remote sensing

MDPI

Article

Multi-Sensor Fusion Self-Supervised Deep Odometry and Depth Estimation

Yingcai Wan [1], Qiankun Zhao [1], Cheng Guo [1], Chenlong Xu [2] and Lijing Fang [1,*]

[1] Faculty of Robot Science and Engineering, Northeastern University, Shenyang 110819, China; 1710333@std.neu.edu.cn (Y.W.); 2110701@std.neu.edu.cn (Q.Z.); 1901941@std.neu.edu.cn (C.G.)
[2] College of Intelligent Systems Science and Engineering, Harbin Engineering University, Harbin 150001, China; xurobot@hrbeu.edu.cn
* Correspondence: ljfang@mail.neu.edu.cn; Tel.: +86-138-4019-2905

Abstract: This paper presents a new deep visual-inertial odometry and depth estimation framework for improving the accuracy of depth estimation and ego-motion from image sequences and inertial measurement unit (IMU) raw data. The proposed framework predicts ego-motion and depth with absolute scale in a self-supervised manner. We first capture dense features and solve the pose by deep visual odometry (DVO), and then combine the pose estimation pipeline with deep inertial odometry (DIO) by the extended Kalman filter (EKF) method to produce the sparse depth and pose with absolute scale. We then join deep visual-inertial odometry (DeepVIO) with depth estimation by using sparse depth and the pose from DeepVIO pipeline to align the scale of the depth prediction with the triangulated point cloud and reduce image reconstruction error. Specifically, we use the strengths of learning-based visual-inertial odometry (VIO) and depth estimation to build an end-to-end self-supervised learning architecture. We evaluated the new framework on the KITTI datasets and compared it to the previous techniques. We show that our approach improves results for ego-motion estimation and achieves comparable results for depth estimation, especially in the detail area.

Keywords: self-supervised; autonomous driving; depth estimation; visual-inertial odometry

Citation: Wan, Y.; Zhao, Q.; Guo, C.; Xu, C.; Fang, L. Multi-Sensor Fusion Self-Supervised Deep Odometry and Depth Estimation. *Remote Sens.* **2022**, *14*, 1228. https://doi.org/10.3390/rs14051228

Academic Editors: Yangquan Chen, Subhas Mukhopadhyay, Nunzio Cennamo, M. Jamal Deen, Junseop Lee and Simone Morais

Received: 16 December 2021
Accepted: 14 February 2022
Published: 2 March 2022

1. Introduction

Dense depth estimation from an RGB image is the fundamental issue for 3D scene reconstruction that is useful for computer vision applications, such as automatic driving [1], simultaneous localization and mapping (SLAM) [2], and 3D scene understanding [3]. With rapid development of in depth estimation (from monocular), many supervised and unsupervised learning methods have been proposed. Instead of traditional supervised methods depending on expensively collected ground truth, unsupervised learning from stereo images or monocular videos is a more universal solution [4,5]. However, due to the lack of perfect ground truth and geometric constraints, unsupervised depth estimation methods that suffer from inherent scale ambiguity and poor performance, perform well in some scenarios, such as occlusion, non-textured regions, dynamic motion objects, and indoor environment.

To overcome the lack of geometric constraints in unsupervised depth estimation training, recent works have used sparse LiDAR data [6–8] to guide depth estimation in the process of image feature extraction and improve the quality of supervised depth map generation. These methods lead to the dependence on sparse LiDAR data, which are relatively expensive. A recent trend in depth estimation methods involves traditional SLAM [9], which could provide an accurate sparse point cloud, learning to predict monocular depth and odometry in a self-supervised manner [10,11].

To integrate visual odometry (VO) or the SLAM system into depth estimation, the authors of [10,12,13] presented a neural network to correct classical VO estimators in a self-supervised manner and enhance geometric constraints. Self-supervised depth estimation,

using the pose and depth between two adjacent frames, establishes a depth reprojection error and image reconstruction error [14–17]. In a monocular depth self-supervised estimation, the depth value estimated by the depth estimation network (DepthNet) and the pose between adjacent images have a decisive influence on the depth estimation result. However, the depth estimation network and the pose estimation network (PoseNet) can only estimate the relative results without the correction of the geometry constraint. As the relative pose estimation is inherently ambiguously scaled, the pose prediction network [18] significantly degrades when applied to challenging scenarios.

Motivated by these observations, we present a new deep visual-inertial odometry (DeepVIO) based ego-motion and depth prediction system that combines the strengths of learning-based VIO and geometrical depth estimation [16,19,20]. It uses DeepVIO geometrical constraints [21], where they are available, to achieve accurate odometry fusing with raw inertial measurement unit (IMU) data and sparse point clouds. To get a sparse point depth, we selected the associated feature points, which extracted and matched between the two adjacent frames, and then solved the triangulation equation to achieve the feature point depth, such as in Figure 1 [22]. Technically, this was implemented using learning-based feature detectors and an IMU raw fusion module, so that it learnt to refine the scale of depth and improve 3D geometric constraints. In addition, the learning process of DeepVIO was supervised in a self-supervised manner during depth estimation, where the depth and DeepVIO could benefit from each other.

The overview of our method is shown in Figure 2; different from other self-supervised depth estimation methods, our method generates a dense depth with the depth value in the 3D structure corresponding to each pixel in the 2D image and accurate pose with DeepVIO, which combines deep visual odometry (DVO) with deep inertial odometry (DIO). In order to improve the pose estimation between adjacent image frames, we introduce the key-point method based on the deep learning visual process, which uses a neural network to complete the pose estimation method of feature point extraction and matching in traditional SLAM, and can be easily combined with an IMU network. First, we used the deep keypoint-based [23,24] feature extraction and matching method, such as in Figure 1c,d, and then obtained the relative pose from input sequence frames by the traditional two-view geometry triangulation method [24]. Second, we used the fusion module to integrate the DVO and DIO and output a relative pose with an absolute scale [21]. Finally, we obtained the sparse depth map, in which the depth value corresponded with keypoints by matching the depth feature points, using the two-view triangulation module to solve the depth of the feature points with the camera pose [25]. We refined the scale of the prediction depth and constraint depth network regression by the sparse depth in the training and testing stages. In summary, the main contributions of our work are as follows.

We propose a new self-supervised depth and odometry estimation framework that combines DepthNet with DeepVIO to supervise each other:

- Based on the SuperPoint [23] dense feature point extraction method, we added the sparse depth pose with absolute scale to the depth estimation geometric constraints;
- The DeepVIO pipeline joint keypoint is based on DVO with DIO and uses the EKF module to update the relative pose;
- We tested our framework on the KITTI dataset, showing that our approach produces more accurate absolute depth maps than contemporaneous methods. Our model also demonstrates stronger generalization capabilities and robustness across datasets.

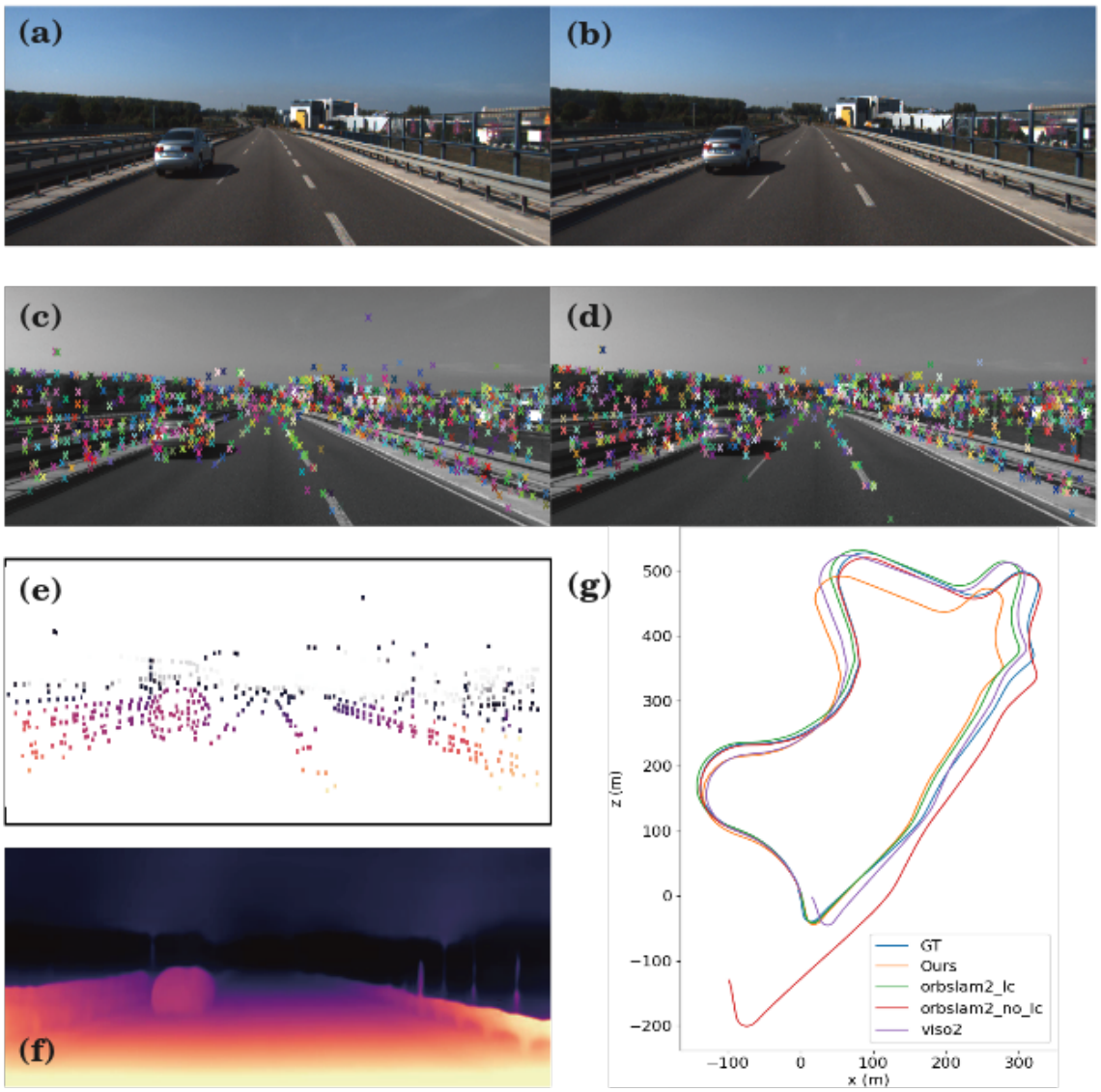

Figure 1. Depth and pose estimation results from KITTI; (**a**,**b**) show the input image to the network selected from KITTI; (**c**,**d**) show the feature dense match using the DVO pipeline; (**e**) shows the sparse depth map solved by the DeepVIO; (**f**) shows dense depth map from DepthNet, which is refined by the sparse depth map with the absolution scale; (**g**) shows the DeepVIO odometry result on sequence 09.

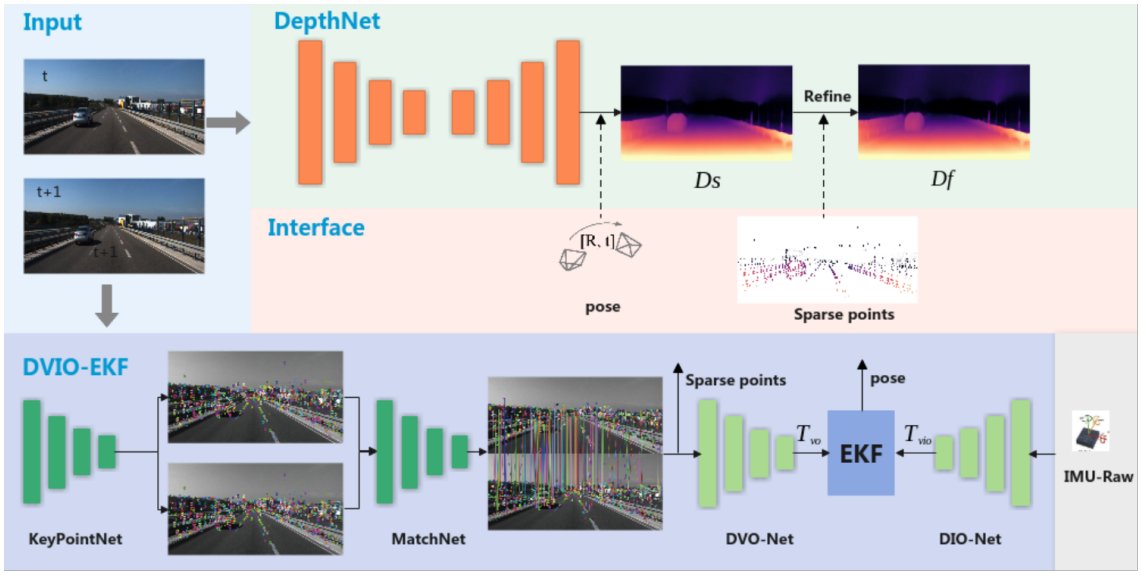

Figure 2. Overview of the system. At time t and $t+1$, the system inputs two image pairs and IMU data, and outputs pose and optimized depth D_s from original dept D_f.The bottom DeepVIO pipeline extracts and matches dense feature points and solves the original pose by the geometric method. The fusion module combines the DVO and DIO pose with the EKF method. The sparse depth is generated by feature matching points and the pose of DeepVIO. The top depth estimation pipeline joins the sparse depth and pose, supervises the depth and pose, to produce a dense image with absolute scale.

2. Related Work

In this section, we provide an overview of current methods for self-supervised depth estimation and techniques for learning-based feature extraction and match.

2.1. Self-Supervised Monocular Depth Prediction

Depth estimation from a monocular image is significant for scene understanding in computer vision. Supervised learning-based methods for depth prediction rely upon the availability of ground-truth depth [26–28], while the effort to collect large amounts of labeled images is high. In the self-supervised depth estimation method, the photometric errors originate from static stereo warping with the rectified baseline or two adjacent frame temporal warping. Based on that theory, a lot of research in the field of supervised depth estimation has been conducted to overcome the need for ground truth data [29], self-supervised learning methods [30], minimizing photometric reprojection errors, and using the binary mask to filter dynamic objects in videos [30,31]. However, these methods lack geometric constraints and scale ambiguity in the learning process. Recently, a combination with geometric constraint depth estimation methods have been proposed [10,11,32–34]. For example, the average depth varies greatly between adjacent frames when there are limited image pixel movement ranges, relative estimated poses, and inconsistent reference frames between the poses [15]. Since visual odometry based on monocular image sequences could only estimate the relative poses, constraining monocular depth estimation will result in inconsistent depth map scales. We introduce an IMU with an absolute scale to form VIO, which essentially eliminates the problem of inconsistency in depth scales between adjacent frames. Thus, our work combines the advantages of the visual odometry method based on deep keypoints and raw IMU data, and essentially disentangles the scale and enhances geometry construction.

2.2. Learning-Based Feature Extraction and Matching

Traditional feature detectors and descriptors have been used on classical SLAM systems. Based on classical handcraft feature detectors and feature extractor-like features from

an accelerated segment test (FAST) [35], oriented fast and rotated brief (ORB) [36], and scale-invariant feature transform (SIFT) [37], these methods attempt to dedicate to dimensionality reduction and utilize various approaches to map high-dimensional descriptors to low-dimensional spaces. However, they lose a great amount of information on the raw image. With deep learning "booming", some researchers have attempted to use higher-level features obtained through deep learning models to build-up deep feature extractors. CNN-based descriptors, such as MatchNet [38], which consists of a featured network for extracting feature representation, significantly improves feature descriptor results.

However, most deep learning methods rely heavily on data used for training and cannot fit well into unknown environments. Instead of using human supervision to define interest points in real images [39], SuperPoint [23] proposes a fully-convolutional neural network architecture for interest point detection and descriptions using a self-supervised pipeline. Our work adopts the deep feature descriptor detector, feature extractor, and VIO pipeline as our foundation to improve the pose and depth estimation result.

2.3. Deep Visual-Inertial Odometry Learning Methods

Traditional VIO fusion relies on manually crafted image processing pipelines, which can be divided into loosely-coupled and tightly-coupled methods [40]. Recently, deep learning methods [41] have been used to state estimation tasks, including VIO. Instead of using human supervision to define interest points in real images, such as FAST [35], SIFT [37], Daniel DeTone [42], designed SuperPoint, which operates on a full-sized image and produces interest point detections accompanied by fixed-length descriptors in a single forward pass.

For supervised learning VO methods, these approaches infer the camera pose by learning directly from real image data, such as Flowdometry [43], cast the VO problem as a regression problem by using FlowNet [44] to extract optical flow features and a fully connected layer to predict camera translation and rotation, and DVO [13] and ESP-VO [45] incorporate recurrent neural networks (RNNs), to implicitly model the sequential motion dynamics of the image sequences. Han, L. et al. [13] presented a self-supervised deep learning network for monocular VIO; Shamwell et al. [46] presented an unsupervised deep neural network approach to the fusion of RGB-D imagery with inertial measurements for absolute trajectory estimation. Inspired by this work, we incorporated raw IMU data into a visual, odometry-based deep keypoint with a fusion model to regularize the camera pose and alignment depth map.

3. Materials and Methods

We propose a framework to predict the dense depth and odometry with an absolute scale only using the monocular images and IMU raw data. Figure 2 depicts an overview of our system; we used DeepVIO to replace the PoseNet. The SuperPoint [23] network has two sub networks—KeypointNet and MatchNet—to estimate VO. After that, the DVO and DIO fusion module are used to estimate the odometry of the camera and sparse depth based on the 3D points triangulated. Then DepthNet combines sparse depth and pose to output the depth map with absolute scale. Specifically, we propose DeepVIO, self-supervised, by the depth estimation process.

3.1. Self-Supervised Depth Estimation

The depth module is an encoder–decoder network, DepthNet; it takes a target image and outputs depth values $\hat{D}_T(p)$ for every pixel p in the image.The encoder of DepthNet uses ResNet to extract the features of the input images with four scale layers, while the skip connections fuse the encoder layer features with the decoder upsample convolution network, and the decoder finally outputs a depth map corresponding to the pixels of the input image. The pose module (PoseNet) uses ResNet [47] to extract image features, and then the decoder adopts convolution layer regression, six parameters of $[R, t]$. PoseNet take, as input, the concatenation of the target image I_t and two neighbor (source) images I_S,

$S \in \{t-1, t+1\}$. It outputs transformation matrices, $\hat{T}_{T\to S}$ represent the six degrees of freedom (6DoF) relative poses between the images. Self-supervised learning is proceeded by image reconstruction using the inverse warping technique. The inputs of the training sample include the target frames I_T at t and the source frames I_S at the nearby frame $I_S \in \{I_{t-1}, I_{t+1}\}$ [31]. The self-supervised training uses the source images I_S to synthesize the target image I_T. When the depth, together with the pose, is provided, the source image can synthesize a new view (target) by applying a projective warping from the source camera point of view. The sampling is done by projecting the homogeneous coordinates of the target pixel p_t onto the source view p_s [30]. Given the camera intrinsics K, the encoder–decoder network DepthNet estimated depth of $\hat{D}_T(p)$ and the pose module-predicted transformation matrix $\hat{T}_{T\to S}$, the projection is done by the equation:

$$xp_S \sim K\hat{T}_{T\to S}\hat{D}_T(p_t)K^{-1}p_t. \tag{1}$$

we adopt the popular combination of the least absolute deviation loss (L1 loss) and structural similarity index (SSIM) by [4] computing the photometric errors,

$$L_{pe} = \sum_S \sum_p |I_S(p_t) - \hat{I}_S(p_t)|. \tag{2}$$

where $\hat{I}_S(p_t)$ is the intensity value of p_t in the reconstructed image $\hat{I}_S$, p represents the pixel in the image, and S represents the source image. We use the edge-aware depth smoothness loss, which uses the image gradient to weigh the depth gradient [30]:

$$L_{ds} = |\partial_x d_t^*|e^{-|\partial_x I_T|} + |\partial_y d_t^*|e^{-|\partial_y I_T|} \tag{3}$$

3.2. Deep Visual Odometry Based on Keypoint

We chose SuperPoint [23] as our DVO network backbone instead of traditional feature extractors, e.g., ORB [9,36], SIFT [37]. SuperPoint is a learning-based feature extraction method that has a shared encoder with two decoders, similar to the traditional feature extraction method SIFT, and has both feature point detection and description functions. The encoder is based on VGG network architecture and consists of convolutional layers, spatial downsampling via pooling, non-linear activation functions, and rectified linear unit (ReLU). After the encoder, the architecture splits into two decoder "heads", which learn task-specific weights for interest point detection and interest point description. When the feature points of two adjacent frames are obtained from KeypointNet, we associate the feature points of the two frames through MatchNet. Taking advantage of geometric constraints of 3D structures from sequence frames, we join estimate depth and pose in a self-supervised manner using photometric consistency, we get correspondences from matched deep features by using a deep detector and descriptor and recover the camera pose via traditional geometry methods. Specifically, the correspondences located in occluded or out-of-bounds dynamics regions, are masked out to improve the accuracy of 2D–2D correspondences.

We refer to the image pair I_i and I_j as the input of feature extractions, the transformation matrix from I_i to I_j as $T_{ij} = [R, t]$, where $R \in R^{3\times3}$ is the rotation matrix and $t \in R^{3\times1}$ is the translation vector. The DVO network includes a shared encoder and detector and descriptor heads as the detector and descriptor, respectively. It extracts features from input images $I_i, I_j \in R^{H\times W\times1}$, an output detector feature $H_{det} \in R^{H\times W\times1}$, and descriptor feature $H_{desc} \in R^{H\times W\times D}$. Then H_{det} applies non-maximum suppression to get sparse keypoints. Moreover, the descriptor is sampled from H_{desc} using bilinear interpolation, which filter out redundant candidates by non-maximum suppression.

Deep Pose Estimation Decode

Typically, the traditional visual odometry pose$[R, t]$ estimation method includes epipolar geometry-based and PnP-based. When the 2D–2D pixel correspondences (p_i, p_j) between the image pair builds, we can use the epipolar geometry-based method to solve

the fundamental matrix F via the simple normalized 8-point algorithm in random sample consensus (RANSAC) loop [47]. The epipolar geometry solves **F**:

$$p_j^T \mathbf{F} p_i = \mathbf{0}, \mathbf{F} = \mathrm{K}^{-T}[\mathrm{t}]_{\times}\mathrm{R}\mathrm{K}^{-1} \tag{4}$$

where the correspondences (p_i, p_j) are formed from SuperPoint, **F** is the fundamental matrix, K is the camera intrinsics. However, in some cases, the fundamental matrix will fail to solve. Perspective-n-Point (PnP) is used to solve camera pose given 3D–2D correspondences when the camera motion is pure rotation or the camera translation, tinily. PnP minimizes the reprojection error:

$$e = \sum_i \| (\mathrm{R}\mathrm{X}_{1,i} + \mathrm{t}) - p(2,i) \|_2 \tag{5}$$

Epipolar and PnP methods need constant judgments, a switch in motion process, and difficult-to-solve complex motions, which are not robust and are not accurate. Therefore, we use the network to replace the geometric solution method and fuse the network prediction with IMU poses in the training strategy.

The MatchNet outputs match N feature points. We feed the points $[6 \times N]$ in which the correspondences (p_i, p_j) are formed from SuperPoint detection, matching into the one-dimensional CNN network, then process them through long short-term memory (LSTM) layers with 128 and 256 cells and a fully connected (FC) layer. The output layer contains two linear layers to produce the prediction of rotation and translation $\mathbb{SE}(3)_{dvo}$.

3.3. DeepVIO Fusion Module

As aforementioned, we can resolve the inherently scaled ambiguity, and DVO significantly degrades in some scenarios by fusing DVO with IMU data. Different from the previous learning-based method, which directly feeds the IMU and images into the network to predict the pose or use the IMU as the L1 loss of the DVO output, we designed a monocular DeepVIO that combined the DVO with DIO by using EKF to predict and update the pose state. We first define the IMU model at time τ, the measured accelerometer values $\mathbf{a}_m$, gyroscope values $\mathbf{w}_m$, and the robot state $\mathbf{S}_{\mathbf{v_ø}}$ at time τ.

The accelerometer and gyroscope random noise $\mathbf{n}_a$, $\mathbf{n}_w$, $\dot{\mathbf{b}}_{a_ø} = \mathbf{n}_{b_a}$, $\dot{\mathbf{b}}_{a_ø} = \mathbf{n}_{b_a}$, $\dot{\mathbf{b}}_{w_ø} = \mathbf{n}_{b_w}$ are assumed to mean Gaussian $\mathbf{n}_a \sim \mathcal{N}(\mathbf{0}, \sigma_a^2 \mathbf{I})$:

$$\begin{aligned} \mathbf{a}_m &= \mathbf{a}_{v_\tau}^{v_\tau i} + \mathbf{C}_{v_\tau r_k} \mathbf{g}_{r_k} + \mathbf{b}_{a_\tau} + \mathbf{n}_a \\ \boldsymbol{\omega}_m &= \boldsymbol{\omega}_{v_\tau}^{v_\tau i} + \mathbf{b}_{\omega_\tau} + \mathbf{n}_\omega \end{aligned} \tag{6}$$

where $\mathbf{C}_{v_ø r_k}$ is the robot states, $\mathbf{g}_{rk}$ is the gravity vector. Moreover, the robot states are defined:

$$\begin{aligned} \dot{\mathbf{C}}_{r_k v_\tau} &= \mathbf{C}_{r_k v_\tau} \left[\boldsymbol{\omega}_{v_\tau}^{v_\tau i} \right]^\wedge \\ \dot{\mathbf{r}}_{r_k}^{v_\tau r_k} &= \mathbf{C}_{r_k v_\tau} \mathbf{v}_{v_\tau}^{v_\tau i} \\ \dot{\mathbf{v}}_{v_\tau}^{v_\tau i} &= \mathbf{a}_{v_\tau}^{v_\tau i} - \left[\boldsymbol{\omega}_{v_\tau}^{v_\tau} \right]^\wedge \mathbf{v}_{v_\tau}^{v_\tau i} \\ \dot{\mathbf{b}}_{\omega_\tau} &= \mathbf{n}_{b_\omega} \\ \dot{\mathbf{b}}_{a_\tau} &= \mathbf{n}_{b_a} \end{aligned} \tag{7}$$

We get the linearized system from (6) and (7). The system matrix is defined **F**, the linearized error matrix **G**, and the noise $\mathbf{n} = [\mathbf{n}_w^T \ \mathbf{n}_{b_w}^T \ \mathbf{n}_a^T \ \mathbf{n}_{b_a}^T]$. To solve the error states, $\delta \dot{\mathbf{x}}_\tau$ is used to propagate error state covariances:

$$\delta \dot{\mathbf{x}}_\tau = \mathbf{F}\mathbf{f}\mathbf{f}\mathbf{i} x_ø + \mathbf{G}\mathbf{n} \tag{8}$$

We apply Euler's method transform continuous model (8) to discrete time. From time t_τ to $t_{\tau+1}$ $\delta t = t_{\tau+1} - t_\tau$, the sate transition matrix $\Phi_{\tau,\tau+1}$ use the order approximation:

$$\Phi_{\tau,\tau+1} = exp\left(\int_{t_\tau}^{t_{\tau+1}} F(s)dt)\right) \approx \mathbf{I} + \mathbf{F_{\text{ø}}ffi}t \tag{9}$$

Then the IMU measurement propagates state covariance $\mathbf{Q}$ to the next step state covariance:

$$Q = diag(\sigma_w^2 I, \sigma_{b_w}^2, I, \sigma_a^2 I, \sigma_{b_a}^2 I) \tag{10}$$

$$\check{\mathbf{P}}_{\tau+1} = \Phi_{\tau,\tau+1}\check{\mathbf{P}}_\tau\Phi_{\tau,\tau+1}^T + \mathbf{GQG}^T\mathbf{ffi_t} \tag{11}$$

3.3.1. DIO-Net Measurement Model

In this paper, we propose a DIO-Net deep inertial odometry network to replace the inertial odometry data process, preintegration, and pose prediction. The DIO-Net architecture is illustrated in Figure 3. To obtain the IMU data that have space features, we associate CNN and LSTM in the deep inertial odometry. The model is comprised of two CNNs to extract the deep feature firstly; two LSTM layers, preintegrated features, and two linear layers produce the final odometry prediction. Furthermore, the IMU data enters 32×32 , 64×64, 128×128 CNNs for feature extraction, and then the features enter the LSTM layers after ReLU, and finally FC outputs the six parameters of pose. In this process, the CNN transforms the input feature to a 128-channel feature, then LSTM processes the last layer feature and outputs 256 channels feature, the FC regresses the 3D rotation, and the 3D translation presents as $\mathbb{SE}(3)_{dio}$.

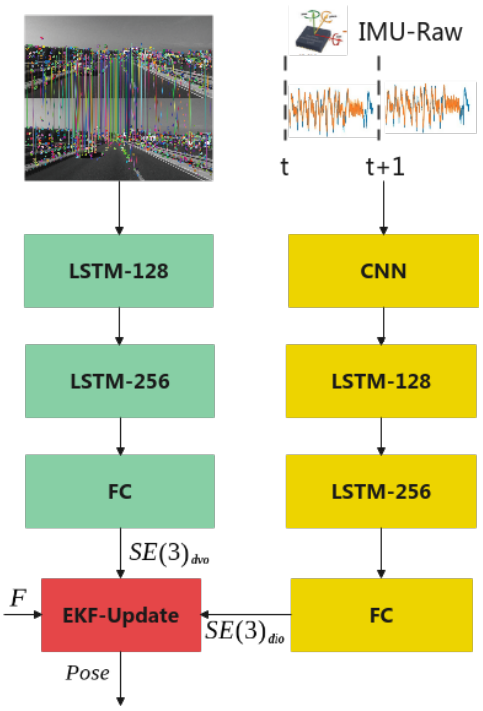

Figure 3. Pose fusion module based on EKF. On the (**left**), the DVO-Net inputs the matching points of the matching network, encodes the matching points through the LSTM network, and outputs $\mathbb{SE}(3)_{dvo}$ through the FC as the measurement value of EKF. On the (**right**), the DIO-Net inputs the original IMU data and extracts the features through CNN, and then outputs $\mathbb{SE}(3)_{dio}$ as the initial value of EKF through the 128-channel LSTM network and the FC network. The EKF module integrates DVO and DIO modules, and finally outputs the pose.

3.3.2. DIO and DVO EKF Fusion Model

Our EKF model is robot-centric-based, and the EKF propagates its state based on the kinematics theory, maintaining the system's differentiability. It incorporates vision and IMU relative measurements learned from deep networks in its update step, as well as uncertainty. With new image and IMU data input, the result of DIO prediction is set to the observations of EKF and the pose of DVO prediction is set to a robot-centric state. EKF fuses the DVO and DVI results, performs the EKF operation to obtain the fused system state, and finally moves the time stamp from k to k + 1, and repeats the above steps. We express the DVO estimated pose $\mathbb{SE}(3)_{dvo}$ as: $\widetilde{z} = \begin{bmatrix} \widetilde{\phi}^T_{r_k v_{k+1}} & \widetilde{r}^{v_{k+1} r_k \mathrm{T}}_{r_k} & \mathrm{w}^{\mathrm{T}}_{\mathrm{\ddot{E}}} & \mathrm{w}^{\mathrm{T}}_{\mathrm{r}} \end{bmatrix}^T$ The covariance R_k can be represented as a diagonal matrix. The measurement residual $\epsilon_{k+1} = \begin{bmatrix} \epsilon^T_\theta & \epsilon^T_r \end{bmatrix}^T$:

$$\begin{bmatrix} \epsilon_\theta \\ \epsilon_r \end{bmatrix} = \begin{bmatrix} \ln(\exp(\widetilde{\phi}^\vee_{r_k v_{k+}}) C^T_{r_k v_{k+1}}) \\ \widetilde{r}^{v_{k+1} r_k}_{r_k} - r^{v_{k+1} r_k}_{r_k} \end{bmatrix} \tag{12}$$

where $(\cdot)^\wedge$ is skew symmetric operator.

In the training process, to make the network residual differentiable from the measurement residual, we approximate the network output residual to $\epsilon_\theta = \widetilde{\phi}_{r_k v_{k+1}} - \phi_{r_k v_{k+1}}$ by using Baker–Campbell–Hausdorff (BCH). The error states could find the DVO Jacobin $H_{k+1} = \frac{\partial \epsilon_{k+1}}{\partial \delta \mathrm{x}_{k+1}}$, and the ϵ_θ and ϵ_r are represented as:

$$\begin{bmatrix} \epsilon_\theta \\ \epsilon_r \end{bmatrix} \begin{bmatrix} \widetilde{\phi}_{r_k v_{k+1}} \hat{\phi}_{r_k v_{k+1}} + J_r(\phi_{r_k v_{k+1}})^{-1} \delta \phi_{r_k v_{k+1}} \\ \widetilde{r}^{v_{k+1} r_k}_{r_k} + \hat{r}^{v_{k+1} r_k}_{r_k} - \delta r^{v_{k+1} r_k}_{r_k} \end{bmatrix} \tag{13}$$

The final DVO Jacobin H_{k+1} is shown in (14).

$$H_{k+1} = \begin{bmatrix} \mathbf{0}_{9\times 0} & -\mathrm{J}(-\ddot{\mathrm{E}}_{\mathrm{r_k vk+1}}) & \mathbf{0} & \mathbf{0}_{9\times 0} \\ \mathbf{0}_{9\times 0} & \mathbf{0} & -\mathbf{I} & \mathbf{0}_{9\times 0} \end{bmatrix} \tag{14}$$

The EKF estimation update and error $\delta \hat{\mathbf{x}}_{k+1}$ are shown in (12)(13)(14); the calculation of the Kalman Gain:

$$\mathbf{K}_{\mathrm{k+1}} = \check{\mathbf{P}}_{\mathrm{k+1}} \mathbf{H}^{\mathrm{T}}_{\mathrm{k+1}} (\mathbf{H}_{\mathrm{k+1}} \check{\mathbf{P}}_{\mathrm{k+1}} \mathbf{H}^{\mathrm{T}}_{\mathrm{k+1}} + \mathbf{R}_{\mathrm{k+1}})^{-1} \tag{15}$$

The calculation of the posterior state and covariance:

$$\hat{\mathbf{P}}_{k+1} = (\mathbf{I} - \mathbf{K}_{k+1} \mathbf{H}_{k+1}) \check{\mathbf{P}}_{k+1} \tag{16}$$

The error $\delta \hat{\mathbf{x}}_{k+1}$:

$$\delta \hat{\mathbf{x}}_{k+1} = \mathbf{K}_{k+1} \overline{\boldsymbol{\epsilon}}_{k+1} \tag{17}$$

where $\mathbf{H}_{k+1}$ is the measurement Jacobian, $\mathbf{R}_k$ is corresponding covariances, $\overline{\boldsymbol{\epsilon}}_{k+1}$ is the measurement residual.

Finally, the reference frame for all states is shifted forward from frame $\mathbf{S}_{rk}$ to frame $\mathbf{S}_{rk+1}$, and the robot pose transforms to the next EKF iteration after it is composed with the DVO pose. The VIO fusion final output:

$$\hat{\mathbf{P}}_{k+1,r_{k+1}} = \mathbf{U}_{k+1} \hat{\mathbf{P}}_{k+1} \mathbf{U}^T_{k+1}, \mathbf{U}_{k+1} = \frac{\partial \delta \hat{\mathbf{x}}_{k+1,r_{k+1}}}{\partial \delta \hat{\mathbf{x}}_{k+1}} \tag{18}$$

Using the output of the EKF fusion module, the DVO updates $\hat{T}_{vo} = [R, t] \in \mathbb{SE}(3)_{dvo}$ and regresses the network, with the rotation component $R \in \mathbb{SO}(3)$, and the translation component $t \in \mathbb{R}^3$. The pose loss $L_{rt} = |\hat{T}_{dvo} - T_{dio}|$ is defined for all pairs of relative pose transformations; it contains the rotation loss L_{rot} and translation loss L_{trans}:

$$L_{rot} = min(\|R_{dvo} - R_{dio}\|_2)$$

$$L_{trans} = min(\|t_{dvo} - t_{dio}\|_2) \tag{19}$$

The pose total loss is defined:

$$\begin{aligned} L_{rt} = min([L_{rot}(R_{dvo}, R_{dio}) + c_r] + \\ \beta_{rt}[L_{trans}(t_{dvo}, t_{dio}) + c_t)]) \end{aligned} \tag{20}$$

3.4. Supervised with Sparse Depth from DeepVIO

In order to resolve the scale ambiguity problem and enhance the geometric constraints, we fuse the self-supervised depth estimation process with the output of DeepVIO based on the 3D geometry structure. Depending on the dense correspondences and the pose from DeepVIO, we can directly recover the sparse depth map D_s with an absolution scale through the two view triangulation module [42]. Then the sparse depth map D_s aligns the scale of prediction depth D_p by using the scale factor $w_s = mean(D_s/D_p)$. Then, the refined depth $D_f = w_s D_p$ is supervised by sparse depth D_s to minimize error. The depth loss L_{sd} is defined with D_s:

$$L_{sd} = \sum_p \|D_s - D_f\| \tag{21}$$

The total training loss is given by

$$L_{total} = L_{pe} + \lambda_{ds} L_{ds} + \lambda_{rt} L_{rt} + \lambda_{sd} L_{sd} \tag{22}$$

where λ_{ds}, λ_{rt}, λ_{sd} are the weight of edge-aware loss L_{ds}, pose loss L_{rt} and sparse depth L_{sd}.

4. Results

In this section, we conduct several experiments to present the evaluation results of depth and odometry estimation on the KITTI [48] and Oxford RobotCar dataset [49] dataset. We support our analysis with some visualizations, to verify our design decisions.

4.1. Implementation Details

As shown in Figure 2, our framework includes three subnetworks—SuperPoint, DepthNet, and DeepVIO—implemented in PyTorch. There are around 20 M trainable parameters and it takes 40 h to train the network on a GTX 2080Ti GPU. The input image resolution is set to 640 × 192; the batch size is set to 4. Adam optimizer is used for minimizing the loss function, with $\beta_1 = 0.9$, $\beta_2 = 0.99$, and the batch size is set to 4. The weights λ_{ds}, λ_{ds} in the loss function are set to 0.55 and λ_{ds} is set to 0.1. For DeepVIO, we use the pre-trained SuperPoint network to extract and match the correspondences and then connect to DVO for pose estimation. We set $c_r = 0.15$, $c_r = 0.75$ and $\beta_{rt} = 0.1$. Firstly, we only train the DeepVIO network with DIO supervising 25 epochs, then use the trained DeepVIO to train depth estimation in an unsupervised manner via image reconstruction loss. After 25 epochs, we then jointly train both networks for 10 epochs.

4.2. Datasets

To train, validate, and test our system, we validate our design on the original KITTI dataset [50] and KITTI Odometry dataset. The original KITTI dataset consists of 389 pairs of stereo images and depth maps, 39.2 km of visual ranging sequences, a Velodyne laser scanner, and a GPS/IMU localization unit, sampled and synchronized at 10 Hz. The odometry benchmark consists of 22 stereo sequences, saved in lossless png format. It provides 11 sequences (00–10) with ground truth trajectories for training and 11 to 21 sequences) without ground truth for evaluation. For the original KITTI dataset, according to Eigen et al., we split 23,488 images from 32 scenes for training and 697 images from 29 scenes for testing [50].

The Oxford RobotCar dataset [49] uses the Oxford RobotCar platform, an autonomous Nissan LEAF, to traverse the route through the Oxford city centre twice a week on average between May 2014 and December 2015. The dataset records over 1000 km of driving

records, collecting nearly 20 million images from 6 cameras mounted on the vehicle, as well as LiDAR, GPS, and INS ground truth. We use the Oxford RobotCar dataset to test the robustness of our algorithm.

4.3. Depth Estimation

We adopted evaluations on the KITTI Raw and KITTI Odometry datasets. There were four error metrics already used in previous works [4,6,50], namely absolute relative error (Abs Rel), square relative error (Sq Rel), root mean square error (RMSE), and the root mean square error in log space (RMSE log). Other accuracy metrics are the percentages of pixels where the ratio (δ) between the estimated depth and ground truth depth is smaller than 1.25, 1.25^2, and 1.25^3. We compare our method with several self-supervised depth estimation SOT methods and summarize our results in Table 1. In addition,we illustrate their performance qualitatively in Figure 4. In contrast to previous methods, our method outperforms other competitors and shows improvements in most evaluation metrics. It improves the baseline method by 8%. We show that our proposed DeepVIO architecture can increase the geometric constraints of monocular depth and improve the accuracy of monocular depth estimation. In Figure 4, we compare the supervised depth estimation DORN [29], unsupervised depth estimation Monodepth2 [30] with end-to-end PoseNet and unsupervised depth estimation TrainFlow [24], with PoseNet, based on optical flow, respectively. The results show that our proposed DeepVIO method can improve the accuracy of depth estimation and enhance the detail of depth estimation at the edge of objects.

Table 1. Quantitative comparison between our proposed system and state-of-the-art depth learning methods for monocular depth estimation on the KITTI dataset. Bold font indicates best results.

Methods	**Error**				**Accuracy, δ**		
	AbsRel	**SqRel**	**RMS**	**RMSlog**	**<1.25**	**$<1.25^2$**	**$<1.25^3$**
Zhou et al. [51]	0.183	1.595	6.709	0.270	0.734	0.902	0.959
Mahjourian et al. [52]	0.163	1.240	6.220	0.250	0.762	0.916	0.968
Geonet [53]	0.155	1.296	5.857	0.233	0.793	0.931	0.973
DDVO [54]	0.151	1.257	5.583	0.228	0.810	0.936	0.974
DF-Net [55]	0.150	1.124	5.507	0.223	0.806	0.933	0.973
CC [56]	0.140	1.070	5.326	0.217	0.826	0.941	0.975
EPC++ [57]	0.141	1.029	5.350	0.216	0.816	0.941	0.976
Struct2depth (-ref.) [58]	0.141	1.026	5.291	0.215	0.816	0.945	0.979
GLNet (-ref.) [59]	0.135	1.070	5.230	0.210	0.841	0.948	0.980
SC-SfMLearner [60]	0.137	1.089	5.439	0.217	0.830	0.942	0.975
Gordon et al. [61]	0.128	0.959	5.230	0.212	0.845	0.947	0.976
Monodepth2 (w/o pretrain) [30]	0.132	1.044	5.142	0.210	0.845	0.948	0.977
Monodepth2 [30]	0.115	0.882	4.701	0.190	0.879	0.961	0.982
Ours	**0.105**	**0.842**	**4.628**	**0.208**	0.860	**0.973**	**0.986**

An ablation study is carried out for depth estimation performance of dynamic objects, such as people or cars in the point cloud. We combine RGB and depth projection into 3D point clouds with camera intrinsic K, and compare it with the supervised method DORN [29], and the unsupervised method, TrainFlow and Monodepth2. As shown in Figure 5, compared with the supervised method DORN and the optical flow-based VO supervised depth estimation method TrainFlow [24], we find that the fusion of sparse point clouds and the absolute scale into the unsupervised depth estimation could significantly improve the monocular depth estimation results in the dynamic environment.

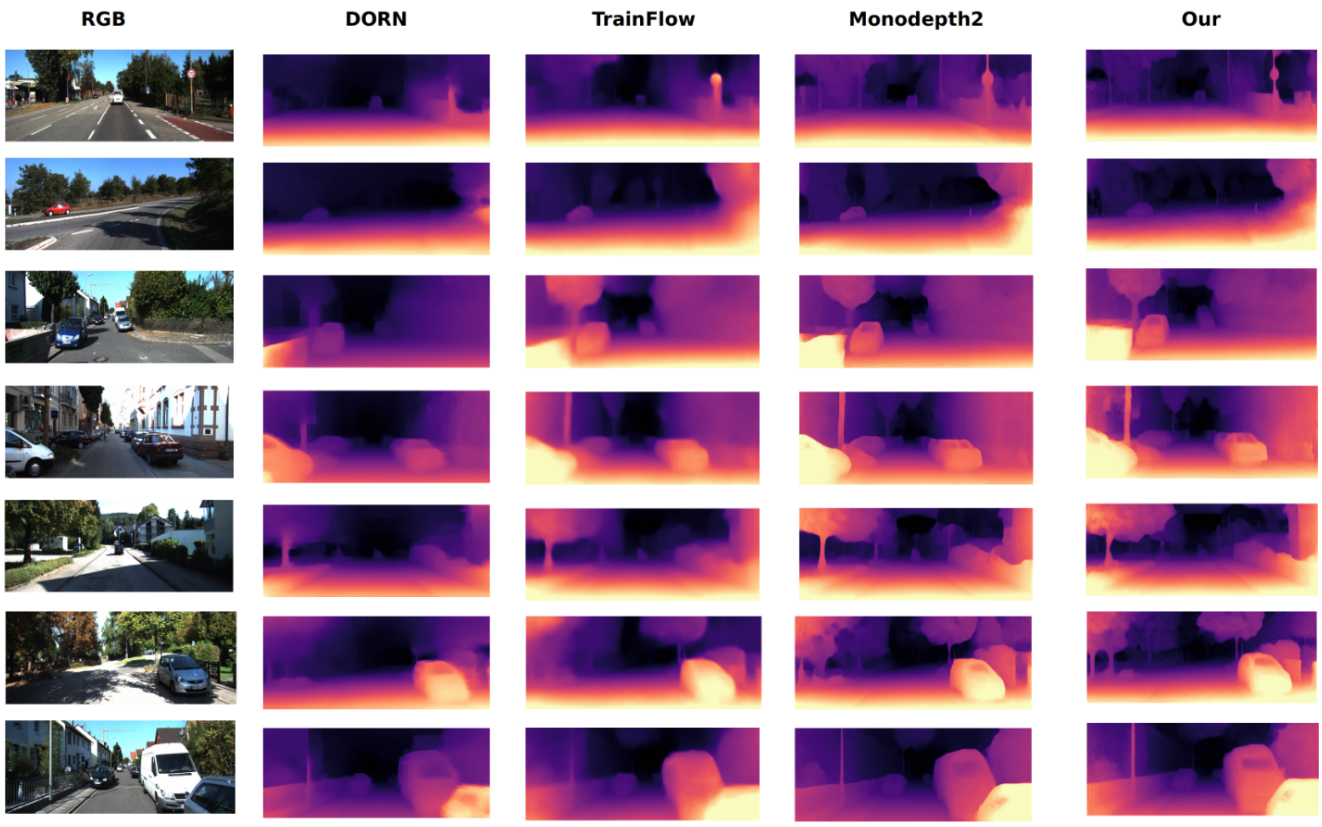

Figure 4. Qualitative results on the KITTI dataset. The depth result from left to right: DORN [29], TrainFlow [24], Monodepth2 [30] and Ours.

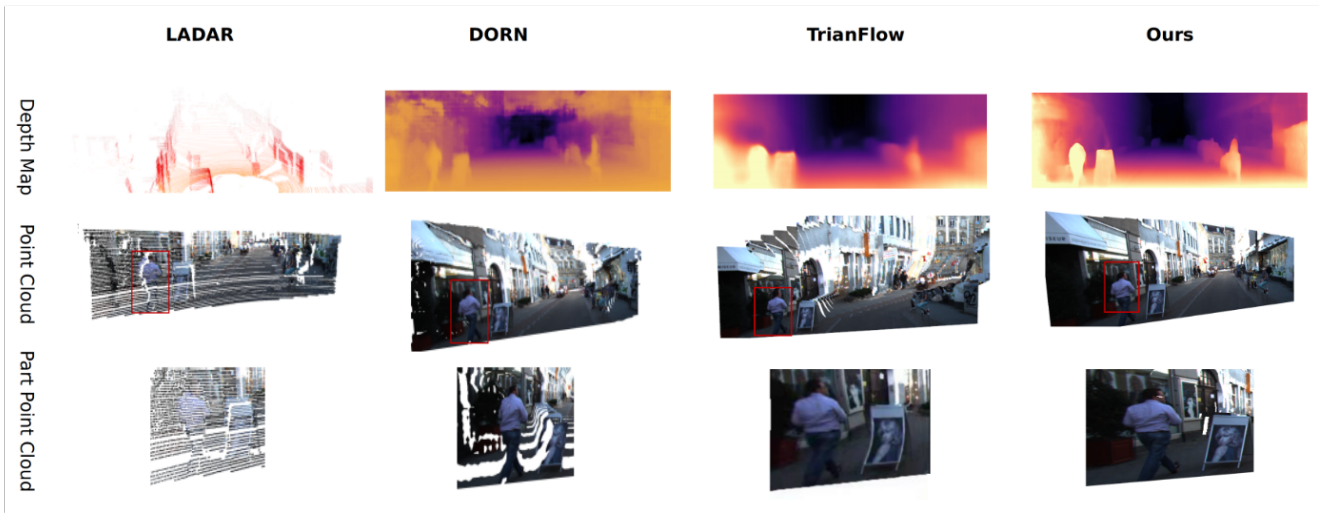

Figure 5. The influence of dynamic objects on the depth map visualization. From top to bottom: depth map, point clouds and partial point clouds generated by raw LiDAR, DORN, TrainFlow, and by our approach.

4.4. Pose Estimation

We follow the previous works on KITTI Odometry criteria evaluating possible subsequences of length (100, 200, . . . , 800) meters and report the average translational errors $t_{err}(\%)$ and rotational errors $r_{err}(^\circ/100\text{ m})$. It measures the difference between the points of the ground truth and the predicted trajectory. Using timestamps to associate the ground truth poses with the corresponding predicted poses, we compute the difference between each pair of poses and output the mean and standard deviation.

Table 2 reports the evaluation results of the DeepVIO output poses, and compares them to the previous works, such as ORB-SLAM2 [36], Deep-VO-Feat [52], SFM-Learner [51], SC-SfMLearner [60]. Both extensions improve the baseline and the attention module performs well. When coupled with the self-supervised depth estimation, the DeepVIO performance training—to have a consistent pose estimation—outperforms all of the state-of-the-art, compared to classical SLAM libviso2 and learning-based techniques, Sc-SfMLearner and SfMLearner [5,13,62]. Figures 6 and 7 show the trajectory in the XY-plane. In Figure 6, our trajectory can start from the starting point and return to the origin, forming a closed loop, such as the graph, proving that our pose estimation is relatively accurate. In Figures 6 and 7, especially in contrast to the GT trajectories, our trajectories are able to follow the GT since our approach of introducing absolute scales could preserve the reality scale of the pose.

Table 2. Visual odometry results in the KITTI Odometry dataset. The average translation and rotation errors are reported. Bold font indicates best results.

Methods	Seq.09		Seq.10	
	t_{err}(%)	r_{err}(°/100 m)	t_{err}(%)	r_{err}(°/100 m)
VISO2	18.06	1.25	26.10	3.26
ORB-SLAM2 [9]	2.84	0.25	3.30	0.30
Deep-VO-Feat [52]	11.89	3.6	12.82	3.41
SC-SfMLearner [60]	7.64	2.19	10.74	4.58
SfMLearner [51]	11.32	4.07	15.25	4.06
Ours	**2.41**	0.31	**2.19**	0.41

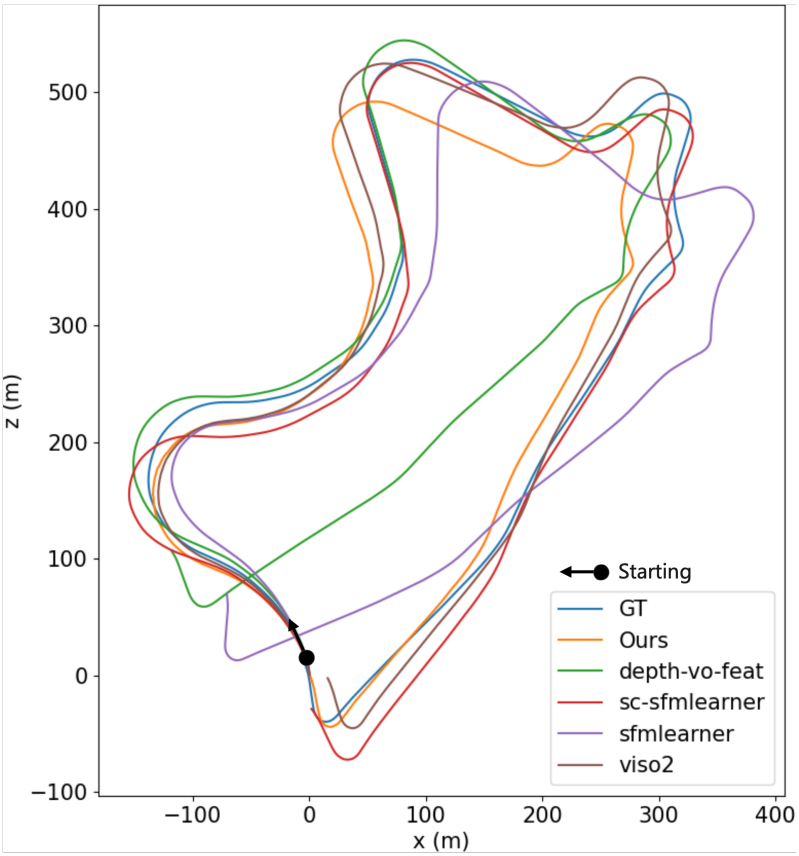

Figure 6. Results of pose estimation: KITTI sequence 09 trajectory.

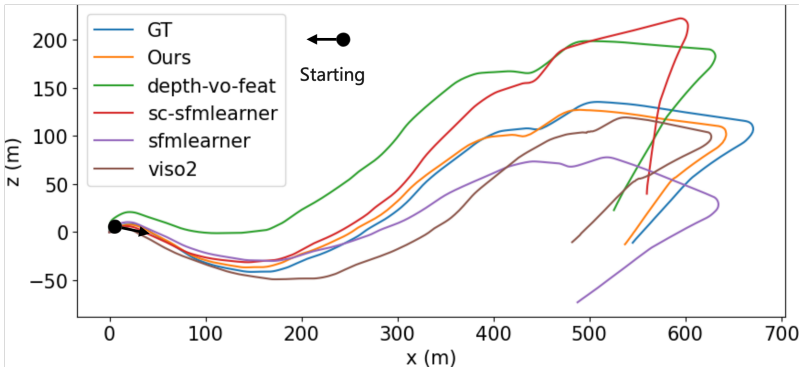

Figure 7. Results of pose estimation: KITTI sequence 10 trajectory.

4.5. Ablation Study

We also performed ablation experiments to examine the effectiveness of our contributions. The first ablation study was carried out by comparing the depth value error between the predicted depth and ground truth. Random snippets of images were taken from the KITTI dataset , testing the images through the framework. Then we selected the points with the larger error between the predicted value and ground truth value. Experimental results are shown in Figure 8. It can be observed that, in the weak texture region, or far away areas, the predicted depth from our framework obtained the absolute scale with DeepVIO, improving generalization ability.

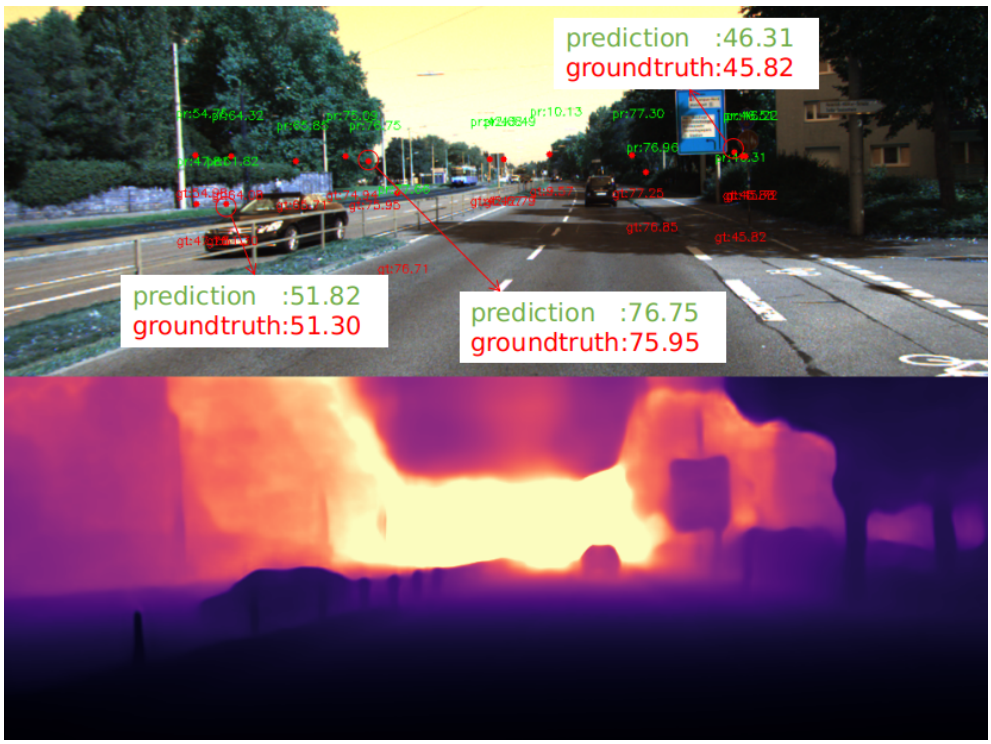

Figure 8. The maximum depth error visulization. The (**top**) mapmask the points which the error between GT depth andpredicted depth is more than 1m. The (**bottom**) is the predicteddepth map.

In addition, we used the Oxford RobotCar dataset, which, including video image sequences and IMU data to test the adaptability of our method. In the RobotCar dataset experiment, we compare it with TrainFlow [24] and Monodepth2 [30]. The results in Figure 9 show that our method is more applicable to other datasets than other methods. This proves our method can adapt to different environments, since we added DeepVIO in the depth estimation.

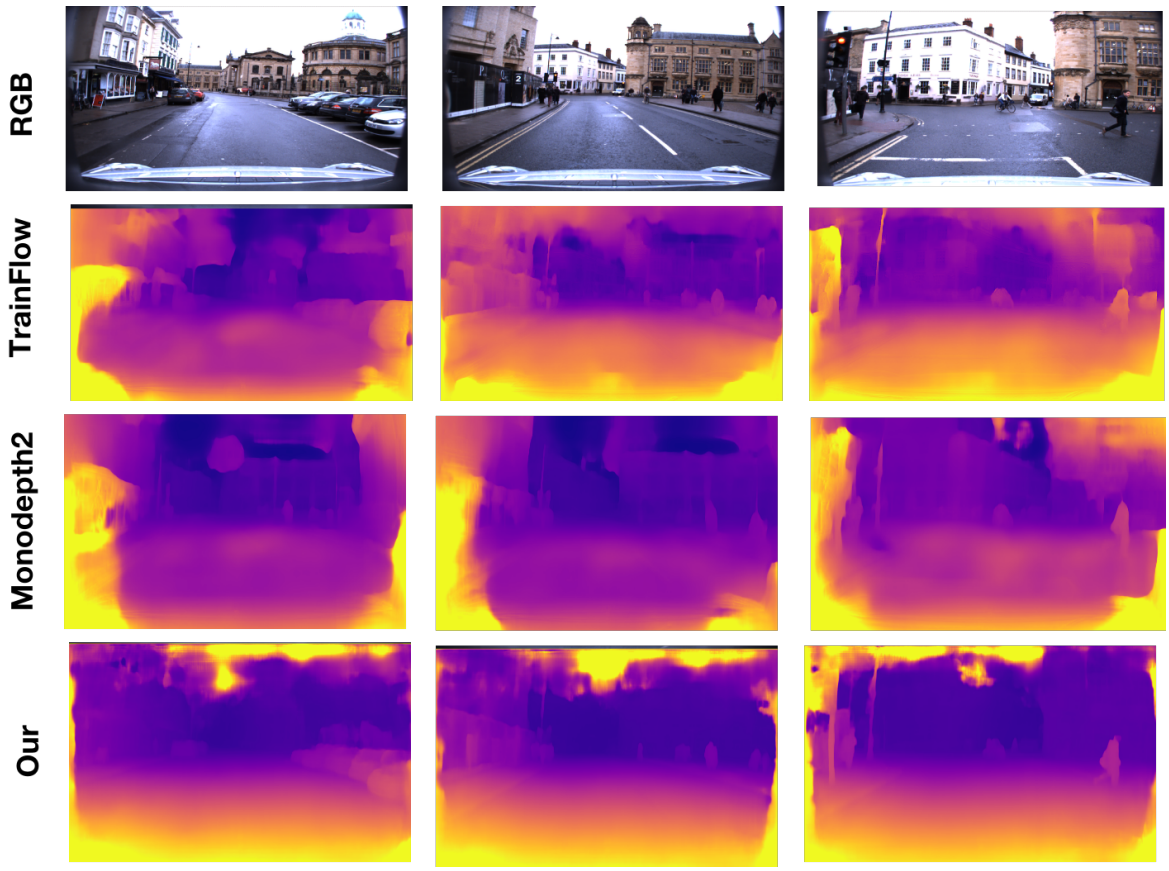

Figure 9. The depth estimation results on Oxford RobotCar [49] dataset.

5. Discussion

The proposed, new, self-supervised depth and pose estimation framework combines DepthNet with DeepVIO to supervise each other. To our knowledge, it is the first such attempt in this domain. The proposed model shows good depth estimation and pose results compared to other reference methods. These experiments also demonstrate the applicability of the EKF fusion is valid at pose estimation and absolute scale. In self-supervised depth estimation, we make full use of the pose and sparse depth produced by Force DeepVIO, where the pose is used to synthesize the target image to minimize the reprojection error, and the sparse depth is used to correct the dense depth output by DepthNet. In depth estimation result evaluations, 3D point clouds synthesized with estimated depths, and camera parameters could "value" the depth and pose accuracy. In particular, in autonomous driving scenarios, where the camera on the car is moving and there are moving objects in the scene, depth estimation is challenging. As shown in Figure 10, in the point cloud restoration experiment, our method reconstructs the point cloud of the detailed parts of the scene, such as cars and utility poles. Compared with other methods, our method can restore the geometry of the objects better.

Despite the overall promising results, our network framework contains many sub-networks: DepthNet, SuperNet, DIO, and DVO. In the process of joint network training, it is necessary to train partial networks and then freeze their parameters to train other networks, which is prone to failure. In the experiments, we first pre-train DeepVIO, use the network parameters provided by SuperNet, train DIO and DVO, and finally train jointly with DepthNet. Therefore, it is necessary to consider how to simplify the network structure and reduce the number of sub-networks in future work. Furthermore, in self-supervised depth estimation, the method of normalizing the dense depth estimated by DepthNet, with the mean of the sparse depths, over-relies on the number of sparse depth values. If

few feature points are extracted and matched, the effect of the depth scale supervision is degraded.

Figure 10. Depth estimation result point cloud visualization on the KITTI dataset.

How to improve the geometric constraints of self-supervised depth estimation has always been an important issue in the field. Recently, some researchers used sparse LiDAR to complement the depth map estimated by the depth estimation network to increase geometric constraints. The fusion of sparse LiDAR points and IMU raw data can directly calculate the pose, which can reduce the feature extraction and matching process of deep learning. This not only reduced the number of sub-networks and computing resources, but also increases the geometric constraints and the pose of the true scale, and could solve complex network problems. In addition, depth estimation also needs to consider some special scenarios, such as dark, foggy, rainy, and snowy weather. These scenes are very challenging scenes, and some recent studies have focused on these problems, such as Wang, K et al. [63] research on depth estimation in night environments. With the widespread application of depth estimation, new research needs to consider special scenarios to make depth estimation more general.

6. Conclusions

We propose a new depth and odometry estimation framework that integrates DeepVIO with depth estimation in a self-supervised learning-based method. We combined the strengths of learning-based VIO and depth estimation to build an end-to-end learning architecture. The deep keypoint-based visual odometry module captures dense correspondences by using the SuperPoint feature detector and descriptor and solves the pose and sparse depth through the two-view triangulation geometry method. The DVO joins the DIO by EKF and predicts and updates the pose state. Finally, the sparse depth and pose are used to refine prediction depth and enhance geometry reconstruction. The experiments show that our presented model outperforms all other state-of-the-art depth estimation methods on the KITTI dataset, and shows excellent generalization ability on the Oxford RobotCar dataset.

Future work includes the depth completion method for guiding depth estimation with the sparse depth from DeepVIO to bring further improvements. Finally, exploring the benefits of the improved depth prediction for 3D reconstruction is another interesting research direction [35,44].

Author Contributions: Conceptualization, Y.W. and Q.Z.; methodology, Y.W. and L.F.; software, Y.W. and Q.Z.; validation, C.G., Y.W. and Q.Z.; formal analysis, Y.W.; investigation, Y.W.; resources, Y.W.; data curation, Y.W. and C.X.; writing—original draft preparation, Y.W.; writing—review and editing, Y.W. and L.F.; visualization, Y.W. and C.G.; supervision, L.F.; project administration, L.F.; funding acquisition, L.F. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the Central Leading Local Science and Technology Development Foundation of Liaoning Province under grant no. 2021JH6/10500132 and the Natural Science Foundation of Liaoning Province under grant no. 2019-KF-03-02.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest. The funder had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

References

1. Ding, M.; Huo, Y.; Yi, H.; Wang, Z.; Shi, J.; Lu, Z.; Luo, P. Learning depth-guided convolutions for monocular 3d object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, Seattle, WA, USA, 14–19 June 2020; pp. 1000–1001.
2. Kang, R.; Shi, J.; Li, X.; Liu, Y.; Liu, X. DF-SLAM: A deep-learning enhanced visual SLAM system based on deep local features. *arXiv* **2019**, arXiv:1901.07223.
3. Yang, X.; Zhou, L.; Jiang, H.; Tang, Z.; Wang, Y.; Bao, H.; Zhang, G. Mobile3DRecon: Real-time Monocular 3D Reconstruction on a Mobile Phone. *IEEE Trans. Vis. Comput. Graph.* **2020**, *26*, 3446–3456. [CrossRef] [PubMed]
4. Godard, C.; Mac Aodha, O.; Brostow, G.J. Unsupervised monocular depth estimation with left-right consistency. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 270–279.
5. Sadek, A.; Chidlovskii, B. Self-Supervised Attention Learning for Depth and Ego-motion Estimation. *arXiv* **2020**, arXiv:2004.13077.
6. Fu, C.; Dong, C.; Mertz, C.; Dolan, J.M. Depth Completion via Inductive Fusion of Planar LIDAR and Monocular Camera. *arXiv* **2020**, arXiv:2009.01875.
7. Lin, J.T.; Dai, D.; Van Gool, L. Depth estimation from monocular images and sparse radar data. *arXiv* **2020**, arXiv:2010.00058.
8. Ji, P.; Li, R.; Bhanu, B.; Xu, Y. MonoIndoor: Towards Good Practice of Self-Supervised Monocular Depth Estimation for Indoor Environments. In Proceedings of the ICCV 2021, Montreal, QC, Canada, 11–17 October 2021.
9. Mur-Artal, R.; Tardós, J.D. Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras. *IEEE Trans. Robot.* **2017**, *33*, 1255–1262. [CrossRef]
10. Yang, N.; Stumberg, L.v.; Wang, R.; Cremers, D. D3vo: Deep depth, deep pose and deep uncertainty for monocular visual odometry. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 1281–1292.
11. Kopf, J.; Rong, X.; Huang, J.B. Robust Consistent Video Depth Estimation. *arXiv* **2020**, arXiv:2012.05901.
12. Jin, F.; Zhao, Y.; Wan, C.; Yuan, Y.; Wang, S. Unsupervised Learning of Depth from Monocular Videos Using 3D-2D Corresponding Constraints. *Remote Sens.* **2021**, *13*, 1764. [CrossRef]
13. Han, L.; Lin, Y.; Du, G.; Lian, S. Deepvio: Self-supervised deep learning of monocular visual inertial odometry using 3d geometric constraints. *arXiv* **2019**, arXiv:1906.11435.
14. Almalioglu, Y.; Turan, M.; Sari, A.E.; Saputra, M.; Gusmão, P.D.; Markham, A.; Trigoni, N. SelfVIO: Self-Supervised Deep Monocular Visual-Inertial Odometry and Depth Estimation. *arXiv* **2019**, arXiv:1911.09968.
15. Wei, P.; Hua, G.; Huang, W.; Meng, F.; Liu, H. Unsupervised Monocular Visual-inertial Odometry Network. In Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence and Seventeenth Pacific Rim International Conference on Artificial Intelligence IJCAI-PRICAI-20, Tokyo, Japan, 11–17 July 2020.
16. Sartipi, K.; Do, T.; Ke, T.; Vuong, K.; Roumeliotis, S.I. Deep Depth Estimation from Visual-Inertial SLAM. In Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Prague, Czech Republic, 27 September–1 October 2020; pp. 10038–10045.
17. You, Z.; Tsai, Y.H.; Chiu, W.C.; Li, G. Towards Interpretable Deep Networks for Monocular Depth Estimation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Nashville, TN, USA, 19–25 June 2021; pp. 12879–12888.
18. Bhutani, V.; Vankadari, M.; Jha, O.; Majumder, A.; Kumar, S.; Dutta, S. Unsupervised Depth and Confidence Prediction from Monocular Images using Bayesian Inference. In Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Prague, Czech Republic, 27 September–1 October 2020; pp. 10108–10115.

19. Zhang, H.; Ye, C. DUI-VIO: Depth uncertainty incorporated visual inertial odometry based on an rgb-d camera. In Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Prague, Czech Republic, 27 September–1 October 2020; pp. 5002–5008.
20. Zhu, Z.; Ma, Y.; Zhao, R.; Liu, E.; Zeng, S.; Yi, J.; Ding, J. Improve the Estimation of Monocular Vision 6-DOF Pose Based on the Fusion of Camera and Laser Rangefinder. *Remote Sens.* **2021**, *13*, 3709. [CrossRef]
21. Wagstaff, B.; Peretroukhin, V.; Kelly, J. Self-supervised deep pose corrections for robust visual odometry. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 31 May 2020; pp. 2331–2337.
22. Jau, Y.Y.; Zhu, R.; Su, H.; Chandraker, M. Deep Keypoint-Based Camera Pose Estimation with Geometric Constraints. In Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Prague, Czech Republic, 27 September–1 October 2020; pp. 4950–4957.
23. DeTone, D.; Malisiewicz, T.; Rabinovich, A. Superpoint: Self-supervised interest point detection and description. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Salt Lake City, UT, USA, 18–22 June 2018; pp. 224–236.
24. Zhao, W.; Liu, S.; Shu, Y.; Liu, Y.J. Towards better generalization: Joint depth-pose learning without posenet. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 9151–9161.
25. Guizilini, V.; Ambrus, R.; Burgard, W.; Gaidon, A. Sparse Auxiliary Networks for Unified Monocular Depth Prediction and Completion. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 19–25 June 2021; pp. 11078–11088.
26. Laina, I.; Rupprecht, C.; Belagiannis, V.; Tombari, F.; Navab, N. Deeper depth prediction with fully convolutional residual networks. In Proceedings of the 2016 Fourth International Conference on 3D Vision (3DV), Stanford, CA, USA, 25–28 October 2016; pp. 239–248.
27. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 770–778.
28. Karsch, K.; Liu, C.; Kang, S.B. Depth extraction from video using non-parametric sampling. In Proceedings of the European Conference on Computer Vision, Florence, Italy, 7–13 October 2012; pp. 775–788.
29. Fu, H.; Gong, M.; Wang, C.; Batmanghelich, K.; Tao, D. Deep ordinal regression network for monocular depth estimation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 2002–2011.
30. Godard, C.; Mac Aodha, O.; Firman, M.; Brostow, G.J. Digging into self-supervised monocular depth estimation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Long Beach, CA, USA, 16–20 June 2019; pp. 3828–3838.
31. Garg, R.; Bg, V.K.; Carneiro, G.; Reid, I. Unsupervised cnn for single view depth estimation: Geometry to the rescue. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; pp. 740–756.
32. Yang, N.; Wang, R.; Stuckler, J.; Cremers, D. Deep virtual stereo odometry: Leveraging deep depth prediction for monocular direct sparse odometry. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 October 2018; pp. 817–833.
33. Zhang, J.; Wang, J.; Xu, D.; Li, Y. HCNET: A Point Cloud Object Detection Network Based on Height and Channel Attention. *Remote Sens.* **2021**, *13*, 5071. [CrossRef]
34. Watson, J.; Aodha, O.M.; Prisacariu, V.; Brostow, G.; Firman, M. The Temporal Opportunist: Self-Supervised Multi-Frame Monocular Depth. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 19–25 June 2021; pp. 1164–1174.
35. Rosten, E.; Drummond, T. Machine learning for high-speed corner detection. In Proceedings of the European Conference on Computer Vision, Graz, Austria, 7–13 May 2006; pp. 430–443.
36. Mur-Artal, R.; Montiel, J.M.M.; Tardos, J.D. ORB-SLAM: A versatile and accurate monocular SLAM system. *IEEE Trans. Robot.* **2015**, *31*, 1147–1163. [CrossRef]
37. Lowe, D.G. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.* **2004**, *60*, 91–110. [CrossRef]
38. Han, X.; Leung, T.; Jia, Y.; Sukthankar, R.; Berg, A.C. Matchnet: Unifying feature and metric learning for patch-based matching. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–15 June 2015; pp. 3279–3286.
39. Yi, K.M.; Trulls, E.; Lepetit, V.; Fua, P. Lift: Learned invariant feature transform. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; pp. 467–483.
40. Qin, T.; Li, P.; Shen, S. Vins-mono: A robust and versatile monocular visual–inertial state estimator. *IEEE Trans. Robot.* **2018**, *34*, 1004–1020. [CrossRef]
41. Zuo, X.; Merrill, N.; Li, W.; Liu, Y.; Pollefeys, M.; Huang, G. CodeVIO: Visual-inertial odometry with learned optimizable dense depth. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 30 May–5 June 2021; pp. 14382–14388.
42. DeTone, D.; Malisiewicz, T.; Rabinovich, A. Toward geometric deep slam. *arXiv* **2017**, arXiv:1707.07410.
43. Muller, P.; Savakis, A. Flowdometry: An optical flow and deep learning based approach to visual odometry. In Proceedings of the 2017 IEEE Winter Conference on Applications of Computer Vision (WACV), Santa Rosa, CA, USA, 27–29 March 2017; pp. 624–631.

44. Dosovitskiy, A.; Fischer, P.; Ilg, E.; Hausser, P.; Hazirbas, C.; Golkov, V.; Van Der Smagt, P.; Cremers, D.; Brox, T. Flownet: Learning optical flow with convolutional networks. In Proceedings of the IEEE International Conference on Computer Vision, Boston, MA, USA, 7–12 June 2015; pp. 2758–2766.
45. Wang, S.; Clark, R.; Wen, H.; Trigoni, N. End-to-end, sequence-to-sequence probabilistic visual odometry through deep neural networks. *Int. J. Robot. Res.* **2018**, *37*, 513–542. [CrossRef]
46. Shamwell, E.J.; Leung, S.; Nothwang, W.D. Vision-aided absolute trajectory estimation using an unsupervised deep network with online error correction. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018; pp. 2524–2531.
47. Schnabel, R.; Wahl, R.; Klein, R. Efficient RANSAC for Point-Cloud Shape Detection. In *Computer Graphics Forum*; Blackwell Publishing Ltd.: Oxford, UK, 2010; pp. 214–226.
48. Geiger, A.; Lenz, P.; Urtasun, R. Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite. In Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR), Providence, RI, USA, 16–21 June 2012 .
49. Maddern, W.; Pascoe, G.; Linegar, C.; Newman, P. 1 year, 1000 km: The oxford robotcar dataset. *Int. J. Robot. Res.* **2017**, *36*, 3–15. [CrossRef]
50. Eigen, D.; Puhrsch, C.; Fergus, R. Depth map prediction from a single image using a multi-scale deep network. *arXiv* **2014**, arXiv:1406.2283.
51. Zhou, T.; Brown, M.; Snavely, N.; Lowe, D.G. Unsupervised Learning of Depth and Ego-Motion from Video. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 22–25 July 2017.
52. Mahjourian, R.; Wicke, M.; Angelova, A. Unsupervised Learning of Depth and Ego-Motion from Monocular Video Using 3D Geometric Constraints. In Proceedings of the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018.
53. Yin, Z.; Shi, J. GeoNet: Unsupervised Learning of Dense Depth, Optical Flow and Camera Pose. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018.
54. Wang, C.; Buenaposada, J.M.; Rui, Z.; Lucey, S. Learning Depth from Monocular Videos using Direct Methods. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018.
55. Zou, Y.; Luo, Z.; Huang, J.B. DF-Net: Unsupervised Joint Learning of Depth and Flow using Cross-Task Consistency. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 October 2018.
56. Ranjan, A.; Jampani, V.; Balles, L.; Kim, K.; Black, M.J. Competitive Collaboration: Joint Unsupervised Learning of Depth, Camera Motion, Optical Flow and Motion Segmentation. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–22 June 2019.
57. Luo, C.; Yang, Z.; Peng, W.; Yang, W.; Yuille, A. Every Pixel Counts ++: Joint Learning of Geometry and Motion with 3D Holistic Understanding. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *42*, 2624–2641. [CrossRef]
58. Casser, V.; Pirk, S.; Mahjourian, R.; Angelova, A. Depth Prediction without the Sensors: Leveraging Structure for Unsupervised Learning from Monocular Videos. *Proc. AAAI Conf. Artif. Intell.* **2019**, *33*, 8001–8008. [CrossRef]
59. Chen, Y.; Schmid, C.; Sminchisescu, C. Self-supervised Learning with Geometric Constraints in Monocular Video: Connecting Flow, Depth, and Camera. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Long Beach, CA, USA, 16–22 June 2019.
60. Bian, J.W.; Li, Z.; Wang, N.; Zhan, H.; Shen, C.; Cheng, M.M.; Reid, I. Unsupervised scale-consistent depth and ego-motion learning from monocular video. *arXiv* **2019**, arXiv:1908.10553.
61. Gordon, A.; Li, H.; Jonschkowski, R.; Angelova, A. Depth from Videos in the Wild: Unsupervised Monocular Depth Learning from Unknown Cameras. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Long Beach, CA, USA, 16–22 June 2019.
62. Wang, S.; Clark, R.; Wen, H.; Trigoni, N. Deepvo: Towards end-to-end visual odometry with deep recurrent convolutional neural networks. In Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA), Singapore, 29 May–3 June 2017; pp. 2043–2050.
63. Wang, K.; Zhang, Z.; Yan, Z.; Li, X.; Xu, B.; Li, J.; Yang, J. Regularizing Nighttime Weirdness: Efficient Self-supervised Monocular Depth Estimation in the Dark. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 11–17 October 2021.