	49.5	17.3	53.7	19.5	61.6	40.4	13.2
GeoWizard + E2E FT	278K	**16.1**	60.7	15.3	63.6	16.2	69.4	**33.4**	**22.4**
Marigold + E2E FT	74K	16.2	**61.4**	**14.7**	66.0	**15.8**	**69.9**	33.5	21.5
Stable Diffusion [39] + E2E FT	74K	16.5	60.4	**14.7**	**66.1**	16.1	69.7	33.5	22.3

ter than both the fixed single-step model and the claimed previous best results with 50 steps and ensembling of 10 predictions, which we were not able to reproduce. For depth, we observe smaller, but consistent improvements. We provide more results in the supplementary materials.

Qualitative Results. Fig. 6 shows several qualitative examples. The single-step model fails to produce sharp results, while the multi-step ensemble method starts to hallucinate high frequency details. The end-to-end fine-tuned models predict sharp depth maps and high-quality normals.

State-of-the-art Landscape. As shown in Tab. 5, the fine-tuned models outperform current state-of-the-art generative methods for depth estimation on most datasets. Among discriminative methods, only Depth Anything [50, 51] and Metric3D [21, 53] demonstrate superior performance; however, these methods were trained on datasets that are two to three orders of magnitude larger. For surface normal estimation, the fine-tuned models set new state-of-the-art results across all evaluated datasets, with the exception of NYUv2, where Metric3D v2 continues to lead, as shown in Tab. 6.

7. Conclusion

We have shown that a critical flaw in the implementation of the DDIM scheduler causes several prior works to draw possibly wrong conclusions. We found simple end-to-end fine-tuning to outperform more complicated training pipelines and architectures. Nonetheless, our work supports the hypothesis that diffusion pretraining does provide excellent priors for geometric tasks such as monocular depth and normal estimation. The resulting models allow accurate single-step inference, enabling to profit from large-scale data using sophisticated self-training procedures as used in prior works [50, 51]. We believe that further improvements in diffusion models will lead to even more reliable priors, which might further improve the performance of this kind of geometry estimation models. We regard this as a promising avenue for future research.

Acknowledgements. Karim Abou Zeid's research is funded by the Bosch-RWTH LHC project "Context Understanding for Autonomous Systems". Christian Schmidt is funded by BMBF project bridgingAI (16DHBKI023). Computations were performed with computing resources granted by RWTH Aachen University under project rwth1690.

References

[1] Gwangbin Bae, Ignas Budvytis, and Roberto Cipolla. Estimating and exploiting the aleatoric uncertainty in surface normal estimation. In *ICCV*, 2021. 3

[2] Gwangbin Bae and Andrew J Davison. Rethinking inductive biases for surface normal estimation. In *CVPR*, 2024. 3, 8, 12

[3] Aayush Bansal, Bryan C. Russell, and Abhinav Kumar Gupta. Marr revisited: 2d-3d alignment via surface normal prediction. *CVPR*, 2016. 3

[4] 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

[5] Daniel J Butler, Jonas Wulff, Garrett B Stanley, and Michael J Black. A naturalistic open source movie for optical flow evaluation. In *ECCV*, 2012. 6, 7, 8, 12

[6] Yohann Cabon, Naila Murray, and Martin Humenberger. Virtual kitti 2. *arXiv preprint arXiv:2001.10773*, 2020. 5, 6, 11

[7] 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. 5, 6, 7, 8, 12, 13

[8] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR*, 2021. 2

[9] Yiqun Duan, Xianda Guo, and Zheng Zhu. DiffusionDepth: Diffusion denoising approach for monocular depth estimation. *arXiv preprint arXiv:2303.05021*, 2023. 3

[10] Ainaz Eftekhar, Alexander Sax, Jitendra Malik, and Amir Zamir. Omnidata: A scalable pipeline for making multi-task mid-level vision datasets from 3d scans. In *ICCV*, 2021. 2, 3, 8

[11] David Eigen and Rob Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *ICCV*, 2015. 3

[12] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. *NeurIPS*, 2014. 12

[13] Yi Feng, Bohuan Xue, Ming Liu, Qijun Chen, and Rui Fan. D2NT: A high-performing depth-to-normal translator. In *ICRA*, 2023. 11, 12

[14] 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. *ECCV*, 2024. 2, 3, 5, 7, 8, 12, 13

[15] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *CVPR*, 2012. 6, 7, 8, 12, 13

[16] Ming Gui, Johannes S. Fischer, Ulrich Prestel, Pingchuan Ma, Dmytro Kotovenko, Olga Grebenkova, Stefan Andreas Baumann, Vincent Tao Hu, and Björn Ommer. Depthfm: Fast monocular depth estimation with flow matching. *arXiv preprint arXiv:2403.13788*, 2024. 2, 3, 8, 12, 13

[17] Xiankang He, Guangkai Xu, Bo Zhang, Hao Chen, Ying Cui, and Dongyan Guo. Diffcalib: Reformulating monocular camera calibration as diffusion-based dense incident map generation. *arXiv preprint arXiv: 2405.15619*, 2024. 3

[18] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *NeurIPS*, 2020. 4

[19] Derek Hoiem, Alexei A. Efros, and Martial Hebert. Automatic photo pop-up. *SIGGRAPH*, 2005. 3

[20] Derek Hoiem, Alexei A. Efros, and Martial Hebert. Recovering surface layout from an image. *IJCV*, 2007. 3

[21] 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*, 2024. 2, 3, 8

[22] Oğuzhan Fatih Kar, Teresa Yeo, Andrei Atanov, and Amir Zamir. 3d common corruptions and data augmentation. In *CVPR*, 2022. 2, 3, 8

[23] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *NeurIPS*, 2022. 4

[24] 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, 3, 4, 5, 6, 7, 8, 11, 12, 13

[25] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 4

[26] Tobias Koch, Lukas Liebel, Friedrich Fraundorfer, and Marco Körner. Evaluation of cnn-based single-image depth estimation methods. In *ECCVW*, 2018. 6, 7, 8, 12

[27] Zhengqi Li and Noah Snavely. Megadepth: Learning single-view depth prediction from internet photos. In *CVPR*, 2018. 2

[28] Shanchuan Lin, Bingchen Liu, Jiashi Li, and Xiao Yang. Common diffusion noise schedules and sample steps are flawed. *WACV*, 2023. 2, 3, 5, 11

[29] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *ICLR*, 2023. 3

[30] Xiaoxiao Long, Yuan-Chen Guo, Cheng Lin, Yuan Liu, Zhiyang Dou, Lingjie Liu, Yuexin Ma, Song-Hai Zhang, Marc Habermann, Christian Theobalt, et al. Wonder3d: Single image to 3d using cross-domain diffusion. In *CVPR*, 2024. 3

[31] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019. 6

[32] Simian Luo, Yiqin Tan, Longbo Huang, Jian Li, and Hang Zhao. Latent consistency models: Synthesizing high-resolution images with few-step inference. *arXiv preprint arXiv:2310.04378*, 2023. 3, 7

[33] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. DINOv2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 2

[34] 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

[35] Lingteng Qiu, Guanying Chen, Xiaodong Gu, Qi Zuo, Mutian Xu, Yushuang Wu, Weihao Yuan, Zilong Dong, Liefeng Bo, and Xiaoguang Han. Richdreamer: A generalizable normal-depth diffusion model for detail richness in text-to-3d. In *CVPR*, 2024. 3

[36] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *ICCV*, 2021. 2, 8

[37] 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*, 2020. 2, 5, 6, 8

[38] Mike Roberts, Jason Ramapuram, Anurag Ranjan, Atulit Kumar, Miguel Angel Bautista, Nathan Paczan, Russ Webb, and Joshua M. Susskind. Hypersim: A photorealistic synthetic dataset for holistic indoor scene understanding. In *ICCV*, 2021. 5, 6, 11

[39] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 2, 3, 4, 6, 7, 8, 11, 12, 13

[40] Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. In *ICLR*, 2022. 4, 5

[41] Saurabh Saxena, Charles Herrmann, Junhwa Hur, Abhishek Kar, Mohammad Norouzi, Deqing Sun, and David J. Fleet. The surprising effectiveness of diffusion models for optical flow and monocular depth estimation. In *NeurIPS*, 2023. 3

[42] Saurabh Saxena, Abhishek Kar, Mohammad Norouzi, and David J Fleet. Monocular depth estimation using diffusion models. *arXiv preprint arXiv:2302.14816*, 2023. 3

[43] Thomas Schops, Johannes L Schonberger, Silvano Galliani, Torsten Sattler, Konrad Schindler, Marc Pollefeys, and Andreas Geiger. A multi-view stereo benchmark with high-resolution images and multi-camera videos. In *CVPR*, 2017. 6, 7, 8, 12, 13

[44] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *ECCV*, 2012. 5, 6, 7, 8, 12, 13

[45] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *ICLR*, 2021. 3, 4

[46] Igor Vasiljevic, Nick Kolkin, Shanyi Zhang, Ruotian Luo, Haochen Wang, Falcon Z Dai, Andrea F Daniele, Mohammadreza Mostajabi, Steven Basart, Matthew R Walter, et al. Diode: A dense indoor and outdoor depth dataset. *arXiv preprint arXiv:1908.00463*, 2019. 6, 7, 8, 12, 13

[47] X. Wang, David F. Fouhey, and Abhinav Kumar Gupta. Designing deep networks for surface normal estimation. *CVPR*, 2015. 3

[48] Jonathan Whitaker. Multi-resolution noise for diffusion model training. https://wandb.ai/johnowhitaker/multires_noise/reports/Multi-Resolution-Noise-for-Diffusion-Model-Training--VmlldzozNjYyOTU2 (2024-09-09), 2023. 4

[49] Guangkai Xu, Yongtao Ge, Mingyu Liu, Chengxiang Fan, Kangyang Xie, Zhiyue Zhao, Hao Chen, and Chunhua Shen. Diffusion models trained with large data are transferable visual models. *arXiv preprint arXiv:2403.06090v2*, 2024. 13

[50] 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, 8

[51] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *arXiv preprint arXiv:2406.09414*, 2024. 2, 8

[52] Wei Yin, Yifan Liu, and Chunhua Shen. Virtual normal: Enforcing geometric constraints for accurate and robust depth prediction. *IEEE TPAMI*, 2021. 2

[53] 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, 8

[54] Wei Yin, Jianming Zhang, Oliver Wang, Simon Niklaus, Long Mai, Simon Chen, and Chunhua Shen. Learning to recover 3d scene shape from a single image. In *CVPR*, 2021. 2, 8

[55] Chi Zhang, Wei Yin, Zhibin Wang, Gang Yu, Bin Fu, and Chunhua Shen. Hierarchical normalization for robust monocular depth estimation. *NeurIPS*, 2022. 2, 8

[56] Wenliang Zhao, Yongming Rao, Zuyan Liu, Benlin Liu, Jie Zhou, and Jiwen Lu. Unleashing text-to-image diffusion models for visual perception. In *ICCV*, 2023. 3

[57] Shengjie Zhu, Abhinav Kumar, Masa Hu, and Xiaoming Liu. Tame a wild camera: In-the-wild monocular camera calibration. In *NeurIPS*, 2023. 3

Fine-Tuning Image-Conditional Diffusion Models is Easier than You Think

Supplementary Material

A. DDIM Inference

During training, the highest noise level corresponds to the last timestep $t = T$, and $t = 1$ corresponds to a very small noise level. The DDIM inference scheduler iterates over a series of k timesteps $\tau_1 > \tau_2 > \ldots > \tau_k > 0$ and iteratively denoises the initial noise input $\mathbf{z}_{\tau_1}$. We consider the `leading` and `trailing` schedules that are also discussed by Lin *et al*. [28] and show the selected timesteps for different k in Tab. A-1. The original `leading` timestep selection strategy of the DDIM scheduler excludes the final timestep T. This leads to a mismatch between training and inference; using the `leading` schedule, the model receives noise as input, even though the timestep embedding indicates a partially denoised input. In contrast, the fixed `trailing` strategy always starts with $t = T$ for the first denoising step, properly aligning training and inference. In the limit of $k \to T$ inference steps, both strategies converge to the same behavior.

In Fig. A-1, we illustrate the difference between single-step predictions using the broken `leading` and the fixed `trailing` DDIM scheduler for Marigold [24] and Stable Diffusion [39]. Both models output noise when using the broken scheduler. With the fixed implementation, both models predict the mean of their respective conditional distribution. For single-step Marigold this results in a well-defined depth map, whereas for single-step Stable Diffusion, it produces a blurry image with coarse structures that roughly align with the input prompt.

Fig. A-3 further demonstrates the scheduler's impact when multiple steps are considered. It clearly shows that the effect of the broken scheduler becomes less noticeable as the number of inference steps increases. Additionally, the weak text conditioning in Stable Diffusion leads to blurry images, which gradually sharpen as more inference steps are taken. In contrast, the strong image conditioning in Marigold allows the model to predict reasonably accurate depth maps already in the first step. As shown by the heatmap in Fig. 2b in the main text, subsequent steps only lead to small changes in the predicted distances, and most of the scene remains unchanged.

B. Detailed Experimental Setup

Training Datasets. For a direct comparison with Marigold [24], we use the same synthetic training datasets offering high quality ground-truth annotations, *i.e*., Hypersim [38] and Virtual KITTI 2 [6].

Figure A-1. **Single-step outputs of Marigold and Stable Diffusion.** With a single step, Stable Diffusion produces a blurry image at best, while Marigold outputs a sensible depth map. Note that the input prompt is text for Stable Diffusion, but an RGB image for Marigold.

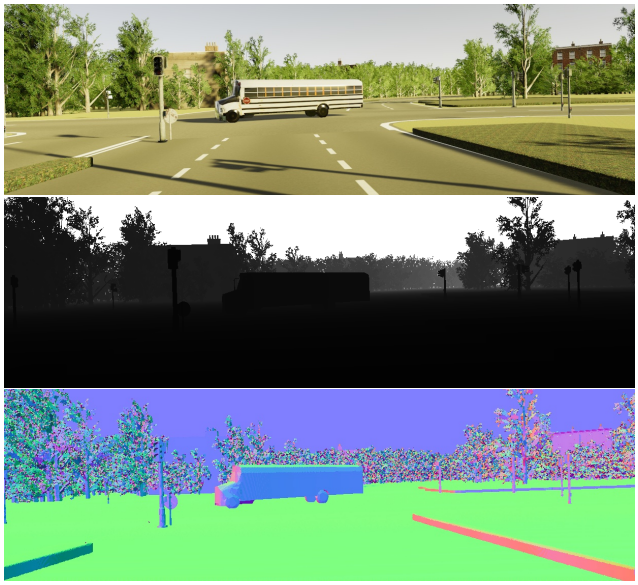


Figure A-2. **Virtual KITTI 2 example.** Top: Synthetic RGB image. Middle: Ground-truth depth map. Bottom: Ground-truth surface normals, generated using discontinuity-aware gradient filters [13].

Hypersim consists of 54K photorealistic images from 365 indoor scenes, which we resize to a resolution of 480×640 with a far plane at 65 meters. Virtual KITTI 2 contains approximately 20K samples from four synthetic driving scenarios under various weather conditions. These images are cropped to 352×1216 pixels, and the far plane is set to 80 meters.

Table A-1. **Comparison of `leading` vs. `trailing` timestep selection.** The timesteps selected by two DDIM scheduler timestep selection strategies for $T = 1000$ timesteps and varying numbers of inference steps.

Inference Steps	`leading` timestep selection	`trailing` timestep selection
1	[1]	[1000]
2	[501, 1]	[1000, 500]
4	[751, 501, 251, 1]	[1000, 750, 500, 250]
10	[901, 801, 701, 601, 501, 401, 301, 201, 101, 1]	[1000, 900, 800, 700, 600, 500, 400, 300, 200, 100]

Since Virtual KITTI 2 does not provide annotations for surface normals, we compute them ourselves with the ground-truth depth maps, employing discontinuity-aware gradient filters from [13]. A qualitative example of the resulting normals can be seen in Fig. A-2.

Data Preprocessing. Following Marigold's approach for depth estimation, we remove outliers, *i.e.*, values below the 2^{nd} percentile and above the 98^{th} percentile, and normalize the depth map to the range $[-1, 1]$. Then, we repeat the normalized depth map 3 times along the color channel to match the VAE encoder's expected input shape. Normals, on the other hand, can be encoded directly since they are already in the desired range of $[-1, 1]$ and match the number of channels. The only data augmentation we utilize is random horizontal flipping.

Training Details. We mask out undefined depth values in the Hypersim dataset, and pixels surpassing the far plane for Virtual KITTI 2. When training Marigold for normal prediction as a diffusion estimator, the mask is downsampled by a factor of 8 to match the latent resolution. Thus, we neither enforce nor supervise undefined regions. For the end-to-end fine-tuning of GeoWizard, both the scale and shift invariant depth loss and the angular loss are optimized jointly. Scaling the depth loss by a factor of 0.5 roughly ensures equal magnitude.

Evaluation Datasets. For monocular depth estimation, we follow the evaluation strategy of Marigold and evaluate on commonly used benchmarks. NYUv2 [44] and ScanNet [7] provide RGB-D data of indoor environments captured with Kinect cameras. We use the official NYUv2 test split, consisting of 654 instances, while for ScanNet, Marigold's set of 800 randomly sampled images from the 312 validation scenes [24] is employed. ETH3D [43] and DIODE [46] offer high-resolution depth data for both indoor and outdoor scenes, derived from LiDAR sensors. We evaluate on all 454 samples in ETH3D and on DIODE's validation set, comprising 325 indoor and 446 outdoor examples. For KITTI [15], consisting of outdoor driving scenes captured by vehicle-mounted cameras and LiDAR sensors, the Eigen test split [12] is used, containing 652 images.

Regarding surface normal estimation we utilize the official DSINE [2] evaluation pipeline and data, comprised of the NYUv2 test split, 300 ScanNet [44] samples, the full iBims-1 [26] dataset, which is a small high-quality RGB-D dataset of 100 samples, and Sintel [5], made up of 1064 synthetic outdoor examples derived from an open-source 3D animated short film.

Table A-2. **Frozen vs. fine-tuned VAE decoder.** We conduct end-to-end fine-tuning of Marigold [24] for depth estimation, and assess the effect of freezing or fine-tuning the weights of the pretrained VAE decoder.

Decoder	NYUv2 [44]		KITTI [15]		ETH3D [43]		ScanNet [7]		DIODE [46]	
	AbsRel↓	$\delta1$↑	AbsRel↓	$\delta1$↑	AbsRel↓	$\delta1$↑	AbsRel↓	$\delta1$↑	AbsRel↓	$\delta1$↑
Frozen	**5.2**	**96.6**	**9.6**	**91.9**	**6.2**	95.9	**5.8**	**96.2**	**30.2**	**77.9**
Fine-tuned	5.3	96.5	**9.6**	**91.9**	**6.2**	**96.0**	**5.8**	96.1	**30.2**	77.7

Evaluation Details. For most existing methods in Tab. 5 and Tab. 6 we obtain the performance metrics either from the papers introducing these methods or from the Marigold and DSINE papers. The missing scores, like those of the newer GeoWizard [14] and DepthFM [16] models, are obtained by reevaluating the respective models with their official inference code and released checkpoints. In the case of DepthFM, the prediction alignment with respect to the ground-truth metric depth happens in the log metric space.

C. Additional Results

GeoWizard for Depth Estimation. GeoWizard [14] jointly predicts depth and surface normals, using a similar training and evaluation setup as Marigold. We find that GeoWizard suffers from the same flaw in the DDIM implementation as Marigold, and end-to-end fine-tuning the model for depth and normal estimation significantly boosts the performance (see Tab. A-3 and Tab. 3 in the main text). In particular, the fine-tuned model performs better than both the fixed single-step model and the previously best reported results with 50 steps and ensembling of 10 predictions.

Further Comparisons to DepthFM. DepthFM [16] proposes a direct mapping from input images to depth maps through flow matching, leveraging Stable Diffusion v2 [39] as a prior. We observe that, apart from the ETH3D $\delta1$ and DIODE [46] metrics, a simpler approach like E2E FT achieves better performance with a more than $10\times$ speedup as seen in Tab. A-4.

Fine-Tuning the VAE Decoder. By default, we keep the pretrained VAE decoder frozen while conducting end-to-

Table A-3. **Fixed DDIM scheduler and end-to-end fine-tuning (E2E FT) for GeoWizard's [14] depth estimation.** We use the official code and model weights to re-evaluate the method on all datasets. Inference time is for a single 576×768-pixel image, evaluated on an NVIDIA RTX 4090 GPU. We obtain significant speed-ups, improving results.

Method	Steps	Ensemble	Inference time	NYUv2 [44]		KITTI [15]		ETH3D [43]		ScanNet [7]		DIODE [46]	
				AbsRel↓	$\delta1$↑	AbsRel↓	$\delta1$↑	AbsRel↓	$\delta1$↑	AbsRel↓	$\delta1$↑	AbsRel↓	$\delta1$↑
GeoWizard [14]	50	10	72 s	5.2	96.6	9.7	92.1	6.4	96.1	6.1	95.3	29.7	79.2
↳ reproduced by us	50	10	72 s	5.7	**96.2**	14.4	82.0	7.5	94.3	6.1	95.8	31.4	77.1
GeoWizard + DDIM fix	1	1	**254** ms	5.8	96.1	13.3	84.7	7.8	94.3	6.2	95.7	32.0	76.0
GeoWizard + E2E FT	1	1	**254** ms	**5.6**	96.1	**9.8**	**91.4**	**6.3**	**95.7**	**5.9**	**96.2**	**30.6**	**77.9**

Table A-4. **Comparison of DepthFM [16] with the DDIM-fixed and end-to-end fine-tuned (E2E FT) Marigold and Stable Diffusion models.** We re-evaluated DepthFM [16] on all datasets using the official code and model weights, with 4 inference steps and an ensemble size of 6. Inference time is for a single 576×768-pixel image, evaluated on an NVIDIA RTX 4090 GPU.

Method	Steps	Ensemble	Inference time	NYUv2 [44]		KITTI [15]		ETH3D [43]		ScanNet [7]		DIODE [46]	
				AbsRel↓	$\delta1$↑	AbsRel↓	$\delta1$↑	AbsRel↓	$\delta1$↑	AbsRel↓	$\delta1$↑	AbsRel↓	$\delta1$↑
DepthFM [16]	4	6	1.67 s	6.5	95.6	8.3	93.4	—	—	—	—	22.5	80.0
↳ reproduced by us	4	6	1.67 s	6.9	95.4	11.4	88.1	6.5	**96.2**	8.1	92.5	**25.0**	78.3
DepthFM	1	1	132 ms	7.5	95.0	11.6	87.5	6.7	96.0	8.3	92.3	25.3	**77.9**
Marigold [24] + E2E FT	1	1	**121** ms	**5.2**	**96.6**	**9.6**	91.9	**6.2**	95.9	**5.8**	96.2	30.2	**77.9**
Stable Diffusion [39] + E2E FT	1	1	**121** ms	5.4	96.5	**9.6**	**92.1**	6.4	95.9	**5.8**	**96.5**	30.3	77.6

end fine-tuning. Tab. A-2 shows that fine-tuning the weights of this decoder does not improve performance.

Further Qualitative Samples. Fig. A-4 and Fig. A-5 show qualitative results for depth and normals estimation, respectively, comparing Marigold [24] and the end-to-end fine-tuned models. The fixed single-step model fails to produce sharp results, while the multi-step model exhibits noticeable over-sharpening and high-frequency noise artifacts (even after ensembling), particularly in the normals estimations. In contrast, the end-to-end fine-tuned models do not exhibit these issues.

Addendum

We were made aware of recent work by Xu *et al*. [49]. Similar to us, they directly fine-tune Stable Diffusion in an end-to-end fashion, however, we arrive to this point in a very different way. We initially discovered the issue with the DDIM scheduler, fixed this in Marigold, and in turn arrived to an end-to-end fine-tuning scheme that works for Marigold. Surprisingly, our ablations showed that this also works well for direct fine-tuning of Stable Diffusion. The main contribution of Xu *et al*. is an approach to fine-tune Stable Diffusion (for a broader spectrum of tasks). However, even with additional modules on top, their method achieves lower scores than some of the baselines. As such, these results might lead one to conclude that end-to-end fine-tuning is not a suitable alternative to multi-step, diffusion-based depth and normal estimation. In contrast, our simple end-to-end fine-tuning setup *does* outperform diffusion baselines, demonstrating that it is an effective and efficient alternative.

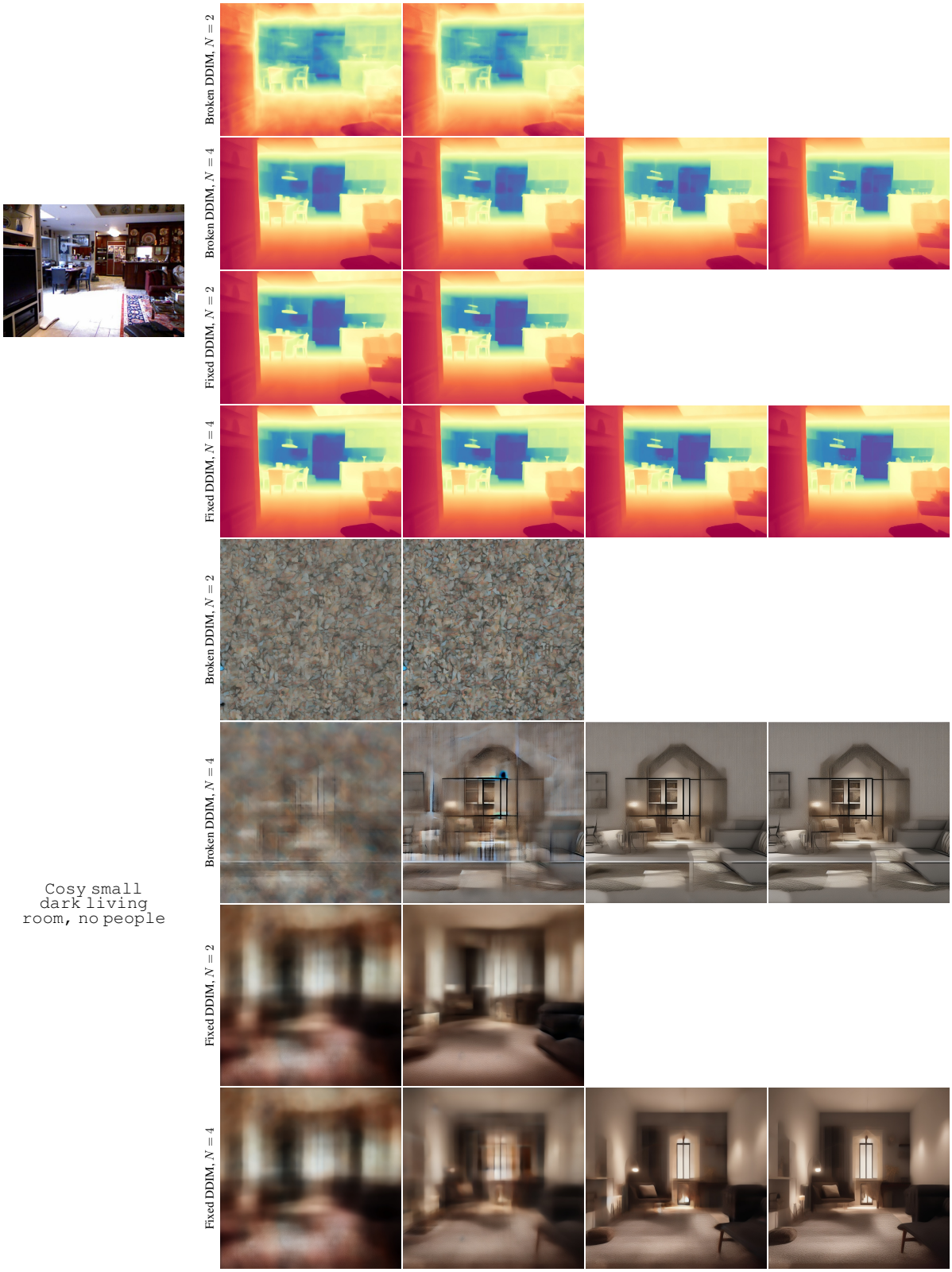


Figure A-3. **Few-step inference of Marigold and Stable Diffusion.** With more steps, the adverse effects of the broken DDIM scheduler get less pronounced. Both Marigold and Stable Diffusion produce sharper outputs with more steps, but the difference is much greater for Stable Diffusion.

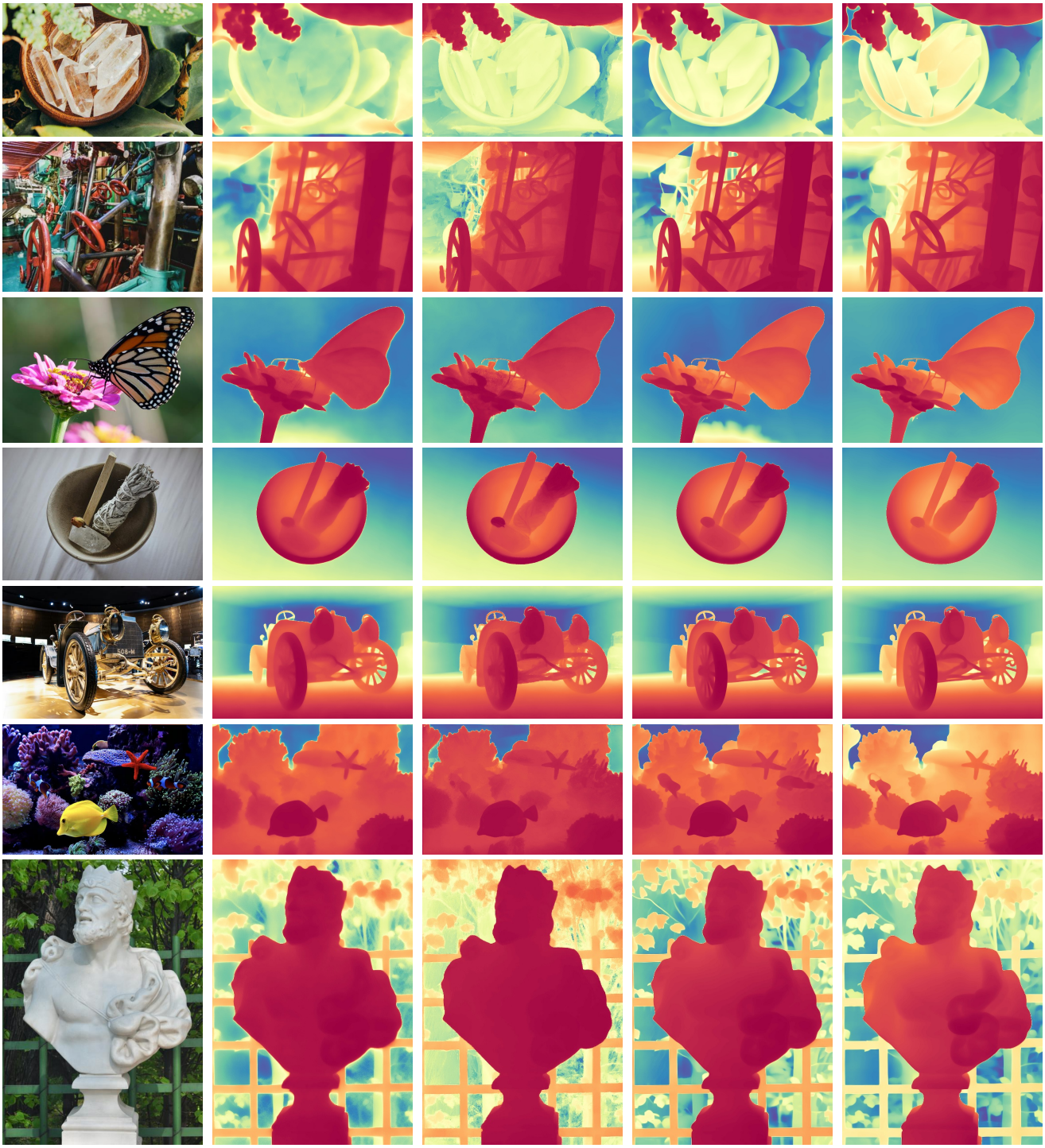


Figure A-4. **Additional qualitative samples for depth estimation.** "Marigold (X, Y)" denotes Marigold using X inference steps with an ensemble of size Y.

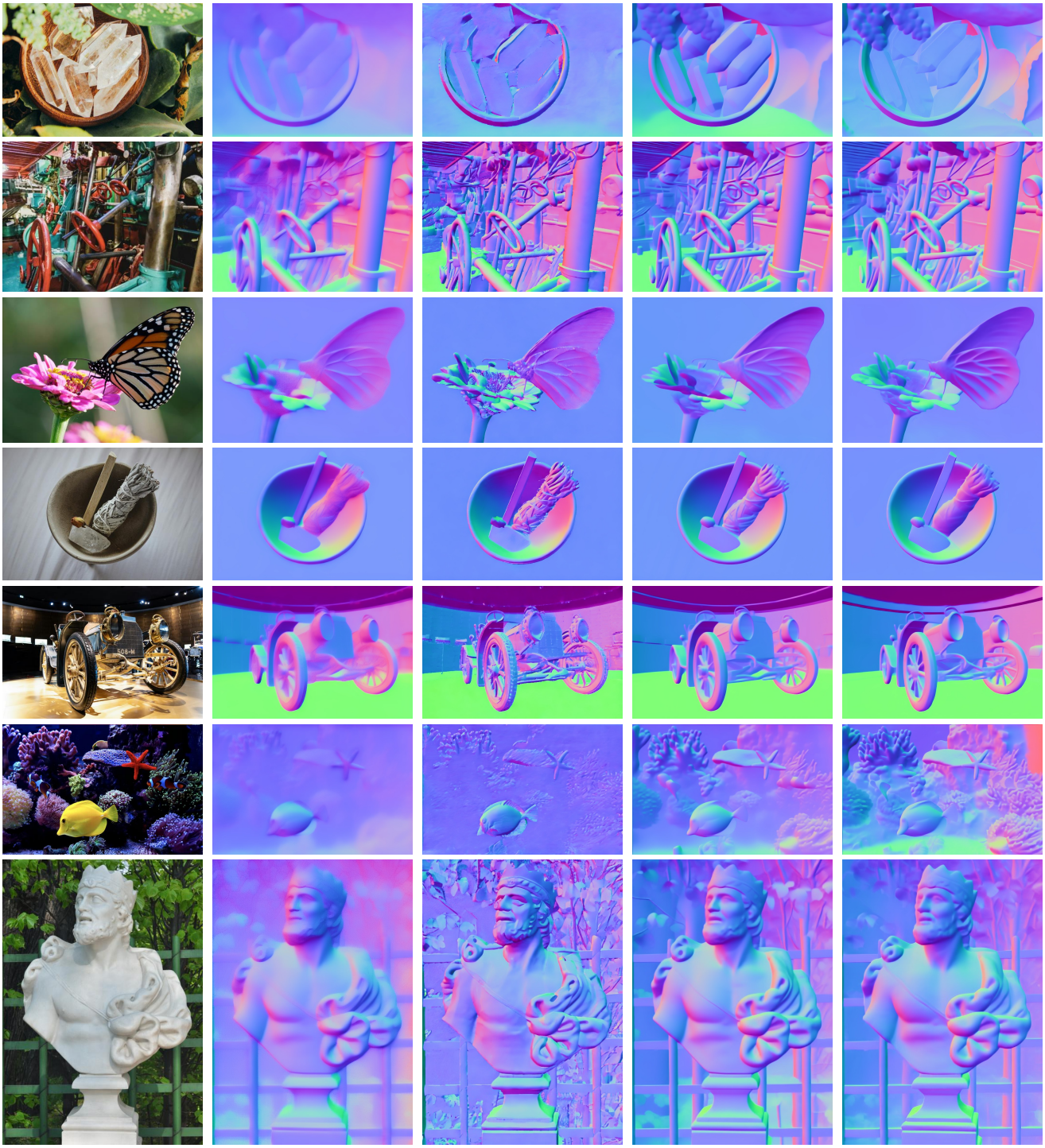


Figure A-5. **Additional qualitative samples for normal estimation.** "Marigold (X, Y)" denotes Marigold using X inference steps with an ensemble of size Y.