IEEE Transactions on Intelligent Transportation Systems, Available online 24 March 2022
DOI: 10.1109/TITS.2022.3159275



# Joint Optimization of Depth and Ego-Motion for Intelligent Autonomous Vehicles

Yongbin Gao, Fangzheng Tian, Jun Li, *Member, IEEE*, Zhijun Fang, *Senior Member, IEEE*, Saba Al-Rubaye, *Senior Member, IEEE*, Wei Song, and Yier Yan

***Abstract*— The three-dimensional (3D) perception of autonomous vehicles is crucial for localization and analysis of the driving environment, while it involves massive computing resources for deep learning, which can't be provided by vehicle-mounted devices. This requires the use of seamless, reliable, and efficient massive connections provided by the 6G network for computing in the cloud. In this paper, we propose a novel deep learning framework with 6G enabled transport system for joint optimization of depth and ego-motion estimation, which is an important task in 3D perception for autonomous driving. A novel loss based on feature map and quadtree is proposed, which uses feature value loss with quadtree coding instead of photometric loss to merge the feature information at the texture-less region. Besides, we also propose a novel multi-level V-shaped residual network to estimate the depths of the image, which combines the advantages of V-shaped network and residual network, and solves the problem of poor feature extraction results that may be caused by the simple fusion of low-level and high-level features. Lastly, to alleviate the influence of image noise on pose estimation, we propose a number of parallel sub-networks that use RGB image and its feature map as the input of the network. Experimental results show that our method significantly improves the quality of the depth map and the localization accuracy and achieves the state-of-the-art performance.**

Manuscript received August 19, 2021; revised November 19, 2021 and January 31, 2022; accepted February 28, 2022. This work was supported in part by the Science and Technology Innovation Action Plan of Shanghai Science and Technology Commission for Social Development Project under Grant 21DZ1204900, in part by the Guangzhou Municipal Science and Technology Project under Grant 202102010416, in part by the International Collaborative Research Program of Guangdong Science and Technology Department under Grant 2020A0505100061, in part by the Guangzhou University–The Hong Kong University of Science and Technology (GZU–HKUST) Joint Research Program under Grant YH202110, and in part by the Guangzhou Key Laboratory of Software-Defined Low Latency Network under Grant 202102100006. The Associate Editor for this article was S. Mumtaz. *(Corresponding authors: Jun Li; Zhijun Fang.)*

Yongbin Gao, Fangzheng Tian, and Zhijun Fang are with the School of Electronic and Electrical Engineering, Shanghai University of Engineering Science, Shanghai 201620, China (e-mail: gaoyongbin@sues.edu.cn; tian_fangzheng@foxmail.com; zjfang@sues.edu.cn).

Jun Li and Yier Yan are with the Research Center of Intelligent Communication Engineering, School of Electronics and Communication Engineering, Guangzhou University, Guangzhou 510006, China (e-mail: lijun52018@gzhu.edu.cn; year0080@gzhu.edu.cn).

Saba Al-Rubaye is with the School of Aerospace, Transport and Manufacturing, Cranfield University, Cranfield MK43 0AL, U.K. (e-mail: alrubaye@cranfield.ac.uk).

Wei Song is with the Department of Electronic Information and Communication Engineering, Applied Technology College of Soochow University, Suzhou 215325, China (e-mail: songw3015@suda.edu.cn).

Digital Object Identifier 10.1109/TITS.2022.3159275

***Index Terms*— Intelligent autonomous vehicles, 6G, depth estimation, ego-motion, feature quadtree, parallel sub-network, V-shaped residual network.**

## I. Introduction

THE 6G networks provide an unprecedented, seamless, reliable, efficient massive connectivity to solve the large-scale computing problem for autonomous vehicles [1]. The ego-motion of the vehicle and the distance from the surrounding environment is one of the most important tasks in the autonomous vehicles [2], [3]. Depth and ego-motion estimation plays an important role in 3D geometric understanding. Generally speaking, the depth information of the environment can be obtained through LiDAR, ultrasound and depth camera. However, the dense ground truth are difficult to acquire. LiDAR has a high-precision and long-distance sensing range, while the number of LiDAR scan lines is limited and only sparse depth values can be provided, such as in the KITTI [4] dataset, where the resolution of depth map is only 30% of the image. RGB-D cameras are mostly used for indoor depth acquisition, they have low resolution and limited precision in obtaining depth maps, and RGB-D cameras are susceptible to glass or pure black objects, which are prone to distortion and incorrect depth values [5]. In addition, these sensors are expensive. To overcome the limitations of traditional hardware-based methods, more and more attention has been paid to predicting depth from monocular images. In terms of ego-motion estimation, although monocular camera-based methods are less stable than other sensors such as stereo input or fusion of IMU and GPS, it has advantages such as lower cost, higher resolution and not limited to outdoor or indoor scenes. It is still preferred to estimate depth and ego-motion based on a monocular camera.

Image-based depth estimation and ego-motion calculation can be considered as multi-view geometry problems, which can traditionally be computed through precise linear mathematical relationships. The representative algorithms are Structure from Motion (SfM) and visual Simultaneous Localization And Mapping (vSLAM). SfM is usually used for offline calculation from a set of disordered images, while vSLAM uses sequence images to calculate pose and depth in real time [6]. These two types of methods construct a globally consistent pose and three-dimensional maps by tracking hand-crafted image geometric features (such as SIFT [7], SURF [8], ORB [9], and so on) over multiple frames, and by

optimizing methods such as Bundle Adjustment or Kalman filter. Among them, MonoSLAM [10] and ORB-SLAM [11] can create sparse 3D maps of key points. DTAM [6] and REMODE [12] generally create dense scenes by optimizing depth values. However, since these methods rely on low-level features for calculation, only reliable sparse depths can be obtained. In a challenging texture-less environment, epipolar search and block matching techniques are used to calculate the dense depth, and the effect is generally very poor. In recent years, with the emergence of Convolutional Neural Networks (CNN) [13], the performance of visual understanding has been greatly improved. Therefore, many methods based on deep learning have emerged to break through the limitations of classical methods. Deep learning has achieved good results in image-based depth map estimation. However, supervised learning requires a large amount of labelled data, either from specialized devices such as LiDAR [4] or synthetic datasets [14], which in many cases results in domain shifts [15]. To solve this problem, the method of joint estimation of depth and pose by self-supervision has recently been promoted [16]–[19]. Through depth and pose, the reference image can be projected under the perspective of the target image, jointly optimize depth and ego-motion by minimizing the photometric error between the target image and the composite image, accurate depth contributes to accurate pose estimation, and vice versa. Over the years, many scholars have done a lot of research on different methods. The current works mainly focus on the joint estimation of multi-tasks such as depth, optical flow, and normal [17], [18], [20]–[23]; the combination of classical methods and deep learning [24]–[30]; network innovation [31]–[37]; design new objective functions or training strategies [19], [38]–[47].

Recently, without using the labeled ground truth of depth map, a self-supervised deep learning network can use relative pose to synthesize the target image from the reference image, and the photometric error between synthesized target image and real one is used as self-supervisory signal. Since the photometric loss only depends on the difference in pixel intensity of the image, it cannot describe the distortion artifacts of the distorted image, so the estimated depth is blurred. At the same time, the photometric loss is easy to fall into the local minimum in the texture-less area [48]. We propose a novel loss function based on the feature values of the image and quadtree coding to solve these two problems. Different hierarchical features of target image and composite image are extracted by Vgg-16 [49] with pre-training weights. For more abstract high-level features, the differences of feature values are directly calculated, quadtree coding is used for shallow features with more detail, and feature differences in uniform textured areas are compiled into average errors to jump out of local minimum values. In terms of depth estimation, V-Net [50] combines shallow features with deep features to show good performance in depth recovery. However, due to the gap between low-level and high-level features in semantic level and spatial resolution, simple fusion may be less effective [51]. Therefore, a multi-level V-shaped residual network is proposed, which connects the multi-level V-shaped network in series through the residual method to better extract the feature information of the image. At the same time, by fusing the features of different V-shaped network outputs, the information of different levels of features is fully utilized. In depth estimation, learning-based systems usually perform quite well in terms of interior points but blur the edges of objects. We designed the contour loss function to overcome this problem. The contour loss function directly uses the photometric error of the contour in the target image and the synthesized target image to constrain the depth at the contour of the object. In previous ego-motion estimation, traditional geometric algorithms or a CNN are often used for pose estimation. We design multiple parallel sub-networks in a pose estimation network. The first sub-network takes the original RGB image as input, and the other sub-networks take the feature maps of different levels of RGB image as input. This design method avoids the influence of RGB image noise on the final pose estimation result. The main contributions are summarized as following three-folds:

1) We propose a novel loss based on feature map with quadtree encoding to calculate the feature error between the target image and its reconstructed image. The features error is calculated in the unit of quadtree block to soelve the local minimum problem during training.
2) We propose a novel joint optimization network for depth and ego-motion estimation. In the pose estimation network, several parallel subnetworks are designed. To improve the accuracy of ego-motion estimation, RGB images are fed into the network in parallel with their multilevel feature maps. Regarding the depth estimation, we propose a novel depth estimation network, which combines the advantages of V-shaped network with residual network.
3) A contour loss function is proposed to strengthen the prediction of the edge depth of the object in the image based on the learning system. Experiments show that our method is suitable for outdoor and indoor datasets. And the proposed method has excellent performance in depth estimation and ego-motion.

The remainder of this paper is organized as follows: Section II is the related work. Section III mainly introduces the system framework and principles used in this paper, including the method architecture and algorithm details. Section IV presents our experimental results and comparisons with other methods. Section V is the conclusion.

## II. Related Work

In this section, we will introduce the work of depth estimation and ego-motion estimation according to different research methods. At the same time, we also briefly introduce some related work under the 6G scenario.

### A. Joint Estimation of Multiple Tasks

Ranjan *et al.* [17] solved the unsupervised learning of several interrelated problems in low-level vision: single-view depth prediction, camera motion estimation, optical flow, and video segmentation into static scenes and moving regions. Chen *et al.* [18] proposed a self-supervised learning framework GLNet to learn depth, optical flow, camera pose and

intrinsic parameters from monocular videos. Yin and Shi [20] proposed a joint unsupervised learning framework for video monocular depth, optical flow and ego-motion estimation. Casser *et al.* [21] proposed an unsupervised monocular learning of depth and ego-motion using structure and semantics. Atapour-Abarghouei and Breckon [22] proposed a joint understanding method of geometric scene and semantic scene based on multi-task learning. Klingner and Fingscheidt [23] approached with monocular depth estimation as a secondary task, which enables us to predict the DNN's performance for various other (primary) tasks by evaluating only the depth estimation task with a physical depth measurement provided, e.g., by a LiDAR sensor. These methods mainly focus on processing multiple tasks in computer vision at the same time, and make use of the complementary advantages between different tasks to better complete tasks such as prediction and estimation.

### *B. Combination of Classical Methods and Deep Learning*

Tateno *et al.* [24] used CNN to predict the single-view depth and input it into LSD-SLAM to achieve dense reconstruction. Laidlow *et al.* [25] fused the output of a semi-dense multi-view stereo algorithm with the depth and gradient predictions of a CNN in a probabilistic fashion, using learned uncertainties produced by the network. Tang and Tan [26] used a deep neural network to predict a set of basic depth maps, combined with Levenberg-Marquardt (LM) optimization method to optimize the coefficients and poses of the depth map. Yang *et al.* [27] incorporated deep depth predictions into Direct Sparse Odometry (DSO) as direct virtual stereo measurements. Lee *et al.* [28] proposed a deep learning algorithm for single-image depth estimation based on the Fourier frequency domain analysis and proposed a new loss function, called depth balanced Euclidean loss. Wang and Xu [29] presented an architecture based on convolutional neural network and Kalman filter, which is used for unsupervised learning of accurate ego-motion and high-resolution single-view depth, and used as little as possible Parameters. Chuah *et al.* [30] predicted pixel-wise affine transformation parameters based on the depth information encoded in the aggregated cost volume, this mothed is robust against the ill-posed regions such as the textureless surfaces. Deep learning can extract deeper semantic information, and traditional methods have better interpretability. The above methods combine the advantages of these two methods to varying degrees.

### *C. Network Innovation*

Eigen *et al.* [31] presented a new method that addresses depth estimation and ego-motion by employing two deep network stacks: one that makes a coarse global prediction based on the entire image, and another that refines this prediction locally. Nath Kundu *et al.* [32] avoided image noise by adversarial learning and explicitly imposing content consistency on the adaptive target representation. Yang *et al.* [33] closely combined the predicted depth, pose and uncertainty into the direct visual ranging method to enhance front-end tracking and back-end nonlinear optimization. Pillai *et al.* [34] proposed a sub-pixel convolutional layer extension for deep super-resolution, which can accurately synthesize high-resolution differences from the corresponding low-resolution convolution features. Laina *et al.* [35] proposed a fully convolutional structure that includes residual learning. The proposed model contains fewer parameters and requires less training data. Spencer *et al.* [36] proposed DeFeat-Net, an approach to simultaneously learn a cross-domain dense feature representation, alongside a robust depth-estimation framework based on warped feature consistency. Park *et al.* [37] propose a deep sensor fusion framework consists of calibration network and depth fusion network for high-precision depth estimation. The above methods mainly focus on the improvement of the network in order to better play the role of the neural network.

### *D. Design a New Objective Function or Training Strategy*

Tang *et al.* [38] predicted by a single network trained in a tightly coupled manner through the depth and normal. Poggi *et al.* [39] solved the effect of binocular image artifacts on the depth map by moving to the trinocular domain for training. A novel interleaved training procedure was introduced that can execute the trinocular assumption outlined from the current binocular dataset. Wang *et al.* [40] used an implicit depth cue extractor which leverages dynamic and static cues to generate useful depth maps. Bozorgtabar *et al.* [19] demonstrated the benefit of using geometric information from synthetic images, coupled with scene depth information, to recover the scale in depth and ego-motion estimation from monocular videos. Mahjourian *et al.* [41] explicitly considered the inferred three-dimensional geometry of the scene and designed a consistent 3D loss function for the three-dimensional point clouds and ego-motion across consecutive frames. Wong and Soatto [42] proposed a novel objective function that exploits the bilateral cyclic relationship between the left and right disparities and introduced an adaptive regularization scheme that allows the network to handle both the co-visible and occluded regions in a stereo pair. Lai *et al.* [43] proposed a single and principled network to jointly learn spatiotemporal correspondence for stereo matching and flow estimation, with a newly designed geometric connection as the unsupervised signal for temporally adjacent stereo pairs. Heo *et al.* [44] designed a new type of filter called WSM to take advantage of the tendency of scenes to have similar depth in the horizontal or vertical direction. Srinivasan *et al.* [45] presented a novel method to train machine learning algorithms to estimate scene depths from a single image, by using the information provided by a camera's aperture as supervision. Su *et al.* [46] addressed monocular depth estimation with a general information exchange convolutional neural network. This method can capture long-range context and fine-grained features by refining the description of local context stage by stage. Jia *et al.* [47] proposed a correlation-aware structure, to dig into the relations between depths, converting the independent depths into a graph-like connected depth map. The above methods mainly study new constraint methods or training strategies to artificially improve the performance of the model.

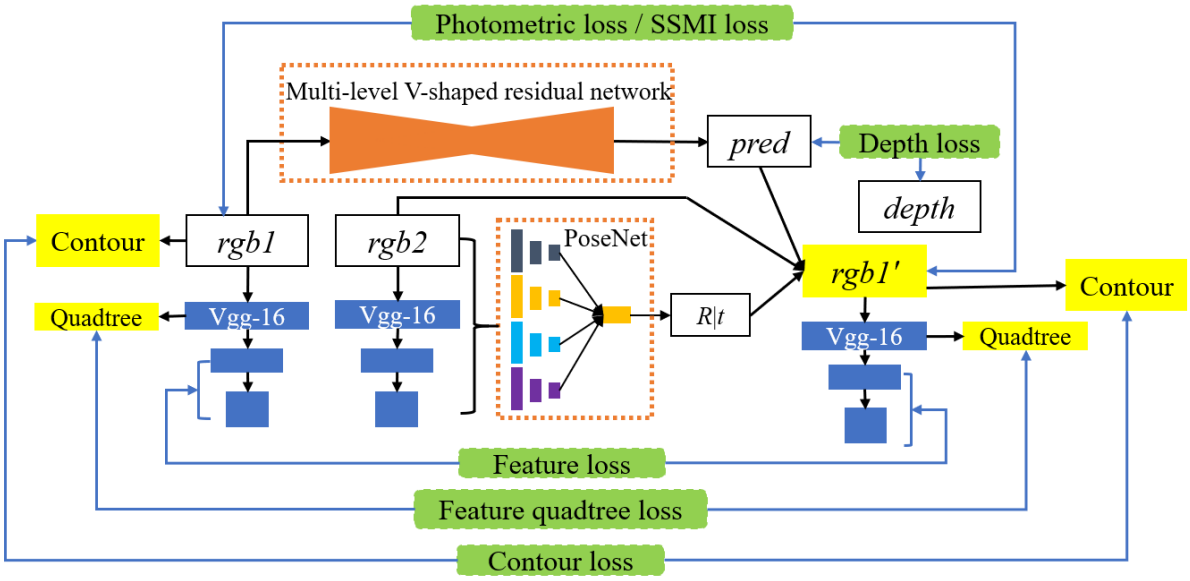


Fig. 1. Schematic diagram of the deep learning framework for joint optimization of depth and ego-motion estimation. The white boxes represent variables, the orange boxe represent the networks that need to be trained, the blue boxes represent the feature maps calculated by Vgg-16, the yellow boxes represent fixed calculations, and the green boxes represent the loss function.

### *E. Related Work Under the 6G Scenario*

Zhang *et al.* [52] proposed a fuzzy probability Bayesian network (FPBN) method for dynamic risk assessment to establish a risk propagation model for industrial control systems (ICSs). Fang *et al.* [53] considered the problem of maximizing the profit of the cloudlets' managing platform that receives computing requests from mobile users and fulfils these requests by leveraging computing service of participating cloudlets. Qu and Xiong [54] presented a Resilient, Fault-tolerant and High-efficient global replication algorithm (RFH) for distributed Cloud storage systems. Wu *et al.* [55] developed a structure fidelity data collection (SFDC) framework leveraging the spatial correlations between nodes to reduce the number of the active sensor nodes while maintaining the low structural distortion of the collected data. Li *et al.* [56] proposed a multi-step trajectory clustering method for robust Automatic Identification System (AIS) trajectory clustering. Zhou *et al.* [57] elaborated the operation details of secure spectrum sharing, incentive mechanism design, and efficient spectrum allocation. Zhou *et al.* [58] considered how to maximize the energy efficiency of M2M-TXs via the joint optimization of channel selection, peer discovery, power control, and time allocation. Li *et al.* [59] investigated the physical layer security (PLS) of the ambient backscatter NOMA systems with emphasis on reliability and security. The above methods are some noteworthy directions in the 6G scenario.

The core contribution of our method is to design a new constraint relationship based on feature graph and quadtree coding, which can effectively solve the problem that the network falls into local minimum if the texture region lacks feature information. The constraint based on contour solves the disadvantage of fuzzy edge depth. In addition, our new network architecture helps to improve the ability of network to extract and effectively use image features.

## III. System Framework

The proposed deep learning framework for joint optimization of depth and ego-motion estimation is shown in Fig. 1. The whole framework contains three neural networks with different objectives, which are a multi-level V-shaped residual network for depth estimation, PoseNet for pose estimation, and VggNet for feature extraction of target image and its reconstructed image. The multi-level V-shaped residual network is used to predict the dense depth $pred$ of the target image $rgb1$. PoseNet is used to predict the relative pose $[\boldsymbol{R}|\boldsymbol{t}]$ of the target image $rgb1$ and the reference image $rgb2$. VggNet uses a pre-trained model to extract features of RGB images. PoseNet is composed of four sub-networks in parallel, $rgb1$ and $rgb2$ and their three different levels of feature maps as the input of the four sub-networks. By using the dense depth map pred and pose $[\boldsymbol{R}|\boldsymbol{t}]$, the reference image $rgb2$ can be converted to the perspective of the target image $rgb1$ to form the reconstructed target image $rgb1'$. Where $rgb2$ is the right view in the stereo image or the keyframe in the continuous image. Based on the feature map and quadtree, the loss of the feature quadtree is designed. The deep feature value error and shallow feature quadtree block error of the target image and the reconstructed target image are used as the supervision signal to jointly optimize the depth and pose. In addition, we also use multiple loss functions such as photometric loss and structural similarity loss to jointly optimize our network model. Section *A* introduces the multi-level V-shaped residual network, section *B* describes the pose estimation network, and section *C* presents the joint optimization loss functions.

### *A. Multi-Level V-Shaped Residual Network for Depth Estimation*

*1) Network Architecture:* The multilevel V-residual network architecture is shown in Fig. 2. Each level of V-shaped

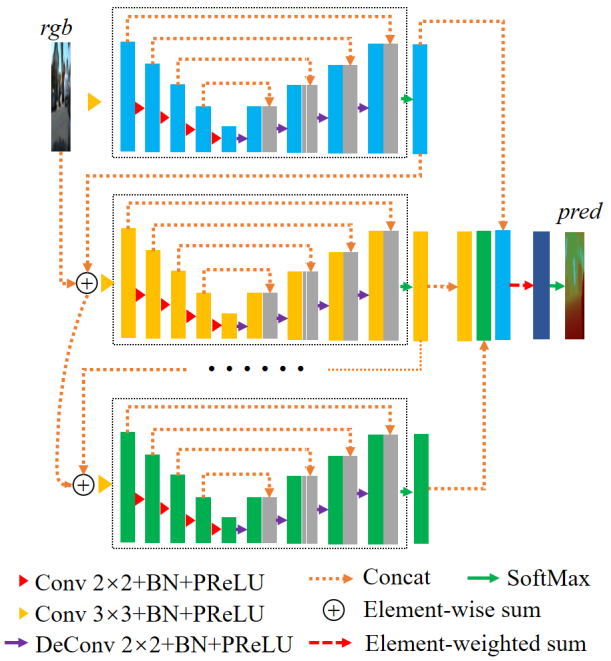


Fig. 2. Multilevel V-shaped residual network architecture diagram.

network has the same architecture, except for the different inputs. Each level of V-shaped network is connected in the residual form of residuals. The network combines the feature maps output by each level of the V-shaped network and then makes predictions, and combines the advantages of the V-shaped network and the residual network to solve the possible adverse effects of the simple fusion of low-level and high-level features. The multi-level V-shaped residual network takes the original resolution of the RGB image as input, and outputs the predicted dense depth map consistent with the resolution of the RGB image.

*2) Network Details:* Each V-shaped module adopts a V-shaped structure with a shallow network depth. The down-sampling part uses two-dimensional convolution kernel, Batch Normalization (BN) and Parametric Rectified Linear Unit (PReLU) to extract features. Between down-sampling each layer of features, a $2 \times 2$ convolution kernel with a step size of 2 is used instead of the maximum pooling layer to reduce the resolution of the feature map. This approach avoids the impact of the loss of information on the segmentation accuracy of the pooling layer in the process of dimensionality reduction. BN forces the data distribution of each layer into a normal distribution, which speeds up the convergence of the network. PReLU is used for all activation functions throughout the network. The PReLU is defined as follows:

$$\mathrm{PReLU}(x_i) = \begin{cases} x_i, & if\ x_i > 0, \\ a_i x_i, & if\ x_i \leq 0, \end{cases} \tag{1}$$

where $i$ represents different channels, $x_i$ represents the feature map of the $i$-th layer channel, $a_i$ is the parameter corresponding to it. PReLU is an activation function with different parameters for each channel (Rectified Linear Unit, ReLU). In the network training process, $a_i$ changes dynamically during the back propagation of the neural network, and the change update process of $a_i$ is shown in Eq. (2):

$$\Delta a_i := \mu \Delta a_i + \delta \frac{\partial \varepsilon}{\partial a_i}, \tag{2}$$

where $\mu$ is momentum, $\delta$ is the learning rate, $\varepsilon$ is the objective function, $\partial \varepsilon / \partial a_i$ represents the gradient of $a_i$, and the initial value of $a_i$ is 0.25. The use of PReLU reduces the risk of overfitting while hardly increasing the computational cost. In the up-sampling part, each layer is cascaded with its corresponding down-sampling layer by means of jumpers, and it is gradually enlarged by a $2 \times 2$ deconvolution kernel until it is the same size as the input.

We concatenate V-networks in a residual form. The first-level V-shaped network takes RGB images as input. The second-level V-shaped network takes the fusion result of the output and input features of the first-level V-shaped network as input. Feature fusion requires feature maps to have the same scale, we use cascade operations to aggregate these feature maps, and a $1 \times 1$ convolution layer is used to reduce the channel of the features. By analogy, the input of each level of V-shaped network can be represented by the following formula:

$$x_{VF}^{l} = \begin{cases} T_l(F(x)), & l = 1 \\ T_l(F(x, x_1, \ldots, x_{l-1})), & l = 2 \ldots \mathrm{L}, \end{cases} \tag{3}$$

where $x$ represents the original RGB image, $x_l$ represents the feature map output by the $l$-th V-shaped network, $L$ represents the number of V-shaped networks, $T_l$ represents the operation processing of the $l$-th V-shaped network, $F$ represents feature fusion processing.

Each V-shaped network is connected in series in the residual form. This forms a residual function when performing back propagation, which speeds up the network convergence in a short time. At the same time, this construction method prevents the network from forgetting the previous information in subsequent learning. The final output of the multi-level V-shaped residual network is the aggregation of the output features of the V-shaped network at all levels. The multi-level features generated by the V-shaped network at all levels are connected together along the channel, with the former features are biased towards detailed information, and the latter features are biased towards semantic information. In order to make detailed information and semantic information can be well integrated, a simple $1 \times 1$ convolution kernel method is adopted. The $1 \times 1$ convolution kernel performs a similar weighted average operation on the channel of the connected feature maps. Since the parameters of the $1 \times 1$ convolution kernel are learned through backpropagation, in the process of network training, without changing the resolution of the feature map, features are encouraged to pay attention to the channels that benefit them most. Finally, Softmax converts it into the probability of foreground and background regions to obtain the predicted dense depth map. The default configuration of the V-shaped network is 5, and each V-shaped network has 5 down-sampling and 4 up-sampling. To reduce the number of parameters, only one convolution kernel is used for each up-sampling or down-sampling.

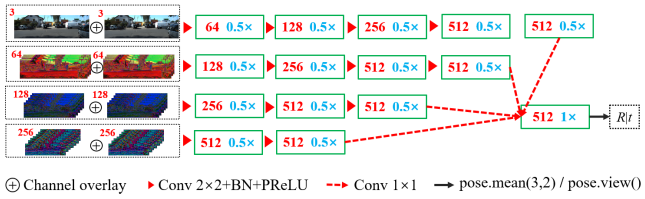


Fig. 3. Schematic diagram of PoseNet network architecture. The green box represents the feature map, the red number represents the channel number of the feature map, and the blue number represents the change of the resolution of the feature map relative to the previous feature map.

### *B. Pose Estimation Network*

*1) Network Architecture:* PoseNet is divided into four sub-networks. As shown in Fig. 3, the target image $rgb1$ and the reference image $rgb2$ are the inputs of the first sub-network. VggNet extracts three sets of feature maps of different levels of target image $rgb1$ and reference image $rgb2$. The first layer feature map has 64 channels, the second layer feature map has 128 channels, and the third layer feature map has 256 channels. Three groups of different feature maps are used as input to the other three sub-networks. PoseNet outputs a relative pose $[\boldsymbol{R}|\boldsymbol{t}]$ with 6 degrees of freedom.

*2) Network Details:* Each sub-network first superimposes the RGB image or feature map of the target view and the reference view, and then performs further processing. Since the resolution and the number of channels of each sub-network input image are inconsistent, the architecture of each sub-network is different. The specific architecture is shown in Fig. 3. PoseNet adopts ReLU as its activation function. In the output phase, the final feature maps of the four sub-networks are fused, and the final relative pose results $[\boldsymbol{R}|\boldsymbol{t}]$ are obtained by integrated network and the mean function.

### *C. Joint Optimization Loss Functions*

The proposed deep learning framework for joint optimization of depth and ego-motion estimation requires training of two network models, a multi-level V-shaped residual network and PoseNet. In order to solve the problem that the photometric loss cannot describe the distortion artifacts of the image and is prone to fall into the local minimum value in the textureless area. A new reprojection function is designed based on the feature map of the target image and the reconstructed target image as well as the feature quadtree. For more abstract deep-level features, we directly calculate the difference in feature values. Quadtree coding is applied to shallow features with more detail, and uniform texture feature areas are compiled into average errors to jump out of local minimum values.

*1) Target Image Reconstruction:* Through the dense depth map of the target image and the relative pose of the target image and the reference image, the reference image can be converted to the perspective of the target image to form a reconstructed target image. In this paper, the target image is represented by $rgb1$ and the reference image is represented by $rgb2$. For outdoor dataset, $rgb2$ is the right view in the stereo image. For indoor datasets, $rgb2$ is the keyframe in the video or continuous image. If $rgb1$ is the last frame in the continuous image, then $rgb2$ selects the image that is exactly the same as $rgb1$.

Specifically, it is divided into three steps. First, the target image can be projected into three-dimensional coordinate space $\boldsymbol{X}$ by using the camera internal parameter $\boldsymbol{K}$ and the depth map $s_{pred}$ of the target image $\boldsymbol{u_1}$ to generate point clouds in the camera coordinate system of the target image perspective. The mathematical form is: $s_{pred}\boldsymbol{u_1} = \boldsymbol{KX}$. Secondly, according to the relative pose $[\boldsymbol{R}|\boldsymbol{t}]$ of the target image and the reference image, the point clouds in the camera coordinate system of the target image perspective $\boldsymbol{X}$ is transferred to the reference image camera coordinate $\boldsymbol{X'}$ system. The mathematical form is: $\boldsymbol{X'} = [\boldsymbol{R}|\boldsymbol{t}]\boldsymbol{X}$. Finally, the point cloud $\boldsymbol{X'}$ in the reference image camera coordinate system is projected onto the reference image $\boldsymbol{u_2}$ through the camera internal parameters $\boldsymbol{K}$. The mathematical form is: $\boldsymbol{u_2} = \boldsymbol{KX'}$. Through the above three steps, a pixel-to-pixel correspondence between the target image and the reference image can be established. The mathematical model is:

$$\boldsymbol{u_2} = \boldsymbol{K}[\boldsymbol{R}|\boldsymbol{t}]s_{pred}\boldsymbol{K^{-1}u_1}, \tag{4}$$

where $s_{pred}$ is the corresponding depth value of the predicted dense depth map under the target image pixel coordinate $\boldsymbol{u_1}$, $\boldsymbol{K}$ is the camera internal parameter, $[\boldsymbol{R}|\boldsymbol{t}]$ is the transformation matrix between the target image and the reference image, $\boldsymbol{u_2}$ is the coordinate position corresponding to the reference image. Through the correspondence of Formula (4), we synthesize $rgb2$ into $rgb1'$ images from the perspective of $rgb1$ using bilinear sampling:

$$rgb1' = C(rgb2, [\boldsymbol{R}|\boldsymbol{t}], pred), \tag{5}$$

where $C$ represents the projection function corresponding to the pixels of the reference image $rgb2$ and the target image $rgb1$.

*2) Feature Map Extraction and Feature Loss:* To measure the differences between the target image $rgb1$ and reconstruct the target image $rgb1'$. We use Vgg16 to extract the feature map of $rgb1$ and $rgb1'$. Taking the resolution of the original image as input, the feature maps of the first three layers are extracted. The first layer feature map has 64 channels, the second layer feature map has 128 channels, and the third layer feature map has 256 channels. The resolution of each layer of feature maps is half of the previous layer. As shown in Fig. 4, shallow features focus on the extraction of detailed information, and deep features focus on the extraction of semantic information. The use of feature maps can solve the distortion artifacts of the reconstructed image and the difference in image intensity caused by different camera exposures under different viewing angles.

For the second and third layer of feature maps, since the semantic information is relatively strong, we directly calculate the feature value error between $rgb1$ and $rgb1'$, which is defined as:

$$L_{feature} = \frac{1}{S}\sum\nolimits_{s\in S}\left\| f_s - f_s{}' \right\|_1, \tag{6}$$

where $S$ represents the set of all feature maps of the second and third layers, $f_s$ and $f_s{}'$ represent the feature values of the

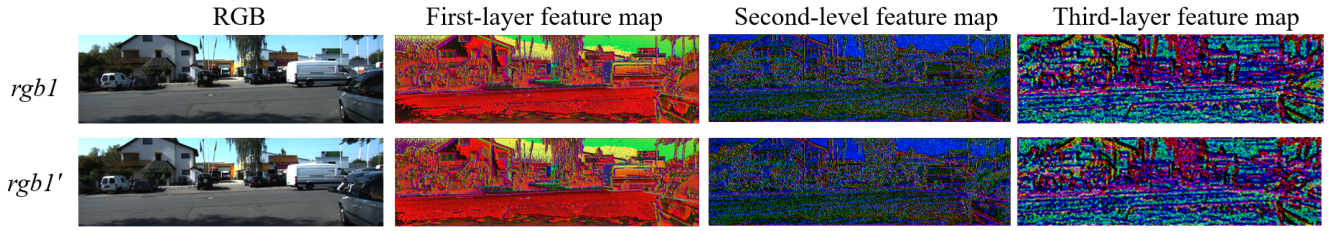


Fig. 4. Different hierarchical feature maps of target image $rgb1$ and reconstructed target image $rgb1'$. $rgb1$ is the target view, and $rgb1'$ is the synthetic view. The number of channels in each layer of feature maps in the neural network is 64, 128, and 256, respectively. Each layer only shows the visualization effects of the first three channels (the resolution of each layer's feature map is halved relative to the previous layer).

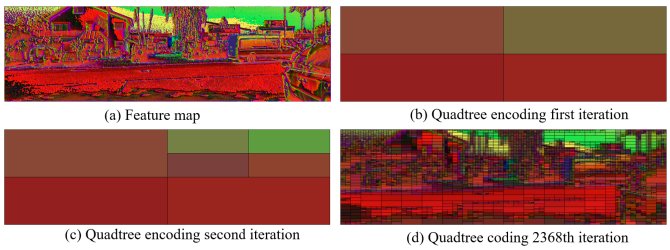


Fig. 5. Schematic diagram of quadtree encoding.

target image and the reconstructed target image under the $s$-th feature map.

*3) Quadtree and Feature Quadtree Loss:* For the first layer of feature map, it contains rich detail features. In order to avoid the direct measure difference of the feature value, the feature of the texture-less area will easily fall into the local minimum. We propose a feature quadtree loss for the first layer features.

Quadtree coding subdivides a portion of two-dimensional space into four quadrants or regions and stores relevant information in the region into quadtree nodes. This area can be square, rectangular or any shape. In this paper, we perform quadtree encoding on the first layer feature map of RGB image, First, the feature map is divided into four first-level sub-blocks, as shown in Fig. 5(a); Then the feature values are checked in each quadtree block. If they are the same, the block is no longer divided. If they are different, the block is further divided into four secondary subblocks, as shown in Fig. 5(c). This is recursively divided until the feature values of each sub-block are equal, as shown in Fig. 5(d).

The specific calculation process of the feature quadtree loss is shown in Fig. 6. First, quadtree coding is performed on the feature map of the target image $rgb1$; Secondly, the obtained quadtree mask are applied onto the feature map of the reconstruction target image $rgb1'$; Then, the average feature value is calculated in each quadtree block according to the mask on the feature map of the reconstructed target image; Finally, the feature difference between the target image and the first-layer feature map of the reconstructed target image is calculated in the unit of quadtree block, which is defined as:

$$L_{feature\_quads} = \sum\nolimits_{\alpha\in\Omega} \left\| I_{1\alpha} - I'_{1\alpha} \right\|_1, \tag{7}$$

where $I_{1\alpha}$ and $I'_{1\alpha}$ represent the feature values of $rgb1$ and $rgb1'$ first layer feature map within a quadtree block. $\Omega$ represents a collection of quadtree area blocks.

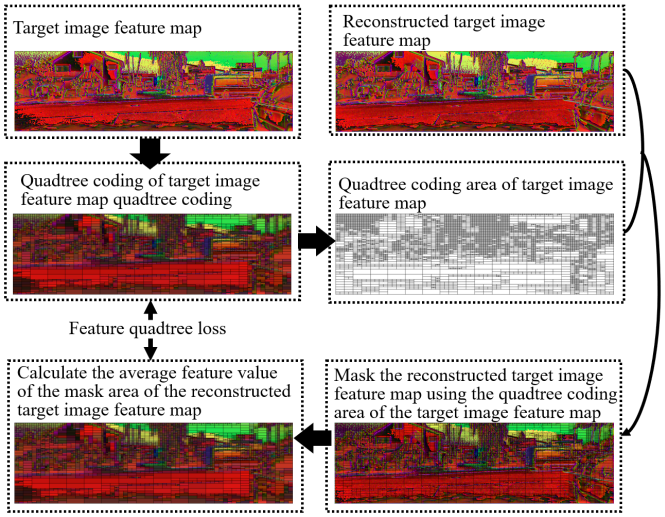


Fig. 6. Schematic diagram of quadtree encoding.

Algorithm 1 shows the computational flow of feature quadtree generation and feature quadtree loss.

In the training process of the framework, in addition to the feature loss and feature quadtree loss mentioned, we use a variety of different loss functions to supervise the training.

*4) Contour Loss:* As shown in [35], deep learning-based systems generally perform fairly well on internal points, but blur the edges of objects. To improve the accuracy of depth estimation in edge prediction, contour loss is designed to increase the penalty for contours. The contour is obtained using the basic gradient, which is defined as:

$$g = (f \oplus b) - (f \ominus b), \tag{8}$$

where $g$ represents the basic gradient image, $f$ represents the original image, $b$ represents the structural element, $\oplus$ represents the dilation operation, and $\ominus$ represents the erosion operation.

The result of the RGB image and its contour is shown in Fig. 7. The contour images extracted by $rgb1$ and $rgb1'$are directly subtracted to increase the penalty on the edges. The contour loss is defined as:

$$L_{contour} = \left\| I_{1C} - I'_{1C} \right\|_1, \tag{9}$$

where $I_{1C}$ and $I'_{1C}$ represent the intensities of images $rgb1$ and $rgb1'$ on their contours, respectively.

**Algorithm 1** Feature Quadtree Generation and Loss Function Calculation

**Require:**

First layer feature maps $rgb1f$ and $rgb1'f$ of target image $rgb1$ and reconstructed target image $rgb1'$,

Number of iterations for quadtree encoding $N$;

1: Quadtree encoding of $rgb1f$:

$i$=1

**for** $i$ to $N$:

Divide $rgb1f$ into four $i$-level sub-blocks

Compute the feature values $V$ in each quadtree block

**if** $V$ is the same:

No longer divide $i$+1 level sub-blocks

**else**:

Divide $i$+1 level sub-blocks

Obtain the $w$ quadtree regions encoded by the $rgb1f$ quadtree and the feature value $V$ of each region;

2: Mask the quadtree area of $rgb1f$ onto $rgb1'f$;

3: Directly calculate the average feature value $V'$ of each quadtree block area on $rgb1'f$;

4: Calculate the feature value difference $\left|V - V'\right|$ in each quadtree block in $rgb1f$ and $rgb1'f$;

**Ensure:**

Mean of all feature differences $\frac{1}{w}\sum_{i=1}^{N}\left(\left|V - V'\right|\right)$.

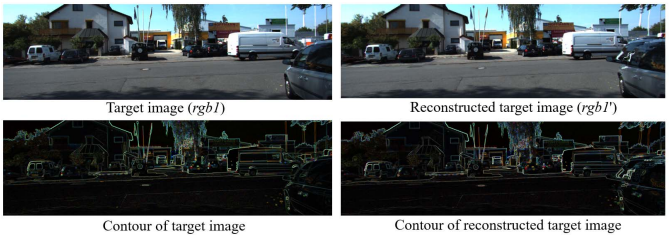


Fig. 7. RGB image and its contour map.

*5) Photometric Loss:* In order to refine the predicted dense depth map and improve the pose estimation accuracy. Pixel-wise photometric error is also used for training, which is defined as:

$$L_{photometric} = \sum_{s \in S} \left\| I_s - I'_s \right\|_1, \tag{10}$$

where $S$ represents a collection of image pixels, $I_s$ and $I'_s$ represent the color intensity of image $rgb1$ and $rgb1'$ at pixel s.

*6) SSIM Loss:* Since the structural similarity (SSIM) [22] conforms to the image quality evaluation standard of human intuition, we follow its approach and define structural similarity loss:

$$L_{ssim} = SSIM(I_1 - I'_1), \tag{11}$$

where $I_1$, $I'_1$ represent the color intensity of $rgb1$ and $rgb1'$ images.

*7) Depth Loss:* Stereo camera sensors are generally used outdoors, and RGB-D sensors are used indoors. Therefore, for outdoor datasets, stereo cameras are used to assist training. First, the sparse depth map is obtained using binocular SLAM technology, and then the predicted dense depth is constrained by the globally optimized sparse depth.

Indoor datasets have a large number of deep-labeled samples, and the network is directly trained in a supervised manner. Specifically, if the predicted depth map has data at the location corresponding to the original depth map, the difference between the two data is penalized. This loss can improve the accuracy of training, stability and convergence speed. Depth loss is defined as:

$$L_{depth}(pred, d) = \sum_{d_{i,j} \neq 0} \left\| pred_{i,j} - d_{i,j} \right\|_2^2, \tag{12}$$

where $i$, $j$ represent the depth map coordinates, $pred_{i,j}$ and $d_{i,j}$ are the depths of the predicted dense depth map and the original depth map at the $i$ and $j$ coordinates, respectively.

*8) Loss of Smooth:* Since there are no adjacent constraints between the depth values of the predicted dense depth map, in order to make the predicted dense depth map smooth, according to [42], a depth smooth loss is used, which uses the image gradient to weight the depth gradient:

$$L_{smooth} = \sum_{p} \left|\nabla pred(p)\right|^T \cdot e^{-\left|\nabla I_1(P)\right|}, \tag{13}$$

where $p$ is the pixel on the predicted dense depth map $pred$ and image $rgb1$.

In summary, the final loss function of the joint self-supervised framework includes 7 items:

$$\begin{aligned} L = {} & \alpha_1 L_{feature} + \alpha_2 L_{feature\_quads} + \alpha_3 L_{contour} \\ & + \alpha_4 L_{photomertric} + \alpha_5 L_{ssim} + \alpha_6 L_{depth} + \alpha_7 L_{smooth}, \end{aligned} \tag{14}$$

where $\alpha_1 \ldots \alpha_7$ are hyperparameters. According to experience, the hyperparameters are set to 0.8, 0.8, 0.4, 0.6, 0.4, 0.8 and 0.4, respectively.

To sum up, the whole process of training of our method is shown in Algorithm 2.

## IV. Performance Analysis

In this section, we present our experimental results to demonstrate the performance of the proposed method. The dataset used and the implementation details are described first, and the performance of our method is compared with other state-of-the-art methods. Finally, we performed ablation studies on our proposed methods to assess the contribution of different components to the overall estimation accuracy.

### A. Dataset and Implementation Details

*1) Outdoor Datasets:* The most common KITTI [4] benchmark dataset is selected. Eigen split [31] is used for training and testing, and we also use KITTI odometry dataset to evaluate our method. The dataset includes raw images, 3D point cloud data from radar, and camera trajectories, which provides an accurate but sparse semi-dense ground truth value with about 30% annotated pixels.

**Algorithm 2** Training of Joint Optimization of Depth and Ego-Motion

**Require:**
Target image $rgb1$,
reference image $rgb2$;

1: VggNet extracts the three-level feature maps of $rgb1$ and $rgb2$;
2: Using Multi-level V-shaped residual network to predict the depth $pred$ of $rgb1$;
3: Input $rgb1$ and $rgb2$ and their three-level feature maps into PoseNet to calculate the relative pose $[\boldsymbol{R}|\boldsymbol{t}]$ of $rgb1$ and $rgb2$;
4: According to the formula (5) convert $rgb2$ to $rgb1$ image perspective to get $rgb1'$;
5: Use VggNet to extract the three-level feature map of $rgb1'$;
6: Calculate the loss error:
   1) Calculate the feature loss of the second and third layer feature maps of $rgb1$ and $rgb1'$,
   2) Use Algorithm 1 to calculate the loss of the first layer feature map of $rgb1$ and $rgb1'$,
   3) Use the basic gradient to calculate the contours of $rgb1$ and $rgb1'$, and calculate the error of the contour,
   4) Calculate the photometric loss and SSMI loss of $rgb1$ and $rgb1'$, depth loss, smoothness of $pred$;

**Ensure:**
min ($loss$).

*2) Indoor Datasets:* We use the SUN-RGB D [60] dataset to train the network, which contains 10k optimized RGB-D images collected from NYUv2 [61], Berkeley B3DO [62] and SUN-3D [63]. The sequences of two public benchmark datasets, TUM RGB-D [64] and ICL-NUIM [65] are tested. The former is acquired through the Kinect sensor, and the latter is synthesized. Both of these two datasets provide the real situation of the camera trajectory and depth map.

*3) Implementation Details:* The training framework is implemented in Pytorch. We expanded the data in the form of online data enhancements, including left-right flips, random gamma color enhancements, brightness, and color shifts, where 50% samples of each augmentation method are randomly selected. The original resolution of the image is used as the input of the network, the batch size is 1, and the training is performed on a single Nvidia TITAN X. The network weights are randomly initialized using zero mean Gaussian, and the network is optimized using Adam. The learning rate is set to $10^{-4}$, and the learning rate decay method is used to reduce the learning rate by half for every two epochs. A total of 200 epochs are trained. The outdoor KITTI dataset uses a binocular visual slam to calculate the sparse depth of the left view, which is approximately 0.3% of the RGB pixel value. The indoor dataset uses the original depth for supervision. Both the sparse depth and the original depth calculated by binocular methods contain scale information, so the dense depth map and pose we predict both contain scale information.

TABLE I
ERROR AND ACCURACY METRICS. $d_{ij}^{pred}$ IS THE PREDICTED DEPTH AT $(i, j) \in I$ AND $d_{ij}^{gt}$ IS THE CORRESPONDING GROUND TRUTH

| Metric | Definition |
|---|---|
| $AbsRel$ | $\frac{1}{\lvert I\rvert}\sum_I \frac{\left\lvert d_{ij}^{pred}-d_{ij}^{gt}\right\rvert}{d_{ij}^{gt}}$ |
| $SqRel$ | $\frac{1}{\lvert I\rvert}\sum_I \frac{\left\lVert d_{ij}^{pred}-d_{ij}^{gt}\right\rVert}{d_{ij}^{gt}}$ |
| $RMSE$ | $\sqrt{\frac{1}{\lvert I\rvert}\sum_I \left\lVert d_{ij}^{pred}-d_{ij}^{gt}\right\rVert^2}$ |
| $logRMSE$ | $\sqrt{\frac{1}{\lvert I\rvert}\sum_I \left\lVert logd_{ij}^{pred}-logd_{ij}^{gt}\right\rVert^2}$ |
| $Accuracy$ | $\%\ of\ d_{ij}^{pred}\ s.t. max\left(\frac{d_{ij}^{pred}}{d_{ij}^{gt}}, \frac{d_{ij}^{gt}}{d_{ij}^{pred}}\right) = \delta < thr$ |

### B. Performance Evaluation

*1) Evaluation Methods:* For outdoor KITTI datasets, we follow most of the previous work. We quantitatively evaluate the performance of monocular depth prediction using the following metrics: mean absolute relative error ($AbsRel$), $sqRel$, root mean squared error ($RMSE$), $logRMSE$, and the accuracy under threshold ($\delta < 1.25, 1.25^2, 1.25^3$). The mathematical formulas of these measurements are shown in Table I.

For the pose estimation of the KITTI dataset, we adopt absolute trajectory error (ATE). ATE is a recognized index for evaluating the quality of camera trajectories and is defined as the root mean square error between the estimated and ground truth camera trajectory. ATE directly shows the final performance of monocular visual tracking.

For the indoor TUM RGB-D and ICL-NUIM datasets, we use the same sequence as [38] for evaluation and compare with their reported results. The percentage of correct depth (PCD) and ATE are used as the evaluation criteria for the indoor dataset. PCD is defined as the percentage of depth prediction whose absolute error is less than 10% of the true ground depth. This reveals the quality of the final depth of our keyframes and other methods.

*2) Quantitative and Qualitative Comparison of KITTI Dataset:* For the outdoor KITTI data set, binocular images are used to assist training. The use of binocular images can correct the accuracy and proportion of the predicted depth map and pose. In the process of network model training, the networks used for depth estimation and pose estimation are coupled and jointly trained. During the testing phase, we test the performance of these two different network models.

The comparison results of the depth of the KITTI dataset are shown in Table II. In the comparison methods, Refs. [31]–[34] adopt a new network structure for depth estimation. Ref. [39] uses a new training strategy. The above methods are described in detail in the related work section. Ref. [66] uses sparse ground truth depth for supervised learning. Ref. [67] refines and distills through cycle inconsistency. Ref. [68] uses a full-resolution multi-scale sampling method to reduce visual artifacts. At the same time, the minimum

TABLE II

COMPARISON RESULTS OF DEPTH ESTIMATION OF KITTI DATASET (EIGEN SPLIT [31]). ('K' REPRESENTS KITTI RAW DATASET). 'CS' REPRESENTS CITYSCAPES TRAINING DATASET. D – DEPTH SUPERVISION; S – STEREO SUPERVISION; M – MONO SUPERVISION)

| Method | Dataset | Train | Test | Resolution | $AbsRel$ | $SqRel$ | $RMSE$ | $logRMSE$ | $\delta < 1.25$ | $\delta < 1.25^2$ | $\delta < 1.25^3$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Eigen et al. Fine [31] | K | D | M | - | 0.203 | 1.548 | 6.307 | 0.282 | 0.702 | 0.890 | 0.958 |
| Kuznietsov et al. [66] | K | D | M | 621×187 | 0.113 | 0.741 | 4.621 | 0.189 | 0.862 | 0.960 | 0.986 |
| Nath et al. [32] | K | Semi | M | 1242×375 | 0.167 | 1.257 | 5.578 | 0.237 | 0.771 | 0.922 | 0.971 |
| Yang et al. [33] | K | MS | M | - | 0.099 | 0.763 | 4.485 | 0.185 | 0.885 | 0.958 | 0.979 |
| Pilzer et al. [67] | CS+K | S | M | 512×256 | **0.098** | 0.831 | 4.656 | 0.202 | 0.882 | 0.948 | 0.973 |
| Pillai et al. [34] | K | S | M | 1024×384 | 0.112 | 0.875 | 4.958 | 0.207 | 0.852 | 0.947 | 0.977 |
| Godard et al. [68] | K | S | M | 1024×320 | 0.107 | 0.849 | 4.764 | 0.201 | 0.874 | 0.953 | 0.977 |
| Poggi et al. [39] | CS+K | S | M | - | 0.111 | 0.849 | 4.822 | 0.202 | 0.865 | 0.952 | 0.978 |
| Our method | K | S | M | 1242×375 | **0.098** | **0.721** | **4.458** | **0.181** | **0.892** | **0.963** | **0.988** |

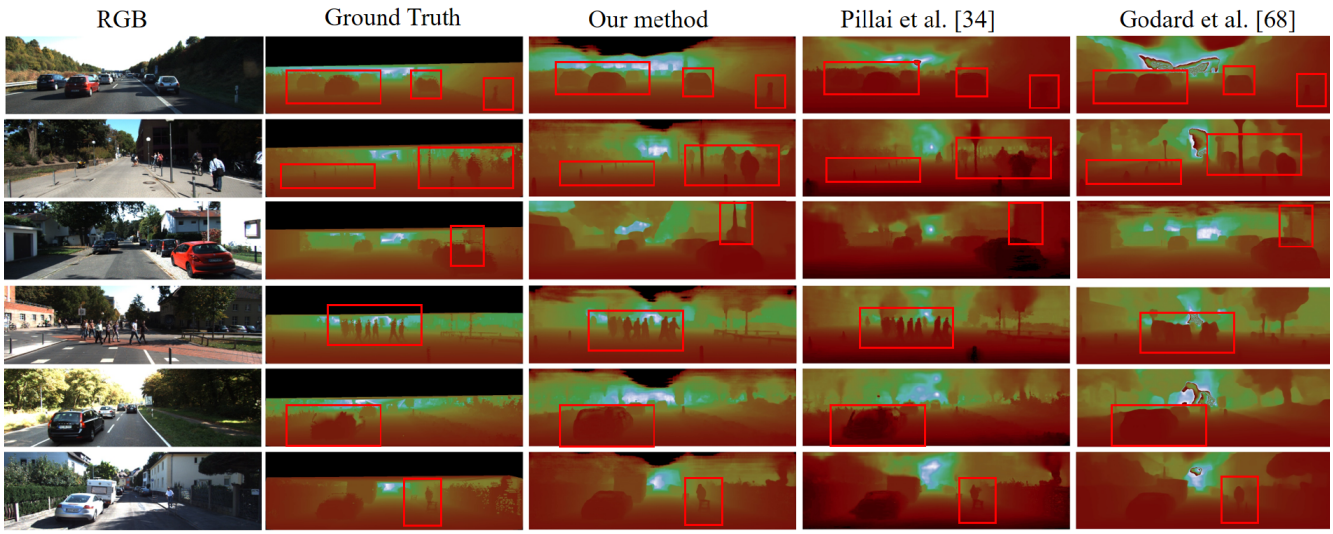


Fig. 8. The depth comparison of some frames of the KITTI dataset.

reprojection constraint can handle the occlusion problem robustly. The quantitative comparison in Table II shows that the error performance of our method is lower than most of the current methods, and the accuracy is high, which proves the advantages of the proposed method. Different ways of modifying the network structure have a great impact on the experimental results, which shows that the network structure has a great impact on the depth structure. The results in [67] show that refinement and distillation are beneficial to the accuracy of depth estimation.

Fig. 8 shows some qualitative results of some pictures of the KITTI dataset. The depth image output by our method has clearer object boundaries. This attributes to the fact that our multi-level V-shaped residual network is able to learn more complex semantic representations. In addition, we use feature loss and feature quadtree loss to make our results retain more image details.

We evaluate the relative pose estimation based on the KITTI odometer sequence 09/10. ORB-SLAM [69] is a classic geometry-based traditional algorithm. Refs. [19], [40], [41] mainly calculate poses by designing new constraints or improving training strategies. Refs. [17], [18], [20], [21] predict multiple visual tasks at the same time, and estimate the camera pose by combining the advantages of different tasks. Ref. [70] is a classic method of self-supervised joint optimization of depth and ego-motion. As shown in Table III,

TABLE III

THE ATE OF KITTI ODOMETRY DATASET

| Method | Sequence 09 | Sequence 10 |
|---|---|---|
| ORB-SLAM(full) [69] | 0.014±0.008 | 0.012±0.011 |
| Wang et al. [40] | 0.016±0.008 | 0.014±0.009 |
| Chen et al. [18] | 0.011±0.006 | 0.011±0.009 |
| Ranjans et al. [17] | 0.012±0.007 | 0.012±0.008 |
| SynDeMo [19] | 0.014±0.008 | 0.013±0.015 |
| Struct2depth [21] | **0.011±0.006** | 0.011±0.010 |
| Mahjourian et al. [41] | 0.013±0.010 | 0.012±0.011 |
| Geonet [20] | 0.012±0.007 | 0.012±0.009 |
| SfMLearner [70] | 0.016±0.009 | 0.013±0.009 |
| Our method | **0.011±0.006** | **0.011±0.008** |

our method is still significantly outperforms than other state-of-the-art methods. It depends on the original resolution of the image as the input of the network, which can retain a lot of detailed information. Instead of intensity values, the feature map is used as input to eliminate noise caused by camera exposure and artifacts caused by high-speed camera movement. Meanwhile, the feature quadtree loss can avoid the model to fall in a local minimum.

*3) Quantitative and Qualitative Comparison of Indoor Datasets:* In order to further verify the validity of the model, we test on indoor data. The comparison results of the indoor dataset PCD are shown in Table IV. Ref. [38] proposes a new network training strategy. Refs. [24], [25] combines traditional methods with deep learning methods. Ref. [35]

TABLE IV
THE ATE OF KITTI ODOMETRY DATASET

| Sequence | TUM/seq1 | TUM/seq2 | TUM/seq3 | ICL/office0 | ICL/office1 | ICL/office2 | ICL/living0 | ICL/living1 | ICL/living2 |
|---|---|---|---|---|---|---|---|---|---|
| S2D [38] | 53.287 | 66.628 | 37.683 | 27.445 | 19.702 | 27.059 | 19.337 | 25.090 | **68.907** |
| CNN-SLAM [24] | 12.477 | 24.077 | 27.396 | 19.410 | 29.150 | 37.226 | 12.840 | 13.038 | 16.560 |
| Laina [35] | 12.982 | 15.412 | 9.450 | 17.194 | 20.838 | 30.639 | 15.008 | 11.449 | 33.010 |
| LSD-BS [71] | 3.797 | 3.966 | 6.449 | 0.603 | 4.759 | 1.435 | 1.443 | 3.030 | 1.807 |
| DeepFusion [25] | 8.069 | 14.774 | 27.200 | 21.090 | 37.420 | 30.180 | 24.223 | 14.001 | 25.235 |
| REMODE [12] | 9.548 | 12.651 | 6.739 | 4.479 | 3.132 | 16.708 | 4.479 | 2.427 | 8.681 |
| our method | **62.362** | **71.645** | **68.721** | **29.174** | **39.737** | **51.361** | **48.431** | **31.472** | 63.490 |

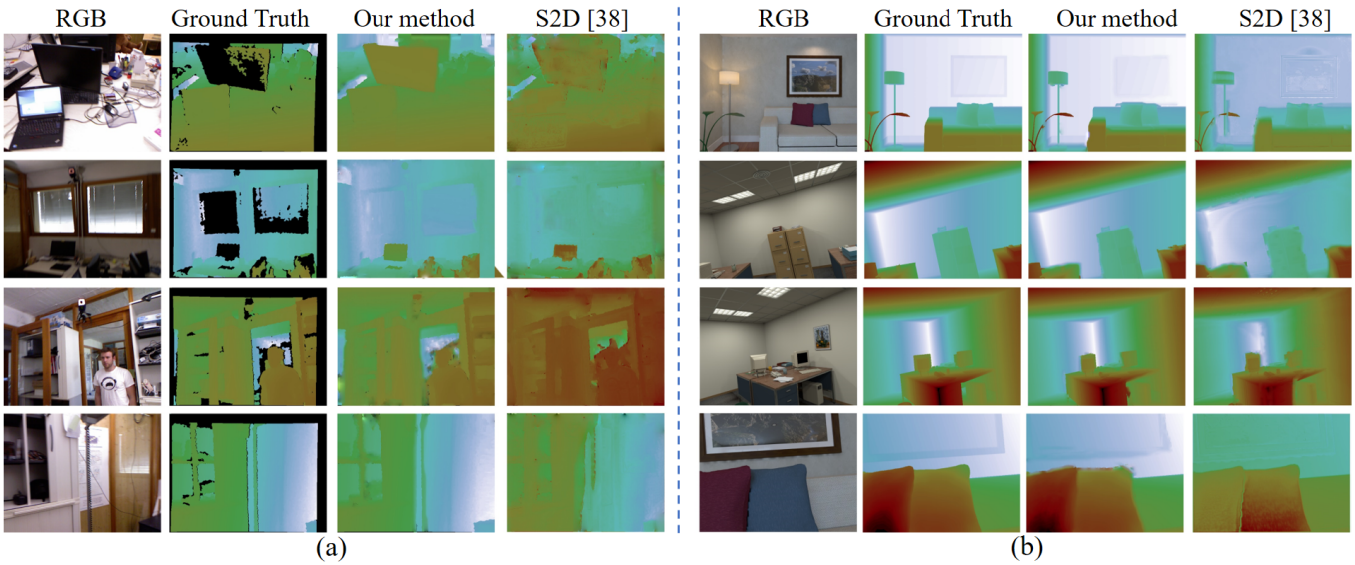


Fig. 9. Depth maps of some frames of indoor TUM RGB-D and ICL-NUIM datasets. (a) is TUM RGB-D dataset, (b) is ICL-NUIM dataset.

TABLE V
THE ATE OF KITTI ODOMETRY DATASET

| Sequence | TUM/seq1 | TUM/seq2 | TUM/seq3 | ICL/office0 | ICL/office1 | ICL/office2 | ICL/living0 | ICL/living1 | ICL/living2 |
|---|---|---|---|---|---|---|---|---|---|
| S2D [38] | 0.071 | 0.078 | 0.072 | 0.132 | 0.131 | 0.085 | 0.137 | 0.082 | 0.045 |
| CNN-SLAM [24] | 0.542 | 0.243 | 0.214 | 0.266 | 0.157 | 0.213 | 0.196 | **0.059** | 0.323 |
| Laina [35] | 0.809 | 1.337 | 0.724 | 0.337 | 0.218 | 0.509 | 0.230 | 0.060 | 0.380 |
| LSD-BS [71] | 1.717 | 0.106 | 0.037 | 0.587 | 0.790 | 0.172 | 0.894 | 0.540 | 0.211 |
| ORB [11] | 1.206 | 0.495 | 0.733 | 0.430 | 0.780 | 0.860 | 0.493 | 0.129 | 0.663 |
| DSO [72] | 1.221 | 0.123 | 0.648 | 1.118 | 0.633 | 0.795 | 0.404 | 0.187 | 0.668 |
| our method | **0.052** | **0.048** | **0.049** | **0.113** | **0.109** | **0.069** | **0.117** | 0.074 | **0.043** |

designs a new network architecture. Ref. [71] proposes a direct featureless monocular SLAM algorithm. Ref. [12] combines the latest techniques of Bayesian estimation and convex optimization of image processing to estimate the depth of the image. The comparison results show that our method is equally effective for indoor data sets. We have achieved better results in all sequences, except for the ICL/living2 sequence. Especially in TUM/seq1, TUM/seq3, ICL/office1, ICL/office2, ICL/living0 sequence, our method has made significant progress. This shows that our method is not affected by outdoor or indoor environments. In addition, the experimental data shows that the methods based on deep learning have better performance than the traditional methods to a certain extent.

In order to show the effectiveness of our method more intuitively, we show some qualitative results of the output of some keyframes of the TUM RGB-D and ICL-NUIM datasets in Fig. 9. From the qualitative results, our predicted depth images also have clearer object boundaries. At the same time, our method overcomes the influence of some special objects such as glass on depth cameras.

The comparison results of the indoor dataset ATE are shown in Table V. Refs. [24], [35], [38] adopt deep learning-based method for pose estimation. Refs. [24], [35], [38] adopt deep learning-based method for pose estimation. Refs. [11], [71], [72] use the traditional geometry-based method to calculate the camera pose. It can be seen from the experimental data in the table that the output result of the method based on deep learning is more accurate and stable than that of the method based on geometry. Our method is generally better than other methods, and has more accurate and stable results. From tables IV and V, we can see that the quality of pose estimated by our method is directly related to the quality of depth map, which is mainly due to the joint optimization of depth and pose estimation during model training.

TABLE VI
COMPARISON RESULTS OF ABLATION STUDIES (CELLS WITH ✓INDICATE THAT THEY CONTAIN THIS COMPONENT)

| Method | | | | | KITTI | | TUM/seq1 | | ICL/office2 | |
|---|---|---|---|---|---|---|---|---|---|---|
| V-Net | Multi-level V-shaped residual network | PoseNet (Only take image as input) | PoseNet (Contains four sub-networks) | Feature loss Feature quadtree loss | RMSE (Eigen split) | ATE (Seq. 09) | PCD | ATE | PCD | ATE |
| ✓ | | ✓ | | | 4.647 | 0.013±0.015 | 52.571 | 0.074 | 43.370 | 0.082 |
| | ✓ | ✓ | | | 4.550 | 0.012±0.010 | 58.327 | 0.068 | 48.591 | 0.077 |
| | ✓ | | ✓ | | 4.493 | 0.011±0.009 | 60.625 | 0.056 | 50.840 | 0.071 |
| | ✓ | | ✓ | ✓ | **4.458** | **0.011±0.006** | **62.362** | **0.052** | **51.361** | **0.069** |

### C. Ablation Study

In order to better assess the contribution of each component of our method to the prediction accuracy, we perform ablation studies by changing the different components of the method, the results are shown in Table VI. From the data of the first and second sets of experiments, it can be seen that the use of multi-level V-shaped residual network improves the accuracy of depth map prediction, and also affects the result of pose estimation to a certain extent. Through the second and third groups of experimental data, it can be seen that the accuracy of pose estimation by PoseNet composed of multiple sub-networks has been improved. By comparing the third and fourth sets of data, it is shown that the feature loss and the feature quadtree loss improve the accuracy of depth estimation and pose estimation to a certain extent, while reducing the error fluctuation of pose estimation. Throughout the ablation experiment, it can be seen that depth estimation and pose estimation complement each other.

## V. CONCLUSION AND THE FUTURE WORK

In this paper, a novel deep learning framework for joint optimization of depth and ego-motion estimation with 6G enabled network is proposed, and a novel loss based on feature map and quadtree coding is presented. Deep feature loss overcomes the problem that photometric loss cannot describe image distortion artifacts. Shallow feature quadtree loss compiles the feature difference of the uniform texture area into the average error to make it jump out of the local minimum. Multilevel V-shaped residual network combines the advantages of V-shaped network and residual network to extract better feature information. The fusion of the features output by different V-shaped networks makes full use of the shallow detail information and the deep semantic information. PoseNet, which is composed of several parallel subnetworks, overcomes the influence of noise in the image on the final pose prediction result, as well as the influence of the rapid camera movement on the image. The proposed method has shown excellent performance on both indoor and outdoor datasets. The proposed method currently has certain limitations. Firstly, the complex network structure increases the computational complexity. Secondly, the contour loss function is too rough. In the future, we should focus on self-supervised or unsupervised methods, and study the use of more reasonable methods to solve the problem of depth blur at the edges of objects at the same time. In addition, a more efficient network model should be designed to reduce the computational complexity.

## REFERENCES


[1] Y. Qu and N. Xiong, "RFH: A resilient, fault-tolerant and high-efficient replication algorithm for distributed cloud storage," in *Proc. 41st Int. Conf. Parallel Process.*, Sep. 2012, pp. 520–529.

[2] K. Gao, F. Han, P. Dong, N. Xiong, and R. Du, "Connected vehicle as a mobile sensor for real time queue length at signalized intersections," *Sensors*, vol. 19, no. 9, p. 2059, 2019.

[3] W. Guo, N. Xiong, A. V. Vasilakos, G. Chen, and H. Cheng, "Multi-source temporal data aggregation in wireless sensor networks," *Wireless Pers. Commun.*, vol. 56, no. 3, pp. 359–370, Feb. 2011.

[4] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 3354–3361.

[5] O. Wasenmüller and D. Stricker, "Comparison of Kinect V1 and V2 depth images in terms of accuracy and precision," in *Proc. Asian Conf. Comput. Vis.*, Cham, Switzerland: Springer, 2016, pp. 34–45.

[6] R. A. Newcombe, S. J. Lovegrove, and A. J. Davison, "DTAM: Dense tracking and mapping in real-time," in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 2320–2327.

[7] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2003.

[8] H. Bay, T. Tuytelaars, and L. V. Gool, "Surf: Speeded up robust features," in *Proc. Eur. Conf. Comput. Vis.* Berlin, Germany: Springer, Heidelberg, 2006, pp. 404–417.

[9] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 2564–2571.

[10] A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse, "MonoSLAM: Real-time single camera SLAM," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 6, pp. 1052–1067, Jun. 2007.

[11] R. Mur-Artal and J. D. Tardós, "ORB-SLAM2: An open-source slam system for monocular, stereo, and RGB-D cameras," *IEEE Trans. Robot.*, vol. 33, no. 5, pp. 1255–1262, Oct. 2017.

[12] M. Pizzoli, C. Forster, and D. Scaramuzza, "REMODE: Probabilistic, monocular dense reconstruction in real time," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2014, pp. 2609–2616.

[13] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 2, pp. 84–90, Jun. 2012.

[14] N. Mayer *et al.*, "A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 4040–4048.

[15] Y. Chen, W. Li, and L. V. Gool, "ROAD: Reality oriented adaptation for semantic segmentation of urban scenes," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7892–7901.

[16] T. Shen *et al.*, "Beyond photometric loss for self-supervised ego-motion estimation," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 6359–6365.

[17] A. Ranjan *et al.*, "Competitive collaboration: Joint unsupervised learning of depth, camera motion, optical flow and motion segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 12240–12249.

[18] Y. Chen, C. Schmid, and C. Sminchisescu, "Self-supervised learning with geometric constraints in monocular video: Connecting flow, depth, and camera," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 7063–7072.

[19] B. Bozorgtabar, M. S. Rad, D. Mahapatra, and J.-P. Thiran, "SynDeMo: Synergistic deep feature alignment for joint learning of depth and ego-motion," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 4210–4219.

[20] Z. Yin and J. Shi, "GeoNet: Unsupervised learning of dense depth, optical flow and camera pose," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1983–1992.

[21] V. Casser, S. Pirk, R. Mahjourian, and A. Angelova, "Unsupervised monocular depth and ego-motion learning with structure and semantics," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2019, pp. 381–388.

[22] A. Atapour-Abarghouei and T. P. Breckon, "Veritatem dies Aperit–temporally consistent depth prediction enabled by a multi-task geometric and semantic scene understanding approach," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3373–3384.

[23] M. Klingner and T. Fingscheidt, "Online performance prediction of perception DNNs by multi-task learning with depth estimation," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 7, pp. 4670–4683, Jul. 2021.

[24] K. Tateno, F. Tombari, I. Laina, and N. Navab, "CNN-SLAM: Real-time dense monocular SLAM with learned depth prediction," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6243–6252.

[25] T. Laidlow, J. Czarnowski, and S. Leutenegger, "DeepFusion: Real-time dense 3D reconstruction for monocular SLAM using single-view depth and gradient predictions," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 4068–4074.

[26] C. Tang and P. Tan, "BA-Net: Dense bundle adjustment network," 2018, *arXiv:1806.04807*.

[27] N. Yang, R. Wang, J. Stuckler, and D. Cremers, "Deep virtual stereo odometry: Leveraging deep depth prediction for monocular direct sparse odometry," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Jun. 2018, pp. 817–833.

[28] J.-H. Lee, M. Heo, K.-R. Kim, and C.-S. Kim, "Single-image depth estimation based on Fourier domain analysis," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 330–339.

[29] Y. Wang and Y. F. Xu "Unsupervised learning of accurate camera pose and depth from video sequences with Kalman filter," *IEEE Access*, vol. 7, pp. 32796–32804, 2019.

[30] W. Chuah, R. Tennakoon, R. Hoseinnezhad, and A. Bab-Hadiashar, "Deep learning-based incorporation of planar constraints for robust stereo depth estimation in autonomous vehicle applications," *IEEE Trans. Intell. Transp. Syst.*, early access, Feb. 26, 2021, doi: 10.1109/TITS.2021.3060001.

[31] D. Eigen, C. Puhrsch, and R. Fergus, "Depth map prediction from a single image using a multi-scale deep network," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2366–2374.

[32] J. N. Kundu, P. K. Uppala, A. Pahuja, and R. V. Babu, "AdaDepth: Unsupervised content congruent adaptation for depth estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2656–2665.

[33] N. Yang, L. von Stumberg, R. Wang, and D. Cremers, "D3 VO: Deep depth, deep pose and deep uncertainty for monocular visual odometry," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 1281–1292.

[34] S. Pillai, R. Ambrus, and A. Gaidon, "SuperDepth: Self-supervised, super-resolved monocular depth estimation," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 9250–9256.

[35] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab, "Deeper depth prediction with fully convolutional residual networks," in *Proc. 4th Int. Conf. 3D Vis. (3DV)*, Oct. 2016, pp. 239–248.

[36] J. Spencer, R. Bowden, and S. Hadfield, "DeFeat-Net: General monocular depth via simultaneous unsupervised representation learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 14402–14413.

[37] K. Park, S. Kim, and K. Sohn, "High-precision depth estimation using uncalibrated LiDAR and stereo fusion," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 1, pp. 321–335, Jan. 2020.

[38] J. Tang, J. Folkesson, and P. Jensfelt, "Sparse2Dense: From direct sparse odometry to dense 3-D reconstruction," *IEEE Robot. Autom. Lett.*, vol. 4, no. 2, pp. 530–537, Apr. 2019.

[39] M. Poggi, F. Tosi, and S. Mattoccia, "Learning monocular depth estimation with unsupervised trinocular assumptions," in *Proc. Int. Conf. 3D Vis. (3DV)*, Sep. 2018, pp. 324–333.

[40] J. Wang, G. Zhang, Z. Wu, X. Li, and L. Liu, "Self-supervised joint learning framework of depth estimation via implicit cues," 2020, *arXiv:2006.09876*.

[41] R. Mahjourian, M. Wicke, and A. Angelova, "Unsupervised learning of depth and ego-motion from monocular video using 3D geometric constraints," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5667–5675.

[42] A. Wong and S. Soatto, "Bilateral cyclic constraint and adaptive regularization for unsupervised monocular depth prediction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 5644–5653.

[43] H.-Y. Lai, Y.-H. Tsai, and W.-C. Chiu, "Bridging stereo matching and optical flow via spatiotemporal correspondence," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1890–1899.

[44] M. Heo, J. Lee, K.-R. Kim, H.-U. Kim, and C.-S. Kim, "Monocular depth estimation using whole strip masking and reliability-based refinement," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 36–51.

[45] P. P. Srinivasan, R. Garg, N. Wadhwa, R. Ng, and J. T. Barron, "Aperture supervision for monocular depth estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6393–6401.

[46] W. Su, H. Zhang, Q. Zhou, W. Yang, and Z. Wang, "Monocular depth estimation using information exchange network," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 6, pp. 3491–3503, Jun. 2021.

[47] S. Jia, X. Pei, X. Jing, and D. Yao, "Self-supervised 3D reconstruction and ego-motion estimation via on-board monocular video," *IEEE Trans. Intell. Transp. Syst.*, early access, Apr. 19, 2021, doi: 10.1109/TITS.2021.3071428.

[48] J. Watson, M. Firman, G. Brostow, and D. Turmukhambetov, "Self-supervised monocular depth hints," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 2162–2171.

[49] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.

[50] F. Milletari, N. Navab, and S.-A. Ahmadi, "V-Net: Fully convolutional neural networks for volumetric medical image segmentation," in *Proc. 4th Int. Conf. 3D Vis. (3DV)*, Oct. 2016, pp. 565–571.

[51] Z. Zhang, X. Zhang, C. Peng, X. Xue, and J. Sun, "Exfuse: Enhancing feature fusion for semantic segmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 269–284.

[52] Q. Zhang, C. Zhou, Y.-C. Tian, N. Xiong, Y. Qin, and B. Hu, "A fuzzy probability Bayesian network approach for dynamic cybersecurity risk assessment in industrial control systems," *IEEE Trans. Ind. Informat.*, vol. 14, no. 6, pp. 2497–2506, Jun. 2018.

[53] W. Fang, X. Yao, X. Zhao, J. Yin, and N. Xiong, "A stochastic control approach to maximize profit on service provisioning for mobile cloudlet platforms," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 48, no. 4, pp. 522–534, Apr. 2018.

[54] Y. Qu and N. Xiong, "RFH: A resilient, fault-tolerant and high-efficient replication algorithm for distributed cloud storage," in *Proc. 41st Int. Conf. Parallel Process.*, Sep. 2012, pp. 520–529.

[55] M. Wu, L. Tan, and N. Xiong, "A structure fidelity approach for big data collection in wireless sensor networks," *Sensors*, vol. 15, no. 1, pp. 248–273, Jan. 2015.

[56] H. Li, J. Liu, R. W. Liu, N. Xiong, K. Wu, and T.-H. Kim, "A dimensionality reduction-based multi-step clustering method for robust vessel trajectory analysis," *Sensors*, vol. 17, no. 8, p. 1792, 2017.

[57] Z. Zhou, X. Chen, Y. Zhang, and S. Mumtaz, "Blockchain-empowered secure spectrum sharing for 5G heterogeneous networks," *IEEE Netw.*, vol. 34, no. 1, pp. 24–31, Jan./Feb. 2020.

[58] Z. Zhou *et al.*, "Energy-efficient resource allocation for energy harvesting-based cognitive machine-to-machine communications," *IEEE Trans. Cognit. Commun. Netw.*, vol. 5, no. 3, pp. 595–607, Sep. 2019.

[59] X. Li *et al.*, "Hardware impaired ambient backscatter NOMA systems: Reliability and security," *IEEE Trans. Commun.*, vol. 69, no. 4, pp. 2723–2736, Apr. 2021.

[60] S. Song, S. P. Lichtenberg, and J. Xiao, "SUN RGB-D: A RGB-D scene understanding benchmark suite," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 567–576.

[61] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from RGBD images," in *Proc. Eur. Conf. Comput. Vis.*, Cham, Switzerland: Springer, 2012, pp. 746–760.

[62] A. Janoch *et al.*, "A category-level 3-D object dataset: Putting the Kinect to work," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCV Workshops)*, Nov. 2011, pp. 141–165.

[63] J. Xiao, A. Owens, and A. Torralba, "SUN3D: A database of big spaces reconstructed using SfM and object labels," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 1625–1632.

[64] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of RGB-D SLAM systems," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2012, pp. 573–580.

[65] A. Handa, T. Whelan, J. McDonald, and A. J. Davison, "A benchmark for RGB-D visual odometry, 3D reconstruction and SLAM," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2014, pp. 1524–1531.

[66] Y. Kuznietsov, J. Stuckler, and B. Leibe, "Semi-supervised deep learning for monocular depth map prediction," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6647–6655.

[67] A. Pilzer, S. Lathuiliere, N. Sebe, and E. Ricci, "Refine and distill: Exploiting cycle-inconsistency and knowledge distillation for unsupervised monocular depth estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9768–9777.
[68] C. Godard, O. M. Aodha, M. Firman, and G. Brostow, "Digging into self-supervised monocular depth estimation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 3828–3838.
[69] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardós, "ORB-SLAM: A versatile and accurate monocular SLAM system," *IEEE Trans. Robot.*, vol. 31, no. 5, pp. 1147–1163, Oct. 2015.
[70] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe, "Unsupervised learning of depth and ego-motion from video," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1851–1858.
[71] J. Engel, T. Schöps, and D. Cremers, "LSD-SLAM: Large-scale direct monocular SLAM," in *Proc. Eur. Conf. Comput. Vis.*, Cham, Switzerland: Springer, 2014, pp. 834–849.
[72] J. Engel, V. Koltun, and D. Cremers, "Direct sparse odometry," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 3, pp. 611–625, Mar. 2016.

**Yongbin Gao** received the Ph.D. degree from Jeonbuk National University, South Korea. He is currently an Associate Professor with the School of Electronic and Electrical Engineering and the Vice Director of the Next Generation Intelligent Research Center, Shanghai University of Engineering Science, Shanghai, China. He has published 30 SCI articles in prestigious journals, such as IEEE TRANSACTIONS ON IMAGE PROCESSING, IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, *Information Science*, and *Pattern Recognition Letters*, in the area of image processing, patter recognition, and computer vision.

**Fangzheng Tian** is currently pursuing the master's degree with the School of Electronic and Electrical Engineering, Shanghai University of Engineering Science, Shanghai, China. His research interests include 3D reconstruction, computer vision, and machine learning.

**Jun Li** (Member, IEEE) received the B.S. degree from the South Central University for Nationalities, Wuhan, China, in 2009, and the Ph.D. degree from Chonbuk National University, Jeonju, South Korea, in 2016. He is currently an Associate Professor with Guangzhou University, Guangzhou, China. He has published more than 50 papers in refereed journals and conference proceedings. His research interests include spatial modulation and OFDM with index modulation. He serves as a Reviewer for IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, IEEE TRANSACTIONS ON COMMUNICATIONS, IEEE JOURNAL OF SELECTED TOPICS IN SIGNAL PROCESSING, IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, and IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY.

**Zhijun Fang** (Senior Member, IEEE) received the Ph.D. degree from Shanghai Jiao Tong University, Shanghai, China. He is currently a Professor and the Dean with the School of Electronic and Electrical Engineering, Shanghai University of Engineering Science. His current research interests include image processing, video coding, and pattern recognition. He was the General Chair of the Joint Conference on Harmonious Human Machine Environment (HHME) 2013 and the General Co-Chair of the International Symposium on Information Technology Convergence (ISITC) in 2014, 2015, 2016, and 2017. He received the "Hundred, Thousand and Ten Thousand Talents Project" in China. He received several major program projects of the National Natural Science Foundation of China and the National Key Research and Development Project of the Ministry of Science and Technology of China.

**Saba Al-Rubaye** (Senior Member, IEEE) received the Ph.D. degree in electronic and computer engineering from Brunel University London, U.K., in 2013. She is currently a Senior Lecturer with the Connected System Research Group, School of Aerospace, Transport and Manufacturing, Cranfield University, U.K. She worked in industry, where she participated in developing loop testbed in control and communications at the Quanta's Sustainable Technology Integration Laboratory (QT-STIL), Toronto, ON, Canada. She has published many papers in IEEE journals and conferences. She has been involved in several projects sponsored by the Natural Sciences and Engineering Research Council of Canada (NSERC), the Engineering and Physical Sciences Research Council (EPSRC) for drone connectivity, the Department for Transport (DfT) funded projects for airport security, and H2020. Her main research interests include airport connectivity, UAV, 5G and beyond, communications networks, and autonomous systems. She is a Voting Member of the IEEE P1920.2 Standard for Vehicle-to-Vehicle Communications for Unmanned Aircraft Systems and the IEEE P1932.1 Standard of License/Unlicensed Interoperability. She was a recipient of the Best Technical Paper Award twice published in the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY in 2011 and 2015. She has served as the general co-chair and a member of the technical program committee for many international conferences. She is registered as a Chartered Engineer (C.Eng.) by the British Engineering Council, an Associate Fellow of the British Higher Education Academy (AFHEA), and a Certified UAV Pilot.

**Wei Song** received the M.S. degree in soft engineering from the Dalian University of Technology, China, and the Ph.D. degree from Chonbuk National University, Jeonju, South Korea, in 2010. He is currently working with the Applied Technology College of Soochow University, Suzhou, China, as a Distinguished Professor. His research interests include spatial modulation, MIMO, STBC, and reconfigurable intelligent surface.

**Yier Yan** received the B.S. degree in applied electronics from the South Central University for Nationalities, Wuhan, Hubei, China, and the M.S. and Ph.D. degrees in communication engineering from Chonbuk National University, Jeonju, South Korea. He is currently working with the School of Electronics and Communication Engineering, Guangzhou University, China. His research interests include information theory, signal processing, and OFDM systems.