Electronic Research Archive

http://www.aimspress.com/journal/era

ERA, 32(5): 3543–3568.
DOI: 10.3934/era.2024163
Received: 17 January 2024
Revised: 28 April 2024
Accepted: 09 May 2024
Published: 31 May 2024

Research article

Hybrid self-supervised monocular visual odometry system based on spatio-temporal features

Shuangjie Yuan[1], Jun Zhang[1], Yujia Lin[2] and Lu Yang[1,*]

[1] School of Automation Engineering, University of Electronic Science and Technology of China, Sichuan, China

[2] Glasgow College, University of Electronic Science and Technology of China, Sichuan, China

* **Correspondence:** Email: yanglu@uestc.edu.cn.

Abstract: For the autonomous and intelligent operation of robots in unknown environments, simultaneous localization and mapping (SLAM) is essential. Since the proposal of visual odometry, the use of visual odometry in the mapping process has greatly advanced the development of pure visual SLAM techniques. However, the main challenges in current monocular odometry algorithms are the poor generalization of traditional methods and the low interpretability of deep learning-based methods. This paper presented a hybrid self-supervised visual monocular odometry framework that combined geometric principles and multi-frame temporal information. Moreover, a post-odometry optimization module was proposed. By using image synthesis techniques to insert synthetic views between the two frames undergoing pose estimation, more accurate inter-frame pose estimation was achieved. Compared to other public monocular algorithms, the proposed approach showed reduced average errors in various scene sequences, with a translation error of 2.211% and a rotation error of 0.418 °/100m. With the help of the proposed optimizer, the precision of the odometry algorithm was further improved, with a relative decrease of approximately 10% intranslation error and 15% in rotation error.

Keywords: autonomous driving; simultaneous localization and mapping; monocular visual odometry; post optimization

1. Introduction

Recent advances in robotics and artificial intelligence research, both academic and industrial, have spurred the growth of autonomous intelligent mobile robots. A key component of their perception system is the visual odometry system.

According to the methodological framework [1], the visual odometry system can be classified into

two categories: traditional multi-view geometry-based visual odometry system and deep learning-based visual odometry system. Traditional visual odometry system relies on the principles of multi-view geometry. It computes the camera's motion by tracking the correspondences of feature points across consecutive frames. Therefore, it depends heavily on feature descriptors for accurate matching. While traditional visual odometry performs well in general textured conditions, it suffers from poor performance in textureless environments. Moreover, an inevitable issue in traditional visual odometry is the accumulation of errors over time, which causes the problem of pose drift during long-term operation [2].

Meanwhile, the advantage of deep learning-based visual odometry is its ability to adaptively learn visual feature representations [3] required for the task of visual odometry using deep neural networks. The learned feature representations exhibit good robustness in weakly textured conditions. However, supervised deep learning-based visual odometry methods require a large amount of expensive annotated data and lack good interpretability. As a result, researchers often have to optimize the algorithm by adjusting the network's hyperparameters based on the results, rather than having direct insights into the inner workings of the model. Given this situation, how to integrate traditional geometric principles with deep learning technology to complement each other has become an important challenge in the field of visual odometry [4], allowing for the combination of strengths and weaknesses.

In response to the challenges mentioned above regarding the estimation accuracy of monocular visual odometry and its generalization ability in complex scenes, this paper presents a method called the hybrid self-supervised monocular visual odometry (Hybrid-VO) system based on spatiotemporal features. Moreover, it further explores a post optimization approach for inter-frame pose refinement and proposes a flexible plug-and-play post-pose estimation optimizer. We show that our odometry with post-optimization module achieves state of the art (SOTA) results on the KITTI [5] dataset which was generated by the Karlsruhe Institute of Technology (KIT) and the Toyota Technological Institute at Chicago (TTI), and outperforms most of the published monocular self-supervised methods.

To summarize, the main contributions of this paper are:

1) This paper introduces a framework for a hybrid monocular visual odometry algorithm that combines geometric principles and multi-frame spatiotemporal information using a self-supervised deep learning network. Experimental results show that the proposed odometry method outperforms other public algorithms, showcasing its superiority.

2) This paper presents a flexible plug-and-play post pose estimation optimizer and proposes the Refined Hybrid-VO. The optimizer utilizes image synthesis techniques to insert synthetic frames between the frames undergoing pose estimation for optimization. The experimental results indicate that this optimization approach further enhances the accuracy of current algorithms.

The rest of this paper is organized as follows. Section II presents related works. Section III details the proposed method. Section IV evaluates and compares the proposed Refined-Hybrid-VO to the other public methods. Section V concludes the paper.

2. Related work

There has been rapid development in the research of visual odometry frameworks in recent years. Based on their fundamental principles, they can be broadly classified into traditional multi-view geometry-based visual odometry and deep learning-based visual odometry. Before reviewing the relevant works on visual odometry, it is essential to note that visual odometry is only the front-end of visual simultaneous localization and mapping (SLAM) systems, not the whole SLAM system itself. It serves as a crucial foundation for the other components of the SLAM system.

2.1. Traditional method

The visual odometry methods based on the traditional multi-view geometry principles can be primarily divided into three categories: feature-based methods utilizing feature point matching and tracking, direct methods based on the photometric invariance assumption, and semi-direct methods. The feature-based methods typically involve three steps: feature extraction, feature matching and tracking, and motion pose estimation, focusing on matching the feature points between adjacent image frames. The assumption of photometric invariance is fundamental to the direct method [6]. It involves estimating self-motion by tracking the variations of photometric values at individual pixel locations across consecutive frames. In the semi-direct method, local pixel patches containing feature points in the image are directly matched to solve for changes in camera motion and pose, unlike the direct method that matches the entire image. Feature-based methods are the mainstream in traditional visual odometry. Nowadays, many deep learning-based visual odometry approaches essentially utilize image feature matching to estimate self-motion, exhibiting high inherent consistency.

For feature-based methods, the key is to focus on feature descriptors and the parameter selection for similarity measurement used in matching. Numerous researchers have made significant contributions to the design of feature descriptors. In 2004, Lowe [7] introduced the renowned scale-invariant feature transform (SIFT) descriptor. In 2006, Bay et al. [8] proposed speeded up robust features (SURF), which offers higher computational efficiency. In 2011, Rubble et al. [9] presented another milestone feature descriptor called oriented FAST and rotated BRIEF (ORB). Experimental results have demonstrated that ORB is nearly two orders of magnitude faster than SIFT while exhibiting invariance to image rotation and noise resistance. However, it lacks scale invariance, which has a significant impact on scale estimation in monocular visual odometry.

During the development of traditional visual odometry, many outstanding system frameworks and solutions have emerged. In 2007, Klein and Murray [10] proposed a monocular SLAM framework called parallel tracking and mapping (PTAM). It utilized a feature descriptor known as FAST. Although PTAM's performance was not entirely satisfactory, it provided a complete and general framework for SLAM. In 2014, Engle et al. [11] introduced large-scale direct monocular SLAM (LSD-SLAM), which utilized all image pixels and established a semi-dense map of the surrounding environment by selecting optimal matching points through stereo matching and filtering. However, due to the strong assumption of photometric invariance, this system performed poorly in dynamic scenes. In 2015, Mur-Artal et al. [12] made various improvements to PTAM, resulting in a complete monocular camera-based SLAM system known as ORB-SLAM. This system adopted ORB features and introduced essential graph theory tools to accelerate the regression process during loop closure detection and correction. One major advantage of this system is its real-time operation. However, due

to the lack of scale invariance in ORB features, the system still faces the challenge of uncertain monocular scale estimation. Moreover, feature-based traditional methods for odometry can also be applied to various complex environments. Nordfeldt-Fiol et al. [13] proposed an improvement to the visual odometry system,Library for Visual Odometry 2 (LIBVISO2), by enhancing its feature detection module to search for a combination of feature detectors and descriptors that are better suited for complex environments and robot motion. Birem et al. [14] proposed a visual odometry method based on Fourier transform The method is particularly suitable for ground surfaces with no obvious visual features. De-Maeztu et al. [15] combined spacetime information that can be applied to most stereo visual odometry algorithms, greatly reducing their computational complexity.

These visual odometers based on traditional methods are more or less dependent on texture and feature information with the environment. Once in a weakly textured or even textureless environment, the performance of these visual odometers will degrade significantly.

2.2. Deep learning-based methods

Benefiting from the adaptability of deep learning techniques in extracting task-specific optimized feature representations, existing deep learning-based methods no longer require the use of manually designed feature descriptors. Deep learning-based visual odometry methods can be categorized based on their training approaches: supervised and learning-based visual odometry.

The earliest approach in supervised learning-based visual odometry involved training neural networks on labeled datasets to directly map consecutive frame images to the space of relative pose transformations, thereby obtaining estimates of relative motion between frames. The first work in this field was conducted by Konda [14], who formulated the visual odometry problem as a classification problem and trained a neural network to predict discrete direction and velocity vectors between input image pairs. Constante et al. [16] utilized deep neural networks to extract visual features from image optical flow for predicting relative motion between frames. These early works only crudely leveraged deep neural networks for visual feature extraction, thus not achieving satisfactory results.

The Towards end-to-end visual odometry with deep recurrent convolutional neural networks (DeepVO) proposed by Wang et al. [17] implemented an end-to-end visual odometry framework. It extracts visual features from input image pairs using a convolutional neural network (CNN) module. The extracted features are then passed to a recurrent neural network (RNN) module to model the temporal relationship between adjacent image frames. Finally, DeepVO establishes a pose estimation module based on the features output by the RNN module. Due to its easily trainable end-to-end style, DeepVO became the foundation for many subsequent research works on supervised learning-based visual odometry. To further enhance the generalization performance of supervised learning-based visual odometry in complex scenes, Saputra et al. [18] proposed an approach that combines incremental learning with traditional geometric loss constraints. Furthermore, Saputra et al. [19] introduced knowledge distillation into the supervised learning-based visual odometry framework, significantly reducing the number of neural network parameters, making it deployable on mobile devices. Additionally, Xue et al. [20] and others improved the generalization performance by introducing a memory module into the framework to store global information and utilize this information to refine pose estimation results.

Research on self-supervised learning-based visual odometry has recently gained significant attention, primarily because self-supervised frameworks can be trained without the need for large and

expensive annotated datasets. The underlying logic of self-supervised frameworks is to utilize the photometric error between synthetic views and real views as the supervision signal for network training, enabling simultaneous estimation of scene depth information and camera relative pose transformations.

In the early works of deep learning-based monocular visual odometry, Zhou et al. [21] proposed a self-supervised learning framework that learns scene depth information and inter-frame camera pose transformations from monocular videos. In this framework, the pose estimation network, PoseNet, and the scene depth estimation network, DepthNet, are coupled together through the photometric error signal generated by comparing synthetic and real views. Subsequently, Godard et al. [22] introduced the minimum reprojection loss to handle occlusion cases between frames and utilized auto-masking techniques to ignore pixel regions that violate the assumption of a static scene with a moving camera during the training process. In contrast to directly discarding pixels that violate the training assumptions, Vijayanarasimhan et al. [23] and Yin and Shi [24] attempted to explicitly model moving objects in the scene. However, the results showed that these complex and fine-grained approaches did not lead to performance improvements. Jiang et al. [25] carefully considered the photometric error in regions with objects moving in the same or opposite direction as the camera and proposed an outlier-masking technique based on statistical methods to eliminate invisible or unstable pixels. This approach helps to correct misleading error signals during network training. Xu et al. [26] proposed a CNN with dense connectivity, building upon the classical monocular visual odometry system. This network aims to eliminate outliers in the matching results, allowing both local and global information to be utilized for outlier rejection, thereby enhancing the system's performance in localization tasks. Hongru and Xiuquan et al. [27] introduced a novel framework, a graph attention network (GAT)-optimized dynamic monocular visual odometry (GDM-VO), which leverages semantic segmentation and multi-view geometry to remove dynamic objects, and used a GAT to capture long-range temporal dependencies. Chen et al. [28] proposed a multimodal SLAM framework that aggregates the information of the two modalities of RGB and depth to accurately detect and segment salient objects to enhance the ability of extracting key features. Yadav and Kala [29] proposed a fusion of place recognition with visual odometry for SLAM under extreme conditions, eliminating the need for human-level identification of distinct places. Misclassifications are reduced using heuristics, albeit with some loss in re-localizations. The fused technique outperforms SOTA vision-based SLAM and place recognition algorithms.

In current research, scholars are also employing implicit representations based on Neural Radiance Fields (NeRF) for visual SLAM studies. Rosinol et al. [30] proposed a novel pipeline for real-time 3D reconstruction from monocular images, by combining dense monocular SLAM and neural radiance fields. They use the poses, depths, and uncertainties estimated by the SLAM front-end to supervise the training of a hierarchical volumetric neural radiance field. Chung et al. [31] developed Orbeez-SLAM, a real-time monocular visual SLAM system that uses ORB features and NeRF-realized mapping to provide dense maps for spatial AI applications. Furthermore, Liang et al. [32] developed a novel RGB-D SLAM system called dynamic instance segmentation and geometric clustering SLAM (DIG-SLAM), which uses instance segmentation, line feature extraction, and geometric clustering to improve the accuracy and robustness of camera pose estimation and map building in dynamic indoor scenes. It can be concluded from these research efforts that combining NeRF with traditional geometric knowledge in SLAM tasks yields effective results.

Currently, most methods struggle to strike a good balance between the interpretability of traditional approaches and the generalization capabilities of deep learning-based methods. In this paper, we propose a novel monocular visual odometry framework that combines deep learning and geometric feature extraction. Additionally, we design a corresponding pluggable optimizer module for this framework.

3. Method

3.1. Hybrid-VO system based on spatiotemporal features

This paper presents a hybrid monocular visual odometry system that utilizes spatiotemporal features for feature extraction and matching. As shown in Figure 1, the visual odometry system comprises three components: the feature extraction and matching front-end, the model selector module, and the final inter-frame pose estimation module. The hybrid visual odometry algorithm framework proposed in this paper includes a scene depth estimation network and a scene optical flow estimation network as the front end of the feature extraction and matching modules. To balance the real-time operational efficiency of the visual odometry, this paper opts to employ a more lightweight LiteFlowNet [33] as the optical flow estimation input in the hybrid visual odometry algorithm framework. The system selects the minimum N number of matches with the smallest errors based on the principle of consistency between forward and backward optical flow, as shown in Figure 2. The matching results from multiple frames are combined to choose the appropriate matching frame, resulting in both 2D-2D and 3D-2D matching pairs. The model selection module determines whether to use 2D-2D or 3D-2D matching pairs based on whether the forward direction optical flow difference is greater than a certain threshold to the inter-frame pose transformation. When the 2D-2D matching pair is chosen, the scale factor s of the translation vector is estimated separately.

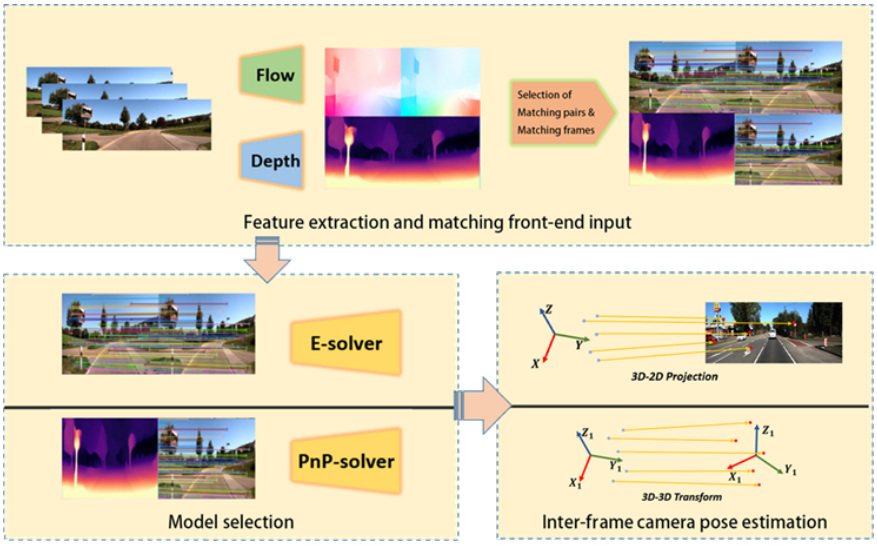


Figure 1. The algorithm process of Hybrid-VO based on spatiotemporal features. The algorithm is composed of three parts: the front end for feature extraction and matching, the 3D-2D/2D-2D matching model selector, and the inter-frame pose estimation part.

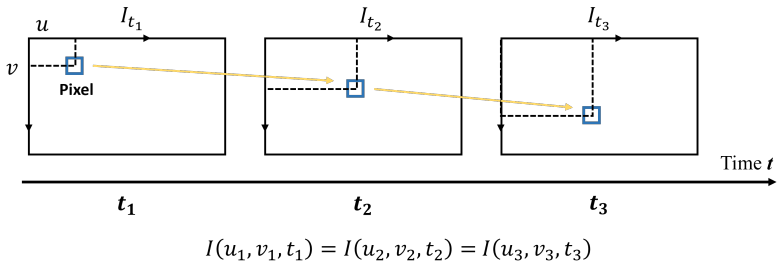


Figure 2. The principle of consistency between forward and backward optical flow. In the optical flow estimation, the position of each pixel in both forward and backward optical flow remains the same.

As shown in Figure 1, the monocular visual odometry algorithm framework proposed in this paper is a hybrid approach. It combines traditional algorithms based on feature tracking and matching and deep learning-based scene optical flow and depth prediction algorithms, giving it the ability to handle 3D information in the stereo space. Thus, it can more confidently choose the actual solution algorithm based on the scene situation. The algorithm process is shown in Figure 1 and the detailed steps can be found in Algorithm 1.

Algorithm 1 Hybrid-VO

Input: Depth-CNN, denoted as M_d. Flow-CNN, denoted as M_f, Image sequence: $[I_1, I_2, \ldots, I_k]$
Output: Camera pose sequence: $[T_1, T_2, \ldots, T_k]$

Initialize: $T_1 = I; i = 2; j \in \{1, 2\}$
while $i \leq k$ **do**
    Get CNN prediction: D_i, F^i_{i-j} and F^{i-j}_i
    Calculate forward and backward optical flow consistency: $|F'| = \left|-F^i_{i-j} - F^{i-j}_i\right|$
    Select N groups of matching pairs $\left(p_i, p_{i-j}\right)$ based on minimum optical flow consistency criteria and determine the matching frame j.
    if avg $(|F'|) > \delta_f$ **then**
        Get the essential matrix E using 2D-2D matching pairs and then obtain $[R, \hat{t}]$.
        Triangulate the matching pair $\left(p_i, p_{i-j}\right)$ to obtain D'_i.
        Compare $\left(D_i, D'_i\right)$ and calculate the scale factor s.
        Calculate $T^{i-j}_i = [R, s\hat{t}]$.
    else
        Obtain 3D-2D matching pairs based on (D_i, F').
        Estimate $[R, t]$ using PnP algorithm.
        Calculate $T^{i-j}_i = [R, t]$.
    end if
    Iterate $T_i \leftarrow T_{i-j} T^{i-j}_i$.
end while

3.1.1. Deep Learning based front-end for feature extraction and matching

The hybrid visual odometry model proposed in this paper includes two deep learning models:

1) M_d is used to estimate the depth of the scene in a single frame.

2) M_f is used to estimate the inter-frame scene optical flow vector.

After obtaining the image pair (I_i, I_{i-1}), the inter-frame scene dense optical flow vectors are estimated by using M_f to obtain 2D-2D matching pairs. However, it should be noted that the estimated dense optical flow vectors corresponding to the whole image may not be entirely accurate. The accuracy of inter-frame pose estimation is directly affected by the accuracy of 2D-2D matching pairs. To eliminate outliers in dense scene optical flow vectors, the proposed hybrid visual odometry in this paper calculates both the forward optical flow F_{i-1}^{i} and the backward optical flow F_{i}^{i-1} simultaneously, and uses the difference between the two, $\left|-F_{i-1}^{i} - F_{i}^{i-1}\right|$, as a measure of optical flow consistency. Based on the criterion of minimum optical flow consistency, N groups of 2D-2D matching pairs are selected to estimate the inter-frame motion pose transformation.

Compared to traditional methods that rely on feature extraction to establish matching pairs, which can only capture limited static rich-texture corner points, using scene flow to obtain matching pairs can maximize the utilization of all possible image pixels. Therefore, it exhibits greater robustness in different scenes. Traditional feature extraction uses the method of image descriptors, which only considers the local image feature patterns around specific pixels. On the other hand, optical flow vectors predicted using deep neural networks can utilize the natural hierarchical architecture of CNN to obtain a larger receptive field. This implies that a greater spatial range of image information around specific pixel locations can be considered, resulting in more accurate prediction of non-outlier optical flow vectors. For the establishment of 3D-2D correspondence pairs, this paper uses the MondepthPlus network [34] for monocular scene depth estimation.

When estimating the pose of the target frame, this paper considers the case of multiple frames in sequence. Based on the minimum optical flow consistency criterion, a new method is proposed to select the matching frame S, as shown in Eq (3.1).

$$s^* = \underset{s \in S}{\operatorname{argmin}} \left|-F_{s}^{t} - F_{t}^{s}\right| \tag{3.1}$$

By using a matching front-end based on deep neural networks to establish inter-frame matching, we can apply appropriate 2D-2D or 3D-2D solving algorithms to compute inter-frame pose transformations. Using the scene depth information provided by the MondepthPlus scene depth estimation network [34], the proposed method in this paper can easily convert 2D-2D matching pairs into 3D-2D matching pairs. In certain scenarios where 2D-2D matching is not suitable, this approach can complement it effectively.

Scene optical flow estimation is a fundamental visual task, and the current SOTA models have demonstrated good scene generalization capabilities. Therefore, in most scenarios, the hybrid visual odometry algorithm proposed in this paper will use 2D-2D matching to solve for the inter-frame essential matrix and recover relative pose transformations. However, as mentioned earlier, there are inherent limitations to the essential matrix solving method based on epipolar geometry. To overcome these shortcomings, this paper utilizes related depth prediction information to assist in alleviating the pain points of small displacement pure rotation scene failure and scale ambiguity of translation components in pose transformation solutions based on 2D-2D matching. The detailed limitations and the solutions are as follows:

1) Scale Ambiguity: The pose translation component t decomposed from the essential matrix E is

normalized and has no absolute scale. In this paper, the scene depth map estimated by the deep neural network is used to calculate its absolute scale factor. Firstly, the pose representation $[R, \hat{t}]$ is decomposed from the essential matrix E, and then the scene point cloud D_i' is obtained by triangulation of the 2D-2D matching pairs $\left(p_i, p_{i-j}\right)$. Finally, the absolute scale factor s can be obtained by comparing $\left(D_i, D_i'\right)$.

2) Rotation motion problem: Pure rotation cases will result in unsolvable rotational components, and small inter-frame displacements will lead to inaccurate results. To overcome these two drawbacks, based on the obtained average optical flow consistency measure index avg $(|F'|)$, the proposed hybrid visual odometry algorithm framework will choose whether to use 2D-2D matching pairs to solve the inter-frame pose transformation or use 3D-2D matching pairs algorithm. At the same time, for the four results obtained by decomposing the essential matrix E, their chirality conditions are checked to select the points that satisfy the triangulation condition and correspond to space points in front of the camera after triangulation, as shown in Figure 3.

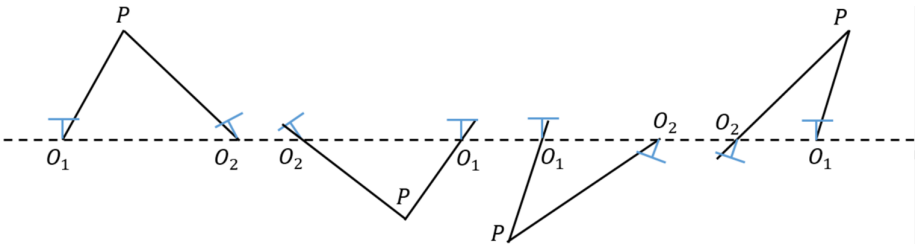


Figure 3. The four results obtained by decomposing the essential matrix E. Only the results that correspond to space points in front of the camera after triangulation satisfy the chirality condition.

3.1.2. Model selector

Traditional visual odometry estimates inter-frame motion primarily based on tracking the inter-frame feature point matching pairs. The matching pairs can be classified into three forms: 3D-2D, 2D-2D, and 3D-3D. This section mainly introduces the solver module proposed in this paper, which adaptively select 3D-2D matching pairs or 2D-2D matching pairs to solve the inter-frame camera pose transformation according to the scene conditions.

For 2D-2D pairs, we use 2D-2D methods based on the principle of epipolar geometry. The method for estimating the relative camera pose transformation between two frames given image pairs (I_1, I_2) is to solve for the essential matrix E corresponding to the two images using the inter-frame matching pairs (p_1, p_2) based on the principle of epipolar geometry. Then, matrix decomposition is used to obtain the rotation component R and the translation component T.

$$p_2^T K^{-T} E K^{-1} p_1 = 0 \text{ where } E = [t]_\times R \tag{3.2}$$

The K in formula 3.2 is the intrinsics of the camera. Generally, the 2D-2D match between I_1 and I_2 is obtained through the feature extraction and tracking module or scene optical flow calculation. However, using the essential matrix E to solve for pose components R and t also poses many problems, such as the following.

1) **Scale Ambiguity**: The estimated pose translation component t, obtained from the essential matrix E, is directionally effective, but the scale is normalized.

2) **Rotation motion problem:**: If there is only rotation motion between inter-frames, then it is not reasonable to obtain the pose rotation component R.

3) **Robustness**: If the translation in inter-frame motion is extremely small, the obtained solution is inaccurate.

There is also the 3D-2D method based on the principle of projective geometry. The Perspective-n-Point (PnP) method is a classic algorithm used to solve the inter-frame motion pose transformation given a 3D-2D matching pair between frames. Considering the given 3D-2D matching pair (X_1, p_2), the inter-frame pose $[R, t]$ solved by the PnP algorithm based on minimizing the reprojection error is optimized, as shown in formula 3.3.

$$e = \sum_i \left\| K\left(RX_{(1,i)} + t\right) - p(2,i) \right\|_2 \tag{3.3}$$

Similarly, the use of PnP algorithms also has certain limitations, such as the following.

1) Obtaining of 3D-2D matching pairs requires 3D information of the scene.

2) Such methods need to obtain correspondences between inter-frame 3D spatial points and 2D pixel points.

The solver module determines the selection of using 2D-2D matching to solve the essential matrix for calculating inter-frame pose transformation, or using 3D-2D matching with the PnP algorithm to solve frame-to-frame pose transformation, based on whether the forward optical flow disparity is greater than a certain threshold.

3.2. Post-optimizer based on Progressive Inter-frame Pose Estimation

In addition to the Hybrid-VO based on the spatiotemporal features method, we also propose a post-odometry optimization module. This post-odometry optimization module uses image synthesis to insert a synthesized new view between the two frames with the unknown pose transformation to be estimated. This method decomposes the pose transformation estimation between the two real frames into multiple intermediate pose transformations that progressively transition from the first frame to the second frame, and then merge these intermediate transformations to form a refined pose transformation estimation between the two frames. This combination of the two methods is called Refined Hybrid-VO.

Unlike Wang et al.'s [35] method of remaking the neural network structure to form a multilevel pose correction prediction and coupling the correction pose in the self-supervised network training, this paper uses a separate module to extract the pose correction function. It does not need to be trained together with the network. As long as there exists the scene depth estimation module and the inter-frame pose estimation module, the rough pose that has been estimated can be corrected afterward to obtain a more accurate pose. The advantage is that it does not need to consider the error propagation of multilevel prediction during network training, which reduces the burden of network training. Moreover, the proposed module is a separate post-refinement module, and this plug-and-play method is more flexible.

The comparative experiments of the proposed progressive inter-frame pose estimation optimizer odometry algorithm with other SOTA odometry algorithms shows that ours odometry algorithms can

refine the initial rough inter-frame pose estimation results, and improve the estimation accuracy and robustness.

The progressive inter-frame pose transformation estimation optimizer proposed in this section is based on the MonodepthPlus [34] and the inter-frame pose transformation estimation algorithm module studied in the previous chapter. The relationship between the three is shown in Figures 3 and 4. When the inter-frame pose estimation module obtains the initial inter-frame pose transformation estimation, it is combined with the scene depth estimation module to perform 3D-2D backprojection, resampling to obtain the synthesized intermediate view, and then the residual transformation is estimated using the synthesized intermediate view. Finally, the initial estimation and several residual estimations are merged to obtain the refined inter-frame pose transformation estimation.

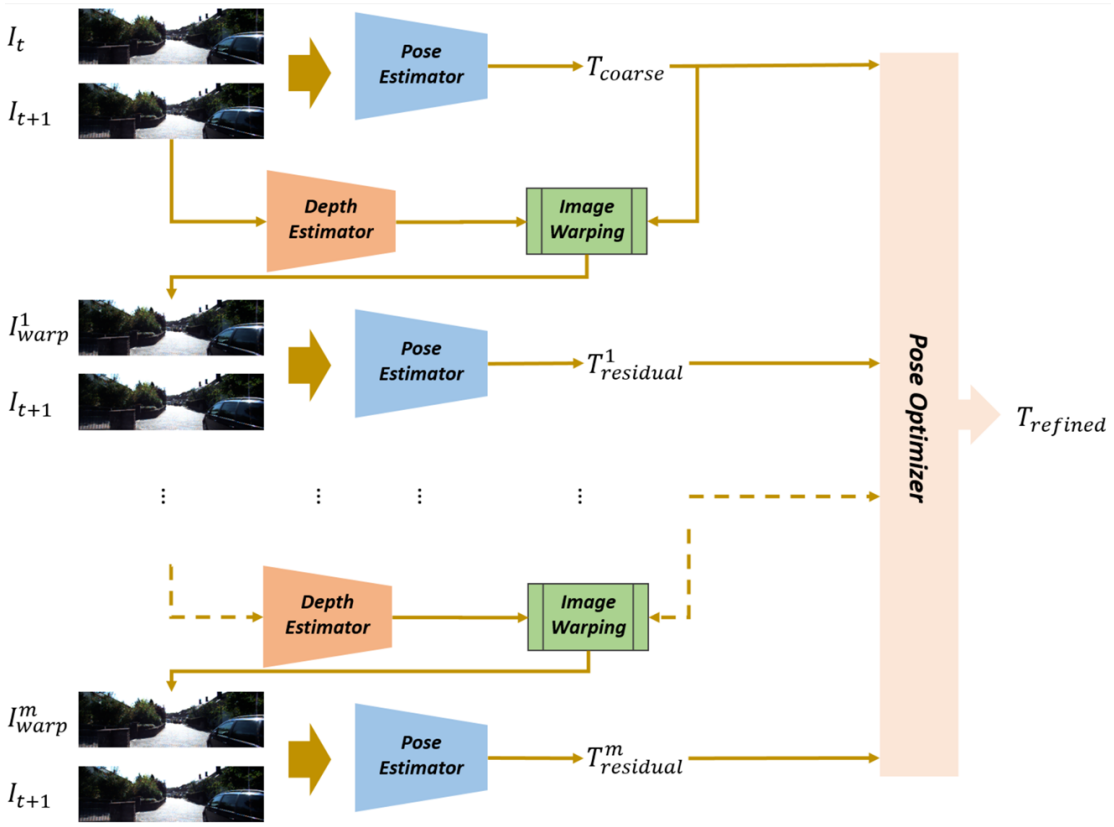


Figure 4. Progressive inter-frame pose transformation optimization diagram.

The optimizer utilizes image synthesis techniques to insert synthetic frames between the inter-frames undergoing pose estimation.

Given the following input: two image frames I_t and I_{t+1}, the scene depth map D_{t+1} estimated by the Depth Estimator module for I_{t+1}, and the inter-frame pose transformation T_{coarse} from I_{t+1} to I_t estimated by the Pose Estimator module, a virtual frame I_{warp} can be synthesized between frames I_t and I_{t+1}. The reason for this being able to synthesize an intermediate view is based on the assumption: The inter-frame pose transformation estimated by the Pose Estimator module is not very accurate and there is an error from the ground truth, as shown in Figure 5.

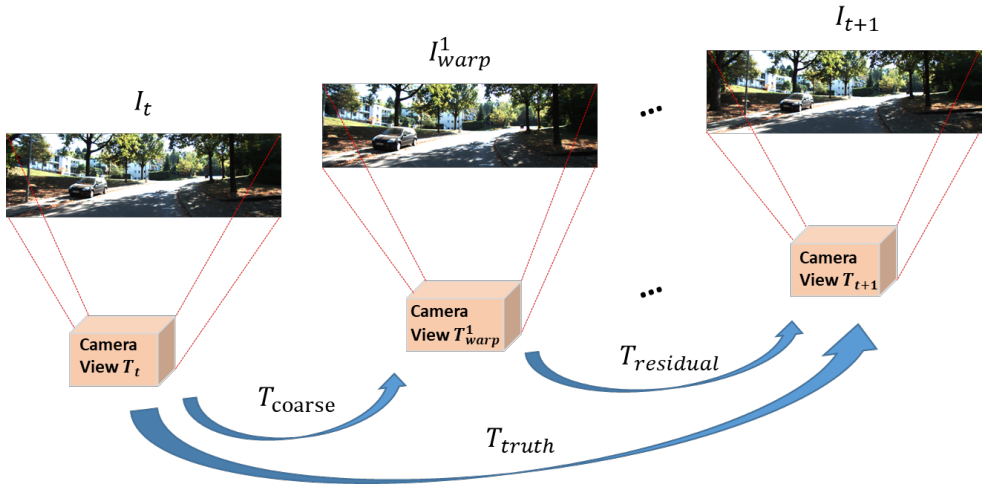


Figure 5. The Pose Estimator can only estimate T_{coarse}, which means that the estimated pose transformation is not accurate.

The synthesis of the intermediate frame first requires the generation of sampling points on frame I_t, as calculated by Eq (3.4).

$$\begin{aligned} d_{\text{proj}} \times \left[u_{\text{proj}}, v_{\text{proj}}, 1\right] \\ = KT_{\text{corase}} \left(D^{(u,v)}_{t+1} K^{-1} \times [u_{t+1}, v_{t+1}, 1]^T\right) \end{aligned} \tag{3.4}$$

In the equation, $D^{(u,v)}_{t+1}$ represents the depth at the position (u_{t+1}, v_{t+1}) on the image frame I_{t+1}. The parameter d_{proj} denotes the normalized depth information required for 3D-2D projection, which is converted to 2D homogeneous coordinates. The coordinates $\left(u_{\text{proj}}, v_{\text{proj}}\right)$ correspond to the continuous coordinates on the image frame I_t. However, since the 2D pixel plane is a discrete coordinate plane, bilinear interpolation is required to synthesize the pixel values at the $\left(u_{\text{proj}}, v_{\text{proj}}\right)$ position, as shown in Figure 6.

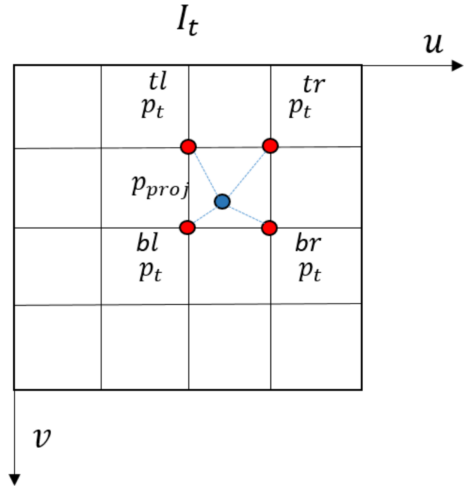


Figure 6. Bilinear interpolation.

The mathematical expression for bilinear interpolation is shown in Eq (3.5).

$$I^1_{\text{warp}}\left(u_{\text{proj}}, v_{\text{proj}}\right) = \sum_{v\in\{t,b\},u\in\{l,r\}} \omega_{u,v} I_t(u,v) \tag{3.5}$$

$$T_{\text{refined}} = T_{\text{coarse}}\, T^1_{\text{residual}} \ldots T^m_{\text{residual}} \tag{3.6}$$

Although a hybrid visual odometry method that combines traditional geometric principles and deep learning to estimate inter-frame pose transformations has been proposed in this paper, errors can still exist between the initial inter-frame pose transformation estimation and the actual pose transformation, regardless of how accurate the initial estimation is. Therefore, this chapter proposes a method for estimating errors by synthesizing intermediate frames. This method decomposes the inter-frame pose transformation between two real frames into an initial rough estimation and a series of error estimations, as depicted in Figure 4. The detailed process is explained in Algorithm 2, and the decomposition relationship is shown in Eq (3.6).

Algorithm 2 Progressive inter-frame pose transformation optimization algorithm

Input: Depth-Estimator, denoted as E_{depth}; Pose-Estimator, denoted as E_{pos}; Image sequence: $[I_t, I_{t+1}]$
Output: Refined inter-frame pose transformation, $T_{refined}$

1: Use E_{pose} to estimate the initial rough pose transformation T_{corase} between frames (I_t, I_{t+1}).
2: To simplify the explanation, let T_{coarse} denote T^0_{residual}, and initialize $i = 1$.
3: **while** $i \le m$ **do**
4: Obtain E_{depth} to predict: D_{t+1}.
5: Synthesize intermediate view I^i_{warp} using $T^{i-1}_{\text{residual}}$ and D_{t+1}.
6: Send frames I^i_{warp} and I_{t+1} into E_{pose}, and estimate to obtain T^i_{residual}.
7: **end while**
8: Combining all polynomials: $T_{\text{refined}} = T_{\text{coarse}} \ldots T^m_{\text{residual}}$.

When actually using a progressive inter-frame pose transformation estimation optimizer, m is adjustable as a hyperparameter. There are two ways to adjust it:

1) Invariant m: Select an invariant for m based on experience, such as 2 or 3, indicating how many terms are included in the residual transformation part.

2) Adaptive m: Combined with the hybrid visual odometry algorithm, the forward-backward optical flow consistency index $|F'|$ can be calculated by I^m_{warp} and I_{t+1}. Set a threshold δ_r. If the calculated avg $(|F'|)$ is greater than the threshold δ_r, continue to decompose the residual. Otherwise, stop and regard until the accuracy reaches the required precision.

4. Experiments and discussion

This section compares the Hybrid-VO based on spatiotemporal features proposed in this paper with other SOTA methods. And Comparative and ablation experiments are carried out on the proposed post-optimizer. Finally, the performance and the effectiveness of the proposed method is discussed.

4.1. Test datasets and evaluation metrics

4.1.1. Dataset

In this paper, we use KITTI Eigen Split [36] , which contains 39,810 monocular triplets for training and 4424 for validation. It is a subset of the KITTI dataset, consisting of 22 image sequences captured during outdoor driving scene. Only the first 11 image sequences (00–10) provide ground-truth trajectory for the vehicle, hence this study evaluates algorithms using only these sequences.

4.1.2. Evaluation metric

When evaluating the accuracy of a visual odometry algorithm, two commonly used evaluation metrics are translation error (t_{err}), rotation error (r_{err}), absolute trajectory error (ATE), and relative pose error (RPE).

The t_{err} quantifies the discrepancy between the actual translation (movement in space), and the r_{err} measures the difference between the actual rotation applied by an algorithm and the ground truth rotation.

The RPE mainly represents the accuracy of the pose difference between two frames with a given time interval Δ, which is equivalent to directly measuring the error of the odometer.

The definition of RPE for the i-th frame is:

$$E_i = \left(Q_i^{-1} Q_{i+\Delta}\right)^{-1} \left(P_i^{-1} P_{i+\Delta}\right) \tag{4.1}$$

In the case of known total number n and interval Δ, m = n - Δ can be obtained, and the root mean square error $RMSE$ is generally used to statistically measure the overall error of the sequence:

$$RMSE\left(E_{1:n}, \Delta\right) = \left(\frac{1}{m} \sum_{i=1}^{m} \left\| trans\left(E_i\right)\right\|^2\right)^{\frac{1}{2}} \tag{4.2}$$

where trans (E_i) represents the translation component part of RPE. The performance of the algorithm can be evaluated based on the $RMSE$ value. In order to comprehensively measure the performance of the algorithm, the average value of $RMSE$ can be obtained by traversing all possible $\triangle$:

$$RMSE\left(E_{1:n}\right) = \frac{1}{n} \sum_{\Delta=1}^{n} RMSE\left(E_{1:n}, \Delta\right) \tag{4.3}$$

The ATE is a direct measure of the difference between the estimated pose and the ground truth in SLAM algorithms. The definition of ATE for the i-th frame is

$$F_i = Q_i^{-1} S P_i \tag{4.4}$$

Similar to RPE, this paper uses $RMSE$ to measure the overall ATE of the sequence:

$$RMSE\left(F_{1:n}, \Delta\right) = \left(\frac{1}{m} \sum_{i=1}^{m} \left\| trans\left(F_i\right)\right\|^2\right)^{\frac{1}{2}} \tag{4.5}$$

It should be noted that ATE only includes translational errors.

4.2. *Hybrid-VO based on spatiotemporal features*

This section compares the hybrid visual odometry algorithm proposed in this paper with several other visual odometry algorithms. For fairness, all algorithms are trained and tested on the KITTI dataset. The training set uses image sequences 00–08 and the test set uses image sequences 09 and 10. The visual odometry algorithms compared include traditional methods such as VISO2 [37] and ORB-SLAM2 [38], as well as deep learning-based methods such as Structure-from-Motion Learner (SfM-Learner) [21], Self-Consistent Structure-from-Motion Learner (SC-SfMLearner) [39] and Depth-VO-Feat [40].

We use the PyTorch [41] framework to implement our network, trained for 20 epochs using Adam Optimizer [42], with a batch size of 4 and an input/output resolution of 640×192. We use a learning rate of 10^{-4} for the first 15 epochs, which is then dropped to 10^{-45} for the remainder. For hyperparameters: $\left[\lambda_{fs}, \lambda_{fc}\right] = \left[10^{-1}, 5 \times 10^{-3}\right]$. Our model took 30 h to train on the RTX 2080Ti.

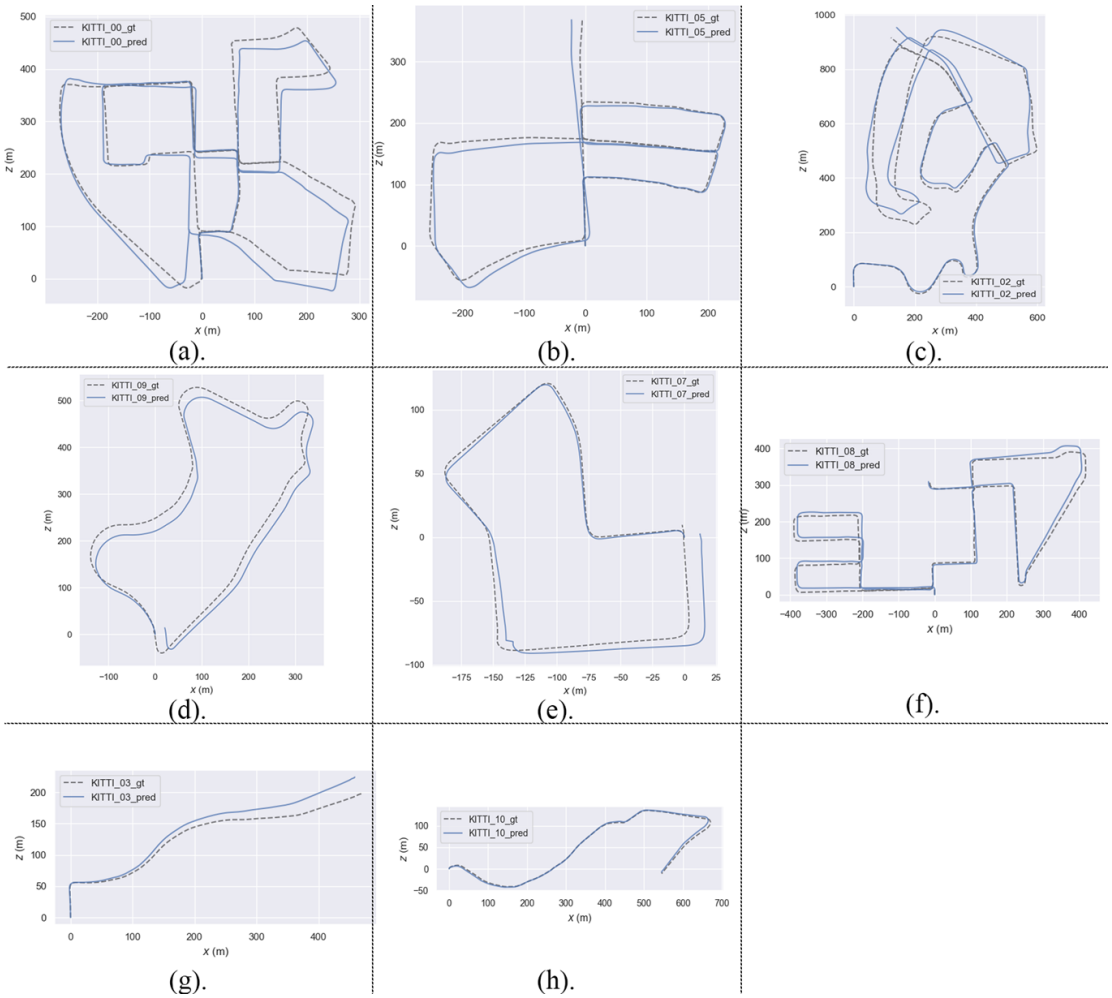


Figure 7. Partial trajectory diagram estimated by the algorithm. The correspondence between the subgraphs and image sequences is: (a)–00; (b)–05; (c)–02; (d)–09; (e)–07; (f)–08; (g)–03; (h)–10.

The qualitative experimental results of the visual odometry algorithm proposed in this paper are shown in Figure 7. It can be seen that the motion trajectory estimated by the hybrid visual odometry algorithm proposed in this paper is very close to the ground truth motion trajectory.

In addition, this paper also shows the deviation of the translational components on the XYZ axes of the estimated trajectory, as shown in Figure 8.

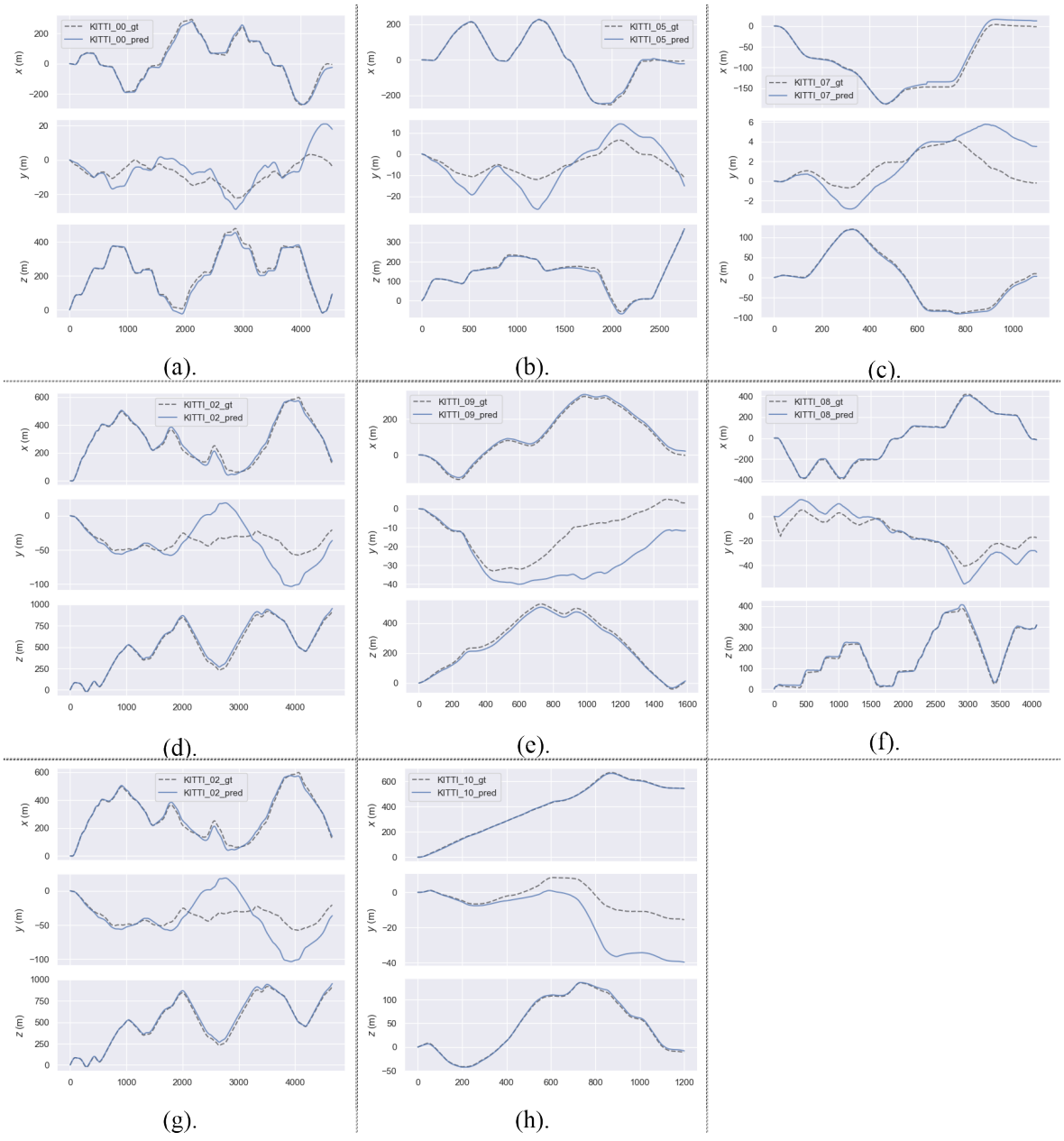


Figure 8. Partial deviation of the trajectory estimated by the algorithm on the XYZ axes. The correspondence between the subgraphs and the image sequence is: (a)−00; (b)−05; (c)−02; (d)−09; (e)−07; (f)−08; (g)−03; (h)−10.

The deviation of each component of the Euler angles (roll, pitch, yaw) of the estimated trajectory is also shown in Figure 9.

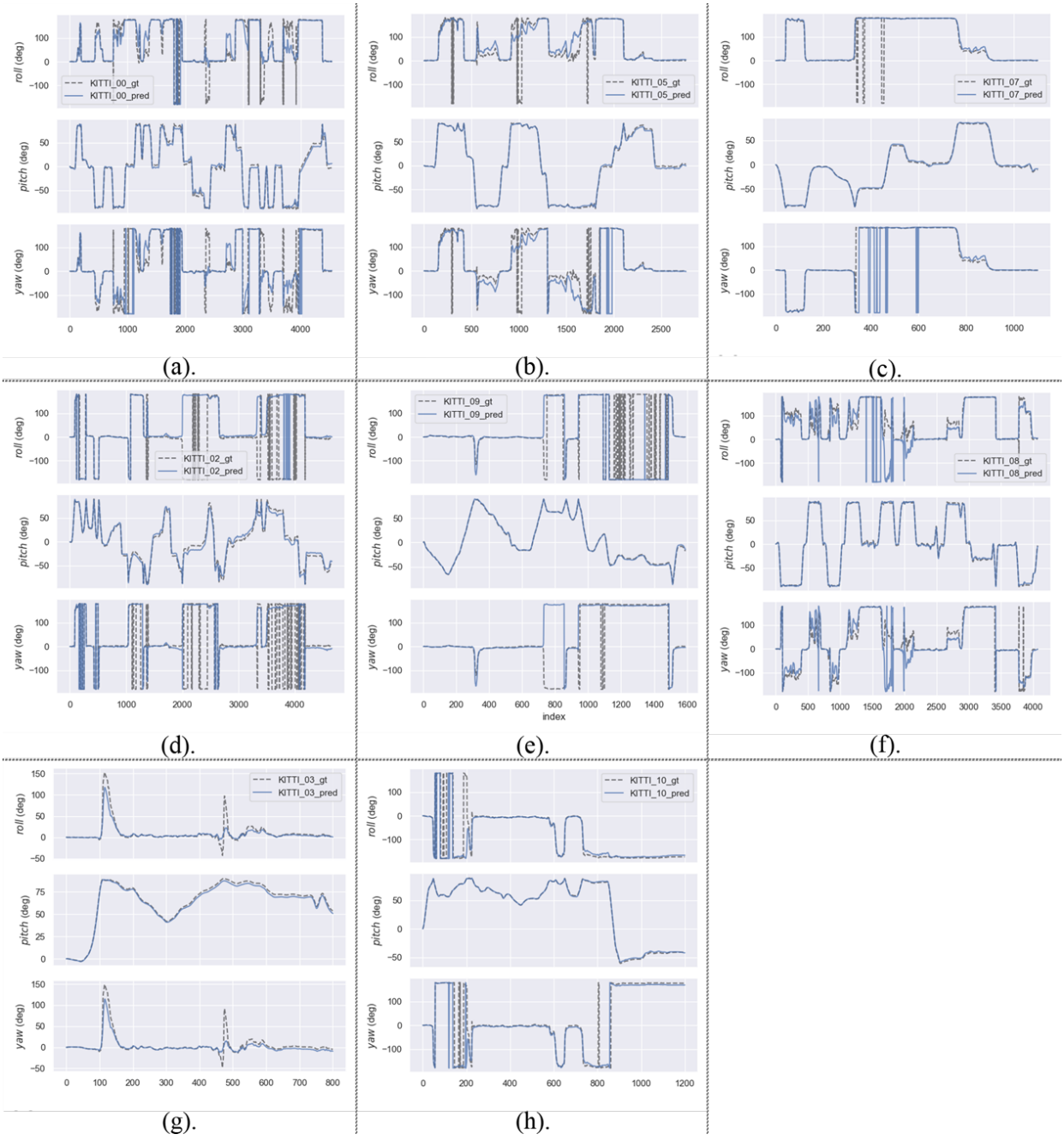


Figure 9. The deviation of the XYZ axes of the trajectory estimated by the algorithm (partial). The correspondence between the subgraphs and the image sequence is: (a)−00; (b)−05; (c)−02; (d)−09; (e)−07; (f)−08; (g)−03; (h)−10.

For Figure 8, by analyzing the individual components of the estimated trajectory and the ground truth, it can be found that the error of the estimated trajectory on the X-Z plane is very small. The

main error is mainly on the Y-axis component, which is the up and down direction during the motion of the vehicle. Its changes are mainly affected by terrain ups and downs. The scene changes caused by motion are often changes in sky or ground boundaries. These texture-less area changes are not easily detected by algorithms, resulting in cumulative errors.

For Figure 9, it can be seen that apart from the pitch angle, the errors of the roll and yaw angles during the motion of the vehicle are quite high. In this paper, the pitch angle is defined as the rotation angle around the Y-axis and is actually related to the slope of the ground while the vehicle is moving. Since most of the scenes involved in the KITTI dataset used in this paper are highway scenes, the road slope is small, so the error is small. The roll and yaw angles are rotation angles around the X-axis and Z-axis, respectively. The plane where the vehicle travels on is the X-Z plane. Therefore, these two angles have a large prediction range and high uncertainty resulting in accumulation of errors.

Table 1 presents the quantitative evaluation metrics for the proposed hybrid visual odometry algorithm. Table 2 shows the quantitative comparisons of the proposed method with traditional and deep learning methods. The texture denotes the percentage of translational error, and r1000 represents the average rotational error per 100. The evaluation tool program is applied to sub-trajectories of lengths (100, 200,..., 800) m, and the overall average values are used as final results. The results indicate that the proposed algorithm outperforms other methods in most of the mean values of the metrics. Similarly, the hybrid algorithm also shows better performance in most of the evaluation metrics compared to other methods on test sequences 09 and 10. The superiority of the proposed algorithm in these quantitative metrics demonstrates the effectiveness of this type of algorithm framework that combines traditional methods and deep learning methods.

Table 1. The algorithm's quantitative evaluation results (sequences 09, 10).

Method	Metric	09	10	Avg.Err.
Ours	t_{err} (%)	2.49	1.97	2.211
	r_{err} (°/100m)	0.33	0.35	0.418
	ATE	12.23	3.48	7.883
	RPE (m)	0.06	0.049	0.071
	RPE (∘)	0.041	0.051	0.073

The qualitative results displayed above, along with the quantitative metrics, fully shows the successful integration of traditional visual odometry algorithms and deep learning-based visual odometry methods in the proposed hybrid visual odometry framework, highlighting the advantages of each approach and enhancing the overall algorithm accuracy. This once again confirms that the next breakthrough in improving visual odometry performance lies in the integration of traditional principles with deep learning techniques. The comparison of test results in Table 2 with the relatively recent studies by Zhao et al. [43] and Zou et al. [44] against the hybrid visual odometry algorithm proposed in this paper, conducted on the same dataset and scenario, reveals that the key performance metrics of the proposed hybrid algorithm, including the t_{err} and the r_{err}, remain superior to these relatively newer deep learning-based methods.

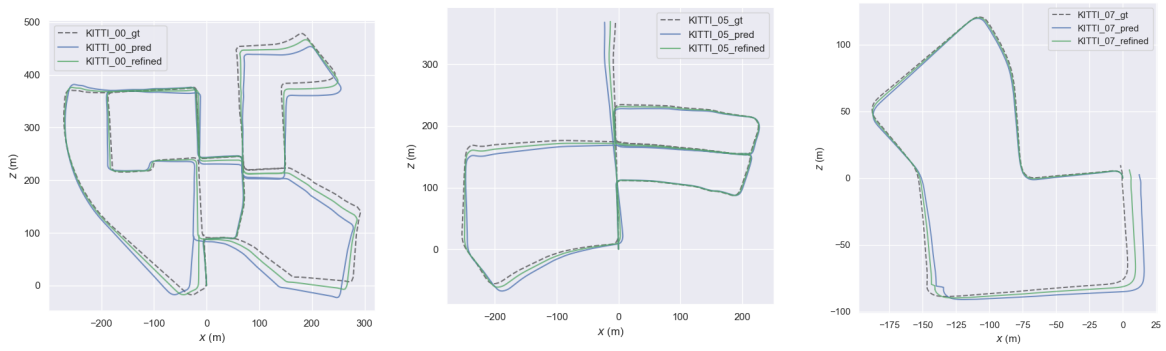
Table 2. The algorithm's quantitative evaluation results (sequences 09, 10). The best result is shown in bold; the second best result is underlined.

Method	Metric	09	10	Method	Metric	09	10
SfM-Learner [23]	t_{err} (%)	11.32	15.25	ORB-SLAM2 [38] (w/o LC)	t_{err} (%)	9.3	<u>2.57</u>
	r_{err} (°/100m)	4.07	4.06		r_{err} (°/100m)	<u>0.26</u>	<u>0.32</u>
	ATE	26.93	24.09		ATE	38.77	<u>5.42</u>
	RPE (m)	0.159	0.171		RPE (m)	<u>0.128</u>	**0.045**
	RPE (∘)	0.159	0.171		RPE (∘)	<u>0.061</u>	<u>0.065</u>
Depth-VO-Feat [40]	t_{err} (%)	11.89	12.82	ORB-SLAM2 [38] (w/ LC)	t_{err} (%)	<u>2.88</u>	3.3
	r_{err} (°/100m)	3.6	3.41		r_{err} (°/100m)	**0.25**	**0.3**
	ATE	52.12	24.7		ATE	**8.39**	6.63
	RPE (m)	0.164	0.159		RPE (m)	0.343	<u>0.047</u>
	RPE (∘)	0.233	0.246		RPE (∘)	0.063	0.066
zou et al. 2020 [44]	t_{err} (%)	3.49	5.81	Ours Hybrid-VO	t_{err} (%)	**2.49**	**1.97**
	r_{err} (°/100m)	1.00	1.8		r_{err} (°/100m)	0.33	0.35
zhao et al. 2020 [43]	t_{err} (%)	7.21	11.43		ATE	<u>12.23</u>	**3.48**
	r_{err} (°/100m)	0.56	2.57		RPE (m)	**0.06**	0.049
					RPE (∘)	**0.041**	**0.051**

4.3. Post-optimizer based on Progressive Inter-frame Pose Estimation

In order to verify the effectiveness of the post-optimizer based on progressive inter-frame pose estimation proposed, this section conducted both quantitative and qualitative experiments using methods similar to those in Section 4.2. All algorithms were trained and tested on the KITTI odometry dataset, using the hybrid visual odometry algorithm proposed in this paper as well as the post-optimizer based on progressive inter-frame pose estimation proposed in Section 3.2. The training image sequences contain 00–08, while the test image sequences contain 09 and 10. The qualitative experimental results of the progressive pose correction module proposed in this section, in combination with the hybrid visual odometry algorithm proposed in this paper, as well as the post-optimizer based on progressive inter-frame pose estimation proposed in Section 3.2, are shown in Figure 10. We use the PyTorch [41] framework to implement our network. For hyperparameters: Posture correction iterations $m = 2$.

The quantitative experimental results of the hybrid self-supervised monocular visual odometry based on spatiotemporal features (Refined Hybrid-VO) proposed in Section 3.1 with the post-optimizer based on progressive inter-frame pose estimation proposed in Section 3.2, which we call this combination of the two methods Refined-Hybrid-VO, are shown in Table 3. In the tables, t_{err} (%) is the percentage metric of translation error, and r_{err} (∘/100 m) is the average rotational error per 100 m. The evaluation process would calculate these metrics on the (100, 200, …, 800) m sub-trajectories length and take the average as the final result. Quantitative experimental results show that the proposed post-optimizer based on progressive inter-frame pose estimation module in Section 4.2 can indeed further improve the accuracy of inter-frame pose estimation.



(a) **(b)** **(c)**

Figure 10. Corrected estimated trajectory map (part of the test set).

Table 3. The algorithm's quantitative evaluation results (sequences 09, 10). The best result is shown in bold; the second best result is underlined.

Method	Metric	09	10	Method	Metric	09	10
Depth-VO-Feat [40]	t_{err} (%)	11.89	12.82	ORB-SLAM2 [38] (w/ LC)	t_{err} (%)	2.88	3.3
	r_{err} (°/100m)	3.6	3.41		r_{err} (°/100m)	**0.25**	**0.3**
	ATE	52.12	24.7		ATE	**8.39**	6.63
	RPE (m)	0.164	0.159		RPE (m)	0.343	0.047
	RPE (∘)	0.233	0.246		RPE (∘)	0.063	0.066
ORB-SLAM2 [38] (w/o LC)	t_{err} (%)	9.3	2.57	Ours Hybrid-VO	t_{err} (%)	2.49	1.97
	r_{err} (°/100m)	0.26	0.32		r_{err} (°/100m)	0.33	0.35
	ATE	38.77	5.42		ATE	12.23	3.48
	RPE (m)	0.128	**0.045**		RPE (m)	0.06	0.049
	RPE (∘)	0.061	0.065		RPE (∘)	**0.041**	0.051
zou et al. [44] 2020	t_{err} (%)	3.49	5.81	Ours Refined-Hybrid-VO	t_{err} (%)	**2.36**	**1.67**
	r_{err} (°/100m)	1.00	1.80		r_{err} (°/100m)	0.3	0.32
zhao et al. [43] 2020	t_{err} (%)	7.21	11.43		ATE	9.68	**3.16**
	r_{err} (°/100m)	0.56	2.57		RPE (m)	**0.053**	0.047
					RPE (∘)	**0.041**	**0.047**

4.4. Ablation study

To validate the effectiveness of the model selector module proposed in this paper, we executed a series of comparative ablation studies. The baseline for these studies is the odometry backbone as proposed by ORB-SLAM2 [38]. In the ablation experiments, we substituted the baseline's 2D-2D matching front-end with the MonodepthPlus 3D-2D matching front-end alongside our proposed model selector module. The image sequences tested were sourced from the KITTI dataset, specifically

sequences 00 through 10. The results of the ablation experiments are displayed in Tables 4 and 5. The best result is shown in bold; the second best result is underlined.

Table 4. The algorithms' ablation experiments results (sequences 00–04). The best result is shown in bold; the second best result is underlined.

Method	Metric	00	01	02	03	04	Avg.Err.
Baseline (2D-2D Matching Front-end)	t_{err} (%)	11.43	107.57	10.34	0.97	1.3	8.074
	r_{err} (°/100m)	0.58	0.89	0.26	0.19	0.27	**0.304**
	ATE	40.65	502.20	47.82	0.94	1.30	26.480
	RPE (m)	0.169	2.970	0.172	0.031	0.078	0.130
	RPE (°)	0.079	0.098	0.072	0.055	0.079	**0.063**
3D-2D Matching Front-end	t_{err} (%)	4.37	68	5.78	4.24	1.25	<u>3.73</u>
	r_{err} (°/100m)	0.74	17.08	0.68	0.72	0.39	<u>0.380</u>
	ATE	29.08	624.39	43.09	1.43	1.48	<u>12.748</u>
	RPE (m)	0.128	1.250	0.089	0.055	0.063	<u>0.083</u>
	RPE (°)	0.069	0.894	0.071	0.048	0.065	0.078
Model Selector Module (2D-2D+3D-2D)	t_{err} (%)	2.45	70.98	3.60	3.69	1.47	**2.211**
	r_{err} (°/100m)	0.68	19.04	0.58	0.56	0.34	0.418
	ATE	16.36	630.75	24.45	1.34	1.39	**7.883**
	RPE (m)	0.078	1.340	0.065	0.043	0.059	**0.071**
	RPE (°)	0.067	0.988	0.049	0.042	0.045	<u>0.073</u>

The results from the ablation experiments show a significant improvement when incorporating the model selector module compared to both the baseline and the 3D-2D matching front-end. The average error (Avg.Err.) across most of the metrics for the model selector module (2D-2D+3D-2D) outperforms both the standalone 3D-2D matching front-end and the baseline 2D-2D matching front-end. Specifically, the model selector module exhibits a substantial reduction in the average t_{err} from 8.074 in the baseline to 2.211 across sequences 00–04, with similar improvements observed in sequences 05–10. Although the r_{err} with the model selector module is marginally higher than the baseline, this is a negligible trade-off considering the significant gains in other metrics. ATE and RPE have also been markedly reduced with the introduction of the model selector module, further validating its efficacy in enhancing the precision of visual odometry.

In summary, the model selector module has significantly improved the performance of visual odometry, especially in terms of translational error and absolute trajectory error, confirming the effectiveness of our proposed module in enhancing the robustness and accuracy of visual odometry systems.

Table 5. The algorithms' ablation experiments results (sequences 05–10). The best result is shown in bold; the second best result is underlined.

Method	Metric	05	06	07	08	09	10	Avg.Err.
Baseline (2D-2D Matching Front-end)	t_{err} (%)	9.04	14.56	9.77	11.46	9.30	2.57	8.074
	r_{err} (°/100m)	0.26	0.26	0.36	0.28	0.26	0.32	**0.304**
	ATE	29.95	40.82	16.04	43.09	38.77	5.42	26.480
	RPE (m)	0.140	0.237	0.105	0.192	0.128	0.045	0.130
	RPE (∘)	0.058	0.055	0.047	0.061	0.061	0.065	**0.063**
3D-2D Matching Front-end	t_{err} (%)	3.41	2.53	5.54	7.35	5.62	2.35	<u>3.73</u>
	r_{err} (°/100m)	0.42	0.43	0.37	0.49	0.39	0.37	<u>0.380</u>
	ATE	8.41	12.53	2.16	5.45	19.3	3.95	<u>12.748</u>
	RPE (m)	0.086	0.154	0.083	0.062	0.083	0.057	<u>0.083</u>
	RPE (∘)	0.052	0.051	0.072	0.051	0.063	0.055	0.078
Model Selector Module (2D-2D+3D-2D)	t_{err} (%)	1.23	1.34	0.98	2.45	2.49	1.97	**2.211**
	r_{err} (°/100m)	0.39	0.31	0.34	0.41	0.33	0.35	0.418
	ATE	3.98	2.78	2.21	7.98	12.23	3.48	**7.883**
	RPE (m)	0.030	0.040	0.034	0.056	0.060	0.049	**0.071**
	RPE (∘)	0.039	0.045	0.045	0.043	0.041	0.051	<u>0.073</u>

4.5. Limitation and challenges

The dataset utilized in this article is the KITTI dataset, which comprises real image data collected in various scenarios such as urban areas, rural settings, and highways. The scenes are predominantly outdoor and feature either clear or cloudy weather conditions. In future work, it may be beneficial to consider incorporating a broader range of weather conditions as well as indoor scenarios for research purposes.

In the current self-supervised monocular visual odometry, the task is built upon the subtask of view synthesis. Although the task itself contains strong geometric clues, that is, the view synthesis supported by the principles of multi-view geometry will only be consistent with the real view when the intermediate composites, such as the scene depth map and the inter-frame pose transformation estimation, are accurate. However, there are also phenomena of multi-shooting, that is, sometimes inaccurate inter-frame pose transformation estimation can lead to synthesized frames consistent with the real target frame in some specific scenes (such as texture-less white walls, empty corridors, etc.). Therefore, it is necessary to explore more geometric clue constraints, and when necessary, to redesign a brand new, more robust self-supervised training scheme.

In summary, the Hybrid-VO method with post-optimizer module proposed in this paper can be flexibly combined with existing visual odometer algorithms to obtain high-precision inter-frame pose estimation results.

5. Conclusions

In this paper, a novel method, Hybrid-VO, based on the spatiotemporal features method is proposed, which combined the advantages of the features extraction ability of deep learning and the interpretability of the traditional methods. Then, we propose a post-optimizer based on progressive inter-frame pose estimation method as a plug-and-play module to improve the Hybrid-VO. Compared with existing methods, the present Hybrid-VO and Refined-Hybrid-VO achieve the optimal effect.

However, there is still room for improvement in the research content of this paper, and related issues are worth further investigation and discussion. This article mainly focuses on the application of monocular visual odometry in the context of autonomous driving, so the KITTI dataset is selected. Nowadays, there are also many publicly available indoor visual odometry datasets, and future research can consider both indoor and outdoor scenes to compare and examine the differences and commonalities between them.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Conflict of interest

The authors declare there is no conflicts of interest.

References

1. J. J. Leonard, H. F. Durrant-Whyte, Mobile robot localization by tracking geometric beacons, *IEEE Trans. Rob. Autom.*, **7** (1991), 376–382. https://doi.org/10.1109/70.88147
2. J. Liu, M. Zeng, Y. Wang, W. Liu, Visual SLAM technology based on weakly supervised semantic segmentation in dynamic environment, in *International Symposium on Artificial Intelligence and Robotics 2020*, **11574** (2020). https://doi.org/10.1117/12.2580074
3. J. Fuentes-Pacheco, J. Ruiz-Ascencio, J. M. Rendon-Mancha, Visual simultaneous localization and mapping: A survey, *Artif. Intell. Rev.*, **43** (2015), 55–81. https://doi.org/10.1007/s10462-012-9365-8
4. A. Li, J. Wang, M. Xu, Z. Chen, DP-SLAM: A visual SLAM with moving probability towards dynamic environments, *Inf. Sci.*, **556** (2021), 128–142. https://doi.org/10.1016/j.ins.2020.12.019
5. A. Geiger, P. Lenz, R. Urtasun, Are we ready for autonomous driving? The KITTI vision benchmark suite, in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, (2012), 3354–3361. https://doi.org/10.1109/CVPR.2012.6248074
6. C. Zach, T. Pock, H. Bischof, A duality based approach for realtime $TV - L^1$ optical flow, *Pattern Recognit.*, **4713** (2007), 214–223. https://doi.org/10.1007/978-3-540-74936-3_22
7. D. G. Lowe, Distinctive image features from scale-invariant keypoints, *Int. J. Comput. Vision* , **60** (2004), 91–110. https://doi.org/10.1023/B:VISI.0000029664.99615.94

8. H. Bay, T. Tuytelaars, L. Van Gool, SURF: Speeded up robust features, in *Computer Vision-ECCV 2006*, **3951** (2006), 404–417. https://doi.org/10.1007/11744023_32
9. E. Rublee, V. Rabaud, K. Konolige, G. Bradski, ORB: An efficient alternative to SIFT or SURF, in *2011 International Conference on Computer Vision*, (2011), 2564–2571. https://doi.org/10.1109/ICCV.2011.6126544
10. G. Klein, D. Murray, Parallel tracking and mapping for small AR workspaces, in *2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality*, (2007), 225–234. https://doi.org/10.1109/ISMAR.2007.4538852
11. J. Engel, T. Schoeps, D. Cremers, LSD-SLAM: Large-scale direct monocular SLAM, **8690** (2014), 834–849. https://doi.org/10.1007/978-3-319-10605-2_54
12. R. Mur-Artal, J. M. M. Montiel, J. D. Tardós, ORB-SLAM: a versatile and accurate monocular SLAM system, *IEEE Trans. Rob.*, **31** (2015), 1147–1163. https://doi.org/10.1109/TRO.2015.2463671
13. B. M. Nordfeldt-Fiol, F. Bonin-Font, G. Oliver, Evolving real-time stereo odometry for auv navigation in challenging marine environments, *J. Intell. Rob. Syst.*, **108** (2023). https://doi.org/10.1007/s10846-023-01932-0
14. M. Birem, R. Kleihorst, N. El-Ghouti, Visual odometry based on the fourier transform using a monocular ground-facing camera, *J. Real-Time Image Process.*, **14** (2018), 637–646. https://doi.org/10.1007/s11554-017-0706-3
15. L. De-Maeztu, U. Elordi, M. Nieto, J. Barandiaran, O. Otaegui, A temporally consistent grid-based visual odometry framework for multi-core architectures, *J. Real-Time Image Process.*, **10** (2015), 759–769. https://doi.org/10.1007/s11554-014-0425-y
16. G. Costante, M. Mancini, P. Valigi, T. A. Ciarfuglia, Exploring representation learning with CNNs for frame-to-frame ego-motion estimation, *IEEE Rob. Autom. Lett.*, **1** (2016), 18–25. https://doi.org/10.1109/LRA.2015.2505717
17. S. Wang, R. Clark, H. Wen, N. Trigoni, DeepVO: Towards end-to-end visual odometry with deep recurrent convolutional neural networks, in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, (2017), 2043–2050. https://doi.org/10.1109/ICRA.2017.7989236
18. M. R. U. Saputra, P. P. B. de Gusmao, S. Wang, A. Markham, N. Trigoni, Learning monocular visual odometry through geometry-aware curriculum learning, in *2019 International Conference on Robotics and Automation (ICRA)*, (2019), 3549–3555. https://doi.org/10.1109/ICRA.2019.8793581
19. M. R. U. Saputra, P. Gusmao, Y. Almalioglu, A. Markham, N. Trigoni, Distilling knowledge from a deep pose regressor network, in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, (2019), 263–272. https://doi.org/10.1109/ICCV.2019.00035
20. F. Xue, X. Wang, S. Li, Q. Wang, J. Wang, H. Zha, Beyond tracking: Selecting memory and refining poses for deep visual odometry, in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, (2019), 8567–8575. https://doi.org/10.1109/CVPR.2019.00877

21. T. Zhou, M. Brown, N. Snavely, D. G. Lowe, Unsupervised learning of depth and ego-motion from video, in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (2017), 6612–6619. https://doi.org/10.1109/CVPR.2017.700

22. C. Godard, O. M. Aodha, M. Firman, G. Brostow, Digging into self-supervised monocular depth estimation, in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, (2019), 3827–3837. https://doi.org/10.1109/ICCV.2019.00393

23. S. Vijayanarasimhan, S. Ricco, C. Schmid, R. Sukthankar, K. Fragkiadaki, SfM-Net: Learning of structure and motion from video, preprint, arXiv:1704.07804.

24. Z. Yin, J. Shi, GeoNet: Unsupervised learning of dense depth, optical flow and camera pose, in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, (2018), 1983–1992. https://doi.org/10.1109/CVPR.2018.00212

25. H. Jiang, L. Ding, Z. Sun, R. Huang, DiPE: Deeper into photometric errors for unsupervised learning of depth and ego-motion from monocular videos, in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, (2020), 10061–10067. https://doi.org/10.1109/IROS45743.2020.9341074

26. J. Xu, L. Su, F. Ye, K. Li, Y. Lai, Densefilter: Feature correspondence filter based on dense networks for VSLAM, *J. Intell. Rob. Syst.*, **106** (2022). https://doi.org/10.1007/s10846-022-01735-9

27. Z. Hongru, Q. Xiuquan, Graph attention network-optimized dynamic monocular visual odometry, *Appl. Intell.*, **53** (2023), 23067–23082. https://doi.org/10.1007/s10489-023-04687-1

28. B. Chen, W. Wu, Z. Li, T. Han, Z. Chen, W. Zhang, Attention-guided cross-modal multiple feature aggregation network for RGB-D salient object detection, *Electron. Res. Arch.*, **32** (2024), 643–669. https://doi.org/10.3934/era.2024031

29. R. Yadav, R. Kala, Fusion of visual odometry and place recognition for SLAM in extreme conditions, *Appl. Intell.*, **52** (2022), 11928–11947. https://doi.org/10.1007/s10489-021-03050-6

30. A. Rosinol, J. J. Leonard, L. Carlone, NeRF-SLAM: Real-time dense monocular SLAM with neural radiance fields, in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, (2023), 3437–3444. https://doi.org/10.1109/IROS55552.2023.10341922

31. C. Chung, Y. Tseng, Y. Hsu, X. Shi, Y. Hua, J. Yeh, et al., Orbeez-SLAM: A real-time monocular visual SLAM with ORB features and NeRF-realized mapping, in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, (2023), 9400–9406. https://doi.org/10.1109/ICRA48891.2023.10160950

32. R. Liang, J. Yuan, B. Kuang, Q. Liu, Z. Guo, DIG-SLAM: an accurate RGB-D SLAM based on instance segmentation and geometric clustering for dynamic indoor scenes, *Meas. Sci. Technol.*, **35** (2024). https://doi.org/10.1088/1361-6501/acfb2d

33. T. Hui, X. Tang, C. C. Loy, Liteflownet: A lightweight convolutional neural network for optical flow estimation, in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, (2018), 8981–8989. https://doi.org/10.1109/CVPR.2018.00936

34. J. Zhang, L. Yang, MonodepthPlus: self-supervised monocular depth estimation using soft-attention and learnable outlier-masking, *J. Electron. Imaging*, **30** (2021), 023017. https://doi.org/10.1117/1.JEI.30.2.023017

35. G. Wang, J. Zhong, S. Zhao, W. Wu, Z. Liu, H. Wang, 3D hierarchical refinement and augmentation for unsupervised learning of depth and pose from monocular video, *IEEE Trans. Circuits Syst. Video Technol.*, **33** (2023), 1776–1786. https://doi.org/10.1109/TCSVT.2022.3215587

36. D. Eigen, R. Fergus, Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture, in *2015 IEEE International Conference on Computer Vision (ICCV)*, (2015), 2650–2658. https://doi.org/10.1109/ICCV.2015.304

37. A. Geiger, J. Ziegler, C. Stiller, Stereoscan: Dense 3d reconstruction in real-time, in *2011 IEEE Intelligent Vehicles Symposium (IV)*, (2011), 963–968. https://doi.org/10.1109/IVS.2011.5940405

38. R. Mur-Artal, J. D. Tardos, ORB-SLAM2: An open-source SLAM system for monocular, stereo, and RGB-D cameras, *IEEE Trans. Rob.*, **33** (2017), 1255–1262. https://doi.org/10.1109/TRO.2017.2705103

39. J. Bian, Z. Li, N. Wang, H. Zhan, C. Shen, M. Cheng, et al., Unsupervised scale-consistent depth and ego-motion learning from monocular video, preprint, arXiv:1908.10553.

40. H. Zhan, R. Garg, C. S. Weerasekera, K. Li, H. Agarwal, I. Reid, Unsupervised learning of monocular depth estimation and visual odometry with deep feature reconstruction, in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, (2018), 340–349. https://doi.org/10.1109/CVPR.2018.00043

41. A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, et al., Pytorch: An imperative style, high-performance deep learning library, preprint, arXiv:1912.01703.

42. D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, preprint, arXiv:1412.6980.

43. W. Zhao, S. Liu, Y. Shu, Y. Liu, Towards better generalization: Joint depth-pose learning without posenet, in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, (2020), 9148–9158. https://doi.org/10.1109/CVPR42600.2020.00917

44. Y. Zou, P. Ji, Q. Tran, J. Huang, M. Chandraker, Learning monocular visual odometry via self-supervised long-term modeling, in *Computer Vision-ECCV 2020*, **12359** (2020), 710–727. https://doi.org/10.1007/978-3-030-58568-6_42

AIMS Press