Lite-Mono: A Lightweight CNN and Transformer Architecture for Self-Supervised Monocular Depth Estimation

Ning Zhang* Francesco Nex George Vosselman Norman Kerle
University of Twente
{n.zhang, f.nex, george.vosselman, n.kerle}@utwente.nl

Abstract

Self-supervised monocular depth estimation that does not require ground truth for training has attracted attention in recent years. It is of high interest to design lightweight but effective models so that they can be deployed on edge devices. Many existing architectures benefit from using heavier backbones at the expense of model sizes. This paper achieves comparable results with a lightweight architecture. Specifically, the efficient combination of CNNs and Transformers is investigated, and a hybrid architecture called Lite-Mono is presented. A Consecutive Dilated Convolutions (CDC) module and a Local-Global Features Interaction (LGFI) module are proposed. The former is used to extract rich multi-scale local features, and the latter takes advantage of the self-attention mechanism to encode long-range global information into the features. Experiments demonstrate that Lite-Mono outperforms Monodepth2 by a large margin in accuracy, with about 80% fewer trainable parameters. Our codes and models are available at https://github.com/noahzn/Lite-Mono.

1. Introduction

Many applications in the field of robotics, autonomous driving, and augmented reality rely on depth maps, which represent the 3D geometry of a scene. Since depth sensors increase costs, research on inferring depth maps using Convolutional Neural Networks (CNNs) from images emerged. With the annotated depth one can train a regression CNN to predict the depth value of each pixel on a single image [10, 11, 22]. Lacking large-scale accurate dense ground-truth depth for supervised learning, self-supervised methods that seek supervisory signals from stereo-pairs of frames or monocular videos are favorable and have made great progress in recent years. These methods regard the depth estimation task as a novel view synthesis problem and minimize an image reconstruction loss [5, 14, 15, 41, 45].

*Corresponding Author
global contexts into the features. To reduce the computational complexity the cross-covariance attention [1] is calculated in the channel dimension instead of the spatial dimension. The contributions of this paper can be summarized in three aspects.

- A new lightweight architecture, dubbed Lite-Mono, for self-supervised monocular depth estimation, is proposed. Its effectiveness with regard to the model size and FLOPs is demonstrated.

- The proposed architecture shows superior accuracy on the KITTI [13] dataset compared with competitive larger models. It achieves state-of-the-art with the least trainable parameters. The model’s generalization ability is further validated on the Make3D [32] dataset. Additional ablation experiments are conducted to verify the effectiveness of different design choices.

- The inference time of the proposed method is tested on an NVIDIA TITAN Xp and a Jetson Xavier platform, which demonstrates its good trade-off between model complexity and inference speed.

The remainder of the paper is organized as follows. Section 2 reviews some related research work. Section 3 illustrates the proposed method in detail. Section 4 elaborates on the experimental results and discussion. Section 5 concludes the paper.

2. Related work

2.1. Monocular depth estimation using deep learning

Single image depth estimation is an ill-posed problem, because a 2D image may correspond to many 3D scenes at different scales. Methods using deep learning can be roughly divided into two categories.

Supervised depth estimation. Using ground-truth depth maps as supervision, a supervised deep learning network is able to extract features from input images and learn the relationship between depth and RGB values. Eigen et al. [10] first used deep networks to estimate depth maps from single images. They designed a multi-scale network to combine global coarse depth maps and local fine depth maps. Subsequent works introduced some post-processing techniques, such as Conditional Random Fields (CRF), to improve the accuracy [24,25,38]. Laina et al. [22] proposed to use a new up-sampling module and the reverse Huber loss to improve the training. Fu et al. [11] adopted a multi-scale network, and treated the depth estimation as an ordinal regression task. Their method achieved higher accuracy and faster convergence.

Self-supervised depth estimation. Considering that large-scale annotated datasets are not always available, self-supervised depth estimation methods that do not require ground truth for training have attracted some attention. Garg et al. [12] regarded the depth estimation as a novel view synthesis problem, and proposed to minimize a photometric loss between an input left image and the synthesized right image. Their method was self-supervised, as the supervisory signal came from the input stereo pairs. Godard et al. [14] extended this work and achieved higher accuracy by introducing a left-right disparity consistency loss. Apart from using stereo pairs the supervisory signal can also come from monocular video frames. Zhou et al. [45] trained a separate multi-view pose network to estimate the pose between two sequential frames. To improve the robustness when dealing with occlusion and moving objects they also used an explainability prediction network to ignore target pixels that violate view synthesis assumptions. To model dynamic scenes other works introduced multi-task learning, such as optical flow estimation [41] and semantic segmentation [5,20], or introduced additional constraints, such as uncertainty estimation [30,40]. Godard et al. [15] found that without introducing extra learning tasks they could achieve competitive results by simply improving the loss functions. They proposed Monodepth2, which used a minimum reprojection loss to mitigate occlusion problems, and an auto-masking loss to filter out moving objects that have the same velocity as the camera. This work is also based on their self-supervised training strategy.

2.2. Advanced architectures for depth estimation

Network architectures also play an important role in achieving good results in monocular depth estimation. By replacing the network architecture from the VGG model [33] with a ResNet [17] Yin et al. [41] achieved better results. Yan et al. [39] used a channel-wise attention module to capture long-range multi-level information and enhance local features. Zhou et al. [44] also used an attention module to obtain a better feature fusion. Zhao et al. [46] proposed a small architecture that used a feature modulation module to learn multi-scale features, and demonstrated the method’s superiority. To reduce model parameters they only used the first three stages of ResNet18 [17] as the backbone. With the rise of vision transformer (ViT) [8] recent work applied it to various computer vision tasks [4,16,29,31,34], and achieved promising results. However, research incorporating Transformers in depth estimation architectures is still limited. Varma et al. [35] adopted the Dense Prediction Transformer [31] for self-supervised monocular depth estimation, and added another prediction head to estimate the camera’s intrinsic. Bae et al. [3] proposed a hybrid architecture of CNN and Transformer that enhanced CNN features by Transformers. However, due to the high computational complexity of Multi-Head Self-Attention (MHSA) in a ViT, the above-mentioned Transformer-based methods have more
3. The proposed framework: Lite-Mono

3.1. Design motivation and choices

Several papers demonstrated that a good encoder can extract more effective features, thus improving the final result [15, 17, 44]. This paper focuses on designing a lightweight encoder that can encode effective features from the input images. Figure 2 shows the proposed architecture. It consists of an encoder-decoder DepthNet (Section 3.2) and a PoseNet (Section 3.3). The DepthNet estimates multi-scale inverse depth maps of the input images, and the PoseNet estimates the camera motion between two adjacent frames. Then, a reconstructed target image is generated, and the loss is computed to optimize the model (Section 3.4).

Enhanced local features. Using shallow instead of deeper networks can effectively reduce the size of a model. As mentioned shallow CNNs have very limited receptive fields, while using dilated convolution [42] is helpful to enlarge receptive fields. By stacking the proposed Consecutive Dilated Convolutions (CDC) the network is able to "observe" the input at a larger area, while not introducing extra training parameters.

Low-computation global information. The enhanced local features are not enough to learn a global representation of the input without the help of Transformers to model long-range information. The MHSA module in the original Transformer [8] has a linear computational complexity to the input dimension, hence it limits the design of lightweight models. Instead of computing the attention across the spatial dimension the proposed Local-Global Features Interaction (LGFI) module adopts the cross-covariance attention [1] to compute the attention along the feature channels. Comparing with the original self-attention [8] it reduces the memory complexity from $O(hN^2 + Nd)$ to $O(d^2/h + Nd)$, and reduces the time complexity from $O(N^2d)$ to $O(Nd^2/h)$, where $h$ is the number of attention heads. The proposed architecture is described in detail below.

3.2. DepthNet

Depth encoder. The proposed Lite-Mono aggregates multi-scale features across four stages. The input image...
The down-sampling layers in the second and the third stage also receive the concatenated features output by the previous down-sampling layer. This design is similar to the residual connection proposed in ResNet [17], and is able to model better cross-stage correlation. Similarly, the output feature maps are further fed into the third and the fourth stages, and output features of dimension $H/8 \times W/8 \times C_3$ and $H/16 \times W/16 \times C_4$.

Consecutive Dilated Convolutions (CDC). The proposed CDC module utilizes dilated convolutions to extract multi-scale local features. Different from using a parallel dilated convolution module only in the last layer of the network [6] we insert several consecutive dilated convolutions with different dilation rates into each stage for adequate multi-scale contexts aggregation.

Given a two-dimensional signal $x[i]$ the output $y[i]$ of a 2D dilated convolution can be defined as:

$$y[i] = \sum_{k=1}^{K} x[i + r \cdot k] w[k],$$

where $w[k]$ is a filter with length $K$, and $r$ denotes the dilation rate used to convolve the input $x[i]$. In a standard non-dilated convolution $r = 1$. By using a dilated convolution the network can keep the size of the output feature map fixed while achieving a larger receptive field. Considering an input feature $X$ with dimension $H \times W \times C$ our CDC module outputs $\hat{X}$ as follows:

$$\hat{X} = X + \text{Linear}_C(\text{Linear}(\text{BN}(\text{DDWConv}_r(X)))),$$

where $\text{Linear}_C$ denotes a point-wise convolution operation, followed by the $\text{GELU}$ [18] activation. $\text{BN}$ is a batch normalization layer, and $\text{DDWConv}_r(\cdot)$ is a $3 \times 3$ depth-wise dilated convolution with dilation rate $r$.

Local-Global Features Interaction (LGFI). Given an input feature map $X$ with dimension $H \times W \times C$ it is linearly projected to the same dimensional queries $Q = X W_q$, keys $K = X W_k$, and values $V = X W_v$, where $W_q$, $W_k$, and $W_v$ are the parameters of the LGFI module.
and $W_v$ are weight matrices. The cross-covariance attention [1] is used to enhance the input $X$:

$$
\hat{X} = \text{Attention}(Q, K, V) + X,
$$

where $\text{Attention}(Q, K, V) = V \cdot \text{Softmax}(Q^T \cdot K)$. Then, the non-linearity of the features can be increased:

$$
\hat{X} = X + \text{Linear}_C(\text{Linear}(LN(\hat{X}))),
$$

where $LN$ is a layer normalization [2] operation. According to different channel numbers, CDC blocks, and dilation rates, four variants of the depth encoder are designed. Table 1 shows more details.

**Depth decoder.** Different from using a complicated up-sampling method [46] or introducing additional attention modules [3] Lite-Mono uses a depth decoder adapted from [15]. As shown in Figure 2 it increases the spatial dimension using bi-linear up-sampling, and uses convolutional layers to concatenate features from three stages of the encoder. Each up-sampling block follows a prediction head to output the inverse depth map at full, $\frac{1}{2}$, and $\frac{1}{4}$ resolution, respectively.

### 3.3. PoseNet

Following [15,46] this paper uses the same PoseNet for pose estimation. To be specific, a pre-trained ResNet18 is used as the pose encoder, and it receives a pair of color images as input. A pose decoder with four convolutional layers is used to estimate the corresponding 6-DoF relative pose between adjacent images.

### 3.4. Self-supervised learning

Different from the supervised training that utilizes ground truth of depth this work treats depth estimation as the task of image reconstruction. Similar to [45] the learning objective is modeled to minimize an image reconstruction loss $\mathcal{L}_r$, between a target image $I_t$ and a synthesized target image $\hat{I}_t$, and an edge-aware smoothness loss $\mathcal{L}_\text{smooth}$ constrained on the predicted depth map $D_t$.

**Image reconstruction loss.** The photometric reprojection loss is defined as:

$$
\mathcal{L}_p(\hat{I}_t, I_t) = \mathcal{L}_p(\mathcal{F}(I_s, P, D_t, K), I_t),
$$

where $\hat{I}_t$ can be obtained by a function $\mathcal{F}$ in terms of the source image $I_s$, the estimated pose $P$, the predicted depth $D_t$, and the camera’s intrinsics $K$. As introduced in [45] $\mathcal{L}_p$ is computed by a sum of the pixel-wise similarity $SSIM$ (Structural Similarity Index [37]) and the $L1$ loss between $\hat{I}_t$ and $I_t$:

$$
\mathcal{L}_p(\hat{I}_t, I_t) = \alpha \frac{1 - SSIM(\hat{I}_t, I_t)}{2} + (1 - \alpha)\|\hat{I}_t - I_t\|,
$$

where $\alpha$ is set to 0.85 empirically [15]. In addition, to deal with out-of-view pixels and occluded objects in a source image the minimum photometric loss [15] is computed:

$$
\mathcal{L}_p(I_s, I_t) = \min_{I_r \in [-1, 1]} \mathcal{L}_p(\hat{I}_t, I_t),
$$

where $I_s$ can be either the previous or the next frame with respect to the target image. Another binary mask [15] is used to remove moving pixels:

$$
\mu = \min_{I_r \in [-1, 1]} \mathcal{L}_p(I_s, I_t) > \min_{I_r \in [-1, 1]} \mathcal{L}_p(\hat{I}_t, I_t).
$$

Therefore, the image reconstruction loss is defined as:

$$
\mathcal{L}_r(I_t, I_t) = \mu \cdot \mathcal{L}_p(I_s, I_t),
$$

**Edge-aware smoothness loss.** To smooth the generated inverse depth maps an edge-aware smoothness loss is calculated, followed by [15, 44]:

$$
\mathcal{L}_\text{smooth} = |\partial_x d^*_t| e^{-|\partial_x I_t|} + |\partial_y d^*_t| e^{-|\partial_y I_t|},
$$

where $d^*_t = d_t / d_t$ denotes the mean-normalized inverse depth. The total loss can be expressed as:

$$
\mathcal{L} = \frac{1}{3} \sum_{s \in \{1, \frac{1}{2}, \frac{1}{4}\}} (\mathcal{L}_r + \lambda \mathcal{L}_\text{smooth}),
$$

where $s$ is the different scale output by the depth decoder. $\lambda$ is set to $1e^{-3}$ as in [15].

## 4. Experiments

This section evaluates the proposed framework and demonstrates the superiority of Lite-Mono.

### 4.1. Datasets

**KITTI.** The KITTI [13] dataset contains 61 stereo road scenes for research in autonomous driving and robotics, and it was collected by multiple sensors, including camera, 3D Lidar, GPU/IMU, etc. To train and evaluate the proposed method the Eigen split [9] is used, which has a total of 39,180 monocular triplets for training, 4,424 for evaluation, and 697 for testing. The self-supervised training is based on the known camera intrinsics $K$, as indicated in Eq. 5. By averaging all the focal lengths of images across the KITTI dataset this paper uses the same intrinsics for all images during training [15]. In the evaluation the predicted depth is restricted in the range of $[0, 80]$m, as is common practice.

**Make3D.** To evaluate the generalization ability of the proposed method it is further tested on the Make3D [32] dataset, which contains 134 test images of outdoor scenes. The model trained on the KITTI dataset is loaded and inferred directly on these test images.
Table 2. Comparison of Lite-Mono with some recent representative methods on the KITTI benchmark using the Eigen split [9]. All input images are resized to 640 × 192 unless otherwise specified. The best and the second best results are highlighted in bold and underlined, respectively. "M": KITTI monocular videos, "M+Se": monocular videos + semantic segmentation, "M*": input resolution 1024 × 320, "M†": without pre-training on ImageNet [7].

4.2. Implementation details

Hyperparameters. The proposed method is implemented in PyTorch and trained on a single NVIDIA TITAN Xp with a batch size of 12. AdamW [27] is the optimizer, and the weight decay is set to 1e−2. Drop-path is used in the CDC and LGFI modules to mitigate overfitting. For models trained from scratch an initial learning rate of 5e−4 with a cosine learning rate schedule [26] is adopted, and the training epoch is set to 35. It is found that pre-training on ImageNet
Data augmentation. Data augmentation is adopted as a preprocessing step to improve the robustness of the training. To be specific, the following augmentations are performed with a 50% chance: horizontal flips, brightness adjustment (±0.2), saturation adjustment (±0.2), contrast adjustment (±0.2), and hue jitter (±0.1). These adjustments are applied in a random order, and the same augmentation method is also used by [15, 28, 46].

Evaluation metrics Accuracy is reported in terms of seven commonly used metrics proposed in [10], which are Abs Rel, Sq Rel, RMSE, RMSE log, \(\delta < 1.25\), \(\delta < 1.25^2\), and \(\delta < 1.25^3\).

4.3. KITTI results

The proposed framework is compared with other representative methods with model sizes less than 35M, and the results are shown in Table 2. Lite-Mono beats all methods except MonoViT-tiny and is the smallest model (3.1M). Specifically, Lite-Mono greatly exceeds Monodepth2 [15] with a ResNet18 [17] backbone, but the model size is only about one-fifth of this model. It also outperforms the ResNet50 version of Monodepth2, which is the largest model (32.5M) in this table. Besides, Lite-Mono surpasses the recent well-designed small model R-MSFM [46]. Compared with the new MonoFormer [3] with a ResNet50 backbone the proposed Lite-Mono outperforms it in all metrics. Our other two smaller models also achieve satisfactory results, considering that they have fewer trainable parameters. In the last two rows of the table, the proposed Lite-Mono-8M also performs better than MonoViT-tiny, the smallest model of MonoViT [43], with fewer parameters. Figure 4 shows that Lite-Mono achieves satisfactory results, even on challenging images where moving objects are close to the camera (column 1).

4.4. Make3D results

The proposed method is evaluated on the Make3D dataset to show its generalization ability in different outdoor scenes. The model trained on KITTI is directly inferred without any fine-tuning. Table 3 shows the comparison of Lite-Mono with the other three methods, and Lite-Mono performs the best. Figure 5 shows some qualitative results. Owing to the proposed feature extraction modules Lite-Mono is able to model both local and global contexts, and perceives objects with different sizes.

4.5. Complexity and speed evaluation

The proposed models’ parameters, FLOPs (floating point of operations), and inference time are evaluated on an NVIDIA TITAN Xp and a Jetson Xavier and are compared with Monodepth2 [15], R-MSFM [46], and MonoViT-tiny [43]. Table 4 shows that the proposed design has a good balance between model size and speed. Notice that Lite-Mono-tiny outperforms Monodepth2 both in speed and accuracy (Table 2). Although R-MSFM [46] is a lightweight model it is slow. The latest MonoViT-tiny [43] runs the slowest due to its parallel blocks and multiple layers of self-attention. Our models also infer quickly on the Jetson Xavier, which allows them to be used on edge devices.

4.6. Ablation study on model architectures

To further demonstrate the effectiveness of the proposed model the ablation study is conducted to evaluate the importance of different designs in the architecture. We remove or adjust some modules in the network, and report their results on KITTI, as shown in Table 5.

The benefit of LGFI blocks. When all the LGFI blocks in stage 2, 3, and 4 are removed, the model size decreases by 0.4M, but the accuracy also drops. The proposed LGFI is crucial to make Mono-Lite encode long-range global contexts, thus making up for the drawback that CNNs can only extract local features.

The benefit of dilated convolutions. If all the dilation rates of convolutions in CDC blocks are set to 1, \(i.e.,\) there are no dilated convolutions used in the network. It can be observed that although the model size remains the same the accuracy drops more than for not using LGFI blocks. The benefit of introducing the CDC module is to enhance the locality by gradually extracting multi-scale features, while not adding additional trainable parameters.

The benefit of pooled concatenations. Accuracy also decreases when three pooled concatenations are removed.

<table>
<thead>
<tr>
<th>Method</th>
<th>Abs Rel</th>
<th>Sq Rel</th>
<th>RMSE</th>
<th>RMSE log</th>
</tr>
</thead>
<tbody>
<tr>
<td>DDVO [30]</td>
<td>0.387</td>
<td>4.720</td>
<td>8.090</td>
<td>0.204</td>
</tr>
<tr>
<td>Monodepth2 [15]</td>
<td>0.322</td>
<td>3.589</td>
<td>7.417</td>
<td>0.163</td>
</tr>
<tr>
<td>R-MSFM6 [46]</td>
<td>0.334</td>
<td>3.285</td>
<td>7.212</td>
<td>0.169</td>
</tr>
<tr>
<td>Lite-Mono (Ours)</td>
<td>0.305</td>
<td>3.060</td>
<td>6.981</td>
<td>0.158</td>
</tr>
</tbody>
</table>

Table 3. Comparison of the proposed Lite-Mono to some other methods on the Make3D [32] dataset. All models are trained on KITTI [13] with an image resolution of 640 × 192.
Table 4. Model complexity and speed evaluation. We compare parameters, FLOPs (floating point of operations), and inference speed. The input size is 640 × 192, and the batch size is 16.

<table>
<thead>
<tr>
<th>Method</th>
<th>Params. (M)</th>
<th>FLOPs (G)</th>
<th>Params. (M)</th>
<th>FLOPs (G)</th>
<th>Params. (M)</th>
<th>FLOPs (G)</th>
<th>Speed (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Monodepth2 [15]</td>
<td>11.2</td>
<td>4.5</td>
<td>3.1</td>
<td>3.5</td>
<td>14.3</td>
<td>8.0</td>
<td>3.8</td>
</tr>
<tr>
<td>R-MSFMs [46]</td>
<td>0.7</td>
<td>2.4</td>
<td>2.8</td>
<td>14.1</td>
<td>3.5</td>
<td>16.5</td>
<td>7.8</td>
</tr>
<tr>
<td>R-MSFMs [46]</td>
<td>0.7</td>
<td>2.4</td>
<td>3.1</td>
<td>28.8</td>
<td>3.8</td>
<td>31.2</td>
<td>13.1</td>
</tr>
<tr>
<td>Monovit [13]</td>
<td>5.6</td>
<td>7.6</td>
<td>4.7</td>
<td>15.9</td>
<td>10.3</td>
<td>23.7</td>
<td>13.5</td>
</tr>
<tr>
<td>Lite-Mono-tiny (Ours)</td>
<td>2.0</td>
<td>2.4</td>
<td>0.2</td>
<td>0.5</td>
<td>2.2</td>
<td>2.9</td>
<td>3.3</td>
</tr>
<tr>
<td>Lite-Mono-small (Ours)</td>
<td>2.5</td>
<td>4.1</td>
<td>0.2</td>
<td>0.7</td>
<td>2.5</td>
<td>4.8</td>
<td>4.3</td>
</tr>
<tr>
<td>Lite-Mono (Ours)</td>
<td>2.9</td>
<td>4.4</td>
<td>0.2</td>
<td>0.7</td>
<td>3.1</td>
<td>5.1</td>
<td>4.5</td>
</tr>
<tr>
<td>Lite-Mono-8M (Ours)</td>
<td>8.1</td>
<td>9.5</td>
<td>0.6</td>
<td>1.7</td>
<td>8.7</td>
<td>11.2</td>
<td>6.5</td>
</tr>
</tbody>
</table>

Table 5. Ablation study on model architectures. All the models are trained and tested on KITTI with the input size 640 × 192.

Table 6. Ablation study on dilation rates.

5. Conclusions

This paper presents a novel architecture Lite-Mono for lightweight self-supervised monocular depth estimation. A hybrid CNN and Transformer architecture is designed to model both multi-scale enhanced local features and long-range global contexts. The experimental results on the KITTI dataset demonstrate the superiority of our method. By setting optimized dilation rates in the proposed CDC blocks and inserting the LGFI modules to obtain the local-global feature correlations, Lite-Mono can perceive different scales of objects, even challenging moving objects close to the camera. The generalization ability of the model is also validated on the Make3D dataset. Besides, Lite-Mono achieves a good trade-off between model complexity and inference speed.

Acknowledgements

This project has received funding from the European Union’s Horizon 2020 Research and Innovation Programme and the Korean Government under Grant Agreement No 833435. Content reflects only the authors’ view and the Research Executive Agency (REA) and the European Commission are not responsible for any use that may be made of the information it contains.

This work also uses the Geospatial Computing Platform of the Center of Expertise in Big Geodata Science (CRIB) (https://crib.utwente.nl). We thank Dr. Serkan Girgin for providing the computing infrastructure.
References


[12] Ravi Garg, Vijay Kumar Bg, Gustavo Carneiro, and Ian Reid. Unsupervised cnn for single view depth estimation: Geometry to the rescue. In ECCV, 2016. 2


