MonoMixer: Marrying Convolution and Vision Transformer for Efficient Self-Supervised Monocular Depth Estimation

Zhiyong Chang[1] , **Yan Wang**[2]
[1]Peking University
[2]Zhejiang University
changzy@pku.org.cn, wangyan001@zuaa.zju.edu.cn,

Abstract

Self-supervised monocular depth estimation that does not require hard-to-source depth labels for training has been widely studied in recent years. Due to its significant and growing needs, many lightweight but effective architectures have been designed for edge devices. Convolutional Neural Networks (CNNs) have shown its extraordinary ability in monocular depth estimation. However, their limited receptive field stints existing methods to reason only locally, inhibiting the effectiveness of the self-supervised paradigm. Recently, Transformers has achieved great success in estimating depth maps from monocular images. Nevertheless, massive parameters in the Transformers hinder the deployment to edge devices. In this paper, we propose MonoMixer, a brand-new lightweight CNN-Transformer architecture with three main contributions: 1) The details-augmented (DA) block employs graph reasoning unit to capture abundant local details, resulting depth prediction at a higher level of precision. 2) The self-modulate channel attention (SMCA) block adaptively adjust the channel weights of feature maps, aiming to emphasize the crucial features and aggregate channel-wise feature maps of different patterns. 3) The global-guided Transformer (G2T) block integrates global semantic token into multi-scale local features and exploit cross-attention to encode long-range dependencies. Furthermore, experimental results demonstrate the superiority of our proposed MonoMixer both at model size and inference speed, and achieve state-of-the-art performance on three datasets: KITTI, Make3D and Cityscapes. Specifically, our proposed MonoMixer outperforms MonoFormer by a large margin in accuracy, with about **95** % fewer parameters.

1 Introduction

Depth estimation is a fundamental and crucial task in various computer vision applications such as autonomous driving, augmented reality and robotics navigation. Recently, the fully-supervised monocular depth estimation methods [Eigen and Fergus, 2014; Fu *et al.*, 2018; Guo *et al.*, 2018], have achieved remarkable results, while they require numerous annotated depth maps which could only be collected from expensive LiDAR sensors. In contrast, self-supervised methods are more favorable, as they utilize synchronized stereo-pairs of frames [Garg *et al.*, 2016] or geometrical constraints on monocular video [Godard *et al.*, 2016] as the supervisory signals. Therefore, this paper use monocular videos for training purposes.

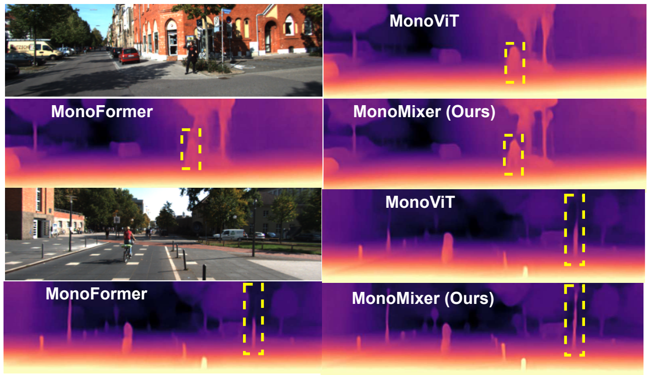

Figure 1: The proposed MonoMixer has fewer parameters than MonoViT and MonoFormer, but obtain more accurate depth maps.

In addition to improving the accuracy of self-supervised monocular training by exploiting semantic information and innovative loss functions to address the occlusion problems, several works focus on developing more efficacious CNN models [Yan *et al.*, 2021; Zhou *et al.*, 2024]. However, the local nature of convolution operation can not capture long-range global relationships. To attain better performance a CNN-based architecture can adopt a more intricate backbone [Godard *et al.*, 2018; Wang and Cheng, 2023], which also results in more parameters. In addition, CNN is limited because it does not consider the characteristics of the object geometric appearances, which leading to an inaccurate perception of entire layout for the complex scenes. Recently, several studies introduce Vision Transformer (ViT) to monocular depth estimation architectures, which enables the model to capture the global dependencies. Nevertheless, the quadratic complexity of the Multi-Head Self-Attention (MHSA) block in a

Transformer impedes the design of lightweight and real-time models, compared with CNN architectures.

In this paper, we propose MonoMixer, a lightweight and efficient hybrid model for self-supervised monocular depth estimation. To reduce the computational complexity, we propose a global-guided transformer (G2T) block, which injecting the global information into the multi-scale local features via the shared global semantic token. Furthermore, to generate sharper object boundaries, we introduce the details-augmented (DA) block employing graph convolution network (GCN) to enhance specific details, as shown in Figure 1. To comprehensively understand the 3D structure of the complex scene, we design the self-modulate channel attention (SMCA) block to capture abundant contextual information of scene geometry and emphasize the crucial feature representations. The main contributions of our work are summarized as follows:

- We propose a novel CNN and Transformer hybrid architecture (MonoMixer) for self-supervised monocular depth estimation.
- We develop a simple but effective global-guided Transformer (G2T) block composed of a global semantic token and cross-attention, capturing global contextual information and reducing the computational complexity.
- we introduce the details-augmented (DA) block exploiting graph reasoning unit to obtain rich local details and more discriminative feature representation.
- we design the self-modulate channel attention (SMCA) block to efficiently capture global context of scene structure and highlight salient channel-wise features.
- We conduct extensive experiments on KITTI, Make3D, and Cityscapes datasets, demonstrating our model achieves state-of-the-art performance with the least trainable parameters.

2 Related Work

2.1 Self-supervised Depth Estimation

Considering that large numbers of accurate ground truth depth maps are not always obtainable, self-supervised depth estimation approaches that do not need ground truth for training have gained significant attention. [Garg *et al.*, 2016] trains the convolutional encoder for predicting the depth map by minimizing the reconstruction loss between the source image and target image. [Godard *et al.*, 2016] extends this work and attain better performance by using a left-right disparity consistency loss. Furthermore, several works [Casser *et al.*, 2019; Godard *et al.*, 2018; Jung *et al.*, 2021; Shyam *et al.*, 2024; Bello *et al.*, 2024] exploit multi-task learning to perceive dynamic scenes. Monodepth2 [Godard *et al.*, 2018] utilizes a minimum reprojection loss to address occlusion problems, an auto-masking loss to neglect invalid pixels, and a full-resolution multi-scale sampling method to reduce visual artifacts. [Jung *et al.*, 2021] introduces a multi-task neural network with cross-task attention and semantics-guided triplet loss to successfully extract semantics-aware feature representation. [Shyam *et al.*, 2024] intertwines depth estimation and panoptic segmentation networks to facilitating depth estimation in dynamic scenes. [Bello *et al.*, 2024] exploits pixel positional information and moving object mask to learn single image depth estimation from monocular videos.

2.2 Network Architectures

The underlying neural network architecture plays a pivotal role in determining the effectiveness of monocular depth prediction. CADepth-Net [Yan *et al.*, 2021] exploits channel-wise attention modules to capture global dependencies and boost local details information. DaCCN [Han *et al.*, 2023] designs a direction-aware module to learn to adjust the feature extraction in various directions, and a cumulative convolution to efficiently aggregate crucial environmental features. With the emergence of Vision Transformer (ViT) [Dosovitskiy *et al.*, 2020], several efforts [Ranftl *et al.*, 2021; Yang *et al.*, 2021; Zhao *et al.*, 2022; Bae *et al.*, 2023; Xing *et al.*, 2023; Zhang *et al.*, 2023; Wang *et al.*, 2024] apply it to monocular depth estimation task, and achieve significant advancement. MonoViT [Zhao *et al.*, 2022] achieves higher depth accuracy by combining plain convolutions with transformer blocks. MonoFormer [Bae *et al.*, 2023] proposes a CNN-Transformer hybrid architecture with multi-scale feature fusion block, which captures both local and global information. SQLdepth [Wang *et al.*, 2024] exploits a novel Self Query Layer (SQL) to captures the intrinsic geometry of the scene, which achieving state-of-the-art performance. However, due to the expensive computational cost of Multi-Head Self-Attention (MHSA) in Transformer block, the aforementioned approaches have more trainable parameters and have a significant latency gap compared with methods only using CNNs [Wofk *et al.*, 2019; Rudolph *et al.*, 2022; Zhou *et al.*, 2024].

3 Proposed Framework

3.1 Motivation

Combining convolutions and Transformers leads to robust and high-performing models compared to traditional ViTs. However, a challenge remains: *how can we effectively merge the strengths of convolutions and transformers to create lightweight networks suitable for self-supervised monocular depth estimation?* Although several lightweight hybrid architectures [Zhang *et al.*, 2022; Bae *et al.*, 2023] have achieved promising results, their performance still lags behind that of heavyweight networks. This paper aims to develop lightweight CNN-Transformer models that surpass state-of-the-art architectures with the least trainable parameters. To achieve this goal, we present MonoMixer, which merges the advantages of CNNs and Transformers to create a lightweight yet powerful network for self-supervised monocular depth estimation.

3.2 DepthNet Architecture

As illustrated in Figure 2, we design our DepthNet as an encoder-decoder architecture.

Depth Encoder. The proposed MonoMixer extracts multi-scale local features across five stages. Given the current input image $\mathbf{I} \in \mathbb{R}^{3\times h\times w}$, where 3, h, w denote the RGB channels, height, width of $\mathbf{I}$ respectively, we adopt stacked

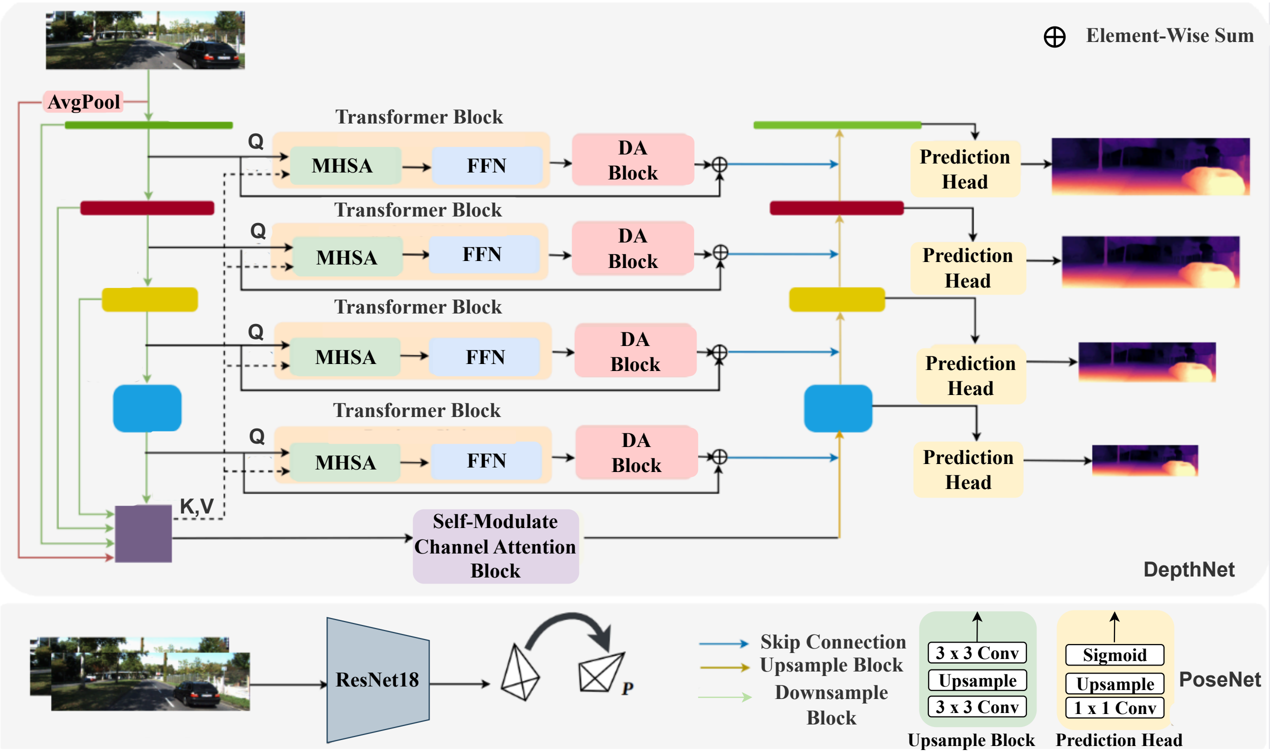

Figure 2: **Overview of our MonoMixer architecture.** Our MonoMixer contains two parts, encoder-decoder DepthNet and PoseNet. The encoder of DepthNet is composed of a CNN and global-guided Transformer (G2T) block. The details-augmented (DA) block dynamically extract rich local details. The self-modulated channel attention (SMCA) block captures global context information and refine salient features. For pose estimation between temporally adjacent frames, we adopt a lightweight PoseNet as in previous work [Godard *et al.*, 2018].

MBConv blocks [Sandler *et al.*, 2018] to generate a set of local tokens $\mathbf{F} = \{\mathrm{F}_1, \mathrm{F}_2, ...\mathrm{F}_N\}$, where N represents the number of scales and $\mathbf{F_n} \in \mathbb{R}^{c_n \times \frac{h}{2^n} \times \frac{w}{2^n}}$. It is worth noting that our intention is not to acquire abundant semantics and a large receptive field, but rather to use fewer blocks to construct a feature pyramid. Afterwards, to further reduce the computational cost, we exploit the average pooling operator to obtain a series of pooled tokens $\mathbf{P} = \{\mathrm{P}_1, \mathrm{P}_2, ...\mathrm{P}_N\}$, where $\mathbf{P}_n \in \mathbb{R}^{c_n \times \frac{h}{32} \times \frac{w}{32}}$. The pooled tokens from different stages have the same size, and they are concatenated along the channel dimension to obtain a global semantic token $\mathbf{G} \in \mathbb{R}^{C \times \frac{h}{32} \times \frac{w}{32}}$, where $C = \sum_{n=1}^{N} c_n$. The global semantic token $\mathbf{G}$ and local tokens F_n will be together as the inputs of global-guided Transformer.

Global-Guided Transformer. The global-guided Transformer (G2T) is composed of a few stacked Transformer blocks. The number of Transformer blocks is L. Each Transformer block consists of the Multi-Head Attention (MHA) block, the Feed-Forward Network (FFN) and residual connections. In MHA, G2T computes the cross-attention between the global semantic token $\mathbf{G}$ and each local token F_n. Specifically, we first use 1×1 convolution layers to linearly project $\mathbf{G}$ and F_n to obtain the same dimensional query $\mathbf{Q}_f = \mathrm{F}_n\mathbf{W}_q$, key $\mathbf{K}_g = \mathbf{G}\mathbf{W}_k$, and value $\mathbf{V}_g = \mathbf{G}\mathbf{W}_v$, where $\mathbf{W}_q$, $\mathbf{W}_k$, and $\mathbf{W}_v$ are weight matrices. Next, the output feature $\mathbf{Q}_o$ is obtained by computing the cross-attention:

$$\mathbf{Q_o} = \mathrm{Attention}(\mathbf{Q_f}, \mathbf{K_g}, \mathbf{V_g}) \tag{1}$$

$$= \mathrm{Softmax}(\frac{\mathbf{Q_f}\mathbf{K_g}^\top}{\sqrt{d}})\mathbf{V_g}, \tag{2}$$

where d is the channel dimension of $\mathbf{K_g}$. Then, we exploit FFN with two 1×1 convolution layers and a GELU [Hendrycks and Gimpel, 2016] activation to refine the output feature $\mathbf{X}$:

$$\mathbf{X} = \mathrm{FFN}(\mathrm{BN}(\mathbf{Q_o})) + \mathbf{Q_o}, \tag{3}$$

where BN denotes the batch normalization [Ioffe and Szegedy, 2015]. Since the proposed model employs a lightweight CNN to maintain low computational complexity, the low-level and mid-level features lack sufficient global information. Nevertheless, the cross-attention mechanism introduces global information into each local token, enabling the G2T to learn multi-level features that incorporate global information.

Details-Augmented Block. One of the main limitations with CNN models is that they can yield a significant loss in the details of the objects in the complex scene [Bronstein *et al.*, 2016]. Besides, due to the inherent locality of convolution operation, which even break the topological structure

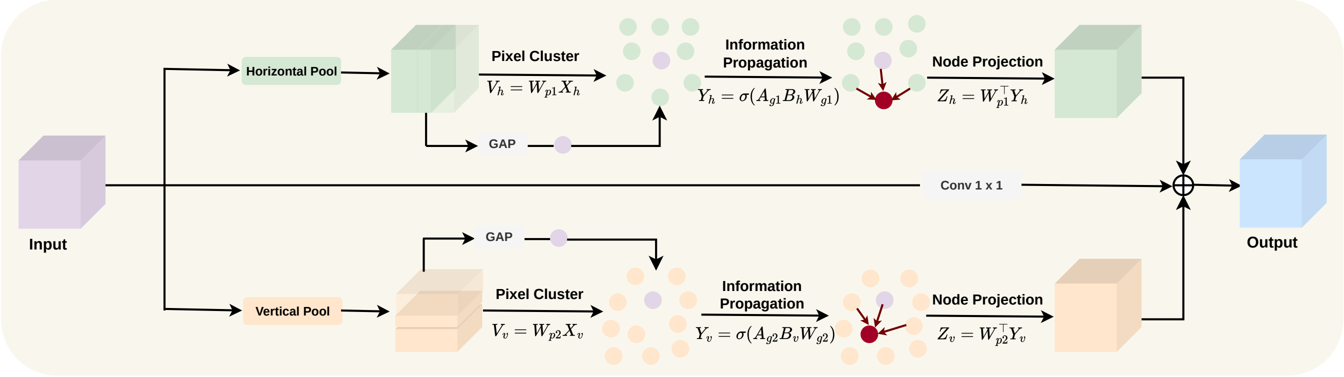

Figure 3: The implementation details of the proposed details-augmented (DA) block.

of the scene [Masoumian *et al.*, 2021]. In contrast, graph convolutional networks (GCNs) can represent the topological structure of the scene by modeling the relationships between nodes. This helps the model to capture global details of the scene and boost the quality of predicted depth maps. Towards this end, we propose a lightweight details-augmented (DA) block to capture finer details. Figure 3 illustrates the implementation diagram of our proposed DA block. It comprises three branches. Besides the linear projection shown in the middle, two other branches are responsible for mixing features along horizontal and vertical directions, respectively. Let $\mathbf{X} \in \mathbb{R}^{d_t \times h_t \times w_t}$ represents the output feature of G2T block. In the horizontal mixing branch, we first employ horizontal pool operation to obtain row tokens, and then use global average pooling to generate global row token. Next, we concatenate them along spatial dimension to obtain aggregated horizontal features $\mathbf{X}_\mathrm{h} \in \mathbb{R}^{(w_t+1)\times d_t}$. To be specific, our DAB is mainly composed of three phases. 1) pixel cluster; 2) information propagation; 3) node projection.

Phase 1, Pixel Cluster. Pixel cluster seeks to map pixel features from the geometric domain into the graph domain, with each node serving as an implicit visual center for a cluster of pixel features. To comply this goal, we initially exploit the trainable transformation matrix, which can be formalized as:

$$\mathbf{V}_h = \mathbf{W}_{p1}\mathbf{X}_h, \tag{4}$$

where $\mathbf{W}_{p1} \in \mathbb{R}^{M\times(w_t+1)}$ is the learnable weight matrix, M denotes the number of total nodes.

Phase 2, Information Propagation. After mapping the pixel features from geometric domain into the graph domain, we construct a graph where each node represents discrete region feature. Based on this graph, we exploit a single-layer graph convolution network to perform information propagation, which can be described as:

$$\mathbf{Y}_h = \sigma(\mathbf{A}_{g1}\mathbf{B}_h\mathbf{W}_{g1}), \tag{5}$$

where $\mathbf{B}_h = \sigma(\mathbf{V}_h) \in \mathbb{R}^{M\times d_t}$, σ is the GELU activation function. The adjacency matrix $\mathbf{A}_{g1} \in \mathbb{R}^{M\times M}$ is randomly initialized and learned by gradient decent during training. $\mathbf{W}_{g1} \in \mathbb{R}^{d_t\times d_t}$ represents the learnable state update matrix. In addition, we introduce an identity matrix $\mathbf{E} \in \mathbb{R}^{M\times M}$ to assuage the obstruction during the model optimization process. The Eqn. 5 can be reformulated as:

$$\mathbf{Y}_h = \sigma(((\mathbf{E} - \mathbf{A}_{g1})\mathbf{B}_h)\mathbf{W}_{g1}), \tag{6}$$

where the first step $(\mathbf{E} - \mathbf{A}_{g1})$ in information propagation phase performs Laplacian smoothing [Zhu and Koniusz, 2021]. The evolved global representations $\mathbf{Y}_h \in \mathbb{R}^{M\times d_t}$ can further strength the capability of local feature representations. This helps model to capture finer details.

Phase 3, Node Projection. After information propagation, we project the output feature from the graph domain back into the geometry domain. In light of the inverse relationship between pixel cluster and node projection, and with the purpose of reducing model parameters, we adopt the transpose of $\mathbf{W}_{p1}$ for node projection. Given the node representation $\mathbf{Y}_h \in \mathbb{R}^{M\times d_t}$, the output feature can be formulated as:

$$\mathbf{Z}_h = \mathbf{W}_{p1}^\top \mathbf{Y}_h. \tag{7}$$

Given that the global row token primarily serves to facilitate interactions during the model training stage, we remove it to obtain the horizontal output feature $\mathbf{Z}_h \in \mathbb{R}^{d_t\times 1\times w_t}$. Similar operation is applied in the vertical branch and obtain the vertical output feature $\mathbf{Z}_v \in \mathbb{R}^{d_t\times h_t\times 1}$. Finally, the output feature from the three branches are fused together to produce an output tensor which has the same size as the input tensor $\mathbf{X}$. We implement this fusion block with element-wise addition and a 1 × 1 convolution layer:

$$\mathbf{X}_o = \mathbf{Z}_h + \mathrm{Conv}_{1\times1}(\mathbf{X}) + \mathbf{Z}_v. \tag{8}$$

Compared to the local interaction of convolution operation, since each node of DA block is an enhanced semantic representation for a cluster of image patches, DA block can augment feature representation and extract finer local details.

Self-Modulated Channel Attention Block. In depth estimation, each channel map can be considered a region-specific response, and different regional responses being interrelated. If each feature map obtains more distinct regional responses from all other feature maps, it will acquire more relative depth cues from distant regions and greatly heighten the perception of scene structure [Yan *et al.*, 2021]. Therefore, we propose a self-modulated channel attention (SMCA) block to capture cross-dependencies between channel-wise feature maps

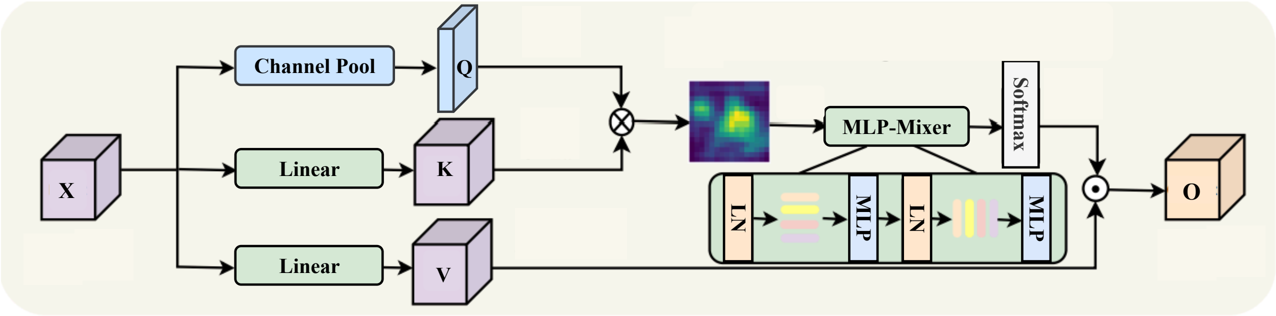

Figure 4: The implementation details of the proposed self-modulated channel attention (SMCA) block.

and accentuate important region responses. As illustrated in Figure 4, given the global semantic token $\mathbf{G} \in \mathbb{R}^{C\times H\times W}$, we firstly reshape $\mathbf{G}$ to $\mathbb{R}^{C\times HW}$, then exploit the channel pool operation to obtain global channel map $\mathbf{Q} \in \mathbb{R}^{1\times HW}$, and linearly project the $\mathbf{G}$ to generate $\mathbf{K} \in \mathbb{R}^{C\times HW}$ and $\mathbf{V} \in \mathbb{R}^{C\times HW}$. Next, we perform a matrix multiplication between global channel map $\mathbf{Q}$ and the transpose of $\mathbf{K}$ to compute the global context score $\mathbf{S}$ to $\mathbb{R}^{C}$. The global context score indicates the relationships between the global channel map and local feature maps *i.e.* any channel has higher score means that it has powerful responses to a specific region. To update the channel maps, we group $\mathbf{S}$ into g groups and apply MLPMixer [Tolstikhin *et al.*,] to propagate global information among channels. Specifically, our MLPMixer consists of two consecutive MLPs. We can updates $\mathbf{S} \in \mathbb{R}^{g\times d}$ with the MLPMixer by computing:

$$\mathbf{S}' = \mathbf{S} + \mathrm{MLP}_1(\mathrm{LayerNorm}(\mathbf{S})^{\top})^{\top}, \tag{9}$$

$$\bar{\mathbf{S}} = \mathbf{S}' + \mathrm{MLP}_2(\mathrm{LayerNorm}(\mathbf{S}')), \tag{10}$$

where the first MLP is responsible for exchanging information between each group, and the second is used to mix channel-wise feature. After updating the channel features, we convert the $\bar{\mathbf{S}}$ to original dimension and apply a softmax layer to obtain the channel attention map $\mathbf{A}_c \in \mathbb{R}^{C}$:

$$\mathbf{A}_{c,i} = \frac{\exp(\bar{\mathbf{S}}_i)}{\sum_{j=1}^{C}\exp(\bar{\mathbf{S}}_j)}. \tag{11}$$

Finally we perform an element-wise multiplication operation between channel attention map $\mathbf{A}_c$ and value $\mathbf{V}$ to obtain the final output $\mathbf{G}' \in \mathbb{R}^{C\times H\times W}$ as follows:

$$\mathbf{G}' = \mathbf{V} \odot \mathbf{A}_c. \tag{12}$$

Depth Decoder. MonoMixer departs from complex upsampling techniques and attention mechanisms by adopting a simpler, more efficient depth decoder inspired by [Godard *et al.*, 2018]. As shown in Figure 2, this decoder progressively enlarges spatial dimensions through bilinear interpolation and merges features from different encoder stages using convolutional layers.

3.3 PoseNet

In line with previous works [Godard *et al.*, 2018; Zhao *et al.*, 2022], this paper employs the same PoseNet architecture for pose estimation. Specifically, a pre-trained ResNet18 backbone processes a pair of color images to extract features, which are subsequently fed into a pose decoder composed of four convolutional layers to predict the relative 6-DoF pose between the image pair.

3.4 Self-Supervised Learning

Different from the supervised learning that exploits ground truth of depth this work casts depth estimation as the task of image reconstruction. Specifically, given two images I_t and I_s from different viewpoints. A synthesized target image $\hat{I}_t$ is obtained by translating the image I_s according to the predicted depth D_t, the relative position $P_{t\to s}$ and the intrinsic K:

$$\hat{I}_t = I_s\langle F(D_t, P_{t\to s}, K)\rangle, \tag{13}$$

where $P_{t\to s}$ is the predicted position by the PoseNet. Then, we use the disparity L_d between the synthesized image $\hat{I}_t$ and the original target image I_t to measure the accuracy of the depth D_t:

$$\mathcal{L}_d = \frac{\lambda}{2}(1 - \mathrm{SSIM}(\hat{I}_t, I_t)) + (1-\lambda)\|\hat{I}_t - I_t\|, \tag{14}$$

where λ is a hyperparameter that controls the weight of the two similarity metrics and SSIM denotes the (Structural Similarity Index. In addition, we exploit an edge-aware smoothness loss to smooth the produced disparity:

$$\mathcal{L}_s = |\partial_x d_t^*|e^{-|\partial_x I_t|} + |\partial_x d_t^*|e^{-|\partial_y I_t|}, \tag{15}$$

where $d_t^* = \frac{d_t}{\hat{d}_t}$ represents the mean-normalized inverse depth. The total loss can be defined as:

$$\mathcal{L} = \mathcal{L}_d + \alpha\mathcal{L}_s, \tag{16}$$

where α is the weight of edge-aware smoothness regulation.

4 Experiments

In this section, we evaluates the proposed framework h on three public datasets including KITTI [Thinh *et al.*, 2020], Cityscapes [Cordts *et al.*, 2016] and Make3D [Saxena *et al.*, 2009], significantly demonstrating the superiority of MonoMixer at estimating depth.

Method	Year	Data	Depth Error (↓)				Depth Accuracy (↑)			Model Size (↓)
			Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$	Params.
Monodepth2 [Godard *et al.*, 2018]	ICCV2019	M	0.115	0.903	4.863	0.193	0.877	0.959	0.981	14.3M
HR-Depth [Lyu *et al.*, 2021]	AAAI2021	M	0.109	0.792	4.632	0.185	0.884	0.962	0.983	14.7M
Lite-HR-Depth [Lyu *et al.*, 2021]	AAAI2021	M	0.116	0.845	4.841	0.190	0.866	0.957	0.982	3.1M
MonoViT [Zhao *et al.*, 2022]	3DV2022	M	0.099	0.708	4.372	0.175	0.900	0.967	0.984	10.3M
MonoFormer [Bae *et al.*, 2023]	AAAI2023	M	0.108	0.806	4.594	0.184	0.884	0.963	0.983	23.9M
ROIFormer [Xing *et al.*, 2023]	AAAI2023	M	0.103	0.695	4.438	0.178	0.889	0.964	0.984	-
Lite-Mono-8M [Zhang *et al.*, 2022]	CVPR2023	M	0.101	0.729	4.454	0.178	0.897	0.965	0.983	8.7M
DaCCN [Han *et al.*, 2023]	ICCV2023	M	0.099	0.661	4.316	0.173	0.897	0.967	0.985	13.0M
R-MSFMX3 [Zhou *et al.*, 2024]	TPAMI2024	M	0.111	0.775	4.666	0.900	0.879	0.960	0.981	5.0M
AQUANet [Bello *et al.*, 2024]	TIP2024	M	0.105	0.621	4.227	0.179	0.889	0.964	0.984	25.0M
SQLdepth [Wang *et al.*, 2024]	AAAI2024	M	0.094	0.697	4.320	0.172	0.904	0.967	0.984	34.0M
RPrDepth [Han and Shen, 2024]	ECCV2024	M	0.097	0.658	4.279	0.169	0.900	0.967	0.985	-
MonoMixer	Ours	M	**0.081**	**0.576**	**4.039**	**0.068**	**0.923**	**0.984**	**0.990**	3.9M
Monodepth2 [Godard *et al.*, 2018]	ICCV2019	M*	0.115	0.882	4.701	0.190	0.879	0.961	0.982	14.3M
MonoViT [Zhao *et al.*, 2022]	3DV2022	M*	0.096	0.714	4.292	0.172	0.908	0.968	0.984	10.3M
ROIFormer [Xing *et al.*, 2023]	AAAI2023	M*	0.100	0.674	4.335	0.175	0.896	0.966	0.983	-
Lite-Mono-8M [Zhang *et al.*, 2022]	CVPR2023	M*	0.097	0.710	4.309	0.174	0.905	0.967	0.984	8.7M
DaCCN [Han *et al.*, 2023]	ICCV2023	M*	0.094	0.624	4.145	0.169	0.909	0.970	0.985	13.0M
R-MSFMX3 [Zhou *et al.*, 2024]	TPAMI2024	M*	0.110	0.788	4.555	0.187	0.883	0.962	0.982	5.0M
SQLdepth [Wang *et al.*, 2024]	AAAI2024	M*	0.087	0.649	4.149	0.165	0.918	0.969	0.984	34.0M
RPrDepth [Han and Shen, 2024]	ECCV2024	M*	0.091	0.612	4.098	0.162	0.910	0.971	0.986	-
MonoMixer	Ours	M*	**0.076**	**0.563**	**4.018**	**0.065**	**0.929**	**0.988**	**0.991**	3.9M

Table 1: **Performance comparison on KITTI [Thinh *et al.*, 2020] benchmark**. In the Data column, M: trained with monocular videos, M*: input resolution 1024 × 320. The best results are in bold, and second best are underlined. For the error-based metrics , the lower value is better; and for the accuracy-based metrics , the higher value is better.

4.1 Implementation Details

For fair comparison, we follow the same training strategies as previous works [Godard *et al.*, 2018; Zhao *et al.*, 2022]. Specifically, We implement our model in Pytorch framework [Paszke *et al.*, 2019]. The model is trained for 20 epochs on a single NVIDIA RTX 3090 GPU, with a batch size of 12. We use the Adam optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.999$ to jointly train both DepthNet and PoseNet. In addition, we employ same data augmentation detailed in [Godard *et al.*, 2018; Zhao *et al.*, 2022]. For evaluation, we adopt the seven standard metrics (AbsRel, SqRel, RMSE, RMSElog, $\delta_1 < 1.25$, $\delta_2 < 1.25^2$, $\delta_3 < 1.25^3$) proposed in [Eigen *et al.*, 2014] as our evaluation criteria, which are commonly used in the depth estimation field.

4.2 Comparison on KITTI

The KITTI dataset [Thinh *et al.*, 2020] is renowned for its comprehensive range of challenges, encompassing optical flow, visual odometry, and semantic segmentation tasks. This has made it a cornerstone for computer vision research. Furthermore, it is considered the defacto standard for benchmarking self-supervised monocular depth estimation methods. We conduct experiments under two different training resolutions. As shown in Table 1, our proposed MonoMixer clearly outperforms the existing SOTA self-supervised methods in all metrics. Compared to baseline model MonoFormer [Bae *et al.*, 2023], our MonoMixer achieves **0.026**, **0.230** and **0.555** gains in terms of AbsRel, SqRel and RMSE, respectively. Additionally, MonoMixer achieves superior performance compared to the recently introduced, carefully designed lightweight models Lite-Mono [Zhang *et al.*, 2022]and R-MSFMX [Zhou *et al.*, 2024]. Compared with the new SQLdepth [Wang *et al.*, 2024] the proposed MonoMixer beats it in all metrics, but the model size is only about one-tenth of this model. Figure 5 illustrates that our model exhibits superior depth estimation capabilities for slender structures, including road signs and poles. Furthermore, our model is able to accurately estimate depth on challenging images where moving objects are close to the camera (column 3). These advancements can be attributed to the enhanced scene and object perception afforded by our self-modulated channel attention (SMCA) block, and the enhanced detail information provided by the details-augmented (DA) block. All the above results demonstrate that MonoMixer outperforms its counterparts by evident margins.

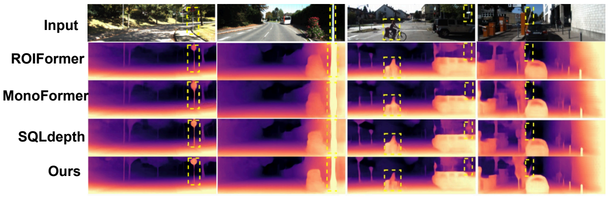

Figure 5: **Qualitative results on the KITTI**. Here are some depth maps generated by ROIFormer [Xing *et al.*, 2023], MonoFormer [Bae *et al.*, 2023], SQLdepth [Wang *et al.*, 2024], and MonoMixer (ours), respectively.

4.3 Comparison on Make3D and Cityscapes

Make3D is a dataset containing monocular RGB images and their corresponding depth maps, primarily used for evaluating the generalization capacity of self-supervised monocular depth estimation models. As shown in Table 2, MonoMixer achieve superior performance compared with other methods, which demonstrates the excellent zero-shot generalization ability of our model. With monocular training and 640 × 192 input, our model attains 0.282 and 6.589 in terms of AbsRel and RMSE with considerable improvements from other SOTA methods.

Method	Abs Rel ↓	Sq Rel ↓	RMSE ↓	RMSE log ↓	Params
Monodepth2 [Godard *et al.*, 2018]	0.322	3.589	7.417	0.163	14.3M
MonoViT [Zhao *et al.*, 2022]	0.286	2.758	6.623	0.147	10.3M
Lite-Mono [Zhang *et al.*, 2022]	0.305	3.060	6.981	0.158	8.7M
DaCCN [Han *et al.*, 2023]	0.290	2.873	6.656	0.149	13.0M
R-MSFM [Zhou *et al.*, 2024]	0.334	3.285	7.212	0.169	5.0M
SQLdepth [Wang *et al.*, 2024]	0.306	2.402	6.856	0.151	34.0M
MonoMixer (Ours)	**0.282**	**2.388**	**6.589**	**0.146**	3.9M

Table 2: Comparison of the proposed MonoMixer to some other methods on the Make3D [Saxena *et al.*, 2009] dataset. All models are trained on KITTI [Thinh *et al.*, 2020] with an image resolution of 640 × 192.

Method	Abs Rel ↓	Sq Rel ↓	RMSE ↓	δ ¡ 1.25 ↑	Params
Monodepth2 [Godard *et al.*, 2018]	0.129	1.569	6.876	0.849	14.3M
Videos in the Wild [Gordon *et al.*, 2019]	0.127	1.330	6.960	0.830	-
Li et al. [Li *et al.*, 2020]	0.119	1.290	6.980	0.846	-
DaCCN [Han *et al.*, 2023]	0.113	1.380	6.305	0.888	13.0M
SQLdepth [Wang *et al.*, 2024]	0.110	1.130	6.264	0.881	34.0M
MonoMixer (Ours)	**0.103**	**1.112**	**6.178**	**0.896**	3.9M

Table 3: Comparison of the proposed MonoMixer to some other methods on the Cityscapes [Cordts *et al.*, 2016] dataset.

Cityscapes is a challenging dataset which comprises multitudinous moving objects. We train MonoMixer from scratch on the dataset under the same setting with other methods. As shown in Table 3, our MonoMixer significantly outperforms other state-of-the-art models on this dataset.

4.4 Efficiency

The proposed method is evaluated on a NVIDIA TITAN Xp and a Jetson Xavier and are compared with more SOTA models. As shwon in Table 4, our proposed model attains a good balance between model size and inference speed. Notice that MonoMixer outperforms the recent well-designed lightweight models Lite-Mono [Zhang *et al.*, 2022] and R-MSFMX [Zhou *et al.*, 2024] both in speed and accuracy (Table 1).

	Full Model		Speed (ms)	
Method	Params. (M)	FLOPs (G)	Titan XP	Jetson Xavier
Monodepth2	14.3	8.0	3.8	14.3
MonoViT	10.3	23.7	13.5	47.4
Lite-Mono	8.7	11.2	6.5	32.2
DaCCN	13.0	4.3	3.7	12.8
DIFFNet	12	2.3	3.6	12.3
MonoMixer (Ours)	3.9	4.1	**2.6**	**10.5**

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.

Architecture	Params	Speed(ms)	Abs Rel↓	RMSE ↓	$\delta < 1.25 \uparrow$
MonoMixer	3.886M	2.6	**0.081**	**4.039**	**0.923**
w/o G2T block	3.803M	2.5	0.099	4.182	0.899
w/o DA block	3.876M	2.6	0.091	4.097	0.902
w/o SMCA block	3.864M	2.6	0.095	4.086	0.904

Table 5: All the models are trained and tested on KITTI with the input size 640 × 192.

Method	Abs Rel ↓	RMSE ↓	$\delta < 1.25 \uparrow$
baseline	**0.081**	**4.039**	**0.923**
G2T w/o transformer	0.091	4.146	0.907
G2T w/self-attn	0.086	4.097	0.913
DA w/o horizontal branch	0.085	4.067	0.906
DA w/o vertical branch	0.082	4.075	0.906
DA w/o identity branch	0.087	4.042	0.907
SMCA w/o MLPMixer	0.086	4.059	0.908

Table 6: Ablation studies on the role of different components of the core module in MonoMixer.

4.5 Ablation Study

In this section, we conduct several ablation studies on the KITTI dataset to validate the effectiveness of designs in MonoMixer, including global-guided transformer (G2T) block, details-augmented (DA) block and self-modulated channel attention (SMCA) block, respectively.

The benefit of G2T block. As shown Table 5, when the G2T blocks are removed, the accuracy quickly drops. The proposed G2T block is essential for enabling MonoMixer to capture long-range global contexts, overcoming the inherent drawback that CNNs can only extract local features.

The benefit of DA block. As illustrated in Table 5, accuracy decreases when the DA blocks are removed. Note that the DA block only adds Negligible additional parameters (0.004M), demonstrating the improvements benefit from the better local details rather than an increase in network complexity.

The benefit of SMCA block. SMCA block is responsible for capturing global contexts of scene structure and extracting informative feature. As shown in Table 5, our SMCA block improves the performance on all the metrics.

The effect of components of G2T block. The results are shown in Table 6. It is clear that the role of the transformer is essential for the accuracy of G2T. In addition, compared to using self-attention, G2T with cross-attention can attain better performance.

The effect of components of DA block. As shown in Table 6, when any branch is removed, the accuracy decreases. This indicates that every branch learns specific finer details of scene structure.

The effect of components of SMCA block. The influence of the MLPMixer in the proposed SMCA block on the accuracy is studied. As shown in Table 6, MLPMixer plays an important role in the SMCA block.

5 Conclusion

In this work, we present a novel architecture MonoMixer for efficient self-supervised monocular depth estimation. The proposed hybrid architecture effectively integrates the strengths of CNNs and Transformers, enabling it to capture both fine-grained local details and long-range global contexts. Experimental results show that the proposed models have achieved significant improvements on three prevalent datasets and attain a new state-of-the-art performance.

References

[Bae *et al.*, 2023] Jinwoo Bae, Sungho Moon, and Sunghoon Im. Deep digging into the generalization of self-supervised monocular depth estimation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2023.

[Bello *et al.*, 2024] Juan Luis Gonzalez Bello, Jaeho Moon, and Munchurl Kim. Self-supervised monocular depth estimation with positional shift depth variance and adaptive disparity quantization. *IEEE Transactions on Image Processing*, 2024.

[Bronstein *et al.*, 2016] Michael M. Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, and Pierre Vandergheynst. Geometric deep learning: Going beyond euclidean data. *IEEE Signal Process. Mag.*, 2016.

[Casser *et al.*, 2019] Vincent Casser, Soeren Pirk, Reza Mahjourian, and Anelia Angelova. Unsupervised monocular depth and ego-motion learning with structure and semantics. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2019.

[Cordts *et al.*, 2016] 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. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[Dosovitskiy *et al.*, 2020] 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. *ArXiv*, 2020.

[Eigen and Fergus, 2014] David Eigen and Rob Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. *2015 IEEE International Conference on Computer Vision (ICCV)*, 2014.

[Eigen *et al.*, 2014] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. In *Neural Information Processing Systems*, 2014.

[Fu *et al.*, 2018] Huan Fu, Mingming Gong, Chaohui Wang, K. Batmanghelich, and Dacheng Tao. Deep ordinal regression network for monocular depth estimation. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018.

[Garg *et al.*, 2016] Ravi Garg, B. V. Kumar, G. Carneiro, and Ian D. Reid. Unsupervised cnn for single view depth estimation: Geometry to the rescue. In *European Conference on Computer Vision*, 2016.

[Godard *et al.*, 2016] Clément Godard, Oisin Mac Aodha, and Gabriel J. Brostow. Unsupervised monocular depth estimation with left-right consistency. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[Godard *et al.*, 2018] Clément Godard, Oisin Mac Aodha, and Gabriel J. Brostow. Digging into self-supervised monocular depth estimation. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2018.

[Gordon *et al.*, 2019] A. Gordon, Hanhan Li, Rico Jonschkowski, and Anelia Angelova. Depth from videos in the wild: Unsupervised monocular depth learning from unknown cameras. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019.

[Guo *et al.*, 2018] Xiaoyang Guo, Hongsheng Li, Shuai Yi, Jimmy S. J. Ren, and Xiaogang Wang. Learning monocular depth by distilling cross-domain stereo networks. In *European Conference on Computer Vision*, 2018.

[Han and Shen, 2024] Wencheng Han and Jianbing Shen. High-precision self-supervised monocular depth estimation with rich-resource prior. In *European Conference on Computer Vision*, 2024.

[Han *et al.*, 2023] Wencheng Han, Junbo Yin, and Jianbing Shen. Self-supervised monocular depth estimation by direction-aware cumulative convolution network. *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

[Hendrycks and Gimpel, 2016] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv: Learning*, 2016.

[Ioffe and Szegedy, 2015] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *ArXiv*, 2015.

[Jung *et al.*, 2021] Hyun-Joo Jung, Eunhyeok Park, and Sungjoo Yoo. Fine-grained semantics-aware representation enhancement for self-supervised monocular depth estimation. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[Li *et al.*, 2020] Hanhan Li, A. Gordon, Hang Zhao, Vincent Casser, Anelia, and Angelova. Unsupervised monocular depth learning in dynamic scenes. In *Conference on Robot Learning*, 2020.

[Lyu *et al.*, 2021] Xiaoyang Lyu, L. Liu, Mengmeng Wang, Xin Kong, Lina Liu, Yong Liu, Xinxin Chen, and Yi Yuan. Hr-depth: High resolution self-supervised monocular depth estimation. In *AAAI Conference on Artificial Intelligence*, 2021.

[Masoumian *et al.*, 2021] Armin Masoumian, Hatem A. Rashwan, Saddam Abdulwahab, Julián Cristiano, and Domenec Puig. Gcndepth: Self-supervised monocular depth estimation based on graph convolutional network. *Neurocomputing*, 2021.

[Paszke *et al.*, 2019] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. *ArXiv*, 2019.

[Ranftl *et al.*, 2021] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[Rudolph *et al.*, 2022] Michael Bernard Rudolph, Youssef Dawoud, Ronja Guldenring, Lazaros Nalpantidis, and Vasileios Belagiannis. Lightweight monocular depth estimation through guided decoding. *2022 International Conference on Robotics and Automation (ICRA)*, 2022.

[Sandler *et al.*, 2018] Mark Sandler, Andrew G. Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018.

[Saxena *et al.*, 2009] Ashutosh Saxena, Min Sun, and A. Ng. Make3d: Learning 3d scene structure from a single still image. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2009.

[Shyam *et al.*, 2024] Pranjay Shyam, Alexandre Okon, and Hyunjin Yoo. Enhancing self-supervised monocular depth estimation via piece-wise pose estimation and geometric constraints. *2024 IEEE/CVF Winter Conference on Applications of Computer Vision Workshops (WACVW)*, 2024.

[Thinh *et al.*, 2020] Nguyen Hong Thinh, Tran Hoang Tung, and Le Anh Vu Ha. Depth-aware salient object segmentation. *VNU Journal of Science: Computer Science and Communication Engineering*, 2020.

[Tolstikhin *et al.*,] Ilya O. Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Daniel Keysers, Jakob Uszkoreit, Mario Lucic, and Alexey Dosovitskiy. Mlp-mixer: An all-mlp architecture for vision. *ArXiv*.

[Wang and Cheng, 2023] Fei Wang and Jun Cheng. Hqdec: Self-supervised monocular depth estimation based on a high-quality decoder. *IEEE Transactions on Circuits and Systems for Video Technology*, 2023.

[Wang *et al.*, 2024] Youhong Wang, Yunji Liang, Hao Xu, Shaohui Jiao, and Hongkai Yu. Sqldepth: Generalizable self-supervised fine-structured monocular depth estimation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 2024.

[Wofk *et al.*, 2019] Diana Wofk, Fangchang Ma, Tien-Ju Yang, Sertac Karaman, and Vivienne Sze. Fastdepth: Fast monocular depth estimation on embedded systems. *2019 International Conference on Robotics and Automation (ICRA)*, 2019.

[Xing *et al.*, 2023] Daitao Xing, Jinglin Shen, Chiuman Ho, and Anthony Tzes. Roiformer: Semantic-aware region of interest transformer for efficient self-supervised monocular depth estimation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2023.

[Yan *et al.*, 2021] Jiaxing Yan, Hong Zhao, Penghui Bu, and Yusheng Jin. Channel-wise attention-based network for self-supervised monocular depth estimation. *2021 International Conference on 3D Vision (3DV)*, 2021.

[Yang *et al.*, 2021] Guanglei Yang, Hao Tang, Mingli Ding, N. Sebe, and Elisa Ricci. Transformer-based attention networks for continuous pixel-wise prediction. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[Zhang *et al.*, 2022] Ning Zhang, Francesco Nex, George Vosselman, and Norman Kerle. Lite-mono: A lightweight cnn and transformer architecture for self-supervised monocular depth estimation. *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[Zhang *et al.*, 2023] Youming Zhang, Xianda Guo, Matteo Poggi, Zheng Zhu, Guan Huang, and S. Mattoccia. Completionformer: Depth completion with convolutions and vision transformers. *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[Zhao *et al.*, 2022] Chaoqiang Zhao, Youming Zhang, Matteo Poggi, Fabio Tosi, Xianda Guo, Zheng Zhu, Guan Huang, Yang Tang, and S. Mattoccia. Monovit: Self-supervised monocular depth estimation with a vision transformer. *2022 International Conference on 3D Vision (3DV)*, 2022.

[Zhou *et al.*, 2024] Zhongkai Zhou, Xinnan Fan, Pengfei Shi, Yuanxue Xin, Dongliang Duan, and Liuqing Yang. Recurrent multiscale feature modulation for geometry consistent depth learning. *IEEE transactions on pattern analysis and machine intelligence*, 2024.

[Zhu and Koniusz, 2021] Hao Zhu and Piotr Koniusz. Simple spectral graph convolution. In *International Conference on Learning Representations*, 2021.