Deep Learning Boosts Visual Odometry

[1]*Yao-Ting Huang* (黃曜廷) [2]*Chiou-Shann Fuh* (傅楸善)

[1]Department of Electrical Engineering,
National Taiwan University, Taipei, Taiwan,
[2]Department of Computer Science and Information Engineering,
National Taiwan University, Taipei, Taiwan,
*E-mail: b08901189@ntu.edu.tw fuh@csie.ntu.edu.tw

ABSTRACT

This paper presents an enhanced approach for performing Visual-Inertial Odometry of automobile task by leveraging 3D reconstruction technique. Our primary objective is to refine the 3D reconstruction pipeline and incorporate deep learning to aid in the visual odometry process. While Incremental Structure-from-Motion has emerged as a widely used method for reconstructing 3D environments and establishing camera geometry relations, there are still areas that can benefit from further advancements. Notably, the feature extraction and point cloud merging stages of the 3D reconstruction pipeline offer room for improvement and the selection of good feature points is crucially importance in this task because the feature points got from traditional method are mostly not at road marker but gather to the tree nearby. To address these limitations, we apply a novel feature extraction method based on Deep Learning within our 3D reconstruction framework, specifically tailored for visual odometry tasks. By integrating deep learning techniques, we enhance the accuracy and robustness of feature extraction, ultimately improving the overall performance of Visual-Inertial Odometry.

Keywords: *Visual-Inertial Odometry, 3D reconstruction, deep learning.*

1. INTRODUCTION

Visual odometry plays a critical role in the field of robotics as it encompasses the vital task of estimating a robot's pose by analyzing data obtained from visual sensors. The accurate determination of a robot's pose holds immense importance for various robotic applications, including robot control, Simultaneous Localization And Mapping (SLAM), and robot navigation, particularly in situations where external reference data like Global Positioning System (GPS) information is inaccessible. Visual odometry can be regarded as a specific instance of the broader pose tracking problem, which represents a fundamental challenge in robotic perception.

Over the years, researchers have dedicated significant efforts to exploring and implementing diverse visual odometry methods that rely on different types of sensor information. Among these methods, the Iterative Closest Point (ICP) algorithm has gained substantial recognition. The ICP algorithm estimates a robot's pose by minimizing the distance between corresponding points in two laser scanning snapshots. However, this method often faces challenges, especially when an accurate initial guess is not provided, as it tends to get trapped in local optima. Additionally, researchers have investigated odometry methods that utilize camera images. These approaches typically involve extracting point features from camera images and matching them through a series of steps, such as descriptor matching and RANdom SAmple Consensus (RANSAC). Unfortunately, the computational demands of these methods often make them unsuitable for real-time applications. Sparse point features have been used as an approach to enhance computational efficiency, but this technique underutilizes the available image data, thereby discarding valuable information.

Consequently, there is a pressing need for an improved visual odometry method that strikes a balance between computational efficiency and the effective utilization of image data. In this paper, we present a novel approach that addresses these challenges and aims to enhance the accuracy, efficiency, and utilization of visual odometry. Our proposed method incorporates advanced techniques such as deep learning and optimization algorithms to provide robust and real-time visual odometry capabilities for a wide range of robotic applications. By leveraging the power of deep learning, we aim to extract more meaningful information from visual sensor data and optimize the pose estimation process to achieve superior performance compared to existing methods. Through extensive experimentation and evaluation, we

demonstrate the effectiveness and practicality of our approach in real-world scenarios.

The remainder of this paper is organized as follows: Section 2 provides an overview of related work in the field of visual odometry, highlighting the strengths and limitations of existing methods. Section 3 presents the proposed method in detail, including the architecture of our deep learning model and the optimization algorithms employed. Section 4 describes the experimental setup and datasets used for evaluation, while Section 5 presents the results and analysis of our method's performance. Finally, Section 6 concludes the paper, summarizing the contributions and discussing future directions for research in the field of visual odometry.

2. RELATED WORKS

2.1. SLAM

The SLAM++ [1] framework addresses the challenges of simultaneous localization and mapping (SLAM) by incorporating object-level information into the mapping process. Traditional SLAM methods primarily focus on reconstructing the geometry and position of the environment. In contrast, SLAM++ extends this approach by considering objects as key components of the scene. By leveraging object-level semantic information, the SLAM++ system achieves higher accuracy and robustness in both localization and mapping tasks. The authors propose a novel map representation that incorporates object-level features and demonstrate improved performance compared to traditional SLAM approaches in complex and dynamic environments.

ORB-SLAM [2] is a monocular SLAM system that offers versatility and accuracy in real-time environments. The system leverages a combination of efficient keypoint extraction, robust feature matching, and loop closure detection to achieve reliable pose estimation and map reconstruction using a single camera. ORB-SLAM adopts the ORB (Oriented FAST and Rotated BRIEF) feature descriptor, which provides excellent performance in terms of both speed and accuracy. The system also incorporates loop closure detection to handle large-scale environments and maintain map consistency over extended periods. Experimental evaluations demonstrate the effectiveness of ORB-SLAM in various challenging scenarios, including indoor and outdoor environments with significant camera motion and dynamic objects.

These two related works highlight advancements in the field of SLAM, which is closely related to visual odometry. SLAM approaches address the simultaneous tasks of estimating the robot's pose and mapping the environment, incorporating semantic object information (SLAM++) or focusing on monocular systems (ORB-SLAM). Both works demonstrate the potential to improve the accuracy, robustness, and versatility of SLAM, which can significantly benefit visual odometry methods in terms of providing reliable and comprehensive scene understanding.

2.2. Structure-from-Motion

Structure-from-Motion (SfM) is a fundamental technique in computer vision that aims to reconstruct the 3D structure of a scene from a collection of 2D images or video frames. SfM involves estimating the camera poses (position and orientation) and the 3D locations of points in the scene.

Visual-Inertial Odometry (VO) is a specific application of SfM that focuses on estimating the motion of a camera or an observer in a 3D environment using visual information. In VO, the camera motion is estimated by tracking visual features across consecutive frames and computing the relative camera poses between them. By accumulating these poses over time, the trajectory of the camera can be reconstructed.

SfM and VO are closely related and share similar principles. Both techniques rely on extracting visual features from images or video frames and establishing correspondences between them. These correspondences are then used to estimate camera motion and reconstruct the scene.

The main difference between SfM and VO lies in their objectives. SfM aims to reconstruct the complete 3D structure of the scene, including the 3D locations of points and camera poses, whereas VO focuses primarily on estimating the camera motion.

However, VO can be seen as a crucial component of SfM pipelines. In many SfM systems, VO is used as an initial step to estimate camera motion and bootstrap the scene reconstruction process. By accurately estimating camera poses through VO, SfM algorithms can refine the 3D structure of the scene and optimize the camera poses further.

2.3. Object Segmentation

Segment Anything (SA) [3] project focuses on image segmentation. Image segmentation is the task of dividing an image into different segments or regions based on their visual characteristics. The authors aim to develop a promptable model for image segmentation that can generalize to new data distributions and tasks. They address three main components: task, model, and data.

The proposed task is promptable segmentation, where the model generates a valid segmentation mask given a segmentation prompt. The prompt can provide spatial or text information to identify the object to be segmented. The model's output should be a reasonable mask, even in cases of ambiguity where a prompt could refer to multiple

objects. This promptable segmentation task serves as both a pre-training objective and a means to solve downstream segmentation tasks using prompt engineering.

The model architecture, called the Segment Anything Model (SAM), is designed to meet the requirements of the promptable segmentation task and real-world use. SAM consists of an image encoder, a prompt encoder, and a mask decoder. The image encoder computes an image embedding, the prompt encoder embeds prompts, and the mask decoder predicts segmentation masks by combining the information from the image and prompt encoders. SAM's design allows for flexible prompts, real-time mask computation, and ambiguity awareness. It can predict multiple masks for a single prompt to handle ambiguity effectively.

To train the model, the authors built a large-scale dataset called SA-1B, which contains over 1 billion masks on 11 million licensed and privacy-respecting images. They developed a data collection loop using their efficient model to assist in data collection and improve the model iteratively. The dataset and experiments demonstrate the effectiveness of their approach, with zero-shot performance often competitive or superior to prior fully supervised results.

In summary, the paper introduces the SA project, which focuses on promptable image segmentation. They propose a promptable segmentation task, develop the Segment Anything Model (SAM) architecture, and create a large-scale dataset for training. The results show promising zero-shot performance and highlight the importance of object segmentation in visual odometry, as it enables powerful generalization to new data distributions and tasks.

Mask2Former [4], a universal image segmentation architecture that outperforms specialized architectures in various segmentation tasks. The architecture consists of a backbone feature extractor, a pixel decoder, and a Transformer decoder. They propose several key improvements to enhance the performance and training efficiency of the model.

Masked Attention in Transformer Decoder: We introduce masked attention in the Transformer decoder, which limits the attention mechanism to localized features centered around predicted segments. This approach, compared to the standard cross-attention used in a Transformer decoder, leads to faster convergence and improved performance.

Multi-Scale High-Resolution Features: They utilize multi-scale high-resolution features to aid in segmenting small objects or regions. By incorporating these features into the architecture, the model becomes more effective at capturing fine details.

Optimization Improvements: They suggest optimization improvements to further enhance performance without increasing computational complexity. These include switching the order of self and cross-attention, making query features learnable, and removing dropout.

Memory Optimization: They optimize the memory usage during training by calculating the mask loss on only a few randomly sampled points. This approach reduces training memory requirements by a factor of three without compromising performance.

Through extensive evaluation on three segmentation tasks (panoptic, instance, and semantic segmentation) and four popular datasets (COCO [5], Cityscapes, ADE20K, and Mapillary Vistas), we demonstrate that Mask2Former achieves state-of-the-art results. Specifically, our architecture achieves 57.8 PQ on COCO panoptic segmentation, 50.1 AP on COCO instance segmentation, and 57.7 mIoU on ADE20K semantic segmentation using the exact same architecture.

Overall, Mask2Former architecture surpasses specialized architectures in performance while maintaining ease of training. The improvements we introduce, such as masked attention, multi-scale features, optimization enhancements, and memory optimization, collectively contribute to the success of Mask2Former as a universal image segmentation solution.

3. METHOD

In this section, we first introduce the pipeline of our method and then explain the method we use in details, finally, discuss the difference of the feature extraction result and the reconstruction quality.

3.1. Pipeline

The complete pipeline of our proposed methodology is visually depicted in Figure 1, illustrating the sequential flow of three distinct steps: feature extraction, reconstruction, and localization. This pipeline is designed to effectively process the input data and derive accurate localization information for the robot's navigation in real-world environments.

In the initial step of feature extraction, we employ the Mask2former model in conjunction with the SAM model to perform precise semantic segmentation of the road markers. This process allows us to isolate the relevant regions of interest, effectively filtering out unwanted objects and background noise. By leveraging these advanced models, our pipeline ensures the extraction of high-quality features that are essential for subsequent stages.

Moving on to the reconstruction step, we utilize the pinhole model method to reconstruct the precise coordinates of each detected road marker. This approach leverages the inherent geometric properties of the camera and the captured image data to infer the accurate spatial positions of the road markers in the environment. By reconstructing the 3D coordinates, we establish a reliable representation of the physical world that enables further analysis and processing.

The final step in our pipeline is the localization process, where we employ the well-established Iterative Closest Point (ICP) algorithm. This algorithm plays a crucial role in estimating the transformation between the instance point cloud, representing the detected road markers, and the target point cloud, representing the reference or known environment. By iteratively refining the transformation estimation, ICP enables us to derive the precise localization information, which encompasses the robot's pose and position within the environment.

The integration of these three steps within our pipeline provides a comprehensive and robust framework for accurate localization in real-world scenarios. By combining advanced feature extraction techniques, geometric reconstruction methods, and the reliable ICP algorithm, our methodology offers a holistic solution that addresses the challenges of localization in robotics. This pipeline serves as a solid foundation for various applications, including robot control, Simultaneous Localization And Mapping (SLAM), and efficient robot navigation in environments where external reference data, such as Global Positioning System (GPS) information, is not accessible.

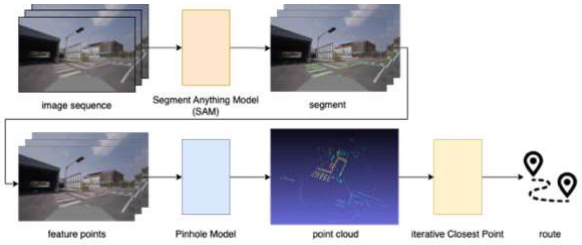

Fig. 1. The pipeline of our method.

3.2. Feature Extraction

In the specific scenario of our task, the majority of feature points are located on the road surface. To ensure accurate detection and avoid false positives, we leverage a previously developed technique called Mask2former [4], which excels in performing semantic segmentation. By utilizing the segmentation results obtained from Mask2former, we are able to identify and isolate the road region within our input images. This segmentation of the road region enables us to employ the Sequential Approximate Median (SAM) algorithm to further segment each road marker and extract its contour, as illustrated in Figure 2.

Fig. 2. The contour of each road marker.

Once we have obtained the contours of the road markers, we optimize the computational efficiency and alignment complexity by down sampling the extracted points. This down sampling process involves approximating the numerous contour points to form a polygon with fewer edges. By reducing the number of points, we not only decrease the computation time but also simplify the alignment process, making it more manageable and efficient. This down sampling step ensures that our method can handle real-time applications and provides an improved balance between computational resources and accuracy in feature extraction.

3.3. 3D Reconstruction

We use pinhole method instead of incremental SfM because the detected feature points extracted from the model segmentation result are not distinguishable and could lead to poor matches. Owing to the number of outlies, the RANSAC method of SfM could not find the optimized fundamental matrix of consecutive views. Thus, we utilize the transform of main camera with respect to the vehicle and then project the detected feature points to the plane of $z = -camera\ height$ at vehicle coordinate.

We use the pinhole model to project the pixel point in the frame to vehicle coordinate by solving Eq.1 by setting $z_w = -camera\ height$.

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ z_w \end{bmatrix}$$

Eq. 1. The pinhole camera model. The R and T represent the rotation matrix and transpose from world coordinate to camera coordinate

3.4 Merge View

Once we have obtained the 3D points corresponding to each contour from multiple cameras, our next step is to merge these point clouds, as shown in the Figure 3, there several contours which are not aligned and should be merged into one contour. To achieve this, we employ a filtering process that identifies and retains the point clouds representing the same road markers. We accomplish this by calculating the Intersection over Union (IoU) for each pair of contour points.

To establish a suitable threshold for the IoU, we set it to 0.7, although this value can be fine-tuned as a hyperparameter. Contours with an IoU exceeding this threshold are considered potential matches and are subjected to further analysis. To determine which contours should be kept, we employ two criteria.

First, we calculate the ratio of their respective areas. If one contour is approximately twice the size of the other, we retain the larger contour since it contains a greater number of points, thereby providing a more robust representation of the road marker.

In cases where the areas of the two contours are roughly equal, we resort to a secondary criterion. We select the contour whose center is closer to the origin of the vehicle coordinate system, as depicted in Figure 4. By making this selection, we prioritize the contour that is more central in the vehicle's frame of reference, thereby enhancing the accuracy and reliability of the merged point cloud.

By applying these filtering and selection criteria, we ensure that the merged point cloud comprises the most relevant and accurate representation of the road markers from multiple camera perspectives. This approach improves the overall quality and consistency of the merged point cloud, enabling robust perception and localization capabilities for the autonomous vehicle.

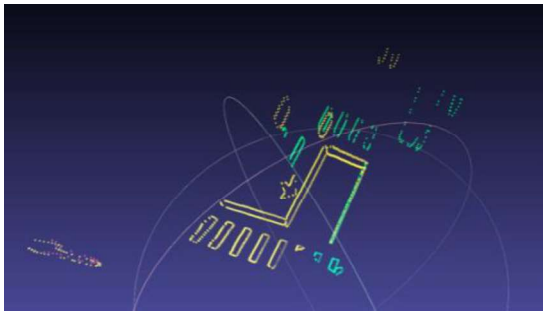

Fig. 3. The point cloud reconstructed, and the color of point clouds represents their source camera. We can observe that there is unaligned contour between yellow and green source at the zebra cross part.

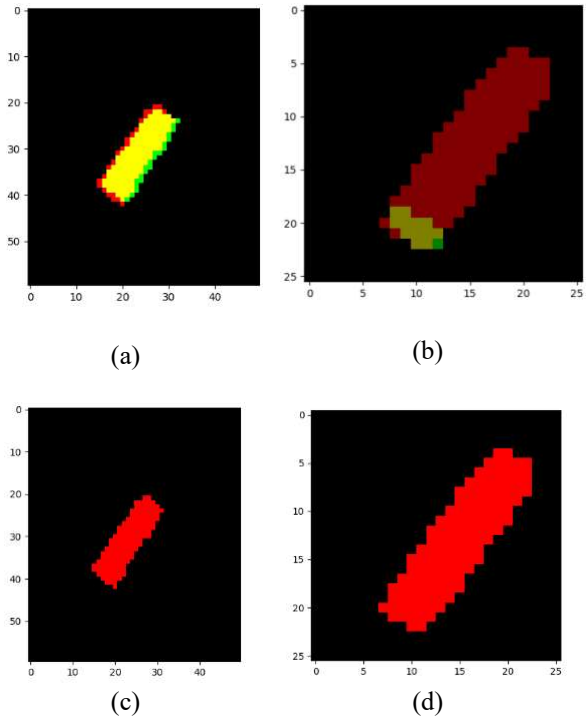

Fig. 4. The merged point clouds. (a) shows the contours that IoU greater than 0.7, (b) shows the contours that IoU lower than 0.7, (c) shows the better contour from (a), (d) shows the better contour from (d)

3.5 ICP

Initialization: The ICP algorithm begins by initializing the estimation process. In your implementation, an initial transformation is provided in the dataset. This initial guess serves as a starting point for the iterative refinement process.

Correspondence Search: Each point in the source point cloud is matched with its closest point in the target point cloud using the Euclidean distance as the distance metric. This correspondence search establishes point-to-point associations between the two point clouds.

Transformation Estimation: Utilizing the established correspondences, the transformation between the source and target point clouds is estimated. The most common transformation used in ICP is the rigid transformation, represented by a 4x4 transformation matrix. Various techniques such as least squares optimization or singular value decomposition (SVD) can be employed to estimate the transformation.

Point Cloud Alignment: The estimated transformation is applied to the source point cloud, aligning it with the target point cloud. This alignment step brings the source point cloud closer to the target point cloud, minimizing the distance between corresponding points.

Error Minimization: The alignment error is calculated by measuring the discrepancy between the transformed source point cloud and the target point cloud. The specific error metric employed in your implementation was not provided, but typically, it involves measuring the sum of squared differences between corresponding points.

Convergence Check: At each iteration, the algorithm checks if the alignment error has reached a satisfactory level or if the estimated transformation has converged. Convergence can be determined by evaluating the change in error between iterations or comparing the alignment error to a predefined threshold. If the convergence criteria are not met, the algorithm proceeds to the next iteration.

Iteration: Steps 2 to 6 are repeated iteratively until convergence is achieved or a maximum number of iterations, set to 30 in your implementation, is reached. Each iteration refines the transformation estimation and improves the alignment between the source and target point clouds.

Output: The final estimated transformation is obtained when the algorithm converges. This transformation can be used to accurately align the two point clouds, providing an estimation of their relative positions and orientations. The result is shown in Figure 5.

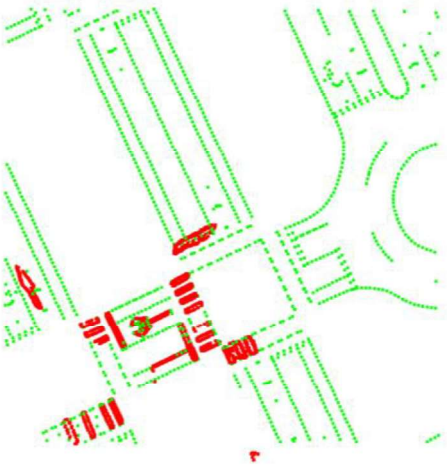

Fig. 5. Aligned point cloud.

3.5 Trajectory refinement

In order to refine the trajectory estimation, additional steps were taken to address outliers and improve the continuity of the predicted position graph. Examination of the prediction position graph revealed irregularities and discontinuities in the expected continuous line. To rectify this issue, a Gaussian filter was applied to the prediction position data.

The Gaussian filter is a commonly used technique for smoothing data and reducing noise. By convolving the prediction position graph with a Gaussian kernel, the filter effectively suppresses outliers and enhances the overall continuity of the trajectory. This process enables a more accurate estimation of the robot's motion trajectory.

The filtered prediction position graph, as shown in the Figure 6, illustrates the efficacy of the Gaussian filter in eliminating outliers and producing a smoother trajectory. The refined trajectory provides a more reliable representation of the robot's actual movement, allowing for improved analysis and decision-making in robotic applications.

By employing the Gaussian filter for trajectory refinement, the accuracy and continuity of the predicted position graph are significantly enhanced. The resulting trajectory provides a more reliable basis for subsequent tasks such as path planning, control, and mapping, facilitating improved performance and reliability in various robotic applications.

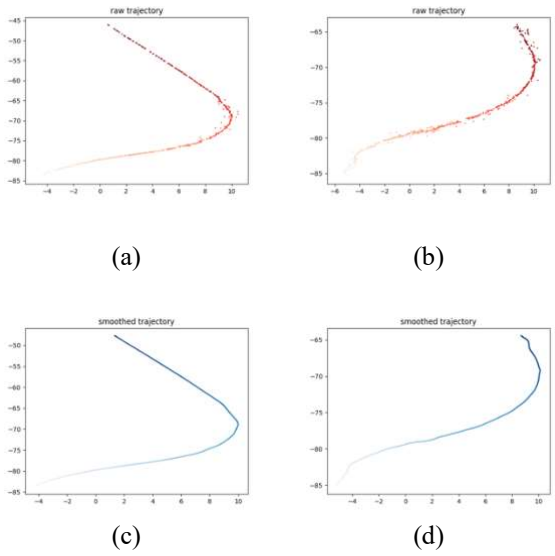

Fig. 6. The trajectories of raw prediction and the filtered trajectories. (a) is the raw predicted trajectory of sequence 1 in our dataset. (b) is the raw predicted trajectory of sequence 2 in our dataset. (c) is the filtered trajectory of sequence 1 in our dataset. (d) is the filtered trajectory of sequence 2 in our dataset.

4. EXPERIMENTAL SETUP

4.1 Dataset

The dataset utilized in our study comprises several sequences of data, each offering a diverse range of information essential for accurate localization. Within each sequence, the dataset provides a collection of raw images capturing the environment, vehicle outline masks

delineating the vehicle's boundaries, YOLO detection bounding boxes outlining detected objects, and precise timestamps corresponding to each frame. This dataset has been graciously provided by the Industrial Technology Research Institute (ITRI), offering a comprehensive and reliable resource for our research.

In addition to the rich image data, the dataset also includes crucial supplementary information necessary for our localization pipeline. Specifically, it provides the projection matrix, a fundamental component for the camera's perspective projection, enabling accurate mapping of 3D world coordinates to the 2D image plane. Furthermore, the dataset furnishes the transformations between each camera employed on the vehicle, facilitating the synchronization of data captured from different camera perspectives. This information plays a vital role in aligning and integrating multiple sources of visual data, allowing for comprehensive analysis and interpretation.

To facilitate the application of the Iterative Closest Point (ICP) algorithm, a widely adopted technique for point cloud matching and alignment, the dataset further supplies a sub_map. This sub_map, acquired by a LiDAR system deployed in the data collection environment, serves as a reference map capturing the spatial layout and structural information of the surroundings. Additionally, the dataset incorporates manual labeling of road markers on the depth map, offering precise ground truth information that aids in the accurate alignment of the predicted point cloud with the corresponding regions in the sub_map.

The inclusion of these supplementary components within the dataset enhances the robustness and accuracy of our localization pipeline. By leveraging the projection matrix, camera transformations, sub_map, shown in Figure 7, and manual labeling of road markers, we ensure the alignment of the predicted point cloud with the ground truth, allowing for precise localization and mapping. These meticulously curated data elements from ITRI contribute significantly to the reliability and validity of our research findings, enabling us to evaluate and enhance the performance of our proposed methodology with confidence.

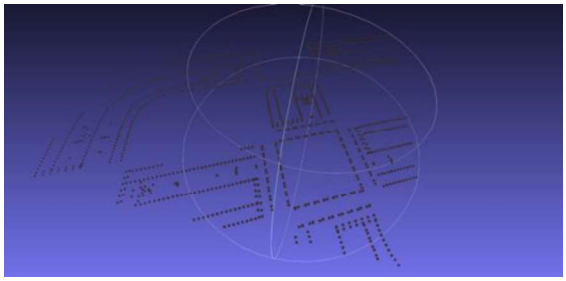

Fig. 8. This figure shows the 3D point cloud of Sub_map, which is collected by LiDAR and manually labelled road marker.

5. RESULT

5.1 Feature extraction method

A comparative analysis was conducted between the features extracted directly from SIFT [6] detection and the features constrained within the segment region, as depicted in Figure 8. The examination revealed that the latter method provides more comprehensive and accurate descriptions of the road marker compared to the former. The utilization of features exclusively within the segment region enables a more focused representation of the informative regions in the image, leading to improved localization accuracy. The abundance of feature points within these regions allows for a richer and more detailed representation of the road marker, further enhancing the precision and reliability of the results obtained from the visual odometry system. This observation highlights the significance of capturing and utilizing a sufficient number of feature points within the informative region to ensure the robustness and accuracy of the localization process.

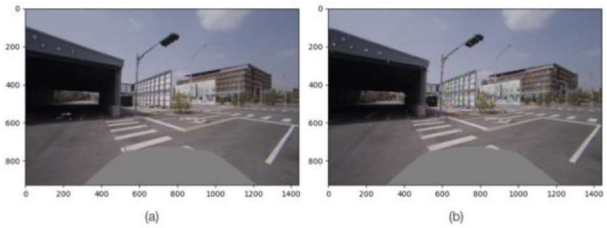

Fig. 8. Feature points extracted by different methods. (a) The features constrained by the segmentation of SAM. (b) The features got directly from the SIFT.

5.2 Ablation study

To evaluate the effectiveness of our proposed method, we conducted comprehensive testing on every sequence of data available in the dataset. Additionally, we performed an ablation study to assess the impact of each optimization technique employed in our methodology. The results of these experiments are summarized in Table 1, highlighting the performance improvements achieved through the integration of our proposed optimizations.

Upon analyzing the results, it is evident that the application of all the optimization methods significantly contributes to reducing the Mean Squared Error (MSE) loss. The observed decrease in MSE loss serves as a clear indicator of the enhanced accuracy and reliability attained through our proposed approach. This improvement in performance underscores the effectiveness of our optimization techniques in refining the trajectory estimation and enhancing the overall quality of the localization output.

The findings from our ablation study provide valuable insights into the individual contributions of each optimization component. By systematically evaluating

the impact of these optimizations, we gain a deeper understanding of their significance in improving the overall performance of our method. The results highlight the synergistic effect of combining multiple optimization techniques, demonstrating their cumulative impact in achieving superior localization accuracy.

The comprehensive evaluation and ablation study carried out on our proposed method confirm its effectiveness in enhancing localization performance. We have test 4 setting of our methods, 1) O: SAM + pinhole model, 2) O+R: SAM + pinhole model + contour removal, 3) O+F: SAM + pinhole model + trajectory filtering, 4) O+R+F: SAM + pinhole model + contour removal + trajectory filtering. By implementing the optimized approach, we achieve a notable reduction in MSE loss, thereby enhancing the precision and reliability of the localization results. These findings validate the efficacy of our proposed optimization techniques and reinforce the applicability of our method in real-world robotic scenarios.

	O	O+R	O+F	O+R+F
Case1	0.27208	0.26228	0.22335	0.22008
Case2	0.25746	0.26101	0.18863	0.18325
Case3	0.26666	0.2608	0.22657	0.22241

Table. 1. The MSE loss of different settings of our method.

6. CONCLUSION

In conclusion, this paper presents a novel approach to visual odometry that addresses the challenges of computational efficiency and effective utilization of image data. Through the integration of advanced techniques such as deep learning and optimization algorithms, our proposed method achieves robust and real-time visual odometry capabilities for various robotic applications. By leveraging deep learning, we extract more meaningful information from visual sensor data, enhancing the accuracy of pose estimation. Furthermore, the optimization algorithms employed in our methodology refine the trajectory estimation and improve the overall quality of localization output.

Experimental evaluation conducted on the dataset demonstrates the effectiveness of our proposed method. By comparing the features extracted directly from SIFT detection with the features constrained by the segmentation of the SAM model, we observe that the latter provides more accurate and informative descriptions of road markers. Our comprehensive testing on every sequence data in the dataset, along with the ablation study of our proposed optimizations, further validates the performance improvements achieved through our methodology.

The results clearly indicate a notable decrease in Mean Squared Error (MSE) loss, highlighting the enhanced accuracy and reliability of our approach. This decrease in MSE loss serves as evidence of the positive impact of our optimization techniques on trajectory refinement and localization accuracy. Additionally, the ablation study reveals the synergistic effect of combining multiple optimizations, further emphasizing the cumulative impact of these techniques on overall performance.

The proposed method presented in this paper represents a significant advancement in the field of visual odometry. By leveraging deep learning and optimization algorithms, we have achieved a balance between computational efficiency and the effective utilization of image data. Our findings not only contribute to the improvement of visual odometry but also hold promising implications for various robotic applications requiring accurate pose estimation. Future research directions may involve exploring further optimization techniques and investigating the application of our method in different robotic scenarios to maximize its potential and expand its applicability in real-world settings.

6. SOURCE CODE

The source code of the project can be viewed in GitHub link.

REFERENCES

[1] Salas-Moreno, R. F., Newcombe, R. A., Strasdat, H., Kelly, P. H., & Davison, A. J. (2013). Slam++: Simultaneous localization and mapping at the level of objects. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1352-1359).

[2] Mur-Artal, R., Montiel, J. M. M., & Tardos, J. D. (2015). ORB-SLAM: a versatile and accurate monocular SLAM system. IEEE transactions on robotics, 31(5), 1147-1163.

[3] Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., ... & Girshick, R. (2023). Segment anything. arXiv preprint arXiv:2304.02643.

[4] Cheng, B., Misra, I., Schwing, A. G., Kirillov, A., & Girdhar, R. (2022). Masked-attention mask transformer for universal image segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 1290-1299).

[5] Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., ... & Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13 (pp. 740-755). Springer International Publishing.

[6] D. Lowe, Object recognition from local scale-invariant features, in: Proceedings of the International Conference on Computer Vision ICCV, Corfu, 1999, pp. 1150–1157.