TFDEPTH: SELF-SUPERVISED MONOCULAR DEPTH ESTIMATION WITH MULITI-SCALE SELECTIVE TRANSFORMER FEATURE FUSION

HONGLI HU[1], JUN MIAO [1,2,✉], GUANGHUI ZHU [1], JIE YAN [2], JUN CHU [3]

[1]The School of Aeronautical Manufacturing Engineering Nanchang Hangkong University; [2] Key Laboratory of Lunar and Deep Space Exploration, CAS, [3]The Department of Key Laboratory of Jiangxi Province for Image Processing and Pattern Recognition Nanchang Hangkong University

e-mail: huhongli0610@163.com; miaojun@nchu.edu.cn; 2314045303@qq.com; 2586783910@qq.com; chuj@nchu.edu.cn

(Received July 21, 2023; revised April 26, 2024; accepted April 26, 2024)

ABSTRACT

Existing self-supervised models for monocular depth estimation suffer from issues such as discontinuity, blurred edges, and unclear contours, particularly for small objects. We propose a self-supervised monocular depth estimation network with multi-scale selective Transformer feature fusion. To preserve more detailed features, this paper constructs a multi-scale encoder to extract features and leverages the self-attention mechanism of Transformer to capture global contextual information, enabling better depth prediction for small objects. Additionally, the multi-scale selective fusion module (MSSF) is also proposed, which can make full use of multi-scale feature information in the decoding part and perform selective fusion step by step, which can effectively eliminate noise and retain local detail features to obtain a clear depth map with clear edges. Experimental evaluations on the KITTI dataset demonstrate that the proposed algorithm achieves an absolute relative error (Abs Rel) of 0.098 and an accuracy rate (δ) of 0.983. The results indicate that the proposed algorithm not only estimates depth values with high accuracy but also predicts the continuous depth map with clear edges.

Keywords: monocular depth estimation, multi-scale fusion, self-supervised learning, transformer.

INTRODUCTION

Depth estimation from a single monocular image is a fundamental task in computer vision (Masoumian *et al.*, 2022) with numerous applications in fields such as robotics, augmented reality, and autonomous driving (Chen *et al.*, 2015, Li *et al.*, 2020). Accurately estimating the depth of objects in a scene from a single image remains a challenging problem due to various factors including occlusions, texture ambiguities, and lighting variations (Khan *et al.*, 2020).

Traditional monocular depth estimation methods rely on features of the image itself such as fading points, shadows, in-focus, and out-of-focus in the image to construct mathematical models to achieve a mapping between pixel values and depth values (Huang *et al.*, 2019). Although these methods have achieved certain results in some environments, they can only roughly obtain the depth information of some regions, and these methods are usually difficult to effectively deal with fine target information in complex scenes, and only fuzzy depth maps can be obtained.

In recent years, deep learning methods based on convolutional neural networks have shown promising results in estimating pixel-level depth maps from single images (Yuru and Haitao, 2020). Among these methods, supervised depth estimation techniques (Agarwal and Arora, 2023, Li *et al.*, 2023, Yuan *et al.*, 2022) have achieved remarkable performance, demonstrating good fitting with labeled data and obtaining high accuracy. However, these methods heavily rely on a large amount of accurately annotated depth labels, which are expensive and difficult to obtain. As an alternative approach, self-supervised depth estimation (Godard *et al.*, 2019, Peng *et al.*, 2021) does not require depth labels but instead leverages monocular video sequences or synchronized stereo images for network training. Self-supervised methods are cost-effective, easily applicable, and can be quickly deployed in various scenarios, thus becoming a hot research topic.

However, depth estimation for small objects in images poses challenges due to their small size, lack of distinct texture, and complex background (Farooq and Chachoo, 2023). These objects are often difficult to accurately detect and estimate in depth estimation tasks.

Most existing self-supervised depth estimation networks are based on encoder-decoder architectures, which tend to lose shape and edge information during the encoding process. Additionally, these networks heavily rely on the minimum-resolution feature maps, leading to discontinuities or inconsistencies in the generated depth maps and unclear edges. One common approach to improve the accuracy of depth estimation for small objects is using multi-scale features. However, if the selection of feature information is inadequate, the fusion process can introduce a significant amount of noise.

The successful application of Transformer models (Vaswani *et al.*, 2017) in the field of natural language processing has inspired us to introduce them into the task of single image depth estimation. By constructing a lightweight Transformer encoder, our approach is able to better capture global contextual information in images and effectively model feature correlations at different scales. Additionally, to handle small objects and details, we propose a multi-scale selective Transformer feature fusion method that can adaptively select and integrate feature information from different scales. This selective feature extraction strategy enables our model to estimate depth more accurately while effectively addressing the issue of depth map inconsistencies in small object regions, resulting in sharper object edges. We evaluate our method on the KITTI dataset and compare it with related algorithms, demonstrating the effectiveness of our approach.

In summary, this paper makes the following contributions:

– We propose an end-to-end monocular self-supervised depth estimation network that utilizes a multi-scale Transformer encoder. By leveraging self-attention mechanisms, the network effectively captures global information, thus improving the accuracy of depth estimation for small objects and ameliorating the small objects discontinuity problem.

– We introduce a multi-scale feature selection and fusion decoder that selectively chooses relevant features and progressively fuses them to obtain sharp object edges.

– Our method is evaluated on the KITTI dataset, and experimental results demonstrate that our approach can estimate continuous depth maps with clear object edges. Overall, our method outperforms other algorithms in terms of performance.

The structure of this paper is as follows: Section 2 reviews the related work in the field of monocular depth estimation; Section 3 provides a detailed description of our proposed method; Section 4 describes the experimental setup and result analysis; finally, Section 5 summarizes the main contributions of this paper and discusses future research directions.

RELATED WORKS

Supervised Learning Depth Estimation

Estimating depth from monocular images is an ill-posed problem to obtain a unique depth map (Farooq and Chachoo, 2023). With the development of technology, the ability to collect sparse depth labels is available, and the depth labels are used as supervised information to construct supervised deep learning models for monocular depth estimation. Then Eigen (Eigen *et al.*, 2014) used Deep CNN to estimate the depth of a single image, and the network was divided into two branches, one roughly predicting the global information of the whole image and the other refining the local information of the predicted image to achieve the depth estimation of a single-color image. However, the estimated depth image suffers from edge distortion and blurred object contours, Hu (Hu *et al.*, 2019) proposed a fusion module, which is able to fuse multi-scale information and effectively recover the edge information. In order to obtain more accurate depth maps, a priori information can be added to assist the network for depth estimation, Wang (Wang *et al.*, 2020) proposed a depth estimation network guided by segmentation results, which performs instance-level depth estimation by segmentation maps of the network, and then aggregates each instance to output the final depth map. More accurate depth information can be obtained. There are also related studies that convert depth estimation into a classification problem (Cao *et al.*, 2017), where the depth is discretized by classification, which is able to obtain an accurate depth map, but the estimated depth map is subject to stratification. All of the above supervised learning-based methods require a large number of accurate depth labels, and although they can achieve high accuracy, the depth labels are costly to collect and the generalization ability of the model is poor.

Self-supervised learning depth estimation

In the depth estimation of supervised methods, the depth labels used in them are difficult to obtain accurately and conveniently in practical applications, and it is difficult to collect the depth information at longer distances. In order to overcome the limitations of supervised methods, many researchers construct self-supervised methods that do not require depth labels to solve the depth estimation problem. Self-supervised depth estimation methods are mainly divided into two categories, one for binocular stereo vision-based depth estimation and the other for video continuous frame based depth estimation (Khan *et al.*, 2020). At first, by imitating the human visual system, Garg (Garg *et al.*, 2016) proposed the self-supervised depth estimation based on stereo vision, which laid the foundation for self-supervised depth estimation by using the parallax information of binocular images, reconstructing the left view with the right view, and indirectly supervising the whole network by comparing the similarity between the original left view and the reconstructed left view. After that, Godard (Godard *et al.*, 2017) proposed a self-supervised monocular depth estimation network based on stereo vision of binocular images, which can estimate the left and right parallax maps simultaneously. With the development of technology, the technique SFM (Schonberger and Frahm, 2016), which recovers camera parameters as well as the three-dimensional structure of the scene from successive frames of video, was developed to facilitate the application of self-supervised depth estimation. For example, Zhou proposed a self-supervised monocular depth estimation network framework based on SfM (Zhou *et al.*, 2017), which improved the accuracy by jointly estimating the depth estimation network and the bit-pose estimation network. Later, researchers found that depth estimation using continuous video frames suffers from dynamic scene failure. To solve this problem monodepth2 (Godard *et al.*, 2019) constructs a mask to filter out the pixel points that constitute the scene failure, Struct2depth (Casser *et al.*, 2019) uses segmentation to segment out dynamic objects and estimate them separately to avoid dynamic scene interference, and GeoNet (Yin and Shi, 2018) uses optical flow to remove dynamic objects to solve the dynamic scene failure and thus improve the network accuracy. These methods above do not require depth labels, and only monocular or binocular images are required for training, with convenient data set acquisition, wide application scenarios, and strong generalization ability.

Transformer depth estimation

Transformer has been widely used in the field of vision in recent years, such as the VIT model (Dosovitskiy *et al.*, 2021), whose self-attentive mechanism can correlate global information and has a large perceptual field, which can be a good solution for problems like classification and detection. Recently, Transformer has also been applied to solve the depth estimation problem. For example, Yuan (Yuan *et al.*, 2022) constructed a lightweight NeW CRF network using VIT as an encoder and obtained good results. Some other methods (Agarwal and Arora, 2023, Bae *et al.*, 2023) also improved based on the visual Transformer and led to a significant improvement in network performance. However, all of these methods are applied in supervised depth estimation, and the method in this paper applies the visual Transformer to self-supervised depth estimation.

OUR METHODS

Fig. 1 illustrates the network architecture of our model, which takes the image sequences as input and consists of two parts: multi-scale encoding-decoding and camera pose estimation. Images often contain information from objects at different scales. Therefore, in the encoding stage, the image frame data is passed through the lightweight Transformer Block at four scales. The multi-scale Transformer allows feature extraction at different scales, enabling the network to capture both details and global information, thus improving the model's perception of objects at different scales.

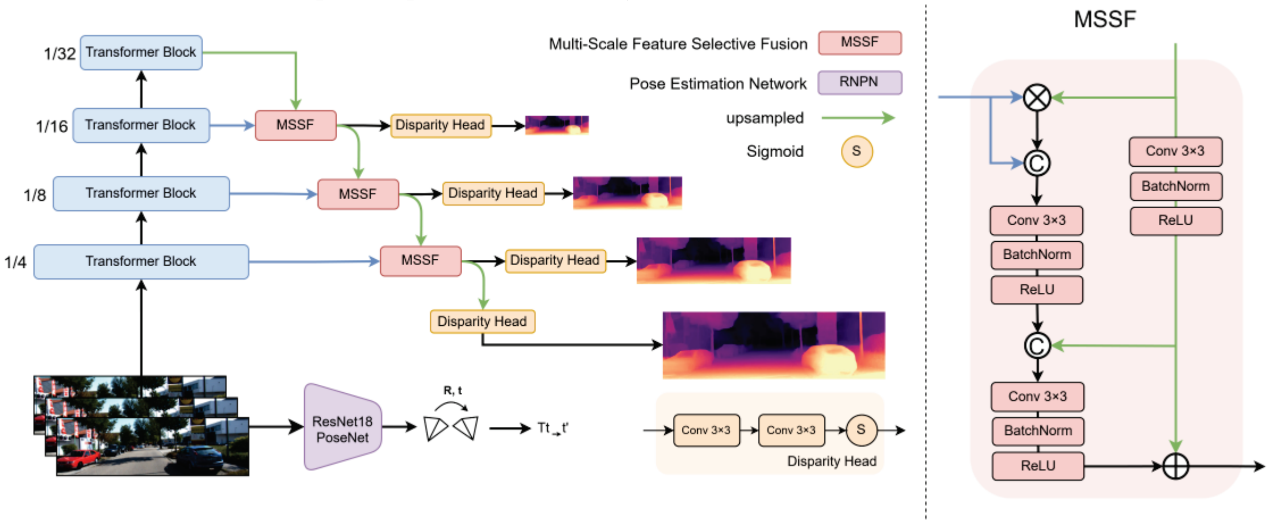


Fig. 1. *TFDepth Overall Network Architecture. The left side of the figure shows the network structure, and the right side shows the MSSF module.*

In the decoding stage, to better understand the structure and semantics of the features obtained from the multi-scale Transformer, we propose a multi-scale selective fusion (MSSF) module to capture the overall structure and local features of the data. Firstly, the last layer is upsampled to match the size of the previous layer, and both are fed into the MSSF module for selective fusion. One branch is passed through Disparity Head to output the depth map on the right, while the other branch is upsampled and fed into the next layer for further fusion. This process is repeated layer by layer, ultimately estimating depth maps at four scales.

The camera pose estimation part involves inputting the image sequences into a pose estimation network to estimate the relative pose of adjacent frames. The loss function is then computed at four scales to assist the depth estimation process.

Lightweight Vision Transformer encoder (LVT)

The Vision Transformer have a larger receptive field, enabling them to learn global image features and capture relationships between different positions in the image through self-attention mechanism (Vaswani *et al.*, 2017). Therefore, we construct a Lightweight Vision Transformer encoder to improve depth discontinuity and depth inconsistency. The attention part is proportionally reduced to decrease computation cost and improve training efficiency. The detailed network structure is shown in Fig. 2. The input image is divided into four scales, which are 1/4, 1/8, 1/16, and 1/32 of the original image size to better extract feature information. Then, images of different scales are fed into the Transformer Blocks to extract features. Assuming the input image size is H×W×3, it is first divided into N image patches, where N is H×W/d^2 and d is the size of each image patch. Next, the image patches are passed through a linear projection layer to convert the 2D images into 1D vectors. This vector is then passed through Patch Embedding and Position Embedding before being inputted into Transformer encoder, which consists of a normalization layer, multi-head attention, and multi-layer perceptron.

The encoder first normalizes the data and then computes Q (query), K (key), and V (value). In order to reduce the number of parameters, we apply convolution to scale down K and V by a factor of R after their computation. This helps in lightweighting the model. The scaled-down results are denoted as $\bar{K}$ and $\bar{V}$. Next, Q, $\bar{K}$ and $\bar{V}$ are passed into the multi-head attention for computation, and the resulting dimensions remain unchanged at $\mathbb{R}^{H\times W\times C}$. The computation process can be represented by Eq.1. Finally, the data is fed into a multi-layer perceptron (MLP), and the computation results are reshaped before being inputted into the decoder.

$$\mathrm{Attention}(Q,\bar{K},\bar{V}) = \mathrm{Softmax}\left(\frac{Q\bar{K}^{T}}{\sqrt{d_{\mathrm{head}}}}\right)\bar{V} \qquad (1)$$

Where $Q \in \mathbb{R}^{H\times W\times C}$, $\bar{K} \in \mathbb{R}^{\frac{H\times W}{R}\times C}$ $\bar{V} \in \mathbb{R}^{\frac{H\times W}{R}\times C}$, $Q\bar{K}^{T}\bar{V} \in \mathbb{R}^{H\times W\times C}$.,

As shown in Table 1, a lightweight evaluation on the VIT-B/16 (Dosovitskiy *et al.*, 2021) model is conducted. The parameter amount (Param) decreased by 29.7% and FLOPs decreased by 48.7% when testing on the KITTI dataset with size 640×192 data. This clearly demonstrates that our proposed lightweight is effective in reducing model size and improving computational speed.

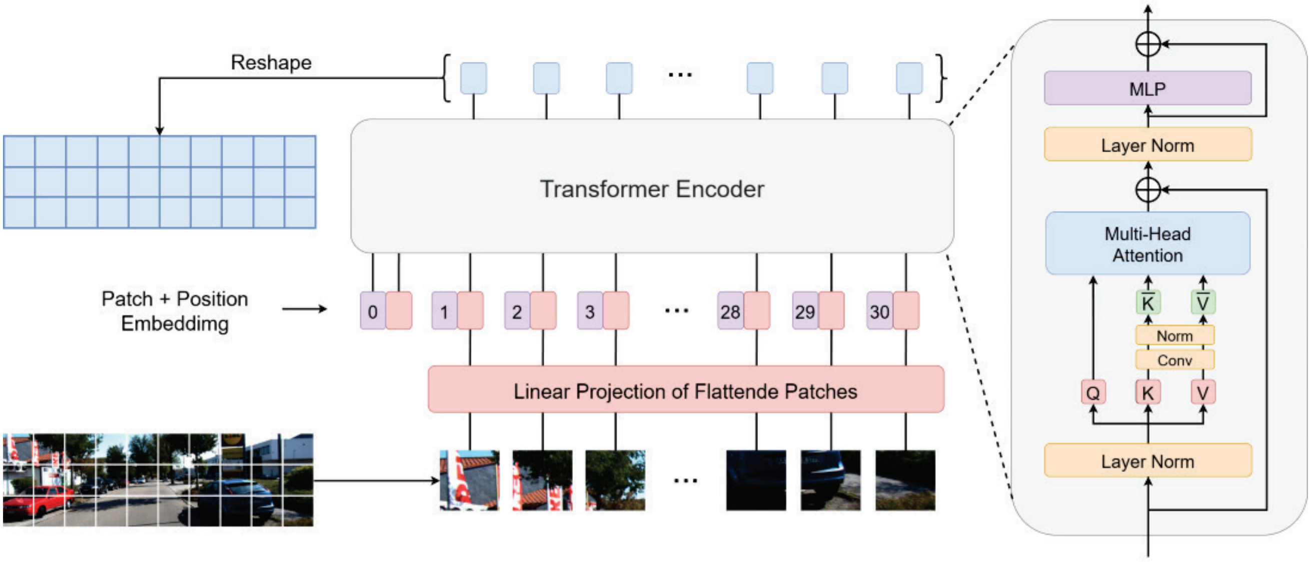


Fig. 2: *Transformer Block structure diagram.*

Table 1. *Quantitative comparisons of the parameter amount and FLOPs when testing on the KITTI dataset with size 640×192 data.*

Method	Param	FLOPs
VIT-B/16	86.63 M	41.27 G
LVT (Ours)	60.84 M	21.16 G

Multi-Scale Feature Selective Fusion Decode (MSSF)

In depth estimation, edge blurring is primarily caused by the loss of shape and edge information during the encoding process, which cannot be fully recovered during the upsampling process. Moreover, skip connections introduce a considerable amount of noise. If we can make full use of the existing feature information in the encoder and selectively fuse the features, it can significantly improve edge blurring and the problem of unclear object contours. Therefore, we propose a multi-scale selective fusion decoder (MSSF), and its specific structure is shown in Fig. 1.

MSSF is designed as a structure with two inputs and one output based on the characteristics of multi-scale, allowing fusion at each scale to fully utilize feature information and restore shape and edge information. MSSF takes the feature maps from the Transformer and upsampled feature maps as inputs. The upsampled feature maps contain rich positional and depth information, with less noise. The Transformer features contain edge and global information but have more noise. Therefore, we use attention mechanisms to multiply the two input features, refining the Transformer features and correspondingly reducing the impact of noise. At the same time, in order to achieve the optimal combination of Transformer features and upsampled features, we further select these features through three feature extraction layers.

Specifically, the first feature extraction layer learns to select and aggregate the Transformer features by taking the concatenation of the original and refined Transformer features as input. The second feature extraction layer refines and selects the upsampled features to obtain the selected upsampled features. These two selected features are then concatenated and used as input for the third feature extraction layer, which further performs feature fusion and selection between the selected Transformer features and selected upsampled features. Finally, the output is added to the selected upsampled features, achieving a data augmentation effect. The aggregated features are then passed to the next layer. The specific process can be expressed as in Eq. 2.

$$\begin{cases} f^0 = f^T \odot f^U \\ F^T = Sel\left[Concat(f^T, f^0)\right] \\ F^U = Sel\left[f^U\right] \\ F^0 = Sel\left[Concat(F^T, F^U)\right] \\ F = F^0 + F^U \end{cases} \tag{2}$$

where f^i is the original feature, F^i is the selected feature, T is the Transformer feature, U is the upsampled feature, f^0 is the refinement feature, F^0 is the final selected feature, $\odot$ is the Hadamard product operation, $Sel[\ ]$ is the selected feature operation, and $Concat(\ )$ is the splicing operation.

Pose Estimation Network

To create a lightweight network architecture, the pose estimation network adopts the ResNet18 encoder structure (He *et al.*, 2016) to estimate poses. This network inputs the adjacent frame images of a video sequence and outputs the relative pose $T_{t\to t'}$ with 6 degrees of freedom, consisting of a translation vector t and a rotation matrix R

Loss function

For self-supervised depth estimation, a real depth map acquired in advance is not required; only consecutive video frame images and a loss function are needed to constrain the whole network and estimate the corresponding depth image. The loss function in this paper is as follows:

Minimum reprojection loss. When using consecutive frames for depth estimation, a predicted image I_P can be generated by coordinate projection and bilinear interpolation $f(\ )$ using the camera parameter K, the camera pose $T_{t\to t'}$ estimated by the pose network, the depth map D_t and the image of the neighboring frames $I_{t'}$:

$$I_P = I_{t'}\left[f(D_t, T_{t\to t'}, K)\right] \tag{3}$$

In addition, the entire network can be indirectly supervised by comparing the similarity between the predicted image I_P and the current image I_t. The luminosity error function is usually constructed from the structural similarity SSIM and L_1 loss, as proposed in (Godard *et al.*, 2019):

$$F\left(I_t, I_p\right) = \alpha \frac{1-\mathrm{SSIM}\left(I_t, I_p\right)}{2} + (1-\alpha)\left\|I_t - I_P\right\|_1 \tag{4}$$

where α is usually set to 0.85 (Godard *et al.*, 2019). In addition, to obtain the minimum projection error, the minimum reprojection loss L_P can be constructed by taking the minimum occluded image calculation error in the adjacent video frames to reduce the effect of occlusion (Godard *et al.*, 2019):

$$L_p = \min_{t'} F\left(I_t, I_p\right) \tag{5}$$

where I_t is the image at momentt and I_p is the predicted image.

Smoothing loss. To reduce the error in the texture-free region, edge-aware smoothing loss L_s is used:

$$L_s = \left|\partial_x d_t^*\right| e^{-\left|\partial_x I_t\right|} + \left|\partial_y d_t^*\right| e^{-\left|\partial_y I_t\right|} \tag{6}$$

where $d_t^* = d_t/\overline{d_t}$ is the inverse of the average normalization, which prevents the estimated depth value from shrinking.

Total loss function. To take full advantage of the multiscale nature of the network, multiscale losses are constructed to supervise the whole network. In addition, we applied a masking approach (Godard *et al.*, 2019) to automatically compute the mask μ to filter static frames and objects that maintain the same motion as the camera. The final loss L is computed as a combination of the minimum reprojection loss L_P and the smoothing loss L_S at multiple scales, and then the average loss is computed to train the TFDepth:

$$L = \frac{1}{S} \sum_i^S \left(\mu L_p^i + \lambda L_s^i\right) \tag{7}$$

where S is the scale number and λ is usually set to 10^{-3}.

EXPERIMENTS

Experimental setup

We evaluate our model on the KITTI raw dataset (Geiger *et al.*, 2013), which contains a large portion of the scenarios in autonomous driving, with a total data size of approximately 175 GB. We adopt the Eigen segmentation method (Eigen *et al.*, 2014) to partition the entire dataset. Specifically, we divide it into 39,810 images representing different scenes for training, while the remaining 4,424 images are used for evaluation. The original images in the KITTI dataset have a size of 1242×375 pixels. We performed a central crop to resize them to 640×192 pixels and apply various data augmentation techniques to create an experimental dataset.

Our models are implemented in PyTorch 1.4 running on Ubuntu 18.04, and trained for 20 epochs using Adam, with a batch size of 4. We use a learning rate of 10−4 for the first 15 epochs which is then dropped to 10−5 for the remainder. Training takes about 30 hours on a NVIDIA GeForce RTX 2080TI.

To accurately evaluate the overall performance of our model, we follow the evaluation methodology outlined in the Eigen paper (Eigen *et al.*, 2014). It includes five evaluation metrics: Absolute Relative Error (Abs Rel), Squared Relative Error (Sq Rel), Root Mean Squared Error (RMSE), Root Mean Squared Logarithmic Error (RMSE log), and Accuracy within different thresholds (δ). The expressions for these metrics are as follows:

$$\text{Abs Rel} = \frac{1}{N} \sum_{i \in N} \frac{\|d_i - d_i^*\|}{d_i^*} \quad \text{Sq Rel} = \frac{1}{N} \sum_{i \in N} \frac{\|d_i - d_i^*\|^2}{d_i^*}$$
$$\text{RMSE} = \sqrt{\frac{1}{\mathrm{N}} \sum_{i \in N} \|d_i - d_i^*\|^2} \quad \text{MSElog} = \sqrt{\frac{1}{\mathrm{N}} \sum_{i \in N} \|\log d_i - (d_i^*)\|^2} \tag{8}$$

where d_i is the predicted depth value of pixel i, d_i^* is the true depth value of pixel i, N represents the number of true depth values, and thr denotes the threshold value.

Ablation Study

To better evaluate how the components of our model contribute to the overall performance, we conduct ablation experiments on the LVT encoder and MSSF module. We use the monodepth2 (Godard *et al.*, 2019) as the baseline and perform ablation experiments on different datasets to validate the effectiveness of our method. The baseline used a ResNet18 encoder and a decoder composed of upsampling and skip connections. The LVT+Skip used a LVT encoder with four scales to extract features, while the decoder remained the same as the baseline. The Baseline+MSSF employed the same encoder as the baseline, but the decoder consisted of MSSF. The TFDepth utilized LVT encoder to extract features, and the decoder was constructed using MSSF, representing the network architecture of our proposed algorithm.

We evaluate the performance of various versions of our model, trained using different forms of self-supervision: solely monocular video (M) and a combination of monocular video and stereo pairs (MS). The experimental results are shown in Table 2. Our method exhibits significant improvements over the Baseline, with notable reductions in error and increased accuracy. Particularly, Sq Rel shows the most significant decrease under M training, dropping by 14.9%, while RMSE decreases by 6%. The accuracy δ < 1.25 increases by 1%, and there is a decrease in absolute relative error. The same trends are observed under MS training. The performance enhancement is attributed to the larger receptive field of multi-scale LVT, which captures global information and addresses the issue of discontinuities in small objects. Furthermore, the MSSF decoder better selects and integrates favorable information, filters out noise, and produces clearer object edges. The combined effects of these factors contribute to achieving impressive results.

Fig. 3 reports a comparison of depth maps estimated by four different networks. The top part displays the input images and the corresponding network output images. The bottom part shows magnified local comparisons. It can be observed that the depth map estimated by the baseline exhibits discontinuities in small objects, blurry object shapes, and unclear edges. The depth map estimated by the LVT+Skip network, leveraging the feature extraction capabilities of the Transformer encoder, shows continuous depth values with small object contours, addressing the issue of discontinuities in small objects and optimizing object shapes. The baseline+MSSF network estimation produces depth maps with clear edge information, benefiting from the selective integration of multi-scale features by MSSF. TFDepth estimates depth maps that are complete, continuous, and consistent in shape, with sharp edges. It accurately captures depth information and yields dense and clear depth maps.

The experimental results indicate that our method achieves stable improvements across all evaluation metrics and consistently gains performance benefits on different datasets, demonstrating the effectiveness of the network structure.

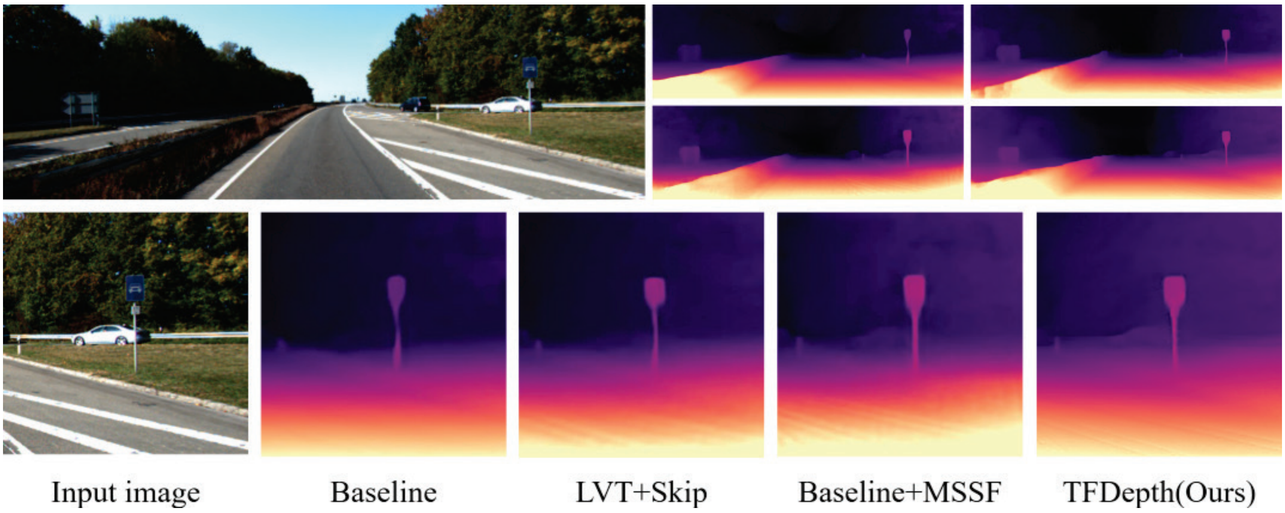
Comparative experimental analysis

We further compare our proposed method with some very recent concurrent works on monocular self-supervised. As shown in Table 3, our algorithm has achieved optimal performance across all metrics. Our algorithm shows significant improvement compared to Monodepth2(Godard *et al.*, 2019) under M training and even outperforms the high-resolution depth estimation model HR-Depth (Lyu *et al.*, 2021). The Abs Rel reaches 0.107, REMS reaches 4.567, and the accuracy reaches 0.887 when $\delta<1.25$. Our model performs better under MS training, with superior parameter values compared to the CADepth (Yan *et al.*, 2021). The Abs Rel reaches 0.098, REMS reaches 4.420, and the accuracy can go up to 0.902 when $\delta<1.253$. This indicates that our

Table 2. *Results for different variants of our model on KITTI raw dataset.*

The table shows the ablation results of two training methods, monocular video (M), monocular binocular mixed (MS), and the results show that our method is effective.

Method	Training method	LVT	MSSF	Error↓				Accuracy↑		
				Abs Rel	Sq Rel	RMSE	RMSE log	$\delta<1.25$	$\delta<1.25^2$	$\delta<1.25^3$
Baseline	M			0.115	0.903	4.863	0.193	0.877	0.959	0.981
LVT+Skip	M	✓		0.112	0.759	4.674	0.186	0.872	0.962	0.982
Baseline+MSSF	M		✓	0.110	0.758	4.621	0.186	0.876	0.962	**0.983**
TFDepth (Ours)	M	✓	✓	**0.107**	**0.754**	**4.567**	**0.184**	**0.887**	**0.963**	**0.983**
Baseline	MS			0.106	0.818	4.750	0.196	0.874	0.957	0.979
LVT+Skip	MS	✓		0.101	0.775	4.486	0.180	0.901	0.964	0.982
Baseline+MSSF	MS		✓	0.101	0.752	4.454	0.182	0.893	0.964	**0.983**
TFDepth (Ours)	MS	✓	✓	**0.098**	**0.719**	**4.420**	**0.178**	**0.902**	**0.965**	**0.983**



Input image

Baseline

LVT+Skip

Baseline+MSSF

TFDepth(Ours)

Fig. 3. *Comparison of the depths estimated by the ablation network.*

The figure shows the depth maps generated by the four networks in the ablation experiment, and the detailed parts are enlarged to show the role of each part.

algorithm has lower error, higher accuracy, and is capable of estimating more precise depth maps.

Fig. 4 demonstrates a comparison between the depth maps generated by our algorithm (TFDepth), Monodepth2 (Godard *et al.*, 2019) and PackNet (Guizilini *et al.*, 2020). This study conducted comparative experiments by selecting five specific targets from autonomous driving scenarios: intersections, urban areas, dense scenes, highways, and pedestrians. In Fig. 4, the first column is input RGB images into the networks, while the next two columns display the depth maps estimated by the networks, The depth maps calculated by our algorithm are shown in the last column. The regions of interest are highlighted in the corresponding positions for easy observation in the comparative images. It is easy to see from the figure that the depth map

Table 3. *Comparisons with recent monocular self-supervised-based models on the KITTI raw dataset.*

We conduct comparative experiments on monocular self-supervised depth estimation methods for both M and MS training methods, and the results show that our algorithm obtains optimal performance.

Method	Training method	Error↓				Accuracy↑		
		Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Zhou (Zhou *et al.*, 2017)	M	0.183	1.595	6.709	0.270	0.734	0.902	0.959
Monodepth (Godard *et al.*, 2017)	M	0.148	1.344	5.927	0.247	0.803	0.922	0.964
DDVO (Wang *et al.*, 2018)	M	0.151	1.257	5.583	0.228	0.810	0.936	0.974
DF-Net (Zou *et al.*, 2018)	M	0.150	1.124	5.507	0.223	0.806	0.933	0.973
GeoNet (Yin and Shi, 2018)	M	0.149	1.060	5.567	0.226	0.796	0.935	0.975
Struct2Depth (Casser *et al.*, 2019)	M	0.141	1.026	5.291	0.215	0.816	0.945	0.979
SGDepth (Klingner *et al.*, 2020)	M	0.117	0.970	4.844	0.196	0.875	0.958	0.980
Monodepth2 (Godard *et al.*, 2019)	M	0.115	0.903	4.863	0.193	0.877	0.959	0.981
Dyna-DM (Saunders *et al.*, 2023)	M	0.115	0.785	4.698	0.192	0.871	0.959	0.982
SAFENe (Choi *et al.*, 2020)	M	0.112	0.788	4.582	0.187	0.878	**0.963**	**0.983**
PackNet-SFM (Guizilini *et al.*, 2020)	M	0.111	0.785	4.601	0.189	0.878	0.960	0.982
HR-Depth (Lyu *et al.*, 2021)	M	0.109	0.792	4.632	0.185	0.884	0.962	**0.983**
TFDepth (Ours)	M	**0.107**	**0.754**	**4.567**	**0.184**	**0.887**	**0.963**	**0.983**
Monodepth2 (Godard *et al.*, 2019)	MS	0.106	0.818	4.750	0.196	0.874	0.957	0.979
HR-Depth (Lyu *et al.*, 2021)	MS	0.107	0.785	4.612	0.185	0.887	0.962	0.982
CADepth (Yan *et al.*, 2021)	MS	0.102	0.752	4.504	0.181	0.894	0.964	**0.983**
TFDepth(Ours)	MS	**0.098**	**0.719**	**4.420**	**0.178**	**0.902**	**0.965**	**0.983**

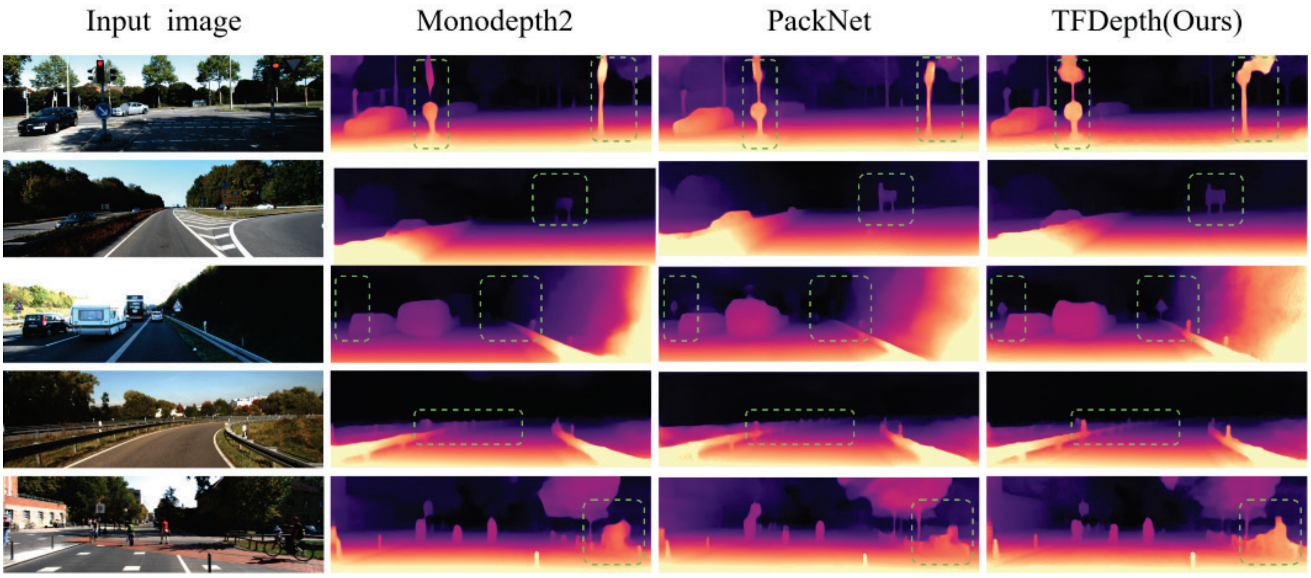


Fig. 4. *Comparison of the results of depth estimation methods. We conduct comparison experiments for five scenarios: intersection, city, dense scene, high speed, and pedestrian, and in the boxed area we can see that our model outperforms other algorithms.*

estimated by Monodepth2 in the intersection scene has the problem of discontinuous and blurred shape of the fine target, and does not estimate the triangle warning sign, while the depth map estimated by PackNet is continuous, but the depth of the traffic light and the triangle warning sign on the right side, which should be in the same plane, is not consistent, and the relative position of depth estimation is not accurate. can accurately estimate the relative position information and output a clear and continuous depth map. In urban scenes, the depth map estimated by Monodepth2 is also discontinuous and the edges of the depth map estimated by PackNet are not clear, but the algorithm in this paper can estimate the depth map with continuous and clear edges. In dense scenes the other two algorithms do not estimate the two triangular road signs information completely, TFDepth algorithm can accurately estimate the two fine targets and has strong multi-target estimation ability in dense scenes. The guardrail estimated by each algorithm in the high-speed scene shows that the TFDepth algorithm can estimate the shape and edge information of the fine targets at a deeper distance. In the pedestrian scene the other two algorithms cannot distinguish between a cyclist and a bicycle, while the algorithm in this paper, TFDepth, accurately estimates the shape and edges of both and can achieve refined depth estimation. Therefore, our algorithm outperforms other self-supervised depth estimation algorithms in estimating the depth map effect, proving the effectiveness of our algorithm.

CONCLUSION

We propose a self-supervised depth estimation framework: TFDepth. we apply the lightweight vision transformer (LVT) with self-supervised depth estimation and construct the multi-scale selection fusion module (MSSF), which selects and aggregates the multiscale features by associating the global feature information through the self-attention mechanism. Our network approach can effectively improve the problems of fine target discontinuity and edge blurring, and generate accurate, continuous and clear depth maps. In addition, experiments on the KITTI dataset show that the method in this paper can estimate richer depth information with lower error, which is significantly better than other methods and achieves optimal performance. However, this paper does not carry out a lot of work on the pose network, and later we will consider designing a more accurate pose network to assist the depth estimation and obtain more accurate depth maps

ACKNOWLEDGMENTS

This work is supported by the grant from Key Laboratory of Lunar and Deep Space Exploration, CAS.

REFERENCES

Agarwal A, Arora C (2023). Attention attention everywhere: Monocular depth prediction with skip attention. Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision:5861-70.

Bae J, Moon S, Im S (2023). Deep digging into the generalization of self-supervised monocular depth estimation. Proceedings of the AAAI Conference on Artificial Intelligence 37:187-96.

Cao Y, Wu Z, Shen C (2017). Estimating depth from monocular images as classification using deep fully convolutional residual networks. IEEE Transactions on Circuits and Systems for Video Technology 28:3174-82.

Casser V, Pirk S, Mahjourian R, Angelova A (2019). Unsupervised monocular depth and ego-motion learning with structure and semantics. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops.

Chen C, Seff A, Kornhauser A, Xiao J (2015). Deepdriving: Learning affordance for direct perception in autonomous driving. Proceedings of the IEEE international conference on computer vision:2722-30.

Choi J, Jung D, Lee D, Kim C (2020). Safenet: Self-supervised monocular depth estimation with semantic-aware feature extraction. International Conference on Learning Representations.

Dosovitskiy A, Beyer L, Kolesnikov A, Weissenborn D, Zhai X, Unterthiner T, Dehghani M, Minderer M, Heigold G, Gelly S (2021). An image is worth 16x16 words: Transformers for image recognition at scale. International Conference on Learning Representations.

Eigen D, Puhrsch C, Fergus R (2014). Depth map prediction from a single image using a multi-scale deep network. Advances in neural information processing systems 27.

Farooq H, Chachoo MA (2023). A review of monocular depth estimation methods based on deep learning. ICIDSSD 2022: Proceedings of the 3rd International Conference on ICT for Digital, Smart, and Sustainable Development, ICIDSSD 2022, 24-25 March 2022, New Delhi, India:133.

Garg R, Bg VK, Carneiro G, Reid I (2016). Unsupervised cnn for single view depth estimation: Geometry to the rescue. Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VIII 14:740-56.

Geiger A, Lenz P, Stiller C, Urtasun R (2013). Vision meets robotics: The kitti dataset. The International Journal of Robotics Research 32:1231-7.

Godard C, Mac Aodha O, Brostow GJ (2017). Unsupervised monocular depth estimation with left-right consistency. Proceedings of the IEEE conference on computer vision and pattern recognition:270-9.

Godard C, Mac Aodha O, Firman M, Brostow GJ (2019). Digging into self-supervised monocular depth estimation. Proceedings of the IEEE/CVF international conference on computer vision:3828-38.

Guizilini V, Ambrus R, Pillai S, Raventos A, Gaidon A (2020). 3d packing for self-supervised monocular depth estimation. Proceedings of the IEEE/CVF conference on computer vision and pattern recognition:2485-94.

He K, Zhang X, Ren S, Sun J (2016). Deep residual learning for image recognition. Proceedings of the IEEE conference on computer vision and pattern recognition:770-8.

Hu J, Ozay M, Zhang Y, Okatani T (2019). Revisiting single image depth estimation: Toward higher resolution maps with accurate object boundaries. 2019 IEEE winter conference on applications of computer vision (WACV):1043-51.

Huang J, Wang C, Liu Y, Bi T (2019). The progress of monocular depth estimation technology. Journal of Image and Graphics 24:2081-97.

Khan F, Salahuddin S, Javidnia H (2020). Deep learning-based monocular depth estimation methods—a state-of-the-art review. Sensors 20:2272.

Klingner M, Termöhlen J-A, Mikolajczyk J, Fingscheidt T (2020). Self-supervised monocular depth estimation: Solving the dynamic object problem by semantic guidance. Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XX 16 Springer International Publishing:582-600.

Li L, Li X, Yang S, Ding S, Jolfaei A, Zheng X (2020). Unsupervised-learning-based continuous depth and motion estimation with monocular endoscopy for virtual reality minimally invasive surgery. IEEE Transactions on Industrial Informatics 17:3920-8.

Li Z, Chen Z, Liu X, Jiang J (2023). Depthformer: Exploiting long-range correlation and local information for accurate monocular depth estimation. Machine Intelligence Research 20:837-45.

Lyu X, Liu L, Wang M, Kong X, Liu L, Liu Y, Chen X, Yuan Y (2021). Hr-depth: High resolution self-supervised monocular depth estimation. Proceedings of the AAAI Conference on Artificial Intelligence 35:2294-301.

Masoumian A, Rashwan HA, Cristiano J, Asif MS, Puig D (2022). Monocular depth estimation using deep learning: A review. Sensors 22:5353.

Peng R, Wang R, Lai Y, Tang L, Cai Y (2021). Excavating the potential capacity of self-supervised monocular depth estimation. Proceedings of the IEEE/CVF International Conference on Computer Vision:15560-9.

Saunders K, Vogiatzis G, Manso LJ (2023). Dyna-dm: Dynamic object-aware self-supervised monocular depth maps. 2023 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC):10-6.

Schonberger JL, Frahm J-M (2016). Structure-from-motion revisited. Proceedings of the IEEE conference on computer vision and pattern recognition:4104-13.

Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I (2017). Attention is all you need. Advances in neural information processing systems 30.

Wang C, Buenaposada JM, Zhu R, Lucey S (2018). Learning depth from monocular videos using direct methods. Proceedings of the IEEE conference on computer vision and pattern recognition:2022-30.

Wang L, Zhang J, Wang O, Lin Z, Lu H (2020). Sdc-depth: Semantic divide-and-conquer network for monocular depth estimation. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition:541-50.

Yan J, Zhao H, Bu P, Jin Y (2021). Channel-wise attention-based network for self-supervised monocular depth estimation. 2021 International Conference on 3D vision (3DV):464-73.

Yin Z, Shi J (2018). Geonet: Unsupervised learning of dense depth, optical flow and camera pose. Proceedings of the IEEE conference on computer vision and pattern recognition:1983-92.

Yuan W, Gu X, Dai Z, Zhu S, Tan P (2022). Neural window fully-connected crfs for monocular depth estimation. Proceedings of the IEEE/CVF conference on computer vision and pattern recognition:3916-25.

Yuru C, Haitao Z (2020). Depth estimation based on adaptive pixel-level attention model. Journal of Applied Optics 41:490-9.

Zhou T, Brown M, Snavely N, Lowe DG (2017). Unsupervised learning of depth and ego-motion from video. Proceedings of the IEEE conference on computer vision and pattern recognition:1851-8.

Zou Y, Luo Z, Huang J-B (2018). Df-net: Unsupervised joint learning of depth and flow using cross-task consistency. Proceedings of the European conference on computer vision (ECCV):36-53.