2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

Self-Supervised Monocular Depth Estimation with Effective Feature Fusion and Self Distillation

Zhenfei Liu[1,2,3], Chengqun Song[1,2,3,*], Jun Cheng[1,2,3,*], Jiefu Luo[1,2,3], Xiaoyang Wang[1]
1 Guangdong Provincial Key Laboratory of Robotics and Intelligent
System, Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, China
2 University of Chinese Academy of Sciences, Beijing, China
3 The Chinese University of Hong Kong, Hong Kong, China
Emails: {zf.liu2, cq.song, jun.cheng, xy.wang, jf.luo}@siat.ac.cn

***Abstract*—Monocular depth estimation obtaining scene depth information from a single image is an important task in the field of computer vision. Constrained by the limitations of convolutional networks in conducting long-distance modeling and the underutilization of datasets, the generalization of existing models is not satisfactory. In this paper, we propose an adaptive backbone named Internal Fusion Transformer to improve generalization ability compared to convolutional backbone, like HRNet, and a Bilateral Attention module which pays more attention to low-level semantic features compared to previous fuse methods. Meanwhile, we introduce three data augmentation methods, namely cropping-resizing (cr), cropping-shuffling (cs), and mirroring (mi), for self distillation, as well as discuss their contributions to model performance improvement. Our model is trained on the KITTI dataset, and without fine-tuning, tested on NYUv2 and Make3D datasets to confirm the generalization. The experimental results illustrate the effectiveness of our design. Our model also demonstrates better performance compared to other models on the KITTI dataset.**

***Index Terms*—Monocular Depth Estimation, Self-Supervised Learning, Generalization, Self Distillation**

I. Introduction

Depth estimation is an important part of the scene perception task. It can usually be inferred from the point cloud captured by LiDAR [1] and the RGB image captured by the camera [2]. Although LiDAR can obtain more reliable depth estimation result [3], its expensive purchase cost and larger volume bring inconvenience to daily use. On the contrary, visual depth estimation relying on RGB images is more cost-effective, and it can be further categorized into monocular depth estimation [4]-[7] and binocular depth estimation [8], [9]. In general, binocular depth estimation can be accomplished through precise stereogeometric calculations [8], whereas monocular depth estimation requires deep learning based on extensive datasets, owing to the limited availability of sufficient texture and scale cues [4]. Furthermore, monocular systems offer the advantage of lower cost compared to binocular systems, thus encouraging researchers to explore related work extensively.

Monocular depth estimation is usually divided into supervised [4] and self-supervised [5]-[7]. Supervised monocular depth estimation is superior to self-supervised monocular depth estimation in terms of performance so far. However, given the challenge of acquiring dense supervised labels,

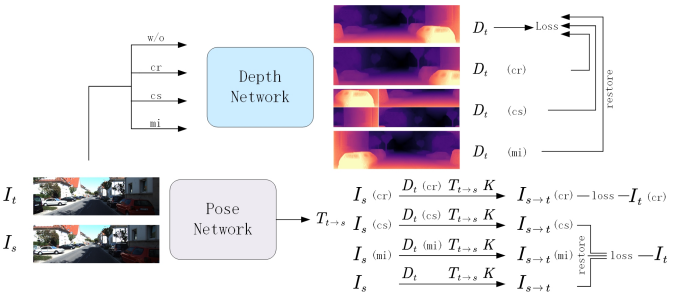

Fig. 1. **The overview of the training process**. We use images without data augmentation to infer the relative pose. Afterward, we apply three data augmentation methods to the target image I_t before depth estimation. Consequently, we obtain four depth maps: one from the original image and three from data augmentation. Subsequently, we restore the three data augmentation results to conduct self distillation with the original result and photometric error.

self-supervised monocular depth estimation can overcome this limitation, indicating significant development potential for the self-supervised method. With the advancement of self-supervised depth estimation, a common challenge arises when the object is in motion while the camera remains fixed, thereby violating the assumption of Structure from Motion (SfM) [20]. Proposed solutions to this issue involve masking objects that violate assumptions through the use of optical flow estimation [10] or semantic segmentation [11] processing. For instance, Monodepth2 [5] masks pixels with photometric changes exceeding a certain threshold after pose mapping, ensuring such pixels do not contribute to the loss calculation.

For an extended period, depth estimation has grappled with the issue of indistinct object boundaries [12]. Essentially, the depth estimation task can be viewed as a more intricate semantic segmentation task. Inspired by the utilization of multi-level semantic features in semantic segmentation tasks, along with the efficacy of an encoder-decoder structure, we employ the encoder to generate semantic features at multiple scales and the decoder to facilitate the fusion of semantic information. Commonly used backbones for the encoder-decoder structure include ResNet [16], HRNet [40], etc., which typically yield favorable results. However, their drawback is evident—they necessitate extensive pretraining

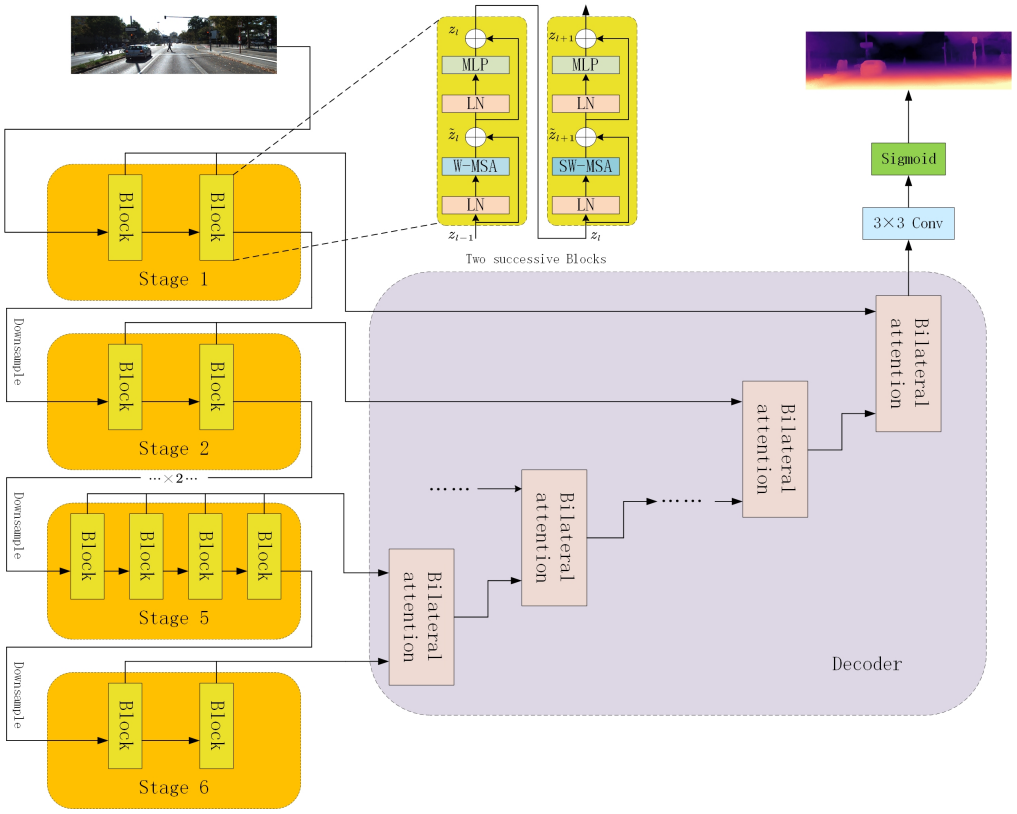

Fig. 2. **The overview of depth network with Internal Fusion Transformer**. Contrasting to the normal method, we utilize the result of every block in each stage.

epochs, indicating a substantial dependency on computing resources and limited generalizability. Hence, we introduce the Internal Fusion Transformer, which incorporates the window attention mechanism and integrates the outcomes of each intermediate block through the concatenation strategy at each stage. Simultaneously, another reason for the lower generalization is the underutilization of the available training data. Common practices involve the joint training of various datasets [13]. However, this method requires extensive training and faces challenges due to scale differences between datasets, necessitating detailed pre-processing. Therefore, we also introduce self distillation through data augmentation to enhance the performance of model.

Overall, the contributions of this paper include the following parts:

- We introduce the Internal Fusion Transformer, capable of capturing richer semantic information and achieving commendable performance within a limited training epoch.
- We propose using a Bilateral Attention mechanism in the decoder to improve semantic fusion for the final inference, particularly enhancing the integration of the low level semantic branch.
- We present self-distillation strategies through data augmentation to bolster the model's robustness, exploring the contributions of various data augmentation methods to model.

II. RELATED WORK

A. Supervised Monocular Depth Estimation

Eigen et al [14] firstly used CNN for global and local processing. Later, Laina et al. [15] proposed FCRN (Fully Convolutional Residual Networks), a network that combines residual networks [16] with upsampling blocks. They also introduced the inverse Huber function, a key element for high-resolution depth estimation. Subsequently, depth estimation has been redefined, shifting from a regression task to a classification task. Typically, Bhat et al. [17] divided the depth range into multiple intervals, with the size of each interval adapting dynamically based on the distribution of estimations. The ultimate depth value of the image was estimated as a linear combination of the interval centers with corresponding probabilities. Fu et al. [15] introduced the SID (spacing-increasing discretization) strategy to discretize depth values, treating the depth estimation network as an ordinal regression problem. They achieved higher accuracy by employing the ordinal regression strategy in training the network, which led to faster convergence. Simultaneously, in recent years, researchers have extended mathematical concepts and deep learning to this field. Yuan et al. [4] introduced NeW CRFs, transforming the globally fully connected CRFs (conditional random fields) with window attention to achieve reduced computation. Song et al. [12] introduced Lap-Depth, incorporating the Laplacian Pyramid into the decoder architecture. While these methods have achieved commendable

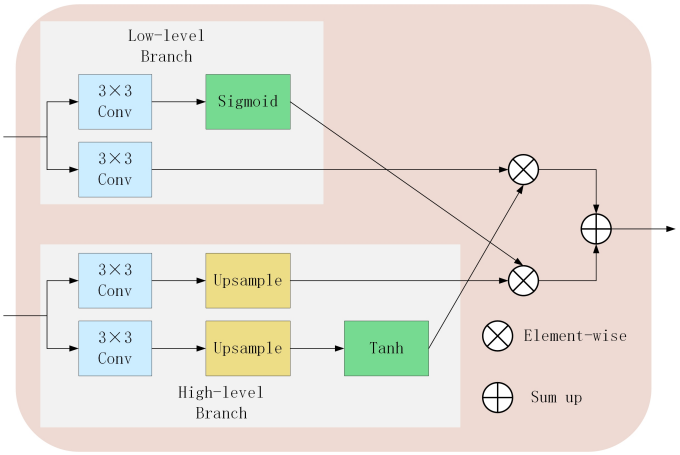

Fig. 3. **The detail of Bilateral Attention**.

Algorithm 1

1. Input: $I(C, H, W)$
2. Data Augmentation
Cropping-resizing
$I^{cr} = I[x^{cr} : x^{cr} + w^{cr}, y^{cr} : y^{cr} + h^{cr}]$
Croping-Shuffling
$I_1^{cs} = I[0 : w^{cs}, 0 : h^{cs}]$
$I_2^{cs} = I[w^{cs} : W, 0 : h^{cs}]$
$I_3^{cs} = I[0 : w^{cs}, h^{cs} : H]$
$I_4^{cs} = I[w^{cs} : W, h^{cs} : H]$
$I^{cs} = \begin{bmatrix} I_4^{cs} & I_3^{cs} \\ I_2^{cs} & I_1^{cs} \end{bmatrix}$
Mirroring
$I^{mi} = I[W : 0, 0 : H]$
3. Inference
$D = \text{depth}(I)$
$D^{cr} = \text{depth}(\langle I^{cr} \rangle)$
$D^{cs} = \text{depth}(I^{cs})$
$D^{mi} = \text{depth}(I^{mi})$
4. Loss Calculation
$L_{cr} = ||\langle D[x^{cr} : x^{cr} + w^{cr}, y^{cr} : y^{cr} + h^{cr}] \rangle - D^{cr}||_2$
$L_{cs} = || \begin{bmatrix} D[W : w^{cs}, H : h^{cs}] & D[0 : w^{cs}, H : h^{cs}] \\ D[W : w^{cs}, 0 : h^{cs}] & D[0 : w^{cs}, 0 : h^{cs}] \end{bmatrix} - D^{cs}||_2$
$L_{mi} = ||D[W : 0, 0 : H] - D^{mi}||_2$
$\mathcal{L}_{consistency} = \lambda_1 L_{cr} + \lambda_2 L_{cs} + \lambda_3 L_{mi}$

$<>$ represents the upsampling operation. The discussion of λ_1, λ_2 and λ_3 can be seen in TABLE V.

results, supervised depth estimation heavily relies on accurate dense labels.

B. Self-Supervised Monocular Depth Estimation

Self-supervised monocular depth estimation was initially pioneered by Zhou et al. [19]. They employed a pose network to obtain the relative pose between two images, used a depth network to generate the depth result of the target image, and combined this information to map the source image to the target image coordinate. This mapping was then used to compute photometric error with the target image. Self-supervised monocular depth estimation assumes a moving camera and a stationary object [20]. However, real-world scenarios often deviate from this, leading to the introduction of Monodepth2 [5]. This model categorizes pixels into occluded and non-occluded types, addressing the issue of excessive loss caused by occlusion. The commonly used photometric error loss function in self-supervised depth estimation faces challenges in low-texture regions. Even with a significant difference between depth estimates and true values, it may produce small loss values, leading to slow convergence or convergence towards local optima. Shu et al. [21] proposed a loss function that encourages convergence to the global optimum by incorporating regularization through the first and second derivatives. It performs well, even in low-texture regions. For indoor scenes, Ji et al. [22] introduced an algorithm called MonoIndoor. Its depth estimation decoder is partially divided into global scale estimation and relative scale estimation. Despite the effectiveness of unsupervised training, enhancing its resilience to physical attacks remains an unresolved challenge.

C. Generalization in Monocular Depth Estimation

To solve the generalization problem of the model in different scenarios, related works [23] can be broadly divided into two types, increasing the diversity of samples by synthesizing data [24], [25], and methods of data augmentation [26]. The form of data enhancement is mainly embodied in image transformation, which divides image representation into structure-invariant attributes and structure-specific attributes [27], [28]. The main purpose is to find most relevant factors. Among them, PlaneDepth [29] has achieved significant improvement in stereo point pairs. However, these techniques largely rely on obfuscated gradients which may not lead to true generalization.

III. METHOD

A. Foundational theory

The entire process of proposed self-supervised depth estimation method, shown in Fig. 1, includes two parts: relative pose estimation and depth estimation. We implement the photometric error between target image I_t and mapped source image $I_{s\to t}$. Mapped source image can be computed from source image I_s by the equation as follow:

$$I_{s\to t} = I_s < proj(D_t, T_{t\to s}, K) > \tag{1}$$

where D_t is the depth estimation result of target image, $T_{t\to s}$ is the relative pose estimation of I_t and I_s, $<>$ is sampling operator, and K is the intrinsic matrix. The photometric error is conducted by:

$$pe(I_a, I_b) = \frac{\alpha}{2}(1 - SSIM(I_a, I_b)) + (1-\alpha)\,\|I_a - I_b\| \tag{2}$$

$$\mathcal{L}_{ph} = \min_s pe(I_a, I_b) \tag{3}$$

α is a hyper-parameter, we use a and b to replace t and $s \to t$. And $SSIM(I_a, I_b)$ stands for structural similarity index which can be expressed as:

$$SSIM(I_a, I_b) = \frac{(2\mu_a\mu_b + C_1)(2\sigma_{ab} + C_2)}{(\mu_a^2 + \mu_b^2 + C_1)(\sigma_a^2 + \sigma_b^2 + C_2)} \tag{4}$$

μ_a and μ_b are mean values for I_a and I_b respectively. σ_a and σ_b are variance for I_a and I_b respectively. σ_{ab} is covariance

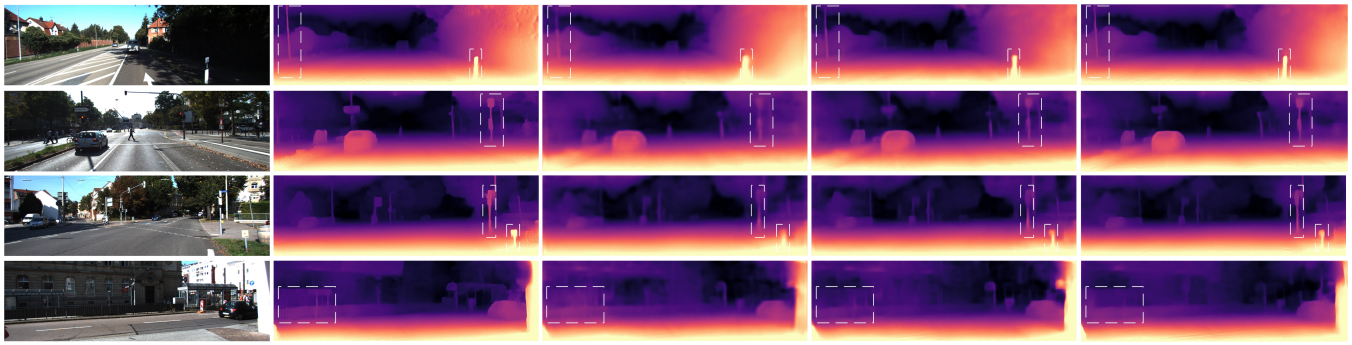

Fig. 4. **Qualitative results on the KITTI dataset**. The results of our model have clearer outline.

between I_a and I_b. C_1 and C_2 are aim to prevent singularity occurring.

Another smooth loss function to ensure the smoothness of the prediction results can be written as:

$$\mathcal{L}_{sm} = |\partial_x D_t^*| e^{-|\partial_x I_t|} + |\partial_y D_t^*| e^{-|\partial_y I_t|} \tag{5}$$

D_t^* is the result of D_t normalized, ∂_x and ∂_y are gradients in direction x and y.

B. Internal Fusion Transformer Backbone

Different from the traditional global attention Transformer, the Transformer structure adopted in this paper is based on window attention [30]. The complexity of global attention Transformer is:

$$\Omega(MSA) = 4hwC^2 + 2(hw)^2C \tag{6}$$

while window attention is:

$$\Omega(W-MSA) = 4hwC^2 + 2M^2hwC \tag{7}$$

where h and w are input size, C is number of channels, M is window size. Due to M is set smaller than h and w, window attention has lower complexity but fair performance. According to the experimental results of the relevant literature [30], we set M as 7. The structure is shown in Fig. 2, with totally 6 stages. Except for stage 5, each stage has 2 blocks. The details in every two successive blocks include:

$$\widetilde{z}_l = W-MSA(LN(z_{l-1})) + z_{l-1} \tag{8}$$

$$z_l = MLP(LN(\widetilde{z}_l)) + \widetilde{z}_l \tag{9}$$

$$\widetilde{z}_{l+1} = SW-MSA(LN(z_l)) + z_l \tag{10}$$

$$z_{l+1} = MLP(LN(\widetilde{z}_{l+1})) + \widetilde{z}_{l+1} \tag{11}$$

where $W-MSA(\bullet)$ is window attention, $SW-MSA(\bullet)$ is window attention with shift offset, $LN(\bullet)$ is linear layer, z_{l-1} is the output of last block, and $\widetilde{z}_l$ is an intermediate variable.

The Internal Fusion Transformer introduces additional connections among blocks to increase the number of channels for encoded features. This is motivated by the idea that more channels generally contain richer semantic information.

$$F_{encoded} = cat[z_0, \cdots, z_t] \tag{12}$$

where t is the number of blocks in each stage. Besides, there is a downsampling operation between stages, and thus we can obtain coded feature in 1, 1/2, 1/4, 1/8, 1/16 and 1/32 of original input size.

C. Bilateral Attention

To fuse encoded features, a Bilateral Attention mechanism is introduced for the decoder to integrate high- and low-level semantic information. Given high- and low-level semantic inputs, we use the convolution operation to generate the key and value of each branch, and sum up the result of two levels as the final fusion output. In the high-level semantic branch, we use tanh as the activation function, while in the low-semantic branch, the activation function is sigmoid. In comparison to the straightforward direct combination of two branches, Bilateral Attention enhances the effectiveness of integration between the two branches, overcoming the limitation of neglecting low-level semantic information. The detail of Bilateral Attention module can be seen in Fig. 3. In summary, Bilateral Attention forms a unified understanding between high- and low-level branches, and thus improves the effectiveness of the learning outcome.

D. Self-Distillation Method

Given the premise that the depth estimation result for the same region should remain consistent, even when the image undergoes cropping, resizing, shuffling, or mirror operations. Simultaneously, the depth estimation task is treated as an ordinal regression task [18], which means that the prediction result of a pixel is influenced by the information of the surrounding pixels. Wrong pixel arrangement, such as cropping-shuffling, will inevitably affect the prediction results. Therefore, we believe that a lower coefficient should be assigned to self-distillation loss caused by wrong pixel arrangement data augmentation, and a higher coefficient for correct pixel arrangement. In this paper, three different data augmentation methods are selected for verification in the

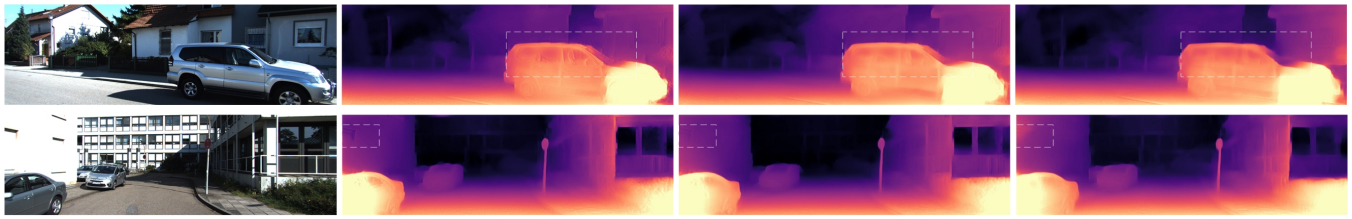

Fig. 5. **Qualitative results of different coefficient combinations**. (a) λ_1 and λ_3 are higher than λ_2. (b) They are equal. (c) λ_1 and λ_3 are lower than λ_2. The results of (a) are more detailed than others, mainly reflected in the car windows and the wall windows.

TABLE I
THE QUANTITATIVE RESULTS ON KITTI WITH RAW GT.

	Method	Train	W×H	AbsRel	SqRel	RMSE	RMSElog	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Pretraining backbone	Monodepth2[5]	M	640*192	0.115	0.903	4.863	0.193	0.877	0.959	0.981
	Mono-Uncertainty [36]	M	640*192	0.111	0.863	4.756	0.188	0.881	0.961	0.982
	Fang [37]	M	640*192	0.111	-	4.660	0.186	0.884	0.962	0.982
	HR-Depth [38]	M	640*192	0.109	0.792	4.632	0.185	0.884	0.962	0.983
	DIFFNet [7]	M	640*192	0.102	0.764	4.483	0.180	0.896	0.965	0.983
	Ours(HRNet18 backbone)	M	640*192	**0.096**	**0.617**	**4.260**	**0.172**	**0.898**	**0.968**	**0.985**
	HR-Depth [38]	M+S	640*192	0.107	0.785	4.612	0.185	0.887	0.962	0.982
	Monodepth2 [5]	M+S	640*192	0.106	0.818	4.750	0.196	0.874	0.957	0.979
	DIFFNet [7]	M+S	640*192	**0.101**	**0.749**	**4.445**	**0.179**	**0.898**	**0.965**	**0.983**
No pre-training	SfMlearner[19]	M	640*192	0.183	1.595	6.709	0.270	0.734	0.902	0.959
	Li [34]	M	416*128	0.130	0.950	5.138	0.209	0.843	0.948	0.978
	Packnet [35]	M	640*192	0.111	0.785	4.601	0.189	0.878	0.960	0.982
	Lite-Mono [6]	M	640*192	0.107	0.765	4.561	0.183	0.886	0.963	0.983
	BRNet [39]	M	640*192	0.105	0.698	4.462	0.179	**0.890**	0.965	0.984
	Ours(IFT)	M	640*192	**0.103**	**0.656**	**4.326**	**0.175**	0.888	**0.966**	**0.986**
	BRNet [39]	M+S	640*192	**0.099**	**0.685**	**4.453**	**0.183**	**0.885**	**0.962**	**0.983**

"M" means monocular setting and "M+S" means monocular plus stereo setting. "IFT" is the acronym of "Internal Fusion Transformer".

experiment, which are cropping-resizing, cropping-shuffling and mirroring operation. In comparison to cropping-shuffling, the other two methods do not introduce incorrect correlations between pixels in the image. Thus, they are divided into two categories. The specifics of self distillation in training are shown in Algorithm 1.

IV. EXPERIMENT

We implement the experiment at the RTX Titan V (12GB). In the experiment, we train on the dataset KITTI [31] split by Zhou et al. [19], and test on Eigen split [14]. In addition, without fine-tuning, we directly use our model to test Make3D [32] and NYUv2 [33]. During the training process, for the initial 15 epochs, we utilized a learning rate of 1e-4. Subsequently, in the 16^{th} epoch, the learning rate was adjusted to 1e-5. Notably, the model's performance on the test dataset exhibited a substantial improvement compared to the preceding epochs. The batch size of model with HRNet [40] backbone is 4, and 2 for Internal Fusion Transformer backbone.

A. Implement Details

In the training process, the error loss includes the photometric error $\mathcal{L}_{ph}$ between the mapped source image $I_{s\to t}$ and the target image I_t, the horizontal and vertical smooth error $\mathcal{L}_{sm}$, and the consistency error $\mathcal{L}_{consistency}$ between inference results of original image and Augmented image. The total error loss can be written as:

$$\mathcal{L}_{sm} = \mathcal{L}_{ph} + \mathcal{L}_{sm} + \mathcal{L}_{consistency} \tag{13}$$

Besides, the pose network is only used in training, and only the depth network is used in the test. We use two kinds of backbone in our experiments, one is Internal Fusion Transformer backbone which is proposed by ourselves, and the other one is HRNet [40] pretrained more than 3000 epochs. The size of the input is set as 640*192.

B. Results

The results on KITTI: For the KITTI [31] dataset, we capped the maximum predicted depth value at 80m. We conducted a performance comparison of our network against classical models such as Monodepth2 [5] and newer models like DIFFNet [7], etc.. Almost in all metrics, ours exhibits a better performance. Our model with Internal Fusion Transformer achieves the best performance among all monocular models without pretraining. Our model with HRNet18 even achieved 9% improvement in terms of SqRel compared to stereo pairs model BRNet [39] as an only monocular method. Both of Monodepth2 and HR-Depth use HRNet as the backbone, but our model with HRNet outperforms them, thus confirming the superiority of the Bilateral Attention applied in decoder. AbsRel, SqRel, RMSE, and RMSElog represent differences from the ground truth (GT), and lower

values indicate better performance. Accuracy-based metrics ($\delta < 1.25$, $\delta < 1.25^2$, $\delta < 1.25^3$) are higher for better performance. Quantitative details are provided in TABLE I, while visual results are presented in Fig. 4.

The results on Make3D and NYUv2: In order to verify the generalization ability of our model, we use the model trained on KITTI [31] dataset to test on Make3D [32] and NYUv2 [33] without any fine-tuning. The maximum

TABLE II
THE QUANTITATIVE RESULTS ON MAKE3D.

Method	AbsRel	SqRel	RMSE	RMSElog
Monodepth2 [5]	0.322	3.589	7.418	0.163
HR-Depth [38]	0.315	3.208	7.024	0.159
Lite-Mono [6]	0.305	3.060	6.981	0.158
Ours(IFT)	**0.288**	**2.698**	6.696	**0.146**
Ours(HRNet18)	0.289	2.802	**6.693**	0.150

estimated depth of the Make3D and NYUv2 is set to 70m and 10m respectively. Through comparison, we find that our monocular depth estimation model demonstrates satisfactory generalization ability, and Internal Fusion Transformer backbone is better than HRNet [40]. It also demonstrates that Internal Fusion Transformer has better generalization ability than HRNet. The quantitative results are shown in TABLE II and TABLE III respectively.

TABLE III
THE QUANTITATIVE RESULTS ON NYUV2

Method	AbsRel	SqRel	RMSE	RMSElog
Monodepth2 [5]	0.377	0.778	1.388	0.414
HR-Depth [38]	0.321	0.521	1.150	0.367
Ours(IFT)	**0.280**	**0.369**	**0.921**	**0.306**
Ours(HRNet18)	0.284	0.384	0.945	0.311

Ablation experiment: We have discussed the effect of proposed modules before. Therefore, the ablation experiment primarily concentrates on self distillation, examining whether to utilize them and assessing different coefficients for distinct data augmentation methods. In the first ablation experiment, we employ the Internal Fusion Transformer backbone with the same coefficient for each data augmentation. In the second experiment, we use HRNet as backbone [40]. We conduct experiments on the KITTI [31] dataset. Based on the results, our model with all data augmentations surpasses the performance of the model lacking any one of them or lacking all of them. The detailed results are presented in TABLE IV. Meanwhile, the outcome of assigning a higher coefficient to the category with correct adjacent relation and a lower coefficient to wrong adjacent relation is superior to assigning a lower coefficient to correct adjacent relation and a higher coefficient to wrong adjacent relation. We consider cropping-resizing and mirroring to belong to the same category since they do not disrupt the adjacent relation between pixels, unlike cropping-shuffling. Due to resource constraints, we only select three combinations of coefficients, representing higher for correct adjacent relation, higher for wrong adjacent relation, and equal for both. Following the mentioned assumption, we assign cropping-resizing and mirroring the same coefficient. The results are presented in TABLE V. Visual results for different coefficient combinations are provided in Fig. 5.

TABLE IV
ABLATION RESULTS ON SELF DISTILLATION.

Method	AbsRel	SqRel	RMSE	$\delta < 1.25$
Ours w/o cr	0.106	0.672	4.475	0.880
Ours w/o cs	0.107	0.677	4.487	0.878
Ours w/o mi	0.106	0.643	4.451	0.880
Ours w/o all	0.113	0.743	4.628	0.871
Ours	**0.103**	**0.656**	**4.326**	**0.888**

TABLE V
RESULTS OF DIFFERENT COEFFICIENT TO SELF DISTILLATION.

Method	λ_1	λ_2	λ_3	AbsRel	SqRel	RMSE
Ours w/o all	-	-	-	0.107	0.672	4.472
Ours	1.1	0.8	1.1	**0.096**	**0.617**	4.260
Ours	1.0	1.0	1.0	0.098	0.623	**4.221**
Ours	0.9	1.2	0.9	0.100	0.668	4.301

*We assume cropping-resizing and mirroring are belonging to same category, because they don't bring wrong adjacent relation to pixels in image. Therefore, we give λ_1 and λ_3 the same coefficient.

V. CONCLUSION

In this work, we not only focus on enhancing the accuracy of the model but also consider its generalization. We introduce the Internal Fusion Transformer backbone to achieve a performance close to HRNet18 without pretraining. Additionally, we propose Bilateral Attention, which emphasizes low-level semantic information in the fusion. Simultaneously, we employ self distillation through data augmentation to enhance the robustness and accuracy of the model. We also discuss the impact of different coefficients on each data augmentation. Introducing data augmentation for network's self-distillation training is no longer rare. However, this study fully considers the inherent characteristics of depth estimation task, where the prediction of a single pixel is associated with the surrounding pixels. Meanwhile, cropping-resizing lacks context at the edges, and thus, considering it the same as mirroring is not a perfect solution. We hope that our research can inspire readers and anticipate further development in this area.

ACKNOWLEDGMENT

This work is supported by Guangdong Major Project of Basic and Applied Basic Research (2023B0303000016), Shenzhen Technology Project (JCYJ20220818101211025) and CAS Key Technology Talent Program.

REFERENCES

[1] H. Tang, X. Niu, T. Zhang, et al., "LE-VINS: A Robust Solid-State-LiDAR-Enhanced Visual-Inertial Navigation System for Low-Speed Robots," IEEE Transactions on Instrumentation and Measurement, vol. 72, no. 8502113, pp. 1-13, 2023.

[2] Y. Li, Z. Yu, C. Choy, et al., "Voxformer: Sparse Voxel Transformer for Camera-Based 3D Semantic Scene Completion," in Proceedings of the 2023 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 9087-9098, 2023.

[3] C. Song, M. Niu, Z. Liu, J. Cheng, P. Wang, H. Li, L. Hao, "Spatial-temporal 3D dependency matching with self-supervised deep learning for monocular visual sensing," Neurocomputing, vol. 481, no. 7, pp. 11-21, 2022.

[4] W. Yuan, X. Gu, Z. Dai, S. Zhu, and P. Tan, "NeWCRFs: Neural Window Fully-connected CRFs for Monocular Depth Estimation," in Proceedings of the 2022 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3906-3915, 2022.

[5] C. Godard, O. Mac Aodha, M. Firman, and G. J. Brostow, "Digging into Self-Supervised Monocular Depth Prediction," in Proceedings of the 2019 IEEE International Conference on Computer Vision (ICCV), pp. 3827-3837, 2019.

[6] N. Zhang, F. Nex, G. Vosselman, and N. Kerle, "Lite-Mono: A Lightweight CNN and Transformer Architecture for Self-Supervised Monocular Depth Estimation," in Proceedings of the 2023 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 18537-18546, 2023.

[7] H. Zhou, D. Greenwood, and S. Taylor, "Self-Supervised Monocular Depth Estimation with Internal Feature Fusion," in Proceedings of the British Machine Vision Conference (BMVC), 2021.

[8] Z. Shen, Y. Dai, and Z. Rao, "CFNet: Cascade and Fused Cost Volume for Robust Stereo Matching," in Proceedings of the 2021 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 13901-13910, 2021.

[9] J. Jeong, S. Jeon, and Y.S. Heo, "An Efficient Stereo Matching Network Using Sequential Feature Fusion," Electronics, vol. 10, no. 9, pp. 1045, 2021.

[10] J. Li, J. Zhao, S. Song, and T. Feng, "Occlusion Aware Unsupervised Learning of Optical Flow from Video," in Proceedings of the Thirteenth International Conference on Machine Vision, pp. 224-231, 2021

[11] M. Klingner, J. A. Termöhlen, J. Mikolajczyk, and T. Fingscheidt, "Self-supervised Monocular Depth Estimation: Solving the Dynamic Object Problem by Semantic Guidance," in Proceedings of the European Conference on Computer Vision (ECCV), pp. 582-600, 2020.

[12] M. Song, S. Lim, and W. Kim, "Monocular Depth Estimation Using Laplacian Pyramid-Based Depth Residuals," IEEE Transactions on Circuits and Systems for Video Technology, vol. 31, no. 11, pp. 4381-4393, 2021.

[13] S. F. Bhat, R. Birkl, D. Wofk, P. Wonka, and M. Müller, "ZoeDepth: Zero-shot Transfer by Combining Relative and Metric Depth," [Online]. Available: https://arxiv.org/abs/2302.12288, 2023. DOI: 10.48550/ARXIV.2302.12288.

[14] D. Eigen, C. Puhrsch, and R. Fergus, "Depth Map Prediction from a Single Image using a Multi-Scale Deep Network," in Proceedings of the Neural Information Processing Systems (NIPS), 2014.

[15] I. Laina, C. Rupprecht, V. Belagiannis, et al., "Deeper Depth Prediction with Fully Convolutional Residual Networks," in Proceedings of the 2016 Fourth International Conference on 3D Vision (3DV), pp. 239-248, 2016.

[16] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770-778, 2016.

[17] S. A. Bhat, I. Alhashim, and P. Wonka, "Adabins: Depth Estimation Using Adaptive Bins," in Proceedings of the 2021 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4008-4017, 2021.

[18] H. Fu, M. Gong, C. Wang, K. Batmanghelich, and D. Tao, "Deep Ordinal Regression Network for Monocular Depth Estimation," in Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2002-2011, 2018.

[19] T. Zhou, M. Brown, N. Snavely, et al., "Unsupervised Learning of Depth and Ego-Motion from Video," in Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6612-6619, 2017.

[20] X. Wei, Y. Zhang, Z. Li, et al., "DeepSFM: Structure from Motion via Deep Bundle Adjustment," in Proceedings of the European Conference on Computer Vision (ECCV), pp. 230-247, 2020.

[21] C. Shu, K. Yu, Z. Duan, and K. Yang, "Feature-metric Loss for Self-supervised Learning of Depth and Egomotion," in Proceedings of the European Conference on Computer Vision (ECCV), pp. 572-588, 2020.

[22] P. Ji, R. Li, B. Bhanu, and Y. Xu, "MonoIndoor: Towards Good Practice of Self-Supervised Monocular Depth Estimation for Indoor Environments," in Proceedings of the 2021 IEEE International Conference on Computer Vision (ICCV), pp. 12767-12776, 2021.

[23] K. Saito, K. Watanabe, Y. Ushiku, and T. Harada, "Maximum Classifier Discrepancy for Unsupervised Domain Adaptation," in Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3723-3732, 2018.

[24] C. Zheng, T. J. Cham, and J. Cai, "T2Net: Synthetic-to-Realistic Translation for Solving Single-Image Depth Estimation Tasks," in Proceedings of the European Conference on Computer Vision (ECCV), pp. 798-814, 2018.

[25] J. N. Kundu, P. K. Uppala, A. Pahuja, and R. V. Babu, "AdaDepth: Unsupervised Content Congruent Adaptation for Depth Estimation," in Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2656-2665, 2018.

[26] S. Zhao, H. Fu, M. Gong, and D. Tao, "Geometry-Aware Symmetric Domain Adaptation for Monocular Depth Estimation," in Proceedings of the 2019 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 9780-9790, 2019.

[27] X. Huang and S. Belongie, "Arbitrary Style Transfer in Real-Time with Adaptive Instance Normalization," in Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), pp. 1510-1519, 2017.

[28] H. Kazemi, S. M. Iranmanesh, and N. Nasrabadi, "Style and Content Disentanglement in Generative Adversarial Networks," in Proceedings of the 2019 IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 848-856, 2019.

[29] R. Wang, Z. Yu, and S. Gao, "PlaneDepth: Self-Supervised Depth Estimation via Orthogonal Planes," in Proceedings of the 2023 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 21425-21434, 2023.

[30] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin Transformer: Hierarchical Vision Transformer using Shifted Windows," in Proceedings of the 2021 IEEE International Conference on Computer Vision (ICCV), pp. 9992-10002, 2021.

[31] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," International Journal of Robotics Research, vol. 32, no. 11, pp. 1231–1237, 2013.

[32] A. Saxena, M. Sun, and A. Y. Ng, "Make3D: Learning 3D Scene Structure from a Single Still Image," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 31, no. 5, pp. 824–840, 2009.

[33] N. Silberman et al., "Indoor segmentation and support inference from RGBD images," in Proceedings of the European Conference on Computer Vision (ECCV), pp. 746–760, 2012.

[34] H. Li, A. Gordon, H. Zhao, V. Casser, and A. Angelova, "Unsupervised monocular depth learning in dynamic scenes," in Proceedings of the 2020 Conference on Robot Learning (CORL), pp. 1908-1917, 2021.

[35] V. Guizilini, R. Ambrus, S. Pillai, A. Raventos, and A. Gaidon, "3D packing for self-supervised monocular depth estimation," in Proceedings of the 2020 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2482-2491, 2020.

[36] M. Poggi, F. Aleotti, F. Tosi, and S. Mattoccia, "On the uncertainty of self-supervised monocular depth estimation," in Proceedings of the 2020 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3224-3234, 2020.

[37] Z. Fang, X. Chen, Y. Chen, and L. Van Gool, "Towards good practice for CNN-based monocular depth estimation," in Proceedings of the 2020 IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 1080-1089, 2020.

[38] X. Lyu, L. Liu, M. Wang, X. Kong, L. Liu, Y. Liu, X. Chen, and Y. Yuan, "HR-Depth: High resolution self-supervised monocular depth estimation," in Proceedings of the AAAI Conference on Artificial Intelligence (AAAI), pp. 2294-2301, 2021.

[39] W. Han, J. Yin, X. Jin, X. Dai, and J. Shen, "BRNet: Exploring comprehensive features for monocular depth estimation," in Proceedings of the European Conference on Computer Vision (ECCV), pp. 586–602, 2022.

[40] J. Wang et al., "Deep High-Resolution Representation Learning for Visual Recognition," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 43, no. 10, pp. 3349-3364, 2021.