Improving Repurposing Diffusion-Based Image Generators for Monocular Depth Estimation

Ridin Datta, 210840

EE798R

Abstract

Monocular depth estimation, transforming a 2D image to a depth map, remains a challenging task. Marigold, a diffusion model built on Stable Diffusion, leverages visual knowledge for zero-shot depth estimation on unseen datasets. Fine-tuned with synthetic data, Marigold demonstrates substantial improvements in monocular depth estimation across a wide variety of datasets.

1 Introduction

Monocular depth estimation aims to produce a depth map from a single RGB image, requiring prior scene understanding due to the inherent ambiguity in translating 2D to 3D. Advances in deep learning have led to significant improvements, yet generalization to novel scenes remains challenging. Leveraging diffusion models' comprehensive visual priors, Marigold builds upon Stable Diffusion to provide a broadly applicable, state-of-the-art monocular depth estimator.

2 Related Work

2.1 Monocular Depth Estimation

This field has evolved with methods from CNNs to transformers, achieving generalization through large, varied datasets. Techniques for estimating affine-invariant depth, which do not require known camera intrinsics, are increasingly popular for in-the-wild depth estimation.

2.2 Diffusion Models

Diffusion models reverse a noise diffusion process, with latent diffusion models (LDMs) using a compact latent space. Conditional diffusion, used in Marigold, enables depth estimation by conditioning on input images while retaining essential image priors.

3 Methodology

In this section I outline the approach of Marigold for monocular depth estimation, including the generative diffusion framework, network architecture adaptations, and fine-tuning protocol.

3.1 Generative Formulation

Marigold leverages a denoising diffusion probabilistic model (DDPM) to predict depth maps conditioned on RGB images. The goal is to model the conditional distribution $D(d|x)$, where d is the depth map and x is the RGB image. This process uses a forward and reverse procedure in a diffusion model framework:

- **Forward Process:** Gaussian noise is progressively added to a clean depth map d over T steps. At each time step t, a noisy depth sample d_t is created using:

  $$d_t = \sqrt{\bar{\alpha}_t} d_0 + \sqrt{1 - \bar{\alpha}_t}\epsilon,$$

  where $\epsilon \sim \mathcal{N}(0, I)$ is Gaussian noise, and $\bar{\alpha}_t$ is the product of the variance schedule parameters.

- **Reverse Process:** The model trains a denoising function ϵ_θ to iteratively reverse the noise, reconstructing d from d_T. The learning objective minimizes the denoising error at each timestep t using:

  $$L = \mathbb{E}_{d_0, \epsilon \sim \mathcal{N}(0,I), t \sim U(T)} \left[\|\epsilon - \epsilon_\theta(d_t, x, t)\|_2^2 \right].$$

  At inference, starting from a random noise distribution, depth is reconstructed by progressively denoising back to d_0, yielding the final depth map.

3.2 Network Architecture

Marigold's architecture is based on Stable Diffusion's latent diffusion model (LDM) but is adapted to condition on images. It consists of a VAE for encoding and decoding, along with a U-Net for denoising in the latent space.

- **VAE Encoder and Decoder:** Both RGB and depth maps are encoded to a latent space using a pretrained VAE. Since the encoder expects a 3-channel RGB input, single-channel depth maps are replicated to match this structure, ensuring compatibility without altering the VAE.

- **Conditioning in Latent Space:** Marigold conditions depth prediction on RGB images by concatenating the image and depth latent codes. These are fed into the U-Net, which has been modified to accept the concatenated latent codes. The input channels are doubled, and weights are duplicated and scaled to preserve pretrained model stability.

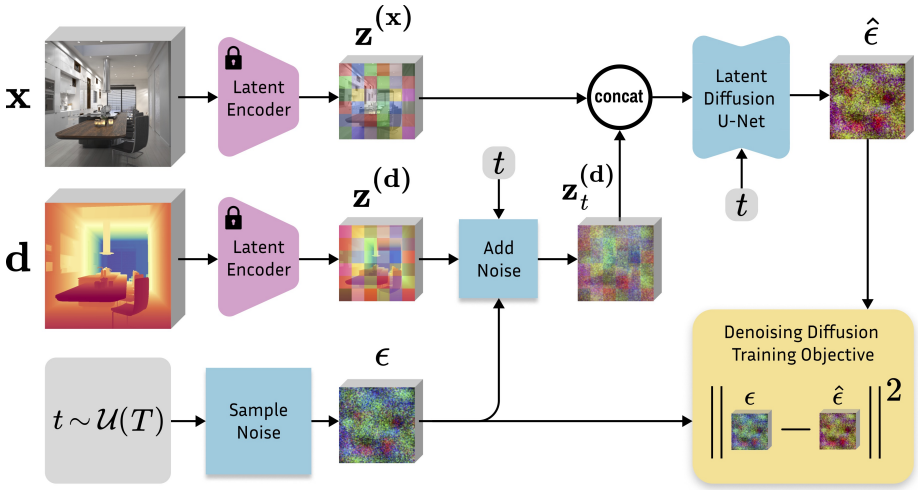

Figure 1: Fine-Tuning Protocol in Marigold

3.3 Fine-Tuning Protocol

The fine-tuning strategy adapts the pretrained Stable Diffusion model to depth estimation with minimal computational resources.

- **Affine-Invariant Depth Normalization:** To ensure compatibility across diverse scene depths, depth values are normalized using affine-invariant scaling:

$$\tilde{d} = \frac{d - d_2}{d_{98} - d_2} \times 2 - 1,$$

  where d_2 and d_{98} are the 2% and 98% depth percentiles, respectively. This transforms all scenes to a common depth range, allowing robust generalization.

- **Training on Synthetic Data:** Marigold is trained solely on synthetic datasets, specifically Hypersim and Virtual KITTI, which provide dense, accurate depth maps. Synthetic data reduces ground truth noise and ensures complete depth information, essential for VAE-based encoding.

- **Multi-Resolution Noise and Annealing:** The training process incorporates multi-resolution noise, combining Gaussian noise at different scales to encourage detail retention. An annealing strategy progressively reduces noise levels across training steps, leading to faster convergence and enhanced depth estimation quality.

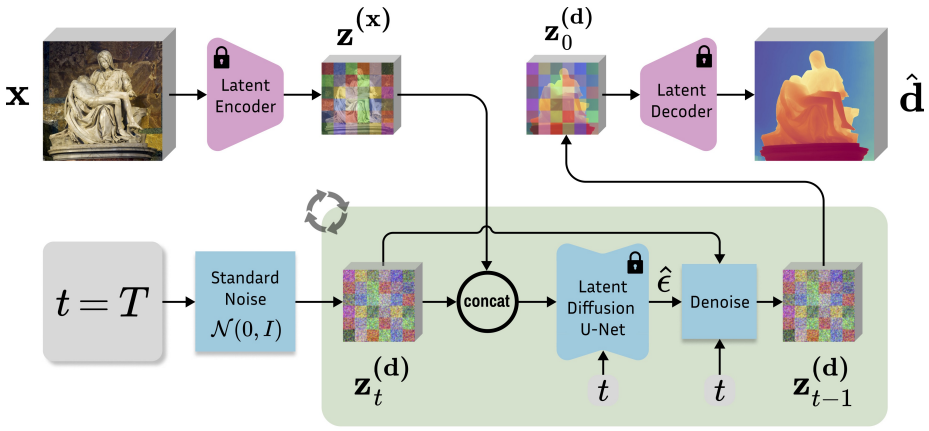

Figure 2: Inference Scheme in Marigold

3.4 Inference

At test time, Marigold encodes the RGB input into latent space, initializes the depth as Gaussian noise, and applies the learned denoising schedule to reconstruct the depth map. The denoising process is enhanced by test-time ensembling:

- **Test-Time Ensembling:** By running the inference multiple times with different noise initializations and averaging results, Marigold improves depth estimation consistency. The affine-invariant predictions are aligned using scale and shift factors before ensembling.

Through this architecture and training design, Marigold achieves state-of-the-art monocular depth estimation performance across varied datasets without extensive real-world data training.

3.5 Multi-Resolution Noise

Multi-resolution noise is applied by layering Gaussian noise at different scales, effectively preserving spatial details across scales. This approach is beneficial in depth estimation, where fine-grained details (like edges) are crucial. During training, multi-resolution noise is added to the depth maps, encouraging the model to denoise effectively across a variety of scales, which results in better generalization to real-world depth maps.

3.6 Justification for Latent Diffusion Models (LDMs)

Latent Diffusion Models (LDMs) are computationally efficient because they perform the diffusion process in a compressed latent space rather than the pixel

Method	# Training samples		NYUv2		KITTI		ETH3D		ScanNet		DIODE		Avg. Rank
	Real	Synthetic	AbsRel↓	$\delta1$↑	AbsRel↓	$\delta1$↑	AbsRel↓	$\delta1$↑	AbsRel↓	$\delta1$↑	AbsRel↓	$\delta1$↑	
DiverseDepth [54]	320K	—	11.7	87.5	19.0	70.4	22.8	69.4	10.9	88.2	37.6	63.1	6.6
MiDaS [33]	2M	—	11.1	88.5	23.6	63.0	18.4	75.2	12.1	84.6	33.2	71.5	6.3
LeReS [55]	300K	54K	9.0	91.6	14.9	78.4	17.1	77.7	9.1	91.7	27.1	76.6	4.3
Omnidata [11]	11.9M	310K	7.4	94.5	14.9	83.5	16.6	77.8	7.5	93.6	33.9	74.2	3.8
HDN [58]	300K	—	6.9	94.8	11.5	86.7	12.1	83.3	8.0	93.9	24.6	**78.0**	2.4
DPT [34]	1.2M	188K	9.8	90.3	10.0	90.1	7.8	94.6	8.2	93.4	**18.2**	75.8	3.1
Marigold (ours)	—	74K	**5.5**	**96.4**	**9.9**	**91.6**	**6.5**	**96.0**	**6.4**	**95.1**	30.8	77.3	**1.4**

Figure 3: Comparison with other methods

space. This reduced dimensionality is ideal for high-resolution depth estimation, as it requires fewer resources to achieve high-quality results. Additionally, the LDM latent space has been shown to capture semantically meaningful representations, allowing Marigold to leverage this feature for accurate depth predictions.

4 Experiments

4.1 Datasets and Evaluation

Marigold is trained on synthetic datasets and tested on real datasets like NYUv2, KITTI, and ETH3D. It shows robust zero-shot performance, outperforming several state-of-the-art baselines in affine-invariant depth accuracy.

4.2 Ablation Studies

Several ablation studies highlight the impact of multi-resolution noise, the choice of synthetic datasets, and the effectiveness of test-time ensembling. Marigold benefits from increased ensemble size, which improves prediction accuracy.

5 Comparison

Quantitative comparison of Marigold with SOTA affine-invariant depth estimators on several zero-shot benchmarks. All metrics are presented in percentage terms; bold numbers are the best, underscored second best. Marigold's method outperforms other methods on both indoor and outdoor scenes in most cases, without ever seeing a real depth sample.

6 Challenges and Limitations

While Marigold achieves high performance, it currently relies on synthetic data for training, which may not fully capture the intricacies of real-world data. Furthermore, the model's dependency on fine-tuning pre-trained LDMs may

limit its applicability to models with similar architectures. Future work includes addressing these limitations to enhance generalization.

7 Applications

Marigold's robust monocular depth estimation can be applied in various fields, including:

- **Autonomous Driving**: Enabling accurate depth estimation in real-time without expensive LiDAR systems.
- **Augmented Reality (AR)**: Providing depth perception to enable realistic object placement and interaction in AR applications.
- **Robotics**: Assisting robots in depth-aware navigation and manipulation in unstructured environments.

8 My Improvements

8.1 Overview

The goal of my approach is to implement a dynamic mechanism that determines the number of denoising steps T based on the complexity C of the input image. For complex images, more denoising steps are retained, whereas simpler images use fewer steps, reducing computational costs while maintaining depth estimation accuracy.

8.2 1. Image Complexity Metric

Let x denote the input image, and ∇x its gradient map. We define the image complexity C based on the average gradient magnitude or gradient variance, representing the image's structural complexity:

$$C = \mathbb{E}\left[\|\nabla x\|^2\right] \tag{1}$$

Alternatively, complexity C can be approximated using Shannon entropy $H(x)$:

$$C = H(x) = -\sum_{i=1}^{N} p(x_i) \log p(x_i) \tag{2}$$

where $p(x_i)$ is the probability distribution of pixel intensities in x.

8.3 2. Adaptive Denoising Step Selection Function

Given the complexity C, we define the denoising steps $T(C)$ as a function of C that scales linearly between minimum $T_{\min}$ and maximum $T_{\max}$ steps, based on observed complexity values:

$$T(C) = T_{\min} + \alpha(C - C_{\min}) \tag{3}$$

where:

- $T_{\min}$ and $T_{\max}$ are the minimum and maximum allowable denoising steps,
- $C_{\min}$ and $C_{\max}$ represent the minimum and maximum complexity values over the dataset,
- $\alpha = \frac{T_{\max} - T_{\min}}{C_{\max} - C_{\min}}$ is a scaling factor.

This scaling function ensures:

- $T(C) = T_{\min}$ when $C = C_{\min}$,
- $T(C) = T_{\max}$ when $C = C_{\max}$.

8.4 3. Denoising Process with Adaptive Steps

For each denoising step $t \in \{1, \ldots, T(C)\}$, we apply the conditional denoising function ϵ_θ on the latent variable z as follows:

$$z_{t-1} = z_t - \alpha_t \epsilon_\theta(z_t, x, t) \tag{4}$$

where α_t is a noise schedule parameter.

8.5 4. Efficiency Gains

By adapting $T(C)$ based on image complexity, I can reduce the number of denoising steps for simpler images, resulting in significant computational savings without compromising depth estimation accuracy. Complex scenes retain more denoising steps for finer details, while simpler scenes achieve faster processing with fewer steps.

References

- Ke, B., Obukhov, A., Huang, S., Metzger, N., Caye Daudt, R., Schindler, K. *Repurposing Diffusion-Based Image Generators for Monocular Depth Estimation*, CVPR 2024.