2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

Monocular Visual Odometry via Diffusion-Based Joint Learning of Optical Flow and Depth

Qingyuan Hu[1,2], Wei Li[1,3*], *Member, IEEE*, Xuebin Meng[1,2], Yu Hu[1,3*], *Member, IEEE*

***Abstract*— Monocular visual odometry (VO) often suffers from scale ambiguity and interference from moving objects in real-world scenarios. Jointly learning optical flow and depth estimation provides a promising solution for these issues by leveraging their geometric correlation and task complementarity. In this paper, we propose JFD-VO, a novel monocular VO framework that integrates jointly learned optical flow and depth networks. We design a two-stage training process with recursive noise diffusion and a specialized loss function, which enables the model to predict dense and scale-aware depth and optical flow using only readily available sparse LiDAR data and pose ground truth, thereby eliminating the need for expensive and difficult-to-obtain dense annotations. Furthermore, a dedicated masking module is incorporated during joint training to enhance robustness in dynamic environments. Within the VO pipeline, we introduce a Keypoint-weighted Matching Selection module that prioritizes stable features based on forward-backward flow consistency, rather than treating all pixels equally as in conventional optical flow methods. Extensive experiments on public datasets demonstrate the effectiveness of our joint training approach. JFD-VO achieves state-of-the-art accuracy, reducing absolute trajectory error by 14.99% and 27.37% over KPDepth-VO and DF-VO. Code and our self-collected dataset are available at: https://github.com/huqingyuan-9952/JFD-VO.**

I. INTRODUCTION

Autonomous navigation and other high-level intelligent tasks for unmanned systems require accurate and robust localization. Cameras offer the advantage of providing rich semantic information with relatively low-cost and lightweight. This capability has led to the widespread adoption of monocular visual odometry (VO) systems, favored for their simplicity and not requiring extrinsic calibration and complex sensor setups. However, monocular VO is inherently limited by scale ambiguity and is susceptible to moving objects, hindering real-world deployment. Leveraging deep learning to infer critical intermediate representations of VO, such as optical flow and depth, offers a promising path to mitigate these issues, as neural networks can provide dense predictions and learn powerful priors from data.

The intrinsic relationship between optical flow and depth makes them ideal candidates for joint learning [1]–[4], a synergy that offers multifaceted advantages. Firstly, they are geometrically correlated through the camera's ego-motion. The motion of a pixel between frames is directly related to both the camera's displacement and the distance from the corresponding 3D point to the camera, which are the outputs of optical flow and depth estimation respectively. This interdependence allows depth information to help resolve the aperture problem in optical flow estimation, particularly in areas with repetitive textures. Secondly, this joint learning framework facilitates self-supervised training of optical flow. Acquiring dense and accurate ground truth for optical flow is difficult and expensive. By integrating a depth estimation network and camera pose, the resulting photometric reprojection error can be employed as a supervisory signal for the optical flow, eliminating the need for manual annotation.

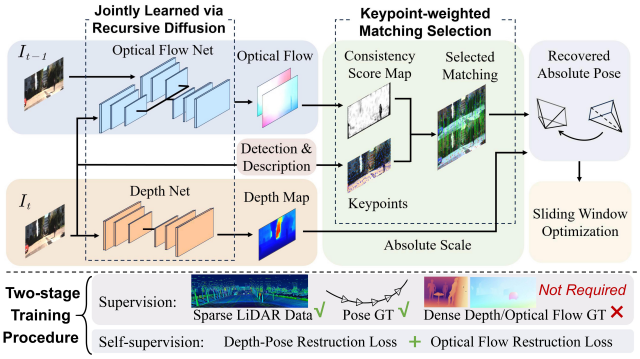

Fig. 1. The architecture of our proposed monocular visual odometry.

Similar to optical flow, depth estimation also suffers from a lack of dense, accurate ground truth data. Many datasets provide only sparse depth measurements obtained from LiDARs, which results in the presence of holes (missing labels) in ground-truth depth data [5]. To address the sparse-to-dense completion problem, diffusion models [6] offer a generative paradigm that shifts the objective from regressing the mean value to generating the most probable and geometrically coherent depth structure. Existing methods applying diffusion models to depth estimation have demonstrated their effectiveness in improving accuracy and scene generalization [7]–[9]. Nonetheless, as these methods are not specifically designed for visual odometry, they mainly focus on recovering relative scale. In our joint learning framework for optical flow and depth, we introduce a novel two-stage training strategy accompanied by a specialized loss function, which enables the depth estimation network to learn absolute scale with recursive noise diffusion. This capability is critical for resolving the scale ambiguity inherent in monocular VO.

This work was supported by Beijing Natural Science Foundation (L243008), and in part by National Natural Science Foundation of China under Grant No. 62003323 and No. 62176250.

[1]The Research Center for Intelligent Computing Systems, SKLP, Institute of Computing Technology, Chinese Academy of Sciences, Beijing, 100190, China.
[2]Hangzhou Institute for Advanced Study, University of Chinese Academy of Sciences, Hangzhou, 310024, China.
[3]University of Chinese Academy of Sciences, Beijing, 100049, China.
*Corresponding author. Email: liwei2019@ict.ac.cn, huyu@ict.ac.cn.

Beyond the aforementioned advantages, the joint learning of optical flow and depth offers further robustness in handling dynamic environments, which constitutes a common challenge in robotic and autonomous driving applications. In our framework, a dedicated module for estimating dynamic masks is designed. This enables the model to explicitly learn to distinguish between static scenes and moving objects, thereby enhancing the accuracy of VO in real-world dynamic settings. Furthermore, while optical flow can provide dense correspondence [10], we observe that not all pixels contribute equally to motion estimation. Distinctive, scale-invariant keypoints (e.g., corners) typically offer more stable and reliable matches across frames. Therefore, we design a Keypoint-weighted Matching Selection module to prioritize reliable features during optical flow consistency computation, thus enhancing pose estimation accuracy and robustness.

In this paper, we propose JFD-VO, a novel monocular visual odometry system built upon a diffusion-based joint learning framework for optical flow and depth, as shown in Fig.1. Three key contributions are as follows:

- A joint learning framework for optical flow and depth estimation is proposed. We design a two-stage training strategy with dynamic masks and recursive noise diffusion, effectively leveraging easily available sparse LiDAR data and pose ground truth to generate dense, scale-aware outputs.
- A Keypoint-weighted Matching Selection module is introduced, utilizing forward-backward flow consistency to prioritize stable and salient keypoints for correspondence matching, thereby improving the accuracy of VO.
- We integrate the above components into a unified pipeline, JFD-VO, and conduct experiments on both public and self-collected datasets. The results demonstrate that JFD-VO achieves state-of-the-art accuracy on various datasets.

II. RELATED WORK

A. Monocular Visual Odometry

Traditional Monocular VO approaches can be broadly categorized into feature-based and direct methods. Feature-based methods, such as ORB-SLAM2 [11], rely on detecting, matching, and followed by bundle adjustment for motion refinement. However, these methods often fail in low-texture or repetitive regions. In contrast, direct methods like DSO [12] and LDSO [13] optimize photometric consistency across images. While effective under consistent lighting, they are highly sensitive to illumination variations and motion blur.

With the advent of deep learning, data-driven methods have gained prominence. SfMLearner [14] combines a pose network and a depth network trained via photometric loss. PVO [15] integrates video panoptic segmentation and uses a panoptic update module to mitigate the impact of dynamic objects. DPVO [16] proposes tracking sparse patch flow between images. This line was extended in DPV-SLAM [17] with loop closure integration, forming a complete SLAM system with enhanced global consistency. Nonetheless, many learning-based approaches still lack explicit geometric constraints, which can lead to instability in pose predictions, especially under challenging motion conditions.

Hybrid methods have emerged to combine the advantages of learned representations and geometric constraints. DynaSLAM [18] employs a segmentation network to detect dynamic objects using multi-view consistency, while CNN-SVO [19] replaces manual depth filters with predicted depths for faster feature initialization. Recent efforts such as *Wang et al.* [20] propose filtering keypoints with high uncertainty when computing the scale factor, and KPDepth-VO [21] further introduces a point filter network to increase reliable matches. KP3D [22] focus on 3D keypoint extraction and tracking to support motion estimation. DF-VO [10] integrates depth and optical flow networks to recover metric scale and establish dense correspondences, illustrating the trend toward multi-task learning in VO.

B. Joint Learning for Optical Flow and Depth Estimation

Existing joint learning methods [14], [23]–[26] train monocular depth and pose networks together to simultaneously predict both depth and camera motion from a single image. These methods optimize depth and pose networks using photometric consistency. Due to the absence of supervision from the ground truth (GT) pose, they lack knowledge of absolute scale. Recent works, such as DF-Net [2], align depth and optical flow through cross-task consistency, effectively leveraging complementary information from both tasks to improve overall accuracy. Similarly, GeoNet [1] combines depth and optical flow networks through 3D scene geometry for joint learning. Competitive Collaboration (CC) [3] proposes a framework for joint optimization of depth, pose, optical flow, and segmentation, aiming to simultaneously improve multiple aspects of scene understanding. DOPlearning [4] enhances performance by jointly estimating depth, optical flow, and pose, while also incorporating explicit segmentation of rigid, non-rigid, and occluded regions to improve the robustness of the model in complex environments. Although these joint learning methods combine the advantages of multiple tasks, many of them still struggle with scale ambiguity in monocular scenes. The lack of absolute scale limits their direct application in monocular VO.

C. Diffusion-Based Depth Network

Recent studies integrate diffusion models into depth estimation, using their ability to model complex distributions through denoising. DDP [7] uses an encoder to extract image features and iteratively decodes the depth map using a diffusion model. DiffusionDepth [8] combines Swin Transformers with latent-space diffusion to reduce complexity while preserving fine-grained details. EcoDepth [9] refines depth predictions through a diffusion architecture guided by a Vision Transformer. However, many diffusion-based methods rely on dense depth map as input to the model, which limits the ability of sparse depth maps to effectively integrate with diffusion models. Ground truth annotations for depth data are often sparse [5]. In recent work, *Kolbeinsson*

et al. propose recursive noise diffusion [27] and apply it to the segmentation task. During training, this method avoids using ground truth as input to the model. It has latent capacity to use sparse depth data to improve network performance.

III. METHODOLOGY

A. Framework of monocular visual odometry

JFD-VO adopts a hybrid visual odometry pipeline, consisting of the motion tracking stage and the local optimization stage. The overall architecture is depicted in Fig.1, and details of the key modules are elaborated below.

*1) **keypoint-weighted matching selection**:* During the motion tracking stage, the system takes a pair of images, I_t and I_{t-1} as input. Forward and backward optical flows are estimated using a jointly learned network. Keypoints and their confidence scores are extracted from using SuperPoint [28] to evaluate match quality, as outlined in Algorithm 1. We use a score weight λ to adjust the influence of the keypoint confidence scores on optical flow consistency. A forward-backward consistency check is applied to filter out unreliable matches caused by occlusion or lighting variations. This process yields uniform and robust pixel correspondences between I_t and I_{t-1}.

*2) **Pose recovery**:* Ego-motion is estimated from pixel correspondences under the epipolar constraint by computing the essential matrix. Singular Value Decomposition recovers the relative ego-motion from the essential matrix but lacks absolute scale. To resolve this scale ambiguity, the system uses triangulated depth as initialization and aligns it with scale-aware depth predictions from the jointly-learned depth network. In pure rotation scenarios, where translation cannot be estimated from the essential matrix, the depth network establishes 3D–2D correspondences instead. The camera pose is then computed via Perspective-n-Point (PnP), with RANSAC applied to filter outlier correspondence.

*3) **Local Optimization**:* In the local optimization stage, we follow the approach used in DSO [12]. The motion tracking stage provides robust pixel correspondences $\mathbf{kp}_{\text{robust}}$, their depth information, and initial camera poses. Keyframes are selected based on camera motion and point coverage, and then stored in a sliding window for local optimization. This further improves the accuracy of the entire VO system.

B. Joint optical flow-depth learning

We combine depth and optical flow networks through a two-stage training procedure, as shown in Fig. 2. To improve the inter frame consistency of prediction, both networks take three consecutive frames $\boldsymbol{I}_{t-1}, \boldsymbol{I}_t, \boldsymbol{I}_{t+1} \in \mathbb{R}^{H\times W\times 3}$ as input. The depth network outputs the depth maps $\boldsymbol{D}_{t-1}, \boldsymbol{D}_t, \boldsymbol{D}_{t+1} \in \mathbb{R}^{H\times W\times 1}$, and the optical flow network produces the optical flow $\boldsymbol{F}_{t-1,t}, \boldsymbol{F}_{t+1,t} \in \mathbb{R}^{H\times W\times 2}$. In the first stage, the network is jointly trained using sparse depth and true pose. The sparse GT-depth introduces a scale-invariant error, quantified by the following function:

$$S(g) = \frac{1}{\Omega}\sum_i g_i^2 - \frac{\alpha}{\Omega^2}\left(\sum_i g_i\right)^2, \tag{1}$$

Algorithm 1 Keypoint-weighted Matching Selection

Input: keypoints $\mathbf{kp}$, keypoint scores $\mathbf{ks}$, flow $\mathbf{F}_{\text{forward}}$, flow $\mathbf{F}_{\text{backward}}$, threshold τ, keypoint score weight λ, grid point number Ψ

Output: selected correspondences $\mathbf{kp}_{\text{robust}}$

Step 1: Calculate forward-backward consistency score

for each pixel p in $\mathbf{F}_{\text{forward}}$ and $\mathbf{F}_{\text{backward}}$ **do**

  Compute the consistency score:

$$\mathbf{score}_{\text{consistency}}[p] = \frac{1}{1 + \|\mathbf{F}_{\text{forward}}[p] - \mathbf{F}_{\text{backward}}[p]\|_2}$$

end for

Set scores less than τ to 0 in $\mathbf{score}_{\text{consistency}}$.

Step 2: Apply weighted scoring on keypoints

for each keypoint pt in $\mathbf{kp}$ **do**

  Find corresponding pixel $p \leftarrow \mathbf{kp}[pt]$

  Combine consistency score with keypoint score:

$$\mathbf{score}_{\text{final}}[p] = \lambda \cdot \mathbf{ks}[pt] \cdot \mathbf{score}_{\text{consistency}}[p]$$

end for

Step 3: Grid-based matching selection

Divide $\mathbf{score}_{\text{final}}$ into grid cells

for each grid cell **do**

  Select top Ψ points with the highest scores.

end for

Each selected point is paired with a correspondence.

return selected correspondences $\mathbf{kp}_{\text{robust}}$

where $g_i = \log \tilde{\boldsymbol{D}}_t^{(i)} - \log \boldsymbol{D}_t^{(i)}$ represents the logarithmic error between the GT-depth $\tilde{\boldsymbol{D}}_t^{(i)}$ and the estimated depth $\boldsymbol{D}_t^{(i)}$ at the i-th pixel, with $\alpha = 0.85$ and Ω refers to the total number of pixels with GT-depth labels. To improve convergence, the loss function is scaled as $L_{\text{sparse}} = \beta\sqrt{S(g)}$, where$\beta$ is set as 10 based on practical experience. This scaling stabilizes training and ensures proper weight of depth estimation error.

Additionally, the pose transformation between I_t and its adjacent frames is used to compute the reconstructed image:

$$\hat{I}_{t+1} = K T_{t+1,t}^{gt}\left(D_{t+1}K^{-1}I_{t+1}\right), \tag{2}$$

where K represents the camera intrinsic matrix and $T_{t+1,t}^{gt}$ represent the GT-transformation between I_{t+1} and I_t. The consistency between the reconstructed and original images is used as a supervisory signal:

$$L_{\text{restruct}} = \sum_{r\in\{t-1,t+1\}} \sigma\left(I_t, \hat{I}_r\right). \tag{3}$$

$\sigma(I_t, \hat{I}_r)$ measures the similarity between the original image I_t and reconstructed image $\hat{I}_r$, here $r \in \{t-1, t+1\}$. The similarity σ can be calculated by:

$$\sigma(a,b) = \lambda_s \frac{(1 - SSIM(a,b))}{2} + (1-\lambda_s)\|a-b\|. \tag{4}$$

SSIM denotes the structural similarity index, $\|\cdot\|$ represents L1 norm, and $\lambda_s = 0.85$. Restruction loss assumes correct

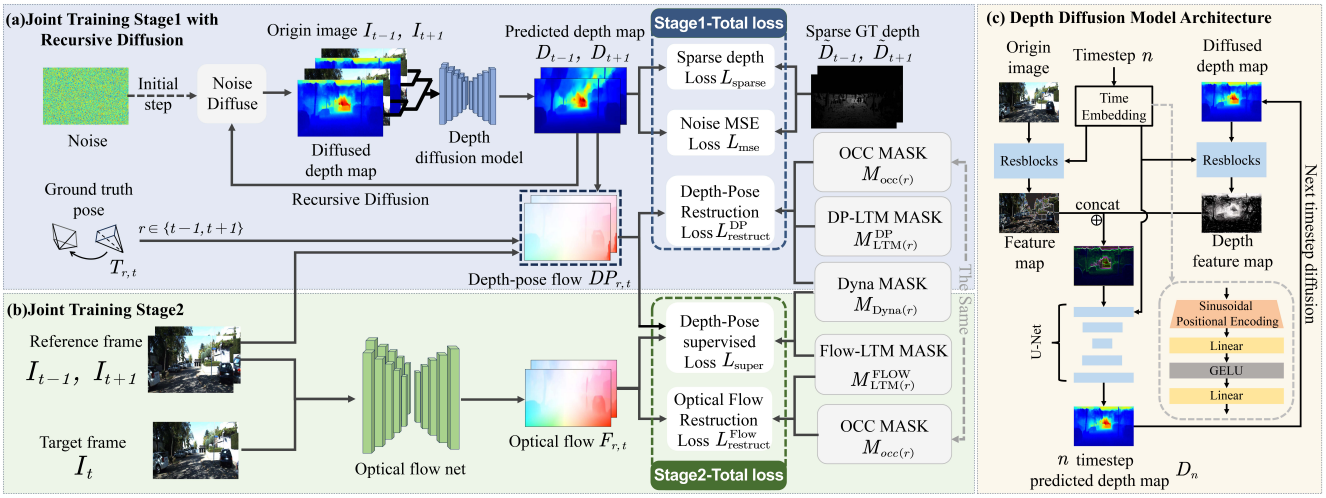

Fig. 2. The architecture of our proposed joint learning method with recursive noise diffusion.

matching between the reconstructed and original images, but camera motion may introduce occlusion, overlap, and boundary issues. To improve matching accuracy, we use an occlusion mask composed of edge, overlap, and blank masks, following the DOPlearning [4] method, defined as $M_{\text{occ}(t)} = M_{\text{edg}(t)} M_{\text{ove}(t)} M_{\text{bla}(t)}$. Besides, outliers in the reconstructed image may affect the loss function. To address this, we discard pixels with errors greater than the average using a mask to exclude outliers, as shown in (5):

$$M^{\text{DP}}_{\text{ltm}(r)} = \mathbb{I}\left(\sigma\left(I_t, \hat{I}_r\right) < \frac{1}{\mu}\sum_{\xi}\sigma\left(I_t, \hat{I}_r M_{\text{occ}(r)}\right)\right), \quad (5)$$

where $\mathbb{I}(\cdot)$ is an indicator function. ξ presents the pixels within masks and μ is the total number. Regions with smaller reprojection error from depth-pose flow $re(DP_{r,t}) = re(T_{r,t}, D_r)$ than optical flow $re(F_{r,t})$, or where both predictions are consistent within a threshold λ_m, are considered static. Here $re(\cdot)$ represents the reprojection error between the reconstructed and original images. $m\left(F_{r,t}\right)$ denotes the pixel motion predicted by the optical flow network, and $m\left(T_{r,t}, D_r\right)$ denotes the pixel motion calculated by depth map and pose transformation. The dynamic mask is set as:

$$\begin{aligned} M_{\text{Dyna}(r)} = &\mathbb{I}(re\left(DP_{r,t}\right) < re\left(F_{r,t}\right)) \\ &\vee \mathbb{I}(\|m\left(DP_{r,t}\right) - m\left(F_{r,t}\right)\|_2 < \lambda_m), \end{aligned} \quad (6)$$

where $\lambda_m = 0.3$, $\vee$ means logical *or* and $\|\cdot\|_2$ represents L2 norm. The final reconstruction loss in the first stage is:

$$L^{\text{DP}}_{\text{restruct}} = \sum_{r\in\{t-1,t+1\}}\sum_{\xi}\sigma\left(I_t, \hat{I}_r\right) M_{\text{occ}(r)} M^{\text{DP}}_{\text{ltm}(r)} M_{\text{Dyna}(r)}. \quad (7)$$

Finally, a smoothness loss is introduced to enforce spatial smoothness on the predicted depth, inspired by GeoNet [1]:

$$L_{\text{smooth}} = \left|\frac{\partial D_t}{\partial x}\right| e^{-\left|\frac{\partial I_t}{\partial x}\right|} + \left|\frac{\partial D_t}{\partial y}\right| e^{-\left|\frac{\partial I_t}{\partial y}\right|}. \quad (8)$$

In summary, during the first stage of joint training, the loss function is computed as a combination of several components, $L^{\text{stage1}}_{\text{total}} = \lambda_1 L_{\text{sparse}} + \lambda_2 L^{\text{DP}}_{\text{restruct}} + \lambda_3 L_{\text{smooth}}$.

In the second stage, the optical flow network provides the pixel motion $[u_i, v_i]^T$. By adding this motion vector to the pixel positions coordinates $[u, v]^T$, the restructed image $\bar{I}_{t+1}$ of image I_{t+1} is obtained by projecting it onto I_t. Since the optical flow network is robust to dynamic objects, it does not require a dynamic mask for reconstruction loss, as in (9):

$$L^{\text{Flow}}_{\text{restruct}} = \sum_{r\in\{t-1,t+1\}}\sum_{\xi}\sigma\left(I_t, \bar{I}_r\right) M_{\text{occ}(r)} M^{\text{Flow}}_{\text{ltm}(r)}. \quad (9)$$

Since depth-pose flow offers more accurate flow estimation for static pixels, we design a depth supervision loss L_{super}, using the trained depth network from the first stage.

$$L_{\text{super}} = \sum_{r\in\{t-1,t+1\}}\sum_{\xi}\phi\left(DP_{r,t}, F_{r,t}\right) M_{\text{Dyna}(r)} M^{\text{DP}}_{\text{ltm}(r)}, \quad (10)$$

where $\phi(DP_{r,t}, F_{r,t})$ calculates the pixel-wise motion difference between the depth-pose flow and optical flow using L2 loss. The final training function for the second stage is $L^{\text{stage2}}_{\text{total}} = \lambda_4 L^{\text{Flow}}_{\text{restruct}} + \lambda_5 L_{\text{super}}$. These two stages improve the performance of both the depth and optical flow networks.

C. Recursive noise diffusion

As illustrated in Fig. 2(c), we introduce a recursive noise diffusion model that helps predict the missing regions of sparse depth ground truth. This model comprises a timestep head, an image encoder head, a diffused depth encoder head, and an encoder–decoder module. The timestep head converts the time step to a sinusoidal positional embedding, while both the image and depth heads each comprise three ResNetBlocks, and their outputs are concatenated along the channel dimension. We use U-Net [29] as encoder–decoder module, where time-step embeddings are incorporated at multiple stages of the network by injecting them into each ResNetBlock. Given an image $I \in \mathbb{R}^{W\times H\times 3}$, the corresponding depth map $D \in \mathbb{R}^{W\times H\times 1}$, and a time step $n \in [0, N]$, we define the forward noise addition process q, which adds Gaussian noise with variance $\beta_n = n/N$:

$$q(D_n|D_{n-1}) = D_{n-1} + \mathcal{N}(\mathbf{0}, \beta_n \mathbf{I}). \quad (11)$$

The noise added at each step is represented as $\epsilon_n = D_n - D_0$. We employ a neural network to approximate the reverse denoising process, where $\epsilon_n \approx \epsilon_\theta(D_n, I, n)$. The predicted depth map at any time step is $D_0 \approx D_n - \epsilon_\theta(D_n, I, n)$. In the training phase, the loss function is the mean squared error (MSE) between the predicted and actual noise:

$$L_{\text{mse}} = \mathbb{E}_{D_0, I, n, \epsilon_n} \left[\|\epsilon_n - \epsilon_\theta(D_n, I, n)\|^2 \right]. \quad (12)$$

Our method starts denoising with pure noise. Subsequent steps diffuse noise based on the predicted depth map from the previous step. The model uses the original image to denoise the diffused depth map. In the final step, the denoised depth map is compared with the ground truth. Notably, ground-truth depth is never directly input into the model during training, mitigating bias from depth completion. By adopting this recursive denoising strategy, we achieve more precise depth predictions over time, with each step refining the result based on previous predictions. Based on this, the first stage of joint training is integrated with the recursive noise diffusion model with the final loss function $L_{\text{total}}^{\text{stage1}'} = L_{\text{total}}^{\text{stage1}} + L_{\text{mse}}$.

IV. EXPERIMENTS AND ANALYSIS

A. Implementation Details

In the two-stage training procedure, we use image sequences from the raw KITTI [30] dataset. Sparse ground-truth depth is obtained from LiDAR measurements. The true pose matrix is obtained from the provided Global Navigation Satellite System/Inertial Measurement Unit (GNSS/IMU) and transformed into a local coordinate system. In this section, the optical flow network refers to LiteFlowNet [31] while the depth network refers to [27]. Note that the network architectures for depth and optical flow are interchangeable, as the proposed diffusion-based joint learning framework is designed to accommodate these replacements.

Our models are implemented in PyTorch and trained on an RTX 3090 GPU, while the VO was tested on a laptop with an Intel i9-13900HX CPU and an RTX 4060 GPU. All images are scaled to 640×192. In the first stage of joint training, the depth network is trained for 10 epochs using the sparse ground-truth depth from the raw KITTI dataset to learn denoising capabilities for depth map. Then we used the optical flow network along with true pose to provide dynamic masks and built the reconstruction loss for depth. The values for the loss terms were set to $\lambda_1 = 1.0$, $\lambda_2 = 0.5$, and $\lambda_3 = 0.1$. This step lasted for 30 epochs. In the second joint training stage, the depth network and true pose together provide a more stable static scene flow to supervise the optical flow network. We set $\lambda_4 = 1.0$ and $\lambda_5 = 0.5$. This stage trained for 20 epochs. Then the entire training process continued until the validation loss converged, using the Adam optimizer with a learning rate of 10^{-4}.

B. Depth Estimation Results

The widely used evaluation protocol for depth prediction applies scale alignment preprocessing, with a scale $\hat{s} = median\,(D_{gt})\,/median\,(D_{pred})$. This procedure prevents a faithful assessment of the depth network's ability to estimate absolute scale, which is essential for monocular VO. We thus also report results without scale alignment. Additionally, to verify the contribution of the recursive noise diffusion model, we compare with a variant trained without it, which uses the U-Net architecture from Monodepth2 [23]. All depth evaluations are conducted on the test set of *Eigen et al.* [32] with depth capped at 80 meters, as per standard practice. The evaluation involves 697 images.

The results are presented in Table I. The top section shows results with scale alignment applied, while the bottom section shows results without it. Since the proposed joint learning framework allows the depth network architecture to be replaced, Table 1 primarily compares our method against baseline and relevant joint learning frameworks. Compared to [1]–[4], our joint training framework not only learns absolute scale but also improves depth prediction accuracy. Moreover, by incorporating recursive noise diffusion, the depth prediction precision is further enhanced.

C. Optical Flow Estimation Results

The evaluation of the optical flow network is conducted on the KITTI 2015 training set using 200 samples. Since the proposed joint learning framework eliminates the dependency on expensive dense optical ground truth, this section focuses primarily on comparing against other self-supervised optical flow methods and joint-learning frameworks. Among the compared methods, UnFlow2 [33] utilizes a 3-level cascade network for flow refinement. Joint learning methods GeoNet [1] and DF-Net [2] utilize deeper ResNet-50 networks as their feature extraction backbone, while CC [3] and DOP [4] incorporate masking mechanisms within the pose network. Since the optical flow network integrated into our joint learning framework is adapted from the LiteFlowNet [31] architecture, this baseline is also included in the comparison. The results in Table II demonstrate that our jointly trained flow network achieves clear improvements over the baseline. Compared with DOP [4], our method reduces EPE by 52.40% and the F1-all error by 41.67%. This improvement can be attributed to the incorporation of depth estimation with absolute scale and a dynamic-static masking strategy.

D. Monocular Visual Odometry Evaluation

We evaluated the VO performance on both open-source and self-collected datasets. Due to space limitations, only Absolute Trajectory Error (ATE) results on the KITTI dataset are presented here. For additional metrics, such as rotational errors, please refer to our open-source repository. To validate JFD-VO's capability for absolute-scale ego-motion estimation, we use ATE computed by the root mean square error between the predicted and ground-truth camera trajectories, without aligning the scale between them. During the matching selection, λ is set to 5, and pixels with consistency scores below 10 are discarded. As shown in Fig. 3, pixels with darker colors in the optical flow consistency score map indicate lower consistency scores. The optical flow consistency score map effectively filters out pixels that

TABLE I

COMPARISON FOR THE DEPTH ESTIMATION ON THE EIGEN ET AL. [32] TEST SPLIT. THE BEST RESULT IS IN **BOLD** AND SECOND BEST IS UNDERLINE.

Method	Error metric(lower are better)				Accuracy metric(higher are better)		
	Abs Rel	Sq Rel	RMSE	RMSE log	$a < 1.25$	$a < 1.25^2$	$a < 1.25^3$
GeoNet [1]	0.153	1.328	5.737	0.232	0.802	0.934	0.972
DF-Net [2]	0.146	1.182	5.215	0.213	0.818	0.943	0.978
CC [3]	0.139	1.032	5.199	0.213	0.827	0.943	0.977
DOPlearning [4]	0.132	0.986	5.173	0.212	0.835	0.945	0.977
Wang [20]	0.105	0.727	4.491	0.180	0.888	0.964	0.984
Monodepth2 [23]	0.115	0.903	4.863	0.193	0.877	0.959	0.981
RM-Depth [25]	0.107	0.687	4.476	0.181	0.883	0.964	0.984
Lite-Mono [26]	0.107	0.765	4.561	0.183	0.886	0.963	0.983
Ours (w/o RND.)	0.106	0.688	4.480	0.180	0.880	0.964	**0.985**
Ours	**0.102**	**0.671**	**4.330**	**0.178**	**0.890**	**0.965**	0.984
GeoNet* [1]	0.633	7.218	13.650	1.070	0.010	0.029	0.077
CC* [3]	0.991	15.238	19.218	3.971	-	-	-
DOPlearning* [4]	0.980	15.219	19.207	3.949	-	-	-
Ours (w/o RND.)*	0.109	0.721	4.686	0.192	0.861	0.956	**0.982**
Ours*	**0.102**	**0.688**	**4.459**	**0.188**	**0.875**	**0.958**	**0.982**

Note: $*$ denotes scale alignment was not applied when evaluating error metric and accuracy metric. 'RND' dicates recursive noise diffusion.

TABLE II

COMPARISON FOR OPTICAL FLOW ESTIMATION.

Type	Optical Flow Models	EPE ↓	Fl ↓
Baseline	LiteFlowNet [31]	10.39	28.50%
Unsupervised Learning	UnFlow-C [33]	8.80	28.94%
	UnFlow-CSS [33]	8.10	23.27%
	Back2Future [34]	6.59	22.94%
	NLFlow [35]	6.05	22.75%
	DDFlow [36]	5.72	14.29%
	SelFlow [37]	4.84	14.19%
	STFlow [38]	3.56	13.83%
Jointly Learning	GeoNet [1]	10.81	-
	DF-Net [2]	8.98	26.01%
	CC [3]	6.21	26.41%
	Doplearning [4]	6.66	23.04%
	Ours	**3.17**	**13.44%**

Note: 'EPE' represents the average endpoint error, 'F1' denotes the percentage of pixels with an EPE greater than 3 pixels and more than 5% of the ground truth value.

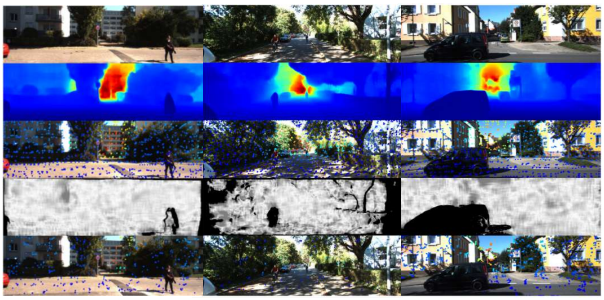

Fig. 3. Visual result of JFD-VO motion tracking stage. Top to bottom: sample image, estimated depth, extracted keypoints, optical flow consistency score map, selected matching points.

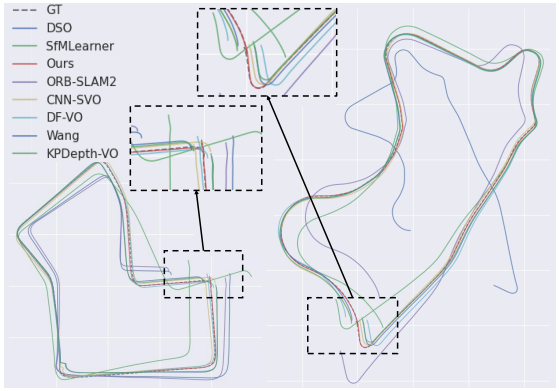

Fig. 4. Comparison results on KITTI. Left to Right: seq 07, seq 09 .

are significantly affected by lighting changes or occlusions caused by object motion. The final selected matching points are evenly distributed and robust, minimizing the impact of less reliable points on VO. For comparison, we choose the traditional geometric-based methods [11]–[13], deep learning-based methods [14], [15], [17], and hybrid methods [10], [19]–[21]. Additionally, because Sequence 01 of the KITTI dataset contains a sub-sequence without trackable features, which causes many monocular VO methods to fail or perform poorly on this sequence, we exclude the impact of Sequence 01 when calculating the average error. As shown in Table III, our method outperforms other methods, compared to DF-VO, the average ATE is reduced by 27.37%, and compared to KPDepth-VO, it is reduced by 14.99%. Notably, we find that although our method does not include loop closure detection, it exhibits negligible accumulated drift when revisiting prior locations, as shown in Fig. 4.

E. Ablation Study

As for the ablation study, we change the keypoint-weighted matching selection method and jointly trained networks to evaluate their impact on VO performance. The baseline matching selection method used is the optical flow consistency sampling method referenced in DF-VO [10], which filters only the pixels with poor optical flow consistency, neglecting the effect of keypoints. To demonstrate the improvement brought by joint learning, the baseline uses networks that were trained separately. Comparing the second and third rows of the Table IV with the first row, it can be observed that the proposed keypoint-based matching

TABLE III

ABSOLUTE TRAJECTORY ERROR (IN METRE) ON KITTI ODOMETRY SEQ. 00-10. THE BEST RESULT IS IN **BOLD** AND SECOND BEST IS UNDERLINE.

Method	00	01	02	03	04	05	06	07	08	09	10	Avg. Err.
ORB-SLAM2(w/o LC.) [11]	40.65	502.20	47.82	0.94	1.30	29.95	40.82	16.04	43.09	38.77	5.42	26.48
ORB-SLAM2(w/ LC.) [11]	6.03	508.34	14.76	1.02	1.57	4.04	11.16	2.19	38.85	8.39	6.63	9.46
DSO [12]	98.86	failed	77.55	3.08	0.75	48.65	84.59	15.78	105.58	63.26	13.71	51.18
LDSO [13]	9.32	**11.68**	31.98	2.85	1.22	5.10	13.55	2.96	129.02	21.64	17.36	23.50
SfMLearner [14]	104.87	109.61	185.43	8.42	3.10	60.89	52.19	20.12	30.97	26.93	24.09	51.70
CNN-SVO [19]	17.53	failed	50.51	3.46	2.44	8.15	11.51	6.51	10.98	10.70	4.84	12.66
PVO [15]	5.69	91.19	23.60	0.86	0.81	8.41	13.57	8.89	6.67	14.65	8.66	9.18
DF-VO [10]	12.64	695.75	23.11	1.23	1.36	3.75	2.63	1.74	7.87	11.02	3.37	6.87
DPV-SLAM++ [17]	8.30	11.86	39.64	2.50	0.78	5.74	11.60	**1.52**	110.9	76.70	13.70	27.14
Wang [20]	12.95	54.08	18.39	0.93	0.65	3.21	**2.10**	6.27	6.14	6.06	2.64	5.93
KPDepth-VO [21]	11.59	45.53	20.98	**0.84**	0.61	3.47	2.26	5.47	**5.10**	6.01	2.34	5.87
JFD-VO(Ours)	**5.56**	70.47	**13.86**	2.82	**0.45**	**3.17**	3.52	1.63	11.76	**4.81**	**2.28**	**4.99**

TABLE IV

ABLATION STUDY. THE BEST RESULT IS IN **BOLD** AND SECOND BEST IS UNDERLINE.

Method	JT-Net	KMS	00	02	03	04	05	06	07	08	09	10	Avg.Err.
Baseline			9.47	23.80	4.21	2.54	7.43	5.80	5.71	8.37	10.73	4.42	8.25
Baseline + JT-Net	✓		6.02	22.76	3.52	2.10	6.43	4.64	5.57	7.56	5.33	3.78	6.77
Baseline + KMS		✓	7.10	19.32	**1.22**	0.53	5.55	4.57	5.56	8.65	10.68	4.65	6.78
Ours(w/o Local Optim.)	✓	✓	7.65	17.63	3.12	2.31	5.90	3.82	5.22	**6.81**	4.87	3.86	6.12
Ours(w/ Local Optim.)	✓	✓	**5.56**	**13.86**	2.82	**0.45**	**3.17**	**3.52**	**1.63**	11.76	**4.81**	**2.28**	**4.99**

Note: '+ JT-Net' denotes that the VO uses the jointly trained networks, '+ KMS' indicates that the keypoint-weighted matching selection method is used.

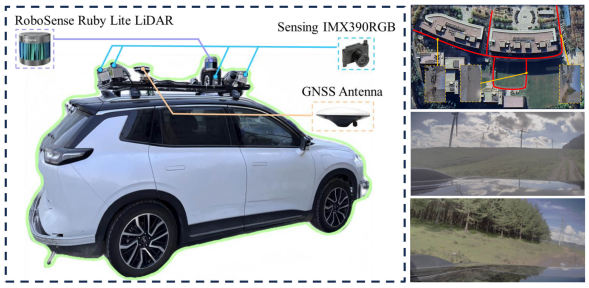

Fig. 5. The experimental setup and typical scenes from our dataset.

selection method indeed enhances accuracy by helping to avoid the influence of less robust points. Additionally, when the VO uses the jointly trained network, more accurate predictions from the network can minimize the impact of ignored keypoints. When both modules are used together in VO, a lower average ATE is achieved. Finally, the complete JFD-VO system achieves a further enhancement in accuracy through the integration with sliding window optimization. While local optimization is generally expected to improve localization accuracy, our method with the added optimization module exhibits a performance decrease on sequence 08. A possible explanation is a lack of compatibility between our implemented optimization and the environmental characteristics of the sequence 08. This potential issue is supported by the results in Table III, which show that other methods employing similar optimization, such as DSO and LDSO, also do not perform well on this particular sequence.

F. Self-collected Dataset Tests

Our data collection platform is a sport utility vehicle equipped with multiple sensors, as shown in Fig. 5, which

TABLE V

ABSOLUTE TRAJECTORY ERROR (IN METER) ON OUR DATASET.

Sequences	ORB-SLAM2 (w/o LC.) [11]	DSO [12]	CNN-SVO [19]	DF-VO [10]	KPDepth-VO [21]	Ours
00	22.88	17.68	9.75	7.18	5.74	**2.92**
01	30.34	20.20	10.01	7.17	7.29	**3.49**
02	43.28	X	14.26	10.92	8.56	**3.78**
03	45.97	32.51	8.15	10.23	6.01	**3.38**

illustrates both the vehicle and the experimental scene. Unlike the urban environment of the KITTI dataset, our self-collected data were captured within an enclosed campus and rural roads, encompassing approximately 5.2 kilometres in total length. The ground truth trajectory is derived from the post-processed results of the NovAtel PP7D-E2, a GNSS/IMU integrated navigation system. As shown in Table V, by comparing JFD-VO with other methods, we validate the superiority of our approach. Since the networks were not trained on this dataset, we also demonstrate its generalization ability, showing its effectiveness across different environments. We evaluated the runtime on a laptop detailed in Section IV-A. The most time-consuming modules in JFD-VO are the depth network inference, the optical flow network inference, and the sliding window optimization for keyframes, with average execution times of 12.27 ms, 16.44 ms, and 21.85 ms, respectively. The remaining components, including pose recovery, matching selection, etc, contribute less to the computation. On a laptop, the JFD-VO system achieves an average rate of 11 FPS, demonstrating that the method possesses the capability for practical deployment.

V. CONCLUSIONS

In this paper, we proposed JFD-VO, a novel monocular visual odometry framework with jointly learned optical flow

and depth estimation networks through a diffusion-based paradigm. Our method primarily aims to address key challenges of monocular VO in real-world environments, including scale ambiguity and sensitivity to dynamic scenes, while simultaneously eliminating the need for expensive dense ground-truth annotations. The introduction of a two-stage training strategy with a specialized loss function, a dynamic object masking module, and a keypoint-weighted matching selection mechanism collectively enhances the robustness and accuracy of pose estimation. Through experiments, we demonstrate the effectiveness of our joint training framework and the accuracy of JFD-VO. In future work, we plan to make further improvements by adding loop closure detection, incorporating additional sensor modalities and lightweighting networks to enhance our system.

REFERENCES

[1] Z. Yin and J. Shi, "Geonet: Unsupervised learning of dense depth, optical flow and camera pose," in *IEEE conference on computer vision and pattern recognition*, 2018, pp. 1983–1992.

[2] Y. Zou, Z. Luo, and J.-B. Huang, "Df-net: Unsupervised joint learning of depth and flow using cross-task consistency," in *European conference on computer vision (ECCV)*, 2018, pp. 36–53.

[3] A. Ranjan, V. Jampani, L. Balles, K. Kim, D. Sun, J. Wulff, and M. J. Black, "Competitive collaboration: Joint unsupervised learning of depth, camera motion, optical flow and motion segmentation," in *IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 12 240–12 249.

[4] G. Wang, C. Zhang, H. Wang, J. Wang, Y. Wang, and X. Wang, "Unsupervised learning of depth, optical flow and pose with occlusion from 3d geometry," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 1, pp. 308–320, 2020.

[5] S. Saxena, C. Herrmann, J. Hur, A. Kar, M. Norouzi, D. Sun, and D. J. Fleet, "The surprising effectiveness of diffusion models for optical flow and monocular depth estimation," in *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[6] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in *Proceedings of the 34th International Conference on Neural Information Processing Systems*. Red Hook, NY, USA: Curran Associates Inc., 2020.

[7] Y. Ji, Z. Chen, E. Xie, L. Hong, X. Liu, Z. Liu, T. Lu, and et al., "Ddp: Diffusion model for dense visual prediction," in *IEEE/CVF International Conference on Computer Vision*, 2023, pp. 21 741–21 752.

[8] Y. Duan, X. Guo, and Z. Zhu, "Diffusiondepth: Diffusion denoising approach for monocular depth estimation," in *European Conference on Computer Vision*. Springer, 2024, pp. 432–449.

[9] S. Patni, A. Agarwal, and C. Arora, "Ecodepth: Effective conditioning of diffusion models for monocular depth estimation," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 28 285–28 295.

[10] H. Zhan, C. S. Weerasekera, J.-W. Bian, and I. Reid, "Visual odometry revisited: What should be learnt?" in *IEEE international conference on robotics and automation (ICRA)*. IEEE, 2020, pp. 4203–4210.

[11] R. Mur-Artal and J. D. Tardós, "Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras," *IEEE transactions on robotics*, vol. 33, no. 5, pp. 1255–1262, 2017.

[12] J. Engel, V. Koltun, and D. Cremers, "Direct sparse odometry," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 3, pp. 611–625, 2017.

[13] X. Gao, R. Wang, N. Demmel, and D. Cremers, "Ldso: Direct sparse odometry with loop closure," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 2198–2204.

[14] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe, "Unsupervised learning of depth and ego-motion from video," in *IEEE conference on computer vision and pattern recognition*, 2017, pp. 1851–1858.

[15] W. Ye, X. Lan, S. Chen, Y. Ming, X. Yu, H. Bao, Z. Cui, and G. Zhang, "Pvo: Panoptic visual odometry," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 9579–9589.

[16] Z. Teed, L. Lipson, and J. Deng, "Deep patch visual odometry," *Advances in Neural Information Processing Systems*, vol. 36, pp. 39 033–39 051, 2023.

[17] L. Lipson, Z. Teed, and J. Deng, "Deep patch visual slam," in *European Conference on Computer Vision*, 2024, pp. 424–440.

[18] B. Bescos, J. M. Fácil, J. Civera, and J. Neira, "Dynaslam: Tracking, mapping, and inpainting in dynamic scenes," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 4076–4083, 2018.

[19] S. Y. Loo, A. J. Amiri, S. Mashohor, S. H. Tang, and H. Zhang, "Cnn-svo: Improving the mapping in semi-direct visual odometry using single-image depth prediction," in *2019 International conference on robotics and automation (ICRA)*. IEEE, 2019, pp. 5218–5223.

[20] C. Wang, G. Zhang, and W. Zhou, "Self-supervised learning of monocular visual odometry and depth with uncertainty-aware scale consistency," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 3984–3990.

[21] C. Wang, G. Zhang, Z. Cheng, and W. Zhou, "Kpdepth-vo: Self-supervised learning of scale-consistent visual odometry and depth with keypoint features from monocular video," *IEEE Transactions on Circuits and Systems for Video Technology*, 2025.

[22] J. Tang, R. Ambrus, V. Guizilini, S. Pillai, H. Kim, P. Jensfelt, and A. Gaidon, "Self-supervised 3d keypoint learning for ego-motion estimation," in *Conference on robot learning*, 2021, pp. 2085–2103.

[23] C. Godard, O. Mac Aodha, M. Firman, and G. J. Brostow, "Digging into self-supervised monocular depth estimation," in *IEEE/CVF international conference on computer vision*, 2019, pp. 3828–3838.

[24] W. Han, J. Yin, X. Jin, X. Dai, and J. Shen, "Brnet: Exploring comprehensive features for monocular depth estimation," in *European Conference on Computer Vision*. Springer, 2022, pp. 586–602.

[25] T.-W. Hui, "Rm-depth: Unsupervised learning of recurrent monocular depth in dynamic scenes," in *IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 1675–1684.

[26] N. Zhang, F. Nex, G. Vosselman, and N. Kerle, "Lite-mono: A lightweight cnn and transformer architecture for self-supervised monocular depth estimation," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 18 537–18 546.

[27] B. Kolbeinsson and K. Mikolajczyk, "Multi-class segmentation from aerial views using recursive noise diffusion," in *IEEE/CVF Winter Conference on Applications of Computer Vision*, 2024, pp. 8439–8449.

[28] D. DeTone, T. Malisiewicz, and A. Rabinovich, "Superpoint: Self-supervised interest point detection and description," in *IEEE conference on computer vision and pattern recognition workshops*, 2018, pp. 224–236.

[29] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

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

[31] T.-W. Hui, X. Tang, and C. C. Loy, "Liteflownet: A lightweight convolutional neural network for optical flow estimation," in *IEEE conference on computer vision and pattern recognition*, 2018, pp. 8981–8989.

[32] D. Eigen, C. Puhrsch, and R. Fergus, "Depth map prediction from a single image using a multi-scale deep network," *Advances in neural information processing systems*, vol. 27, 2014.

[33] S. Meister, J. Hur, and S. Roth, "Unflow: Unsupervised learning of optical flow with a bidirectional census loss," in *AAAI conference on artificial intelligence*, vol. 32, no. 1, 2018.

[34] J. Janai, F. Guney, A. Ranjan, M. Black, and A. Geiger, "Unsupervised learning of multi-frame optical flow with occlusions," in *European Conference on Computer Vision (ECCV)*, September 2018.

[35] L. Tian, Z. Tu, D. Zhang, J. Liu, and et al., "Unsupervised learning of optical flow with cnn-based non-local filtering," *IEEE Transactions on Image Processing*, vol. 29, pp. 8429–8442, 2020.

[36] P. Liu, I. King, M. R. Lyu, and J. Xu, "Ddflow: Learning optical flow with unlabeled data distillation," in *AAAI conference on artificial intelligence*, vol. 33, no. 01, 2019, pp. 8770–8777.

[37] P. Liu, M. Lyu, I. King, and J. Xu, "Selflow: Self-supervised learning of optical flow," in *IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 4571–4580.

[38] Z. Ren, W. Luo, J. Yan, W. Liao, X. Yang, A. Yuille, and H. Zha, "Stflow: Self-taught optical flow estimation using pseudo labels," *IEEE Transactions on Image Processing*, vol. 29, pp. 9113–9124, 2020.