End-to-End Learned Visual Odometry Based on Vision Transformer

University of Turku
Department of Computing
Master of Science Thesis
Autonomous System Lab
July 2024
Aman Manishbhai Vyas

Supervisors:
Adjunct Professor Hashem Haghbayan
Professor Juha Plosila

UNIVERSITY OF TURKU
Department of Computing

Aman Manishbhai Vyas: End-to-End Learned Visual Odometry Based on Vision Transformer

Master of Science Thesis, 63 p.
Autonomous System Lab
July 2024

Estimating the camera's pose from images of a single camera, a task known as monocular visual odometry, is fundamental in mobile robots and autonomous vehicles. Traditional approaches often rely on geometric methods that require significant engineering effort tailored to specific scenarios. Deep learning methods, while generalizable with extensive training data, have shown promising results. Recently, transformer-based architectures, which have been highly successful in natural language processing and computer vision, are proving to be superior for this task as well. In this study, we introduce the Vision Transformer (ViT) model, which leverages spatio-temporal self-attention mechanisms to extract features from images and estimate camera motions in an end-to-end manner.

Extensive experimentation on the KITTI visual odometry dataset demonstrates that ViT achieves competitive state-of-the-art performance. Remarkably, it surpasses both traditional geometry-based methods and existing deep learning approaches, including DeepVO, MagicVO, and PoseNet. This significant improvement underscores the effectiveness of transformer-based architectures in capturing complex spatio-temporal dependencies essential for accurate visual odometry. Our results highlight ViT's potential as a powerful tool for enhancing pose estimation in dynamic environments, making it a valuable contribution to the advancement of autonomous navigation technologies. Our results over five different route trajectories with varying environmental conditions show that ViT achieves up to an 8% improvement in translation error and a 4% improvement in rotation error compared to previous deep learning methods. This highlights ViT's potential to enhance pose estimation in dynamic environments and advance autonomous navigation.

Keywords: Visual Odometry, Vision Transformer, Self Attention, spatio-temporal information

Contents

List of Figures

List of Tables

List of Acronyms

IMU - Inertial Measurement Unit

KNN - k-nearest neighbors

SVM - support vector machines

DRCNN - Deep recurrent convolutional neural network

VO - Visual Odometry

ViT - Vision Transformer

CNN- Convolution Neural Network

KF - Kalman Filter

EKF - Extended Kalman Filter

UKF - Unscented Kalman Filter

MSCKF - Multi-State Constraint Kalman Filter

BERT - Bidirectional Encoder Representations from Transformer

GPT -Generative Pre-trained Transformer

PF - Particle Filter

RNN - Recurrent Neural Network

LSTM - Long Short Term Memory

Bi-LSTM - Bi-Directional Long Short-Term Memory

DNNs - Deep Neural Networks

GANs - Generative Adversarial Networks

SGANs - stacked generative adversarial networks

MHSA - Multi-Head Self-Attention

MLP - Multi-layer perceptron

FFN - Feed-Forward Network

ReLU - Rectified Linear Unit

MSE - Mean Squared Error

Adam - Adaptive Moment Estimation

RPE - Relative Pose Error

TE - Translation Error

RE - Rotation Error

GPS - Global Positioning System

LIDAR - Light Detection and Ranging

1 Introduction

Determining the precise pose of a robot within its environment is a fundamental and longstanding task in the field of robotics. One approach that has gained significant traction in this domain is Visual Odometry (VO). VO involves localizing a robot by relying solely on visual sensors, such as monocular cameras [1], stereo cameras [2], and event-based cameras [3]. This technique is pivotal for various robotic applications, from mobile robots to drones and autonomous driving vehicles [4], [5]. Its versatility lies in the ability to predict the camera's pose based on a sequence of frames, providing a dynamic scene understanding of the robot's position and orientation in real time [6].

Traditional methods like IMU and wheel-based odometry [7], which rely on sensor calculations with mechanical components, struggle with understanding dynamic environments due to drift and mechanical constraints, making them less effective on complex terrains [8]. In contrast, VO has emerged as a transformative alternative [9], [10], enhancing accuracy and offering cost-effective solutions for various robotic applications, particularly where traditional methods face challenges. Monocular visual odometry can be implemented through traditional geometry-based methods [11], [12], which involve camera calibration, feature detection, matching, outlier rejection, and local optimization. Despite its structured pipeline, this approach requires extensive fine-tuning and suffers from scale ambiguity, making it less adaptable across different scenarios.

In contrast to traditional methods, the second type of monocular visual odometry employs deep learning-based approaches with end-to-end architectures [13], eliminating the need for intricate feature engineering. These methods learn to map sequential frames to camera poses directly, enhancing generalizability across diverse environments. By automatically extracting relevant features and inferring motion patterns, they reduce the need for extensive manual tuning. However, challenges include the demand for substantial labeled data and potential computational complexities. The exploration of these methodologies aims to balance the robustness of traditional geometric principles with the adaptability of end-to-end learning architectures.

The conventional approach for Learning based VO has predominantly leaned on Convoluation Neural Networks (CNNs) [14], leveraging their prowess in extracting local features from images. While this has proven effective in various image processing tasks, the nature of VO, as a video understanding task, necessitates the extraction of global information from visual cues within a sequence of images. To bridge this gap, researchers are turning their attention to a novel architecture known as the Vision Transformer. The motivation behind this shift is to harness the Vision Transformer's ability to capture long-range dependencies and global contextual information. The proposed research seeks to employ Vision Transformer and architectures for Visual Odometry, with a dual focus on minimizing losses of the chosen framework.

One of the recently used methods designed for image classification tasks is Vision Transformer (ViT) [15]. In ViT, a transformer encoder block is incorporated to capture sequential information from image data. Mainly, ViT is used for various applications such as image classification and object detection. However, its capability to encapsulate spatial-temporal information makes it a promising method for

visual odometry. In this paper, we leverage this capability by adapting ViT for RGB-image-based vo to capture sequential information. The obtained results based on our experimental setup, which includes the KITTI dataset [16] for training and testing, show that the proposed method aligns well with state-of-the-art methods, both formal and data-driven.

Leading players in the autonomous driving industry, including Tesla, Waymo, and Mobileye, are increasingly embracing a camera-centric approach for their perception stacks [17]. This strategic shift towards relying primarily on cameras for environmental perception in autonomous vehicles has opened a significant avenue for research and innovation. The prominence of these marquee names in adopting camera-based solutions underscores the industry's confidence in the potential of visual information for robust and reliable perception. This paradigm shift presents a compelling opportunity for researchers to delve into the intricacies of this approach, contributing insights that push the boundaries of cutting-edge technology. By addressing the challenges and nuances inherent in camera-based perception systems, researchers have the chance to not only advance the capabilities of autonomous driving platforms but also pave the way for the broader adoption of vision-centric solutions in the rapidly evolving landscape of self-driving vehicles.

The rest of the section in this chapter is organized as follows: Section 1.1 outline the research questions and objectives of the study, while section 1.2 detail the contribution of the study. Section 1.3 underscores the impact and importance of the research, and section 1.4 explain organization of the remaining thesis chapters.

1.1 Research questions and objectives

The research questions outlined in the provided text revolve around improving VO through the utilization of the Vision Transformer architecture, with a specific focus on addressing accuracy and precision. The primary research questions can be inferred as follows:

(1). How can Vision Transformer architectures be effectively applied to VO to address the need for capturing long-range dependencies and global contextual information in VO?

(2). What specific architectural features of ViT-based models contribute to enhanced robustness and reliability in dynamic and complex visual environments for Visual Odometry applications?

(3). Can the self-attention mechanisms in ViT-based VO models effectively mitigate common issues such as drift and noise in pose estimation, and how do these mechanisms compare to those in conventional methods?

(4). How does Vit-based VO compare to other Learning Based methods in terms of accuracy and precision in pose estimation? What specific advantages does the ViT architecture offer in enhancing pose estimation accuracy and robustness, and how can these benefits be quantified and demonstrated through empirical research?

(5). To what extent do current learning-based approaches in VO fall short in terms of generalization across diverse environments and terrains?

1.2 Contribution

In this thesis, we explore how Vision Transformer can be utilized to learn the VO task. The main contributions include:

(1). The study delves into specific architectural features of ViT-based models that contribute to their robustness and reliability in dynamic and complex visual environments. Using multi-head self-attention mechanisms allows for better integration of contextual information across the entire scene, improving the model's performance in various scenarios.

(2). This research investigates whether self-attention mechanisms in ViT-based VO models can effectively mitigate common issues such as drift and noise in pose estimation. By comparing these mechanisms to those used in conventional methods, the study aims to demonstrate the superiority of ViT-based approaches in maintaining accurate and stable pose estimates over time.

(3). The study addresses the limitations of current learning based VO approaches regarding generalization across diverse environments and terrains. By leveraging the global contextual information captured by ViTs, the research aims to improve the adaptability and performance of VO systems in various real-world conditions without extensive retraining.

Overall, the research seeks to advance the capabilities of Visual Odometry by leveraging the Vision Transformer architecture

1.3 Research significance of the study

This thesis addresses several critical knowledge gaps in VO, particularly in leveraging novel architectures like ViTs for improved performance and practical viability. By introducing ViT architectures to the domain of VO, this research fills a significant void in understanding how these models can effectively capture long-range dependencies and global contextual information in video understanding tasks. Before this work, the predominant reliance on CNNs for local feature extraction often overlooked the crucial aspect of extracting global information from image sequences. This oversight is particularly limiting in dynamic environments where accurate pose estimation depends on a holistic understanding of the scene.

Furthermore, the research explores the specific architectural features of ViT-based models that contribute to enhanced robustness and reliability in complex visual environments. The self-attention mechanisms inherent in ViTs allow for better integration of contextual information across the entire scene, significantly improving the model's ability to maintain accurate pose estimates amidst noise and variability. This advancement not only provides a more detailed and accurate approach to pose estimation but also highlights the limitations of CNN based learning based methods that primarily focus on local features. By demonstrating the efficacy of ViT-based approaches through empirical research and comparisons with SOTA learning based methods, this thesis establishes a new benchmark for VO systems in term of accuracy and precision, emphasizing the importance of global context in achieving high precision and robustness.

1.4 Structural outline of this thesis

The rest of the thesis is organized as follows:

(1). Chapter 2 explores existing work related to traditional methods and learning-based VO. It provides a comprehensive overview of prior research, methodologies, and key findings in the field. The review highlights the limitations and gaps in current approaches, such as the reliance on CNNs for local feature extraction, which often neglects the importance of capturing global contextual information crucial for accurate pose estimation in dynamic environments.

(2). Chapter 3 establishes the theoretical foundation for integrating Vision Transformer architectures into VO. This chapter includes a brief description of the KITTI dataset, which is used for training and evaluation. Additionally, it outlines the training procedure, experimental setup, and evaluation metrics for VO and providing a detailed exploration of various aspects such as losses. ensuring a robust framework for assessing the performance of the proposed approach.

(3). Chapter 4 constitutes the primary focus of the thesis, It analyzes and interprets the obtained results, offering insights into the efficacy and practical implications of the proposed ViT-based approach to existing learning-based VO methods.

(4). Chapter 5 summarizes the key findings of the research, discusses their implications, and addresses the limitations of ViT-based methods. It outlines avenues for future work, providing a comprehensive conclusion to the thesis.

2 Literature review

In this chapter of the thesis, a detailed literature review for VO is provided. Firstly, section 2.1 review traditional VO methods such as filtering-based, optimization-based, and geometry-based VO. The chapter then delves deeper into data-driven VO literature in section 2.2, exploring the architecture of current state-of-the-art models. Section 2.3 explains the limitations and gaps in current approaches and proposes the ViTVO method and highlighting its high-level architecture. Section 2.4 and 2.5 give advantages, and applications of ViTVO.

2.1 Tradinational approaches

In this section, we delve deeply into the literature on filtering-based, optimization-based, and geometry-based VO methods. We review some state-of-the-art methods within each category, providing detailed insights into their approaches and performances.

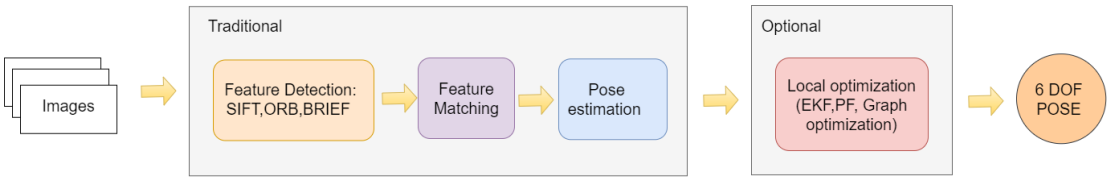

Figure 2.1: Traditional vo Pipeline for optimization, filtering and geometry based method

2.1.1 Filtering based approach

VO aims to estimate the motion of a camera through a sequence of images, a crucial component in the navigation of mobile robots, autonomous vehicles, and other vision-based applications. Filtering-based approaches [18] to VO use probabilistic filters to integrate sensor data over time, providing a robust estimate of the camera's pose.

1. Kalman Filter (KF): KF [19]is a foundational method in estimation theory, applied to VO for tracking the camera's position and orientation over time. In VO, the state vector typically includes parameters such as the camera's position, orientation, and possibly velocity. The KF operates under the assumption that both the process (how the state evolves over time) and observation (how measurements relate to the state) models are linear, and that noise follows a Gaussian distribution.

   In practice, the KF predicts the next state of the camera's pose based on its current state and the dynamics of the system. This prediction is updated when new measurements, typically derived from feature tracking in image sequences, become available. The update process adjusts the predicted state based on the difference between the predicted and observed measurements, weighted by their respective uncertainties (covariances). While effective in certain scenarios, the KF's linear assumptions limit its application in VO, where nonlinearities are prevalent due to complex camera motions and scene variations.

2. Extended Kalman Filter (EKF): EKF [20] extends the KF to handle nonlinearities in both process and observation models, making it more suitable for VO. Nonlinearities arise when the motion or measurement models are not

linear functions of the state variables. To mitigate this, the EKF linearizes these models around the current state estimate using the Jacobian matrix. The Jacobian provides a local linear approximation [21], allowing the filter to propagate the state uncertainty and update the state estimate based on nonlinear transformations.

In VO, the EKF [22] uses this linearization to predict the camera's pose by extrapolating from previous states and motion dynamics. When new visual features are observed, the EKF adjusts its state estimate and covariance matrix to optimize the alignment between predicted and observed feature positions. However, the accuracy of the EKF heavily relies on the quality of the linearization, which can degrade if the system undergoes significant nonlinearities or if the initial estimate is poor.

3. Unscented Kalman Filter (UKF):

   UKF [23] improves upon the EKF by addressing the limitations of linearization. Instead of linearly approximating the state space, the UKF employs a deterministic sampling approach known as the Unscented Transform. This technique selects a set of representative sample points, called sigma points, which capture the mean and covariance of the state distribution more accurately than linearization.

   In VO applications, the UKF uses these sigma points [24] to propagate the state estimate through nonlinear transformations, such as camera pose changes over time. By integrating the sigma points into the state prediction and update steps, the UKF maintains a more accurate representation of the state distribution in nonlinear scenarios. This makes the UKF more robust in VO tasks compared to the EKF, especially when handling complex camera motions or

varying visual environments.

4. Particle Filter (PF): PF [25] offers a non-parametric approach to state estimation in VO, particularly suited for highly nonlinear and non-Gaussian scenarios. Instead of maintaining a single state estimate, the PF represents the state distribution using a set of particles, each with its own pose hypothesis and associated weight. These particles evolve over time according to the motion model and are updated based on observed visual features.

   In VO, the PF initializes particles with potential camera poses and adjusts their weights based on the likelihood of observed feature positions. As the camera moves, particles that better align with observed features are assigned higher weights, while less probable particles are discarded. By propagating a large number of particles through the sequence of images, the PF provides a robust estimate of the camera's trajectory, effectively handling uncertainties and nonlinearities inherent in VO tasks.

5. Multi-State Constraint Kalman Filter (MSCKF): MSCKF [26] enhances traditional KF-based VO methods by incorporating multiple states and constraints into the estimation process. Unlike standard KF approaches that track only the current state, the MSCKF maintains a sliding window of past camera poses and feature observations as constraints. These constraints are used to reduce drift and improve pose accuracy over time, especially in environments with high visual feature density or rapid camera motion.

   In VO applications, the MSCKF optimizes camera poses by jointly estimating the current state and leveraging constraints from previous states [27] within the sliding window. This approach enhances pose estimation by integrating in-

formation from multiple viewpoints and motion dynamics observed over time. By updating the state estimate based on both current and historical constraints, the MSCKF achieves improved accuracy and robustness compared to single-state filtering methods

2.1.2 Optimization based approach

Optimization-Based VO methods[18] represent a category of techniques that address pose estimation through the lens of optimization theory. These approaches aim to iteratively refine the camera pose estimation by minimizing the reprojection error of observed features [28]. This section explores the principles of optimization-based VO[29], particularly focusing on graph optimization and pose graph optimization.

Graph Optimization: Graph optimization [30]is a fundamental technique within VO that leverages graph-based representations to solve complex spatial estimation problems. In the context of VO, the graph consists of nodes and edges:

1. Graph Representation: Nodes in the graph correspond to camera poses $\mathbf{X}_i$, where i indexes the frames or keyframes in the sequence. Edges capture the relative pose constraints $\mathbf{C}_{ij}$ between consecutive frames or loop closures.

   Nodes: Represent entities such as camera poses and landmarks.

   Edges: Represent spatial constraints between these entities, typically derived from visual feature correspondences or other sensor measurements.

2. **Objective Function:** The optimization process seeks to minimize the overall error in the pose graph. This error is usually defined as the sum of reprojection errors of the visual features observed across the sequence. The objective function can be formulated as:

$$\min_{\{\mathbf{X}_i\}} \sum_{(i,j)\in\mathcal{E}} \|\mathbf{e}_{ij}\|^2$$

where $\mathcal{E}$ denotes the set of edges representing relative pose constraints, $\mathbf{e}_{ij}$ represents the reprojection error between poses $\mathbf{X}_i$ and $\mathbf{X}_j$.

3. **Optimization Process** Various optimization algorithms such as Gauss-Newton, Levenberg-Marquardt, or nonlinear least squares are employed to iteratively minimize the objective function. These algorithms adjust the camera poses $\mathbf{X}_i$ to reduce the discrepancy between observed and predicted feature positions.

2.1.3 Geometry based approach

In this subsection, we explore several state-of-the-art geometry-based methods for visual odometry in detail. These methods are as followed:

1. VINS Mono [31] integrates visual and inertial sensors to estimate camera motion using monocular images. This approach combines the strengths of visual feature tracking with inertial measurements from gyroscopes and accelerometers to achieve robust pose estimation. By tightly coupling visual information with inertial data, VINS Mono excels in scenarios where visual cues may be sparse or ambiguous, such as fast-moving environments or areas with limited features. The algorithm operates in real-time, making it suitable for applications requiring accurate localization in GPS-denied environments. It addresses challenges like drift and scale ambiguity common in monocular systems by leveraging inertial measurements for improved accuracy and robustness.

2. ORB-SLAM [32] is a feature-based SLAM method designed for real-time camera localization and mapping. It employs ORB features for fast and robust feature detection and matching across frames. ORB-SLAM builds a map of the environment and estimates the camera trajectory concurrently using a keyframe-based approach. This method is particularly effective in environments with dynamic lighting conditions and is capable of running on standard

CPU-based systems in real-time. ORB-SLAM's key strengths lie in its ability to handle feature-rich environments and its robustness to lighting changes, making it suitable for applications ranging from indoor navigation to augmented reality.

3. MONO SLAM [33] focuses on using a single camera to estimate camera pose and map the environment simultaneously. It relies on feature tracking and bundle adjustment techniques to refine camera pose estimates and reconstruct the scene's structure in real-time. MONO SLAM is cost-effective as it requires only a monocular camera setup, making it suitable for lightweight platforms and scenarios where hardware constraints are a concern. While it may face challenges in environments with low-texture or repetitive patterns, MONO SLAM's simplicity and efficiency make it an attractive choice for applications in robotics, augmented reality, and autonomous vehicles.

4. SVO [34] introduces a fast and efficient monocular VO technique based on semi-direct methods. Unlike traditional feature-based methods that rely heavily on sparse feature tracking, SVO combines direct methods for dense depth estimation with sparse feature tracking to achieve real-time performance. This hybrid approach allows SVO to leverage both the accuracy of direct methods in texture-rich regions and the speed of feature-based methods in sparse environments.

   The key innovation of SVO lies in its ability to perform direct sparse odometry by directly minimizing photometric errors between consecutive frames while maintaining a sparse map representation. By using photometric error minimization, SVO robustly handles challenging visual conditions such as motion blur, dynamic scenes, and illumination changes. The method also incorporates efficient keyframe selection and bundle adjustment techniques to optimize cam-

era poses and map consistency in real-time.

2.2 Data-driven visual odometry

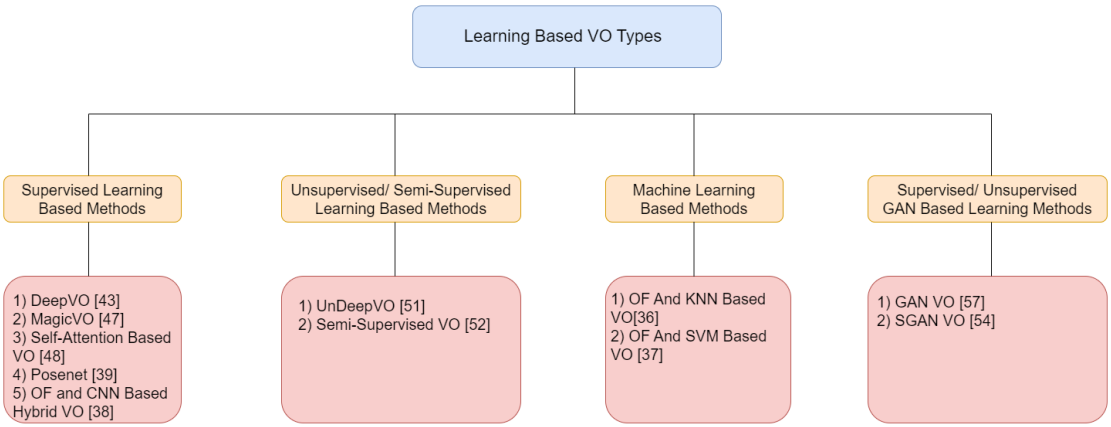

Figure 2.2: Types of learning-based vo

In the realm of robotics and computer vision, the approach to pose estimation has undergone a significant shift with advancements in CNNs. Traditionally, geometry-based methods heavily relied on intricate feature detection and matching, along with camera intrinsic and extrinsic parameters, to estimate the pose of objects or robots within a scene. However, with the rise of CNNs, this reliance on explicit feature extraction has diminished. CNNs have emerged as powerful feature extractors, capable of extracting relevant patterns directly from raw sensor data without the need for explicit feature detection. This paradigm shift has revolutionized learning-based visual odometry approaches, where DL models can now directly learn from sensor readings, effectively bypassing the geometric preprocessing that was previously considered essential (see Figure 2.3).

The integration of CNNs into pose estimation frameworks has not only streamlined the process but has also opened new avenues for research and application in

robotics and computer vision. By leveraging CNNs as powerful feature extractors, the computational burden of explicit geometric preprocessing has been alleviated, leading to more scalable and adaptable systems. Furthermore, the adoption of DL models in visual odometry has enabled the development of more robust and versatile systems capable of handling diverse environmental conditions and sensor modalities. In this section, I will examine foundational literature that underpins the evolution of visual odometry, highlighting the progressive integration of ML and DL techniques over time. I will elucidate the pivotal role of ViT architectures in shaping the future of learning-based visual odometry, offering unparalleled efficiency and scalability for pose estimation tasks in diverse real-world environments.

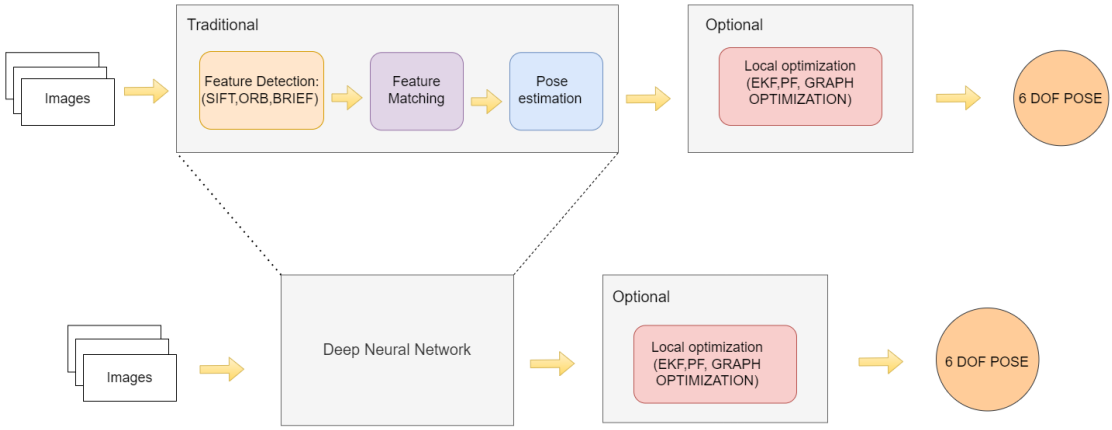

Figure 2.3: System architecture of learning based vo

Moreover, optical flow [35], a technique used to predict sparse or dense pixel movement between consecutive frames, has played a crucial role in feature detection for localization tasks. In machine learning (ML) approaches, optical flow was utilized to create datasets based on dense or sparse optical flow information, capturing the motion patterns within the scene. These datasets were then employed in conjunction with methods like KNN and SVM [36], [37] for localization tasks. By leveraging optical flow information, ML algorithms could effectively discern spa-

tial relationships and motion patterns, enabling accurate localization of objects or robots within dynamic environments. In Figure 2.2, types of learning-based VO are shown.

Gabriele Consante [38] proposed a purely CNN based approach for VO utilizing a monocular camera. Their study introduced three distinct architectures for pose estimation, each exploring different aspects of optical flow integration. Two of these architectures investigated the impact of global and local optical flow separately on pose estimation accuracy. Meanwhile, the third architecture leveraged both local and global optical flow information, combining them to enhance the overall performance of the pose estimation process. By examining various configurations and strategies for integrating optical flow within CNN architectures, Gabriele Consante's work provided valuable insights into improving the robustness and accuracy of monocular camera-based visual odometry systems.

Alex and Matthew [39] introduced the PoseNet model, an end-to-end CNN architecture designed for pose estimation tasks. In their approach, the authors utilized the GoogleNet model renowned for its inception modules, originally developed for image classification tasks. However, they repurposed this architecture by discarding its classification head and instead implemented a pose regression head. This novel design enabled the network to output a 7-dimensional vector comprising three components for position and four for orientation estimation. By leveraging the powerful feature extraction capabilities of GoogleNet and tailoring it for pose regression, the PoseNet model demonstrated effectiveness in accurately estimating the spatial positioning and orientation of objects or cameras within a scene. In Figure 2.3, System architecture of Learning Based VO method is shown.

These above-mentioned Methods solely relying on CNNs or OF for visual odom-

Figure 2.4: Figure shows the internal architecture of the DNN module in figure 2.3

etry often excels at feature extraction but may overlook the temporal relationships between consecutive frames, leading to decreased accuracy. To address this limitation, Long Short-Term Memory (LSTM) [40] and its variants such as Recurrent Neural Network (RNN) [41], Bidirectional Long Short-Term Memory (Bi-LSTM) [42] are employed to capture temporal information effectively from sequential images.By integrating recurrent neural networks like LSTM into VO frameworks, models can better comprehend the temporal dynamics of the scene, thereby enhancing the robustness and accuracy of pose estimation over time. Sen Wang and Ronald Clark [43] introduced the first end-to-end VO model based on a DRCNN, which they called DeepVO. Their approach utilized an encoder-decoder architecture, wherein a pre-trained FlowNet [44], a CNN-based optical flow predictor, served as the encoder, while an LSTM network acted as the decoder. Incorporating the encoder-decoder structure aimed to capture spatial and temporal information crucial for accurate pose estimation in VO systems. They also mentioned that DeepVO performs well compared to VISO2M [45], but it does not perform as well as VISO2S, a traditional VO method using monocular and stereo cameras. Additionally, they noted that in sequence 12, DeepVO struggles due to the featureless environment and the presence of many moving objects in the camera's field of view, which causes issues in accurate pose estimation.In Figure 2.4, pipeline forlearning-based VO are shown which uses encider-decoder architecture for pose estimation.

	DeepVO	**MagicVO**	**UnDeepVO**	**Pose Net**	**Proposed (ViTVO)**
Encoder	Flownet [44]	Flownet [44]	CNN [14]	Google-net [46]	-
Decoder	LSTM [40]	Bi-LSTM [42]	-	-	-
Transformer	-	-	-	-	ViT [15]

Table 2.1: Architectural differences between the different components

Building upon this framework, Jian Jiao [47] conducted similar experiments but opted for a Bi-LSTM as the decoder component. This choice was driven by the Bi-LSTM's superior ability to capture temporal dependencies, offering a more reliable mechanism for integrating temporal information into the VO model. By refining the decoder architecture, Jiao's work enhanced the robustness and accuracy of end-to-end VO models. By simply replacing the decoder with a Bi-LSTM, they witnessed a slight improvement in terms of rotation and translation error, which decreased by 1 to 1.5

Hamed Damirchi [48] have augmented their VO models by incorporating attention mechanisms [49] between the CNN and LSTM components. These attention mechanisms help the model focus on relevant spatial and temporal features, thereby enhancing pose estimation accuracy. By dynamically weighting the importance of different parts of the input sequence, attention mechanisms enable the VO model to effectively capture the most informative cues for robust motion estimation. N. Kaygusuz [50] employs a familiar CNN-RNN hybrid architecture and an unsupervised learning framework, complemented by incorporating a Mixture Density Network (MDN). This MDN facilitates the camera motion estimation by modeling it as a mixture of Gaussians, utilizing the extracted spatio-temporal representations to provide robust and nuanced predictions. Such a framework enhances the model's ability to capture complex motion patterns and uncertainty, contributing to more accurate and reliable visual odometry and depth estimation results.

UnDeepVO [51] presents a novel approach to addressing the VO task unsuper-

vised. The architecture of UnDeepVO is trained using pairs of stereo images to estimate the depth of scenes. Leveraging this depth information, the model learns visual odometry without direct supervision. While ground truth data is employed for the depth estimation task, the model learns visual odometry in an unsupervised manner, without explicit annotations for pose estimation. By integrating depth estimation and visual odometry within a unified framework, UnDeepVO demonstrates the potential to advance the field of autonomous navigation and scene understanding by reducing the reliance on manual annotations and facilitating more robust and scalable learning approaches.

D. Chen [52] introduces a pioneering method for VO by employing a semi-supervised deep learning framework. VO involves estimating the camera's motion relative to its environment, typically requiring substantial amounts of labeled data for training, which can be costly and time-intensive to obtain. However, this paper proposes a novel approach by harnessing both labeled and unlabeled data during training, leveraging semi-supervised learning principles. By incorporating unlabeled data, the model can extract additional features and patterns from the data distribution, enhancing generalization and robustness. This innovative strategy presents a promising avenue for advancing VO systems, offering improved performance without the need for extensive manual labeling efforts.

Y. Almalioglu [51]introduces a pioneering method that leverages Generative Adversarial Networks (GANs) [53] for unsupervised deep monocular VO and depth estimation. The proposed framework utilizes the adversarial training paradigm to simultaneously learn depth estimation and visual odometry tasks from unlabeled monocular image sequences. By employing GANs, the model can effectively capture the complex relationship between consecutive frames and estimate accurate depth maps while also inferring camera motion. This unsupervised approach eliminates

the need for costly labeled data, making it more scalable and practical for real-world applications. T. Feng [54] introduces a novel method employing stacked generative adversarial networks (SGANs) [54]for unsupervised deep visual odometry and depth estimation. This approach leverages the power of adversarial training to simultaneously learn accurate depth maps and camera motion from unlabeled monocular image sequences.

2.3 Limitations and gaps in current approaches

Figure 2.5: High-level architecture of proposed ViTVO

Deep Neural Networks (DNNs) are revolutionizing VO by consolidating traditional steps such as feature detection, feature matching, and camera calibration into a single end-to-end block, as illustrated in Figure 1. This streamlined approach eliminates the need for intricate manual processes, instead allowing the network to learn directly from raw sensor data to predict camera poses. Inside the DNN block, there are two sub-blocks:Convoluation Neural Networks (CNNs) [14] for robust feature extraction and sequential information capturing blocks (Long Short-Term Memory (LSTM) [40], Recurrent Neural Network (RNN) [41], Bidirectional Long Short-Term Memory (Bi-LSTM) [42]) as shown in the figure 2. By integrating these blocks, it becomes possible to achieve end-to-end visual odometry directly from RGB images, enabling comprehensive learning of spatial and temporal relationships for accurate pose estimation. Additionally, optical flow techniques, which predict pixel movement between frames, have been instrumental in feature detection for localization

tasks within ML.

You can refer to the table to see how previous learning-based methods utilize various encoder-decoder blocks to achieve enhanced performance. In previous learning-based approaches, integrating separate blocks for feature extraction and sequential information capture is common. However, leveraging ViT architectures allows for the consolidation of these blocks into a single end-to-end learning framework for VO, as depicted in Figure 3. This unified approach enables ViT to directly learn and infer camera poses from RGB images, streamlining the VO process while potentially enhancing accuracy and efficiency. Over recent years, transformers, rooted in attention mechanisms, have become foundational in state-of-the-art models for natural language processing, surpassing traditional models like RNNs, LSTMs, and Bi-LSTMs. Inspired by the success of transformer-based architectures such as Bidirectional Encoder Representations from Transformers (BERT) [55] and Generative Pre-trained Transformer (GPT) [56], researchers have begun applying transformers to vision tasks, resulting in remarkable advancements. ViTs have achieved state-of-the-art accuracy in tasks ranging from image recognition to object detection, demonstrating the transformative potential of transformers across various domains in computer vision.

After a thorough analysis of literature on learning-based odometry, particularly focusing on deep learning approaches, this paper introduces an end-to-end architecture centered around ViT. ViT is renowned for its capability to extract hierarchical information from sequences, effectively capturing the chain of knowledge in visual data. The proposed architecture aims to directly estimate the 6 degrees of freedom (DOF) pose from raw images, leveraging ViT's ability to integrate spatial and temporal knowledge harmoniously. In adapting ViT for this task, the classification head, typically used for image classification tasks, was replaced with a regression head. This modification aligns the model's architecture with the regression nature

of visual odometry, optimizing performance by adjusting the loss function to mean squared error for accurate pose estimation rather than classification. Figure 2.5 shows how ViT is replacing the encoder-decoder architecture with a single ViT block.

2.4 Advantages of ViT in VO

The following are the advantages of using ViT in VO:

(1). Global Context Understanding:

Transformers excel in capturing global dependencies across the entire sequence of images. In VO, where maintaining spatial and temporal coherence is crucial for accurate pose estimation, Transformers can effectively model these dependencies without relying on fixed local receptive fields as in CNNs.

(2). Attention Mechanism:

The core mechanism in Transformers, the self-attention mechanism, allows them to selectively focus on relevant parts of the input sequence. This is beneficial in VO tasks where different frames may contribute differently to the estimation of camera motion, especially in dynamic scenes or environments with varying lighting conditions.

(3). Sequence Modeling:

Unlike CNNs that process images spatially, Transformers process sequences directly, treating images as sequences of patches. This approach preserves spatial relationships across frames while leveraging the Transformer's ability to model sequential data effectively.

(4). Adaptability to Non-Uniform Sampling:

Transformers are naturally suited to handle sequences with non-uniform sampling rates or irregular time intervals, which can occur in VO due to varying frame rates or occlusion events.

(5). Hierarchical Representation:

Transformers can learn hierarchical representations of visual data, capturing both low-level features such as edges, textures and high-level semantics such as objects, scenes. This capability is advantageous in VO for robustly estimating camera poses across different scales and complexities in scenes.

2.5 Applications of ViT in VO

The following are the Applications of using ViT in VO:

(1). Pose Estimation:

Transformers can predict the camera's pose (position and orientation) relative to a starting point using sequences of images. This is crucial for tasks such as autonomous navigation, robot localization, and augmented reality, where precise understanding of camera motion is essential.

(2). Semantic Understanding:

Transformers can incorporate semantic information from images, facilitating tasks such as vo and simultaneous localization and mapping (SLAM) where understanding the scene's structure aids in better pose estimation .

(3). Dynamic Environments:

Transformers' ability to handle dynamic scenes, occlusions, and changes in lighting conditions makes them suitable for VO applications in real-world environments where conditions may vary unpredictably.

3 Methedology

In this chapter, section 3.1 establish the theoretical foundation for integrating Vision Transformer architectures into VO. It discusses the rationale for leveraging these models, highlighting their ability to capture long-range dependencies and global contextual information in VO tasks. Section 3.2, 3.3 and 3.4 includes a brief description of the KITTI dataset used for training and evaluation, outlines the training procedure, experimental setup, and evaluation metrics, and provides a detailed exploration of various aspects such as losses.

3.1 Model architecture

The ViTVO architecture consists of the following components:

3.1.1 Patch embedding:

First, a 224×224 image $\mathbf{I}$ is divided into patches, each of size 16×16. This results in $N = \frac{224 \times 224}{16 \times 16} = 196$ patches. Each patch is then flattened and projected to a vector of dimension D using a linear projection layer. Figure 3.1 shows how patch embedding and positional embedding help ViT to understand spatial as well as temporal information from images. This system architecture 3.1 illustrates the data flow with arrows and the processing functions with rectangular blocks, which transform the data into more abstract representations. This enables the model to harmoniously extract spatial and temporal information. The operation can be

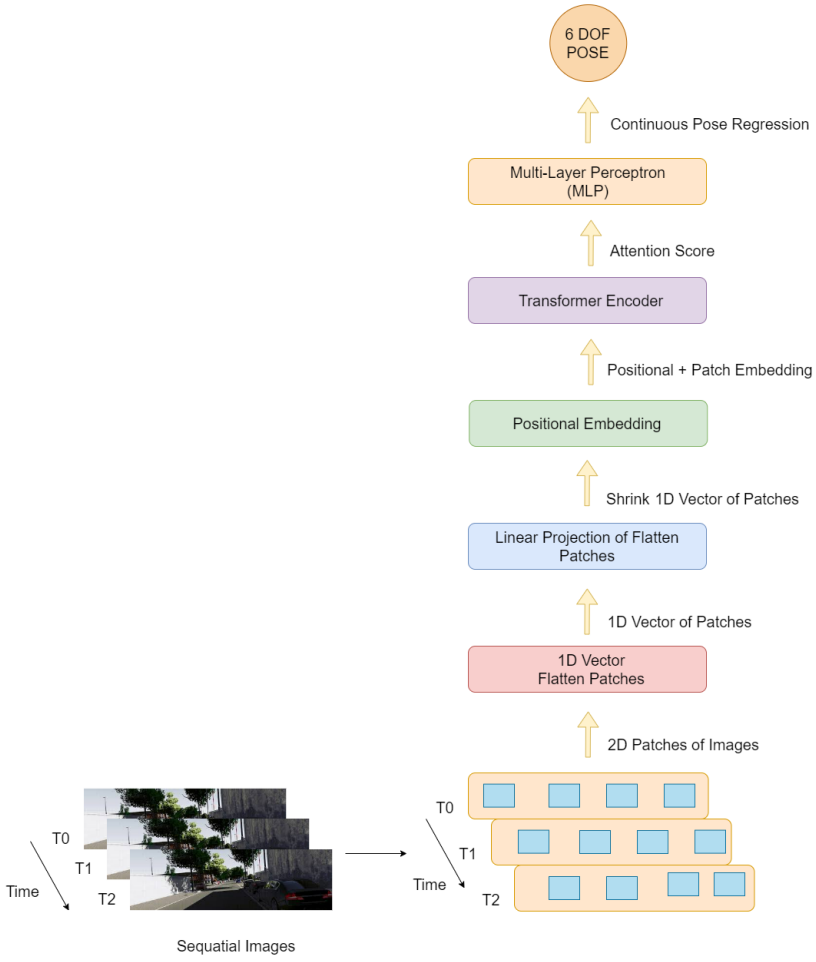

Figure 3.1: System architecture of the proposed ViTVO

expressed as:

$$\mathbf{X} = \text{PatchEmbedding}(\mathbf{I})$$

$$\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N] \quad \text{where} \quad \mathbf{x}_i \in \mathbb{R}^D$$

Positional Embedding used Positional embeddings are added to these patch embeddings to retain positional information:

$$\mathbf{Z}_0 = [\mathbf{x}_1 + \mathbf{p}_1, \mathbf{x}_2 + \mathbf{p}_2, \ldots, \mathbf{x}_N + \mathbf{p}_N]$$

A common choice for positional embeddings is the sinusoidal positional embedding, defined as:

$$\mathbf{p}_i^{(2j)} = \sin\left(\frac{i}{10000^{2j/D}}\right)$$

$$\mathbf{p}_i^{(2j+1)} = \cos\left(\frac{i}{10000^{2j/D}}\right)$$

where i is the position index, j is the dimension index, and D is the embedding dimension.

Importance in VO: Patch embedding and positional embedding are crucial components in ViT-based VO as they enable the model to effectively process and interpret visual data. Patch embedding divides an input image into fixed-size patches and projects each patch into a high-dimensional space using a linear transformation, converting the image into a sequence of patch embeddings. This transformation allows the ViT to handle images similarly to how it processes sequences in natural language processing. However, unlike sequences of words, the spatial arrangement of image patches is vital for understanding visual context. Positional embedding addresses this by adding positional information to each patch embedding, ensuring the model retains the spatial relationships between patches. This combined approach allows the ViT to capture both local and global dependencies in the visual data, which is essential for accurate and robust pose estimation in VO tasks, especially in

complex and dynamic environments.

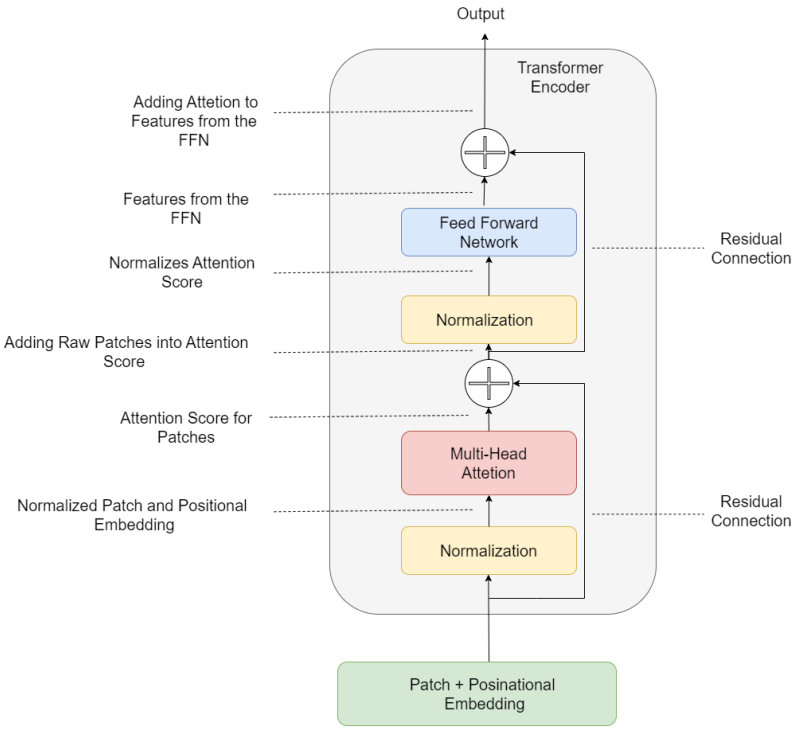

Figure 3.2: Figure shows the internal architecture of the transformer encoder in the figure 3.1

3.1.2 Transformer encoder layers:

Each Transformer encoder layer processes the input through two main components: Multi-Head Self-Attention (MHSA) and Feed-Forward Network (FFN). The output from the l-th layer serves as the input to the $(l+1)$-th layer. Figure 3.2 shows internal architecture of transformer encoder. Here, The dashed lines indicate comments on operations between functional blocks, while the arrows represent the data flow and the rectangular blocks denote functions that transform data into more abstract information, enabling the model to extract harmonious information between spatial and temporal dimensions. Following bullets explain use of multi-head self-attention and feed-forward networks (FFN) in ViTVO in detail.

1. Multi-head attention: Multi-head attention is another critical component in ViTs that allows the model to capture various aspects of the scene simultaneously. By utilizing multiple attention heads, the model can attend to different parts of the image in parallel, with each head focusing on different features or patterns. This parallel processing enhances the feature representation by aggregating diverse information from different perspectives, thereby enriching the model's understanding of the scene. In VO applications, multi-head attention helps capture the complex and dynamic nature of real-world environments, leading to improved robustness and reliability in pose estimation. It ensures that the model can extract and combine relevant features in the temporal dimension, which is crucial for maintaining accuracy in challenging conditions.

   Let $\mathbf{H}^{(l)}$ be the input to the l-th layer. The multi-head self-attention is defined as:

$$\mathbf{H}_{\text{MHSA}}^{(l+1)} = \text{MHSA}(\mathbf{H}^{(l)})$$

2. Self-attention mechanism: Self-attention allows the model to weigh the importance of different parts of the input sequence when encoding a particular token. It calculates the attention scores between all pairs of tokens and generates a weighted sum of the token representations. This mechanism is crucial for capturing dependencies and relationships within the input data, which is essential for accurate visual odometry.

   Self-attention mechanisms are a pivotal feature in ViTs that allow the model to weigh the importance of different patches within an image. By computing the relationships between each patch and every other patch, self-attention dynamically adjusts the influence of each patch based on its relevance to the task at hand. This capability enhances the integration of contextual informa-

tion across the entire image, enabling the model to understand global patterns and dependencies. In the context of VO, self-attention mechanisms help in accurately estimating the camera pose by effectively considering the spatial relationships and contextual cues from various parts of the scene.

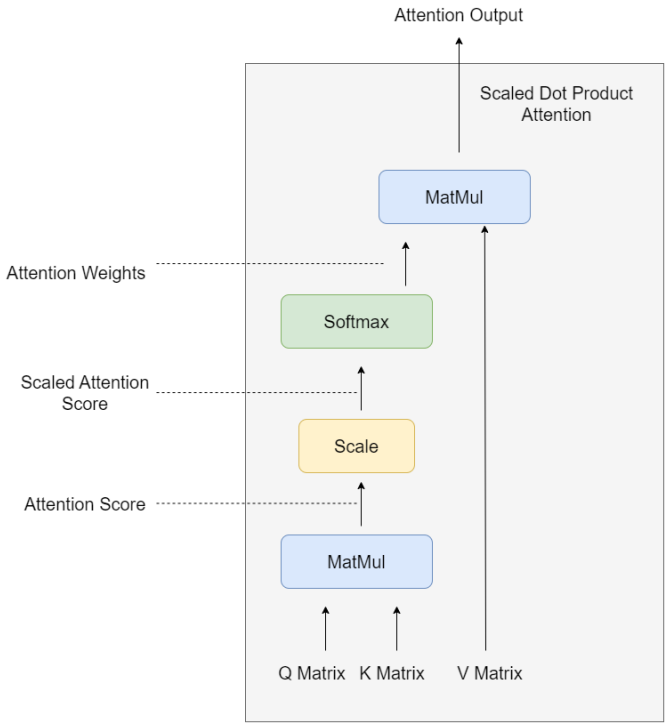

Figure 3.3: Figure shows the internal architecture of MHSA in figure 3.2

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d_k}}\right)\mathbf{V}$$

In the scaled dot-product attention mechanism, attention is computed by first taking the query matrix Q and multiplying it with the transpose of the key matrix K. This product is then normalized by dividing by $\sqrt{d_k}$, where d_k is the dimension of the keys. The result is passed through a softmax function to obtain the attention weights, which are then multiplied by the value matrix V to generate the attention output. This process is illustrated in Figure 3.3,the dashed lines indicate comments on operations between functional blocks, while

the arrows represent the data flow and the rectangular blocks denote functions that transform data into more abstract information which highlights its role in balancing spatial and temporal information crucial for VO.

For MHSA, the attention mechanism is applied in parallel across multiple heads. Each head computes attention independently, resulting in multiple attention outputs. These outputs are concatenated and linearly transformed using a weight matrix W_O. This can be represented as:

$$\text{MHSA}(H^{(l)}) = \text{Concat}(\text{head}_1, \text{head}_2, \ldots, \text{head}_h)W_O$$

where head_i denotes the output of the i-th attention head. This approach enables the model to capture and integrate various aspects of the input data, enhancing its ability to handle complex visual odometry tasks.

where each head_i is computed as:

$$\text{head}_i = \text{Attention}(\mathbf{H}^{(l)}\mathbf{W}_i^Q, \mathbf{H}^{(l)}\mathbf{W}_i^K, \mathbf{H}^{(l)}\mathbf{W}_i^V)$$

3. Feed-forward network (FFN):

   The output of layer normalization is then passed to the Feed-Forward Network (FFN), which helps to stabilize and accelerate training, allowing the model to reach convergence more quickly. This network consists of two linear transformations with a ReLU activation in between. It independently processes each token representation (output from MHSA) to introduce non-linearity and further refine the representation.

   Mathematically, this can be expressed as:

$$\mathbf{H}^{(l+1)} = \text{FFN}(\mathbf{H}_{\text{MHSA}}^{(l+1)})$$

where the FFN is defined as:

$$\text{FFN}(\mathbf{x}) = \text{ReLU}(\mathbf{x}\mathbf{W}_1 + \mathbf{b}_1)\mathbf{W}_2 + \mathbf{b}_2$$

Combining both components, the complete operation of the l-th Transformer encoder layer is:

$$\mathbf{H}^{(l+1)} = \text{MHSA}(\mathbf{H}^{(l)}) + \text{FFN}(\text{MHSA}(\mathbf{H}^{(l)}))$$

4. Residual connections and normalization

   Shortcut connections (or residual connections) are added around each sub-layer (MHSA and FFN), allowing gradients to flow more easily during back-propagation(See Figure 3.2), which helps in training deeper networks. Layer normalization is applied after each sub-layer along with the residual connections, ensuring that the outputs are well-behaved (normalized). Sequential Processing: The output of the l-th encoder layer serves as the input to the $(l + 1)$-th layer. This sequential stacking of layers allows the model to progressively build more abstract and higher-level representations of the input data.

3.1.3 Pose regression head

The final Transformer encoder layer output is passed through a multi-layer perceptron (MLP) head to regress the 6-DoF pose (3D translation and Euler angles for rotation).We use a pretrained ViT model from Hugging Face in this approach. The original classification head of the ViT model is discarded, and a pose regression head is mounted on it. This new head is designed to regress the 6-DoF pose (3D translation and Euler angles for rotation). The pose regression head can be defined

as:

$$\hat{Y} = \text{MLP}(H^{(L)})$$

where $\hat{Y} \in \mathbb{R}^6$ represents the estimated 6-DoF pose.

After mounting the pose regression head, we fine-tune some layers of the ViT model along with the newly added pose regression head. This fine-tuning process helps the model adapt to the specific task of pose estimation while leveraging the pre trained knowledge from the original ViT model.

After the pose regression head computes the estimated pose $\hat{Y}$, the model calculates the loss using the Mean Squared Error (MSE) between the estimated poses and the ground truth poses. The MSE loss function is defined as:

$$\text{MSE Loss} = \frac{1}{N} \sum_{i=1}^{N} \left(\hat{Y}_i - Y_i \right)^2$$

where N is the number of samples, $\hat{Y}_i$ is the estimated pose, and Y_i is the ground truth pose.

To optimize the model parameters, we use the Adam optimizer, which updates the model weights based on the computed gradients. The loss function is backpropagated through the network to update the weights, aiming to minimize the MSE loss. The Adam optimization algorithm adjusts the learning rate adaptively for each parameter, improving convergence speed and performance.

3.2 Data collection

3.2.1 Description of datasets

The KITTI dataset is one of the most widely used datasets for evaluating VO and other computer vision tasks related to autonomous driving. It provides a rich col-

lection of data recorded from a moving vehicle in diverse environments. Here's an explanation of the KITTI dataset and how it can be used for evaluating Vision Transformer-based Visual Odometry:

3.2.2 Overview of the KITTI dataset

The KITTI dataset was collected by the Karlsruhe Institute of Technology and the Toyota Technological Institute at Chicago. It includes various sensor modalities such as stereo cameras, LiDAR, and GPS/IMU. The dataset is divided into different subsets, each designed for specific tasks like odometry, object detection, tracking, and more.

The KITTI dataset is renowned for its dynamic environments and extensive route records, making it a valuable resource for evaluating Vision Transformer-based Visual Odometry (ViT-VO) systems. Each sequence within the dataset represents real-world driving scenarios captured across different urban and rural settings, encompassing diverse challenges such as varying lighting conditions, occlusions, and traffic scenarios. The dataset includes sequences with varying path lengths and average velocities, providing a comprehensive range of motion dynamics for testing and benchmarking ViT-VO models. For instance, sequences can range from several hundred meters to kilometers in length, capturing trajectories that reflect both urban street scenes and highway driving. Average velocities across these sequences vary widely, reflecting realistic driving conditions that are essential for assessing how well ViT-VO models generalize across different speeds and environments. Thus, leveraging the KITTI dataset enables rigorous evaluation of ViT-VO systems under conditions that closely mimic real-world driving scenarios, ensuring robustness and reliability in pose estimation across diverse and complex environments.

(1). Sequences (Routes Records):

The VO subset consists of 22 drive sessions (00-21), each representing a contin-

uous sensor data recording as the vehicle drives through different environments. These drive sessions include urban, rural, and highway settings.

(2). Sensor Data:

Stereo Images: High-resolution left and right camera images (color and grayscale). IMU/GPS Data: Provides ground truth poses and orientation, useful for benchmarking VO algorithms.

(3). Velodyne LiDAR: 3D point clouds capturing the surrounding environment.

(4). Annotations:

Ground truth poses for evaluating the accuracy of VO algorithms. Calibration files containing intrinsic and extrinsic parameters of the sensors

3.2.3 Data preprocessing steps

In this section, we will discuss data loading and preprocessing techniques:

(1). Loading Sequences:

For this example, we'll use the left images from the KITTI odometry dataset as input to the model. The dataset contains stereo image pairs (left and right), but we'll focus on the left images. Each route record (sequence) includes these image pairs along with ground truth poses.

(2). Preprocessing:

The images need to be preprocessed to match the input requirements of the Vision Transformer. This includes resizing the images, normalizing the pixel values, and applying data augmentation. Additionally, the ground truth poses, provided as rotation matrices and translation vectors, need to be converted into a 6-DOF representation for supervised training of the model (Euler angles for rotation and 3D vectors for translation).

3.3 Training procedure

In this section, we outline the training procedure for our Visual Odometry (VO) model using the KITTI dataset. We focus on two critical aspects: the loss functions and the optimization techniques employed during training.

3.3.1 Loss functions

For our Visual Odometry task, the primary objective is to accurately predict the 6-DOF pose (3-DOF for rotation and 3-DOF for translation) of the camera. We utilize the Mean Squared Error (MSE) loss function to achieve this.

Mean Squared Error (MSE) Loss:

The MSE loss measures the average squared difference between the predicted and ground truth poses. It is defined as: The Mean Squared Error (MSE) loss is defined as:

$$\text{MSE Loss} = \frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2$$

The use of MSE loss ensures that the model is penalized more heavily for larger errors, thereby encouraging precise pose predictions.

3.3.2 Optimization techniques

To train our model effectively, we use the Adam optimizer, which combines the advantages of two other popular optimizers: AdaGrad and RMSProp. Adam is particularly well-suited for problems with sparse gradients and noisy data, making it a robust choice for training deep learning models.

The Adam optimizer updates the model parameters based on estimates of the first and second moments of the gradients. The following update rules define it:

(1). Compute the gradients g_t at timestep t.

(2). Update the biased first moment estimate:

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t$$

(3). Update the biased second moment estimate:

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2$$

(4). Compute bias-corrected first and second moment estimates:

$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t}$$

$$\hat{v}_t = \frac{v_t}{1 - \beta_2^t}$$

(5). Update the parameters:

$$\theta_t = \theta_{t-1} - \frac{\eta}{\sqrt{\hat{v}_t} + \epsilon} \hat{m}_t$$

3.3.3 Training and test splits

The training and test splits for evaluating the proposed method involve to assess VO's performance robustness across different datasets and environments. The experiment focuses on evaluating the proposed method using specific sequences for training and testing:

(1). Training Data: Sequences 00, 02, 08, and 09 are used for training. These sequences are chosen because they are relatively long and provide ground truth data necessary for training.

(2). Training Process: Trajectories from the selected sequences are segmented into different lengths to generate a sufficient amount of training data. This results in a total of 7410 samples being used for training.

(3). Testing Data: The trained models are evaluated on sequences 03, 04, 05, 06, 07, and 10 to assess their performance. These sequences were not used during training and serve as a separate dataset for evaluation.

3.4 Evaluation metrics

3.4.1 Relative pose error

Relative Pose Error (RPE) evaluates the accuracy of relative poses estimated between consecutive frames by the VO algorithm. Here's how RPE is understood:
Definition: RPE focuses on assessing the consistency and accuracy of the motion estimates between successive frames.

Metrics:

- **Translation Error (TE):** Measures the discrepancy in translation vectors between consecutive frames. It quantifies how accurately the VO algorithm estimates the distance the camera moves from one frame to the next.

  Calculation: Suppose $\mathbf{t}_{gt}(t)$ denotes the ground truth translation vector at time t, and $\mathbf{t}_{est}(t)$ denotes the estimated translation vector by the VO algorithm at time t.

$$\text{TE} = \sqrt{\frac{1}{N}\sum_{t=1}^{N} \|\mathbf{t}_{gt}(t) - \mathbf{t}_{est}(t)\|_2^2}$$

- **Rotation Error (RE):** Quantifies the differences in orientation (rotation) between consecutive frames. It assesses how well the VO algorithm estimates the angular change of the camera's pose from one frame to the next.

Calculation: Suppose $\mathbf{R}_{gt}(t)$ denotes the ground truth rotation matrix at time t, and $\mathbf{R}_{est}(t)$ denotes the estimated rotation matrix by the VO algorithm at time t.

$$\text{RE} = \sqrt{\frac{1}{N}\sum_{t=1}^{N} \text{trace}((\mathbf{I} - \mathbf{R}_{est}(t)\mathbf{R}_{gt}(t)^T)^2)}$$

where $\text{trace}(\cdot)$ denotes the trace of a matrix, and $\mathbf{I}$ is the identity matrix.

Interpretation: RPE metrics provide insights into how well the VO algorithm estimates relative motions between consecutive frames. Low translation and rotation errors indicate accurate estimation of camera motion, which is critical for applications requiring precise navigation and scene understanding.

3.4.2 Translation error against path length

Translation error e_t measures the discrepancy between estimated and ground truth translations. It varies with path length L, indicating the distance traveled by the camera or vehicle. The relationship can be expressed as:

$$e_t(L) = f_t(L)$$

where $f_t(L)$ is the function describing how translation error changes with path length L.

3.4.3 Rotation error against path length

Rotation error e_r quantifies the difference between estimated and actual rotations, influenced by path length L. This metric is crucial for assessing the rotational accuracy over varying distances, represented as:

$$e_r(L) = f_r(L)$$

where $f_r(L)$ represents the function depicting rotation error with respect to path length L.

3.4.4 Translation error against speed

Translation error e_t also correlates with the speed S at which the camera or vehicle moves. The error varies nonlinearly with speed and can be modeled as:

$$e_t(S) = f_t(S)$$

where $f_t(S)$ denotes the function characterizing translation error against speed S.

3.4.5 Rotation error against speed

Rotation error e_r is sensitive to the velocity S of the camera or vehicle, influencing how accurately rotational changes are estimated. This relationship can be mathematically represented as:

$$e_r(S) = f_r(S)$$

where $f_r(S)$ describes the function describing rotation error concerning speed S.

4 Results

This chapter is primary focus of the thesis. Section 4.1 evaluation covers the comparison of rotation and translation errors across different trajectories for learning-based models. Another section 4.2 evaluates different methods, including ViTVO, against different speed and path length. The evaluation of the VO model is primarily based on RPE, which measures the accuracy of relative poses estimated between consecutive frames. RPE consists of TE and RE, where TE quantifies the discrepancy in translation vectors and RE assesses the differences in orientation. Additionally, evaluation metrics include Translation Error Against Path Length, Rotation Error Against Path Length, Translation Error Against Speed, and Rotation Error Against Speed. These metrics provide insights into how well the VO algorithm estimates camera motion under varying conditions, such as different path lengths and speeds, ensuring robust and precise navigation and scene understanding.

4.1 Rotation and translation error comparison across routes for learning-based models

In this section, we analyze the results across five distinct route trajectories with varying environmental conditions. We compare the performance of state-of-the-art learning-based methods that use a combination of CNN and RNN, as well as standalone CNN models, against ViTVO on these different routes in detail.

4.1.1 Sequence 03

Model	Translation Error Sequence 03	Rotation error Error Sequence 03
DeepVO [43]	12.92	0.048
MagicVO [47]	12.14	0.041
UnDeepVO [51]	15.20	0.170
PoseNet [39]	27.30	0.150
ViTVO [Proposed]	11.21	0.039

Table 4.1: Translation and rotation error for different models for sequences 03

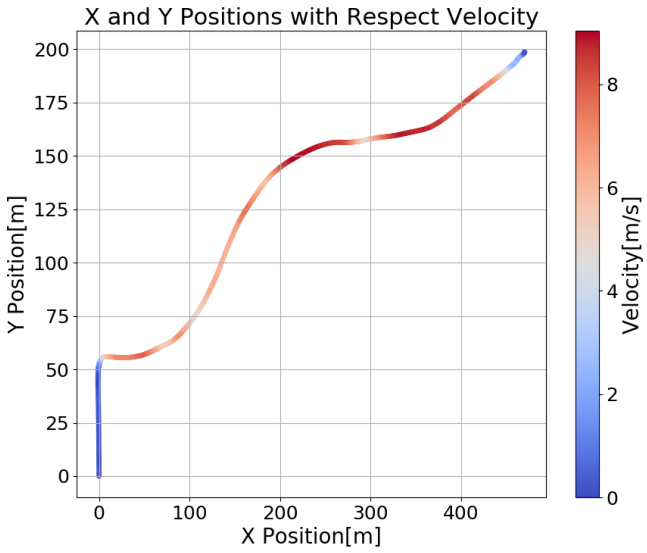

Figure 4.1: Sequence 03 velocity

Sequence 03 typically involves moderate-speed movement with some sharp turns at high speed, as can be seen in Figures 9 and 10, along with varying lighting conditions. The performance of different visual odometry models in this sequence varies significantly based on their ability to handle sequential information. DeepVO performs reasonably well with a translation error of 12.92% and a rotation error of 0.048 (see Table 4.1). This can be attributed to the LSTM's capacity to manage sequential data over short durations. However, MagicVO outperforms DeepVO by exhibiting lower translation and rotation errors, with a translation error of 12.14%

and a rotation error of 0.041 (see Table 4.1). This advantage is due to the Bi-LSTM's bidirectional processing, which captures temporal dependencies more effectively, particularly during sharp turns and changes in lighting conditions. The superior handling of temporal information by MagicVO demonstrates its robustness in complex dynamic environments.

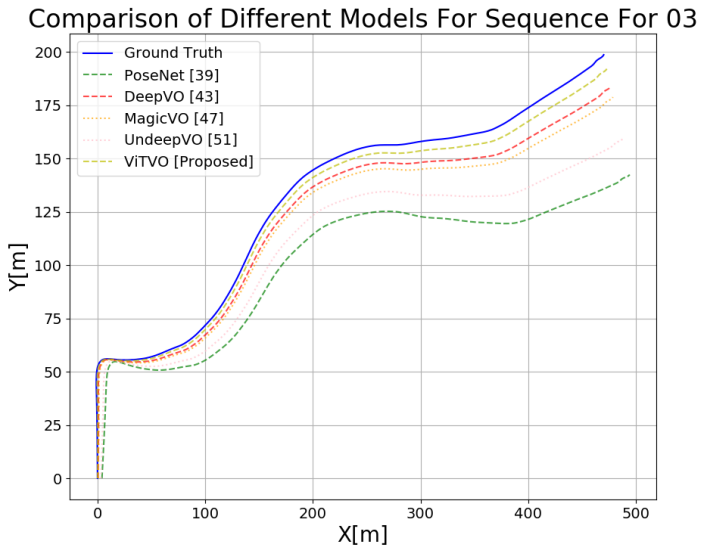

Figure 4.2: Comparison of Different Models for sequence 03

In comparison, PoseNet and UnDeepVO show different performance metrics in Sequence 03. PoseNet demonstrates a translation error of about 27.30%, which is considerably higher, indicating its struggle to accurately estimate poses due to the lack of mechanisms for capturing sequential information. UnDeepVO, while better than PoseNet, still faces challenges with a translation error of approximately 15.20% (see Table 4.1). Despite these limitations, UnDeepVO performs better than PoseNet because it slightly better captures temporal context. The rotation errors for both PoseNet and UnDeepVO hover just above 0.1, with PoseNet at 0.150 and UnDeepVO at 0.170, reflecting some consistency in rotational estimation despite the shortcomings in translation accuracy (see Table 4.1).

On the other hand, ViT shows the lowest translation error for Sequence 03, with a rate closer to 11.21%. This suggests that ViT's architecture is more effective at handling the complexities of the sequence, including moderate-speed movements, sharp turns, and varying lighting. The performance of ViT can be attributed to its advanced capabilities in capturing and processing spatial-temporal information, resulting in more accurate pose estimations compared to the other models. This highlights the critical role of effective sequential data handling in visual odometry, where models like MagicVO and ViT, which incorporate mechanisms for robust temporal information processing, consistently achieve better results (see Table 4.1).

Figure 4.2 shows a comparative analysis of different visual odometry models for Sequence 03. According to Figure 9, the vehicle takes a turn at high speed, which is where drift starts to add. Every model performs poorly during these sharp turns, indicating the challenges in accurately estimating poses in such dynamic conditions. However, ViTVO stands out by effectively extracting hierarchical information from the environment, a crucial feature for handling these complexities. Despite the drift affecting ViTVO as well, it performs significantly better compared to the other models, as can be seen in Figure 4.2. This highlights the superior capability of ViTVO in managing sequential data and dynamic scenarios. The velocity profile in Figure 4.1 clearly illustrates the high-speed turn taken by the vehicle. This is the point where most models, including ViTVO, experience an increase in drift. However, as shown in Figure 4.2, ViTVO's advanced architecture allows it to mitigate the drift more effectively than other models, demonstrating its robustness in challenging scenarios.

4.1.2 Sequence 04

Sequence 04 involves scenarios characterized by moderate-speed movements and varying environmental conditions, posing distinct challenges for visual odometry models. DeepVO achieves a translation error of 10.56% and a rotation error of 0.048

(Table 4.2), demonstrating its capability to manage sequential data effectively over short durations, similar to its performance in Sequence 03. In comparison, MagicVO outperforms DeepVO in Sequence 04 with lower translation (9.74%) and rotation (0.041) errors, leveraging its bidirectional LSTM for enhanced temporal dependency handling (Table 4.2). This highlights MagicVO's robustness in capturing accurate pose estimations amidst varying environmental dynamics.

PoseNet and UnDeepVO exhibit noticeable differences in their performance metrics in Sequence 04. PoseNet struggles with a higher translation error of 19.30%, indicative of its challenges in accurately estimating poses without effective sequential information processing mechanisms (Table 4.2). On the other hand, UnDeepVO shows improved performance over PoseNet, yet faces challenges with a translation error of 12.23% and a rotation error of 0.170. Despite these limitations, UnDeepVO's slightly better capture of temporal context aids in mitigating translation errors to some extent (Table 4.2).

The proposed model, ViTVO, exhibits promising results in Sequence 04 with the lowest translation error of 7.20% and a rotation error of 0.035 (Table 4.2). This signifies ViTVO's advanced architecture in handling spatial-temporal complexities, including moderate-speed movements and varying environmental dynamics. Figure 4.4 presents a comparative analysis of different visual odometry models, highlighting ViTVO's superior performance in accurately estimating poses amidst challenges encountered in Sequence 04. Despite facing drift during high-speed movements, ViTVO demonstrates robust sequential data management, underscoring its effectiveness in dynamic scenarios (Figure 4.4).

PoseNet and UnDeepVO exhibit noticeable differences in their performance metrics in Sequence 04. PoseNet struggles with a higher translation error of 19.30%, indicative of its challenges in accurately estimating poses without effective sequential information processing mechanisms (Table 4.2). On the other hand, UnDeepVO

Model	Translation Error Sequence 04	Rotation error Error Sequence 04
DeepVO [43]	10.56	0.048
MagicVO [47]	9.74	0.041
UnDeepVO [51]	12.23	0.170
PoseNet [39]	19.30	0.150
ViTVO [Proposed]	7.20	0.035

Table 4.2: Translation and rotation error for different models for sequences 04

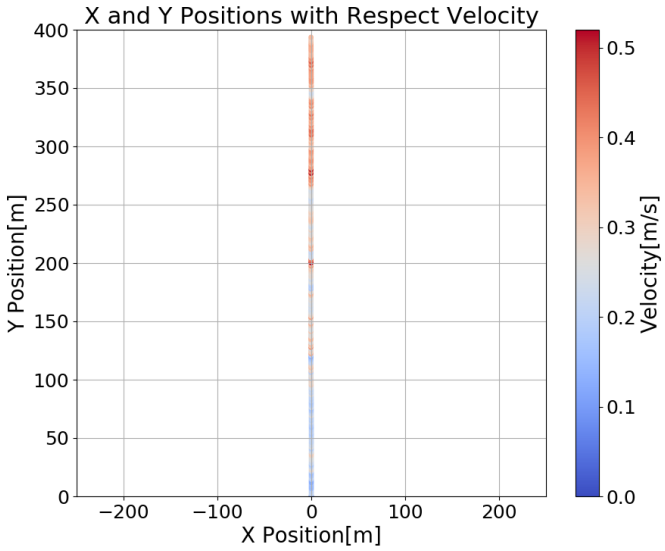

Figure 4.3: Sequence 04 velocity

shows improved performance over PoseNet, yet faces challenges with a translation error of 12.23% and a rotation error of 0.170. Despite these limitations, UnDeepVO's slightly better capture of temporal context aids in mitigating translation errors to some extent (Table 4.2).

The proposed model, ViTVO, exhibits promising results in Sequence 04 with the lowest translation error of 7.20% and a rotation error of 0.035 (Table 4.2). This signifies ViTVO's advanced architecture in handling spatial-temporal complexities, including moderate-speed movements and varying environmental dynamics. Figure 4.4 presents a comparative analysis of different visual odometry models, highlighting ViTVO's superior performance in accurately estimating poses amidst challenges encountered in Sequence 04. Despite facing drift during high-speed movements,

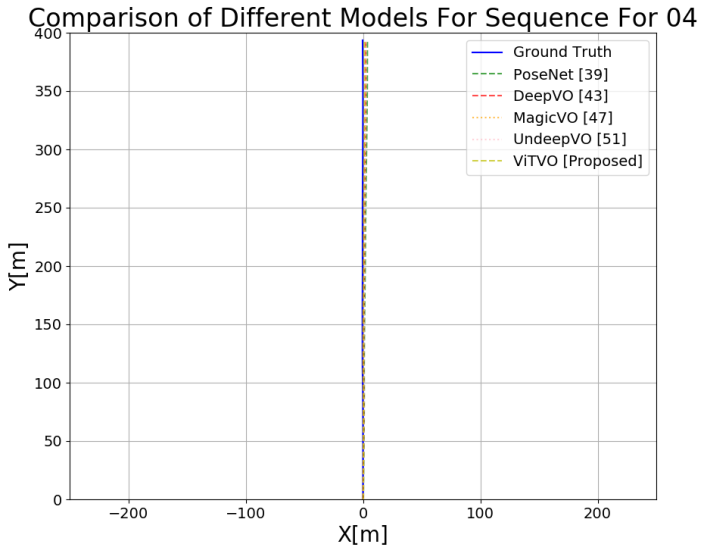

Figure 4.4: Translation error against path length

ViTVO demonstrates robust sequential data management, underscoring its effectiveness in dynamic scenarios (Figure 4.4).

4.1.3 Sequence 05

Sequence 05 presents challenges characterized by dynamic movements and varying environmental conditions(figure 4.5), influencing the performance of visual odometry models. DeepVO demonstrates a translation error of 14.61% and a rotation error of 0.038 (Table 4.3), indicating its ability to handle sequential data moderately well over short durations. In contrast, MagicVO performs slightly better with a translation error of 14.18% and a rotation error of 0.041, leveraging its bidirectional LSTM for enhanced temporal dependency handling (Table 4.3). This underscores MagicVO's robustness in capturing accurate pose estimations despite the dynamic nature of Sequence 05.

PoseNet and UnDeepVO exhibit higher translation errors in Sequence 05, reflecting their challenges in accurately estimating poses under dynamic conditions.

PoseNet records a translation error of 22.30%, highlighting its struggle without effective mechanisms for sequential information processing (Table 4.3). UnDeepVO, though performing better than PoseNet with a translation error of 18.47% and a rotation error of 0.1, still faces significant challenges in managing temporal context (Table 4.3).

Model	Translation Error Sequence 05	Rotation error Error Sequence 05
DeepVO [43]	14.61	0.038
MagicVO [47]	14.18	0.041
UnDeepVO [51]	18.47	0.1
PoseNet [39]	22.30	0.15
ViTVO [Proposed]	12.30	0.026

Table 4.3: Translation and rotation error for different models for sequences 05

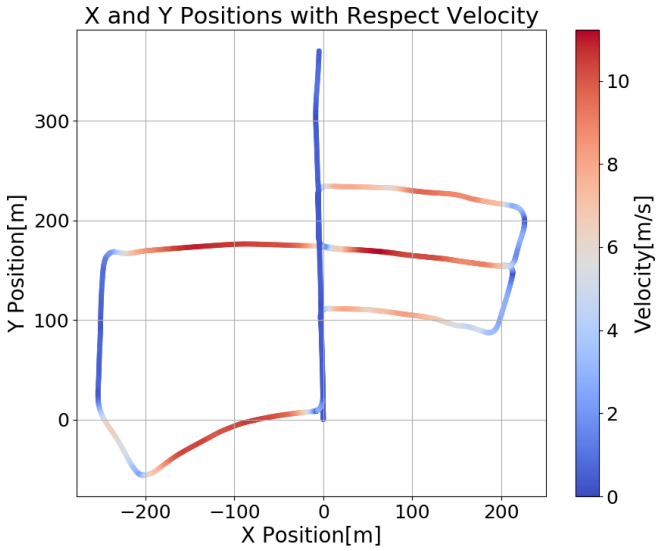

Figure 4.5: Sequence 05 velocity

The proposed model, ViTVO, demonstrates competitive performance in Sequence 05 with a translation error of 12.30% and a rotation error of 0.026 (Table 4.3). Figure 4.6 provides a comparative analysis of different visual odometry models, showcasing ViTVO's effectiveness in handling dynamic movements and varying environmental conditions. Despite inherent challenges such as drift during high-

speed maneuvers, ViTVO excels in sequential data management, highlighting its suitability for dynamic scenarios (Figure 4.6).

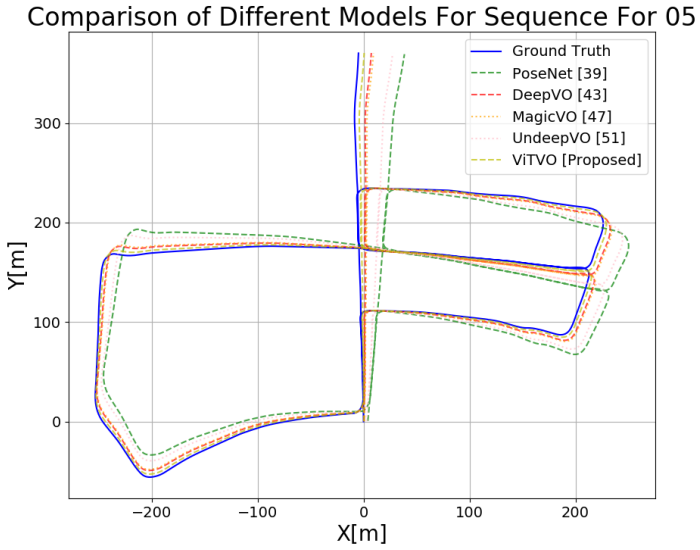

Figure 4.6: Comparison of different models for sequence 05

4.1.4 Sequence 06

Sequence 06 introduces challenges characterized by rapid movements and diverse environmental conditions, significantly influencing the performance of visual odometry models. DeepVO exhibits a translation error of 17.86% and a rotation error of 0.068 (Table 4.4), showcasing its moderate ability to handle sequential data over short durations. In contrast, MagicVO performs marginally better with a translation error of 17.36% and a rotation error of 0.051, leveraging its bidirectional LSTM for more effective temporal dependency management (Table 4.4). This highlights MagicVO's robustness in accurately estimating poses amidst the dynamic elements of Sequence 06.

PoseNet and UnDeepVO face notable challenges in Sequence 06, evidenced by their higher translation errors. PoseNet records a translation error of 24.10%, un-

derscoring its difficulty in accurately estimating poses without effective mechanisms for capturing sequential information (Table 4.4). Similarly, UnDeepVO shows a translation error of 20.60% and a rotation error of 0.19, indicating ongoing struggles in managing temporal context (Table 4.4).

Model	Translation Error Sequence 06	Rotation error Error Sequence 06
DeepVO [43]	17.86	0.068
MagicVO [47]	17.36	0.051
UnDeepVO [51]	20.60	0.19
PoseNet [39]	24.10	0.22
ViTVO [Proposed]	15.28	0.034

Table 4.4: Translation and rotation error for different models for sequences 06

The proposed model, ViTVO, demonstrates promising results in Sequence 06 with a translation error of 15.28% and a rotation error of 0.034 (Table 4.4). ViTVO's effectiveness in handling rapid movements and diverse environmental dynamics. Despite challenges such as increased drift during high-speed actions, ViTVO excels in sequential data processing, highlighting its suitability for dynamic scenario.

4.1.5 Sequence 07

Model	Translation Error Sequence 07	Rotation error Error Sequence 07
DeepVO [43]	14.56	0.090
MagicVO [47]	13.60	0.081
UnDeepVO [51]	17.70	0.21
PoseNet [39]	26.20	0.23
Proposed [ViTVO]	12.70	0.040

Table 4.5: Translation and rotation error for different models for sequences 07

Sequence 07 presents challenges characterized by intricate movements(figure 4.7) varied environmental conditions, significantly impacting the performance of visual odometry models. DeepVO exhibits a translation error of 14.56% and a rotation error of 0.090 (Table 4.5), indicating its moderate capability to manage sequential data over short durations. In contrast, MagicVO performs slightly better with a

translation error of 13.60% and a rotation error of 0.081, leveraging its bidirectional LSTM for more effective handling of temporal dependencies (Table 4.5). This underscores MagicVO's robustness in accurately estimating poses amidst the complexities of Sequence 07.

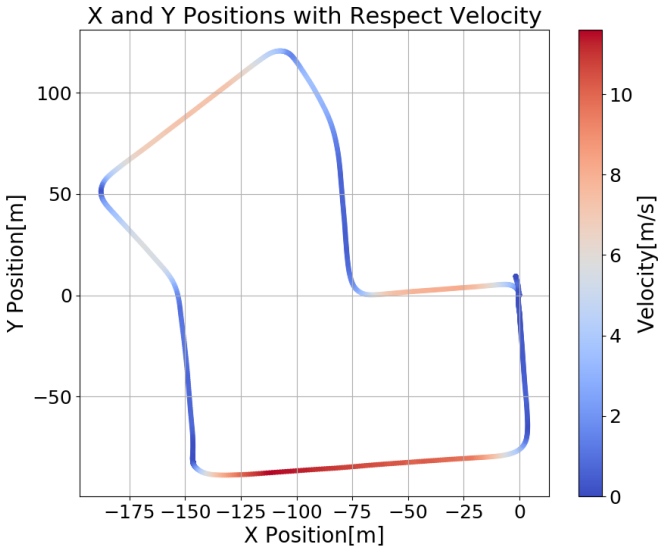

Figure 4.7: Sequence 07 Velocity

PoseNet and UnDeepVO encounter significant challenges in Sequence 07, as reflected by their higher translation errors. PoseNet records a translation error of 26.20%, highlighting its struggle without effective mechanisms for capturing sequential information (Table 4.5). Similarly, UnDeepVO shows a translation error of 17.70% and a rotation error of 0.21, indicating ongoing difficulties in managing temporal context (Table 4.5).

The proposed model, ViTVO, demonstrates competitive performance in Sequence 07 with a translation error of 12.70% and a rotation error of 0.040 (Table 4.5). Figure 4.8 provides a comparative analysis of visual odometry models, illustrating ViTVO's effectiveness in handling intricate movements and varied environmental dynamics. Despite challenges such as increased drift during complex maneuvers,

ViTVO excels in sequential data processing, emphasizing its suitability for dynamic scenarios (Figure 4.8). conditions.

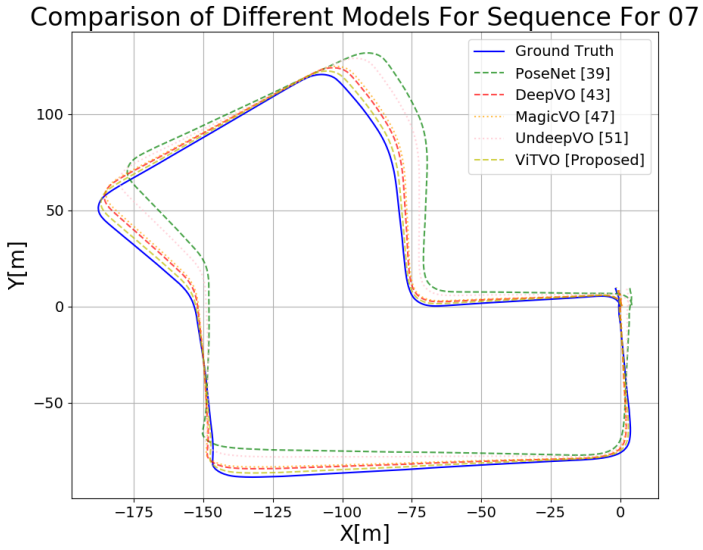

Figure 4.8: Comparison of different models for sequence 07

The following tables summarize the mean, standard deviation, minimum, and maximum values of translation and rotation errors for different visual odometry models across various test sequences.

The figure 4.9 presented provide a detailed comparison of translation and rotation errors for various visual odometry models. In terms of translation error, DeepVO has a mean error of 14.49 with a standard deviation of 2.35, and its errors range from 11.61 to 17.86. MagicVO exhibits a slightly lower mean translation error of 13.40 and a standard deviation of 2.45, with a minimum of 11.18 and a maximum of 17.36. UnDeepVO, however, shows a higher mean translation error of 16.84 and a standard deviation of 2.26, with values ranging between 14.47 and 20.60. Pose Net has the highest mean translation error at 25.12, with a significant standard deviation of 4.29, indicating more variability, and a range from 18.30 to 29.70. The proposed ViTVO model demonstrates the lowest mean translation error at 12.54

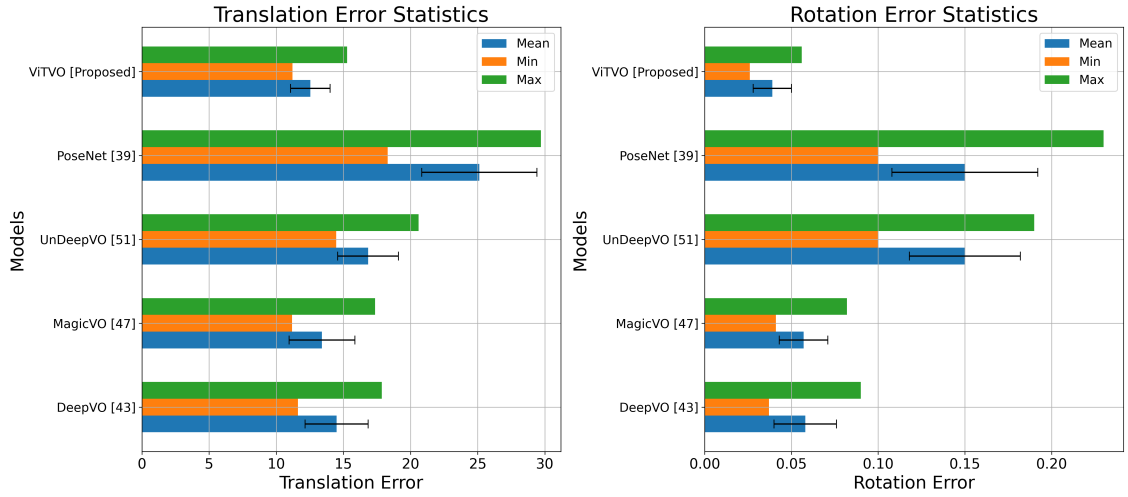

Figure 4.9: Translation and rotation error statistics

with a standard deviation of 1.47, and its errors vary from 11.21 to 15.28, indicating more stable performance across sequences.

For rotation errors, DeepVO has a mean of 0.058 and a standard deviation of 0.018, with values ranging from 0.037 to 0.090. MagicVO's rotation error statistics are similar, with a mean of 0.057 and a standard deviation of 0.014, ranging from 0.041 to 0.082. UnDeepVO presents a higher mean rotation error of 0.150 and a standard deviation of 0.032, with errors between 0.100 and 0.190. Pose Net also has a mean rotation error of 0.150 but with a higher standard deviation of 0.042, indicating a broader error range from 0.100 to 0.230. The proposed ViTVO model achieves the lowest mean rotation error of 0.039 with a standard deviation of 0.011, and its errors range from 0.026 to 0.056, showcasing its effectiveness in maintaining low rotational discrepancies.

4.2 Rotation and translation error comparison by path length and speed

In evaluating the visual odometry system, it is critical to analyze the rotation and translation errors against varying path lengths and speeds to understand the long-term accuracy and robustness of the models. Path length directly influences the drift accumulation over time in VO systems, making it a key factor in determining their reliability. Additionally, the relationship between a vehicle's speed and the VO system's performance provides insights into the practical viability of the system in dynamic environments such as autonomous driving, drone navigation, and mobile robot navigation. This analysis is essential for ensuring that the VO system can maintain accuracy and effectiveness in real-world scenarios where conditions are constantly changing.

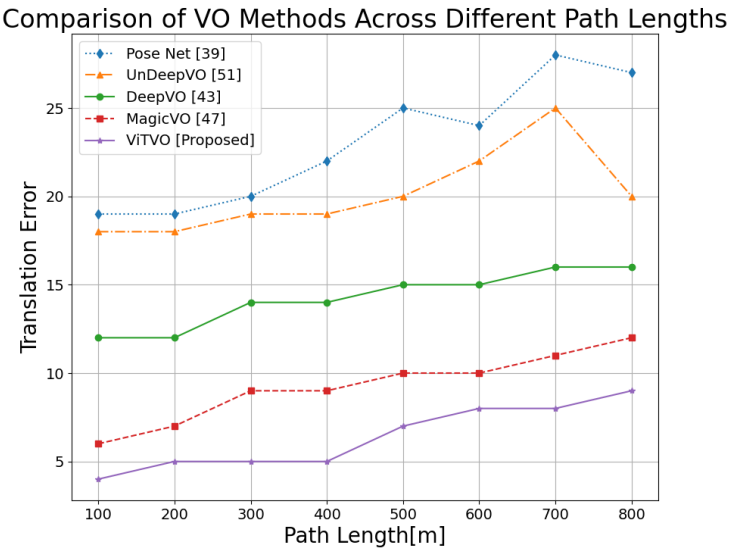

Figure 4.10: Translation error against path length

Posenet relies exclusively on Googlenet as an encoder, lacking mechanisms to capture spatial-temporal information, which is critical for VO performance. This deficiency results in poor performance regarding path length and speed, with trans-

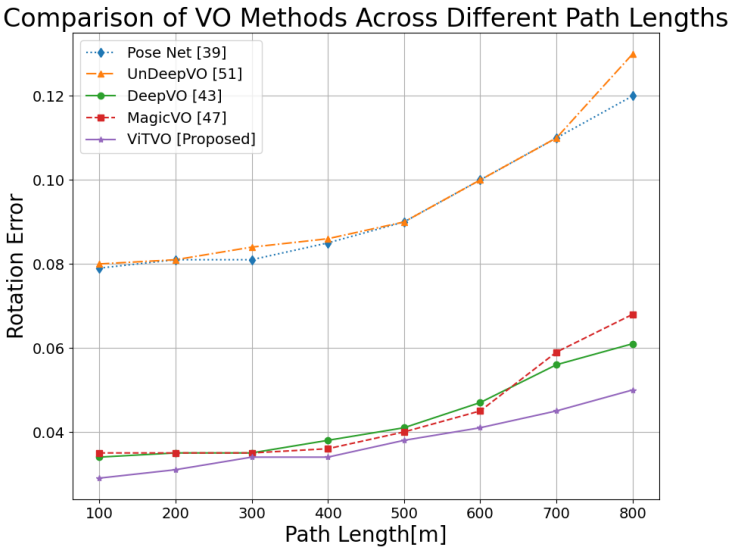

Figure 4.11: Rotation error against path length

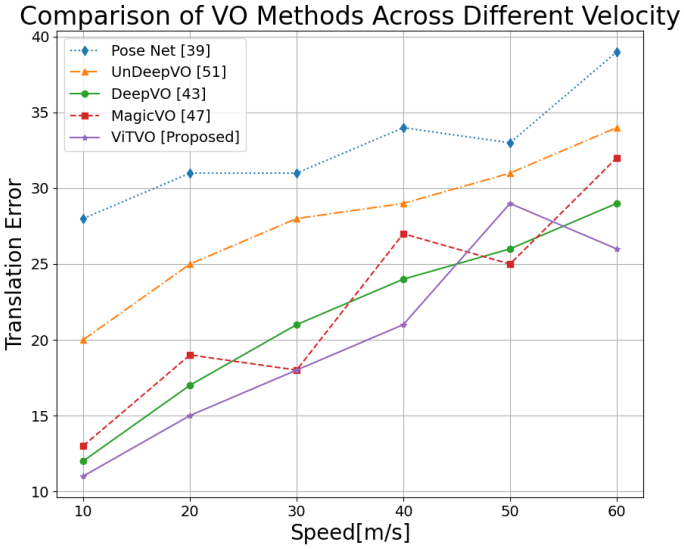

Figure 4.12: Translation error against speed

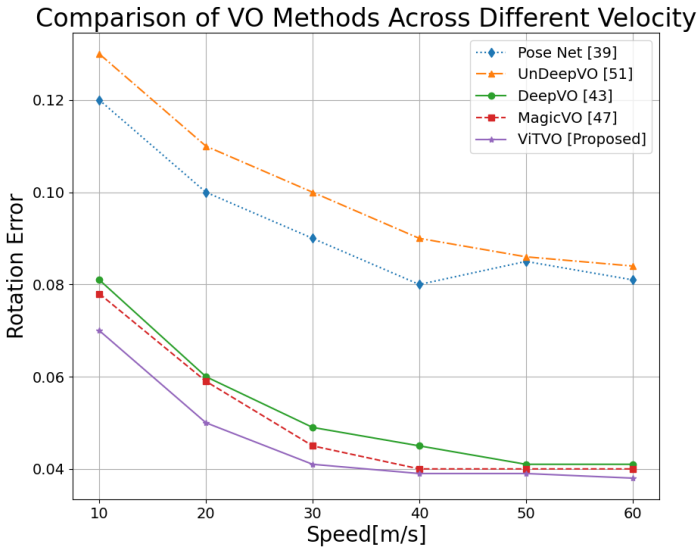

Figure 4.13: Rotation error against speed

lation and rotation errors particularly affected. Analysis shows Posenet struggles with translation errors reaching 30% and rotation errors around 0.09% at moderate speeds (10-30 km/h), making it the worst performer among evaluated methods. From figure 4.12 and figure 4.13, it's clear that as speed increases, translation errors rise while rotation errors decrease, further highlighting the model's inability to capture the sequential nature of spatial data effectively.

At lower path lengths below 300 meters, Posenet's translation and rotation errors are between 20% and 0.08%, respectively (see figure 4.10 and figure 4.11). These errors increase to above 27% and 0.12% as path lengths increase, but the overall performance remains subpar due to the model's inability to handle spatial-temporal data. The absence of mechanisms like LSTM networks exacerbates these issues, whereas learning-based methods incorporating such techniques show better performance by effectively capturing sequential information. Consequently, Posenet's limitations make it inadequate for dynamic environments requiring accurate and robust VO performance.

DeepVO and MagicVO employ an encoder-decoder framework with FlowNet as

the encoder, differing in their choice of decoders: DeepVO utilizes LSTM while MagicVO employs Bi-LSTM. Bi-LSTM in MagicVO proves advantageous by effectively capturing sequential information bidirectionally, overcoming LSTM's vanishing gradient issue with long-term dependencies. Both models achieve high accuracy on shorter paths (100-300 meters) with translation errors below 15% and rotation errors under 0.04 degrees per meter (see figure 4.10 and figure 4.11). They maintain robust performance on longer paths (700-800 meters) with translation errors increasing slightly to 15% and rotation errors under 0.06 degrees per meter, demonstrating their effective pose estimation capabilities (see figure 4.10 and figure 4.11). At lower speeds (10-30 km/h), both DeepVO and MagicVO exhibit translation errors below 20%, though errors rise to 35-40% with increasing speed, reflecting challenges in capturing temporal context effectively (see figure 4.12 and figure 4.13). Rotation errors remain stable across speeds, indicating consistent performance in estimating rotation despite speed variations (see figure 4.12 and figure 4.13).

Learning-based VO methods utilizing Vision Transformer (ViT) architectures exhibit enhanced robustness against varying speeds due to their self-attention mechanisms. ViT's ability to weigh the importance of different image patches and integrate contextual information across the entire scene allows it to capture balanced spatial-temporal information from the environment. This capability is particularly beneficial in mitigating the adverse effects of high-speed motion, such as motion blur and rapid scene changes. Empirical results demonstrate that ViT-based VO models maintain relatively low translation and rotation errors even at higher speeds, indicating their potential for reliable performance in dynamic scenarios.

Moreover, the use of multi-head attention in ViT-based models enables the capture of various aspects of the scene simultaneously, enhancing feature representation and robustness. This architectural feature allows the model to adapt to different speeds by focusing on relevant features and contextual cues critical for accurate

pose estimation. Consequently, ViT-based VO models achieve superior accuracy and precision in pose estimation compared to traditional methods, particularly in high-speed conditions. Extensive experiments across different speed ranges consistently show that ViTVO outperforms traditional VO methods in both translation and rotation errors. At speeds ranging from 10 to 70 km/h, ViTVO maintains translation errors below 26% and rotation errors under 0.03 degrees per meter, whereas other methods exhibit significantly higher errors. These findings underscore the advantages of incorporating Vision Transformers in VO systems, highlighting their potential to enhance pose estimation accuracy and robustness across diverse speed conditions.

5 Discussion

In this chapter, section 5.1 and 5.2 provide a summary of the key findings and contribution of the research. This is followed by a section 5.3 on the limitations of the study. Finally, section 5.4 review potential avenues for future work.

5.1 Summary of findings

In this study, we explored the application of ViT for VO, a critical task in computer vision essential for autonomous navigation and robotics. We investigated the viability of ViT architectures traditionally used for image classification in the context of sequential pose estimation from image sequences.

Our findings indicate that ViTs, with their self-attention mechanism and ability to capture global dependencies in images, can be adapted effectively for VO tasks. By treating sequences of images as a 1D sequence of patches, ViTs demonstrate promising performance in predicting camera poses, comparable to or surpassing traditional CNN approaches. Specifically, ViTs leverage their capacity to model long-range dependencies and semantic context across frames, which is advantageous in scenarios with varying lighting conditions, occlusions, and dynamic environments.

5.2 Contributions of the study

This study contributes to the field of VO in several ways:

(1). Introduction of ViTs to VO: We have shown the feasibility of applying ViTs to sequential pose estimation tasks, expanding the repertoire of models beyond traditional CNN-based approaches.

(2). Performance evaluation: Through rigorous experiments and comparative analysis, we have provided empirical evidence of ViTs' efficacy in capturing spatial and temporal dependencies for accurate pose estimation.

(3). Insights into model adaptation:

We have discussed methodologies to adapt ViTs for VO, including input representation, training strategies, and architectural modifications, providing insights for future research and applications.

5.3 Limitation

One of the main limitations of integrating Vision Transformer algorithms into mobile robotics is the significant computational workload required, which challenges the achievement of real-time performance on edge devices. By leveraging GPUs, NPUs, or specialized hardware accelerators, enable ViTVO to deliver high accuracy and detailed hierarchical information from environment. However, in scenarios with limited computational resources, opting for less accurate ViT variants may compromise the ability to capture detailed spatial-temporal information, thereby ensuring the system remains operational but at the expense of reduced performance. This trade-off between accuracy and power consumption highlights the need for careful selection and configuration of ViT models based on specific hardware constraints and application requirements.

5.4 Future research

The successful application of ViTs in VO opens up several avenues for future exploration:

(1). Generalization across datasets:

Future work can focus on enhancing ViTs' generalization capabilities across diverse datasets and real-world scenarios. Techniques such as self-supervised learning can be employed to leverage large-scale unlabeled data for pretraining ViTs, enabling them to learn robust representations that generalize well beyond specific datasets.

(2). Meta-learning and continual learning for adaptability:

Meta-learning approaches can be investigated to enhance ViTs' adaptability to new environments and tasks with minimal additional training. By learning to learn from limited data, ViTs can quickly adapt to new scenes and conditions encountered during deployment in the field. Additionally, incorporating continual learning strategies can further improve the model's ability to adapt over time, enabling it to continuously learn from new data and experiences without forgetting previously acquired knowledge.

(3). Integration with sensor fusion:

Combining visual information from ViTs with data from other sensors such as LiDAR, IMU can further improve VO accuracy and robustness. Future research should explore multimodal fusion techniques that leverage ViTs' semantic understanding with precise geometric data from complementary sensors.

In conclusion, the integration of Vision Transformers into Visual Odometry represents a significant advancement, promising enhanced accuracy and efficiency in pose

estimation tasks. Moving forward, leveraging self-supervised learning and meta-learning techniques will be pivotal in advancing ViTs' capabilities for real-world applications beyond controlled environments, thereby bridging the gap between research findings and practical deployment in autonomous systems.

References

[1] M. He, C. Zhu, Q. Huang, B. Ren, and J. Liu, "A review of monocular visual odometry", *The Visual Computer*, vol. 36, no. 5, pp. 1053–1065, 2020.

[2] N. Sünderhauf and P. Protzel, "Stereo odometry—a review of approaches", *Chemnitz University of Technology Technical Report*, 2007.

[3] Y. Zhou, G. Gallego, and S. Shen, "Event-based stereo visual odometry", *IEEE Transactions on Robotics*, vol. 37, no. 5, pp. 1433–1450, 2021.

[4] K. L. Lim and T. Bräunl, "A review of visual odometry methods and its applications for autonomous driving", *arXiv preprint arXiv:2009.09193*, 2020.

[5] G. Cioffi, L. Bauersfeld, E. Kaufmann, and D. Scaramuzza, "Learned inertial odometry for autonomous drone racing", *IEEE Robotics and Automation Letters*, vol. 8, no. 5, pp. 2684–2691, 2023.

[6] J. J. Tarrio and S. Pedre, "Realtime edge-based visual odometry for a monocular camera", in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 702–710.

[7] L. Kneip, M. Chli, and R. Siegwart, "Robust real-time visual odometry with a single camera and an imu", in *Proceedings of the British Machine Vision Conference 2011*, British Machine Vision Association, 2011.

[8] K. Konolige, M. Agrawal, and J. Sola, "Large-scale visual odometry for rough terrain", in *Robotics Research: The 13th International Symposium ISRR*, Springer, 2011, pp. 201–212.

[9] S. A. Mohamed, M.-H. Haghbayan, T. Westerlund, J. Heikkonen, H. Tenhunen, and J. Plosila, "A survey on odometry for autonomous navigation systems", *IEEE access*, vol. 7, pp. 97 466–97 486, 2019.

[10] S. Atiya and G. D. Hager, "Real-time vision-based robot localization", *IEEE transactions on robotics and automation*, vol. 9, no. 6, pp. 785–800, 1993.

[11] M. O. Aqel, M. H. Marhaban, M. I. Saripan, and N. B. Ismail, "Review of visual odometry: Types, approaches, challenges, and applications", *SpringerPlus*, vol. 5, pp. 1–26, 2016.

[12] S. Ramalingam, S. Bouaziz, and P. Sturm, "Pose estimation using both points and lines for geo-localization", in *2011 IEEE International Conference on Robotics and Automation*, IEEE, 2011, pp. 4716–4723.

[13] C. Chen, B. Wang, C. X. Lu, N. Trigoni, and A. Markham, "Deep learning for visual localization and mapping: A survey", *IEEE Transactions on Neural Networks and Learning Systems*, 2023.

[14] K. O'shea and R. Nash, "An introduction to convolutional neural networks", *arXiv preprint arXiv:1511.08458*, 2015.

[15] A. Dosovitskiy, L. Beyer, A. Kolesnikov, *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale", *CoRR*, vol. abs/2010.11929, 2020. arXiv: 2010.11929. [Online]. Available: https://arxiv.org/abs/2010.11929.

[16] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite", in *2012 IEEE conference on computer vision and pattern recognition*, IEEE, 2012, pp. 3354–3361.

[17] L. R. Agostinho, N. M. Ricardo, M. I. Pereira, A. Hiolle, and A. M. Pinto, "A practical survey on visual odometry for autonomous driving in challenging scenarios and conditions", *IEEE Access*, vol. 10, pp. 72 182–72 205, 2022.

[18] J. Gui, D. Gu, S. Wang, and H. Hu, "A review of visual inertial odometry from filtering and optimisation perspectives", *Advanced Robotics*, vol. 29, no. 20, pp. 1289–1301, 2015.

[19] S. Sirtkaya, B. Seymen, and A. A. Alatan, "Loosely coupled kalman filtering for fusion of visual odometry and inertial navigation", in *Proceedings of the 16th International Conference on Information Fusion*, IEEE, 2013, pp. 219–226.

[20] M. Bloesch, S. Omari, M. Hutter, and R. Siegwart, "Robust visual inertial odometry using a direct ekf-based approach", in *2015 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, IEEE, 2015, pp. 298–304.

[21] V. Jeyakumar and D. Luc, "Approximate jacobian matrices for nonsmooth continuous maps and c1-optimization", *SIAM Journal on Control and Optimization*, vol. 36, no. 5, pp. 1815–1832, 1998.

[22] J. Civera, O. G. Grasa, A. J. Davison, and J. M. Montiel, "1-point ransac for extended kalman filtering: Application to real-time structure from motion and visual odometry", *Journal of field robotics*, vol. 27, no. 5, pp. 609–631, 2010.

[23] J. Kelly and G. S. Sukhatme, "Visual-inertial sensor fusion: Localization, mapping and sensor-to-sensor self-calibration", *The International Journal of Robotics Research*, vol. 30, no. 1, pp. 56–79, 2011.

[24] R. v. d. Merwe and E. A. Wan, "Sigma-point kalman filters for integrated navigation", in *Proceedings of the 60th annual meeting of the institute of navigation (2004)*, 2004, pp. 641–654.

[25] R. Jurevičius, V. Marcinkevičius, and J. Šeibokas, "Robust gnss-denied localization for uav using particle filter and visual odometry", *Machine Vision and Applications*, vol. 30, no. 7, pp. 1181–1190, 2019.

[26] M. Cook, L. Gugelmann, F. Jug, C. Krautz, and A. Steger, "Interacting maps for fast visual interpretation", in *The 2011 International Joint Conference on Neural Networks*, IEEE, 2011, pp. 770–776.

[27] D. Xueyu, Z. Lilian, L. Ruochen, W. Maosong, W. Wenqi, and M. Jun, "Po-msckf: An efficient visual-inertial odometry by reconstructing the multi-state constrained kalman filter with the pose-only theory", *arXiv preprint arXiv:2407.01888*, 2024.

[28] E. Hong and J. Lim, "Visual inertial odometry using coupled nonlinear optimization", in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2017, pp. 6879–6885.

[29] C. Forster, L. Carlone, F. Dellaert, and D. Scaramuzza, "On-manifold preintegration for real-time visual–inertial odometry", *IEEE Transactions on Robotics*, vol. 33, no. 1, pp. 1–21, 2016.

[30] F. Dellaert, M. Kaess, *et al.*, "Factor graphs for robot perception", *Foundations and Trends® in Robotics*, vol. 6, no. 1-2, pp. 1–139, 2017.

[31] T. Qin, P. Li, and S. Shen, "Vins-mono: A robust and versatile monocular visual-inertial state estimator", *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 1004–1020, 2018.

[32] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "Orb-slam: A versatile and accurate monocular slam system", *IEEE transactions on robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.

[33] A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse, "Monoslam: Real-time single camera slam", *IEEE transactions on pattern analysis and machine intelligence*, vol. 29, no. 6, pp. 1052–1067, 2007.

[34] C. Forster, M. Pizzoli, and D. Scaramuzza, "Svo: Fast semi-direct monocular visual odometry", in *2014 IEEE international conference on robotics and automation (ICRA)*, IEEE, 2014, pp. 15–22.

[35] M. Zhai, X. Xiang, N. Lv, and X. Kong, "Optical flow and scene flow estimation: A survey", *Pattern Recognition*, vol. 114, p. 107 861, 2021.

[36] R. Roberts, H. Nguyen, N. Krishnamurthi, and T. Balch, "Memory-based learning for visual odometry",

[37] T. A. Ciarfuglia, G. Costante, P. Valigi, and E. Ricci, "Evaluation of non-geometric methods for visual odometry", *Robotics and Autonomous Systems*, vol. 62, no. 12, pp. 1717–1730, 2014.

[38] G. Costante, M. Mancini, P. Valigi, and T. A. Ciarfuglia, "Exploring representation learning with cnns for frame-to-frame ego-motion estimation", *IEEE robotics and automation letters*, vol. 1, no. 1, pp. 18–25, 2015.

[39] A. Kendall, M. Grimes, and R. Cipolla, "Posenet: A convolutional network for real-time 6-dof camera relocalization", in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 2938–2946.

[40] A. Pulver and S. Lyu, "Lstm with working memory", in *2017 International Joint Conference on Neural Networks (IJCNN)*, IEEE, 2017, pp. 845–851.

[41] R. M. Schmidt, "Recurrent neural networks (rnns): A gentle introduction and overview. arxiv 2019", *arXiv preprint arXiv:1912.05911*, 1912.

[42] Z. Huang, W. Xu, and K. Yu, "Bidirectional lstm-crf models for sequence tagging", *arXiv preprint arXiv:1508.01991*, 2015.

[43] S. Wang, R. Clark, H. Wen, and 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)*, IEEE, May 2017. DOI: 10.1109/icra.2017.7989236. [Online]. Available: http://dx.doi.org/10.1109/ICRA.2017.7989236.

[44] A. Dosovitskiy, P. Fischer, E. Ilg, *et al.*, "Flownet: Learning optical flow with convolutional networks", in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 2758–2766.

[45] A. Geiger, J. Ziegler, and C. Stiller, "Stereoscan: Dense 3d reconstruction in real-time", in *2011 IEEE intelligent vehicles symposium (IV)*, Ieee, 2011, pp. 963–968.

[46] C. Szegedy, W. Liu, Y. Jia, *et al.*, "Going deeper with convolutions", in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

[47] J. Jiao, J. Jiao, Y. Mo, W. Liu, and Z. Deng, *Magicvo: End-to-end monocular visual odometry through deep bi-directional recurrent convolutional neural network*, 2018. arXiv: 1811.10964 [cs.CV].

[48] H. Damirchi, R. Khorrambakht, and H. D. Taghirad, "Exploring self-attention for visual odometry", *arXiv preprint arXiv:2011.08634*, 2020.

[49] A. Vaswani, N. Shazeer, N. Parmar, *et al.*, "Attention is all you need", *Advances in neural information processing systems*, vol. 30, 2017.

[50] N. Kaygusuz, O. Mendez, and R. Bowden, "Mdn-vo: Estimating visual odometry with confidence", in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2021, pp. 3528–3533.

[51] R. Li, S. Wang, Z. Long, and D. Gu, "Undeepvo: Monocular visual odometry through unsupervised deep learning", *CoRR*, vol. abs/1709.06841, 2017. arXiv: 1709.06841. [Online]. Available: http://arxiv.org/abs/1709.06841.

[52] D. Chen, Y. Yu, and X. Gao, "Semi-supervised deep learning framework for monocular visual odometry", 2019.

[53] I. Goodfellow, J. Pouget-Abadie, M. Mirza, *et al.*, "Generative adversarial networks", *Communications of the ACM*, vol. 63, no. 11, pp. 139–144, 2020.

[54] T. Feng and D. Gu, "Sganvo: Unsupervised deep visual odometry and depth estimation with stacked generative adversarial networks", *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 4431–4437, 2019.

[55] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding", *arXiv preprint arXiv:1810.04805*, 2018.

[56] B. Mann, N. Ryder, M. Subbiah, *et al.*, "Language models are few-shot learners", *arXiv preprint arXiv:2005.14165*, vol. 1, 2020.