Depth Estimation with Simplified Transformer

John Yang, Le An, Anurag Dixit, Jinkyu Koo, Su Inn Park
NVIDIA
{johnyang, lean, anuragd, jinkyuk, joshp}@nvidia.com

Abstract

Transformer and its variants have shown state-of-the-art results in many vision tasks recently, ranging from image classification to dense prediction. Despite of their success, limited work has been reported on improving the model efficiency for deployment in latency-critical applications, such as autonomous driving and robotic navigation. In this paper, we aim at improving upon the existing transformers in vision, and propose a method for self-supervised monocular Depth Estimation with Simplified Transformer (DEST), which is efficient and particularly suitable for deployment on GPU-based platforms. Through strategic design choices, our model leads to significant reduction in model size, complexity, as well as inference latency, while achieving superior accuracy as compared to state-of-the-art. We also show that our design generalize well to other dense prediction task without bells and whistles.

1. Introduction

Accurate depth estimation is an essential capability for geometric perception within a scene. Estimated depth provides rich visual cues for general perception, navigation, planning, and reasoning against occlusions for applications such as robotics [8] and advanced driver-assistance systems (ADAS) [1]. Recently, deep learning based methods [8, 18] have shown that depth can be learned from a single image by using convolutional neural networks (CNN). However, direct supervision requires large amount of ground-truth depth maps, which are expensive to obtain in reality. On the other hand, self-supervised, or sometimes referred to as unsupervised methods can take advantage of geometrical constraints on image sequences as the sole source of supervision. For example, previous work [12, 20, 31] showed that CNN-based depth and ego-motion networks can be solely trained on monocular video sequences without using ground-truth depth or stereo image pairs.

The key to the self-supervised learning methods is to build a task consistency for training separated CNN networks, where predictions from depth network and pose network are jointly constrained by image reconstruction error [2, 11, 12, 14, 22, 31, 33]. While this structure of paired depth-pose networks has been largely adopted, they are mainly built with CNNs that have evolved towards more complex architectures that are computationally demanding.

On the other hand, inspired by the seminal work on transformer [25], vision transformers have emerged in many applications [7, 19, 24], benefiting from the attention mechanism and simpler network structure. In order to exploit the capacity of vision transformers and self-supervised learning systems [3], in this paper we aim to improve the performance of self-supervised monocular depth estimation in terms of both accuracy and latency for deployment, through improved design choices. To this end, our main contributions are the following:

- Simplified transformer with strategical design choices that are hardware friendly, yielding networks that only consist of very basic operations and operate efficiently
- Transformer-based Depth-Net and Pose-Net with joint attention mechanism for more effective learning

We show that on public benchmark dataset our model is over 85% smaller in model size and complexity, while being significantly faster in terms of latency and more accurate, as compared to previous state-of-the-art. We also show that this architecture can generalize well to other dense prediction tasks such as semantic segmentation.

2. Related Works

Monocular Depth Estimation. With the emergence of learning-based methods, it has been shown that depth can be learned directly from a single image in a supervised manner [8, 18]. To date, self-supervised methods, which take advantage of abundant unlabeled data for training, have drawn great attention. Godard *et al*. [11] proposed a novel training objective that enforces consistency between disparity produced by left and right images, and this method was further improved in [10] with new choices of loss functions and training strategies. In [12], symmetrical packing and

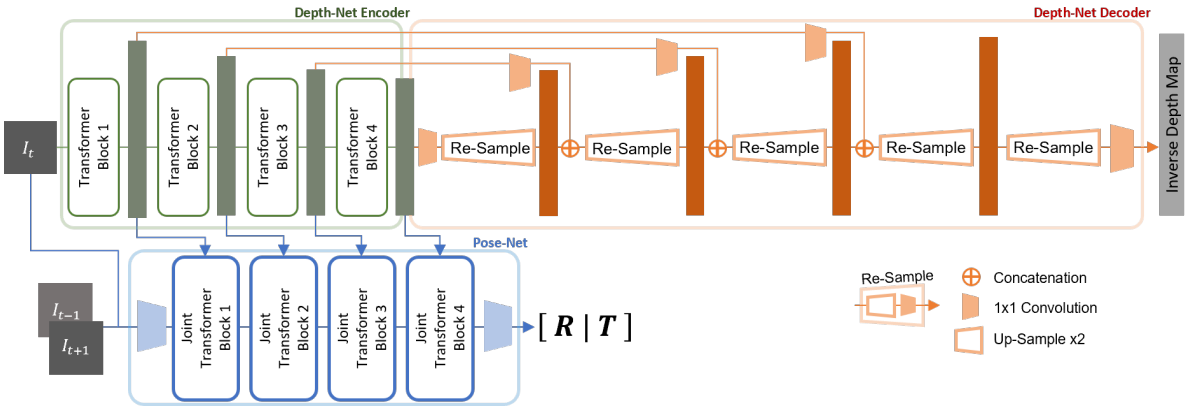


Figure 1. The proposed framework for self-supervised depth estimation with simplified transformer (DEST). Both Depth-Net and Pose-Net are trained together. For inference, only Depth-Net is needed.

unpacking blocks are proposed to generate detailed representation using 3D convolutions along with velocity information for better scale-awareness.

Vision Transformers. The pioneer work of Vision Transformer (ViT) [7] showed that a pure transformer with sequence of image patches as input works as well as CNNs. To alleviate the need of large amount of pre-training data, Data-efficient image Transformer (DeiT) [24] was proposed with a novel distillation strategy. To address image scale and resolution variation, Swin Transformer, a hierarchical transformer with shifted windows, was invented [19]. Pyramid Vision Transformer (PVT) [28] utilizes a pyramid structure that allows fine-grained inputs for dense prediction tasks in conjunction with progressive shrinking to reduce computations. SegFormer [29] leverages multi-scale features for accurate pixel-level prediction, while positional encoding and complexity in decoder are dropped, resulting in performance boost in both accuracy and latency. Recently, Li *et al*. proposed STereo TRansformer (STTR) which employs transformer for sequence-to-sequence correspondence modeling for stereo depth estimation [16]. ViT backbone is employed as a encoder along with a convolutional decoder for supervised depth estimation in [21].

Efficient Transformers. Some recent efforts focus on reducing the complexity in transformers mainly from a theoretical perspective [4, 27]. For vision tasks, Li *et al*. [15] introduced a multi-stage efficient transformer (EsViT) in conjunction with sparse self-attention. Jia *et al*. [13] proposed an efficient fine-grained manifold distillation approach that allows student network to achieve better performance when learning from the more complex teacher transformer. However, in reality the projected speedup may not be realized due to hardware limitations that inherently determine software capability. Different from the scope of the aforementioned approaches, we introduce a simplified design of transformer architecture, which is particular friendly to GPU hardware in order to achieve efficient inference.

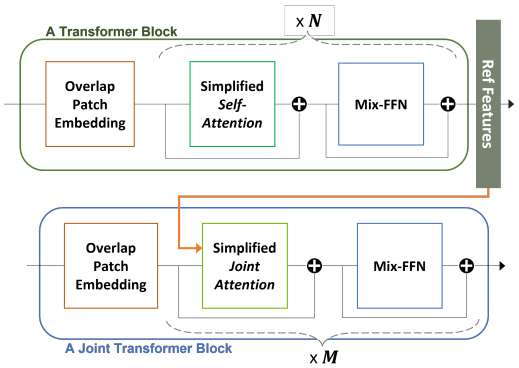


Figure 2. Transformer block for Depth-Net (top) and joint transformer block for Pose-Net (bottom). In joint transformer block, the features from Depth-Net are used to computed joint attention.

3. Method

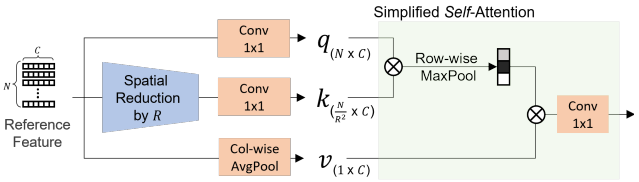
The design of Depth-Net and Pose-Net is shown in Fig. 1. The encoder of Depth-Net and the Pose-Net consist of four transformer blocks. In the training process, the predicted depth map and pose are used for view synthesis, and the photometric loss is used as training objective [12, 33]. Only Depth-Net is needed for inference.

3.1. Simplified Transformer

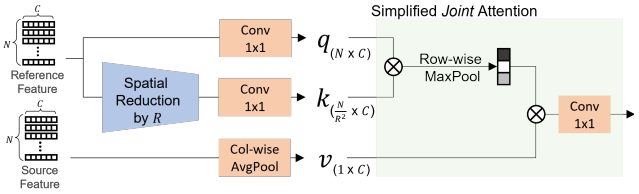
Each transformer block contains an *Overlapping Patch Embedding* layer, followed by repetitive sub-blocks of *Simplified Attention* and *Mix-FFN* layers, as shown in Fig. 2.

Overlapping Patch Embedding. To preserve the continuity of local image context, similar to the approach in [29], we employ an overlapping patch embedding block which simply consists of a 2D convolution followed by a batch normalization (BN). The size of input feature map is reduced by half in the beginning of every transformer block.

Simplified Attention. To reduce the complexity in the attention module, we made several improvements. First, we apply the sequence reduction process [28, 29] with a reduc-



(a) Simplified *Self*-Attention in Depth-Net



(b) Simplified *Joint* Attention in Pose-Net

Figure 3. Simplified attention modules. The softmax used in attention is replaced by max pooling and the v values are obtained by applying average pooling from input features.

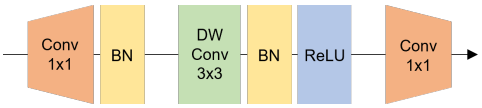


Figure 4. Mix-FFN in the transformer blocks.

tion ratio R such that the size of K is reduced from $N \times C$ to $\frac{N}{R^2} \times C$. Secondly, instead of using a learnable layer to obtain v, we apply a column-wise average pooling to reduce v from $N \times C$ to $1 \times C$. In addition, the softmax operation in attention yields gradients that mainly flow through maximum outputs, which is similar to the max-pooling operation. Therefore, we replace softmax with row-wise max pooling to remove the complexity in computing softmax. The self-attention is shown in Fig. 3a, which is used in the Depth-Net. For the Pose-Net, we propose the joint attention that receives q and k features from corresponding blocks in Depth-Net, and v from features in the Pose-Net, as shown in Fig. 3b. The feature sharing allows additional gradient signals for Depth-Net when it is jointly trained with Pose-Net [26]. With the aforementioned improvements, the computation in the attention is greatly reduced as compared to other vision transformers such as that in [29].

Mix-FFN. Instead of applying MLP after attention modules, we employ mix-FFN layer instead. The structure of our mix-FFN is shown in Fig. 4, which is implemented differently from that in [29]. The BN layers after 1x1 convolution and depth-wise convolution are crucial in stabilizing the training, and ReLU substitutes the GELU activation for reduced computation.

Elimination of Layer-Norms. Layer normalization (LN) is widely adopted in transformer-based networks. However, LN requires statistics of inputs being computed on the fly, which imposes additional cost at inference time. On the other hand, BN uses the accumulated statistics from training and avoids such computation in the inference, and thus can be more favorable in transformers [23, 30]. Note that simply replacing LN with BN would cause the training to diverge, therefore our LN-free networks are implemented as the following: **1**. Two BNs are inserted in the mix-FFN layers as shown in Fig. 4; **2**. LNs in the spatial reduction layer [28, 29] are removed, and the attention layers are also free of normalization; **3**. A BN layer is placed after every joint transformer block in the Pose-Net, while the transformer blocks in the Depth-Net encoder are not followed by any normalization. Such modifications intend to lessen BN computations during inference while still stabilizing the training, since only Depth-Net is needed for inference.

Progressive Decoder. Inspired by the concept of feature pyramid network (FPN) [17], the decoder in the Depth-Net receives outputs from all transformer blocks in the encoder, and combines them in a progressive manner as shown in Fig. 1. Bilinear interpolation is used to upsample the feature maps, and the size of the output depth map from the decoder is equal to the input size.

Network Composition. To this end, our networks are largely composed of very simple and basic operations, namely *convolution*, *depth-wise convolution*, *batch normalization*, *ReLU*, *max-pooling*, and *avg-pooling*. Our design is driven in such a way since these operations have been well supported and optimized on GPU-based platforms, therefore making our networks hardware-friendly and efficient.

4. Experiments

Implementation Details. Our models are implemented in PyTorch 1.10 and trained with 8 NVIDIA V100 GPUs. Adam optimizer with an initial learning rate of 4×10^{-5} is used for training. For Depth-Net, a batch size of one is used, and for Pose-Net, training sequences are generated with a stride of 2, meaning that I_{t-1} and I_{t+1} are concatenated with I_t as an input. The input resolution to our models and benchmark models is 640×192.

Similar to [29], we implement six variants of DEST models. Specifically, our lightweight models DEST-B0 and B1 have a configuration of (2, 2, 2, 2), each of which indicates the number of attention and mix-FFN layers in a transformer block, as shown in Fig. 2. Those settings are (3, 3, 6, 3) for B2, (3, 6, 8, 3) for B3, (3, 8, 12, 5) for B4, and (3, 10, 16, 5) for the largest model B5. We set the numbers of output feature maps to (32, 64, 160, 256) for B0 and (64, 128, 250, 320) for B1-B5. For Pose-Net, we use B3 in all experiments.

Accuracy. We benchmark on KITTI dataset [9] and report the metrics described in [8]. The quantitative results are summarized in Table 1. Compared to the current state-of-the-art PackNet-SfM [12] with pre-training, our Depth-Net B3 achieves over 85% reduction in the number of parame-

Depth-Net	Pose-Net	Connectivity	Dataset	Abs Rel↓	Sq Rel↓	RMSE↓	$\text{RMSE}_{\log}$↓	#MParams↓	#GMACs↓
Monodepth2 [10]	ConvNet	No	K	0.132	1.044	5.142	0.210	14.84	8.04
Monodepth2 [10]	ConvNet	No	IN+K	0.115	0.903	4.863	0.193	14.84	8.04
PackNet-SfM [12]	ConvNet	No	K	0.111	0.800	4.576	0.189	128.29	205.49
PackNet-SfM [12]	ConvNet	No	CS+K	0.108	0.758	4.506	0.185	128.29	205.49
SETR-MLA32 [32]	ConvNet	No	K	0.184	1.669	6.407	0.257	-	-
SegFormer-B0 [29]	ConvNet	No	K	0.139	0.981	5.657	0.205	3.82	4.21
SegFormer-B3 [29]	DEST-B3	No	K	0.119	0.803	5.033	0.182	47.32	35.02
SegFormer-B3 [29]	DEST-B3	Yes	K	0.113	0.798	4.917	0.179	47.32	35.02
SegFormer-B3 [29]	DEST-B3	Yes	CS+K	0.105	0.794	4.707	0.172	47.32	35.02
DEST-B0	DEST-B3	Yes	CS+K	0.116	0.910	4.982	0.219	4.68	4.82
DEST-B1	DEST-B3	Yes	CS+K	0.115	0.868	4.724	0.207	10.12	14.37
DEST-B2	DEST-B3	Yes	CS+K	0.108	0.831	4.636	0.181	16.03	17.19
DEST-B3	DEST-B3	Yes	CS+K	0.103	0.796	4.410	0.170	19.71	19.78
DEST-B4	DEST-B3	Yes	CS+K	0.098	0.767	4.285	0.168	38.53	27.98
DEST-B5	DEST-B3	Yes	CS+K	0.095	0.752	4.207	0.165	45.95	31.50

Table 1. Quantitative results on the KITTI dataset [9]. Connectivity refers to the feature sharing between Depth-Net and Pose-Net. The #MParams and #GMACs are from Depth-Net only since Pose-Net is not needed in inference. CS+K and IN+K refer that the given model is pre-trained either with CityScapes [5] or ImageNet [6] data.

	PyTorch↓	TensorRT FP32↓	TensorRT FP16↓
PackNet-SfM [12]	64.76	42.54	20.54
DEST-B3	18.31	12.36	8.87

Table 2. Comparison of end-to-end inference latency in ms using PyTorch and TensorRT with different precisions.

ters and 90% reduction in the number of MACs, while outperforming in most metrics by a large margin. Even the smallest model B0 yields competitive results compared to other methods. The proposed connectivity between Depth-Net and Pose-Net helps improve the accuracy, as suggested by the results of baseline Depth-Net using SegFormer-B3. Some sample predicted depth maps are shown in Fig. 5, in which fine details and structures are observed.

Latency. We compare the latency of DEST-B3 to PackNet-SfM [12] in their original PyTorch implementations as well as the inference results using NVIDIA TensorRT 8.2 library in FP32 and FP16 precisions. All the latency numbers in our experiments were obtained on an NVIDIA V100 GPU. Table 2 summarizes the runtime with a batch size of one. Compared to PackNet-SfM [12], our model runs notably faster in all settings. With the help from TensorRT, the latency of our model further improved from 18.31ms to 12.36ms in FP32 precision. When FP16 is enabled, the latency of our model drops to 8.87ms. The improvements in latency resonate with the reduction in model size and computation, affirming the efficacy of our design choices.

Generalization to Semantic Segmentation. We directly apply our model to semantic segmentation with a simplified decoder that outputs a segmentation map with half size of that of the input by removing the last re-sample block. We compare ours to SegFormer [29] with both models trained from scratch on the CityScapes dataset [5] . Table 3 shows the results with an input resolution of 1024×1024. Without bells and whistles, our model achieves better segmentation quality with greatly reduced latency.

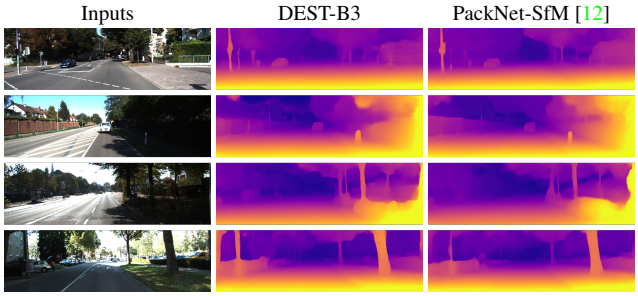


Figure 5. Samples of predicted depth maps.

	mIoU↑	#MParams↓	#GMACs↓	Latency (ms)↓
SegFormer-B0	62.57	3.82	30.58	28.58
DEST-B0	**63.12**	**3.58**	**22.75**	**13.94**
SegFormer-B3	72.30	47.32	299.07	127.17
DEST-B3	**72.58**	**18.08**	**142.78**	**51.41**

Table 3. Comparisons with SegFormer [29] on the CityScapes dataset [5]. Latency is obtained with FP16 precision.

5. Conclusions

In this paper, we presented a design of simplified transformer for self-supervised depth estimation. The simplifications as well as the proposed joint-attention and connectivity mechanism have shown to be effective with greatly reduced model complexity and inference latency, as compared to other benchmark methods. We also showed that our model can be directly generalized to other dense image prediction task such as semantic segmentation with improved accuracy and latency. We hope our design can serve as a practical choice for real-world applications such as autonomous driving and robotics, where both high efficiency and accuracy are demanded at the same time. In the future, we would like to further evaluate our model with quantization-aware training and inference in lower precision for further improved efficiency.

References

[1] Ahmed Ali, Ali Hassan, Afsheen Rafaqat Ali, Hussam Ullah Khan, Wajahat Kazmi, and Aamer Zaheer. Real-time vehicle distance estimation using single view geometry. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1111–1120, 2020. 1

[2] Jiawang Bian, Zhichao Li, Naiyan Wang, Huangying Zhan, Chunhua Shen, Ming-Ming Cheng, and Ian Reid. Unsupervised scale-consistent depth and ego-motion learning from monocular video. *Advances in neural information processing systems*, 32:35–45, 2019. 1

[3] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. *arXiv preprint arXiv:2104.14294*, 2021. 1

[4] Krzysztof Marcin Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamas Sarlos, Peter Hawkins, Jared Quincy Davis, Afroz Mohiuddin, Lukasz Kaiser, David Benjamin Belanger, Lucy J Colwell, and Adrian Weller. Rethinking attention with performers. In *International Conference on Learning Representations*, 2021. 2

[5] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 4

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255, 2009. 4

[7] 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. In *International Conference on Learning Representations*, 2021. 1, 2

[8] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. *arXiv preprint arXiv:1406.2283*, 2014. 1, 3

[9] A Geiger, P Lenz, C Stiller, and R Urtasun. Vision meets robotics: The KITTI dataset. *Int. J. Rob. Res.*, 32(11):1231–1237, sep 2013. 3, 4

[10] Clement Godard, Oisin Mac Aodha, Michael Firman, and Gabriel Brostow. Digging into self-supervised monocular depth estimation. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3827–3837, 2019. 1, 4

[11] Clément Godard, Oisin Mac Aodha, and Gabriel J Brostow. Unsupervised monocular depth estimation with left-right consistency. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 270–279, 2017. 1

[12] Vitor Guizilini, Rares Ambrus, Sudeep Pillai, Allan Raventos, and Adrien Gaidon. 3D packing for self-supervised monocular depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2485–2494, 2020. 1, 2, 3, 4

[13] Ding Jia, Kai Han, Yunhe Wang, Yehui Tang, Jianyuan Guo, Chao Zhang, and Dacheng Tao. Efficient vision transformers via fine-grained manifold distillation. *arXiv preprint arXiv:2107.01378*, 2021. 2

[14] Maria Klodt and Andrea Vedaldi. Supervising the new with the old: learning sfm from sfm. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 698–713, 2018. 1

[15] Chunyuan Li, Jianwei Yang, Pengchuan Zhang, Mei Gao, Bin Xiao, Xiyang Dai, Lu Yuan, and Jianfeng Gao. Efficient self-supervised vision transformers for representation learning. In *International Conference on Learning Representations*, 2022. 2

[16] Z. Li, X. Liu, N. Drenkow, A. Ding, F. X. Creighton, R. H. Taylor, and M. Unberath. Revisiting stereo depth estimation from a sequence-to-sequence perspective with transformers. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6177–6186, 2021. 2

[17] Tsung-Yi Lin, Piotr Dollar, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 3

[18] Fayao Liu, Chunhua Shen, Guosheng Lin, and Ian Reid. Learning depth from single monocular images using deep convolutional neural fields. *IEEE transactions on pattern analysis and machine intelligence*, 38(10):2024–2039, 2015. 1

[19] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10012–10022, October 2021. 1, 2

[20] Reza Mahjourian, Martin Wicke, and Anelia Angelova. Unsupervised learning of depth and ego-motion from monocular video using 3D geometric constraints. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5667–5675, 2018. 1

[21] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 12179–12188, October 2021. 2

[22] Anirban Roy and Sinisa Todorovic. Monocular depth estimation using neural regression forest. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5506–5514, 2016. 1

[23] Sheng Shen, Zhewei Yao, Amir Gholami, Michael Mahoney, and Kurt Keutzer. Powernorm: Rethinking batch normalization in transformers. In *International Conference on Machine Learning*, pages 8741–8751, 2020. 3

[24] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Herve Jegou. Training data-efficient image transformers & distillation through at-

tention. In *Proceedings of the 38th International Conference on Machine Learning*, pages 10347–10357, 2021. 1, 2

[25] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017. 1

[26] Chaoyang Wang, José Miguel Buenaposada, Rui Zhu, and Simon Lucey. Learning depth from monocular videos using direct methods. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2022–2030, 2018. 3

[27] Sinong Wang, Belinda Li, Madian Khabsa, Han Fang, and Hao Ma. Linformer: Self-attention with linear complexity. *arXiv preprint arXiv:2006.04768*, 2020. 2

[28] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 568–578, October 2021. 2, 3

[29] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M. Alvarez, and Ping Luo. SegFormer: Simple and efficient design for semantic segmentation with transformers. In *Advances in Neural Information Processing Systems 34 pre-proceedings (NeurIPS)*, 2021. 2, 3, 4

[30] Zhuliang Yao, Yue Cao, Yutong Lin, Ze Liu, Zheng Zhang, and Han Hu. Leveraging batch normalization for vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, pages 413–422, October 2021. 3

[31] Wang Zhao, Shaohui Liu, Yezhi Shu, and Yong-Jin Liu. Towards better generalization: Joint depth-pose learning without posenet. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9151–9161, 2020. 1

[32] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip H. S. Torr, and Li Zhang. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 4

[33] Tinghui Zhou, Matthew Brown, Noah Snavely, and David G Lowe. Unsupervised learning of depth and ego-motion from video. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1851–1858, 2017. 1, 2