Deep Digging into the Generalization of Self-Supervised Monocular Depth Estimation

Jinwoo Bae[1], **Sungho Moon**[1], **and Sunghoon Im**[1]

[1] Department of Electrical Engineering and Computer Science, DGIST, Daegu, Korea
{sjg02122, byeol3325, sunghoonim}@dgist.ac.kr

Abstract

Self-supervised monocular depth estimation has been widely studied recently. Most of the work has focused on improving performance on benchmark datasets, such as KITTI, but has offered a few experiments on generalization performance. In this paper, we investigate the backbone networks (*e.g.* CNNs, Transformers, and CNN-Transformer hybrid models) toward the generalization of monocular depth estimation. We first evaluate state-of-the-art models on diverse public datasets, which have never been seen during the network training. Next, we investigate the effects of texture-biased and shape-biased representations using the various texture-shifted datasets that we generated. We observe that Transformers exhibit a strong shape bias and CNNs do a strong texture-bias. We also find that shape-biased models show better generalization performance for monocular depth estimation compared to texture-biased models. Based on these observations, we newly design a CNN-Transformer hybrid network with a multi-level adaptive feature fusion module, called MonoFormer. The design intuition behind MonoFormer is to increase shape bias by employing Transformers while compensating for the weak locality bias of Transformers by adaptively fusing multi-level representations. Extensive experiments show that the proposed method achieves state-of-the-art performance with various public datasets. Our method also shows the best generalization ability among the competitive methods.

1 Introduction

How do humans efficiently extract and recognize essential information from complex scenes? The biological vision system treats the object's shape as the single most crucial vision cue, compared with other cues like texture or color (Landau, Smith, and Jones 1988). This enables humans, even small children, to easily recognize an object from a line drawing or a silhouette image. It is widely known that convolutional neural networks (CNNs) are designed with inspiration from the biological neural networks in living organisms (Hubel and Wiesel 1959; Fukushima 1988; Kriegeskorte 2015). CNNs extract the simple patterns (*e.g.* edges) and then build complex patterns by successively composing early neural responses. However, in contrast to human visual representation, recent researches (Geirhos et al. 2019; Morrison et al. 2021; Tuli et al. 2021) have revealed that CNNs are strongly biased towards recognizing textures rather than shapes. CNN-based models rationally classify labels even in images with disrupted shape structures (Gatys, Ecker, and Bethge 2017; Brendel and Bethge 2019). On the other hand, CNN models fail to predict labels correctly in a texture-removed image whose shape is well-preserved (Ballester and Araujo 2016).

Then, how does this observation affect the monocular depth estimation task? Over the past decade, monocular depth estimation has made significant progress using CNNs (Xiong et al. 2021; Yin and Shi 2018; Zhou et al. 2021; Godard et al. 2019; Guizilini et al. 2020a; Casser et al. 2019). These works show the remarkable performance on the KITTI datasets (Geiger et al. 2013) even with the model trained in a self-supervised manner. However, the experiments have been conducted on only a few driving scenes, mostly KITTI datasets, so the generality of these methods has not been closely studied. In this paper, we study the generalization performance of the state-of-the-art methods and investigate how texture-biased representation from CNNs affects monocular depth estimation. We evaluate state-of-the-art models trained on KITTI using six public depth datasets (SUN3D, RGBD, MVS, Scenes11, ETH3D, and Oxford Robotcar). We also conduct experiments on three different texture-shifted datasets including texture-smoothed (Watercolor), textureless (Pencil-sketch), and texture-transferred (Style-transfer) images. Through these extensive experiments, we determine that texture-biased models are vulnerable to generality in monocular depth estimation.

Recently, Transformers (Dosovitskiy et al. 2020) have received a surge of attention for their outstanding performance in the field of computer vision (Carion et al. 2020), despite the lack of a spatial locality bias. Moreover, several works (Zhang et al. 2022; Morrison et al. 2021; Park and Kim 2022) show that Transformers have a strong shape bias, unlike CNNs. We also investigate the Transformers, similar to the experiments conducted for CNNs, and observe that shape bias is key to generalize depth estimation. Thus, we propose a CNN-Transformer hybrid network, called MonoFormer, which are highly complementary to each other. The design intuition behind MonoFormer is to take the strong

shape bias of Transformers and the spatial locality bias of low-level Transformers features projected from CNN features. To do so, we design a layer-wise Attention Connection Module (ACM) and a Feature Fusion Decoder (FFD). The ACM measures the importance of shape bias representation and the local details, and then the FFD adaptively fuses them for depth prediction. The detailed ablation studies show that the shape-biased features are mostly extracted from high-level Transformers and the local details are captured at low layers.

To verify the generality, we evaluate our KITTI-trained model on the six out-of-distribution datasets. These experiments show MonoFormer achieves performance improvement of up to more than 30% over other CNN-based state-of-the-art models (Godard et al. 2019; Zhou et al. 2021; Guizilini et al. 2020a), 7% over a Transformer-based model (Dosovitskiy et al. 2020), and 15% over a conventional hybrid model (Yang et al. 2021). Our model shows strong robustness and generality regardless of the testing distributions. By investigating the network structures, we observe that the CNNs mostly learn texture-based representation while Transformers nearly learn shape-based representation. We also reveal that the shape-biased models achieve superior generalization ability compared with texture-biased models on out-of-distribution training datasets. Our contributions can be summarized as follows:

- We investigate the representation learned by CNNs, Transformers, and hybrid models for monocular depth estimation using various public datasets and stylized datasets.
- We propose a CNN-Transformer hybrid network with multi-level feature aggregation, which complements the shape bias and spatial locality bias toward the generalization of monocular depth estimation.
- Extensive experiments demonstrate the effectiveness of the proposed method, and our method achieves state-of-the-art performance on KITTI datasets, diverse out-of-distribution datasets, and texture-shifted datasets.

2 Related Work

2.1 Self-Supervised Monocular Depth Estimation

Self-supervised depth estimation methods (Zhou et al. 2017; Godard et al. 2019; Guizilini et al. 2020a; Lyu et al. 2021; Klingner et al. 2020; Xiong et al. 2021) simultaneously train depth and motion network by imposing photometric consistency loss between target and source images warped by the predicted depth and motion. Monodepth2 (Godard et al. 2019) presents a minimum reprojection loss to handle occlusions, a full-resolution multi-scale sampling method to reduce visual artifacts, and an auto-masking loss to ignore outlier pixels. PackNet-SfM (Guizilini et al. 2020a) introduces packing and unpacking blocks that leveraged 3D convolutions to learn the dense appearance and geometric information in real-time. HR-Depth (Lyu et al. 2021) analyzes the reason for the inaccurate depth prediction in large gradient regions and designed a skip connection to extract representative features in high resolution.

2.2 Vision Transformers

Recently, Transformers (Vaswani et al. 2017) start to show promises for solving computer vision tasks such as image classification (Dosovitskiy et al. 2020; Touvron et al. 2021), object detection (Carion et al. 2020), and dense prediction. (Zheng et al. 2021; Ranftl, Bochkovskiy, and Koltun 2021; Yang et al. 2021; Guizilini et al. 2022). ViT (Dosovitskiy et al. 2020) employs Transformers architecture on fixed-size image patches for image classification for the first time. DeiT (Touvron et al. 2021) utilizes Knowledge distinction on ViT architecture, showing good performance only with the ImageNet dataset. Some works (Ranftl, Bochkovskiy, and Koltun 2021; Yang et al. 2021) have employed Transformers for monocular depth estimation in a supervised manner. TransDepth (Yang et al. 2021) utilizes multi-scale information to capture local level details. These works (Zheng et al. 2021; Yang et al. 2021) only focus on improving performance on benchmark datasets. Previous works lack studies on whether models behave as intended in another domain dataset.

3 Method

3.1 CNN-Transformer Encoder

The encoder consists of a CNN and Transformers. We use ResNet50 (He et al. 2016) as the CNN backbone ($E(\theta)$ in Fig. 1), and L number of Transformers. In this work, we set the L as 4. An input image I passes through the CNN encoder to extract a feature map $F \in \mathbb{R}^{C\times H\times W}$, then the map is divided into N $(= \frac{H}{16} \times \frac{W}{16})$ number of patches $p_n \in \mathbb{R}^{C\times 16\times 16}$, which is utilized as the input of the first Transformer layer. We additionally use a special token t_s following the work (Ranftl, Bochkovskiy, and Koltun 2021). We input the patch tokens $p_n,\ n \in \{1, ..., N\}$ and the special token t_s with a learnable linear projection layer E as follows:

$$Z_0 = [t_s;\ p_1E;\ p_2E;\ ...\ ;\ p_NE], \tag{1}$$

where Z_0 is the latent embedding vector. The Transformer encoder consists of a Multi-head Self-Attention (MSA) layer, a Multi-Layer Perceptron (MLP) layer, and Layer Norm (LN) layers. The MLP is built with GELU nonlinearity (Hendrycks and Gimpel 2016). The LN is applied before every block and residual connections apply after every block. Self-Attention (SA) at each layer $l \in \{1, ..., L\}$ is processed with the learnable parameters $W_Q^m, W_K^m, W_V^m \in \mathbb{R}^{C\times d}$ of {query, key, value} weight matrices, given the embedding vector $Z_l \in \mathbb{R}^{N\times C}$ as follows:

$$\begin{aligned} \mathrm{SA}_{l-1}^m &= \mathrm{softmax}(\frac{Q_{l-1}^m (K_{l-1}^m)^{\mathrm{T}}}{\sqrt{d}}) V_{l-1}^m,\ m \in \{1, ..., M\}, \\ Q_{l-1}^m &= Z_{l-1}W_Q^m,\ K_{l-1}^m = Z_{l-1}W_K^m,\ V_{l-1}^m = Z_{l-1}W_V^m, \end{aligned} \tag{2}$$

where M and d are the number of SA blocks and the dimension of the self-attention block, which is the same as the dimension of the weight matrices, respectively. The Multi-head Self-Attention (MSA) consists of the M number of SA blocks with the learnable parameters of weight matrices

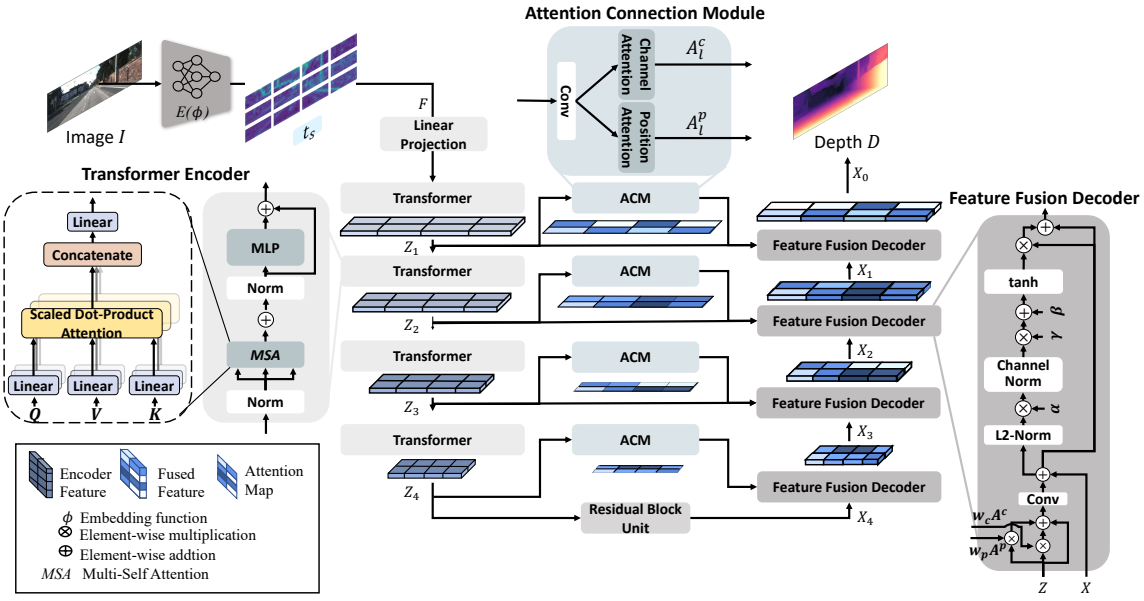


Figure 1: Overall Architecture. We design an encoder-decoder structure with a multi-level feature fusion module. The encoder is composed of a CNN and Transformers. The ACM learns the channel and position attentions. The FFD adaptively fuses the encoder features using the attention maps.

$W \in \mathbb{R}^{Md \times C}$ as follows:

$$\begin{aligned} \text{MSA}_{l-1} &= Z_{l-1} + \text{concat}(\text{SA}_{l-1}^1;\ \text{SA}_{l-1}^2;\ \ldots;\ \text{SA}_{l-1}^M)W, \\ Z_l &= \text{MLP}(\text{LN}(\text{MSA}_{l-1})) + \text{MSA}_{l-1}. \end{aligned} \tag{3}$$

This Transformer layer is repeated L times with unique learnable parameters. The outputs of the Transformers $\{Z_1, ..., Z_L\}$ are utilized as the input of the following layers ACM and FFD.

3.2 Attention Connection Module (ACM)

We design a new skip connection method, ACM, which produces the attention of global context and a semantic presentation of the feature given the features $Z_l,\ l \in \{1, ..., L\}$. The skip connection is widely utilized for the dense prediction tasks (Ronneberger, Fischer, and Brox 2015) because it helps to keep the fine detail by directly transferring the spatial information to the decoder. However, it has been observed that in the naïve skip connection method, concatenating each feature is too simple to preserve local detail, such as object boundaries (Zhou et al. 2018). To tackle the problem, we introduce an ACM that produces attention weight from the spatial domain and the channel domain inspired by (Fu et al. 2019). It consists of position attention, channel attention modules, and a fusion block that gathers important information from two attentions. The position attention module produces a position attention map $A_l^p \in \mathbb{R}^{C \times N}$ as follows:

$$A_l^p = \text{softmax}(Q_l^p (K_l^p)^{\mathrm{T}}) V_l^p, \tag{4}$$

where Q_l^p, K_l^p and V_l^p are the query, key, and value matrices computed by passing Z_l through a single convolutional layer. The channel attention module directly calculate the channel attention map $A_l^c \in \mathbb{R}^{C \times N}$ by computing the gram matrix of Z_l as follows:

$$A_l^c = \text{softmax}(Z_l Z_l^{\mathrm{T}}). \tag{5}$$

The position attention map A_l^p and channel attention map A_l^c enhance the feature representation by capturing long-range context and exploiting the inter-dependencies between each channel map, respectively. These two attention maps are utilized in the following section, which highlights the importance of the features.

3.3 Feature Fusion Decoder (FFD)

The FFD gets the encoder features Z_l, the attention maps $A_l^p,\ A_l^c$, and the output feature X_L of the last Transformer layer passed through a Residual convolutional layer. The decoder fuses the feature $X_{L-l+1},\ l \in \{1, ..., L\}$ through a single Convolutional layer (Conv) and Channel Normalization (CN) with learnable parameters α, β and γ as follows:

$$\begin{aligned} X_{L-l} &= \hat{X}_{L-l}[1 + \tanh(\gamma(\text{CN}(\alpha||\hat{X}_{L-l}||_2 + \beta)], \\ \hat{X}_{L-l} &= \text{Conv}(w_p A_l^p Z_l + w_c A_l^c Z_l + Z_l) + X_{L-l+1}, \end{aligned} \tag{6}$$

where w_p and w_c are the learnable parameters that determine the importance of the position and channel attentions (Zhang et al. 2019). The parameter α works so that each channel can learn about each other individually, and γ and β control the activation channel-wisely following the work in (Yang et al. 2020). Through this process, the FFD is able to produce a depth map from the fused features that preserve local detailed semantic representation while maintaining the global context of features.

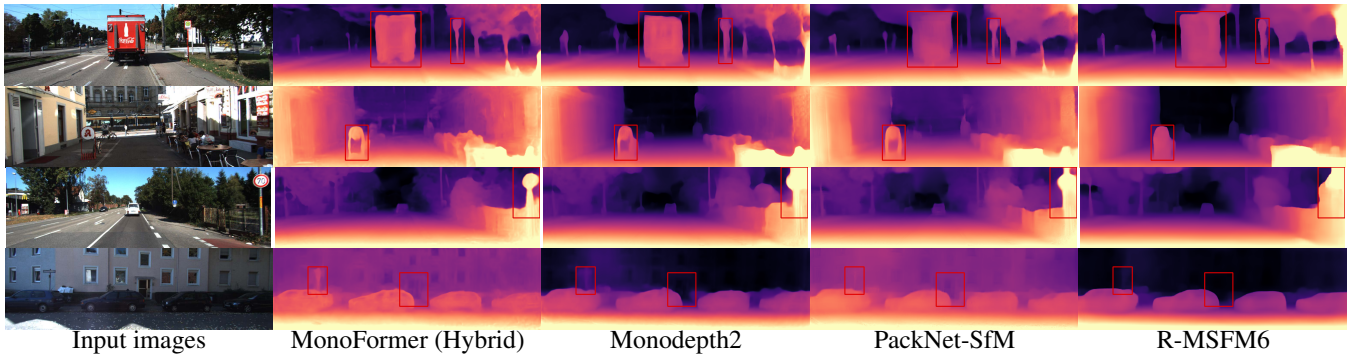


Figure 2: Qualitative comparison to state-of-the-arts. We use KITTI for training and testing.

Model	Backbone	Lower is better ↓				Higher is better ↑		
		Abs Rel	Sq Rel	RMSE	RMSElog	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Monodepth	CNN	0.148	1.344	5.972	0.216	0.816	0.941	0.976
Monodepth2	CNN	0.115	0.903	4.863	0.193	0.877	0.959	0.981
PackNet-Sfm	CNN	0.111	**0.785**	4.601	0.189	0.878	0.960	**0.982**
SGDepth	CNN	0.117	0.907	4.844	0.196	0.875	0.958	0.980
R-MSFM3	CNN	0.114	0.815	4.712	0.193	0.876	0.959	0.981
R-MSFM6	CNN	0.112	0.806	4.704	0.191	0.878	0.960	0.981
Ours-ViT	Transformer	0.118	0.942	4.840	0.193	0.873	0.956	0.981
Ours-Hybird	CNN-Transformer	**0.104**	0.846	**4.580**	**0.183**	**0.891**	**0.962**	**0.982**

Table 1: Quantitative comparison to state-of-the-arts. We evaluate models trained on KITTI (K) with an input image size of 640×192. We only use monocular images (M) for supervision. Bold is the best performance.

4 Experiments

4.1 Comparison on KITTI Datasets

We compare our method with state-of-the-art methods, SGCDepth (Xiong et al. 2021), GeoNet (Yin and Shi 2018), Struct2depth (Casser et al. 2019), Monodepth2 (Godard et al. 2019), PackNet-SfM (Guizilini et al. 2020a), SGDepth (Klingner et al. 2020), R-MSFM (Zhou et al. 2021) in Tab. 1. We use the KITTI Eigen split (Geiger et al. 2013; Eigen and Fergus 2015) consisting of 39,810 training, and 4,424 validation and 697 test data. We additionally sample data about 5% of the total data with infinite-depth problems that mostly occur in dynamic scenes, following the work (Guizilini et al. 2020b). We use typical error and accuracy metrics for depth, absolute relative (Abs Rel), square relative (Sq Rel), root-mean-square-error (RMSE), its log (RMSElog), and the ratio of inliers following the work (Guizilini et al. 2020a). The quantitative results show that the proposed method outperforms other models. The qualitative results in Fig. 2 show that our method precisely preserves object boundaries. This demonstrates that the encoder captures both global context and informative local features and transfers them to the decoder for the pixel-wise prediction.

4.2 Analysis of Texture-/Shape-Bias on CNN and Transformer

Generally, the texture represents a spatial color or pattern of pixel intensity in an image (Armi and Fekri-Ershad 2019). To examine the influence of textures on the inference process in detail, we apply three different texture modification strategies including texture-smoothing (Watercolor), texture removal (Pencil-sketch), and texture-transfer (Style-transfer). Extensive details of image generation to facilitate replication are provided in the Appendix. The generated images and the correspondence results are shown in Fig. 3. The first two images are watercolors, the middle two images and the last two images are pencil-sketch and style-transferred images, respectively. We also conduct the quantitative evaluations in Tab. 2 using all of the KITTI test data (697 images).

In this experiment, we compare the performance of CNN-based models (Monodepth2, PackNet-SfM, R-MSFM6), a Transformer-based model (Ours-ViT), and a hybrid (Ours-Hybrid) model. We note that Ours-Hybrid is equivalent to MonoFormer and Ours-ViT employs only the ViT (Dosovitskiy et al. 2020) encoder structure without any CNN. Both qualitative and quantitative results of the watercolor data show that both the CNN-based and Transformer-based models produce plausible depth maps. However, the CNN-based model tends to lose more details of the object boundaries and has higher errors than the Transformer-based models. The experiments with the pencil-sketch data and the style-transfer data show that the Transformer-based models distinguish objects (*e.g.* pedestrians and cars) and stuff (*e.g.* walls and roads) better than the CNN-based models. Specifically, the CNN-based models produce unrecognizable depth maps on style-transfer data due to the loss of original texture information. These experiments demonstrate our two observations. One is that CNNs have a strong texture bias while Transformers have a strong shape bias. The other is that models with shape bias representation provide better

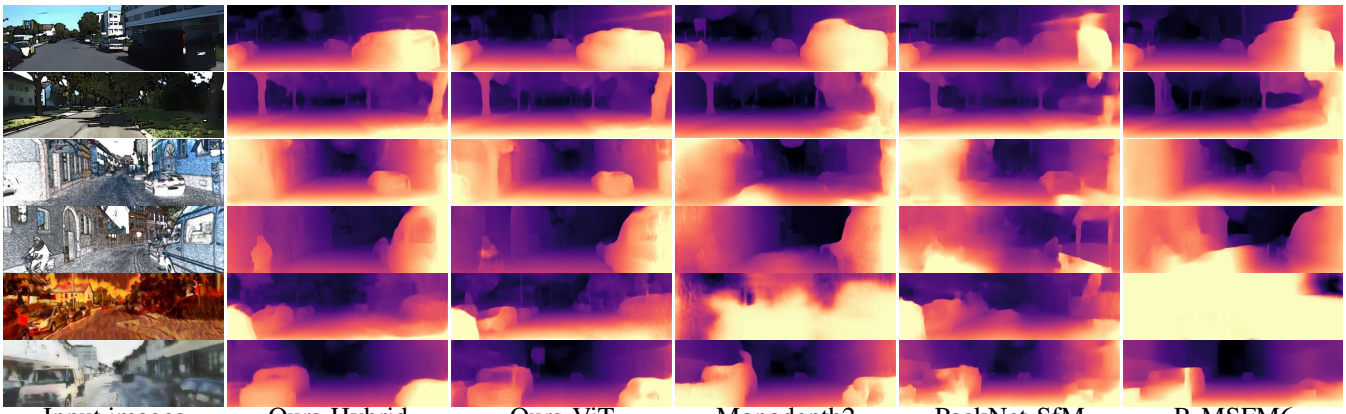


Figure 3: Depth map results on texture-shifted datasets. We test our hybrid/ViT models and the competitive models trained on KITTI using watercolor, pencil-sketch, and style-transfer images (Top to Bottom). Note that the Ours-Hybrid is equivalent to MonoFormer.

D*	Model	Abs Rel ↓	Sq Rel ↓	RMSE ↓	RMSElog ↓	$\delta < 1.25 \uparrow$	$\delta < 1.25^2 \uparrow$	$\delta < 1.25^3 \uparrow$
Water color	Monodepth2	0.170	1.345	6.175	0.263	0.750	0.909	0.960
	PackNet-SfM	0.174	1.364	6.334	0.264	0.742	0.906	0.961
	R-MSFM6	0.194	1.613	7.173	0.302	0.696	0.876	0.943
	Ours-ViT	0.152	1.196	5.668	0.232	0.799	0.932	0.973
	Ours-Hybrid	**0.140**	**1.053**	**5.665**	**0.222**	**0.815**	**0.936**	**0.975**
Pencil sketch	Monodepth2	0.196	1.522	6.232	0.276	0.691	0.898	0.962
	PackNet-SfM	0.204	1.569	6.568	0.290	0.670	0.888	0.957
	R-MSFM6	0.217	1.698	6.719	0.301	0.647	0.872	0.951
	Ours-ViT	0.174	1.311	5.770	0.248	0.756	0.920	0.967
	Ours-Hybrid	**0.151**	**1.084**	**5.615**	**0.227**	**0.786**	**0.934**	**0.976**
Style transfer	Monodepth2	0.435	6.107	10.891	0.509	0.379	0.660	0.821
	PackNet-SfM	0.379	4.462	9.834	0.470	0.418	0.708	0.855
	R-MSFM6	0.394	4.667	10.214	0.490	0.399	0.680	0.837
	Ours-ViT	0.378	4.854	9.869	0.449	**0.447**	0.730	0.869
	Ours-Hybrid	**0.351**	**3.847**	**9.402**	**0.438**	0.446	**0.737**	**0.875**

Table 2: Quantitative comparison on texture-shifted datasets. D* is datasets.

generalization performance for monocular depth estimation compared to models with texture bias. Of particular note, MonoFormer (Ours-Hybrid) more precisely preserves object boundaries than Transformer-based model (Ours-ViT). Ours-ViT also generally produces reliable depth thanks to the shape bias of Transformers, but fails to recover details such as a pedestrians. We believe that the proposed multi-level feature fusion module captures both shape bias and the spatial locality bias.

4.3 Generalization Performance of CNN-Based, Transformer-Based, and Hybrid Models

We compare the generalization performance of all the competitive models and ours trained on the KITTI datasets (Geiger et al. 2013; Eigen and Fergus 2015). We test the models using public depth datasets consisting of indoor scenes (SUN3D (Xiao, Owens, and Torralba 2013), RGBD (Sturm et al. 2012)), synthetic scenes from graphics tools (Scenes11 (Ummenhofer et al. 2017)), outdoor building-focused scenes (MVS (Ummenhofer et al. 2017)), and night driving scenes (Oxford Robotcar (Maddern et al. 2016)). We also use ETH3D (Schops et al. 2017) containing both indoor and outdoor scenes. The results in Fig. 4 show that the CNN-based models fail to estimate depth even though the scenes from the training and test sets share the stuff (*e.g.* road and sky) and things (*e.g.* cars), while the Transformer-based model keeps the details of object and scene. We observe that the texture shifts caused by illumination changes confuse the CNN-based model to estimate accurate depth.

The test results on the other scene environments in Fig. 5 also show aspects similar to the results in Fig. 4. The transformers-based models recover scene depth even in the complex scenes containing things and stuff which never been seen during training. However, the CNN-based models estimate unreliable depth maps, which keep the infinity depth mostly seen in KITTI datasets and loss the depth boundaries of objects. Of particular note, Ours-Hybrid produces more accurate depth maps which preserve the fine structures compared with Ours-ViT. The quantitative evaluations in Tab. 4 show that ours outperforms all competitive

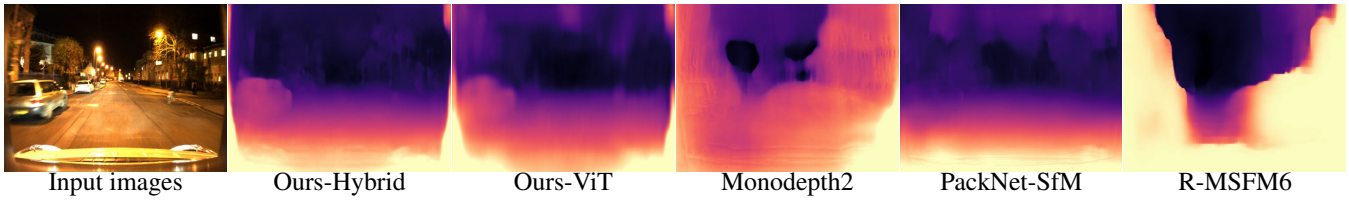


Figure 4: Depth Results on Oxford Robotcar Night time Dataset.

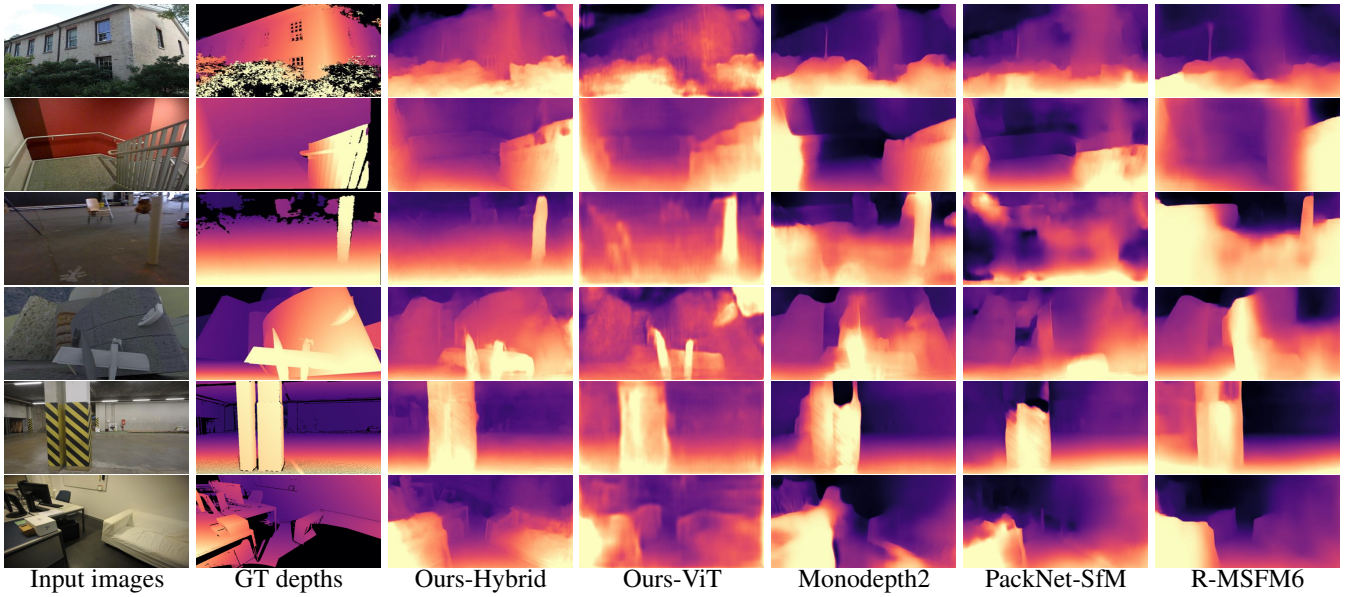


Figure 5: Comparison of depth map results on various dataset. We test our model and the competitive models trained on KITTI using MVS, SUN3D, RGBD, Scenes11 and ETH3D (Top to Bottom).

methods for all datasets and all measurements. MonoFormer achieves performance improvement of up to more than 30% over other CNN-based state-of-the-art models and 7% over a Transformer-based model (Ours-ViT) on average in Abs Rel. We believe that our network efficiently combines the local region information from the proposed module while keeping the shape bias representation from Transformers.

4.4 Analysis of Feature Representation on CNN and Transformers

Previous works (Geirhos et al. 2019; Esser, Rombach, and Ommer 2020; Islam et al. 2021) propose analysis methods for the representation and the mechanisms of CNNs. They contain the method to quantify the amount of shape information and texture information in the feature representation (Islam et al. 2021). Following the method, we freeze the encoder E of the depth network and input the image I to obtain the encoder's feature z ($z = E(I)$). The mutual relationship between z_a and z_b is obtained using image pairs (I_a, I_b) with specific semantic concepts (i.e., texture or shape features) can be used to quantify the types of features that the network has learned. We measure correlation relationships through a simple correlation coefficient ρ in (7).

$$\rho = \frac{\mathrm{Cov}(z_a, z_b)}{\sqrt{\mathrm{Var}(z_a)\mathrm{Var}(z_b)}}. \tag{7}$$

Model	Shape	Texture
Monodepht2	273	411
PackNet-Sfm	75	144
R-MSFM6	145	303
MonoFormer-ViT	697	228
MonoFormer-Hybrid	334	275

Table 3: Estimation result of shape/texture dimensionality. All models are trained on KITTI datasets.

We estimate shape/texture dimensionality using 697 image pairs (*e.g.*, KITTI eigen test images) in the texture shifted dataset and the original KITTI dataset. We calculate ρ for each image pair and then sum it up. Tab. 3 shows that the features of the Transformer-based model involve more shape information than the CNN-based model.

4.5 Ablation study

Comparison to various backbones. We evaluate the performance of models with different backbones in Tab. 5. We compare ours to models whose encoder was built with either CNNs (ResNet50, ResNet101) or Transformers (ViT-B, Vit-L). We also evaluate another CNN-Transformer hybrid model, TransDepth (Yang et al. 2021). The results demonstrate that our model achieves the best performance among them. The numbers show that the model with Transform-

D*	Model	Abs Rel ↓	Sq Rel ↓	RMSE ↓	$\delta < 1.25 \uparrow$	$\delta < 1.25^2 \uparrow$	$\delta < 1.25^3 \uparrow$
MVS	Monodepth2	0.471	0.407	0.503	0.408	0.661	0.806
	PackNet-SfM	0.449	0.295	0.429	0.397	0.670	0.837
	R-MSFM6	0.550	0.603	0.583	0.352	0.591	0.756
	Ours-ViT	0.260	0.102	0.257	0.611	0.877	0.962
	Ours-Hybrid	**0.240**	**0.086**	**0.242**	**0.633**	**0.881**	**0.972**
RGBD	Monodepth2	0.610	0.508	0.488	0.292	0.520	0.681
	PackNet-SfM	0.593	0.416	0.460	0.318	0.562	0.731
	R-MSFM6	0.695	0.553	0.490	0.261	0.471	0.627
	Ours-ViT	0.383	0.185	0.284	0.487	0.701	0.846
	Ours-Hybrid	**0.363**	**0.137**	**0.282**	**0.486**	**0.744**	**0.867**
Scenes11	Monodepth2	1.647	0.763	0.356	0.312	0.529	0.671
	PackNet-SfM	2.065	0.837	0.330	0.310	0.530	0.674
	R-MSFM6	1.727	0.726	0.361	0.280	0.494	0.636
	Ours-ViT	1.671	0.657	0.268	0.355	0.575	0.713
	Ours-Hybrid	**1.511**	**0.404**	**0.255**	**0.388**	**0.615**	**0.755**
SUN3D	Monodepth2	0.554	0.535	0.576	0.324	0.556	0.718
	PackNet-SfM	0.466	0.336	0.471	0.350	0.612	0.792
	R-MSFM6	0.523	0.406	0.506	0.310	0.544	0.721
	Ours-ViT	0.289	0.163	0.298	0.554	0.810	0.910
	Ours-Hybrid	**0.245**	**0.088**	**0.255**	**0.582**	**0.869**	**0.964**
ETH3D	Monodepth2	1.007	0.780	0.396	0.318	0.536	0.687
	PackNet-SfM	0.802	0.401	0.268	0.378	0.639	0.809
	R-MSFM6	0.943	0.632	0.366	0.330	0.541	0.686
	Ours-ViT	0.701	0.312	0.217	0.473	0.760	0.890
	Ours-Hybrid	**0.668**	**0.293**	**0.189**	**0.531**	**0.817**	**0.926**

Table 4: Comparison results. Evaluation of KITTI-trained model on diverse public datasets. D* is datasets.

Backbone	Abs Rel ↓	RMSE ↓	$\delta < 1.25 \uparrow$	$\delta < 1.25^2 \uparrow$
ViT-B/16	0.118	4.840	0.873	0.956
ViT-L/16	0.116	4.832	0.875	0.957
ResNet50	0.123	4.690	0.884	**0.962**
ResNet101	0.113	**4.565**	0.875	**0.962**
TransDepth	0.121	4.809	0.865	0.957
Ours-Hybrid	**0.104**	4.580	**0.891**	**0.962**

Table 5: Ablation study on backbone network. We use only Transformers (ViT), CNNs (ResNet), and hybrid models (TransDepth (Yang et al. 2021) and ours). ViT-B and ViT-L are the base and large ViT (Dosovitskiy et al. 2020), respectively. TransDepth and ours use the combination of ResNet50 and ViT-B/16.

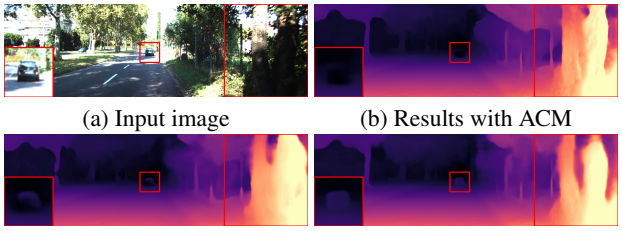
Datasets	Abs Rel	RMSE	$\delta < 1.25$	$\delta < 1.25^2$
KITTI	14.1% ↓	4.77% ↓	3.00% ↑	0.52% ↑
MVS	19.4% ↓	19.2% ↓	13.1% ↑	5.7% ↑
RGBD	15.6% ↓	16.4% ↓	13.3% ↑	7.7% ↑
Scenes11	8.7% ↓	11.3% ↓	10.3% ↑	4.9% ↑
SUN3D	31.5% ↓	30.7% ↓	36.9% ↑	20.6% ↑
ETH3D	6.7% ↓	18.3% ↓	9.9% ↑	8.4% ↑

Table 6: Comparison to another hybrid model. The error (Abs Rel, RMSE) reduction and accuracy ($\delta < 1.25$, $\delta < 1.25^2$) improvement percentage from TransDepth (Yang et al. 2021) to our Monoformer.

ers (ViT) performs worse than CNN (ResNet). This is because the insufficient number of datasets are used to tackle the lack of inductive bias that Transformers typically struggle with. The hybrid network (TransDepth) shows better performance than the pure Transformer-based network, but it still underperforms the pure CNN-based network. Meanwhile, our model outperforms CNNs, as well as the other backbone networks. We believe that this is because the proposed method effectively compensates for the lack of inductive bias in Transformers. We note that we use TransDepth whose model is provided by the author (Yang et al. 2021) and train the model in a self-supervised manner.

Comparison to the conventional hybrid models We compare our model with existing CNN-Transformer hybrid models, TransDepth (Yang et al. 2021). The original TransDepth model is trained with a large number of various datasets in a supervised manner. For a fair comparison, we train the author-provided TransDepth with KITTI eigen split in a self-supervised manner. We conduct the quantitative comparison using the five out-of-distribution datasets as well as the KITTI datasets. The results in Tab. 6 show the performance improvement ratio from TransDepth to MonoFormer. The experiments show that the proposed method achieves performance improvement around 15% on average in Abs Rel over TransDepth. These results show that MonoFormer outperforms all the conventional hybrid models.

Effectiveness of the proposed modules. We conduct an ablation study to demonstrate the effectiveness of the proposed modules, ACM and FFD in Tab. 7. The baseline is DPT (Ranftl, Bochkovskiy, and Koltun 2021). The models with only the ACM module or FFD module marginally improve



(a) Input image (b) Results with ACM

(c) Results with FFD (d) Results with ACM and FFD

Figure 6: Visualization of results with/without ACM and FFD.

	Abs Rel ↓	Sq Rel ↓	RMSE↓	RMSE$_{log}$ ↓	$\delta < 1.25$ ↑
baseline	0.113	0.899	4.783	0.189	0.882
+ACM	0.113	0.879	4.820	0.189	0.879
+FFD	0.112	0.860	4.803	0.186	0.879
+Both	**0.104**	**0.846**	**4.580**	**0.183**	**0.891**

Table 7: Ablation study on ACM and FFD. Both is with ACM and FFD.

the depth estimation performance, due to the absence of proper attention map fusions. On the other hand, our MonoFormer with both ACM and FFD significantly improves the performance. The results show the proposed model achieves the best performance in all measurements. The qualitative comparison in Fig. 6 shows that the model with both ACM and FFD keeps clearer object boundaries, even a small car in far depth.

Visualization of attention maps. We visualize the attention maps from the lower to higher layers of Transformers. As shown in Fig. 7, the encoder in the shallow layer extracts local region features. The deeper the layer, the more global shape contexts are extracted. Another observation is that ACM captures more detailed attention at different depths of the encoder features. FFD enhances the encoder features by fusing them with the attention map from ACM. The fused feature captures features from coarse to fine details. These experiments show that our model is capable of accurate pixel-wise prediction as it secures adequate local details.

5 Conclusion

In this paper, we provide three important observations for the self-supervised monocular depth estimation task: 1) *CNN-based models rely heavily on textures, while Transformer-based models rely on shapes for a monocular depth estimation task.* 2) *Texture-based representations leads to poor generalization performance with texture-shift such as scene changes, illumination changes, and style changes.* 3) *Shape-based representations are more helpful for a generalized monocular depth model than texture-based representations.* Based on these observations, we propose a CNN-Transformer hybrid network, called MonoFormer, which incorporates both shape bias and spatial locality bias. The proposed model achieves the best performance among various competitive methods on diverse unseen datasets as well as KITTI datasets, by a high margin. The extensive experiments demonstrate that our MonoFormer has superior generalization ability. We believe that the performance improvement comes from the design of strong shape-biased models, and this observation can be a useful insight to better understanding of monocular depth estimation.

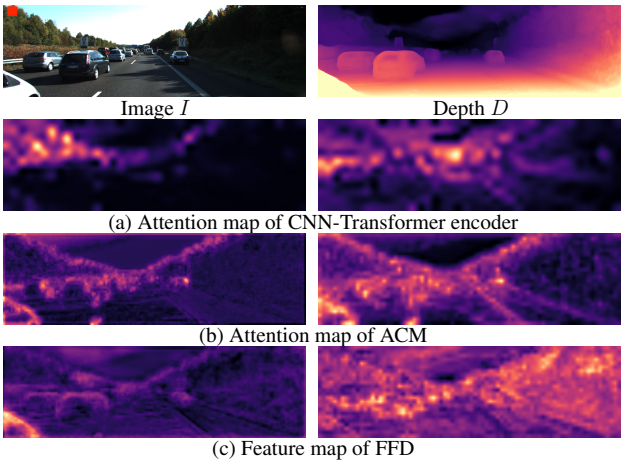


Image I Depth D

(a) Attention map of CNN-Transformer encoder

(b) Attention map of ACM

(c) Feature map of FFD

Figure 7: Visualization of attention map and feature map. We visualize the self-attention map of the patch on the upper left corner of the image I. The left column from the second row is the attention map from shallow layers, whereas the right is the map from deep layers.

Acknowledgments

This work was supported by Electronics and Telecommunications Research Institute (ETRI) grant funded by the Korea government [22ZS1200, Fundamental Technology Research for Human-Centric Autonomous Intelligent Systems], the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2020R1C1C1013210), and the Industry-Academia Collaboration R&D Program of the Ministry of SMEs and Startups (S3250483).

References

Armi, L.; and Fekri-Ershad, S. 2019. Texture image analysis and texture classification methods-A review. *arXiv preprint arXiv:1904.06554*.

Ballester, P.; and Araujo, R. M. 2016. On the performance of GoogLeNet and AlexNet applied to sketches. In *AAAI*.

Brendel, W.; and Bethge, M. 2019. Approximating cnns with bag-of-local-features models works surprisingly well on imagenet. In *ICLR*.

Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; and Zagoruyko, S. 2020. End-to-end object detection with transformers. In *ECCV*.

Casser, V.; Pirk, S.; Mahjourian, R.; and Angelova, A. 2019. Depth prediction without the sensors: Leveraging structure for unsupervised learning from monocular videos. In *AAAI*.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. 2020. An image is worth

16x16 words: Transformers for image recognition at scale. In *ICLR*.

Eigen, D.; and Fergus, R. 2015. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *ICCV*.

Esser, P.; Rombach, R.; and Ommer, B. 2020. A disentangling invertible interpretation network for explaining latent representations. In *CVPR*.

Fu, J.; Liu, J.; Tian, H.; Li, Y.; Bao, Y.; Fang, Z.; and Lu, H. 2019. Dual attention network for scene segmentation. In *CVPR*.

Fukushima, K. 1988. Neocognitron: A hierarchical neural network capable of visual pattern recognition. *Neural networks*.

Gatys, L. A.; Ecker, A. S.; and Bethge, M. 2017. Texture and art with deep neural networks. *Current opinion in neurobiology*.

Geiger, A.; Lenz, P.; Stiller, C.; and Urtasun, R. 2013. Vision meets Robotics: The KITTI Dataset. *IJRR*.

Geirhos, R.; Rubisch, P.; Michaelis, C.; Bethge, M.; Wichmann, F. A.; and Brendel, W. 2019. ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. In *ICLR*.

Godard, C.; Mac Aodha, O.; Firman, M.; and Brostow, G. J. 2019. Digging into self-supervised monocular depth estimation. In *ICCV*.

Guizilini, V.; Ambrus, R.; Pillai, S.; Raventos, A.; and Gaidon, A. 2020a. 3d packing for self-supervised monocular depth estimation. In *CVPR*.

Guizilini, V.; Ambruș, R.; Chen, D.; Zakharov, S.; and Gaidon, A. 2022. Multi-Frame Self-Supervised Depth with Transformers. In *CVPR*.

Guizilini, V.; Hou, R.; Li, J.; Ambrus, R.; and Gaidon, A. 2020b. Semantically-guided representation learning for self-supervised monocular depth. In *ICLR*.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *CVPR*.

Hendrycks, D.; and Gimpel, K. 2016. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*.

Hubel, D. H.; and Wiesel, T. N. 1959. Receptive fields of single neurones in the cat's striate cortex. *The Journal of physiology*.

Islam, M. A.; Kowal, M.; Esser, P.; Jia, S.; Ommer, B.; Derpanis, K. G.; and Bruce, N. 2021. Shape or texture: Understanding discriminative features in cnns. In *ICLR*.

Klingner, M.; Termöhlen, J.-A.; Mikolajczyk, J.; and Fingscheidt, T. 2020. Self-supervised monocular depth estimation: Solving the dynamic object problem by semantic guidance. In *ECCV*.

Kriegeskorte, N. 2015. Deep neural networks: a new framework for modeling biological vision and brain information processing. *Annual review of vision science*.

Landau, B.; Smith, L. B.; and Jones, S. S. 1988. The importance of shape in early lexical learning. *Cognitive development*.

Lyu, X.; Liu, L.; Wang, M.; Kong, X.; Liu, L.; Liu, Y.; Chen, X.; and Yuan, Y. 2021. HR-depth: high resolution self-supervised monocular depth estimation. In *AAAI*.

Maddern, W.; Pascoe, G.; Linegar, C.; and Newman, P. 2016. 1 year, 1000 km: The Oxford RobotCar dataset. *IJRR*.

Morrison, K.; Gilby, B.; Lipchak, C.; Mattioli, A.; and Kovashka, A. 2021. Exploring Corruption Robustness: Inductive Biases in Vision Transformers and MLP-Mixers. *arXiv preprint arXiv:2106.13122*.

Park, N.; and Kim, S. 2022. How Do Vision Transformers Work? In *ICLR*.

Ranftl, R.; Bochkovskiy, A.; and Koltun, V. 2021. Vision transformers for dense prediction. In *CVPR*.

Ronneberger, O.; Fischer, P.; and Brox, T. 2015. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*.

Schops, T.; Schonberger, J. L.; Galliani, S.; Sattler, T.; Schindler, K.; Pollefeys, M.; and Geiger, A. 2017. A multi-view stereo benchmark with high-resolution images and multi-camera videos. In *CVPR*.

Sturm, J.; Engelhard, N.; Endres, F.; Burgard, W.; and Cremers, D. 2012. A benchmark for the evaluation of RGB-D SLAM systems. In *IROS*.

Touvron, H.; Cord, M.; Douze, M.; Massa, F.; Sablayrolles, A.; and Jégou, H. 2021. Training data-efficient image transformers & distillation through attention. In *ICML*.

Tuli, S.; Dasgupta, I.; Grant, E.; and Griffiths, T. L. 2021. Are Convolutional Neural Networks or Transformers more like human vision? In *CogSci*.

Ummenhofer, B.; Zhou, H.; Uhrig, J.; Mayer, N.; Ilg, E.; Dosovitskiy, A.; and Brox, T. 2017. Demon: Depth and motion network for learning monocular stereo. In *CVPR*.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. In *NIPS*.

Xiao, J.; Owens, A.; and Torralba, A. 2013. Sun3d: A database of big spaces reconstructed using sfm and object labels. In *ICCV*.

Xiong, M.; Zhang, Z.; Zhong, W.; Ji, J.; Liu, J.; and Xiong, H. 2021. Self-supervised monocular depth and visual odometry learning with scale-consistent geometric constraints. In *IJCAI*.

Yang, G.; Tang, H.; Ding, M.; Sebe, N.; and Ricci, E. 2021. Transformer-based attention networks for continuous pixel-wise prediction. In *CVPR*.

Yang, Z.; Zhu, L.; Wu, Y.; and Yang, Y. 2020. Gated channel transformation for visual recognition. In *CVPR*.

Yin, Z.; and Shi, J. 2018. Geonet: Unsupervised learning of dense depth, optical flow and camera pose. In *CVPR*.

Zhang, C.; Zhang, M.; Zhang, S.; Jin, D.; Zhou, Q.; Cai, Z.; Zhao, H.; Yi, S.; Liu, X.; and Liu, Z. 2022. Delving deep into the generalization of vision transformers under distribution shifts. In *CVPR*.

Zhang, H.; Goodfellow, I.; Metaxas, D.; and Odena, A. 2019. Self-attention generative adversarial networks. In *ICML*.

Zheng, S.; Lu, J.; Zhao, H.; Zhu, X.; Luo, Z.; Wang, Y.; Fu, Y.; Feng, J.; Xiang, T.; Torr, P. H.; et al. 2021. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In *CVPR*.

Zhou, T.; Brown, M.; Snavely, N.; and Lowe, D. G. 2017. Unsupervised learning of depth and ego-motion from video. In *CVPR*.

Zhou, Z.; Fan, X.; Shi, P.; and Xin, Y. 2021. R-MSFM: Recurrent Multi-Scale Feature Modulation for Monocular Depth Estimating. In *CVPR*.

Zhou, Z.; Rahman Siddiquee, M. M.; Tajbakhsh, N.; and Liang, J. 2018. Unet++: A nested u-net architecture for medical image segmentation. In *Deep learning in medical image analysis and multimodal learning for clinical decision support*, 3–11. Springer.