FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Fusion of Multimodal Information for Egomotion Estimation of Autonomous Vehicles

João André Silva Roleira Marinho

Mestrado em Engenharia Informática e Computação

Supervisor: Prof. Andry Maykol Gomes Pinto

July 20, 2024

Fusion of Multimodal Information for Egomotion Estimation of Autonomous Vehicles

João André Silva Roleira Marinho

Mestrado em Engenharia Informática e Computação

Approved in oral examination by the committee:

President: Prof. Henrique Lopes Cardoso

Referee: Prof. André Fidalgo

Referee: Prof. Andry Maykol

July 20, 2024

Resumo

A condução autónoma tem o potencial de revolucionar a indústria dos transportes, aumentando a segurança, reforçando a sustentabilidade ambiental e contribuindo para o crescimento económico. Um aspeto central é a estimativa da pose, que se refere à capacidade de um sistema para estimar o seu movimento em relação ao meio envolvente, o que é crucial para tarefas como a localização, a navegação e a antecipação de obstáculos.

Avanços na Odometria Visual e na Odometria de Nuvem de Pontos resultaram em implementações precisas de estimativa de pose. No entanto, sistemas de uma só modalidade enfrentam desafios em determinadas condições, como alterações de iluminação e condições climatéricas adversas. Para resolver estes problemas, a investigação orientou-se para a Odometria Multimodal, que utiliza várias modalidades de sensores para reduzir a incerteza e aumentar a robustez. Embora eficaz, a Odometria Multimodal introduz complexidades na gestão das diversas características dos sensores. As abordagens geométricas tradicionais têm dificuldades em ambientes complexos, criando oportunidades para tecnologias do tipo *Machine Learning* (ML), nomeadamente *Deep Learning* (DL). DL, excelente na extração de características e na adaptação a diversos cenários, tem mostrado resultados notáveis em vários domínios, incluindo a visão computacional. A sua aplicação na Odometria Multimodal continua a ser pouco explorada, apresentando uma solução promissora para a estimativa da pose.

Esta dissertação explorou o potencial do DL no tratamento de dados de vários sensores para prever poses relativas de 3 graus de liberdade (DoF). A solução de ponta a ponta utilizou dois *backbones* baseados em modelos ResNet para capturar representações detalhadas de imagens e fluxo ótico concatenado com informações de profundidade, respetivamente. O módulo de estimativa de pose incluía três perceptrons de multicamadas, cada um estimando um dos 3 valores DoF. As características extraídas foram fundidas antes da estimativa da pose, classificando a arquitetura como uma abordagem de fusão intermédia. O modelo foi treinado e avaliado em cenários do mundo real a partir do conjunto de dados KITTI contra metodologias avançadas, incluindo metodologias geométricas e baseadas em redes neuronais que dependem de dados únicos e multimodais. Os testes incluíram ambientes desafiantes como estradas com vegetação densa que transitam do campo para áreas suburbanas, bem como estradas pavimentadas e mais estreitas com alguns veículos em movimento. As métricas de análise avaliaram as metodologias em termos de previsões individuais e erros globais. Além disso, a robustez do modelo e as capacidades de generalização foram avaliadas utilizando dados artificialmente degradados, envolvendo a introdução de ruído, como a desfocagem, o ruído Gaussiano e ajustes de brilho e contraste nos dados de imagem, a partir dos quais o fluxo ótico foi estimado, bem como a aplicação de ruído Gaussiano à informação de profundidade.

O modelo apresentou resultados positivos, superando o DeepVO em até 65 % no Erro de Translação e 73 % no Erro de Rotação. Além disso, o modelo proposto foi capaz de competir com abordagens geométricas recentes, alcançando um desempenho comparável em termos de Erro de Pose Relativo (RPE) de translação em muitas das sequências de treino e na sequência de teste 10,

com um valor de 0.06 m. Por outro lado, este trabalho demonstrou níveis de desempenho comparáveis tanto com dados originais como degradados, com uma discrepância máxima de 0.01 m e 0.03° no RPE de translação e rotação, respetivamente, entre os dois. Finalmente, a capacidade do modelo para lidar com dados heterogéneos de diferentes modalidades de sensores realça a sua robustez inerente a várias restrições ambientais.

Abstract

Autonomous driving has the potential to revolutionise transportation by improving safety, enhancing environmental sustainability, and contributing to economic growth. Central to this is egomotion estimation, which allows agents to estimate their motion relative to their surroundings, which is crucial for tasks like localisation, navigation, and obstacle avoidance.

Advancements in Visual Odometry and Point Cloud Odometry have yielded accurate pose estimation implementations using visual imagery or point cloud data. However, single-modality systems face challenges under conditions like illumination changes and harsh weather. To address these, research has shifted towards Multimodal Odometry, which uses multiple sensor modalities to reduce uncertainty and enhance robustness. Although effective, Multimodal Odometry introduces complexities in managing diverse sensor characteristics. Traditional geometric approaches struggle in complex environments, creating opportunities for Machine Learning (ML), particularly Deep Learning (DL). DL, excelling at extracting features and adapting to diverse scenarios, has shown remarkable results in various domains, including computer vision. Its application in Multimodal Odometry remains underexplored, presenting a promising solution for egomotion estimation.

This dissertation explored the potential of DL in handling multimodal sensor data for predicting 3 Degrees of Freedom (DoF) relative poses. The end-to-end solution employed two ResNet-based backbones to capture detailed representations from high-resolution images and concatenated optical flow with depth information. The pose estimation module comprised three Multilayer Perceptrons, each regressing one of the 3DoF values. Extracted features were merged prior to pose estimation, classifying the architecture as an intermediate fusion approach. The model was trained and evaluated on real-world driving scenarios from the KITTI dataset against various state-of-the-art methodologies, including geometric and learning-based methodologies that rely on both single and multimodal data. Testing included challenging environments featuring densely vegetated roads transitioning from the countryside to suburban areas, as well as paved, narrower roads with a few moving vehicles. Evaluation metrics assessed the methodologies in terms of both individual predictions and global errors. Additionally, the model's robustness and generalisation capabilities were evaluated using artificially degraded data, involving the introduction of noise, such as blur, Gaussian noise, and brightness and contrast adjustments to image data, from which optical flow was estimated, as well as applying Gaussian noise to depth information.

The model exhibited positive results, surpassing DeepVO by up to 65 % in Translational error and 73 % in Rotational error. Additionally, the proposed model was able to compete with state-of-the-art geometric approaches, achieving a comparable performance in terms of translation Relative Pose Error (RPE) in many of the training sequences and the tested sequence 10, with a value of 0.06 m. Moreover, this work demonstrated comparable performance levels under both original and degraded data, with a maximum discrepancy of 0.01 m and 0.03° in translation and rotation RPE, respectively, between the two. Finally, the model's capacity to handle heterogeneous data from different sensor modalities highlights its inherent robustness to various environmental constraints.

Agradecimentos

Ao meu orientador, Prof. Dr. Andry Maykol Pinto, quero agradecer por me ter aceite neste projeto sem antes ter tido contacto comigo, acreditando nas minhas capacidades, pela frontalidade, sentido de qualidade que requeria de mim e dos restantes alunos, assim como por me dar bons conselhos para o futuro profissional que me espera. À minha co-orientadora Maria Inês, cuja disponibilidade e ajuda foram incansáveis. Que, apesar de ficar ofendida quando falava da sua terra, merece umas devidas gomas. Ao Pedro Nuno, que considero meu "co-co" orientador, pois sempre me dava na cabeça para manter os dados bem normalizados e cujo apoio ao longo da tese foi fundamental. À restante equipa do CRAS, que fora a ajuda prestada, criava um bom ambiente de trabalho, sempre tranquilos e com boa energia.

Agradeço também aos amigos que a faculdade me trouxe, e àqueles que continuaram ao meu lado durante estes anos. Ao 121, aos Chicos, aos catalães, franceses e americano que fizeram esta experiência inesquecível.

Por último, e totalmente mais importante, um gigante agradecimento à minha família. Em especial à minha mãe e ao meu pai por tudo que me permitiram vivenciar, pelo amor e presença quando era preciso e por tentarem ao máximo dar-me a melhor educação que estava ao seu alcance. Ao meu irmão aluado, que adoro muito e desejo a maior das sortes para a sua jornada académica. E por fim, aos meus avós, os quais me orgulho de ter e que agradeço poderem estar cá para ver o evoluir do meu percurso. Obrigado.

João André Silva Roleira Marinho

"It's not how much time you have, it's how you use it."

Ekko

Contents

List of Figures

List of Tables

Acronyms

2D	Two dimensional
3D	Three dimensional
AI	Artificial Intelligence
ATE	Absolute Trajectory Error
BERT	Bidirectional Encoder Representations from Transformers
BRIEF	Binary Robust Independent Elementary Features
BRISK	Binary Robust Invariant Scalable Keypoints
CNN	Convolutional Neural Network
CT-ICP	Continuous-Time Iterative Closest Point
DL	Deep Learning
DoF	Degrees of Freedom
DSO	Direct Sparse Odometry
DVSO	Deep Virtual Stereo Odometry
ELiOT	End-to-End LiDAR Odometry using Transformer
ELO	Efficient LiDAR Odometry
FAST	Features from Accelerated Segment Test
FFN	Feed-Forward Network
F-LOAM	Fast LiDAR Odometry and Mapping
GNN	Graph Neural Network
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
ICP	Iterative Closest Point
IMU	Inertial Measurement Unit
INS	Inertial Navigation System
IQR	Interquartile Range
IRFE	Implicitly Represented Flow Embedding
LiDAR	Light Detection and Ranging
LO	Laser Odometry
LOAM	LiDAR Odometry and Mapping
LSTM	Long Short-Term Memory
MAP	Maximum A Posteriori
MAV	Microaerial Vehicle
MCFA	Masked Cross-Frame Attention
ML	Machine Learning
MLP	Multilayer Perceptron
MSE	Mean Squared Error
NCC	Normalized Cross Correlation
NDT	Normal Distribution Transform

ORB	Oriented FAST and Rotated BRIEF
P3P	Perspective-3-Point
PF2D	PoseFormer2D
PF3D	PoseFormer3D
PnP	Perspective-n-Point
Radar	Radio Detection And Ranging
RANSAC	Random Sample Consensus
RCNN	Recurrent Convolutional Neural Network
ReLU	Rectified Linear Unit
RGB	Red Green Blue
RGB-D	Red Green Blue - Depth
RMSE	Root Mean Squared Error
RNN	Recurrent Neural Network
RPE	Relative Pose Error
SAD	Sum of Absolute Differences
Seq2seq	Sequence to Sequence
SGD	Stochastic Gradient Descent
SIFT	Scale-Invariant Feature Transform
SLAM	Simultaneous Localization and Mapping
SMAF	Soft Mask Attention Fusion
SOFT	Stereo Odometry based on careful Feature Selection and Tracking
STPF	Spatio-Temporal PoseFormer
SuMa	Surfel-based Mapping
SURF	Speeded Up Robust Features
ViT	Vision Transformer
VO	Visual Odometry
WMSA	Window-based Masked Self-Attention

Chapter 1

Introduction

1.1 Context

Autonomous driving holds the potential to revolutionise the transportation landscape and redefine mobility as we know it. While fully autonomous systems remain on the horizon, several companies have conducted controlled and simulated trials employing partially self-driving vehicles, aiming to pave the way for widespread adoption. Nevertheless, features like automated parking, collision avoidance, and adaptive cruise control, already prevalent in many modern vehicles, significantly contribute to enhanced driver comfort and informed decision-making. Undoubtedly, autonomous mobility holds paramount importance, with profound implications for safety, environmental well-being, and economic prosperity. According to the World Health Organization's Global status report on road safety 2023 [1], road traffic accidents constituted the leading cause of death for children and young people aged 5–29 years in 2019, with an estimated 1.19 million fatalities globally in 2021. Factors such as human fatigue and distracted driving have been cited as leading causes of such incidents [2] [3], underscoring the imperative for autonomous systems to mitigate human errors. Moreover, road traffic injuries exert a substantial economic burden on societies, amounting to approximately 10–12 % of global gross domestic product [4]. With the ever-increasing number of vehicles on the road, particular transportation systems risk reaching a breaking point, characterised by heightened congestion and pollution, further contributing to greenhouse gas emissions, which already account for roughly one-quarter of global emissions. However, despite the compelling benefits of autonomous driving, crucial challenges pertaining to regulatory frameworks, economic feasibility, societal acceptance, and, most importantly, technological advancements must be addressed before their deployment on public roads.

Central to autonomous systems is egomotion estimation, which refers to the ability of a system to estimate its own motion or movement relative to its surroundings over time. This capability is crucial for accurate vehicle positioning and decision-making in navigation or obstacle avoidance tasks. Improving autonomous driving systems requires the development of novel or enhanced egomotion estimation techniques, coupled with advanced sensor technologies, to empower agents with a more comprehensive and explicit perception of their surroundings.

1.2 Motivation

Environmental uncertainties and complex scenarios pose significant challenges for odometry, particularly at the perception level, where illumination variations, dynamic objects, and harsh weather conditions severely limit the performance of single-modality techniques, raising safety concerns. For example, while Global Navigation Satellite Systems (GNSS) offer accurate positioning in clear environments, their performance degrades significantly in urban canyons due to signal reflections, scattering, and blockage [5]. Moreover, such dense environments demand highly precise localisation, where even minor errors can be critical. Exteroceptive sensors like cameras, Light Detection and Ranging (LiDAR), and Radio Detection and Ranging (Radar) provide alternative, more accurate solutions in relative localisation [6]. Additionally, they are independent of external infrastructure and suitable for resource-constrained platforms. Despite their advantages in cost and reliability, adverse weather conditions can deteriorate data quality. Visual Odometry (VO) suffers in low-light and textureless environments, while Laser Odometry (LO), although unaffected by lighting, exhibits performance drops of up to 25 % in rain, fog, or snow [7]. To address these limitations and enhance robustness, research has shifted towards Multimodal Odometry, which leverages the complementarity and redundancy of multiple sensor modalities (e.g., images and point clouds) to reduce uncertainty and improve decision-making. Despite its benefits, the complexity of sensor fusion arises from managing diverse sensor characteristics, including data imperfections, inconsistencies, alignment issues, and differing operational frequencies.

In the era of significant advancements in Artificial Intelligence (AI), where remarkable progress has been made in fields like computer vision [8] and robotic decision-making [9], odometry techniques can benefit from newer technologies, further advancing the performance of current implementations. While traditional geometric approaches are well-established, they struggle in complex and diverse environments [10]. This limitation stems from the inherent difficulty of hand-crafting a generalised system, paving the way for Machine Learning (ML), specifically Deep Learning (DL), to address this challenge. DL techniques excel at extracting feature representations from data and adapting to diverse scenarios due to the vast amount of information they can be trained on. Furthermore, with traditional techniques exhibiting signs of stagnation, where increased complexity does not translate to improved performance, exploring alternative approaches becomes crucial. Currently developed learning-based methods are far from reaching the performance of state-of-the-art methodologies, which prompts several key questions to be addressed: *Are existing DL architectures sufficient for odometry? Should new or combined architectures be designed? How can DL effectively integrate multimodal data for enhanced robustness?* Convolutional Neural Networks (CNNs), well-established in computer vision, have demonstrated their capability to extract pertinent features from visual data [11]. However, their application in the multimodal domain of odometry remains substantially unexplored. Motivated by these challenges and opportunities, this dissertation outlines a set of goals, detailed in the following section.

1.3 Objectives

To validate the robustness improvement provided by the complementarity and redundancy of multiple sensor modalities, this dissertation focuses on developing an end-to-end Deep Learning-based architecture that integrates various sensor data for egomotion estimation. Specifically, the methodology will rely on the KITTI vision benchmark suite dataset [12] encompassing camera and LiDAR data in on-road scenarios for training, validation, and testing. Emphasis will be placed on the generalisation capability of the proposed solution to ensure effective performance across diverse scenarios.

The following are expected to be met from this work:

- To organise and review the current state of research in Multimodal Odometry.
- To develop an end-to-end intermediate fusion architecture capable of estimating egomotion by leveraging the complementarity of visual and point cloud data.
- To design the methodology based on DL structures, incorporating feature extraction and a regression module aimed at accurately estimating relative poses.
- To train and evaluate the model by comparing its performance against alternative state-of-the-art odometry solutions.
- To assess the generalisation capabilities of the model in real-world challenging scenarios, such as densely vegetated roads with varied backgrounds and dynamic objects.
- To further evaluate the model's robustness under artificially degraded data by introducing noise to simulate sensor imperfections and environmental interferences.

1.4 Document structure

The remainder of the dissertation is structured as follows. Chapter 2 provides a comprehensive overview of state-of-the-art Visual, Point Cloud, and Multimodal Odometry methodologies. The chapter concludes with a critical review of the current landscape, highlighting advancements and identifying existing challenges in the field. Chapter 3 outlines the methodology employed to develop the proposed Deep Learning-based Multimodal Odometry solution. Chapter 4 presents and interprets the achieved results, comparing them to existing state-of-the-art implementations. Finally, Chapter 5 concludes the dissertation by summarising key insights gained from the research and outlining potential avenues for future investigation in this domain.

Chapter 2

State of the Art

The term "odometry" originates from the Greek words *hodos* (meaning "journey") and *metron* (meaning "measure") [6]. Essentially, odometry refers to the estimation of an agent's positional change, encompassing both translation and orientation, tracked over a span of time. This concept closely aligns with egomotion, which can be estimated through various odometry techniques. These techniques exhibit distinct characteristics based on the sensor data used. For instance, Visual Odometry (VO) and Point Cloud Odometry, also known as Laser Odometry (LO), further elucidated in sections 2.1 and 2.2, leverage visual imagery or point cloud data extracted from sensors such as cameras and Light Detection and Ranging (LiDAR), respectively. This differentiation is illustrated in fig. 2.1.

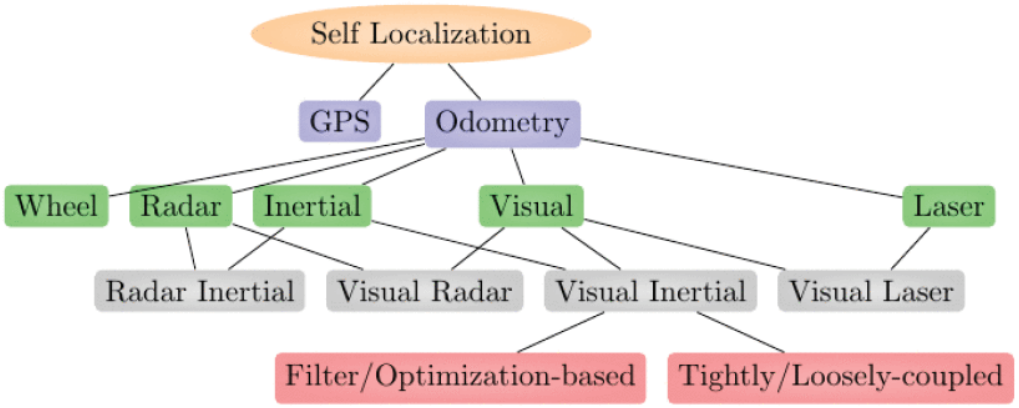


Figure 2.1: Types of Self Localization, focusing on the different types of Odometry [13].

Moreover, the fusion of multiple data sources results in what are termed multimodal techniques. These techniques, discussed in section 2.3, merge diverse data modalities, offering advantages over single-modality methodologies [14]. Concluding this exploration, section 2.4 provides a comprehensive review of insights derived from the literature study, allowing for reflection on the forthcoming steps.

2.1 Visual Odometry

Visual Odometry constitutes a self-contained technique within odometry, reliant on the analysis of variations induced by the motion of the camera(s) on visual data. Originally introduced by Moravec [15] and later named by Nister [16] owing to its resemblance to Wheel Odometry, VO offered distinct advantages over the latter. In its early stages, research in VO was primarily focused on planetary rovers, driven by NASA's Mars exploration program [17], whose emphasis originated from VO's capacity to remain unaffected by wheel slip in uneven terrains or other adverse conditions [18].

VO systems can be characterised by the type of visual sensor employed, its configuration, and the methodology used to predict instantaneous camera poses. Sensor choices encompass a range of options, including monocular, stereo, RGB-D, omnidirectional, thermal, and event-based cameras. Additionally, VO techniques bifurcate into conventional and non-conventional approaches [19], as depicted in fig. 2.2. Traditional methods, also known as geometrical approaches, further subdivide into:

- Feature-based: This technique matches feature points across image sequences, facilitating subsequent correlation.
- Appearance-based: Unlike feature-based methods, appearance-based approaches utilise the entire geometric information within captured images by assessing pixel intensities.
- Hybrid: Combining feature-based and appearance-based strategies, the hybrid approach aims for enhanced robustness and efficiency in estimation.

Conversely, non-conventional techniques leverage Machine Learning (ML) methodologies, offering potential benefits by eliminating the need for camera parameter initialisation and achieving higher levels of scene comprehension without explicit handcrafted formulations.

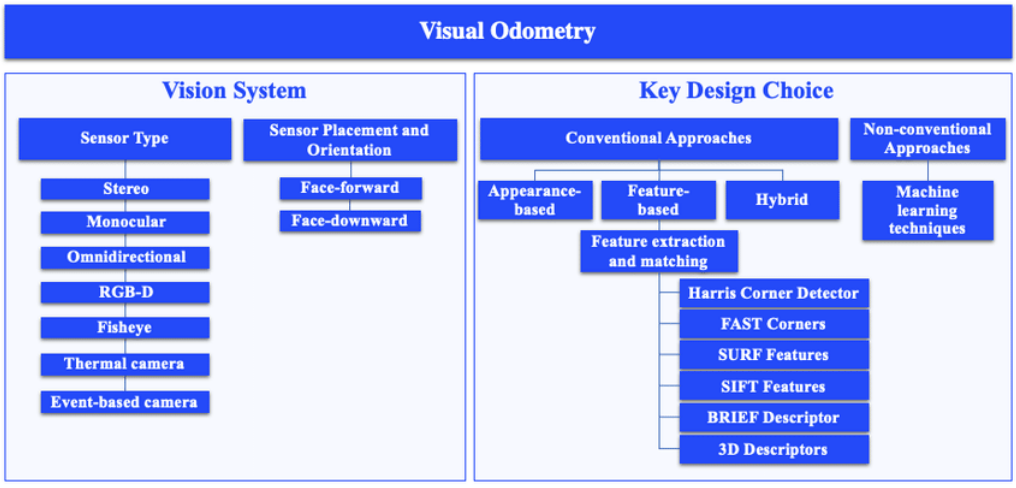


Figure 2.2: Classification of VO Techniques [19].

2.1.1 Geometric approaches

Early VO primarily focused on using feature-based methods [15]. These typically begin by detecting and matching/tracking distinctive features, such as corners, lines, curves, edges, or specific key points across sequential images, as illustrated in fig. 2.5. Subsequently, the fundamental procedural element, motion estimation, is performed according to the nature of these features. Initially, the computation involved a 3D to 3D point registration problem until Nister's paradigm-shifting approach [16] reframed the latter as a 3D to 2D pose estimation problem. Later on, Comport [20] proposed a 2D to 2D technique, which relied on the quadrifocal tensor, leading to more precise motion computations and eliminating the need for triangulations between stereo pairs.

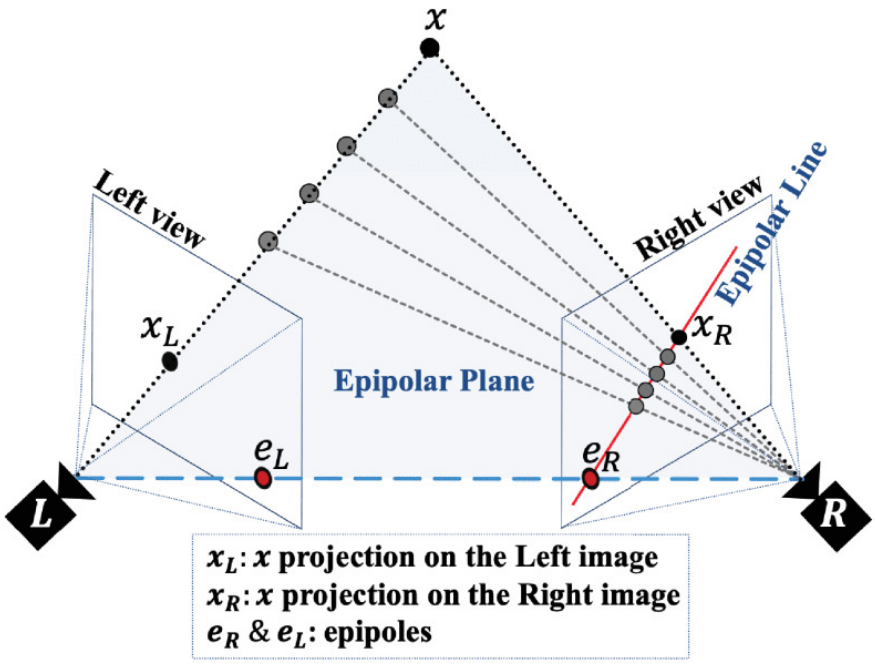


Figure 2.3: Illustration of epipolar geometry and epipolar constraint [19].

The general solution in 3D to 3D feature correspondence involves determining the transformation matrix that minimises the L_2 distance between two sets of 3D features [18]. This process involves capturing two stereo image pairs and triangulating matched features between them. The acquisition of these images requires a stereo camera with two or more lenses, emulating human binocular vision. For 2D to 2D estimation, the translational and rotational motion matrices of the agent are derived from the essential matrix. This technique employs the epipolar constraint to identify the line representing corresponding feature points in another frame, as illustrated in fig. 2.3. Among various 2D to 2D motion estimation techniques, Nister's five-point algorithm [21] stands out. Finally, in 3D to 2D motion estimation, the objective is to ascertain the transformation matrix similar to 3D to 3D, but with respect to the 2D reprojection error on the 3D correspondence feature points. This approach, also known as the Perspective-n-Point (PnP) problem [19], determines the camera pose using a set of n 3D points. The minimal and standard method for robust estimation employs three points and is referred to as perspective-3-point (P3P) [22] [23].

The former two methodologies exhibit superior accuracy compared to the 3D to 3D approach, as they minimise the reprojection error of an image instead of the 3D to 3D feature position error [21] [6] [18]. Additionally, it is noted by [19] that, in practice, 3D to 2D estimations are faster than 2D to 2D. Fig. 2.4 illustrates the concept of estimating motion through matching and tracking featured points in typical feature-based VO schemes. This methodology hinges on the quantity and quality of extracted features, demonstrating robustness in environments characterised by high geometric distortions, allowing for the identification of strong interest points and resilience to illumination variations [24]. Conversely, it may overlook valuable data, limiting its efficacy in scenarios with low-feature conditions, such as dark or sandy environments [19]. These challenging conditions significantly impact feature detection and matching, thereby influencing the algorithm's performance. Consequently, incorporating mechanisms for detecting and eliminating false matches becomes imperative and is regarded as the most intricate aspect in VO [25].

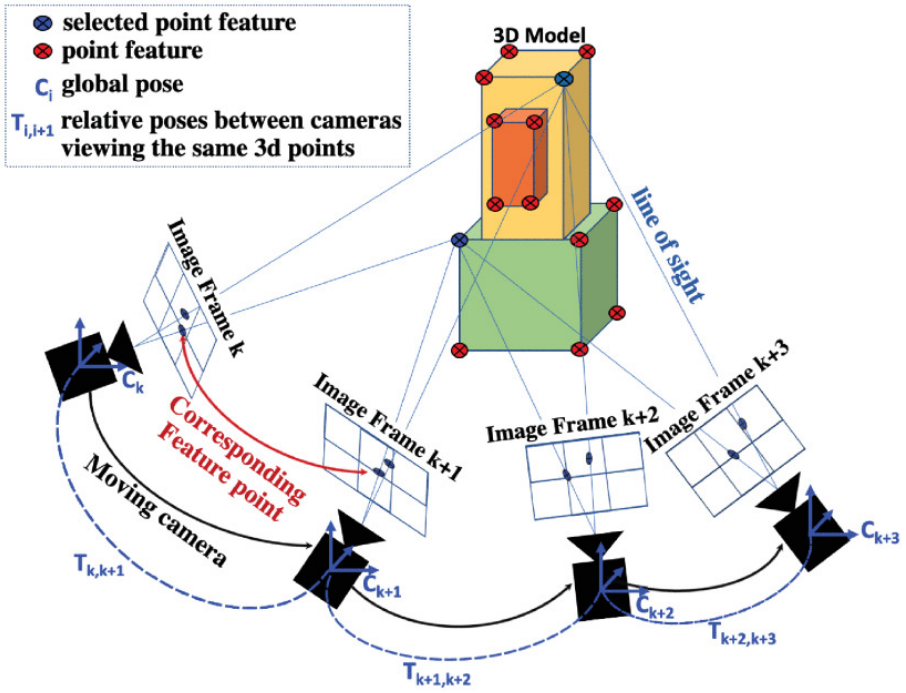


Figure 2.4: Feature-based VO scheme [19]. Tracked points are marked in blue and allow for estimating the relative motion according to position variations in sequent images.

RANdom SAmple Consensus (RANSAC) [23] is a probabilistic method that generates hypotheses from randomly selected subsets of data points. These hypotheses are then validated against the rest of the data to assess their accuracy. This outlier detection algorithm is considered to be the standard and prompted to various other implementations, with preemptive RANSAC [26] emerging as the most prevalent choice [25], primarily owing to its fixed number of iterations, offering advantages in the presence of real-time operations. This diverse array of key design options highlights the absence of an ideal VO architecture for every potential scenario [18]. To derive an optimal solution, one must consider environmental characteristics and available computational

resources. This encompasses the selection of a suitable feature detector and descriptor, as the process of matching features between frames is computationally intensive and scales with the number of features extracted, concurrently enhancing the accuracy of pose estimations.

The evolution of feature detection and matching approaches over the years is notable. Early methodologies involved extracting features in one image and tracking them in subsequent sequences [15] [17]. However, given the potential separation between image captures, contemporary methodologies involve independent feature detection followed by a matching phase based on feature descriptions [16]. This shift enables the exploration of larger-scale environments, mitigating issues related to motion drift. The literature on VO encompasses numerous feature detectors, with some focusing on corner detection (e.g., Harris [27], Shi-Tomasi [28], and FAST [29]), while others prioritise blob [1] detection (e.g., SIFT [30], SURF [31]). The latter can also perform feature description and exhibit invariance to rotation and scale. Enhanced versions of these detectors and descriptors, such as BRIEF [32] and ORB [33] (a combination of the FAST keypoint detector [29] and the latter descriptor), have been developed. Similarly, BRISK [34], a method for keypoint detection, description, and matching, employs a FAST-based detector in combination with a binary descriptor using a configurable sampling pattern. In practical terms, corner detectors are computationally faster but less distinctive, whereas blob detectors are slower yet more distinctive [25]. The selection of an algorithm should consider various performance metrics, including repeatability, computational efficiency, distinctiveness, robustness, and invariance [25]. Noteworthy analyses, such as those conducted by [35] and [36], offer valuable insights into the performance of state-of-the-art methods. The former provides a comprehensive overview of algorithms and recent advancements in the field. At the same time, the latter assesses the outcomes of SIFT, SURF, ORB, and the recent methodology A-KAZE [37], ultimately affirming the superior accuracy of SURF and SIFT, the trade-off between accuracy and lower computational costs presented by ORB, and the middle ground offered by A-KAZE.

An additional category within geometric approaches is appearance-based VO. In contrast to feature-based methods, appearance-based approaches leverage all available geometric information in captured image sequences, monitoring changes in pixel intensity and minimising the photometric error. This approach enhances robustness in scenes characterised by low texture and visibility [38], mitigating aliasing issues associated with similar pattern scenes. Appearance-based methodologies can be categorised into region-based or optical flow-based, as illustrated in fig. 2.5. Region-based matching can be further divided into correlation-based (template matching) and image alignment-based approaches. These methods initially faced limitations, which were addressed by utilising invariant similarities of local areas and global constraints. Vatani *et al.* [39] proposed a practical approach for egomotion estimation using a constrained correlation-based method, later extended by Yu *et al.* [40] through the use of a rotated template, aiding in the estimation of both translation and rotation. Recent studies such as [41], [42], and [43] have either refined previous approaches or explored alternative techniques based on region-based methodologies. However,

[1] An image pattern that differs in properties, such as brightness or colour, compared to its immediate neighbourhood.

region-based VO may encounter challenges such as local minima solutions or divergence issues and susceptibility to dynamic objects in the scene. To address these challenges, optical flow-based techniques offer a viable alternative. These methods estimate motion by analysing brightness displacement patterns between consecutive frames. Optical flow techniques can be categorised as either dense or sparse, depending on whether they analyse the motion of all pixels in the image or only a selected subset of pixels, respectively. Dense algorithms may be less robust to noise [6] when compared to sparse algorithms that exploit local smoothness assumptions. Several techniques have been proposed within the optical flow-based paradigm, such as [44], [45], and [46]. Despite their efficacy, optical flow-based approaches encounter challenges associated with environmental textureless surfaces and computational complexities [47].

Considering the strengths and limitations of both feature-based and appearance-based methodologies, hybrid approaches have been proposed [6]. These approaches combine tracking meaningful features with assessing pixel intensity information between frames, aiming for a more robust solution. Scaramuzza *et al.* [48] introduce a hybrid framework wherein translation and rotation are estimated using feature-based and appearance-based methods, respectively.

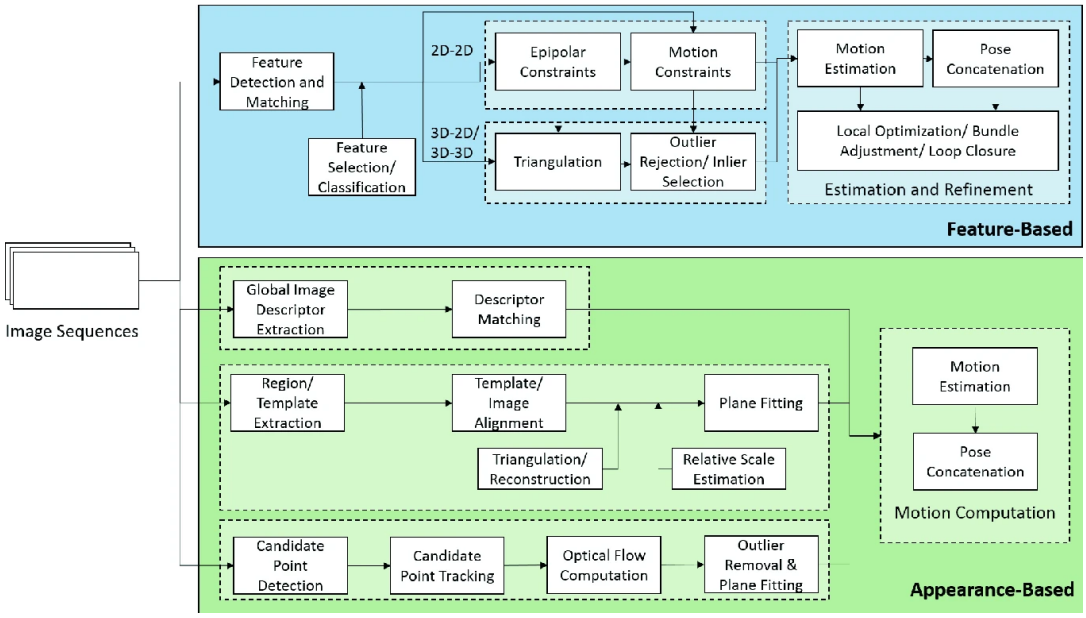


Figure 2.5: Illustration of the visual odometry pipeline depicting common steps in feature- and appearance-based approaches [47].

In VO, the agent's pose is derived by concatenating estimated transformations, each inherently carrying uncertainty, leading to an accumulation of deviation from ground truth over time. To mitigate this issue, geometric approaches often incorporate an optimisation step at the end of the processing pipeline. One widely adopted methodology for this purpose is pose-graph optimization [49]. The essence of pose-graph optimisation lies in refining pose estimates by optimising a specific cost function, leveraging information from preceding transformations. While conventional practices often involve using only the two latest frames to estimate the camera's pose, there exists a fundamental assumption that consistency should be maintained when extending this estimation

to encompass the last n time steps or, indeed, for any time step in the sequence. Another popular approach is bundle adjustment [50]. Similar to pose-graph optimisation, it optimises camera poses and additionally the 3D landmark parameters. This makes it suitable for scenarios where features are tracked across over two frames. The optimisation process involves minimising image reprojection error on the last n frames, with the choice of n influenced by computational constraints for real-time applications.

After analysing the steps in the general geometry-based VO pipeline, as shown in fig. 2.5, it is crucial to highlight studies that achieved state-of-the-art results and provided valuable insights for future research in the field. In this context, we must first introduce key benchmarking datasets, such as KITTI [12], among others [51] [52]. The KITTI vision benchmark suite is widely employed in mobile robotics and autonomous driving. This dataset comprises 22 sequences encompassing visual and LiDAR data captured by a vehicle traversing diverse road traffic scenarios. Notably, 11 of these sequences incorporate ground truth trajectories for training, while the remaining 11 are for evaluation purposes. Moreover, the KITTI benchmark facilitates algorithm comparison through its provided evaluation tool [2], ranking them based on two key metrics: t_{rel}, the Translational error as a percentage, and r_{rel}, the Rotational error in degrees per meter. Evaluation is performed over all possible subsequences ranging from 100 to 800 m.

Firstly, introducing ORB-SLAM2 [53], a versatile Simultaneous Localization and Mapping (SLAM) system built on the monocular feature-based ORB-SLAM [54], which holds significance as a benchmark in the Visual Odometry domain due to its high-performance results and open-source nature. SLAM algorithms aim to concurrently construct a map of the environment and estimate the position of the vehicle or camera within it. The latter module can be conceptualised as a Visual Odometry problem, and as such, both terminologies are closely intertwined. ORB-SLAM2 expands on its predecessor by supporting stereo and RGB-D cameras. It operates through three parallel threads: tracking, local mapping, and loop closing. The VO module, encompassing the first two threads, employs ORB features [33], chosen for their robustness and fast extraction, facilitating real-time application and solving accumulated drift by applying multi-step bundle adjustment.

Building upon the developments in feature-based VO, the contributions of Cvišić *et al.* [55] [56] [57] have bridged the gap between visual methods and LiDAR-based techniques, culminating in state-of-the-art results on datasets such as KITTI. In 2015, the authors presented a novel algorithm for Stereo Odometry based on careful Feature Selection and Tracking (SOFT) [55]. The algorithm initiates feature processing by matching, selecting, and tracking features across image frames. The matching process commences with the extraction of corner-like features, followed by identifying correspondences between left and right images through non-maximum suppression and utilising the Sum of Absolute Differences (SAD). This initial matching is followed by Circular Matching, strategically employed to alleviate susceptibility to outliers. Further refinement ensues with the application of Normalized Cross Correlation (NCC) and RANSAC. The subse-

[2] Available at: https://www.cvlibs.net/datasets/kitti/eval_odometry.php

quent step, feature selection, considers the computational complexity of handling extensive sets of features during motion estimation and experiment conclusions, underscoring the suboptimal outcomes incurred when introducing additional points after prudent selection. Thus, the process uniformly selects features throughout the image. Finally, the SOFT algorithm accomplishes the tracking of features, facilitated by a feature descriptor that encapsulates information on the feature's age, refined current position, and initial descriptor, among others. This approach enables reduced drift and serves as a metric of dissimilarity between the current and initial appearance of the feature, enhancing the overall robustness of the algorithm. Two years after the initial presentation of SOFT, the authors expanded and refined their work, leading to the development of SOFT-SLAM [56]. Although tested in different scenarios, the algorithm was designed to compete in the European Robotics Challenges, focusing on utilising small-scale unmanned aerial vehicles. SOFT-SLAM comprises two distinct threads: one dedicated to the odometry task, leveraging the capabilities of SOFT, and the other dedicated to mapping, supporting loop-closing and ensuring global consistency. These additional functionalities demonstrated improvements over the SOFT algorithm, particularly in scenes with loops, and exhibited comparable or superior performance to other state-of-the-art approaches, such as ORB-SLAM2 and LSD-SLAM [58], in datasets like KITTI. However, the research endeavours did not conclude there. Cvišić *et al.* conducted a series of investigations and proposed enhancements [59] [60] to refine camera setup calibration procedures, placing a significant emphasis on enhancing Visual Odometry accuracy. This meticulous process diminished calibration reprojection errors and elevated the accuracy levels of all examined algorithms. Notably, this includes ORB-SLAM2, VISO2 [61], and a more recent iteration of SOFT, i.e. SOFT2 [57], which emerged as the top-ranking algorithm on the KITTI scoreboard [3], boasting a remarkable t_{rel} of 0.53 % and r_{rel} of 0.0009 °/m. This achievement surpassed even the performance of 3D laser-based approaches, solidifying its superiority in Visual Odometry.

Within the domain of appearance-based VO, Engel et al. introduced Direct Sparse Odometry (DSO) [43] in 2016. Diverging from other direct methods, DSO employs photometric error optimisation for all involved parameters, including camera intrinsics, camera extrinsics, and inverse depth values, which can be considered to perform the photometric equivalent of windowed sparse bundle adjustment. This approach integrates the advantages of direct methods, allowing seamless use and reconstruction of all points (as opposed to just corners), with the efficiency and joint optimisation flexibility of sparse techniques. The experimental findings suggest that utilising precisely calibrated data from cameras explicitly designed for modern applications, such as autonomous cars, drones, VR, and others, may enable direct formulations to surpass the prevailing geometrical/indirect approaches. Additionally, this work lays the groundwork for a subsequent DL approach outlined in the following subsection.

[3] As of June 2024

2.1.2 Learning-based approaches

In recent years, the field of Visual Odometry has experienced a notable paradigm shift, with an increasing focus on learning-based approaches [19]. This shift is driven by the limitations observed in traditional geometric methods, which, while well-established, struggle in specific environments, such as complex or dynamic scenes. In contrast, learning-based approaches exhibit potential increased robustness and adaptability due to their capacity to capture feature representations without explicit handcrafted formulations if trained on large-scale representative datasets [10]. Their advantages include eliminating the need for prior knowledge of camera parameters, correction of scale in translation estimations, and robustness to similar noises and outliers by which they are trained [47]. As depicted in fig. 2.6, non-conventional methodologies diverge from the manual steps inherent in geometric approaches. Instead, they employ a neural network architecture typically composed of multiple sub-networks for tasks such as depth estimation, feature extraction, and egomotion estimation [10]. The final step involves combining all modules and evaluating outputs, either through a supervision signal or against a cost function, aiming at fine-tuning network parameters. Among various learning-based categories, approaches grounded on labelled data are referred to as supervised learning. In the problem of egomotion estimation, the labels represent the camera's relative pose or position, and the model is trained to ensure its predictions closely align with the actual values. However, due to the high cost of labelling data, these methods come with limitations. In unsupervised learning, the prerequisite for labelled training data is eliminated. Instead, these methodologies guide their predictions using geometric constraints such as photometric consistency. This foundational concept, previously introduced in direct traditional-based VO, is computed as the variance in pixel intensities between a warped image and the target image, serving as the loss function to be minimised.

Numerous architectural choices exist when implementing a learning-based approach, specifically within the domain of DL methodologies. One approach involves combining geometric methods with the benefits of learning-based strategies. Deep Virtual Stereo Odometry (DVSO) [62] seeks to capitalise on this integration by introducing deep depth predictions into the existing DSO algorithm. The newly proposed architecture, StackNet, is trained using a semi-supervised methodology, combining both supervised and self-supervised methods. This choice is driven by the high cost of obtaining ground truth values from specific sensors and the current limitations in accuracy observed in self-supervised methods compared to supervised methods. The network's output, i.e. predicted depth maps, are subsequently integrated with photoconsistency constraints into the original DSO optimisation problem. This integration results in a highly accurate system that is not susceptible to scale drift, surpassing learning-based alternatives and achieving results comparable to traditional state-of-the-art stereo VO methods. Evaluation data from the KITTI dataset reveals a t_{rel} of 0.90 % and r_{rel} of 0.0021 °/m. Yang *et al.*, the same authors behind DVSO, delved further into their research by introducing Deep Depth, Deep Pose, and Deep Uncertainty, collectively referred to as D3VO [63]. The primary objective of this implementation is to enhance the performance of geometric approaches by leveraging three levels of deep neural network predictions: monocular

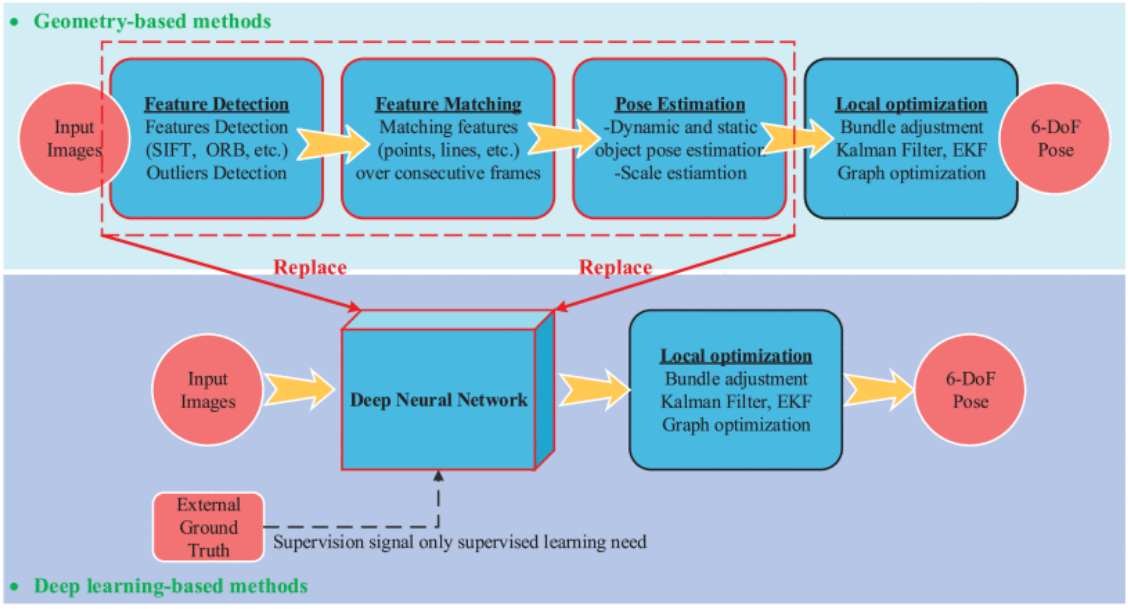


Figure 2.6: Comparing geometry-based and Deep Learning-based Visual Odometry Approaches: Extracting feature points versus direct pose estimation [10]. The shift from manual feature handling to neural network architectures highlights the streamlined process in Deep Learning-based VO methods.

depth, photometric uncertainty, and relative camera pose. In contrast to DVSO, this methodology operates without external supervision, opting instead for training two Convolutional Neural Networks (CNNs) in a self-supervised manner. The first network, DepthNet, undertakes the task of predicting both depth and uncertainty. The latter prediction is crucial for down-weighting errors from pixels that violate the brightness assumption. Simultaneously, pose estimates are generated using PoseNet. To address inconsistent illumination between training image pairs, the network also predicts brightness transformation parameters that dynamically align the brightness of source and target images during training. These predictions are seamlessly integrated into the tracking front-end and the photometric bundle adjustment backend [43] modules, as depicted in fig. 2.7. The results from the KITTI dataset demonstrate an improvement over DVSO, with a t_{rel} of 0.88 %.

Diverging from previously presented methodologies, DeepVO [64] introduces an end-to-end supervised learning approach. In essence, it directly infers poses from a sequence of raw image data without incorporating any module from the VO pipeline, as depicted in fig. 2.8. The network employs a Recurrent Convolutional Neural Network (RCNN) architecture, merging the strengths of the Convolutional Neural Network (CNN) and the Recurrent Neural Network (RNN). CNN, a dominant architecture in the computer vision domain, transforms raw data into concise and meaningful representations, excelling in learning geometric features from images. Meanwhile, RNN, by maintaining an internal state, effectively models the sequential dynamics inherent to VO. This enables the derivation of connections among consecutive image frames, enhancing pose predictions. However, since trained in a supervised manner, the methodology heavily relies on substantial, diverse, labelled data. Consequently, due to the unsatisfactory results compared to

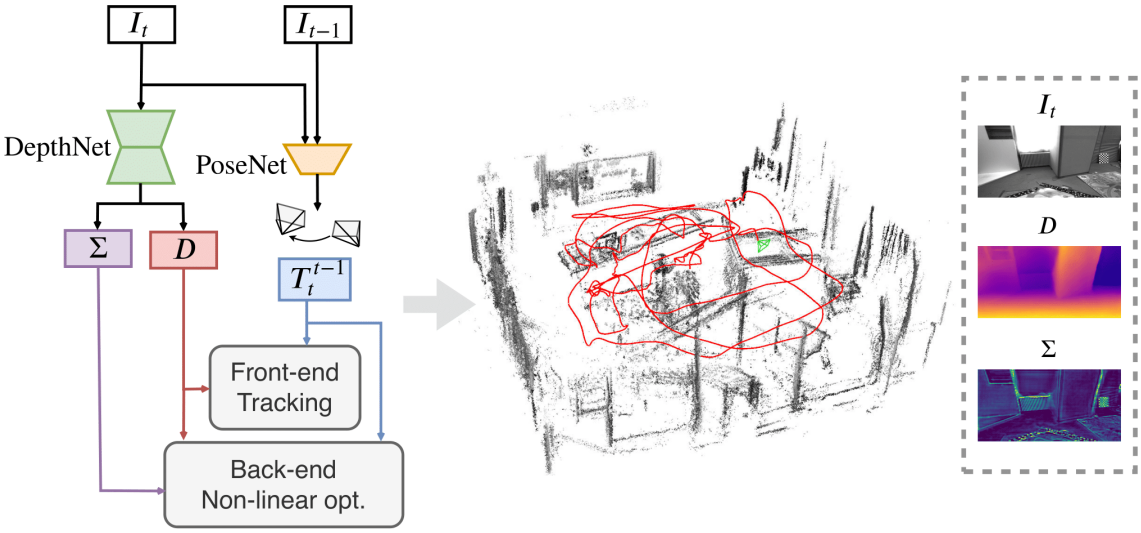


Figure 2.7: D3VO architecture tightly integrates Deep Depth, Deep Pose, and Deep Uncertainty estimations into the front-end tracking and back-end non-linear optimisation of a sparse direct odometry framework [63].

state-of-the-art solutions, its contribution is primarily confined to serving as a proof of concept for end-to-end DL methodologies.

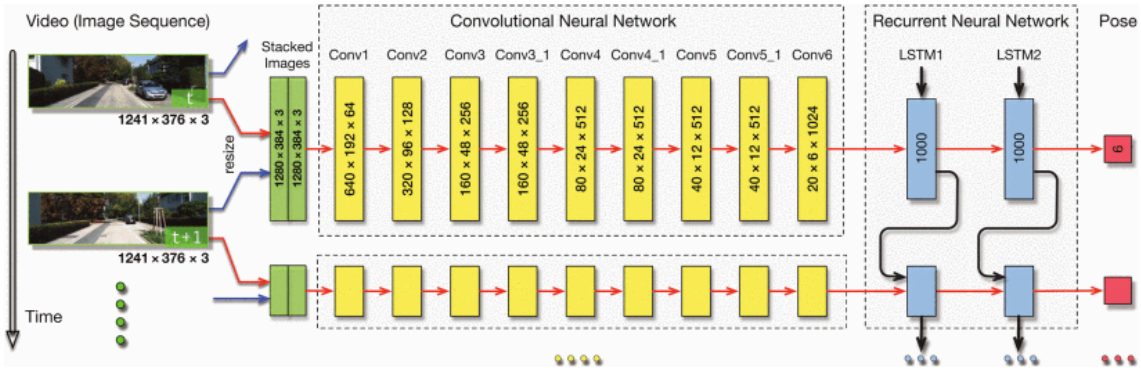


Figure 2.8: Illustration of the RCNN-based monocular VO system architecture, DeepVO [64]. Combining CNNs with RNNs allows for robust feature extraction and capturing sequential relations between image frames.

Regarding the explored techniques, the focus has been limited to Convolutional Neural Networks and Recurrent Neural Networks. Despite their success in various domains, they encounter challenges within the Visual Odometry field. CNNs, for instance, prove less effective in learning relationships and dynamics from sequential data, while RNNs struggle to extract efficient feature representations from high-dimensional data like the former. [10] emphasises the significance of this challenge by highlighting the necessity to delve into more advanced DL techniques. Examples include Graph Neural Networks (GNNs), capable of modelling large-scale optimisation problems, or the adoption of attention-based networks, which align the process more closely with human problem-solving approaches.

Transformer models, introduced by Vaswani *et al.* [65], are built around attention mechanisms and have achieved remarkable success across various machine-learning tasks. Widely acknowledged as the go-to architecture in natural language processing [66] [67], they have also found application in computer vision [8] and speech recognition tasks [68]. However, to comprehend how these models operate, it is crucial to trace their evolution from traditional encoder-decoder architectures, specifically Sequence to Sequence (Seq2seq) learning [69]. Like Transformers, Seq2seq models are based on encoder-decoder architectures, which can solve general sequence-to-sequence problems without requiring a known and fixed dimensionality of the inputs and outputs. These models possess an encoder and a decoder module, usually following a Long Short-Term Memory (LSTM) architecture, a type of RNN chosen for its ability to learn from data with long-range temporal dependencies. Despite these advantages, Seq2seq models have limitations. Processing one input token at a time extends training durations and incurs computational costs. Additionally, the output of the encoder, usually referred to as the context vector, is a compression of all tokens in the input sequence, potentially leading to information loss from the initially inputted data, especially in the case of lengthy data sequences. To overcome these challenges, Transformer models capitalise on their parallelizability and self-attention mechanisms, allowing them to simultaneously consider the entire input sequence and thus provide a more comprehensive understanding of the global context. Examining the vanilla Transformer architecture, depicted in fig. 2.9, both the encoder and decoder modules are distinctly represented, this time, based on a specific attention mechanism known as self-attention. Broadly speaking, self-attention operates by assessing the similarity of each token to all other tokens in the sequence, including itself, enabling the model to discern the significance of each input segment. This process begins by converting a given input into three essential vectors: Query, Key, and Value. Each value is assigned a weight calculated through a compatibility function of the corresponding query with all other keys, typically utilising the softmax function. Subsequently, this weight is multiplied by the respective value to form the self-attention value of the token. Transformer models extend this mechanism on a larger scale by linearly projecting the queries, keys, and values multiple times, achieving parallel scaled dot-product attention functions, referred to as multi-head attention. This robust approach enables the model to collectively attend to information from different representation subspaces at various positions in the sequence. Another pivotal sub-layer present in both modules is a simple, position-wise Feed-Forward Network (FFN). Comprising mainly Fully Connected layers, this network is applied to each position separately and identically, enhancing the model's adaptability to more intricate data by introducing additional weights and non-linearities. Despite the efficacy of the self-attention mechanism, other crucial considerations arise, such as the ordering of input tokens. Vaswani *et al.* address this by employing positional encoding to inject information regarding the sequence's order right from the initial processing stages. These encodings can be learned or fixed, with the primary distinction lying in their ability to update with each pass through the model. In the traditional approach, positional encodings are fixed and computed using sinusoidal functions to generate a unique positional signal, which is added to each token embedding. This combined representation is passed through the network via residual connections between sub-layers, allowing

the self-attention mechanism to focus solely on establishing relationships among the input tokens without retaining additional positional information. While fixed encoding has demonstrated effectiveness, it may limit the model's adaptability to the nuances of the input data. For example, in scenarios where sequences surpass the length of the model's trained data, a learnable implementation, as utilised in Bidirectional Encoder Representations from Transformers (BERT) [66], would more accurately address positional information, as opposed to fixed encodings where positions would be extrapolated.

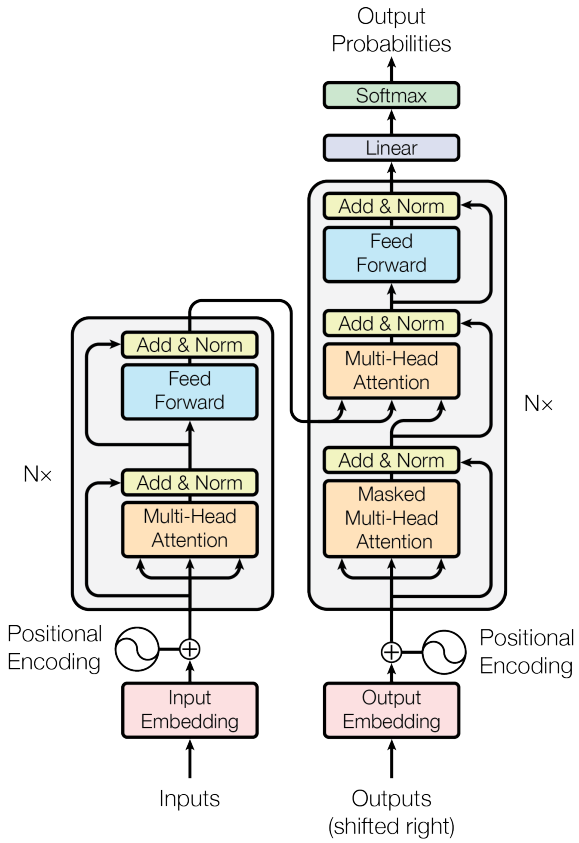


Figure 2.9: Transformer model architecture [65]. The diagram elucidates the structural elements, with the encoder depicted on the left and the decoder on the right, as well as the data flow within the Transformer model.

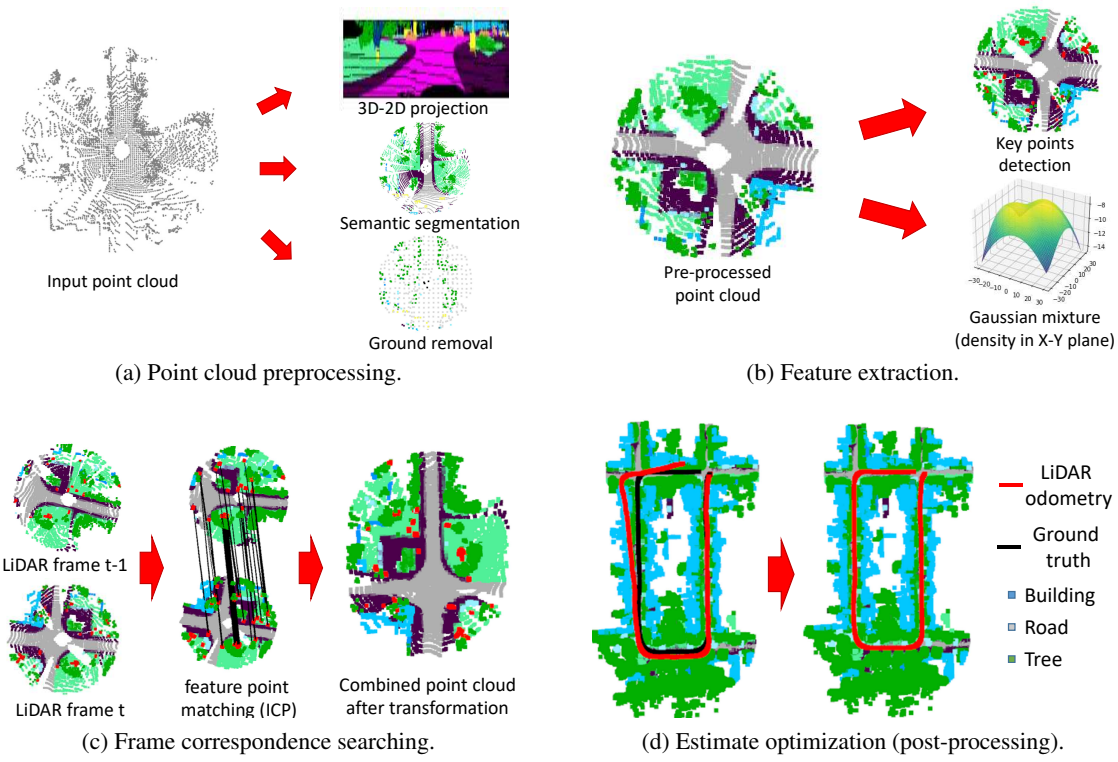
In light of the notable achievements observed in natural language processing, Transformers have sparked considerable interest within the computer vision community. Demonstrating remarkable success across various tasks in the field, including image classification [8] [70], object detection [71] [72], and semantic segmentation [73]. A significant milestone in this trajectory was the introduction of the Vision Transformer (ViT) [8] by Dosovitskiy *et al.*, offering an alternative to conventional CNN-based approaches. Unlike previous methods that grappled with the practical application of self-attention to images, often requiring intricate engineering for efficient execution, ViT sought to showcase the competitiveness of vanilla Transformers when pre-trained on extensive datasets. As part of its approach, input images undergo minimal preprocessing, during which they are flattened into 2D fixed-size patches, thereby mitigating the inherent computational

burden associated with self-attention in lengthy input sequences, followed by an embedding layer. Additionally, learnable positional embeddings are incorporated into patch embeddings to preserve positional information. Following a similar approach to BERT [66], a class token is appended to the sequence of embeddings, with its state at the output of the Transformer encoder serving as input to the classification layer. During pre-training, this layer is implemented using a Multilayer Perceptron (MLP), whereas at fine-tuning time, a single linear layer suffices. Despite the impressive performance of ViT, challenges persist regarding its reliance on large datasets and its limited applicability beyond image classification tasks. Several studies have since been conducted to address these issues and delve deeper into the underlying principles of Vision Transformers (ViTs). Notably, Park *et al.* [74] tackled the model's overfitting tendency on small datasets and highlighted the synergies between ViTs and CNNs, a notion later reinforced by K. Islam [75] and corroborated by implementations combining both architectures. In their research, Park *et al.* illustrated how Visual Transformers excel at capturing low-frequency global signals (e.g., the overall structure of an image), while CNNs effectively extract high-frequency signals (e.g., fine details, edges, and textures), demonstrating their complementary nature. In contrast, Rosário [76] proposed three novel Transformer-based architectures to address the limited exploration of Transformers in VO. These architectures, named PoseFormer2D (PF2D), PoseFormer3D (PF3D), and Spatio-Temporal PoseFormer (STPF), build upon each other by progressively incorporating additional information and tackling the VO problem from different perspectives. PF2D, the baseline architecture, adopts a similar approach to ViT's pre-processing stage, where images are converted into patches, linearly projected, and fed into an encoder-only Transformer. Subsequently, six MLP heads are utilised to regress the vehicle's 6DoF. PF3D expands upon PF2D by incorporating temporal information through the number of received stereo-pair frames. Lastly, STPF diverges from the previous models by splitting the task into two Transformer blocks: one dedicated to spatial attention and another to temporal attention. While PF2D achieves the best results among the three in terms of t_{rel}, their overall performance falls short of geometric approaches. Nevertheless, the methodology offers fresh insights into VO research by leveraging the Transformer model, a domain yet to be fully explored.

2.2 Point Cloud Odometry

Point Cloud Odometry, also referred to as Laser Odometry (LO), is another technique employed to address the challenge of egomotion estimation. Diverging from VO, LO utilises Light Detection and Ranging (LiDAR) sensors with the aim of overcoming the limitations associated with the former. These limitations encompass VO's heavy reliance on scene appearance, making it unsuitable for standalone use in featureless environments or under low-light conditions, such as sandy terrains or during nighttime. In contrast, LiDAR sensors facilitate the creation of highly detailed 3D representations of objects and environments, directly capturing information in world coordinates while remaining unaffected by variations in lighting conditions. The primary objective is to estimate the transformation between consecutive LiDAR frames, commonly referred to

as point clouds, by addressing them in a similar way to the traditional VO pipeline. Illustrated in fig. 2.10, Point Cloud Odometry is summarised into five steps: preprocessing, feature extraction, correspondence searching, transformation estimation, and post-processing.



(a) Point cloud preprocessing.

(b) Feature extraction.

(c) Frame correspondence searching.

(d) Estimate optimization (post-processing).

Figure 2.10: LO pipeline overview [77].

While categorising LO methodologies may not be as clear as in the case of VO, some authors tend to subdivide them based on the type of correspondence made (Jonnavithula *et al.* [77]; Lee *et al.* [78]). An alternative approach would be categorising implementations according to ML and DL techniques, or the absence of such, distinguishing between standard/conventional approaches and learning-based ones. Opting for the former categorisation, Jonnavithula *et al.* [77] provide clarity on the steps present in the LO pipeline. Given the unstructured nature of point clouds, where the neighbourhood concept is not as explicit as in adjacent pixels, the first step involves preprocessing the data by reorganising, segmenting, and filtering for improved feature extraction and matching. Various techniques can be used in this process, including 3D-to-2D projection, semantic segmentation, and ground or dynamic object removal. Feature extraction may follow the VO methodology, leveraging feature detectors like SIFT [30], SURF [31], or ORB [33]. Alternatively, probability densities over the 2D plane, such as those utilised in Normal Distribution Transform (NDT) [79], can be employed to reduce information stored and eliminate the need for the correspondence step. As previously mentioned, the latter categorises the methodology in use and aims to match features between frames. Existing works often exploit keypoint matching algorithms like Iterative Closest Point (ICP) [80], RANSAC [23], or even Neural Networks for this purpose. Finally, motion esti-

mation is accomplished by minimising a loss function between the pairs of correspondences. This process is frequently followed by an optimisation step, leveraging previous estimates to maintain local consistency or incorporating loop closure, which imposes additional pose constraints on the algorithm. However, integrating loop closure requires a dedicated mapping module. In summary, LO methodologies have consistently exhibited superior accuracy compared to VO methods, underscoring the importance of highlighting some of the most influential/state-of-the-art approaches in this field.

ICP [80] stands as one of the earliest approaches to LO, sparking a series of new methodologies aimed at enhancing its performance. The algorithm operates by iteratively minimising an error metric, typically the sum of squared distances between matched point pairs. However, terminating upon reaching a local minimum requires a reliable initial guess. Additionally, it exhibits sensitivity to noise and entails computational expenses associated with the iterative process. Despite these challenges, the algorithm laid the groundwork for some of the most dominant works in the field. For instance, Efficient LiDAR Odometry (ELO) [81], recognised as one of the fastest algorithms documented in the KITTI dataset, builds upon the principles of ICP. Many algorithms encode point clouds in tree-based structures to facilitate neighbour search, yet they struggle to efficiently handle large-scale data. In contrast, ELO follows works which directly map LiDAR measurements into 2D spherical range images, facilitating efficient correspondence finding. It adds to the latter by implementing a bird's-eye-view map that effectively preserves the neighbourhood relationship of ground surface points, mitigating the risk of information loss due to point cloud data sparseness in regions parallel to laser beams. Despite its substantial increase in efficiency, ELO remains comparable in performance to other traditional VO methodologies, such as SOFT-SLAM. In contrast, Continuous-Time ICP (CT-ICP) [82] and KISS-ICP [83] stand out as two top-performing ICP-based implementations among open-source alternatives in the KITTI dataset. The latest iteration of CT-ICP, CT-ICP2, currently leads the open-source ranking [4], with a t_{rel} of 0.58 % and r_{rel} of 0.0012 °/m. This methodology introduces a continuous-time trajectory during the scan, aiming to address the lack of robustness observed in methodologies employing fixed scan distortions based on motion models. It achieves this by parameterising the trajectory into initial and final scan poses and estimating the sensor's pose by interpolating between the two. Moreover, the implementation combines the latter technique with a controlled discontinuity between scans, which enables the model to account for high frequency motions of the sensor. Additionally, CT-ICP is integrated into a comprehensive SLAM module, proposing a novel loop closure procedure to enhance global trajectory estimation. Regarding KISS-ICP, the algorithm prioritises simplicity, aiming to deliver an out-of-the-box working implementation that does not necessitate parameter tuning based on the type of LiDAR sensor used, also supporting different motion profiles and, consequently, types of robots, including ground and aerial ones. The implementation relies on point-to-point ICP combined with adaptive thresholding for correspondence matching, a robust kernel, a motion compensation approach, and a point cloud subsampling strategy. It proves to be a straightforward

[4] As of June 2024

yet effective and versatile solution, comparable to other state-of-the-art approaches.

Dating back to 2014, Lidar Odometry and Mapping (LOAM) [84] stands as another highly significant work in the field. This decade-old methodology achieves remarkable accuracy by breaking down the complex SLAM problem into two algorithms running at different frequencies. One algorithm focuses on odometry, providing high-rate but low-fidelity estimates, while the other performs fine processing to generate maps at a lower rate. This dual-sided architecture ensures real-time problem-solving while allowing the mapping module to incorporate a large number of feature points and employ sufficient iterations for convergence. Features are extracted from sharp edges and planar surfaces by assessing local surface smoothness and then matched to estimate the robot's motion. At the conclusion of a LiDAR sweep, the pose is refined by combining the transforms from the mapping and odometry modules, resulting in a pose estimate with enhanced accuracy. Subsequent iterations of the original methodology have propelled it to third place overall in the KITTI odometry scoreboard [5]. Numerous other works have utilised LOAM as a benchmark for further experimentation and improvement, exemplified by Fast LiDAR Odometry and Mapping (F-LOAM) [85]. The methodology enhances its predecessor's iterative and computationally demanding process by embracing a non-iterative two-stage distortion compensation method. This innovation has resulted in a notable decrease in runtime, achieving speeds of up to 20 Hz on a low-power embedded computer unit. This balance between performance and speed allows the open-source implementation to achieve competitive accuracy at a reduced computational cost.

In a remarkable culmination of conventional LO methodologies, Traj-LO [86] emerges as the most recent [6] advancement in the field. This methodology aims to narrow the gap between LiDAR-only approaches and multimodal systems, which, despite leveraging additional information from Inertial Measurement Units (IMUs) to provide more precise motion compensation, heavily rely on the precise calibration of parameters and potentially struggle in extreme situations where the motion state exceeds the IMU measurement range. Similar to CT-ICP, Traj-LO employs a continuous-time trajectory to tightly integrate geometric information, reflecting the spatial-temporal movement of LiDAR, with kinematics, which impose motion-coherent constraints present in consecutively observed points. By incorporating previously overlooked temporal information, Traj-LO iteratively adjusts point positions without needing motion compensation, ensuring spatial consistency and temporal coherence. Additionally, it utilises linear interpolation, providing a simple parametric approach to describing trajectories, but instead of relying on initial and final points, which may yield unsatisfactory results in cases of aggressive motions (such as those exhibited by handheld devices or aerial vehicles), Traj-LO employs a set number of equidistant small-resolution segments, parameterised with two control poses, essentially comprising multiple linear segments. In terms of performance, this approach rivals state-of-the-art multimodal LiDAR-assisted techniques and even surpasses them in extreme scenarios. Moreover, Traj-LO ranks as the second-best technique in the KITTI dataset among open-source methods, and it is designed to accommodate

[5] As of June 2024
[6] As of June 2024

various types of LiDARs as well as multi-LiDAR systems.

In a parallel trend to Visual Odometry, researchers have explored the potential benefits of leveraging learning-based approaches in Point Cloud Odometry. These novel methodologies aim to address the challenges posed by the nonuniformity and sparsity inherent in LiDAR point clouds, shortcomings often encountered by traditional techniques. One pioneering approach is LO-Net [87], an end-to-end DL solution that learns effective feature representations without requiring explicit feature extraction and matching steps. The scan-to-scan approach employed by LO-Net comprises three interconnected networks jointly trained to estimate a 6 Degrees of Freedom (DoF) relative pose between scans. The first sub-network focuses on estimating the normal vector for each point projected into cylindrical coordinates, while the second sub-network predicts a mask for dynamic regions, thereby enhancing the overall network's robustness. Subsequently, pose estimation is executed by amalgamating the extracted features from both the current and preceding scans, which are then fed into convolutional units. The main network simultaneously performs regression on both translation and rotation, ultimately producing the transformation matrix. Furthermore, the methodology incorporates a mapping block within the estimation framework to refine odometry estimates through scan-to-map matching, leveraging both the normal and mask data. Cho *et al.* proposed another approach, DeepLO [88]. While many approaches favoured a supervised learning framework, DeepLO innovatively incorporated a DL approach adaptable to both supervised and unsupervised training. The former entailed a direct comparison of estimates with ground-truth values, while the latter integrated ICP and field of view loss, with the selection of the training methodology contingent upon the availability of labelled data. The overall process of DeepLO, depicted in fig. 2.11, can be segmented into a feature network and a pose network. Initially, 3D point clouds undergo preprocessing, involving projection into a 2D space akin to LO-Net, but utilizing spherical coordinates this time. Subsequently, the representation undergoes further processing, culminating in the extraction of vertex and normal vector maps, which are then fed into the feature network. Here, a more condensed representation is computed by merging information from two sequential frames. The resulting output is summed into a single vector and forwarded into the pose network to predict the relative motion. DeepLO underwent evaluation on the KITTI dataset using both supervised and unsupervised learning. Remarkably, the unsupervised learning approach yielded superior results compared to the supervised approach, which exhibited tendencies toward overfitting to training sets. However, notwithstanding these advancements, the achieved results still fell short of those attained by state-of-the-art techniques.

Two other approaches freely available and achieving a noteworthy ranking in the KITTI evaluation tool are CAE-LO [89] and SuMa++[90]. CAE-LO utilises efficient computing of a compact 2D structured spherical ring to detect interest points and extracts features from a multi-resolution voxel model, preserving the original 3D shape of point clouds and real-size scale. Moreover, the network is trained in a fully unsupervised manner to enhance its generalizability. On the other hand, SuMa++ builds upon its predecessor, Surfel-based Mapping (SuMa) [91]. This work delves into a SLAM approach by constructing a map augmented with semantic information, estimated using the fully CNN RangeNet++ [92]. The network assigns class labels to each laser scan point

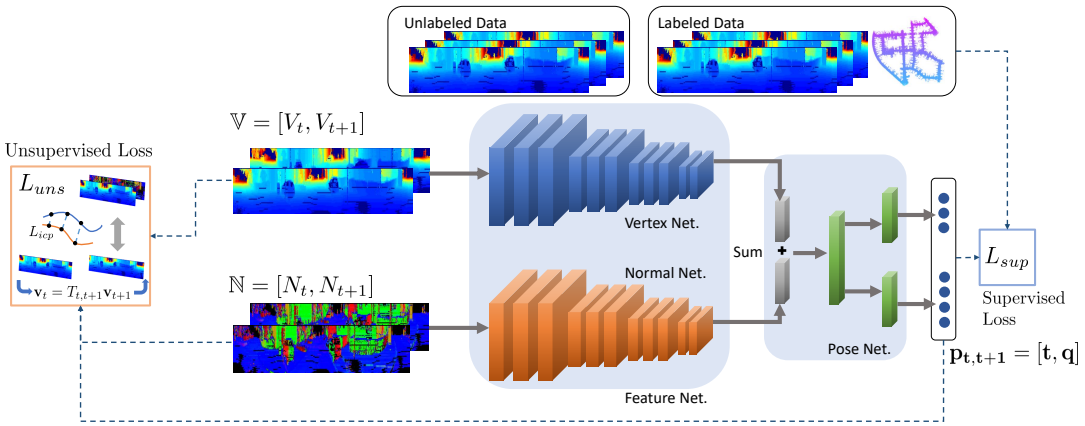


Figure 2.11: DeepLO architecture [88]. The methodology employs Feature Net for compact feature representation, which is subsequently utilised by Pose Net to predict relative motion. In supervised mode, the predicted motion is compared with ground-truth, whereas in unsupervised mode, the output informs the computation of ICP and field of view losses.

by initially projecting the point cloud into spherical coordinates for efficient processing and then back-projecting the classification results to the 3D point cloud. This information is integrated into the surfel-based map and leveraged to remove dynamic objects and improve pose estimation.

The DL approaches presented thus far have predominantly focused on CNN-based architectures for addressing Point Cloud Odometry. However, with the recent advancements in Transformer architectures parallel to those in VO, researchers have begun to explore their potential application to this problem. One such notable approach is TransLO [93]. Similar to several other methodologies, TransLO does not directly feed raw data into the network; instead, it initiates by projecting the point cloud onto a 2D cylindrical surface, which is then inputted into the model. This approach enables the simultaneous processing of a larger number of points. Feature embeddings are subsequently encoded using a coupling between stride-based sampling layers and Window-based Masked Self-Attention (WMSA). The latter introduces a binary mask to handle invalid projected pixels due to the inherent sparsity of point clouds. The estimation of pose between frames is accomplished via a Masked Cross-Frame Attention (MCFA) block. This module also incorporates shifted windows and binary masks like WMSA and is utilised both in the initial pose generation and further refinement blocks. In terms of evaluation, the methodology was compared against other traditional and learning-based approaches in the KITTI dataset. By training the model from sequences 0 through 6 and testing it on the remaining four sequences, TransLO outperformed benchmarks such as LOAM [84] and LO-Net [87]. A very recent approach proposed by Lee *et al.* is the End-to-End LiDAR Odometry using Transformer Framework (ELiOT) [94]. In contrast to previous methods, ELiOT adopts an end-to-end approach that eliminates the necessity for 3D-2D projections. The rationale behind this is that 2D representations may not fully capture the characteristics of 3D data, leading to the potential loss of features. To address this, ELiOT utilises a PointNet++ backbone module, followed by Implicitly Represented Flow Embedding

(IRFE) layers, enabling the extraction of both global and local features for motion flows. Subsequently, a dual-quaternion, representing the translation and rotation components, is predicted via the Transformer-based encoder-decoder architecture. The network is trained in a supervised manner and evaluated using a similar methodology as TransLO. However, the published results indicate poorer performance from the end-to-end framework.

2.3 Multimodal Odometry

Enumerating the potential deployment scenarios for autonomous vehicles is a challenging task. The complexity of scenes, coupled with varying environmental conditions, demands solutions that are highly generalisable and robust to any type of condition or failure. Sensors play a fundamental role in perceiving the vehicle surroundings in an autonomous driving system, and their utilisation and performance can directly impact the safety and feasibility of such systems [95]. Analogous to how humans rely on the commonly recognised five senses, the fusion of multiple sensor modalities, referred to as multimodality, enhances the overall system robustness. These modalities include proprioceptive and exteroceptive sensors, which are already integrated into many modern vehicles and constitute what is commonly termed the *Perception Belt*, as depicted in fig. 2.12.

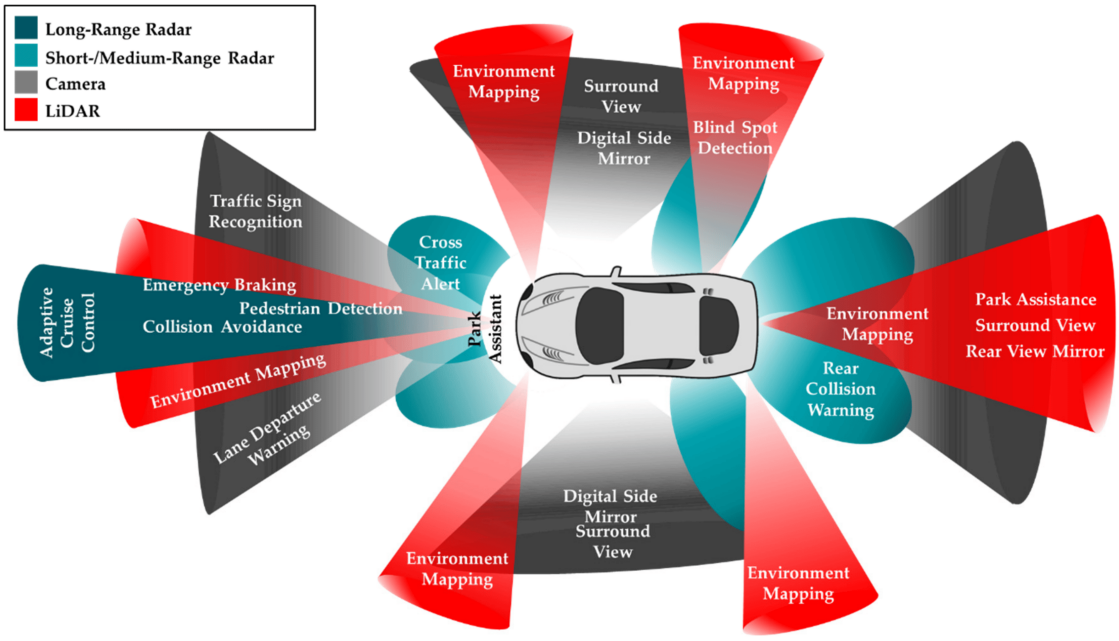


Figure 2.12: Illustration depicting the sensor setup in an autonomous vehicle [95]. Highlighted areas denote LiDAR, camera, and radar coverage, facilitating various applications.

Two of the most prevalent technologies for perceiving the environment are cameras and LiDARs. Cameras are relatively inexpensive and energy-efficient, providing high-resolution images and videos that encompass colour and texture information. However, when applied to VO, cameras are significantly impacted by poor lighting conditions and texture-less environments, further necessitating parameter calibration to rectify geometric distortions. On the other hand, LiDAR

sensors are unaffected by lighting conditions and natively acquire depth information. Renowned for their high accuracy and wide spatial coverage, LiDARs operate using infrared beams or laser light. Nevertheless, they encounter challenges in adverse weather conditions such as rain, fog, and snow, with one study reporting a 25 % degraded performance of the sensor under such conditions [7]. Moreover, LiDARs underperform in non-prominent environments, such as highways or long tunnels, and necessitate meticulous attention to motion distortion. In contrast to these sensors, IMUs serve as internal state sensors that do not require an external reference to estimate the agent's position. Their small size and low power consumption make them ideal for resource-constrained systems. However, these systems tend to exhibit rapid drift due to measurement errors [13]. Consequently, their application to long-range tasks is unfeasible without periodic adjustment. Multimodality plays a crucial role in leveraging complementarity by integrating diverse data sources, thereby facilitating a more comprehensive reconstruction of the environment. Additionally, it reduces the failure cost of individual modalities and introduces redundancy, enabling one modality's strength to aid learning in another. This ultimately leads to lower uncertainty in state estimation and enhances decision-making capabilities.

Categorisation of fusion typically hinges on the processing stage at which data is fused, as illustrated in fig. 2.13. While extensive research has been conducted on environment perception and object detection for autonomous driving, fusion categorisation can be broadened to tasks like motion estimation. This often results in a division into three categories: early, intermediate, and late fusion. Late fusion, for instance, involves determining motion by integrating pose estimations from multiple standalone subsystems. This delayed fusion, also known as decision or loosely coupled fusion, reduces computational complexity by fixing the state space dimension and enhances modularity, as adaptations to individual sensor systems do not necessitate extensive modifications to the existing framework [78]. Moreover, specific weights can be assigned to sensors to ensure robustness in case one sensor underperforms. However, this approach tends to be suboptimal as it requires high computational costs and overlooks correlations between data measurements, potentially discarding valuable intermediate features. In contrast, early fusion, or feature fusion, involves a tight coupling between modalities, with data typically merged at a raw stage. By jointly optimising all sensor measurements, this approach enhances accuracy by incorporating a greater number of constraints. Despite its efficiency, it heavily relies on inter-sensor spatiotemporal calibration and incurs higher complexity [95]. Finally, intermediate fusion, or mid-fusion, occupies a middle ground between the former two. It is considered a more comprehensive approach due to its versatile adoption and strikes a balance between complexity and computational costs, typically involving preprocessing and merging of data before conducting pose estimates.

In line with the categorisation of odometry methodologies, Multimodal Odometry techniques can be further divided into classical sensor fusion and DL sensor fusion. In the context of autonomous driving, specific techniques will be explored more extensively due to their results and contributions to the field. Beginning with classical approaches, V-LOAM [97], a combination of Visual Odometry and Point Cloud Odometry proposed by the same authors of LOAM [84], tightly couples the two modalities to handle aggressive motion and the lack of texture, limitations

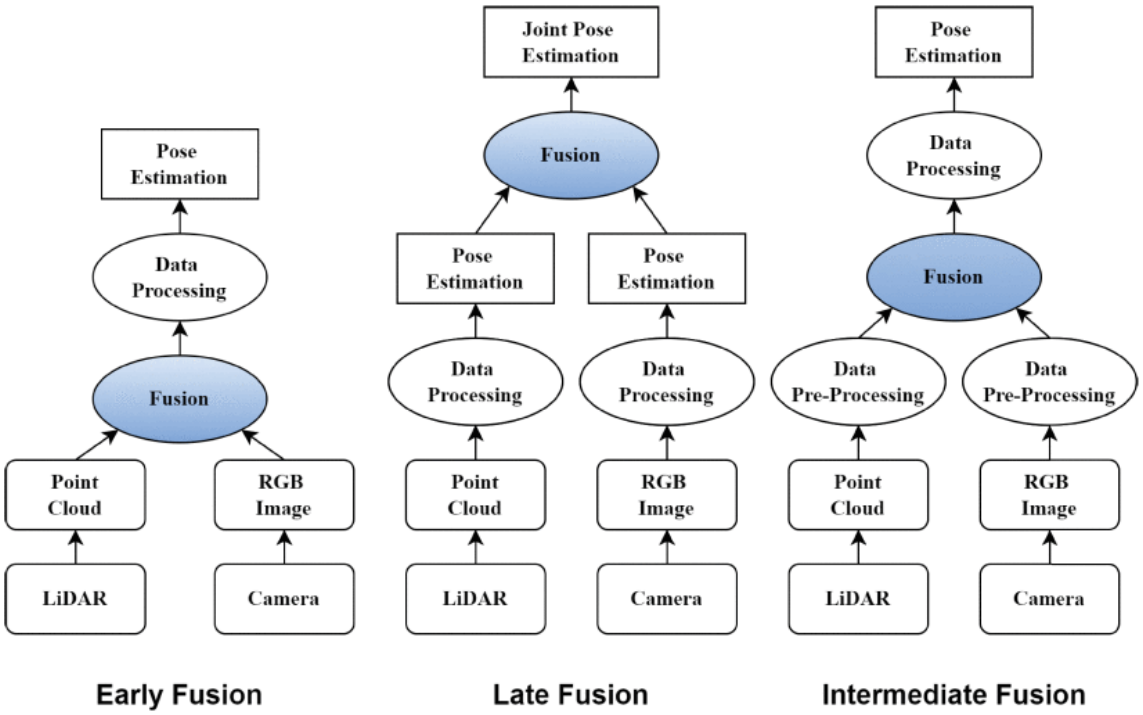


Figure 2.13: Diagram illustrating the different data fusion approaches [96]. Early fusion involves merging raw sensor data at the initial stage, allowing for joint optimisation of all sensor measurements. Late fusion combines outputs from independent subsystems at a later stage, and intermediate fusion inserts between these two approaches, merging preprocessed data before conducting pose estimation.

faced by the individual approaches. It leverages the high frequency of VO to estimate motion and the low drift warrant and robustness in low lighting conditions provided by LO. The VO section is augmented with depth information extracted from LiDAR clouds to facilitate frame-to-frame motion estimation based on visual feature correspondences. Conversely, the LO segment operates once per sweep, processing point clouds similarly to LOAM, yielding distortion-free point clouds used to perform scan-to-map matching, ultimately reducing the drift accumulated from VO and refining motion estimations. Despite its publication in 2015, results in the KITTI evaluation tool [7] rank this methodology as the second best, surpassing its foundation, LOAM. Similarly to V-LOAM, Graeter *et al.* proposed LiDAR-Monocular Visual Odometry, LIMO [98]. The implementation fits into the group of loosely coupled methods. More specifically, it is characterised as a LiDAR-assisted depth-enhanced Visual Odometry technique, which only uses 3D points from LiDAR clouds to provide depth measurements. Firstly, visual features are extracted and tracked from image sequences using the VISO2 implementation [61], chosen for its fast execution. Additionally, a DL algorithm able to semantically label the image is used to prevent feature points present in dynamic objects, such as vehicles or pedestrians, from being utilised. To obtain depth information corresponding to the remaining landmarks, point clouds are projected onto the image

[7] As of June 2024

plane, followed by a one-shot depth estimation algorithm. Finally, separate from pose estimation, a keyframe-based bundle adjustment framework is employed to refine predictions by carefully selecting keyframes and landmarks, thereby enabling lower computational costs, incorporating the estimated depth information for scale optimisation, and implementing a robustification step to remove remaining outlier values. Although not achieving the same performance as the latter approach, LIMO maintains a commendable ranking in KITTI and is accessible as open-source software.

Toward more recent approaches, ORB-SLAM3 [99] represents the latest iteration in the ORB-SLAM algorithm family, building upon the foundations laid by Mur-Artal et al.'s ORB-SLAM2 [53] and ORB-SLAM-VI [100]. This methodology enables Visual or Visual-Inertial SLAM using monocular, stereo, or RGB-D cameras. A notable improvement is the introduction of a novel fast initialisation method based on maximum a posteriori (MAP) estimation, enhancing robustness and accuracy by addressing the shortcomings of slow IMU initialisation observed in previous iterations. Furthermore, the algorithm incorporates a multiple map system, leveraging a novel place recognition method with enhanced recall, allowing the system to navigate through prolonged periods of poor visual information. By exploiting short-, mid-, and long-term data associations and harnessing the multimodal nature of the algorithm, ORB-SLAM3 demonstrates robustness to challenges such as poor texture, motion blur, and occlusions, leading to superior accuracy levels compared to other state-of-the-art systems. While not formally evaluated on the KITTI evaluation tool, the algorithm excels in diverse environments, ranging from small-scale indoor spaces like those captured in the TUM-VI dataset [101] - comprising hand-held recorded scenes with a fisheye stereo-inertial rig - to the EuRoC dataset [102], featuring 11 stereo sequences recorded from a microaerial vehicle (MAV) across varying levels of difficulty, all successfully navigated by the algorithm. However, ORB-SLAM3 faces challenges in low-texture environments due to its feature-based nature. While matching feature descriptors effectively handle long-term and multi-map data associations, they exhibit less resilience compared to direct methods in such contexts. Despite this challenge, the algorithm's rapid estimations, versatility, and high accuracy make it a robust open-source tool well-suited for real-world deployment.

Addressing the SLAM problem from a different perspective, Tightly-Coupled Visual-LiDAR SLAM, TVL-SLAM, proposed by Chou et al. [103], integrates visual and LiDAR information. However, unlike other methods, this implementation does not improve VO through depth estimates nor employ VO as an initial motion guess for LO. Instead, by allowing both the visual and LiDAR frontend modules to operate independently, the implementation leverages all sensor measurement information to attain more accurate estimates and enhance robustness, integrating features at a later stage in the pipeline, a process referred to as intermediate fusion. In the visual SLAM pipeline, ORB-SLAM2 serves as the foundation to extract and match features with a reference keyframe, generating a preliminary pose estimate used in the LiDAR module to aid in point-to-line and point-to-plane residual calculations. Ultimately, residuals from both modules are integrated into a large-scale optimisation problem to refine pose estimation. Moreover, cross-validation between individual motion estimates enables outlier rejection, while inaccurate

camera-LiDAR extrinsics are addressed through an extension to SLAM with calibration. Compared to state-of-the-art methodologies, TVL-SLAM ranks fourth overall in KITTI [8]. In comparison to full-SLAM approaches, i.e., those with loop closing, it emerges as the top performer, achieving a t_{rel} of 0.56 %. Furthermore, experiments conducted on KAIST [51], a complex urban dataset featuring abundant highly dynamic objects and long trajectory scenarios that pose challenges for vision- and LiDAR-only approaches, underscore the efficacy of multimodal sensor fusion and the incorporation of loop closing for reduced drift. Semi-Direct Visual-LiDAR Odometry and Mapping, SDV-LOAM [104], is another approach which builds upon the concept that leveraging all measurements from cameras and LiDAR sensors enhances accuracy. The implementation addresses limitations encountered by conventional VO- and LO-only systems. In the visual scheme, feature-based methodologies like those in V-LOAM face inevitable 3D-2D depth association errors. Direct methods, while capable of avoiding such errors, are prone to falling into local minima and are sensitive to photometric changes. The proposed semi-direct solution combines the strengths of feature-based methods (minimising re-projection error) with those of direct methods (eliminating 3D-2D depth association errors). To achieve this, a novel point extraction method utilises high-gradient points that are well-distributed and present among the set of projected 3D LiDAR points on the image. Addressing scale differences arising from frequency disparities between VO and LO outputs, the approach employs a point-matching algorithm that introduces intermediate keyframes closer to the current frame. Other significant contributions of the work include an adaptive sweep-to-map optimisation method that selects between optimising the full 6 DOF or only the 3 horizontal DOF based on the richness of geometric constraints in the vertical direction and a sweep reconstruction module that operates at the same frequency as image captures, enabling the refinement of every output pose from the visual module by Point Cloud Odometry. The methodology not only demonstrated state-of-the-art results when evaluated in the KITTI dataset but also enhanced the performance of other open-source systems through the adoption of the proposed sweep-to-map LO optimisation.

The use of multimodality across various domains results in the generation of vast amounts of data characterised by diverse types and high integrity. While traditional data fusion techniques can mitigate the limitations of single-modality approaches, they often struggle to fully capture the intricate inter- and cross-modality relationships inherent in complex and imperfect data. The advancement of DL presents an opportunity for bridging the gap between traditional and learning-based methodologies, as it enables the processing and learning from large datasets and the automatic extraction and comprehension of complex associations among multimodal information. The flexibility, high-dimensional representation, and nonlinear mapping capabilities of DL-based approaches are anticipated to significantly enhance the overall performance of multi-sensor data fusion algorithms [105]. Acknowledging the broad spectrum of applications for multimodal systems, each with its specific requirements, and the array of inference mechanisms offered by DL models, such as adaptive learning, deep generative and deep discriminative techniques, identifying

[8] As of June 2024

the optimal solution for egomotion estimation proves to be challenging.

Sun *et al.* proposed TransFusionOdom, an end-to-end supervised Transformer-based framework for LiDAR-Inertial fusion in odometry estimation [106]. This methodology achieves both homogeneous and heterogeneous data fusion through a multi-attention fusion module, as depicted in fig. 2.14. It combines the Transformer encoder architecture with the Soft Mask Attention Fusion (SMAF) approach [107] mitigating potential overfitting from limited amounts of data. Initially, 3D point clouds and IMU signals undergo preprocessing and are then fed into CNN-based backbones for feature extraction. Inter-modality features, including vertex and normal maps, are fused using SMAF, while cross-modality features from LiDAR and IMU measurements are integrated using the Transformer. The incorporation of a mixture of token, positional, and modal-type embeddings benefits the network by reducing computational burden through average pooling, organising point clouds, and providing modality information for each token, respectively. Furthermore, the methodology employs multitask regression, building on the author's previous work, CertainOdom [108], which leverages regressed uncertainty to balance translation and rotation errors. Evaluation on the KITTI dataset demonstrated improved results compared to geometry- and learning-based approaches, with an average t_{rel} of 0.61 % on the validation set (sequences 9 and 10). Finally, the effectiveness of the fusion strategy was validated through an extensive ablation study examining different architectures.

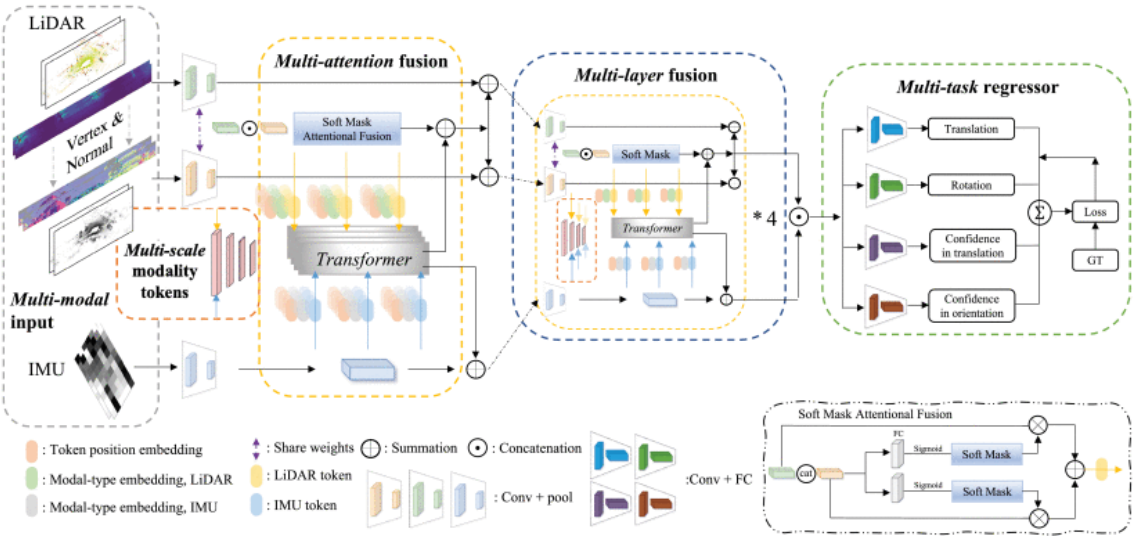


Figure 2.14: TransFusionOdom network architecture [106], integrating LiDAR and IMU sensor inputs. Fusion via SMAF allows for homogeneous fusion of LiDAR vertex and normal maps, while Transformer-based fusion allows for heterogeneous combination of LiDAR-IMU data.

Recent approaches, such as that of Lucas *et al.* [109], have explored the potential of DL-based architectures for egomotion estimation using a late fusion approach. TEFu-Net integrates egomotion estimates from diverse data modalities, including stereo RGB, LiDAR, and GNSS/IMU measurements, utilising an LSTM network to provide robust spatiotemporal estimates and effectively filter erroneous inputs. Experiments have demonstrated the model's ability to improve accuracy

by up to 63% in translation and 21% in rotation compared to input estimators, highlighting the great potential of learning-based technologies to be merged with more traditional approaches.

2.4 Critical Review

The field of odometry is undergoing continuous exploration and enhancement through the application of various techniques and approaches. While geometry-based methods have historically been the primary approaches, the emergence of learning-based methods, which have proven successful in numerous other domains, has garnered significant attention from researchers. This section aims to conclude the state-of-the-art chapter by providing a critical review of the techniques discussed thus far, identifying potential gaps in existing implementations, and establishing connections to the work presented in this dissertation. To achieve this, table 2.1 compiles some of the most relevant odometry works, categorising them into traditional and learning-based approaches, along with their respective modalities. The performance of each technique is evaluated using metrics such as t_{rel} and r_{rel} from the KITTI dataset, with all values obtained from the dataset's evaluation tool [9]. The works are ordered from best to worst based on their t_{rel} results. Additionally, table 2.2 provides an illustrative comparison of learning-based approaches which may not be included in the KITTI evaluation tool. The presented values have been extracted from the relevant papers and represent averaged results obtained from the respective authors' chosen validation sets.

In table 2.1, traditional approaches exhibit clear predominance, with marginal performance improvements observed among the top rankings. It is noteworthy that the majority of these approaches employ either laser sensors or a combination of visual and laser modalities, with the exception of SOFT2 [57]. This trend can be attributed to the limitations faced by cameras under low-light conditions and in texture-less environments, where LiDAR sensors remain unaffected and offer broader field of view coverage. However, while trajectory accuracy is paramount, metrics such as robustness and computational efficiency must also be considered. Many techniques opt to sacrifice accuracy for reduced computational overheads, exemplified by ELO [81], which boasts the fastest runtime among the presented approaches at an average of 0.005 s but ranks lower in performance. Robustness, however, poses a more challenging metric to quantify, with authors often evaluating techniques under diverse and demanding scenarios. While KITTI stands as the most widely evaluated dataset for autonomous driving and thus serves as the benchmark for comparison, other datasets, such as KAIST [51], offer more recent and rigorous evaluation scenarios to address robustness concerns. For instance, TVL-SLAM [103], a multimodal approach evaluated using the KAIST dataset, outperformed both visual- and LiDAR-only approaches, albeit at the expense of increased computational load due to data fusion and the deployment of a mapping module. Striking a balance between computational efficiency and algorithmic complexity may offer a potential solution to this challenge.

[9] https://www.cvlibs.net/datasets/kitti/eval_odometry.php

Table 2.1: Comparative analysis of traditional and learning-based methods discussed throughout this chapter. Metrics t_{rel} and r_{rel} were assessed using the KITTI evaluation tool. V: Visual, L: Laser.

	Method	**Modality**	$t_{\text{rel}}(\%)$	r_{rel} ($^\circ/100\,\text{m}$)
Traditional	SOFT2 [57]	V	0.53	0.09
	V-LOAM [97]	V/L	0.54	0.13
	LOAM [84]	L	0.55	0.13
	TVL-SLAM [103]	V/L	0.56	0.15
	CT-ICP2 [82]	L	0.58	0.12
	Traj-LO [86]	L	0.58	0.14
	SDV-LOAM [104]	V/L	0.60	0.15
	ELO [81]	L	0.68	0.21
	ORB-SLAM2 [53]	V	1.15	0.27
Learning-based	D3VO [63]	V	0.88	0.21
	DVSO [62]	V	0.90	0.21
	Suma++ [90]	V	1.06	0.34

A closer examination of recently proposed geometry-based approaches, such as Traj-LO [86] and SDV-LOAM [104], alongside longstanding approaches like LOAM [84], reveals a lack of performance improvement relative to the increased complexity. In fact, owing to the well-established foundations of these types of approaches, they may be reaching a state of maturity and convergence towards a developmental plateau. Algorithms must exhibit high generalisability to navigate diverse scenarios, ranging from open spaces to crowded environments with dynamic objects. Additionally, robustness is critical; solutions must address adverse conditions such as lighting variations and harsh weather, which may degrade sensor data quality. As discussed in section 2.3, multimodal approaches offer promising avenues for addressing the limitations of single-sensor implementations by leveraging complementarity and redundancy. On the other hand, addressing the inherent complexity and unpredictability of real-world deployment scenarios poses challenges that are difficult to manage with traditional hand-crafted formulations. Hence, harnessing the capabilities of DL to capture intricate underlying data aspects presents a viable option for enhancing system robustness and adaptability simultaneously. Exemplified by D3VO [63] and DVSO [62], learning-based approaches have commenced bridging the gap between conventional and non-conventional techniques.

The illustrative example presented in table 2.2 showcases the potential of new learning-based approaches, with TransFusionOdom [106] outperforming all single-modality methodologies. However, challenges persist, particularly the scarcity of high-quality, labelled data, which poses significant obstacles to selecting appropriate learning techniques. These limitations are particularly

Table 2.2: Illustrative analysis of learning-based methods discussed throughout this chapter, not included in the KITTI odometry ranking. Metrics t_{rel} and r_{rel} represent the achieved scores on the validation sets, as reported in the respective research papers. V: Visual, L: Laser, I: Inertial. * indicates Tranformer-based methods.

Method	Modality	$t_{rel}(\%)$	r_{rel} ($^\circ/100\,m$)
TransFusionOdom [106]*	L/I	0.61	0.71
LO-NET [87]	L	0.83	0.42
TransLO [93]*	L	0.99	0.50
DeepVO [64]	V	5.96	6.12
ELiOT [94]*	L	7.59	2.67
DeepLO [88]	L	9.59	3.99

impactful for architectures such as Transformers, which excel in retaining contextual information and managing large datasets efficiently but require extensive high-quality training data due to their substantial parameter capacity. Moreover, Transformers suffer from quadratic increases in time and memory complexity with input sequence length. The associated costs of data labelling and the reluctance of companies to share resources due to competitive concerns further complicate advancements. As emphasised by Wang *et al.* [10], the emergence of novel DL techniques is pivotal for future odometry advancements. This underscores the importance of exploring DL-based Multimodal Odometry techniques, highlighting the relevance of research presented in works such as this dissertation.

Chapter 3

Methodology

This chapter focuses on the developed methodology for egomotion estimation. It discusses the processes involved in obtaining and processing the input data, details the proposed solution, and outlines the research path that supports the reasoning behind the decisions made.

3.1 Introduction

As discussed in chapter 2, the problem of egomotion estimation involves determining an agent's positional change over time. In the field of autonomous driving, particularly in the context of ground vehicles, this problem can be simplified from the usual 6 degrees of freedom (DoF), illustrated in fig. 3.1, to a 3 DoF problem. Here, two components represent the vehicle's translation, and the third represents its orientation, commonly denoted as *(x, y, yaw(ψ))*, assuming movement in the *xOy* plane. Each estimate is relative to the previous pose of the agent, quantifying the change in position and orientation from the last timestep. While such a spatial representation may be limited in environments with high degrees of mobility, for ground vehicle's egomotion estimation, relative changes in the vertical axis *(z)*, pivots *(roll(ϕ))*, and tilts *(pitch(θ))* are reduced when compared to the other axes. For example, although it is common for vehicles to make high-degree turns, such as in roundabouts, it is rare for a car to pivot abruptly or experience significant vertical movement compared to its forward or backward motion.

The estimation problem involves two phases: extracting features from input signals and regressing positional changes based on the derived features. D3VO and DeepLO exemplify this approach, utilising different CNN-based backbones and feature/pose networks. In multimodal methodologies, it is crucial to identify at which stage inputs should be fused. This concept, elaborated in section 2.3, is addressed with intermediate fusion techniques. These intuitively tackle the problem by preprocessing and merging data (features) before performing pose estimation. Motivated by the aforementioned advantages over other fusion methods, this dissertation aims to implement a mid-level fusion network, detailed in fig. 3.2. The proposed approach involves regressing a 3 DoF pose from multimodal data, such as optical flow and depth projections from point clouds. This methodology must effectively handle the heterogeneous nature of sensor data,

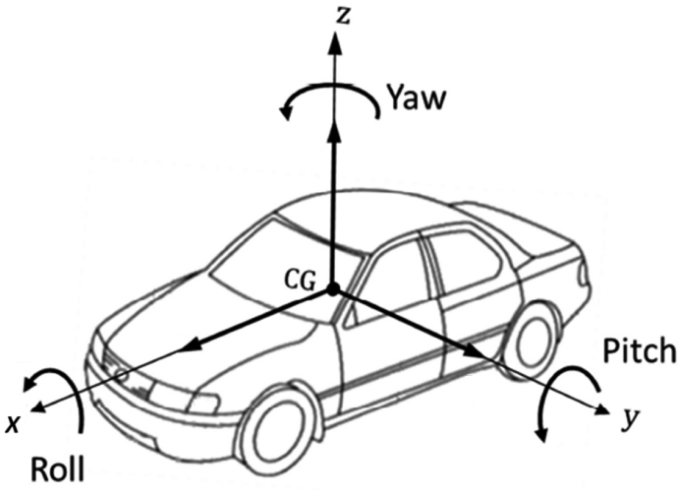


Figure 3.1: Representation of the Vehicle Axis System (6 DoF) [110]. The diagram illustrates the three positional axes *(x, y, z)* and the three rotational degrees of freedom *(roll, pitch, yaw)*.

including varying data formats, different levels of reliability and susceptibility to noise. Extracting and merging complementary features from diverse sensor modalities — specifically, cameras and LiDAR — is an inherently complex task with the potential to enhance robustness and improve overall performance.

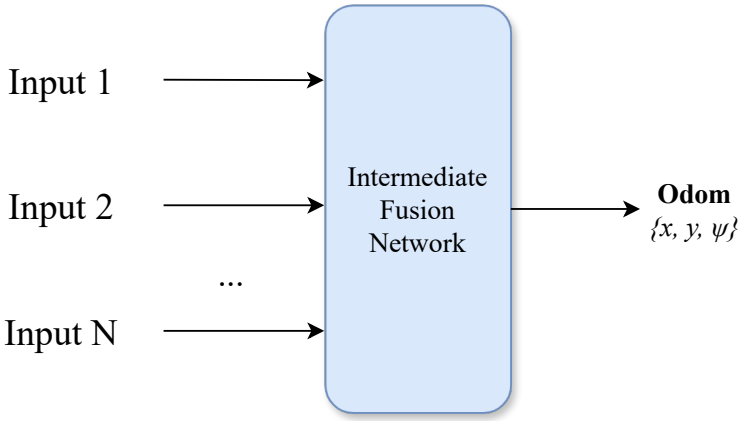


Figure 3.2: Input/output configuration of the intermediate fusion network. The inputs consist of raw sensor data, which undergo preprocessing before being fused to generate odometry estimates in the format of *x, y, yaw (ψ)*.

The following section provides a comprehensive explanation of the data acquisition and preprocessing steps undertaken to train the network and effectively detect patterns from the input data.

3.2 Data Processing and Augmentation

Given that the core objective of this work is to leverage different sensor modalities, a dataset comprising temporally and spatially aligned data allows for a simplification of the fusion problem. For this purpose, the well-established KITTI vision benchmark suite [12] was employed due to its diverse range of on-road scenarios and synchronised, rectified data across multiple sensor modalities. The setup used in KITTI includes two stereo pairs of grayscale and colour cameras, a LiDAR, and an Inertial Navigation System (INS) composed of GPS and IMU. The dataset is organised into 22 sequences, with 11 sequences providing ground truth trajectories sourced from the INS, while the remaining 11 sequences are designated for evaluation purposes. The cameras operate at a frequency of 10 Hz and produce images at two different resolutions: *Height, Width* = 376, 1241 pixels for sequences 00-03, and *Height, Width* = 370, 1226 pixels for the remaining sequences. Additionally, calibration data is provided, which includes information on transforming the reference frame between any cameras, from camera to LiDAR and from camera to GPS/IMU.

The data explored in this work are high-resolution camera images and point clouds from LiDAR. Images contain detailed colour and texture information, which can be leveraged to extract meaningful features crucial for egomotion estimation. Conversely, point clouds generated by LiDAR inherently provide depth information, reducing the need for complex computations and yielding more accurate spatial measurements. This additional data, as detailed in the following description, was utilised as input to train the proposed network:

Image: Images were selected from a single coloured camera (left), as illustrated in fig. 3.3a. These inputs were subjected to random colour jitter, where their original brightness, contrast, and saturation levels were varied to enhance the model's ability to generalise to unseen data. Furthermore, the RGB values were normalised according to ImageNet's mean and standard deviation, as further detailed in section 3.3.

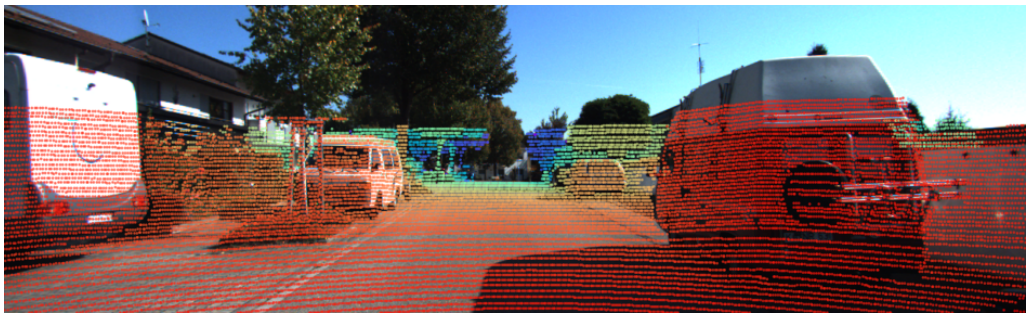
Optical Flow: Derived from appearance-based methodologies, optical flow provides additional temporal information that complements the static spatial data contained in images. Estimated using a pre-trained RAFT model [111] and visually represented in fig. 3.3b, optical flow captures motion vectors for each pixel across consecutive frames in both x and y axes. This capability enables the estimation of dynamic scene changes and object trajectories, which are essential for accurate egomotion estimation. Finally, optical flow values were normalised between 0 and 1 based on their maximum and minimum values for each axis, followed by standardisation to achieve scales similar to the other inputs.

Depth: Leveraging LiDAR point clouds, depth information was extrapolated by projecting these points onto the 2D image plane using the calibration data provided by the KITTI dataset. Despite the constrained vertical field of view of the LiDAR compared to the camera, as shown in fig. 3.3c, the captured depth values provide more accurate spatial measurements, complementing the information derived from images. Similar to other inputs, these values were normalised and standardised, taking into consideration the entire dataset.

To accommodate the different existing image resolutions and the LiDAR's constrained field of view, a standard image size was chosen. All inputs were cropped to a resolution of *Height, Width* = 219, 1152 pixels. This size was determined using the Eigen crop [112] method, a technique specifically validated on the KITTI dataset that accurately aligns the image with depth boundaries.

(a) RGB image providing information about the scene's colours and textures.

(b) Optical flow visualisation with a circular colour palette, where each colour corresponds to a specific flow direction.



(c) Combined image overlaying the original scene with the LiDAR depth projection in a reverse rainbow colour scheme. Red denotes close points, and blue indicates distant ones.

Figure 3.3: Network inputs pre-crop and data augmentation.

A fundamental step prior to training any DL model is the specification of three data sets corresponding to distinct stages of the pipeline: training, validation, and testing. A common split is 60/20/20 percent, respectively, although the overall size of the dataset must be considered. While sufficient training data is essential for the model to learn effectively, the validation and test sets must also be representative of the acquired data. Given that the KITTI dataset is relatively small, with only 11 sequences containing ground truth - one of which (03) lacks LiDAR sensor data - the choice of training, validation, and test splits was determined according to other DL implementa-

tions, namely D3VO [63] and LO-Net [87]. Although these methods train on different sequences, a common ground is the testing on sequences 09 and 10.

Of the 22,400 images available for the left-colour camera, 87 % were allocated for training, covering sequences 00-08 (excluding 03), while the remaining 13 %, sequences 09-10, were reserved for testing. Within the training allocation, 74 % of the overall dataset was used for actual training and 13 % for validation purposes. The train-validation split was randomly applied to each sequence, as well as the order in which sequences were inputted to the network during training. This approach aimed to ensure that the train and validation sets contained a diverse range of trajectory timestamps, enhancing the model's ability to generalise to various real-world scenarios. During testing, the sequence order was maintained to be representative of real-world applications.

3.3 Multimodal Intermediate Fusion Network Architecture

Building on the tasks of feature extraction and pose estimation, the overall model architecture, illustrated in fig. 3.4, is designed to effectively process these components by employing specialised architectures for each module. For feature extraction, two ResNet-based [113] architectures are employed due to their proven efficacy in capturing detailed representations from input data. One ResNet backbone processes the high-resolution images, while the second backbone handles the concatenated optical flow and depth information. Residual Networks facilitate the flow of gradients during backpropagation through residual (skip) connections, enabling the training of deeper networks. This capability allows the model to capture complex features and spatial hierarchies crucial to the problem at hand. For pose estimation, the features extracted by both ResNet backbones are concatenated into a single dimension and passed through a parallel set of three Multilayer Perceptrons (MLPs), each tasked with regressing one of the 3 DoF relative pose values. This task-specific approach enables each MLP to optimise for the unique characteristics of each dimension. Additionally, MLPs are proficient at learning non-linear mappings between input data and output values, demonstrating significant potential for accurate odometry estimation.

Exploring further the encoder block, which incorporates two ResNet18 backbones, each composed of the architecture detailed in table 3.1. ResNet18 was selected due to its simplicity and reduced memory footprint. Additionally, pre-trained weights on the ImageNet dataset for the task of classifying images into 1000 different categories were utilised. The original ResNet was modified by removing the final fully connected layer, thereby leveraging the network's capability of converting input images into a new representation format, which consists of a set of relevant extracted features. As discussed in section 3.2, RGB images were normalised using the mean and standard deviation values provided by the ImageNet dataset: $mean = [0.485, 0.456, 0.406]$ and $std = [0.229, 0.224, 0.225]$. Furthermore, the adapted ResNet architecture received as input a 4D tensor with shape $[B_{atches}, C_{hannels}, H_{eight}, W_{idth}]$. One backbone was dedicated to encoding the RGB image, a task extensively evaluated and tested in other scenarios. The second ResNet backbone was configured to encode the optical flow concatenated with the depth projection, aiming to extract accurate motion features from the x and y displacement values combined with the

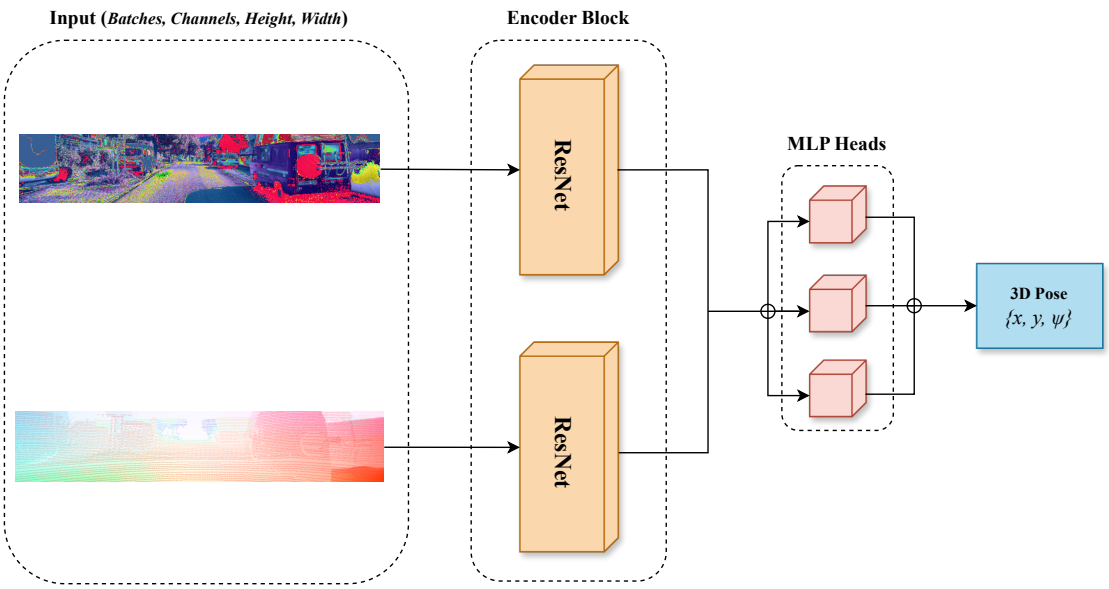


Figure 3.4: Overall proposed network architecture. The top input shows the RGB image after data augmentation, cropping, and normalisation according to ImageNet mean and standard deviation values. The second input illustrates the concatenated optical flow with depth projections. Each ResNet backbone extracts features from these inputs, which are concatenated via the $\oplus$ operation. The concatenated features are fed into the MLP heads, which estimate the 3D relative pose comprising the x, y, and *yaw* values.

Table 3.1: Overview of the layers comprising the adapted ResNet18 [113] architecture. Including the layer type, output size (height, width, channels), activation function, and additional details on the kernel size, number of output channels, and stride (where applicable for convolutional layers).

	Layers	**Output Size**	**Activation**	**Details**
Initial Layers	Conv1	$112 \times 112 \times 64$	ReLU	7×7, 64, stride 2
	MaxPool	$56 \times 56 \times 64$	-	3×3, stride 2
Layer Set 1	Conv2_1	$56 \times 56 \times 64$	ReLU	3×3, 64
	Conv2_2	$56 \times 56 \times 64$	ReLU	3×3, 64
Layer Set 2	Conv3_1	$28 \times 28 \times 128$	ReLU	3×3, 128, stride 2
	Conv3_2	$28 \times 28 \times 128$	ReLU	3×3, 128
Layer Set 3	Conv4_1	$14 \times 14 \times 256$	ReLU	3×3, 256, stride 2
	Conv4_2	$14 \times 14 \times 256$	ReLU	3×3, 256
Layer Set 4	Conv5_1	$7 \times 7 \times 512$	ReLU	3×3, 512, stride 2
	Conv5_2	$7 \times 7 \times 512$	ReLU	3×3, 512
Final Layer	Avg Pool	$1 \times 1 \times 512$	-	Global Average Pooling

spatial depth. Normalisation was applied to all three inputs — images, optical flow, and depth — to ensure consistent scaling and mitigate potential initial misleading biases.

The resulting extracted features, 512 from each backbone, are concatenated into a tensor of shape $[B_{atches}, F_{eatures}]$, with a total feature dimension of 1024. Compared to the original network inputs, this reduced set is expected to contain the most meaningful information, such as fine details, edges, and textures extracted from images, as well as motion information from the optical flow/depth input. The regression of the 3 DoF values is handled by a dedicated MLP head, whose configuration is illustrated in fig. 3.5. The features are passed through a linear layer that doubles their dimension, allowing the model to capture more complex and rich representations. This is followed by batch normalisation, the ReLU activation function, and a dropout layer, which randomly zeros out some elements with a probability of 30 %. This combination helps prevent overfitting, forcing the network to learn more robust features that generalise better to unseen data while also maintaining consistency in activation functions across the entire architecture. Before the final regression, the same pattern of layers is applied, but this time, the feature dimensions are compressed from 2048 to 512, ensuring that only the most important information is retained. Although other configurations were explored, including different dropout factors and solely decreasing dimensionality, the presented head configuration yielded the best output predictions.

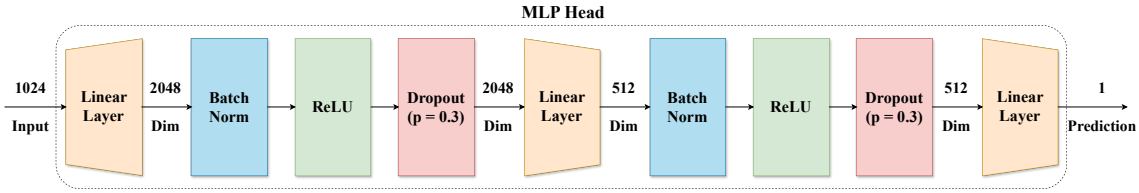


Figure 3.5: Multilayer Perceptron (MLP) head architecture used for regressing each of the 3 DoF values.

3.4 Loss Function

In supervised learning, the loss function measures the discrepancy between the predictions and their respective targets. Within the context of autonomous driving, two primary components - translation and rotation - must be addressed according to their distinct characteristics: translation involves the linear movement of the vehicle along the x and y axes, while rotation pertains to the angular movement around these axes, affecting the vehicle's orientation. It is, therefore, essential to handle these components separately and weigh them according to their importance and impact on accuracy. A careful definition of the loss function is mandatory as even a slight deviation from the ground truth over a long sequence can cause significant discrepancies in the overall trajectory, leading to localisation misjudgements or failures in following designated routes in real-world scenarios. The following loss function was designed to train the proposed network:

$$\mathscr{L} = w_{tr} \cdot \left(\frac{1}{n} \sum_{i=1}^{n} \|\mathbf{p}_{i,tr} - \mathbf{t}_{i,tr}\|^2 \right) + w_{rt} \cdot \left(\frac{1}{n} \sum_{i=1}^{n} \|\text{norm}(\mathbf{p}_{i,rt}) - \text{norm}(\mathbf{t}_{i,rt})\| \right), \tag{3.1}$$

where w_{tr} and w_{rt} are the weights assigned to the translation and rotation components, respectively. The terms $\mathbf{p}_i$ and $\mathbf{t}_i$ denote the predicted and target vectors for the i-th sample in a batch of size n. Specifically, $\mathbf{p}_{i,tr}$ and $\mathbf{t}_{i,tr}$, are vectors represented by (x, y), while $\mathbf{p}_{i,rt}$ and $\mathbf{t}_{i,rt}$ are represented by (ψ). The function *norm* normalises the rotation values to the range $-\pi$ and π rad.

The translation loss is computed using the Mean Squared Error (MSE), which is the mean of the squared Euclidean distances (L_2 norms) between the predicted and target translation vectors over all n values in a batch. The L_2 norm is chosen for its ability to quantify the magnitude of the error in each dimension of the translation vector and focus on reducing large deviations attributed to its sensitivity to outliers resulting from the squaring operation. Conversely, the rotation loss must be addressed differently due to its cyclical nature, unlike the linearity of translation errors. Both predictions and targets are normalised between $-\pi$ and π rad, removing representation ambiguity, after which the shortest angular difference between the latter is computed and averaged over the batch. Finally, the two components are combined into a global loss using a weighted sum based on w_{tr} and w_{rt} weights.

3.5 Optimizer

Through an iterative training process, neural network parameters are adjusted via optimization algorithms, enabling the model to improve predictions by effectively learning from data. These algorithms aim to minimise the value computed by the loss function through backpropagation. This study focuses on analysing the impact of two different optimizers, Stochastic Gradient Descent (SGD) [114] and AdamW [115], each distinguished by their update rules, which dictate how weight updates are computed. SGD operates by estimating the gradient of the loss function using a randomly selected subset of the dataset in each iteration. This approach reduces computational costs, allowing for frequent parameter updates. However, the random sampling introduces noisy updates, which can lead to fluctuations in the loss function and potentially slow convergence to the optimal solution. In contrast, AdamW is an extension of the Adam [116] optimizer that addresses some of its limitations, particularly regarding weight decay. The algorithm combines the benefits of Adam's adaptive learning rates and momentum with weight decay regularisation, promoting more stable training and helping to prevent overfitting by penalising large weights. A crucial hyperparameter for both optimizers is the learning rate, which determines the model's step size towards optimal parameters after each iteration, thus influencing the convergence rate. While selecting a good initial value for the learning rate is important, dynamically adjusting it during training can enhance performance and reduce overfitting. This concept, known as learning rate scheduling, was implemented using a method called ReduceLROnPlateau [1]. This scheduler reduces the learning rate when a specified metric has plateaued or stopped improving. The

[1] For example https://pytorch.org/docs/stable/generated/torch.optim.lr_scheduler.ReduceLROnPlateau.html

scheduler's parameters, such as the reduction factor and the number of epochs allowed with no improvement, require extensive testing and hyperparameter tuning.

Overall, the use of the AdamW or SGD optimizers in conjunction with the ReduceLROnPlateau scheduler balances the need for efficient convergence with the ability to prevent overfitting, thereby enhancing the robustness and accuracy of the trained neural network.

Chapter 4

Results

This chapter presents a comprehensive analysis and discussion of the outcomes produced by the proposed network. The impact of various parameters on the training process is assessed by examining the evolution of the loss throughout training, followed by the presentation of the evaluation metrics used to compare the results against a variety of state-of-the-art methodologies, encompassing single-sensor to multimodal techniques and both geometric and learning-based approaches. Additionally, the performance of the proposed model is assessed under more challenging scenarios, simulated by artificially degrading input data.

For the scope of the analysis, only the presented 3 DoF values are considered by the evaluation metrics, as the model exclusively estimates these. As detailed in section 3.2, the KITTI dataset, comprising 11 ground truth sequences, was used for training and testing the model, as well as for comparing it against state-of-the-art methodologies. However, since sequence 03 lacked LiDAR data, it was excluded from the presented results. Given that the camera's forward and lateral movements correspond to the z and x axes, respectively, sequence trajectories are depicted accordingly. All technical coding aspects regarding the intermediate fusion model were developed using the PyTorch library, which facilitates the rapid configuration and construction of deep neural networks. The hardware specifications used for training the model included an Nvidia GeForce RTX 2080 GPU with 8GB RAM.

4.1 Model Optimization

To optimize the model's learning process, the impact of various parameters, specifically the optimizer and batch size, was thoroughly studied. For this, a baseline for comparison was established, keeping most network hyperparameters listed in table 4.1 constant during training. The initial learning rate of $1e^{-4}$ and 30 training epochs were chosen based on D3VO [63], which also employs a ResNet18 architecture on the KITTI dataset. Although D3VO used 20 epochs for training, the combination of the ReduceLROnPlateau scheduler and an additional 10 training epochs aimed at achieving a more fine-tuned model.

Table 4.1: Network hyperparameters used for testing the impact on the model's learning process.

Hyperparameter	
Number of input modalities	3
Number of outputs	3
Optimizers	SGD, AdamW
Scheduler	ReduceLROnPlateau
Activation function	*ReLU*
Learning rate (with scheduler)	$1e^{-4}$
Batch size	12
Training epochs	30

Prior to examining the impact of the optimizer, different batch sizes were explored. Larger batch sizes can lead to faster convergence and more stable gradients, but they also require more memory and may result in overfitting. Conversely, smaller batch sizes offer a regularisation effect due to noisier gradient estimates, potentially improving generalisation. To balance these constraints, hardware limitations, and the trade-off between speed and stability, batch sizes of 8, 12, and 16 were tested. Overall, a batch size of 12 yielded the best results and was thus set as constant for subsequent experiments.

The influence of SGD and AdamW optimizers was studied. Both optimizers were used in conjunction with the ReduceLROnPlateau scheduler to dynamically adjust the model's learning rate. This scheduler was configured with a patience of 5 epochs, a threshold of 0.5 %, and a decreasing factor of 0.1, meaning the learning rate would be reduced by this factor after 5 successive epochs where the validation loss plateaued. These hyperparameter choices provided a balance between the total number of epochs trained and the patience window, incorporating a moderate improvement threshold. Fig. 4.1 illustrates the evolution of training and validation losses throughout the training process. While both metrics are crucial for assessment, the validation loss provides a better understanding of the model's performance on unseen data, which, despite having similar characteristics to the training data from the same KITTI sequence, is more indicative of real-world scenarios. Analysis reveal that the model converges in both scenarios, with AdamW optimizer yielding superior results. This is especially pronounced in the earlier stages of training, where the model converges much faster with AdamW, demonstrating better compatibility with the ReduceLROnPlateau scheduler. Additionally, the model's generalisation capabilities are corroborated by the validation loss being lower than the training loss throughout the training process. This result can be attributed to several factors: the use of regularisation techniques such as dropout, which is only applied during training, as expected; the calculation of training loss over an entire epoch, which includes initial higher losses as the model improves; and the model's improved performance by the time validation occurs. Moreover, the smaller size of the validation set could also contribute to this discrepancy. These findings indicate promising robustness and generalisation of the model to unseen data, which will be further explored in the following sections.

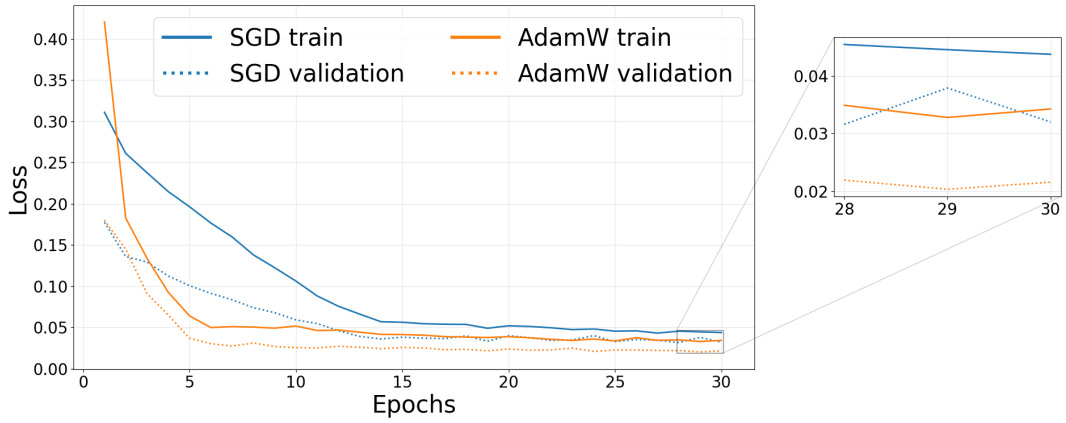


Figure 4.1: Evolution of training and validation loss for the proposed method using SGD and AdamW optimisers combined with the ReduceLROnPlateau scheduler.

Based on the studies presented above, the network parameters yielding the best validation performance will be used for further evaluation in the subsequent sections. These include the AdamW optimizer with the ReduceLROnPlateau scheduler and a batch size of 12, along with the additional parameters detailed in table 4.1.

4.2 Model Evaluation on Real-World Driving Scenarios

A set of standard metrics was used to evaluate the performance of the proposed model against other odometry implementations. Translational error (t_{rel}), Rotational error (r_{rel}), Absolute Trajectory Error (ATE), and Relative Pose Error (RPE) are commonly used measures that assess methodologies based on both relative and global errors, quantifying the deviation of individual predictions and evaluating the overall produced trajectory. The latter are described as follows:

- **Translational error** (t_{rel}), expressed as a percentage, is a relative evaluation metric. It is calculated by averaging the Euclidean distance from the predicted and target translation, (z,x), vectors over all possible subsequences ranging from 100 to 800 m, in increments of 100 m.

$$t_{\text{rel}} = \frac{100}{N} \sum_{i=1}^{N} \frac{t_{\text{err},i}}{l_i}, \tag{4.1}$$

  where N is the total number of valid subsequences, $t_{\text{err},i}$ represents the translational prediction error from the ground truth, computed by the Euclidean distance for the evaluated axes, and l_i is the length of the subsequence. The Translational error is multiplied by 100 to convert it to a percentage.

- **Rotational error** (r_{rel}) is a relative metric for evaluating the rotation component (*yaw*) estimates, following a similar logic to the Translational error. This error is expressed in degrees per 100 meters,

$$r_{\text{rel}} = \frac{180 \times 100}{\pi \times N} \sum_{i=1}^{N} \frac{r_{\text{err},i}}{l_i}, \tag{4.2}$$

where N represents the number of valid subsequences, $r_{\text{err},i}$ represents the rotational prediction error in radians, and l_i denotes the length of each subsequence. The equation calculates the average rotational error over all valid subsequences, normalising each rotational prediction error by its respective subsequence length. The sum of these normalised errors is then averaged over N, converted from radians to degrees by multiplying by $\frac{180}{\pi}$, and finally scaled by 100 to express the error per 100 m.

- **Absolute Trajectory Error** (ATE) measures the similarity between the predicted trajectory and the actual trajectory by calculating the Root Mean Squared Error (RMSE) of the Euclidean distances between corresponding poses in meters.

$$\text{ATE} = \sqrt{\frac{1}{N} \sum_{i=1}^{N} \|\mathbf{t}_{i,\text{tr}} - \mathbf{p}_{i,\text{tr}}\|^2}, \tag{4.3}$$

where N is the total number of poses in the sequence, $t_{i,\text{tr}}$ and $p_{i,\text{tr}}$ represent the ground-truth and predicted position at the i-th pose, respectively. $\|\cdot\|$ denotes the Euclidean norm, which computes the distance between the predicted and ground-truth positions. ATE is sensitive to outliers, making it an effective metric for identifying significant localisation failures throughout the trajectory. However, this sensitivity can also be a drawback, as large errors can dominate its value, making it easier to overshadow inlier trajectory errors, i.e. smaller, more typical errors in the trajectory.

- **Relative Pose Error** (RPE), similar to t_{rel} and r_{rel}, assesses the accuracy of pose estimation by computing average errors in both translation and rotation relative to consecutive poses within a sequence. Specifically, RPE calculates the mean translation error in meters and rotation error in degrees, capturing how accurately the algorithm estimates spatial and rotational displacements between successive poses.

$$\begin{aligned} \text{RPE}_{\text{tr}} &= \frac{1}{N-1} \sum_{i=1}^{N-1} t_{\text{err},i}, \\ \text{RPE}_{\text{rt}} &= \frac{180}{100 \times (N-1)} \sum_{i=1}^{N-1} r_{\text{err},i}, \end{aligned} \tag{4.4}$$

where N denotes the total number of poses in the trajectory. Terms $t_{\text{err},i}$ and $r_{\text{err},i}$ quantify the translation and rotation errors between consecutive poses i and $i+1$, considering the

evaluated axes, respectively. The sum of these errors is averaged over the number of pose pairs $(N-1)$. For RPE_{rt}, the rotation error is converted to degrees, while for RPE_{tr}, the translation error remains in meters.

4.2.1 Odometry benchmarking

The following subsection compares the proposed method with other state-of-the-art odometry techniques. This comparison includes geometric and learning-based methodologies that rely on both single and multimodal data, enabling an assessment and validation of the strengths and limitations of each across various scenarios. The details in which each methodology was tested will also be closely examined, alongside their overall performance under the provided KITTI ground-truth sequences.

The methodologies used for comparison were the following:

CT-ICP [82] [1]: An Iterative Closest Point (ICP) based implementation for Point Cloud Odometry, CT-ICP is integrated into a SLAM module with a loop closure procedure for improved global trajectory estimation. However, the methodology was executed without loop closure and with *constant velocity* motion compensation to accommodate the constraint that *corrected scans (from KITTI) prevent CT-ICP from utilising its elastic formulation* [82].

ORB-SLAM3 [99] [2]: Building upon the foundations of ORB-SLAM2 [53], this implementation enables Visual or Visual-Inertial SLAM using monocular, stereo, or RGB-D cameras. The algorithm incorporates a multiple map system and leverages a novel place recognition method. As no IMU data was considered in this work, only the Visual SLAM component with stereo cameras was used for testing. Additionally, loop closure was not employed, ensuring a fair comparison with the proposed methodology.

SDV-LOAM [104] [3]: Visual-LiDAR Odometry and Mapping system that integrates two distinct frontend modules independently. This implementation utilises all camera and LiDAR sensor measurements to achieve more accurate estimates and enhance robustness. Despite being open source, only its visual module is provided, meaning it operates as a semi-direct depth-enhanced method. The implementation is similar to the proposed model in that both are multimodal and leverage depth projections from point clouds.

DeepVO [64] [4]: End-to-end supervised learning approach which employs a Recurrent Convolutional Neural Network (RCNN) architecture. This methodology directly infers poses from a sequence of raw images, leveraging the CNN's ability to convert data into concise and meaningful representations, as well as the RNN's internal state, which enables the modelling of the sequential dynamics inherent to Visual Odometry (VO). The methodology was tested using an unofficial implementation that provided pre-trained weights on KITTI ground-truth sequences 00, 01, 02, 05, 08, and 09.

Table 4.2: Comparison of results from geometric (+) and learning-based (∗) odometry approaches across 10 ground-truth driving sequences from KITTI (excluding sequence 03) against the proposed network. Evaluation metrics encompass global and relative measures, such as t_{rel}, r_{rel}, ATE, and RPE for translation and rotation. Results have been rounded to two decimal places, with bold highlighting indicating the best performance for each sequence, up to three decimal places. Learning-based approaches present results for both trained and tested sequences.

	Method	Sequence									
		00	**01**	**02**	**04**	**05**	**06**	**07**	**08**	**09**	**10**
t_{rel} (%)	**CT-ICP** +	**0.44**	**0.65**	**0.35**	**0.32**	**0.24**	**0.28**	**0.29**	**0.59**	**0.43**	**0.45**
	ORB-SLAM3 +	1.26	1.25	1.49	1.67	0.76	1.10	0.79	0.97	1.14	0.89
	SDV-LOAM +	0.72	0.96	1.22	0.96	0.84	0.48	0.70	1.13	0.91	1.38
	DeepVO *	57.57	89.45	68.11	17.83	39.58	44.08	58.24	64.21	79.00	110.90
	Ours *	11.05	51.94	20.38	17.23	17.03	17.38	23.75	15.95	22.97	24.57
r_{rel} (°/100 m)	**CT-ICP** +	0.16	**0.08**	**0.10**	**0.04**	**0.10**	**0.06**	**0.11**	**0.15**	**0.08**	**0.12**
	ORB-SLAM3 +	**0.12**	0.27	0.15	0.07	0.14	0.10	0.23	0.18	0.09	0.13
	SDV-LOAM +	0.20	0.14	0.35	0.56	0.38	0.09	0.38	0.39	0.20	0.58
	DeepVO *	26.60	10.00	23.75	1.03	29.15	31.92	50.07	26.75	23.68	21.85
	Ours *	4.19	7.32	6.25	1.33	6.44	5.13	11.62	5.37	7.20	10.93
ATE (m)	**CT-ICP** +	**7.95**	**5.12**	**12.24**	**0.67**	**3.12**	**1.58**	**0.62**	16.64	**2.57**	**2.23**
	ORB-SLAM3 +	10.73	22.13	19.97	3.58	4.46	4.22	1.77	**12.97**	7.06	7.92
	SDV-LOAM +	16.79	100.53	50.46	6.00	19.01	2.49	3.43	18.45	10.45	15.66
	DeepVO *	289.28	207.36	465.16	32.22	217.57	151.29	46.40	398.82	173.37	99.90
	Ours *	189.79	950.63	556.77	36.61	252.07	60.79	112.29	435.38	138.58	86.27
RPE (m)	**CT-ICP** +	0.02	**0.03**	0.02	**0.02**	0.02	0.02	0.02	0.02	0.02	0.02
	ORB-SLAM3 +	0.02	0.04	0.03	0.03	0.02	0.02	0.01	0.02	0.02	**0.01**
	SDV-LOAM +	**0.02**	0.04	**0.02**	0.03	**0.01**	**0.01**	**0.01**	**0.02**	**0.02**	0.02
	DeepVO *	0.62	2.18	1.07	0.28	0.52	0.84	0.59	0.78	0.91	1.04
	Ours *	0.06	1.08	0.12	0.26	0.04	0.12	0.04	0.04	0.10	0.06
RPE (°)	**CT-ICP** +	0.03	0.02	0.02	0.01	**0.02**	0.02	**0.02**	**0.02**	0.02	0.02
	ORB-SLAM3 +	**0.03**	**0.02**	**0.02**	**0.01**	**0.02**	**0.02**	0.02	**0.02**	**0.02**	**0.02**
	SDV-LOAM +	0.04	0.03	0.03	0.03	0.03	0.02	0.02	0.03	0.02	0.03
	DeepVO *	0.67	0.40	0.58	0.06	0.45	0.42	0.63	0.55	0.63	0.49
	Ours *	0.17	0.26	0.18	0.04	0.14	0.13	0.17	0.15	0.27	0.25

Table 4.2 presents the results of the evaluation metrics obtained for both the proposed method and the described state-of-the-art methodologies. The analysis of these results reveals that CT-ICP consistently outperforms other odometry techniques across most metrics. This outcome is expected, considering CT-ICP's status as the best-performing open-source implementation on the KITTI evaluation tool [5] in this analysis. Similarly, the performance of other geometric implementations, ORB-SLAM3 and SDV-LOAM, was notorious, as evidenced by their very small RPE errors, with a maximum translational RPE of 0.04 m for both in sequence 01 - a highway scenario, illustrated in fig. 4.2, where the vehicle's speed is significantly higher than in other sequences. These errors in the centimetre range demonstrate the high performance of geometric approaches. Remarkably, the proposed methodology was able to compete with the latter approaches, achieving

[5] As of June 2024

Figure 4.2: Sequence 01 (highway) scenario featuring numerous moving objects.

a comparable performance in many of the training sequences and an impressive RPE of 0.06 m in the tested sequence 10.

Shifting the focus to learning-based approaches, the proposed methodology surpasses DeepVO's results across most metrics in all sequences. When analysing the relative pose errors, the model maintains an average translational RPE of 0.097 m, excluding the outlier value of 1.08 m in sequence 01, from training to 0.080 m in testing sequences (09 and 10). However, there is a noticeable performance decrease in rotation, with the average rotational RPE increasing from 0.155° during training to 0.260° in testing. This corroborates the premise that rotation is harder to learn and thus requires a higher relative weight in the overall loss function. Despite DeepVO demonstrating the worst overall performance among all methods, its Absolute Trajectory Error (ATE) is lower in some training sequences compared to the proposed method. While ATE is a useful metric, in implementations where global mapping and loop closure are not incorporated, it does not hold the same relevance as relative pose errors. This is because these methodologies do not align to an overall world reference but rather focus on relative estimates.

Following the quantitative analysis of the proposed method against other state-of-the-art implementations, a qualitative assessment of the predictions for the tested KITTI sequences is crucial. These sequences feature winding roads with varying scenarios. Sequence 09, illustrated in figs. 4.3a and 4.3b, transitions from a countryside background to the suburbs. Sequence 10, depicted in fig. 4.3c, presents paved, narrower roads in a suburban area. Figs. 4.4a and 4.4b showcase the estimated trajectory of the compared methodologies under the latter sequences. While both translation and rotation estimates are significant, small discrepancies in rotation, particularly at the beginning of a sequence, can lead to substantial overall trajectory deviations. Both DeepVO and the proposed implementation, which encounter difficulties in learning very abrupt rotations, suffer from significant trajectory discrepancies, as evidenced by the sharp curve at the beginning of sequence 10.

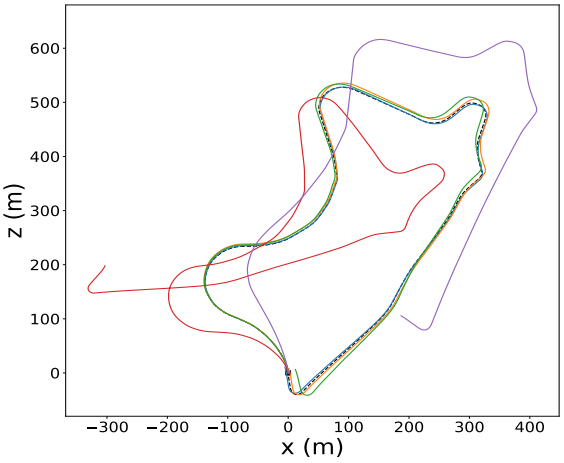
Fig. 4.5 showcases the RPE evolution of the proposed method for both rotation and translation in sequence 10. The analysis of rotation RPE, depicted in fig. 4.5a, evidences two distinct moments where the rotation estimates decrease in performance, indicated by peak rotation errors at the beginning (0 to 6 s) and in the last third of the sequence (89 to 93 s). Analysing both the trajectory

(a) Sequence 09 countryside background.

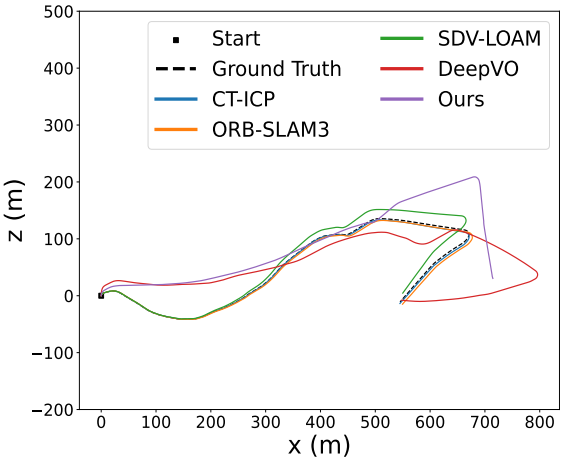
(b) Sequence 09 suburban area.

(c) Sequence 10 paved, narrow roads.

Figure 4.3: Scenarios for KITTI sequences 09 and 10: Sequence 09 includes a transition from a countryside setting to a suburban environment while sequence 10 features narrower, paved roads in a suburban area.



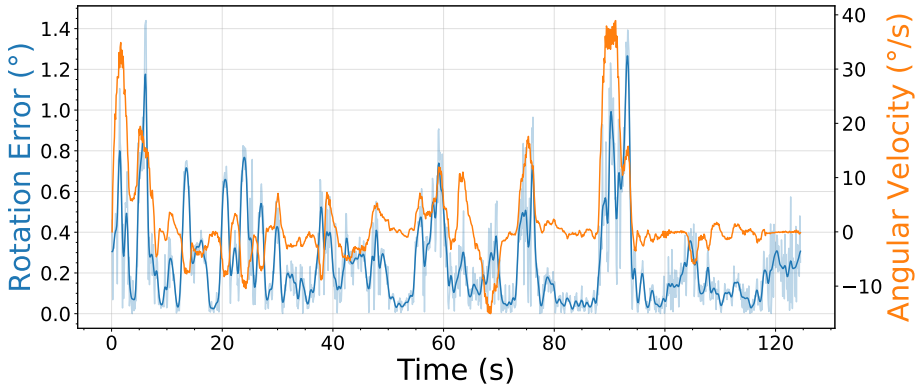
(a) Sequence 09.



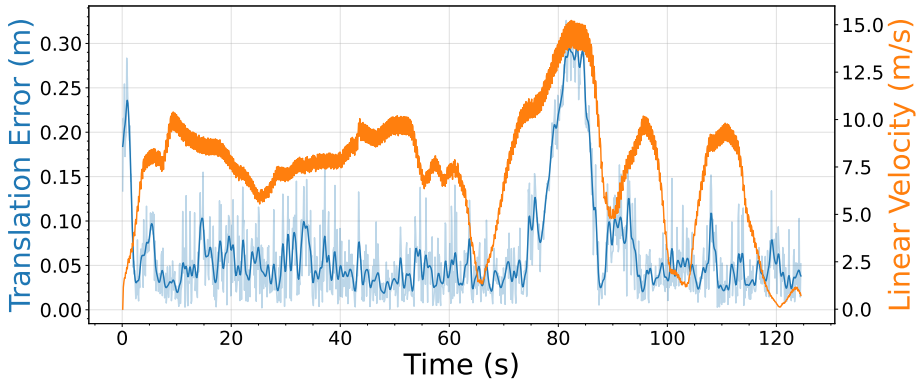
(b) Sequence 10.

Figure 4.4: Estimated trajectories of all compared methodologies on KITTI sequences 09 and 10, both in residential scenarios.

and the combination of rotational RPE evolution with the vehicle's instantaneous angular velocity also presented in fig. 4.5a, these moments correspond to sharp curves where the average vehicle's angular velocity is 17.7 °/s and 28.2 °/s, respectively. These values are significantly higher than the sequence's average angular velocity of 4.6 °/s, explaining why the model struggles in these instances. Similarly, the evolution of the RPE relative to translation, fig. 4.5b, showcases two high points, which only partially match the same instances of rotation error increase, suggesting that a correlation with increased rotation error is not definitive. One primary reason for the decrease in performance relative to translation estimates could be related to the vehicle's high linear velocity. This is corroborated by the high error of up to 0.3 m between 80 to 85 s, where the average linear velocity is 14.3 m/s, much higher than the sequence's average linear velocity of 7.5 m/s. Additionally, a similar trend is visible in sequence 01, which involves a highway scenario with



(a) Evolution of rotational RPE and angular velocity with sequence elapsed time.

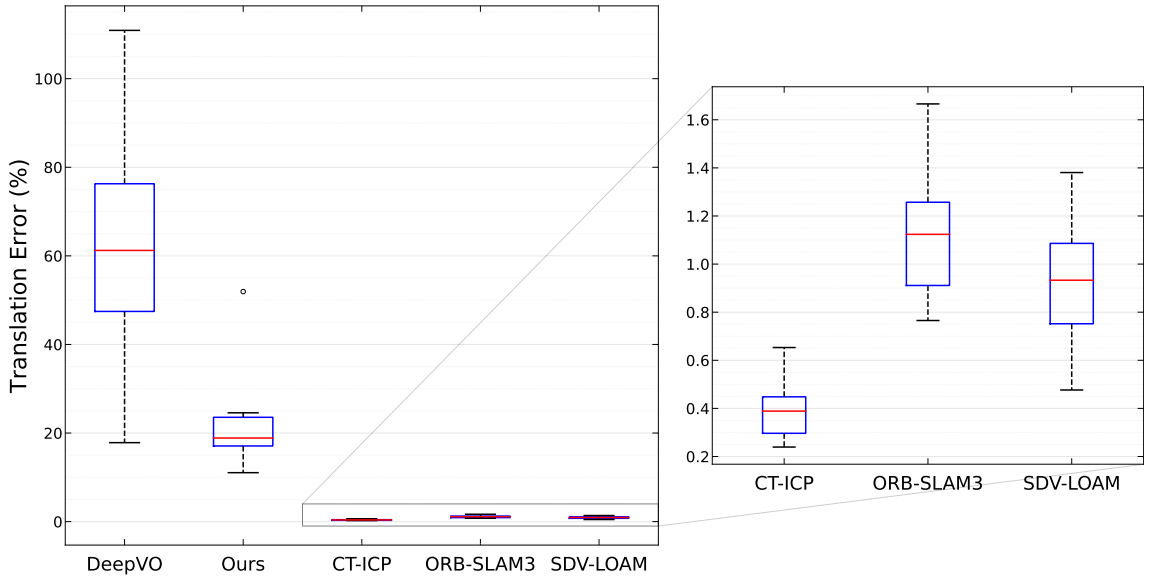


(b) Evolution of translational RPE and linear velocity with sequence elapsed time.

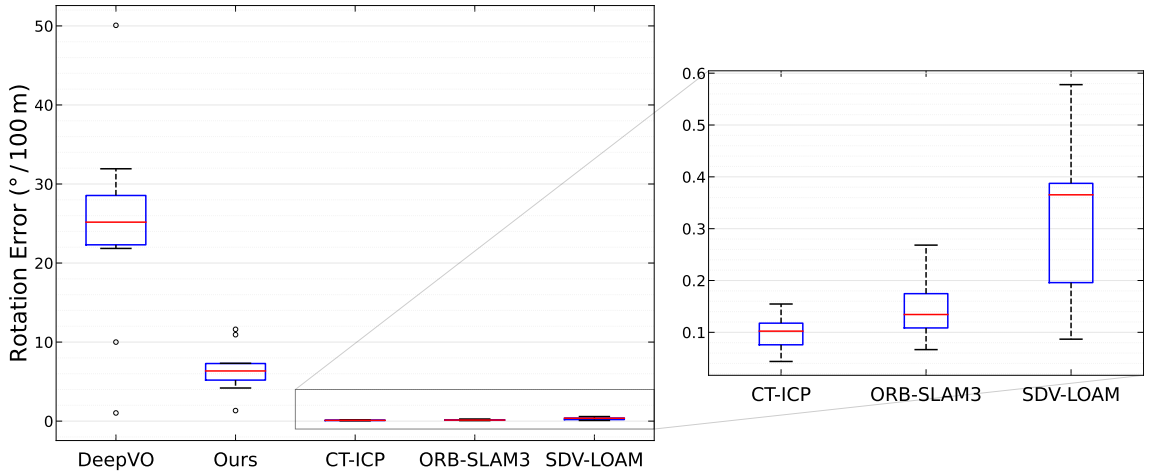
Figure 4.5: Evolution of rotational and translational Relative Pose Error (RPE) (blue), combined with angular and linear velocity (orange), respectively, for the proposed methodology on the KITTI ground-truth sequence 10.

high vehicle speeds. In this sequence, the translation RPE of 1.08 m, as presented in table 4.2, is the highest among all evaluated sequences, also corresponding to the highest vehicle's average linear velocity of 21.5 m/s. However, this conclusion does not hold for the peak in translation RPE present from 0 to 3 s, where the vehicle's linear velocity is low. This scenario appears to be an outlier due to the combination of a very sharp curve and the vehicle's initial movement. This peculiar situation negatively affects both translation and rotation estimates.

The box plots of the translation error (t_{rel}) and rotation error (r_{rel}) across all KITTI ground-truth sequences, illustrated in fig. 4.6, provide a detailed view of the methodologies' performance. CT-ICP stands out as the most reliable approach, exhibiting superior performance in both metrics, with deviations of 0.41 % and 0.11 °/100 m between the maximum and minimum translation and rotation errors, respectively. The other geometric approaches, SDV-LOAM and ORB-SLAM3, achieve comparable median translation errors, as shown in fig. 4.6a, with values of 0.93 % and 1.12 %, respectively. For Rotational errors, fig. 4.6b, ORB-SLAM3 yields estimates close to those of CT-ICP, as demonstrated by the intersection of their interquartile ranges. Regarding learning-based approaches, the proposed methodology shows an improvement of 65 % in Translational and 73 % in Rotational errors relative to DeepVO's average performance, considering all evaluated



(a) Translation Error (t_{rel}) distribution.



(b) Rotation Error (r_{rel}) distribution.

Figure 4.6: Distributions of translation (t_{rel}) and rotation (r_{rel}) errors for all evaluated odometry techniques across all KITTI ground-truth sequences under evaluation. The graph whiskers represent the minimum and maximum registered values, extending from the box to the farthest data point lying within 1.5 times the interquartile range (IQR) from the box. The boxes' limits represent the 25th and 75th percentiles, and the red horizontal lines represent the distribution median. Outliers are denoted by small black circles representing points past the end of the whiskers.

KITTI sequences. Additionally, the multimodal network demonstrates its ability to learn translation effectively, given the very small deviation from the upper box limit (Q3), 23.56 %, and the maximum whisker value of 24.57 %, corresponding to sequence 10, which illustrates the small decrease in performance from training to test sequences. Conversely, one of the two outliers in the rotation error distribution, depicted in fig. 4.6b, corresponds to the test sequence 10, while sequence 09's error of 7.20 °/100 m lies in the top 75% of values, indicating that the model struggles to maintain the same rotation performance levels as in training compared to translation.

4.2.2 Challenging scenarios

Although the KITTI dataset provides a diverse set of on-road driving sequences, including residential and highway scenarios with challenging situations such as dense vegetation, dynamic objects, and partial occlusions, as illustrated in fig. 4.7, it does not fully address environmental uncertainties that lead to reduced data quality, due to harsh weather conditions. To evaluate robustness, a crucial metric presented in chapter 2, it is necessary to consider these additional challenging scenarios. Consequently, experiments were conducted by artificially degrading data, aiming to assess the model's ability to extract meaningful features and produce accurate pose estimates comparable to normal condition contexts.

(a) Densely vegetated road affecting LiDAR reflecting signals.

(b) Fast-moving sequence (highway) with dynamic objects affecting pose estimates.

(c) Partial front occlusion from a moving vehicle.

Figure 4.7: Challenging scenarios for odometry techniques in KITTI sequences, including dense vegetation, dynamic objects, and occlusions.

Brightness adjustments (fig. 4.8b), Gaussian noise (fig. 4.8c), and blur (fig. 4.8d) were applied to the original tested KITTI sequence images. Additionally, optical flow was estimated from these degraded images, and Gaussian noise was applied to depth projections. By applying these techniques to all sensor modality data, the objective was to evaluate the model's robustness across a variety of challenging scenarios:

- **Brightness and Contrast noise** simulates variations in lighting conditions, which are common in real-world driving scenarios. For instance, driving through areas with alternating sunlight and shade or under changing weather conditions can cause significant image brightness and contrast fluctuations.
- **Gaussian noise** was applied to simulate sensor imperfections and environmental interferences that can occur during data acquisition. This noise is represented by random variations in pixel values, following a Gaussian distribution, which can degrade the quality of data.
- **Blur noise** represents scenarios where the camera experiences motion blur or focus issues, which are common in dynamic driving environments. This type of noise degrades the sharpness of the images, making it challenging for the model to extract clear and distinct features.

(a) Original image.

(b) Brightness adjusted image.

(c) Image with applied Gaussian noise.

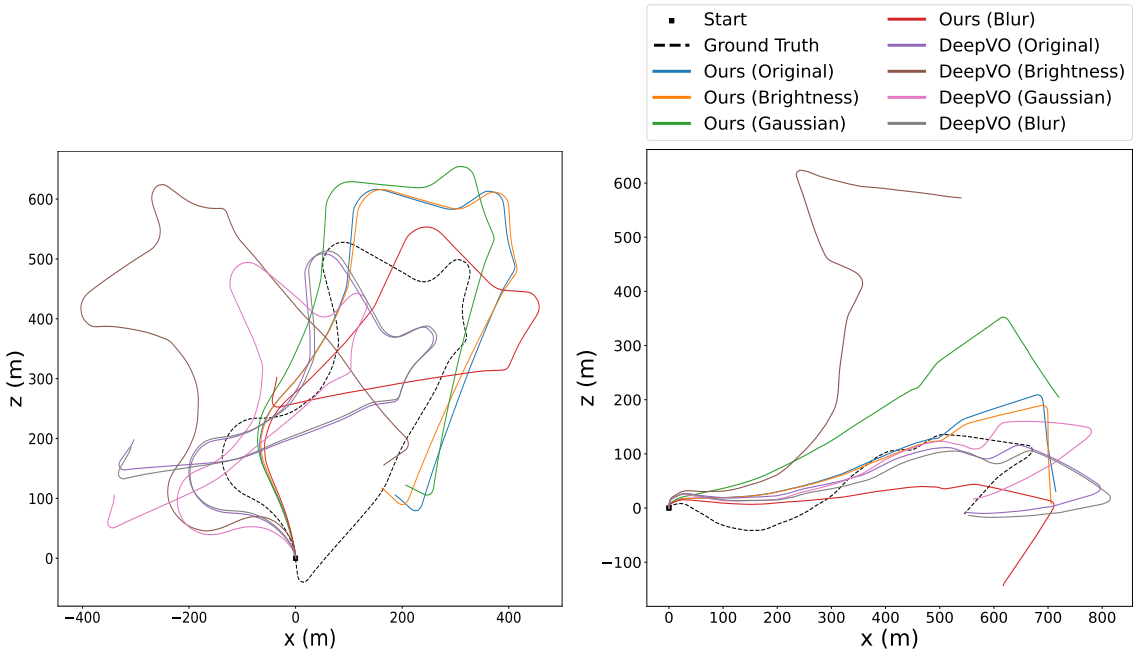
(d) Blurred image.

Figure 4.8: Image representations of randomly introduced noise factors against an original image from testing sequences (09, 10), including brightness and contrast changes, Gaussian noise, and blur effects.

Table 4.3 presents a comparative analysis of DeepVO and the proposed network's performance on both normal and artificially degraded data. Upon analysis, it is evident that the application of noise has a minor impact on the model's results under the tested KITTI sequences. This conclusion is corroborated by the very small differences in RPE values - maximum disparity of 0.01 m in translation and 0.03° in rotation - across all scenarios. These results are indicative of the model's robustness in more challenging environments, representative of reduced data quality. Surprisingly, DeepVO also maintained its RPE values relative to the original data, despite some random variations. For instance, there was a decrease in translational RPE from 0.91 m to 0.61 m with brightness adjusted data in sequence 09. However, there were also significant performance drops, such as the increase in ATE from 99.90 m to 402.12 m under brightness adjusted data in sequence 10. This behaviour may not represent robustness but rather some model randomness in certain adjusted scenarios. Additionally, fig. 4.9 illustrates the predicted trajectories for all experiments on the tested KITTI sequences. In sequence 09, both blur and Gaussian noise negatively impacted the proposed methodology's overall trajectory, with respective ATE values of 145.44 m and 155.39 m, compared to the original data ATE of 138.58 m. Brightness adjustments, on the other hand, had a more severe effect on DeepVO's performance, following a similar trend in sequence 10. Overall, Gaussian noise resulted in the most significant performance drop for the proposed network, particularly in the rotation component, evidenced by the largest discrepancies in Rotational error (r_{rel}), with values of 8.33 °/100 m and 12.58 °/100 m compared to the original data performance of 7.20 °/100 m and 10.93 °/100 m for sequences 09 and 10, respectively. While DeepVO's performance was negatively influenced by brightness adjustments, as depicted in both sequence trajectories of fig. 4.9.

Table 4.3: Comparison of DeepVO's and the proposed network's performance on normal and artificially degraded data from KITII tested sequences (09 and 10). Evaluation metrics include global and relative measures, such as t_{rel}, r_{rel}, ATE, and RPE for translation and rotation. Results have been rounded to two decimal places, with bold highlighting indicating the best performance for each sequence and data.

	Data	**Method**	**Sequence**	
			09	**10**
t_{rel} (%)	Original	DeepVO	79.00	110.90
		Ours	**22.97**	**24.57**
	Brightness adjusted	DeepVO	49.98	73.80
		Ours	**23.37**	**24.33**
	Gaussian noise	DeepVO	71.55	119.35
		Ours	**25.84**	**26.58**
	Blurred	DeepVO	78.83	106.55
		Ours	**23.88**	**24.85**
r_{rel} (°/100m)	Original	DeepVO	23.68	21.85
		Ours	**7.20**	**10.93**
	Brightness adjusted	DeepVO	24.56	24.29
		Ours	**7.38**	**10.36**
	Gaussian noise	DeepVO	23.10	22.47
		Ours	**8.33**	**12.58**
	Blurred	DeepVO	23.60	21.70
		Ours	**7.97**	**8.66**
ATE (m)	Original	DeepVO	173.37	99.90
		Ours	**138.58**	**86.27**
	Brightness adjusted	DeepVO	363.02	402.12
		Ours	**136.48**	**80.78**
	Gaussian noise	DeepVO	215.09	**87.54**
		Ours	**155.39**	166.16
	Blurred	DeepVO	171.16	113.35
		Ours	**145.44**	**93.96**
RPE (m)	Original	DeepVO	0.91	1.04
		Ours	**0.10**	**0.06**
	Brightness adjusted	DeepVO	0.61	0.71
		Ours	**0.10**	**0.07**
	Gaussian noise	DeepVO	0.83	1.10
		Ours	**0.11**	**0.07**
	Blurred	DeepVO	0.90	1.00
		Ours	**0.10**	**0.07**
RPE (°)	Original	DeepVO	0.63	0.49
		Ours	**0.27**	**0.25**
	Brightness adjusted	DeepVO	0.63	0.50
		Ours	**0.28**	**0.24**
	Gaussian noise	DeepVO	0.63	0.49
		Ours	**0.30**	**0.26**
	Blurred	DeepVO	0.63	0.49
		Ours	**0.28**	**0.24**



(a) Sequence 09.

(b) Sequence 10.

Figure 4.9: Estimated trajectories of DeepVO and the proposed methodology on the degraded KITTI tested sequences 09 (Residential) and 10 (Residential).

Chapter 5

Conclusions and Future Work

This dissertation demonstrates the potential of DL, specifically CNN-based architectures, in handling multimodal sensor data for predicting 3DoF relative poses. The proposed solution builds on two key tasks: feature extraction and pose estimation. Feature extraction is addressed by employing two ResNet-based backbones, each responsible for capturing detailed representations from high-resolution images and concatenated optical flow with depth information, respectively. The fusion of the extracted features, known as intermediate fusion, due to the stage at which these are merged, is followed by the pose estimation. This module is composed of three parallel MLPs, each tasked with regressing one of the 3 DoF relative pose values. The model exhibited positive results on the well-known KITTI vision benchmark suite, surpassing DeepVO by up to 65 % in t_{rel} and 73 % in r_{rel}. Additionally, the proposed model achieved performance levels comparable to state-of-the-art geometric approaches, such as CT-ICP and ORB-SLAM3, particularly in terms of translation RPE, with a value of 0.06 m in many training sequences and the tested sequence 10. This work also demonstrated the capability of deep neural networks to achieve robust solutions in challenging scenarios simulated by introducing noise, such as blur, Gaussian noise, and brightness and contrast adjustments to image data, from which optical flow was estimated, also applying Gaussian noise to depth projections. The results showcased performance levels comparable to the original data, with a maximum discrepancy of 0.01 m and 0.03° in translation and rotation RPE, respectively.

In conclusion, the proposed end-to-end multimodal intermediate fusion network has demonstrated its ability to tackle the egomotion estimation problem, significantly surpassing the DeepVO methodology. The network exhibited robustness to environmental constraints, handling heterogeneous sensor data and maintaining comparable performance even under degraded data conditions, underscoring its comprehensive benefits and advanced capabilities in egomotion estimation.

Despite the performance improvement over other learning-based approaches, results are still far from some state-of-the-art geometric approaches studied in this work. However, there is significant potential for further exploration and enhancements to close the gap with these well-established methodologies. Such exploration would benefit from DL models' capacity to generalise and increase robustness. Therefore, future work should be focused on the following:

- **Train the network on larger, more challenging autonomous driving datasets:** Focus on scenarios including high-speed scenes, numerous moving obstacles, different lighting conditions, and other complexities. Additionally, introduce noise into the training data to better simulate common data degradation, enhancing the model's robustness.
- **Explore the use of additional sensor modalities:** Incorporate sensors such as Inertial Measurement Units (IMUs), which do not require an external reference to estimate the agent's position. Their small size and low power consumption make them ideal for resource-constrained systems.
- **Fine-tune the current architecture:** Specifically, improve feature extraction from optical flow and depth by experimenting different architectures to optimise performance.
- **Integrate the current architecture with models that better capture sequential information:** Currently, temporal information is tied to motion estimates from optical flow. Consider incorporating models like Transformers or Recurrent Neural Networks (RNNs) capable of preserving context information and better address the sequential dynamics inherent to visual odometry.

References

[1] World Health Organization. Global status report on road safety 2023. Technical report, World Health Organization, 2023.

[2] A. Chand, S. Jayesh, and A.B. Bhasi. Road traffic accidents: An overview of data sources, analysis techniques and contributing factors. *Materials Today: Proceedings*, 47:5135–5141, 2021.

[3] K. Bucsuházy, E. Matuchová, R. Zůvala, P. Moravcová, M. Kostíková, and R. Mikulec. Human factors contributing to the road traffic accident occurrence. *Transportation Research Procedia*, 45:555–561, 2020.

[4] S. Chen, M. Kuhn, K. Prettner, and D. Bloom. The global macroeconomic burden of road injuries: estimates and projections for 166 countries. *The Lancet Planetary Health*, 3:e390–e398, 2019.

[5] A. Hussain, F. Akhtar, Z. H. Khand, A. Rajput, and Z. Shaukat. Complexity and limitations of gnss signal reception in highly obstructed enviroments. *Engineering, Technology & Applied Science Research*, 11:6864–6868, 2021.

[6] M. O. A. Aqel, M. H. Marhaban, M. I. Saripan, and N. Bt. Ismail. Review of visual odometry: types, approaches, challenges, and applications. *SpringerPlus*, 5, 2016.

[7] M. Kutila, P. Pyykönen, W. Ritter, O. Sawade, and B. Schäufele. Automotive lidar sensor development scenarios for harsh weather conditions. In *2016 IEEE 19th International Conference on Intelligent Transportation Systems*, pages 265–270, 2016.

[8] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv:2010.11929v2*, 2021.

[9] Z. Zhu, K. Lin, A. K. Jain, and J. Zhou. Transfer learning in deep reinforcement learning: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45:13344–13362, 2023.

[10] K. Wang, S. Ma, J. Chen, F. Ren, and J. Lu. Approaches, challenges, and applications for deep visual odometry: Toward complicated and emerging areas. *IEEE Transactions on Cognitive and Developmental Systems*, 14:35–49, 2022.

[11] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. In *2016 IEEE Conference on Computer Vision and Pattern Recognition*, pages 779–788, 2016.

[12] 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*, pages 3354–3361, 2012.

[13] 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*, 7:97466–97486, 2019.

[14] P. Leite and A. M. Pinto. Exploiting motion perception in depth estimation through a lightweight convolutional neural network. *IEEE Access*, 9:76056–76068, 2021.

[15] H. P. Moravec. *Obstacle avoidance and navigation in the real world by a seeing robot rover*. PhD thesis, Stanford University, Stanford, CA, 1980.

[16] D. Nister, O. Naroditsky, and J. Bergen. Visual odometry. In *2004 IEEE Conference on Computer Vision and Pattern Recognition*, pages I–I, 2004.

[17] S. Lacroix, A. Mallet, R. Chatila, and L. Gallo. Rover self localization in planetary-like environments. *Artificial Intelligence, Robotics and Automation in Space*, 440:433, 1999.

[18] D. Scaramuzza and F. Fraundorfer. Visual odometry: Part i: The first 30 years and fundamentals. *IEEE Robotics & Automation Magazine*, 18:80–92, 2011.

[19] Y. Alkendi, L. Seneviratne, and Y. Zweiri. State of the art in vision-based localization techniques for autonomous navigation systems. *IEEE Access*, 9:76847–76874, 2021.

[20] A. I. Comport, E. Malis, and P. Rives. Accurate quadrifocal tracking for robust 3d visual odometry. In *2007 IEEE International Conference on Robotics and Automation*, pages 40–45, 2007.

[21] D. Nister. An efficient solution to the five-point relative pose problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26:756–770, 2004.

[22] L. Kneip, D. Scaramuzza, and R. Siegwart. A novel parametrization of the perspective-three-point problem for a direct computation of absolute camera position and orientation. In *2011 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2969–2976, 2011.

[23] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the Association for Computing Machinery*, 24:381–395, 1981.

[24] L.-P. Morency and R. Gupta. Robust real-time egomotion from stereo images. In *2003 IEEE International Conference on Image Processing*, volume 2, pages II–719, 2003.

[25] F. Fraundorfer and D. Scaramuzza. Visual odometry: Part ii: Matching, robustness, optimization, and applications. *IEEE Robotics & Automation Magazine*, 19:78–90, 2012.

[26] D. Nister. Preemptive ransac for live structure and motion estimation. In *Ninth IEEE International Conference on Computer Vision*, pages 199–206 vol.1, 2003.

[27] C.G. Harris and J.M. Pike. 3d positional integration from image sequences. *Image and Vision Computing*, 6(2):87–90, 1988.

[28] J. Shi and T. Good features to track. In *1994 IEEE Conference on Computer Vision and Pattern Recognition*, pages 593–600, 1994.

[29] E. Rosten and T. Drummond. Machine learning for high-speed corner detection. In *Computer Vision – ECCV 2006*, pages 430–443, 2006.

[30] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60:91–110, 2004.

[31] H. Bay, T. Tuytelaars, and L. Van Gool. Surf: Speeded up robust features. In *Computer Vision – ECCV 2006*, pages 404–417, 2006.

[32] M. Calonder, V. Lepetit, C. Strecha, and P. Fua. Brief: Binary robust independent elementary features. In *Computer Vision – ECCV 2010*, pages 778–792, 2010.

[33] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski. Orb: An efficient alternative to sift or surf. In *2011 International Conference on Computer Vision*, pages 2564–2571, 2011.

[34] S. Leutenegger, M. Chli, and R. Y. Siegwart. Brisk: Binary robust invariant scalable keypoints. In *2011 International Conference on Computer Vision*, pages 2548–2555, 2011.

[35] E. Salahat and M. Qasaimeh. Recent advances in features extraction and description algorithms: A comprehensive survey. *arXiv:1703.06376*, 2017.

[36] H.-J. Chien, C.-C. Chuang, C.-Y. Chen, and R. Klette. When to use what feature? sift, surf, orb, or a-kaze features for monocular visual odometry. *2016 International Conference on Image and Vision Computing New Zealand*, pages 1–6, 2016.

[37] P. F. Alcantarilla, J. Nuevo, and A. Bartoli. Fast explicit diffusion for accelerated features in nonlinear scale spaces. In *British Machine Vision Conference*, 2013.

[38] F. Labrosse. The visual compass: Performance and limitations of an appearance-based method. *Journal of Field Robotics*, 23:913–941, 2006.

[39] N. Nourani-Vatani, J. Roberts, and M. V. Srinivasan. Practical visual odometry for car-like vehicles. In *2009 IEEE International Conference on Robotics and Automation*, pages 3551–3557, 2009.

[40] Y. Yu, C. Pradalier, and G. Zong. Appearance-based monocular visual odometry for ground vehicles. In *2011 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, pages 862–867, 2011.

[41] M. Aqel, M. H. Marhaban, M. I. Saripan, and N. Ismail. Adaptive-search template matching technique based on vehicle acceleration for monocular visual odometry system: Adaptive-search template matching for monocular visual odometry. *IEEJ Transactions on Electrical and Electronic Engineering*, 11, 2016.

[42] D. Caruso, J. Engel, and D. Cremers. Large-scale direct slam for omnidirectional cameras. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 141–148, 2015.

[43] J. Engel, V. Koltun, and D. Cremers. Direct sparse odometry. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40:611–625, 2018.

[44] A. M. Pinto, M. Correia, A. Moreira, and P. Costa. Unsupervised flow-based motion analysis for an autonomous moving system. *Image and Vision Computing*, 32:391–404, 2014.

[45] A. M. Pinto, A. Moreira, M. Correia, and P. Costa. A flow-based motion perception technique for an autonomous robot system. *Journal of Intelligent & Robotic Systems*, 2014.

[46] A. M. Pinto, P. Costa, M. Correia, A. Matos, and A. Moreira. Visual motion perception for mobile robots through dense optical flow fields. *Robotics and Autonomous Systems*, 87:1–14, 2017.

[47] S. Poddar, R. Kottath, and V. Karar. *Motion Estimation Made Easy: Evolution and Trends in Visual Odometry*, pages 305–331. Springer International Publishing, 2019.

[48] D. Scaramuzza and R. Siegwart. Appearance-guided monocular omnidirectional visual odometry for outdoor ground vehicles. *IEEE Transactions on Robotics*, 24:1015–1026, 2008.

[49] E. Olson, J. Leonard, and S. Teller. Fast iterative alignment of pose graphs with poor initial estimates. In *2006 IEEE International Conference on Robotics and Automation*, pages 2262–2269, 2006.

[50] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon. Bundle adjustment — a modern synthesis. In *Vision Algorithms: Theory and Practice*, pages 298–372, 2000.

[51] J. Jeong, Y. Cho, Y.-S. Shin, H. Roh, and A. Kim. Complex urban dataset with multi-level sensors from highly diverse urban environments. *The International Journal of Robotics Research*, 38:642–657, 2019.

[52] A. R. Gaspar, A. Nunes, A. M. Pinto, and A. Matos. Urban@cras dataset: Benchmarking of visual odometry and slam techniques. *Robotics and Autonomous Systems*, 109:59–67, 2018.

[53] R. Mur-Artal and J. D. Tardós. Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras. *IEEE Transactions on Robotics*, 33:1255–1262, 2017.

[54] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardós. Orb-slam: A versatile and accurate monocular slam system. *IEEE Transactions on Robotics*, 31:1147–1163, 2015.

[55] I. Cvišić and I. Petrović. Stereo odometry based on careful feature selection and tracking. In *2015 European Conference on Mobile Robots*, pages 1–6, 2015.

[56] I. Cvišić, J. Ćesić, I. Marković, and I. Petrović. Soft-slam: Computationally efficient stereo visual simultaneous localization and mapping for autonomous unmanned aerial vehicles. *Journal of Field Robotics*, 35, 2017.

[57] I. Cvišić, I. Marković, and I. Petrović. Soft2: Stereo visual odometry for road vehicles based on a point-to-epipolar-line metric. *IEEE Transactions on Robotics*, 39:273–288, 2023.

[58] J. Engel, T. Schöps, and D. Cremers. Lsd-slam: Large-scale direct monocular slam. In *Computer Vision – ECCV 2014*, pages 834–849, 2014.

[59] I. Cvišić, I. Marković, and I. Petrović. Recalibrating the kitti dataset camera setup for improved odometry accuracy. In *2021 European Conference on Mobile Robots*, pages 1–6, 2021.

[60] I. Cvišić, I. Marković, and I. Petrović. Enhanced calibration of camera setups for high-performance visual odometry. *Robotics and Autonomous Systems*, 155:104189, 2022.

[61] A. Geiger, J. Ziegler, and C. Stiller. Stereoscan: Dense 3d reconstruction in real-time. In *2011 IEEE Intelligent Vehicles Symposium (IV)*, pages 963–968, 2011.

[62] N. Yang, R. Wang, J. Stückler, and D. Cremers. Deep virtual stereo odometry: Leveraging deep depth prediction for monocular direct sparse odometry. In *Computer Vision – ECCV 2018*, page 835–852, 2018.

[63] N. Yang, L. Stumberg, R. Wang, and D. Cremers. D3vo: Deep depth, deep pose and deep uncertainty for monocular visual odometry. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1278–1289, 2020.

[64] 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*, pages 2043–2050, 2017.

[65] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. In *31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6000–6010, 2017.

[66] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, 2019.

[67] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei. Language models are few-shot learners. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, 2020.

[68] L. Dong, S. Xu, and B. Xu. Speech-transformer: A no-recurrence sequence-to-sequence model for speech recognition. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 5884–5888, 2018.

[69] I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. In *27th International Conference on Neural Information Processing Systems*, volume 2 of *NIPS'14*, page 3104–3112, 2014.

[70] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jegou. Training data-efficient image transformers & distillation through attention. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139, pages 10347–10357, 2021.

[71] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko. End-to-end object detection with transformers. In *Computer Vision – ECCV 2020*, pages 213–229, 2020.

[72] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai. Deformable DETR: Deformable transformers for end-to-end object detection. In *International Conference on Learning Representations*, 2021.

[73] L. Ye, M. Rochan, Z. Liu, and Y. Wang. Cross-modal self-attention network for referring image segmentation. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10494–10503, 2019.

[74] N. Park and S. Kim. How do vision transformers work? In *International Conference on Learning Representations*, 2022.

[75] K. Islam. Recent advances in vision transformer: A survey and outlook of recent work. *arXiv:2203.01536v5*, 2022.

[76] J. B. Rosário. Ego motion from video data in autonomous vehicles. M.s. thesis, FEUP, UP, Porto, Portugal, 2023.

[77] N. Jonnavithula, Y. Lyu, and Z. Zhang. Lidar odometry methodologies for autonomous driving: A survey. *arXiv:2109.06120v1*, 2021.

[78] D. Lee, M. Jung, W. Yang, and A. Kim. Lidar odometry survey: Recent advancements and remaining challenges. *Intelligent Service Robotics*, 17:95–118, 2024.

[79] P. Biber and W. Strasser. The normal distributions transform: a new approach to laser scan matching. In *2003 IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 3, pages 2743–2748, 2003.

[80] P. J. Besl and N. D. McKay. Method for registration of 3-D shapes. In *Sensor Fusion IV: Control Paradigms and Data Structures*, volume 1611, pages 586–606, 1992.

[81] X. Zheng and J. Zhu. Efficient lidar odometry for autonomous driving. *IEEE Robotics and Automation Letters*, 6:8458–8465, 2021.

[82] P. Dellenbach, J.-E. Deschaud, B. Jacquet, and F. Goulette. Ct-icp: Real-time elastic lidar odometry with loop closure. In *2022 International Conference on Robotics and Automation*, pages 5580–5586, 2022.

[83] I. Vizzo, T. Guadagnino, B. Mersch, L. Wiesmann, J. Behley, and C. Stachniss. Kiss-icp: In defense of point-to-point icp – simple, accurate, and robust registration if done the right way. *IEEE Robotics and Automation Letters*, 8:1029–1036, 2023.

[84] J. Zhang and S. Singh. Loam: Lidar odometry and mapping in real-time. In *Robotics: Science and systems*, volume 2, pages 1–9, 2014.

[85] H. Wang, C. Wang, C.-L. Chen, and L. Xie. F-loam : Fast lidar odometry and mapping. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4390–4396, 2021.

[86] X. Zheng and J. Zhu. Traj-lo: In defense of lidar-only odometry using an effective continuous-time trajectory. *IEEE Robotics and Automation Letters*, 9:1961–1968, 2024.

[87] Q. Li, S. Chen, C. Wang, X. Li, C. Wen, M. Cheng, and J. Li. Lo-net: Deep real-time lidar odometry. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8465–8474, 2019.

[88] Y. Cho, G. Kim, and A. Kim. Unsupervised geometry-aware deep lidar odometry. In *2020 IEEE International Conference on Robotics and Automation*, pages 2145–2152, 2020.

[89] D. Yin, Q. Zhang, J. Liu, X. Liang, Y. Wang, J. Maanpää, H. Ma, J. Hyyppä, and R. Chen. Cae-lo: Lidar odometry leveraging fully unsupervised convolutional auto-encoder for interest point detection and feature description. *arXiv:2001.01354v3*, 2020.

[90] X. Chen, A. Milioto, E. Palazzolo, P. Giguère, J. Behley, and C. Stachniss. Suma++: Efficient lidar-based semantic slam. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4530–4537, 2019.

[91] J. Behley and C. Stachniss. Efficient surfel-based slam using 3d laser range data in urban environments. In *Robotics: Science and Systems*, volume 2018, page 59, 2018.

[92] A. Milioto, I. Vizzo, J. Behley, and C. Stachniss. Rangenet ++: Fast and accurate lidar semantic segmentation. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4213–4220, 2019.

[93] J. Liu, G. Wang, C. Jiang, Z. Liu, and H. Wang. Translo: A window-based masked point transformer framework for large-scale lidar odometry. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37:1683–1691, 2023.

[94] D. Lee, H. Nam, and D. H. Shim. Eliot: End-to-end lidar odometry using transformer framework. *arXiv:2307.11998v4*, 2023.

[95] D. J. Yeong, G. Velasco-Hernandez, J. Barry, and J. Walsh. Sensor and sensor fusion technology in autonomous vehicles: A review. *Sensors*, 21, 2021.

[96] 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*, 10:72182–72205, 2022.

[97] J. Zhang and S. Singh. Visual-lidar odometry and mapping: low-drift, robust, and fast. In *2015 IEEE International Conference on Robotics and Automation*, pages 2174–2181, 2015.

[98] J. Graeter, A. Wilczynski, and M. Lauer. Limo: Lidar-monocular visual odometry. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 7872–7879, 2018.

[99] C. Campos, R. Elvira, J. J. Rodríguez, J. Montiel, and J. D. Tardós. Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam. *IEEE Transactions on Robotics*, 37:1874–1890, 2021.

[100] R. Mur-Artal and J. D. Tardós. Visual-inertial monocular slam with map reuse. *IEEE Robotics and Automation Letters*, 2:796–803, 2017.

[101] D. Schubert, T. Goll, N. Demmel, V. Usenko, J. Stückler, and D. Cremers. The tum vi benchmark for evaluating visual-inertial odometry. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1680–1687, 2018.

[102] M. Burri, J. Nikolic, P. Gohl, T. Schneider, J. Rehder, S. Omari, M. Achtelik, and R. Siegwart. The euroc micro aerial vehicle datasets. *The International Journal of Robotics Research*, 35, 2016.

[103] C.-C. Chou and C.-F. Chou. Efficient and accurate tightly-coupled visual-lidar slam. *IEEE Transactions on Intelligent Transportation Systems*, 23:14509–14523, 2022.

[104] Z. Yuan, Q. Wang, K. Cheng, T. Hao, and X. Yang. Sdv-loam: Semi-direct visual–lidar odometry and mapping. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45:11203–11220, 2023.

[105] Q. Tang, J. Liang, and F. Zhu. A comparative review on multi-modal sensors fusion based on deep learning. *Signal Processing*, 213:109165, 2023.

[106] L. Sun, G. Ding, Y. Qiu, Y. Yoshiyasu, and F. Kanehiro. Transfusionodom: Transformer-based lidar-inertial fusion odometry estimation. *IEEE Sensors Journal*, 23:22064–22079, 2023.

[107] C. Chen, S. Rosa, Y. Miao, C. X. Lu, W. Wu, A. Markham, and N. Trigoni. Selective sensor fusion for neural visual-inertial odometry. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10534–10543, 2019.

[108] L. Sun, G. Ding, Y. Yoshiyasu, and F. Kanehiro. Certainodom: Uncertainty weighted multi-task learning model for lidar odometry estimation. In *2022 IEEE International Conference on Robotics and Biomimetics*, pages 121–128, 2022.

[109] L. Agostinho, D. Pereira, A. Hiolle, and A. M. Pinto. Tefu-net: A time-aware late fusion architecture for robust multi-modal ego-motion estimation. *Robotics and Autonomous Systems*, 177:104700, 2024.

[110] M. Kissai. *Fundamentals on Vehicle and Tyre Modelling*, pages 1–60. Springer International Publishing, 2022.

[111] Z. Teed and J. Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *Computer Vision – ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II*, page 402–419, 2020.

[112] D. Eigen, C. Puhrsch, and R. Fergus. Depth map prediction from a single image using a multi-scale deep network. In *Advances in Neural Information Processing Systems*, volume 27, 2014.

[113] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[114] N. Ketkar. *Stochastic Gradient Descent*, pages 113–132. Apress, 2017.

[115] I. Loshchilov and F. Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

[116] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv:1412.6980v9*, 2017.