Neural Networks 150 (2022) 119–136

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

SelfVIO: Self-supervised deep monocular Visual–Inertial Odometry and depth estimation

Yasin Almalioglu [a,*], Mehmet Turan [b], Muhamad Risqi U. Saputra [a], Pedro P.B. de Gusmão [a], Andrew Markham [a], Niki Trigoni [a]

[a] Computer Science Department, The University of Oxford, UK
[b] Institute of Biomedical Engineering, Bogazici University, Turkey

ARTICLE INFO

Article history:
Received 5 January 2022
Received in revised form 1 March 2022
Accepted 3 March 2022
Available online 10 March 2022

Keywords:
Self-supervised learning
Geometry reconstruction
Machine perception
Generative adversarial networks
Deep sensor fusion
visual–inertial odometry

ABSTRACT

In the last decade, numerous supervised deep learning approaches have been proposed for visual–inertial odometry (VIO) and depth map estimation, which require large amounts of labelled data. To overcome the data limitation, self-supervised learning has emerged as a promising alternative that exploits constraints such as geometric and photometric consistency in the scene. In this study, we present a novel self-supervised deep learning-based VIO and depth map recovery approach (SelfVIO) using adversarial training and self-adaptive visual–inertial sensor fusion. SelfVIO learns the joint estimation of 6 degrees-of-freedom (6-DoF) ego-motion and a depth map of the scene from unlabelled monocular RGB image sequences and inertial measurement unit (IMU) readings. The proposed approach is able to perform VIO without requiring IMU intrinsic parameters and/or extrinsic calibration between IMU and the camera. We provide comprehensive quantitative and qualitative evaluations of the proposed framework and compare its performance with state-of-the-art VIO, VO, and visual simultaneous localization and mapping (VSLAM) approaches on the KITTI, EuRoC and Cityscapes datasets. Detailed comparisons prove that SelfVIO outperforms state-of-the-art VIO approaches in terms of pose estimation and depth recovery, making it a promising approach among existing methods in the literature.

1. Introduction

Estimation of ego-motion and scene geometry is one of the key challenges in many engineering fields such as robotics, autonomous driving, and virtual reality. In the last few decades, visual odometry (VO) systems have attracted a substantial amount of attention due to low-cost hardware setups and rich visual representation (Fraundorfer & Scaramuzza, 2012; Rajan & Saffiotti, 2017). However, monocular VO is confronted with numerous challenges such as scale ambiguity, the need for hand-crafted mathematical features (e.g., ORB, BRISK), strict parameter tuning and image blur caused by abrupt camera motion, which might corrupt VO algorithms if deployed in low-textured areas and variable ambient lighting conditions (Engel, Schöps, & Cremers, 2014; Mur-Artal & Tardós, 2017a). For such cases, visual–inertial odometry (VIO) systems increase the robustness of VO systems, incorporating information from an inertial measurement unit (IMU) to improve motion tracking performance (Li, Besada, Bernardos, Tarrío, & Casar, 2017; Luo et al., 2021; Mur-Artal & Tardós, 2017b; Qin, Li, & Shen, 2018).

Supervised deep learning methods have achieved state-of-the-art results on various computer vision problems using large amounts of labelled data (He, Gkioxari, Dollár, & Girshick, 2020; Krizhevsky, Sutskever, & Hinton, 2017; Long, Shelhamer, & Darrell, 2015). Moreover, supervised deep VIO and depth recovery techniques have shown promising performance in challenging environments and successfully alleviate issues such as scale drift, need for feature extraction and parameter fine-tuning (Clark, Wang, Wen, Markham, & Trigoni, 2017; Gao, Liu, & Ju, 2020; Turan et al., 2018, 2019; Wang, Clark, Wen, & Trigoni, 2017). Although learning-based methods use raw input data similar to the dense VO and VIO methods, they also extract features related to odometry, depth and optical flow without explicit mathematical modelling (Clark et al., 2017; Engel et al., 2014; İncetan et al., 2021; Mur-Artal, Montiel, & Tardós, 2015; Mur-Artal & Tardós, 2017a; Ozyoruk et al., 2021). Most existing deep learning approaches in the literature treat VIO and depth recovery as a supervised learning problem, where they have colour input images, corresponding target depth values and relative

* Correspondence to: Linacre College, St.Cross Rd, Oxford OX1 3JA, UK.
E-mail addresses: yasin.almalioglu@cs.ox.ac.uk (Y. Almalioglu), mehmet.turan@boun.edu.tr (M. Turan), muhamad.saputra@cs.ox.ac.uk (M.R.U. Saputra), pedro.gusmao@cs.ox.ac.uk (P.P.B. de Gusmão), andrew.markham@cs.ox.ac.uk (A. Markham), niki.trigoni@cs.ox.ac.uk (N. Trigoni).

https://doi.org/10.1016/j.neunet.2022.03.005

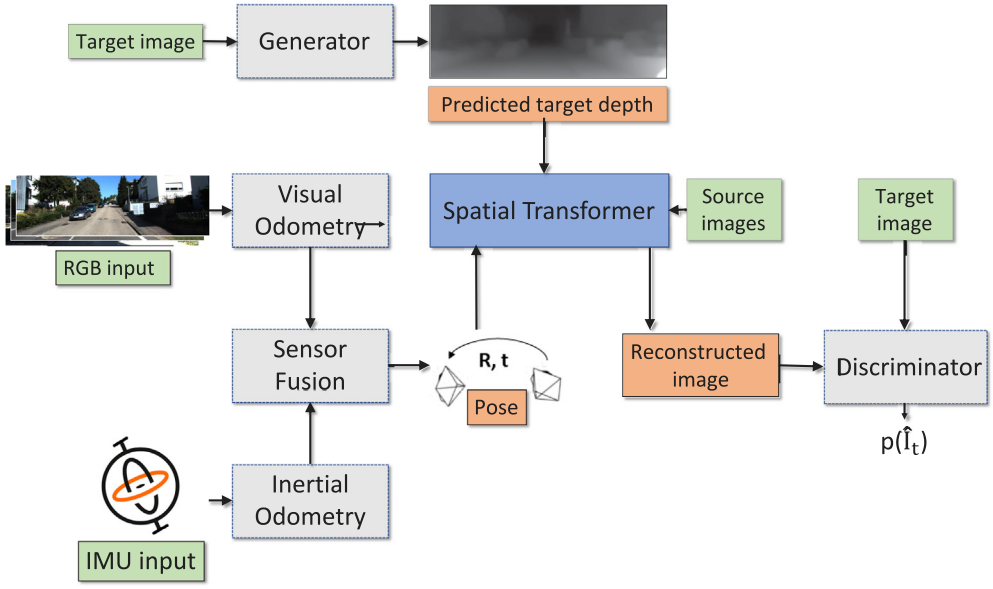


Fig. 1. Architecture overview. The proposed unsupervised deep learning approach consists of depth generation, visual odometry, inertial odometry, visual–inertial fusion, spatial transformer, and target discrimination modules. Unlabelled image sequences and raw IMU measurements are provided as inputs to the network. The method estimates relative translation and rotation between consecutive frames parametrized as 6-DoF motion and a depth image as a disparity map for a given view. The green and orange boxes represent inputs and intermediate outputs of the system, respectively. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

transformation of images at training time. VIO as a regression problem in supervised deep learning exploits the capability of convolutional neural networks (CNNs) and recurrent neural networks (RNNs) to estimate camera motion, calculate optical flow, and extract efficient feature representations from raw RGB and IMU input (Clark et al., 2017; Muller & Savakis, 2017; Turan, Almalioglu, Araujo et al., 2018; Wang et al., 2017). However, for many vision-aided localization and navigation problems requiring dense, continuous-valued outputs (e.g. visual–inertial odometry (VIO) and depth map reconstruction), it is either impractical or expensive to acquire ground truth data for a large variety of scenes (Geiger, Lenz, & Urtasun, 2012). Firstly, a state estimator uses timestamps for each camera image and IMU sample to enable the processing of the sensor measurements, which are typically taken either from the sensor itself, or from the operating system of the computer receiving the data. However, a delay (different for each sensor) exists between the actual sampling of a measurement and its timestamp due to the time needed for data transfer, sensor latency, and OS overhead. Furthermore, even if hardware time synchronization is used for timestamping (e.g., different clocks on sensors), these clocks may suffer from clock skew, resulting in an unknown time offset that typically exists between the timestamps of the camera and the IMU (Qin & Shen, 2018). Secondly, even when ground truth depth data is available, it can be imperfect and cause distinct prediction artefacts. For example, systems employing rotating LIDAR scanners suffer from the need for tight temporal alignment between laser scans and corresponding camera images even if the camera and LIDAR are carefully synchronized (Asvadi, Garrote, Premebida, Peixoto, & J. Nunes, 2018). In addition, structured light depth sensors – and to a lesser extent, LIDAR and time-of-flight sensors – suffer from noise and structural artefacts, especially in the presence of reflective, transparent, or dark surfaces. Last, there is usually an offset between the depth sensor and the camera, which causes shifts in the point cloud projection onto the camera viewpoint. These problems may lead to degraded performance and even failure for learning-based models trained on such data (Mahjourian, Wicke, & Angelova, 2018; Turan et al., 2018).

In recent years, unsupervised deep learning approaches have emerged to address the problem of limited training data (Artetxe, Labaka, & Agirre, 2018; Lundquist & Schön, 2011; Meister, Hur, & Roth, 2018; Yu, Harley, & Derpanis, 2016). The key idea of these methods is to define pretext training tasks to capture and use the dependencies with robustness and smoothness among different dimensions of the input data, e.g., the spatial, temporal, or channel. Self-supervised approaches can also be used to fine-tune the pre-trained deep models for downstream tasks, or as an auxiliary training task that contributes to the performance of main tasks (Xie, Xu, Zhang, Wang, & Ji, 2022). As an alternative, these approaches instead treat depth estimation as an image reconstruction problem during training. The intuition here is that, given a sequence of monocular images, we can learn a function that is able to reconstruct a target image from source images, exploiting the 3D geometry of the scene. Learning a mapping from pixels to depth and camera motion without the ground truth is challenging because each of these problems is highly ambiguous. To address this issue, recent studies imposed additional constraints and exploited the geometric relations between ego-motion and the depth map (Mahjourian et al., 2018; Zhou, Brown, Snavely, & Lowe, 2017). Recently, the optical flow has been widely studied and used as a self-supervisory signal for learning an unsupervised ego-motion system, but it has an aperture problem due to the missing structure in local parts of the single camera (Fortun, Bouthemy, & Kervrann, 2015). However, most unsupervised methods learn only from photometric and temporal consistency between consecutive frames in monocular videos, which are prone to overly smoothed depth map estimations.

To overcome these limitations, we propose a self-supervised VIO and depth map reconstruction system based on adversarial training and attentive sensor fusion (see Fig. 1), extending our GANVO work (Almalioglu, Saputra, d Gusmão, Markham, & Trigoni, 2019). GANVO is a generative unsupervised learning framework that predicts 6-DoF pose camera motion and a monocular depth map of the scene from unlabelled RGB image sequences, using deep convolutional Generative Adversarial Networks (GANs). Instead of ground truth pose and depth values,

GANVO creates a supervisory signal by warping view sequences and assigning the re-projection minimization to the objective loss function that is adopted in multi-view pose estimation and single-view depth generation network. In this work, we introduce a novel sensor fusion technique to incorporate motion information captured by an interoceptive and mostly environment-agnostic raw inertial data into loosely synchronized visual data captured by an exteroceptive RGB camera sensor. Furthermore, we conduct experiments on the publicly available EuRoC MAV dataset (Burri et al., 2016) to measure the robustness of the fusion system against miscalibration. Additionally, we separate the effects of the VO module from the pose estimates extracted from IMU measurements to test the effectiveness of each module. Moreover, we perform ablation studies to compare the performance of convolutional and recurrent networks. In addition to the results presented in Almalioglu et al. (2019), here we thoroughly evaluate the benefit of the adversarial generative approach. In summary, the main contributions of the approach are as
follows:

- To the best of our knowledge, this is the first self-supervised deep joint monocular VIO and depth reconstruction method in the literature;
- We propose a novel unsupervised sensor fusion technique for the camera and the IMU, which extracts and fuses motion features from raw IMU measurements and RGB camera images using convolutional and recurrent modules based on an attention mechanism;
- No strict temporal or spatial calibration between the camera and IMU is necessary for pose and depth estimation, contrary to traditional VO approaches.

Evaluations made on the KITTI (Geiger, Lenz, Stiller, & Urtasun, 2013), EuRoC (Burri et al., 2016) and Cityscapes (Cordts et al., 2016) datasets prove the effectiveness of SelfVIO. The organization of this paper is as follows. Previous work in this domain is discussed in Section 2. Section 3 describes the proposed unsupervised deep learning architecture and its mathematical background in detail. Section 4 describes the experimental setup and evaluation methods. Section 5 shows and discusses detailed quantitative and qualitative results with comprehensive comparisons to existing methods in the literature. Finally, Section 6 concludes the study with some interesting future directions.

2. Related work

In this section, we briefly outline the related works focused on VIO including traditional and learning-based methods.

2.1. Traditional methods

Traditional VIO solutions combine visual and inertial data in a single pose estimator and lead to more robust and higher accuracy compared to VO even in complex and dynamic environments. The fusion of camera images and IMU measurements is typically accomplished by filter-based or optimization-based approaches. Early works of filter-based approaches formulated visual–inertial fusion as a pure sensor fusion problem, which fuses vision as an independent 6-DoF sensor with inertial measurements in a filtering framework (called loosely-coupled) (Weiss, Achtelik, Lynen, Chli, & Siegwart, 2012; Yu et al., 2016). In a recent loosely-coupled method, Omari et al. proposed a filter-based direct stereo visual–inertial system, which fuses IMU with respect to the last keyframe. These loosely coupled approaches allow modular integration of visual odometry methods without modification. However, more recent works follow a tightly coupled approach to optimally exploit both sensor modalities, treating visual–inertial odometry as one integrated estimation problem. The multi-state constraint Kalman filter (MSCKF) (Mourikis & Roumeliotis, 2007) is a standard for filtering-based VIO approaches. It has a low computational complexity that is linear in the number of features used for ego-motion estimation. While MSCKF-based approaches are generally more robust compared to optimization-based approaches especially in large-scale real environments, they suffer from lower accuracy in comparison (as has been recently reported in Delmerico and Scaramuzza (2018)). Qin and Shen (2018) rigorously addressed online calibration for the first time based on MSCKF, unlike offline sensor to sensor spatial transformation and time offset calibration systems such as Furgale, Rehder, and Siegwart (2013). This online calibration method shows explicitly that the time offset is, in general, observable and provides sufficient theoretical conditions for the observability of time offset alone, while practical degenerate motions are not thoroughly examined (Yang, Geneva, Eckenhoff, & Huang, 2019). Li and Mourikis (2013) proved that the standard method of computing Jacobian matrices in filters inevitably causes inconsistencies and accuracy loss. For example, they showed that the yaw errors of the MSCKF lay outside the 3σ bounds, which indicates filter inconsistencies. They modified the MSCKF algorithm to ensure the correct observability properties without incurring additional computational costs. ROVIO (Bloesch, Burri, Omari, Hutter, & Siegwart, 2017) is another filtering-based VIO algorithm for monocular cameras that utilizes the intensity errors in the update step of an extended Kalman filter (EKF) to fuse visual and inertial data. ROVIO uses a robocentric approach that estimates 3D landmark positions relative to the current camera pose.

On the other hand, optimization-based approaches operate based on an energy-function representation in a non-linear optimization framework. While the complementary nature of filter-based and optimization-based approaches has long been investigated (Eustice, Singh, & Leonard, 2006), energy-based representations (Jones & Soatto, 2011; Usenko, Engel, Stückler, & Cremers, 2016) allow easy and adaptive re-linearization of energy terms, which avoids systematic error integration caused by linearization. OKVIS (Leutenegger, Lynen, Bosse, Siegwart, & Furgale, 2015) is a widely used, optimization-based visual–inertial SLAM approach for monocular and stereo cameras. OKVIS uses a nonlinear batch optimization on saved keyframes consisting of an image and an estimated camera pose. It updates a local map of landmarks to estimate camera motion without any loop closure constraint. To avoid repeated constraints caused by the parameterization of relative motion integration, Lupton and Sukkarieh (2012) proposed IMU pre-integration to reduce computation, changing the IMU data between two frames by pre-integrating the motion constraints. Forster, Carlone, Dellaert, and Scaramuzza (2015) further improve this principle by applying it to the visual–inertial SLAM framework to reduce bias. Besides, systems that fused IMU data into the classic visual odometry also attracted widespread attention. Usenko et al. (2016) proposed a stereo direct VIO to combine IMU with stereo LSD-SLAM (Engel, Stückler, & Cremers, 2015). They recovered the full state containing camera pose, translational velocity, and IMU biases of all frames, using a joint optimization method. Concha, Loianno, Kumar, and Civera (2016) devised the first direct real-time tightly-coupled VIO algorithm, but the initialization was not introduced. VINS-Mono (Qin et al., 2018) is a tightly coupled, nonlinear optimization-based method for monocular cameras. It uses pose graph optimization to enforce global consistency, which is constrained by a loop detection module. VINS-Mono features efficient IMU pre-integration with bias correction, automatic initialization of estimator, online extrinsic calibration, failure detection, and loop detection.

2.2. Learning-based methods

Eigen, Puhrsch, and Fergus (2014) proposed a two-scale deep network and showed that it was possible to produce dense pixel depth estimates, training on images, and the corresponding ground truth depth values. Unlike most other previous work in single view depth estimation, their model learns a representation directly from the raw pixel values, without any need for hand-crafted features or an initial over-segmentation. Several works followed the success of this approach using techniques such as the conditional random fields to improve the reconstruction accuracy (Li, Shen, Dai, van den Hengel, & He, 2015), incorporating strong scene priors for surface normal estimation (Wang, Fouhey, & Gupta, 2015), and the use of more robust loss functions (Laina, Rupprecht, Belagiannis, Tombari, & Navab, 2016). Again, like the most previous stereo methods, these approaches rely on existing high quality, pixel aligned, and dense ground truth depth maps at training time.

In recent years, several works adopt the classical GAN to estimate the depth from a single image (Aleotti, Tosi, Poggi, & Mattoccia, 2019; Kumar, Bhandarkar, & Prasad, 2018; Wu, Wu, Zhang, Wang, & Ju, 2019), following the success of GANs in many learning-based applications such as style transfer (Johnson, Alahi, & Fei-Fei, 2016), image-to-image translation (Isola, Zhu, Zhou, & Efros, 2017), image editing (Almalioglu et al., 2020; Zhu, Krähenbühl, Shechtman, & Efros, 2016) and cross-domain image generation (Bousmalis, Silberman, Dohan, Erhan, & Krishnan, 2017). Pilzer, Xu, Puscas, Ricci, and Sebe (2018) proposed a depth estimation model that employs the cycled generative networks to estimate depth from stereo pairs in an unsupervised manner. Vankadari, Kumar, Majumder, and Das (2019) proposed a PatchGAN based method to detect high frequency local structural defects in the reconstructed image. These works demonstrate the effectiveness of GANs in depth map estimation.

VINet (Clark et al., 2017) was the first end-to-end trainable visual–inertial deep network. However, VINet was trained in a supervised manner and thus required the ground truth pose differences for each exemplar in the training set. Recently, there have been several successful unsupervised depth estimation approaches, which use image warping as part of reconstruction loss to create a supervision signal similar to our network. Garg, Kumar, Carneiro, and Reid (2016), Godard, Aodha, and Brostow (2017) and Zhan et al. (2018) used such methods with stereo image pairs with known camera baselines and reconstruction loss for training. Thus, while technically unsupervised, stereo baseline effectively provides a known transformation between two images.

More recent works (Mahjourian et al., 2018; Ummenhofer et al., 2017; Yin & Shi, 2018; Zhou et al., 2017) have formulated odometry and depth estimation problems by coupling two or more problems together in an unsupervised learning framework. Zhou et al. (2017) introduced joint unsupervised learning of ego-motion and depth from multiple unlabelled RGB frames. They input a consecutive sequence of images and output a change in pose between the middle image of the sequence and every other image in the sequence, and the estimated depth of the middle image. Recent work (Wulff & Black, 2019) used a more explicit geometric loss to jointly learn depth and camera motion for rigid scenes with a semi-differentiable iterative closest point (ICP) module. These VO approaches estimate ego-motion only by the spatial information existing in several frames, which means temporal information within the frames is not fully utilized. As a result, the estimates are inaccurate and discontinuous.

UnDeepVO (Li, Wang, Long, & Gu, 2018) is another unsupervised depth and ego-motion estimation work. It differs from Zhou et al. (2017) in that it can estimate the camera trajectory on an absolute scale. However, unlike Zhou et al. (2017) and similar to Garg et al. (2016) and Godard et al. (2017), it uses stereo image pairs for training where the baseline between images is available and thus, UnDeepVO can only be trained on datasets where stereo image pairs are existent. Additionally, stereo images are recorded simultaneously, and the spatial transformation between paired images from stereo cameras are unobservable by an IMU. Thus, the network architecture of UnDeepVO cannot be extended to include motion estimates derived from inertial measurements. VIOLearner (Shamwell, Lindgren, Leung, & Nothwang, 2020) is a recent unsupervised learning-based approach to VIO using multiview RGB-depth (RGB-D) images, which extends the work of Shamwell, Leung, and Nothwang (2018). It uses a learned optimizer to minimize photometric loss for ego-motion estimation, which leverages the Jacobians of scaled image projection errors with respect to a spatial grid of pixel coordinates similar to Clark, Bloesch, Czarnowski, Leutenegger, and Davison (2018). Although no ground truth odometry data are needed, the depth input to the system provides external supervision to the network, which may not always be available.

One critical issue of these unsupervised works is the fact that they use auto encoder–decoder-based traditional depth estimators with a tendency to generate overly smooth images (Dosovitskiy & Brox, 2016). GANVO (Almalioglu et al., 2019) is the first unsupervised adversarial generative approach to jointly estimate multiview pose and monocular depth map. GANVO solves the smoothness problem in the reconstructed depth maps using GANs. Therefore, we apply GANs to provide sharper and more accurate depth maps, extending the work of Almalioglu et al. (2019). The second issue of the aforementioned unsupervised techniques is the fact that they solely employ CNNs that only analyse just-in-moment information to estimate camera pose (Almalioglu, Turan, Lu, Trigoni and Markham, 2021; Huang, Fu, He, Jiang, & Hao, 2021; Turan, Almalioglu, Araujo et al., 2018; Wang et al., 2017). We address this issue by employing a CNN–RNN architecture to capture temporal relations across frames. Furthermore, these existing VIO works use a direct fusion approach that concatenates all features extracted from different modalities, resulting in sub-optimal performance, as not all features are useful and necessary (Chen et al., 2019; Jiang et al., 2019). We introduce an attention mechanism to self-adaptively fuse the different modalities conditioned on the input data. We discuss our reason behind these design choices in the related sections.

3. Self-supervised monocular VIO and depth estimation architecture

Given unlabelled monocular RGB image sequences and raw IMU measurements, the proposed approach learns a function f that regresses 6-DoF camera motion and predicts the per-pixel scene depth. An overview of our SelfVIO architecture is depicted in Fig. 1. We stack the monocular RGB sequences consisting of a target view ($\mathbf{I}_t$) and source views ($\langle \mathbf{I}_{t-1}, \mathbf{I}_{t+1} \rangle$) to form an input batch for the multiview visual odometry module. The VO module consisting of convolutional layers regresses the relative 6-DoF pose values of the source views with respect to the target view. We form an IMU input tensor using raw linear acceleration and angular velocity values measured by an IMU between $t-1$ and $t+1$, which is processed in the inertial odometry module to estimate the relative motion of the source views. We fuse the 6-DoF pose values estimated by visual and inertial odometry modules in a self-adaptive fusion module, attentively selecting certain features that are significant for pose regression. In parallel, the target view ($\mathbf{I}_t$) is fed into the encoder module. The depth generator module estimates a depth map of the target view by inferring the disparities that warp the source views to the target. The spatial transformer module synthesizes the target image using the generated depth map and the nearby colour pixels in a source image

sampled at locations determined by a fused 3D Euclidean transformation. The geometric constraints that provide a supervision signal cause the neural network to synthesize a target image from multiple source images acquired from different camera poses. The view discriminator module learns to distinguish the difference between a fake (synthesized by the spatial transformer) and a real target image. In this way, each subnetwork targets a specific subtask and the complex scene geometry understanding goal is decomposed into smaller subgoals.

In the overall adversarial paradigm, a generator network is trained to produce output that cannot be distinguished from the original image by an adversarially optimized discriminator network. The objective of the generator is to trick the discriminator, i.e. to generate a depth map of the target view such that the discriminator cannot distinguish the reconstructed view from the original view. Unlike the typical use of GANs, the spatial transformer module maps the output image of the generator to the colour space of the target view and the discriminator classifies this reconstructed coloured view rather than the direct output of the generator. The proposed scheme enables us to predict the relative motion and depth map in an unsupervised manner, which is explained in the following sections in detail.

3.1. Depth estimation

The first part of the architecture is the depth generator network that synthesizes a single-view depth map by translating the target RGB frame. A defining feature of image-to-depth translation problems is that they map a high-resolution input tensor to a high-resolution output tensor, which differs in surface appearance. However, both images are renderings of the same underlying structure. Therefore, the structure in the RGB frame is roughly aligned with the structure in the depth map.

The depth generator network is based on a GAN design that learns the underlying generative model of the input image $p(\mathbf{I}_t)$. Unlike auto-encoder networks, we employ GANS to predict sharper and more accurate depth maps. Three subnetworks are involved in the adversarial depth generation process: an encoder network E, a generator network G, and a discriminator network D. The encoder E extracts a feature vector $\mathbf{z}$ from the input target image $\mathbf{I}_t$, i.e. $E(\mathbf{I}_t) = \mathbf{z}$. G maps the vector $\mathbf{z}$ to the depth image space which is used in the spatial transformer module to reconstruct the original target view. D classifies the reconstructed view as synthesized or real.

Many previous solutions (Li et al., 2018; Ranjan et al., 2019; Zhou et al., 2017) to the single-view depth estimation are based on an encoder–decoder network (Hinton & Salakhutdinov, 2006). Such a network passes the input through a series of layers that progressively downsample until a bottleneck layer and, then, the process is reversed by upsampling. All information flow passes through all the layers, including the bottleneck. For the image-to-depth translation problem, there is a great deal of low-level information shared between the input and output, and the network should make use of this information by directly sharing it across the layers. As an example, RGB image input and the depth map output share the location of prominent edges. To enable the generator to circumvent the bottleneck for such shared low-level information, we add skip connections similar to the general shape of a U-Net (Ronneberger, Fischer, & Brox, 2015). Specifically, these connections are placed between each layer i and layer $n-i$, where n is the total number of layers, which concatenate all channels at layer i with those at layer $n-i$.

3.2. Visual odometry

The VO module (see Fig. 2) is designed to take two concatenated source views and a target view along the colour channels as input and to output a visual feature vector $\mathbf{p}_V$ introduced by motion and temporal dynamics across frames. CNNs have improved data-driven models with both more robust feature extractors and deeper structures able to learn complex processes (Costante & Mancini, 2020; Kendall, Grimes, & Cipolla, 2015). The network is composed of 7 stride-2 convolutions followed by the adaptive fusion module. We decouple the convolution layer for translation and rotation using the shared weights as it has been shown to work better in separate branches as in Saputra et al. (2020). We also use a dropout (Srivastava, Hinton, Krizhevsky, Sutskever, & Salakhutdinov, 2014) between the convolution layers at the rate of 0.25 to help regularization. The last convolution layer gives a visual feature vector to encode geometrically meaningful features for movement estimation, which is used to define the 3D Euclidean transformation between target image $\mathbf{I}_t$ and source images $\mathbf{I}_{t-1}$ and $\mathbf{I}_{t+1}$.

3.3. Inertial odometry

SelfVIO takes raw IMU measurements in the following form:

$$\mathbf{M} = \begin{bmatrix} \alpha_{t-1} & \omega_{t-1} \\ \dots & \dots \\ \alpha_{t+1} & \omega_{t+1} \end{bmatrix} \in \mathbb{R}^{n\times 6},$$

where $\alpha \in \mathbb{R}^3$ is linear acceleration, $\omega \in \mathbb{R}^3$ is the angular velocity, and n is the number of IMU samples obtained between time $t-1$ and $t+1$ (no timestamp related to the IMU of the camera is passed to the network). The IMU module receives the same size of padded input in each time frame. The IMU processing module of SelfVIO uses two parallel branches consisting of 5 convolutional layers for the IMU angular velocity and linear acceleration (see Fig. 2 for more detail). Each branch on the IMU measurements has the following convolutional layers:

1. two layers: 64 single-stride filters with kernel size 3×5,
2. one layer: 128 filters of stride 2 with kernel size 3×5,
3. one layer: 256 filters of stride 2 with kernel size 3×5, and
4. one layer: 512 filters of stride 2 with kernel size 3×2.

The outputs of the final angular velocity and linear acceleration branches were flattened into 2×3 tensors using a convolutional layer with three filters of kernel size 1 and stride 1 before they are concatenated into a tensor $\mathbf{p}_M$. Thus, it learns to estimate 3D Euclidean transformation between times $t-1$ and $t+1$.

3.4. Self-adaptive visual–inertial fusion

In learning-based VIO, a standard method for fusion is the concatenation of feature vectors coming from different modalities, which may result in suboptimal performance, as not all features are equally reliable (Chen et al., 2019). For example, the fusion is plagued by the intrinsic noise distribution of each modality such as white random noise and sensor bias in IMU data. Moreover, many real-world applications suffer from poor calibration and synchronization between different modalities. To eliminate the effects of these factors, we employ an attention mechanism (Vaswani et al., 2017), which allows the network to automatically learn the best suitable feature combination given visual–inertial feature inputs.

The convolutional layers of the VO and IMU processing modules extract features from the input sequences and estimate egomotion, which is propagated to the self-adaptive fusion module. In our attention mechanism, we use a deterministic soft fusion

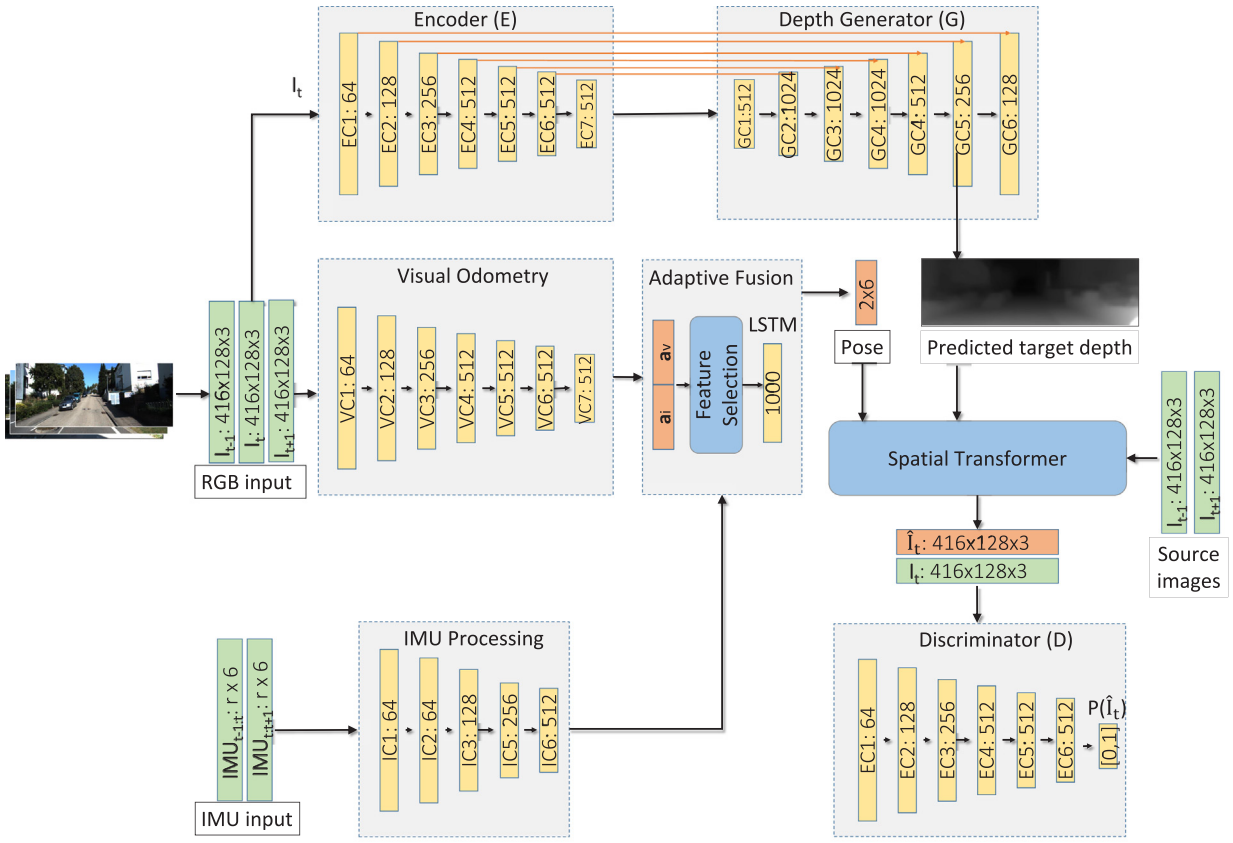


Fig. 2. The proposed architecture for pose estimation and depth map generation. The spatial dimensions of layers and output channels are proportional to the tensor shapes that flow through the network. Generator network G maps the feature vector generated by the encoder network E to the depth image space. In parallel, the visual odometry module extracts VO-related features through a convolutional network, while the inertial odometry module estimates inertial features related to ego-motion. The adaptive sensor fusion module fuses visual and inertial information, and estimates pose using a recurrent network that captures temporal relations among the input sequences. Pose results are collected after adaptive fusion operation, which has $6*(N-1)$ output channels for 6-DoF motion parameters, where N is the length of the input sequence. The spatial transformer module reconstructs the target view using the estimated depth map and pose values. The discriminator D maps the reconstructed RGB image to a likelihood of the target image, which determines whether it is the reconstructed or original target image. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

approach to attentively fuse features. The adaptive fusion module learns visual ($\mathbf{s}_v$) and inertial ($\mathbf{s}_i$) filters to reweight each feature by conditioning on all channels:

$$\mathbf{s}_v = \sigma(\mathbf{W}_v[\mathbf{a}_v, \mathbf{a}_i]) \tag{1}$$

$$\mathbf{s}_i = \sigma(\mathbf{W}_i[\mathbf{a}_v, \mathbf{a}_i]), \tag{2}$$

where $\sigma(x) = 1/(1+e^{-x})$ is the sigmoid function, $[\mathbf{a}_v, \mathbf{a}_i]$ is the concatenation of all channel features, and $\mathbf{W}_v$ and $\mathbf{W}_i$ are the weights for each modality. We multiply the visual and inertial features with these masks to weight the relative importance of the features:

$$\mathbf{W}_{fused} = [\mathbf{a}_v \odot \mathbf{s}_v, \mathbf{a}_i \odot \mathbf{s}_v], \tag{3}$$

where $\odot$ is the elementwise multiplication. The resulting feature matrix $\mathbf{W}_{fused}$ is fed into the RNN part (a two-layer bi-directional LSTM with 1000 hidden units). The LSTM takes the combined feature representation and its previous hidden states as input and models the dynamics and connections between a sequence of features. After the recurrent network, a fully connected layer regresses the fused pose, which maps the features to a 6-DoF pose vector. It outputs $6*(N-1)$ (N is the number of input views, i.e. 3) channels for 6-DoF pose values for translation and rotation parameters, representing the motion over a time window $t-1$ and $t+1$. The output pose vector defines the 3D Euclidean transformation between target image $\mathbf{I}_t$ and source images $\mathbf{I}_{t-1}$ and $\mathbf{I}_{t+1}$. The LSTM improves the sequential learning capacity of the network, resulting in more accurate pose estimation.

3.5. Spatial transformer

A sequence of 3 consecutive frames is given to the pose network as input. An input sequence is denoted by $\langle I_{t-1}, I_t, I_{t+1}\rangle$ where $t > 0$ is the time index, I_t is the target view, and the other frames are source views $I_s = \langle I_{t-1}, I_{t+1}\rangle$ that are used to render the target image according to the objective function:

$$\mathcal{L}_g = \sum_s \sum_p |I_t(p) - \hat{I}_s(p)| \tag{4}$$

where p is the pixel coordinate index, and $\hat{I}_s$ is the projected image of the source view I_s onto the target coordinate frame using a depth image-based rendering module. For the rendering, we define the static scene geometry by a collection of depth maps D_i for frame i and the relative camera motion $T_{t\to s}$ from the target to the source frame. The relative 2D rigid flow from target image I_t to source image I_s can be represented by[1]:

$$f^{rig}_{t\to s}(p_t) = KT_{t\to s}D_t(p_t)K^{-1}p_t - p_t, \tag{5}$$

where K denotes the 4×4 camera transformation matrix and p_t denotes homogeneous coordinates of pixels in target frame

[1] Similar to Zhou et al. (2017), we omit the necessary conversion to homogeneous coordinates for notation brevity.

I_t. Algorithm 1 gives the pseudo-code of the spatial transformer algorithm.

We interpolate the nondiscrete p_s values to find the expected intensity value at that position, using bilinear interpolation with the 4 discrete neighbours of p_s (Zhou, Tulsiani, Sun, Malik, & Efros, 2016). The mean intensity value for projected pixel is estimated as follows:

$$\hat{I}_s(p_t) = I_s(p_s) = \sum_{i\in\{top,bottom\},j\in\{left,right\}} w^{ij} I_s(p_s^{ij}) \tag{6}$$

where w^{ij} is the proximity value between the projected and neighbouring pixels, which sums up to 1. Guided by these positional constraints, we can apply differentiable inverse warping (Jaderberg, Simonyan, Zisserman, & Kavukcuoglu, 2015) between nearby frames, which later becomes the foundation of our self-supervised learning scheme.

Algorithm 1 Spatial Transformer Algorithm

$m, n \leftarrow$ Height, Width ▷ Input image dimensions
$\mathbf{C} \leftarrow Stack((1, 2, \dots, m), (1, 2, \dots, n))$ ▷ Pixel coordinates in homogeneous form
function InverseWarp($\mathbf{I}_s$, $\mathbf{p}_s$, $\mathbf{M}_s$)
  $\mathbf{T}_{t\to s}$ = Rodrigues2TransformationMatrix($\mathbf{p}_s$)
  $\tilde{\mathbf{C}} = \mathbf{T}_{t\to s}\mathbf{C}$ ▷ Transformed points
  $\tilde{\mathbf{C}} =$ Normalize($\tilde{C}$) ▷ Normalized pixel coordinates in $[-1, 1]$
  $\hat{\mathbf{I}}_s =$ BilinearSample($\tilde{\mathbf{C}}$, $\mathbf{I}_s$) ▷ Reconstructed RGB image
  if $\mathbf{M}_s$!= *None* **then**
    $\tilde{\mathbf{M}}_s =$ BilinearSample($\tilde{\mathbf{C}}$, $\mathbf{M}_s$) ▷ Re-sampled mask
  else
    $\tilde{\mathbf{M}}_s = 1$
  return $\hat{\mathbf{I}}_s$, $\tilde{\mathbf{M}}_s$

3.6. View discriminator

The L2 and L1 losses produce blurry results on image generation problems (Larsen, Sønderby, Larochelle, & Winther, 2016). Although these losses fail to encourage high-frequency crispness, in many cases, they nonetheless accurately capture the low frequencies. This motivates restricting the GAN discriminator to model a high-frequency structure, relying on an L1 term to force low-frequency correctness. To model high frequencies, it is sufficient to restrict our attention to the structure in local image patches. Therefore, we employ the PatchGAN (Isola et al., 2017) discriminator architecture that only penalizes the structure at the scale of patches. This discriminator tries to classify each $M \times M$ patch in an image as real or fake. We run this discriminator convolutionally across the image, averaging all responses to provide the ultimate output of D. Such a discriminator effectively models the image as a Markov random field, assuming independence between pixels separated by more than a patch diameter. This connection was previously explored in Li and Wand (2016), and is also the common assumption in models of texture (Gatys, Ecker, & Bethge, 2015), which can be interpreted as a form of texture loss.

The spatial transformer module synthesizes a realistic image by the view reconstruction algorithm using the depth image generated by G and estimated pose value. D classifies the input images sampled from the target data distribution p_{data} into the fake and real categories, playing an adversarial role. These networks are trained by optimizing the objective loss function:

$$\mathcal{L}_d = \min_G \max_D V(G, D) = \mathbb{E}_{\mathbf{I}\sim p_{data}(\mathbf{I})}[\log(D(\mathbf{I}))] + \mathbb{E}_{\mathbf{z}\sim p(\mathbf{z})}[\log(1 - D(G(\mathbf{z})))], \tag{7}$$

where $\mathbf{I}$ is the sample image from the p_{data} distribution and $\mathbf{z}$ is a feature encoding of $\mathbf{I}$ on the latent space.

3.7. The adversarial training

In contrast to the original GAN (Goodfellow et al., 2014), we remove fully connected hidden layers for deeper architectures and use batchnorms in the G and D networks. We replace pooling layers with strided convolutions and fractional-strided convolutions in D and G networks, respectively. For all layer activations, we use LeakyReLU and ReLU in the D and G networks, respectively, except for the output layer that uses tanh nonlinearity. The GAN with these modifications and loss functions generates nonblurry depth maps and resolves the convergence problem during the training (Radford, Metz, & Chintala, 2016). The final objective for the optimization during the training is:

$$\mathcal{L}_{final} = \mathcal{L}_g + \beta\mathcal{L}_d \tag{8}$$

where β is the balance factor that is experimentally found to be optimal by the ratio between the expected values $\mathcal{L}_g$ and $\mathcal{L}_d$ at the end of the training.

4. Experimental setup

In this section, the datasets used in the experiments, network training protocol, evaluation methods are introduced including ablation studies and performance evaluation in cases of poor intersensor calibration.

4.1. Datasets

4.1.1. KITTI

The KITTI odometry dataset (Geiger et al., 2013) is a benchmark for depth and odometry evaluations including vision and LIDAR-based approaches. Images are recorded at 10 Hz via an onboard camera mounted on a Volkswagen Passat B6. Frames are recorded in various environments such as residential, road, and campus scenes adding up to a 39.2 km travel length. Ground truth pose values at each camera exposure are determined using an OXTS RT 3003 GPS solution with an accuracy of 10 cm. The corresponding ground truth pixel depth values are acquired via a Velodyne laser scanner. A temporal synchronization between sensors is provided using a software-based calibration approach.

We evaluate SelfVIO on the KITTI odometry dataset using Eigen et al.'s split (Eigen et al., 2014). We use sequences 00–08 for training and 09–10 for the test set that is consistent across related works (Almalioglu, Santamaria-Navarro, Morrell and Agha-Mohammadi, 2021; Almalioglu et al., 2019; Eigen et al., 2014; Liu et al., 2016; Mahjourian et al., 2018; Yin & Shi, 2018; Zhou et al., 2017; Zou et al., 2018). Additionally, 5% of KITTI sequences 00–08 are withheld as a validation set, which leaves a total of 18,422 training images, 2791 testing images, and 969 validation images. Input images are scaled to 256×832 for training, whereas they are not limited to any specific image size at test time. In all experiments, we randomly select an image for the target and use consecutive images for the source. Corresponding 100 Hz IMU data are collected from the KITTI raw datasets and for each target image, the preceding 100 ms and the following 100 ms of IMU data are combined yielding a tensor of size 20×6 (100 ms between the source images and target). Thus, the network learns how to implicitly estimate a temporal offset between camera and IMU as well as to determine an estimate of the initial velocity at the time of target image timestamp by looking to corresponding IMU data.

Table 1
Results on depth estimation. Supervised methods are shown in the first three rows. Data refers to the training set: Cityscapes (cs) and KITTI (k). For the experiments involving CS dataset, SelfVIO is trained without IMU as CS dataset lacks IMU data.

Method	Data	Error (m)				Accuracy, δ		
		AbsRel	SqRel	RMS	RMSlog	<1.25	$<1.25^2$	$<1.25^3$
Eigen et al. (2014) coarse	k	0.214	1.605	6.563	0.292	0.673	0.884	0.957
Eigen et al. (2014) fine	k	0.203	1.548	6.307	0.282	0.702	0.890	0.958
Liu, Shen, Lin, and Reid (2016)	k	0.202	1.614	6.523	0.275	0.678	0.895	0.965
SfM-Learner (Zhou et al., 2017)	cs+k	0.198	1.836	6.565	0.275	0.718	0.901	0.960
Mahjourian et al. (2018)	cs+k	0.159	1.231	5.912	0.243	0.784	0.923	0.970
Geonet (Yin & Shi, 2018)	cs+k	0.153	1.328	5.737	0.232	0.802	0.934	0.972
DF-Net (Zou, Luo, & Huang, 2018)	cs+k	0.146	1.182	5.215	**0.213**	0.818	0.943	**0.978**
CC (Ranjan et al., 2019)	cs+k	0.139	1.032	5.199	**0.213**	0.827	0.943	0.977
GANVO (Almalioglu et al., 2019)	cs+k	**0.138**	1.155	4.412	0.232	0.820	0.939	0.976
SelfVIO (ours, no-IMU)	cs+k	**0.138**	**1.013**	**4.317**	0.231	**0.849**	**0.958**	0.979
SfM-Learner (Zhou et al., 2017)	k	0.183	1.595	6.709	0.270	0.734	0.902	0.959
Mahjourian et al. (2018)	k	0.163	1.240	6.220	0.250	0.762	0.916	0.968
Geonet (Yin & Shi, 2018)	k	0.155	1.296	5.857	0.233	0.793	0.931	0.973
DF-Net (Zou et al., 2018)	k	0.150	1.124	5.507	0.223	0.806	0.933	0.973
CC (Ranjan et al., 2019)	k	0.140	1.070	5.326	**0.217**	0.826	0.941	0.975
GANVO (Almalioglu et al., 2019)	k	0.150	1.141	5.448	0.216	0.808	0.939	0.975
SelfVIO (ours)	k	**0.127**	**1.018**	**5.159**	0.226	**0.844**	**0.963**	**0.984**

4.1.2. EuRoC

The EuRoC dataset (Burri et al., 2016) contains 11 sequences recorded onboard from an AscTec Firefly micro aerial vehicle (MAV) while it was manually piloted around three different indoor environments executing 6-DoF motions. Within each environment, the sequences increase qualitatively in difficulty with increasing sequence numbers. For example, Machine Hall 01 is "easy", while Machine Hall 05 is a more challenging sequence in the same environment, containing faster and loopier motions, poor illumination conditions etc. We evaluate SelfVIO on the EuRoC odometry dataset using MH02(E), MH04(D), V103(D), V202(D) for testing, and the remaining sequences for training. Additionally, 5% of the training sequences are withheld as a validation set. All the EuRoC sequences are recorded by a front-facing visual–inertial sensor unit with tight synchronization between the stereo camera and IMU timestamps captured using a MAV. Accurate ground truth is provided by laser or motion capture tracking depending on the sequence, which has been used in many of the existing partial comparative evaluations of VIO methods. The dataset provides synchronized global shutter WVGA stereo images at a rate of 20 Hz that we use only the left camera image, and the acceleration and angular rate measurements captured by a Skybotix VI IMU sensor at 200 Hz. In the Vicon Room sequences, ground truth positioning measurements are provided by Vicon motion capture systems, while in the Machine Hall sequences, ground truth is provided by a Leica MS50 laser tracker. The dataset containing sequences, ground truth and sensor calibration data is publicly available.[2] The EuRoC dataset, being recorded indoors on unstructured paths, exhibits motion blur and the trajectories follow highly irregular paths, unlike the KITTI dataset (García, Molina, & Trincado, 2020).

4.1.3. Cityscapes

The Cityscapes Urban Scene 2016 dataset (Cordts et al., 2016) is a large-scale dataset mainly used for semantic urban scene understanding, which contains 22,973 stereo images for autonomous driving in an urban environment collected in street scenes from 50 different cities across Germany spanning several months. The dataset also provides precomputed disparity depth maps associated with the RGB images. Although it has a similar setting to the KITTI dataset, the Cityscapes dataset has a higher resolution (2048×1024), more image quality, and variety. We cropped the input images to keep only the top 80% of the image, removing the very reflective car hoods.

[2] http://projects.asl.ethz.ch/datasets/doku.php?id=kmavvisualinertialdatasets.

4.2. Network training

We implement the architecture with the publicly available Tensorflow framework (Abadi et al., 2016). Batch normalization is employed for all of the layers except for the output layers. Three consecutive images are stacked together to form the input batch, where the central frame is the target view for the depth estimation. We augment the data with random scaling, cropping and horizontal flips. SelfVIO is trained for 100,000 iterations using a batch size of 16. During the network training, we calculate errors on the validation set at intervals of 1000 iterations. We use the ADAM (Kingma & Ba, 2014) solver with $momentum1 = 0.9$, $momentum2 = 0.99$, $gamma = 0.5$, learning rate $= 2e{-}4$, and an exponential learning rate policy. The network is trained using single-point precision (FP32) on a desktop computer with a 3.00 GHz Intel i7-6950X processor and NVIDIA Titan V GPUs. The proposed model runs at 81 ms per frame on a Titan V GPU, taking 33 ms for depth generation, 27 ms for visual odometry, and 21 ms for IMU processing and sensor fusion.

4.3. Evaluation

We evaluate the depth and ego-motion prediction performance of SelfVIO on three benchmark and challenging datasets such as KITTI, EuRoC and CityScapes as explained above. We evaluate ego-motion prediction performance of the proposed approach on benchmark evaluation methods such as relative and absolute pose error in terms of translational and rotational errors. We visualize the resulting full trajectories without involving additional loop closures to show the odometry performance. Furthermore, we analyse the depth prediction performance in terms of widely-used depth error and accuracy metrics, which is a critical component of the self-supervision signal. SelfVIO formulates multi-modal ego-motion using a tight connection between depth prediction and ego-motion estimation to eliminate the need for the labelled data. We also visualize the monocular depth predictions for given input RGB frames. Moreover, we visualize the adaptive fusion of the proposed approach under various motions and diverse settings to show the effectiveness of the proposed fusion approach.

We compare our approach to a collection of recent VO, VIO, and VSLAM approaches described earlier in Section 2:

- Learning-based methods:
  - SFMLearner (Zhou et al., 2017)

Table 2
Monocular VO results with our proposed SelfVIO evaluated on the training sequences. No loop closure is performed in the methods listed in the table. Note that monocular VISO2 and ORB-SLAM (without loop closure) did not work with image resolution 416 × 128, the results were obtained with full image resolution 1242 × 376. 7-DoF (6-DoF + scale) alignment with the ground-truth is applied for SfMLearner and monocular ORB-SLAM.

	Seq.00		Seq.02		Seq.05		Seq.07		Seq.08		Mean	
	t_{rel} (%)	r_{rel} (°)	t_{rel} (%)	r_{rel} (°)	t_{rel} (%)	r_{rel} (°)	t_{rel} (%)	r_{rel} (°)	t_{rel} (%)	r_{rel} (°)	t_{rel} (%)	r_{rel} (°)
SelfVIO	**1.24**	**0.45**	**0.80**	**0.25**	**0.89**	0.63	0.91	**0.49**	**1.09**	**0.36**	**0.95**	**0.44**
VIOLearner	1.50	0.61	1.20	0.43	0.97	**0.51**	**0.84**	0.66	1.56	0.61	1.21	0.56
UnDeepVO	4.14	1.92	5.58	2.44	3.40	1.50	3.15	2.48	4.08	1.79	4.07	2.03
SfMLearner	65.27	6.23	57.59	4.09	16.76	4.06	17.52	5.38	24.02	3.05	36.23	4.56
VISO2	18.24	2.69	4.37	1.18	19.22	3.54	23.61	4.11	24.18	2.47	17.92	2.80
ORB-SLAM	25.29	7.37	26.30	3.10	26.01	10.62	24.53	10.83	32.40	12.13	27.06	10.24

- t_{rel}: average translational RMSE drift (%) on length of 100 m–800 m.
- r_{rel}: average rotational RMSE drift (°/100 m) on length of 100 m–800 m.

Table 3
Comparisons to monocular VIO approaches on KITTI Odometry sequence 10. We present medians, first quartiles, and third quartiles of translational errors in meters. The results for the benchmark methods are reproduced from Clark et al. (2017) and Shamwell et al. (2020). We report errors on distances of 100, 200, 300, 400, 500 m from KITTI Odometry sequence 10 to have identical metrics with Clark et al. (2017) and Shamwell et al. (2020). Full results for SelfVIO on sequence 10 can be found in Table 4.

	100 m			200 m			300 m			400 m			500 m		
	Med.	1st Quar.	3rd Quar.	Med.	1st Quar.	3rd Quar.	Med.	1st Quar.	3rd Quar.	Med.	1st Quar.	3rd Quar.	Med.	1st Quar.	3rd Quar.
SelfVIO	1.18	0.82	1.77	2.85	2.03	3.89	**5.11**	3.09	7.15	**7.48**	5.31	9.26	**8.03**	6.59	10.29
SelfVIO (no IMU)	2.25	1.33	2.64	4.3	2.92	5.57	7.29	5.51	10.93	13.11	10.34	15.17	17.29	15.26	19.07
SelfVIO (LSTM)	1.21	0.81	1.83	3.08	2.11	4.76	5.35	3.18	8.06	7.81	5.76	9.41	9.13	6.61	10.95
VIOLearner	1.42	1.01	2.01	3.37	2.27	5.71	5.7	3.24	8.31	8.83	5.99	10.86	10.34	6.67	12.92
VINET	**0**	0	2.18	**2.5**	1.01	5.43	6	3.26	17.9	10.3	5.43	39.6	16.8	8.6	70.1
EKF+VISO2	2.7	0.54	9.2	11.9	4.89	32.6	26.6	9.23	58.1	40.7	13	83.6	57	19.5	98.9

  - Mahjourian et al. (2018) (results reproduced from Mahjourian et al. (2018))
  - Zhan et al. (2018) (results reproduced from Zhan et al. (2018))
  - VINet (Clark et al., 2017)
  - UnDeepVO (Li et al., 2018)
  - Geonet (Yin & Shi, 2018)
  - DF-Net (Zou et al., 2018)
  - Competitive Collaboration (CC) (Ranjan et al., 2019)
  - VIOLearner-RGB (Shamwell et al., 2020)
- Traditional methods:
  - OKVIS (Leutenegger et al., 2015)
  - ROVIO (Bloesch et al., 2017)
  - VISO2 (results reproduced from Shamwell et al. (2020))
  - ORB-SLAM (results reproduced from Shamwell et al. (2020))
  - SVO+MSF (Faessler et al., 2016; Forster, Zhang, Gassner, Werlberger, & Scaramuzza, 2017)
  - VINS-Mono (Qin et al., 2018)
  - EKF+VISO2 (results reproduced from Clark et al. (2017))
  - MSCKF (Mourikis & Roumeliotis, 2007)

We include monocular versions of competing algorithms to have a common setup with our method. SFMLearner, Mahjourian et al. Zhan et al. and VINet optimize over multiple consecutive monocular images or stereo image pairs; and OKVIS and ORB-SLAM perform bundle adjustment. Similarly, we include the RGB version of VIOLearner for all the comparisons, which uses RGB image input and the monocular depth generation sub-network from SFMLearner (Shamwell et al., 2020) rather than RGB-depth data. We perform 6-DOF least-squares Umeyama alignment (Umeyama, 1991) for trajectory alignment on monocular approaches as they lack scale information. For SFMLearner, we follow (Zhou et al., 2017) to estimate the scale from the ground truth for each estimate. We evaluate the compared methods at images scaled down to size 256 × 832 to match the image resolution used by SelfVIO.

We train separate networks for the KITTI and EuRoC datasets for benchmarking and the Cityscapes dataset (Cordts et al., 2016) for evaluating the cross-dataset generalization ability of the model. SelfVIO implicitly learns to estimate camera-IMU extrinsics and IMU intrinsics directly from raw data without any need for external calibration.

4.3.1. Ablation studies

We perform two ablation studies on our proposed network and call these SelfVIO (no IMU) and SelfVIO (LSTM). The ablation studies show the effect of each component on the overall performance.

Visual vs. Visual–Inertial. We disable the inertial odometry module and omit IMU data; instead, we use vision-only odometry to estimate the initial warp. This version of the network is referred to as SelfVIO (no IMU) and results are only included to provide additional perspective on the vision-only performance of our architecture (and specifically the adversarial training) compared to other vision-only approaches.

CNN vs RNN. Deep learning models might benefit from larger and more complex networks to improve the ego-motion and depth prediction accuracy (Guizilini, Ambrus, Pillai, Raventos, & Gaidon, 2020), which comes at a run-time cost. SelfVIO employs CNNs to process the sequential IMU data, which have much less trainable hyper-parameters than RNNs such as LSTMs. Additionally, we perform ablation studies where we replace the convolutional network described in Section 3.3 with a recurrent neural network, specifically a bidirectional LSTM to process IMU input at the cost of an increase in the number of parameters and, hence, more computational power. This version of the network is referred to as SelfVIO (LSTM).

4.3.2. Spatial misalignments

We test the robustness of our method against camera-sensor miscalibration. We introduce calibration errors by adding a rotation of a chosen magnitude and random angle $\Delta R_s \sim \text{vMF}(\cdot|\mu, \kappa)$

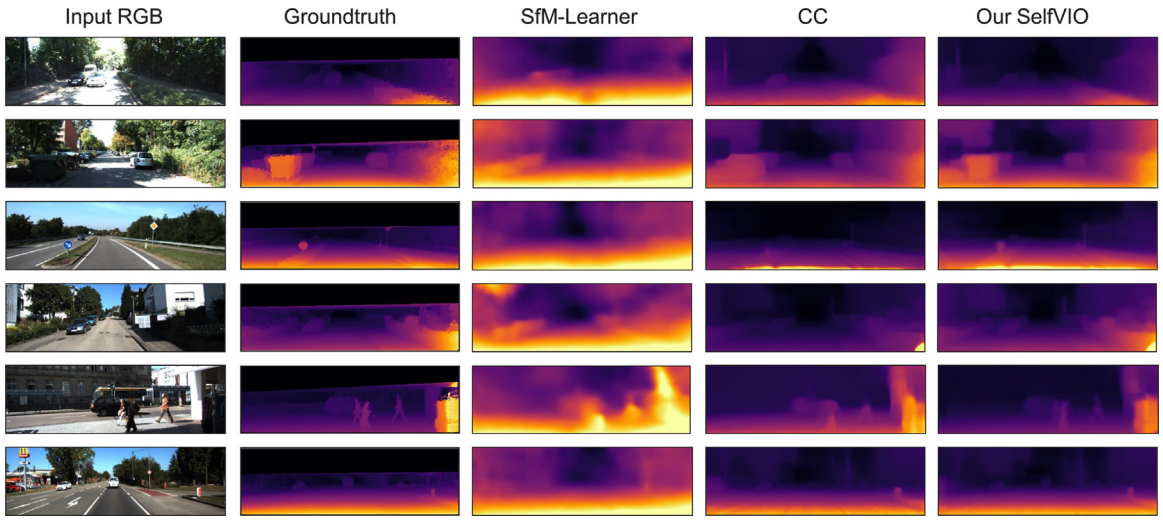


Fig. 3. Qualitative results for monocular depth map estimation. Comparison of unsupervised monocular depth estimation between SfM-Learner (Zhou et al., 2017), CC (Ranjan et al., 2019) and the proposed SelfVIO. To visualize the ground truth depth map, we interpolated the sparse LIDAR point clouds and projected them onto the camera imaging plane using the provided KITTI extrinsic and camera calibration matrices. As seen in the figure, SelfVIO captures details in challenging scenes containing low-textured areas, shaded regions, and uneven road lines, preserving sharp, accurate and detailed depth map predictions both in close and distant regions.

to the camera-IMU rotation matrices R_s, where $\text{vMF}(\cdot|\mu,\kappa)$ is the von Mises–Fisher distribution (Wood, 1994), μ is the directional mean and κ is the concentration parameter of the distribution. We apply the calibration offsets during testing. Note that these are never used during training.

4.3.3. Evaluation metrics

We evaluate our trajectories primarily using the standard KITTI relative error metric (reproduced below from Geiger et al., 2013):

$$E_{rot}(\mathcal{F}) = \frac{1}{|\mathcal{F}|}\sum_{(i,j)\in\mathcal{F}} \|(\hat{\mathbf{q}}_i \ominus \hat{\mathbf{q}}_j) \ominus (\mathbf{q}_i \ominus \mathbf{q}_j)\|_2, \tag{9}$$

$$E_{trans}(\mathcal{F}) = \frac{1}{|\mathcal{F}|}\sum_{(i,j)\in\mathcal{F}} \|(\hat{\mathbf{p}}_i \ominus \hat{\mathbf{p}}_j) \ominus (\mathbf{p}_i \ominus \mathbf{p}_j)\|_2, \tag{10}$$

where $\mathcal{F}$ is a set of frames, $\ominus$ is the inverse compositional operator, $\mathbf{x} = [\mathbf{p}, \mathbf{q}] \in SE(3)$ and $\hat{\mathbf{x}} = [\hat{\mathbf{p}}, \hat{\mathbf{q}}] \in SE(3)$ are estimated and true pose values as elements of Lie group $SE(3)$, respectively.

For the KITTI dataset, we also evaluate the errors at lengths of 100, 200, 300, 400, and 500 m. Additionally, we compute the root mean squared error (RMSE) for trajectory estimates on five frame snippets as has been done recently in Mahjourian et al. (2018) and Zhou et al. (2017).

We evaluate the depth estimation performance of each method using several error and accuracy metrics from prior works (Eigen et al., 2014):

Threshold: % of y_i s.t. $\max(\frac{y_i}{y_i^*}, \frac{y_i^*}{y_i}) = \delta < thr$

RMSE (linear): $\sqrt{\frac{1}{|T|}\sum_{y\in T} \|y_i - y_i^*\|^2}$

Abs relative difference: $\frac{1}{|T|}\sum_{y\in T} |y - y^*|/y^*$

RMSE (log): $\sqrt{\frac{1}{|T|}\sum_{y\in T} \|\log y_i - \log y_i^*\|^2}$

Squared relative difference: $\frac{1}{|T|}\sum_{y\in T} \|y - y^*\|^2/y^*$

5. Results and discussion

In this section, we critically analyse and comparatively discuss our qualitative and quantitative results for depth and motion estimation.

5.1. Monocular depth estimation

We obtain state-of-the-art results on single-view depth prediction as quantitatively shown in Table 1. The depth reconstruction performance is evaluated on the Eigen et al. (2014) split of the raw KITTI dataset (Geiger et al., 2012), which is consistent with previous work (Eigen et al., 2014; Liu et al., 2016; Mahjourian et al., 2018; Yin & Shi, 2018). All depth maps are capped at 80 m. The predicted depth map, D_p, is multiplied by a scaling factor, $\hat{s}$, that matches the median with the ground truth depth map, D_g, to solve the scale ambiguity issue, i.e. $\hat{s} = median(D_g)/median(D_p)$.

Fig. 3 shows examples of reconstructed depth maps by the proposed method, GeoNet (Yin & Shi, 2018) and the Competitive Collaboration (CC) (Ranjan et al., 2019). It is clearly seen that SelfVIO outputs sharper and more accurate depth maps compared to the other methods that fundamentally use an encoder–decoder network with various implementations. An explanation for this result is that adversarial training using the convolutional domain-related feature set of the discriminator distinguishes reconstructed images from the real images, leading to less blurry results (Dosovitskiy & Brox, 2016). Moreover, although GeoNet (Yin & Shi, 2018) and CC (Ranjan et al., 2019) benchmark methods train additional networks to segment and mask inconsistent regions in the reconstructed frame caused by moving objects, occlusions and re-projection errors, SelfVIO implicitly accounts for these inconsistencies without any need for an additional network. Furthermore, Fig. 3 further implies that the depth reconstruction module proposed by SelfVIO is capable of capturing small objects in the scene whereas the other methods tend to ignore them. A loss function in the image space leads to smoothing out all likely detail locations, whereas an adversarial loss function in feature space with a natural image prior makes the proposed SelfVIO more sensitive to details in the scene (Dosovitskiy & Brox, 2016). The proposed SelfVIO also performs better in low-textured areas caused by the shading inconsistencies in a scene and predicts the depth values of the corresponding objects much better in such cases. In Fig. 4, we demonstrate typical performance degradation of the compared unsupervised methods that is caused by challenges such as poor road signs in rural areas and huge objects occluding most of the visual input. Even in these cases, SelfVIO performs slightly better than the existing methods.

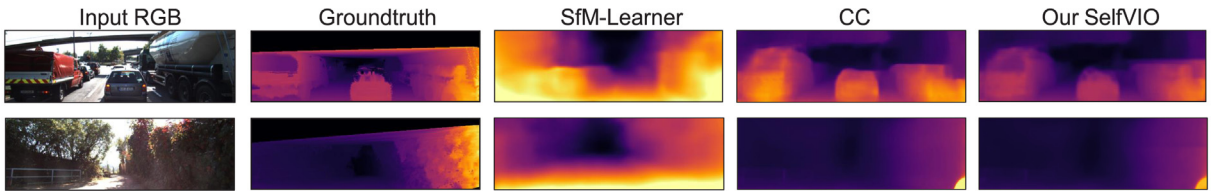


Fig. 4. Degradation in depth reconstruction. The performance of the compared methods SfM-Learner (Zhou et al., 2017), CC (Ranjan et al., 2019) and the proposed SelfVIO degrades under challenging conditions such as vast open rural scenes and huge objects occluding the camera view.

Table 4
Comparisons to monocular VO and monocular VIO approaches on KITTI test sequences 09 and 10. t_{rel}(%) is the average translational error percentage on lengths 100–800 m and r_{rel} (°) is the rotational error (°/100 m) on lengths 100–800 m calculated using the standard KITTI benchmark (Geiger et al., 2013). No loop closure is performed for ORB-SLAM. We evaluate the monocular versions of the compared methods.

	Seq.09		Seq.10		Mean	
	t_{rel} (%)	r_{rel} (°)	t_{rel} (%)	r_{rel} (°)	t_{rel} (%)	r_{rel} (°)
SelfVIO	**1.95**	**1.15**	**1.81**	**1.30**	**1.88**	**1.23**
SelfVIO (no IMU)	2.49	1.28	2.33	1.96	2.41	1.62
SelfVIO (LSTM)	2.10	1.19	2.03	1.44	2.07	1.32
VIOLearner	2.27	1.52	2.74	1.35	2.53	1.31
SfMLearner	21.63	3.57	20.54	10.93	21.09	7.25
Zhan et al.	11.92	3.60	12.62	3.43	12.27	3.52
ORB-SLAM	45.52	3.10	6.39	3.20	25.96	3.15
OKVIS	9.77	2.97	17.30	2.82	13.51	2.90
ROVIO	20.18	2.09	20.04	2.24	10.11	2.17
ORB-SLAM[a]	24.41	2.08	3.16	2.15	13.79	2.12
OKVIS[a]	5.69	1.89	10.82	1.80	8.26	1.85
ROVIO[a]	12.38	1.71	10.74	1.75	11.56	1.73

- t_{rel}: average translational RMSE drift (%) on length of 100 m–800 m.
- r_{rel}: average rotational RMSE drift (°/100 m) on length of 100 m–800 m.

[a]Full resolution input image (1242 × 376).

Moreover, we select a challenging evaluation protocol to test the adaptability of the proposed approach by training on the Cityscapes dataset and fine-tuning on the KITTI dataset (cs+k in Table 1). Although SelfVIO learns the inter-sensor calibration parameters as part of the training process, it can adapt to the new test environment with fine-tuning. As the Cityscapes dataset is an RGB-depth dataset, we remove the inertial odometry part and perform an ablation study (SelfVIO (no IMU)) on depth estimation. While all the learning-based methods in comparison exhibit performance drop in the fine-tuning setting, the results shown in Table 1 show a clear advantage of fine-tuning on data that is related to the test set. In this mode (SelfVIO (no IMU)), our network architecture for depth estimation is most similar to GANVO (Almalioglu et al., 2019). However, the shared features among the encoder and generator networks enable the network to also have access to low-level information. In addition, the PatchGAN structure in SelfVIO restricts the discriminator from capturing high-frequency structure in depth map estimation. We observe that using the SelfVIO framework with inertial odometry results in larger performance gains even when it is trained on the KITTI dataset only.

Fig. 7 visualizes sample depth maps reconstructed from MAV frames in the EuRoC dataset. Although there is no ground-truth depth map of the frames available in the EuRoC dataset for quantitative analysis, qualitative results in Fig. 7 indicates the effectiveness of the depth map reconstruction as well as the efficacy of the proposed approach in datasets containing diverse 6-DoF motions.

Table 5
Absolute Trajectory Error (ATE) in meters on KITTI odometry dataset. We also report the results of the other methods for comparison that are taken from Yin and Shi (2018) and Zhou et al. (2017). Our method outperforms all of the compared methods. No loop closure is performed for ORB-SLAM.

Method	Seq.09	Seq.10
ORB-SLAM (full)	0.014 ± 0.008	0.012 ± 0.011
ORB-SLAM (short)	0.064 ± 0.141	0.064 ± 0.130
Zhou et al. (2017)	0.021 ± 0.017	0.020 ± 0.015
SfM-Learner (Zhou et al., 2017)	0.016 ± 0.009	0.013 ± 0.009
GeoNet (Yin & Shi, 2018)	0.012 ± 0.007	0.012 ± 0.009
CC (Ranjan et al., 2019)	0.012 ± 0.007	0.012 ± 0.008
GANVO (Almalioglu et al., 2019)	0.009 ± 0.005	0.010 ± 0.013
VIOLearner (Shamwell et al., 2020)	0.012	0.012
SelfVIO (ours)	**0.008 ± 0.006**	**0.009 ± 0.008**

5.2. Motion estimation

In this section, we comparatively discuss the motion estimation performance of the proposed method in terms of both vision-only and visual–inertial estimation modes.

5.2.1. Visual odometry

SelfVIO (no IMU) outperforms the VO approaches listed in Section 4.3 as seen in Table 2, which confirms that our results are not due solely to our inclusion of IMU data. We evaluated monocular VISO2 and ORB-SLAM (without loop closure) using full image resolution 1242 × 376 as they did not work with image resolution 416 × 128. It should be noted that the results in Table 2 are for SelfVIO, VIOLearner, UnDeepVO, and SFMLearner networks that are tested on data on which they are also trained, which corresponds with the results presented in Li et al. (2018) and Shamwell et al. (2020). Although the sequences in Table 2 are used during the training, the results in Table 2 indicate the effectiveness of the supervisory signal as the unsupervised methods do not incorporate ground-truth pose and depth maps. We compare SelfVIO against UnDeepVO and VIOLearner using these results.

We also evaluate SelfVIO more conventionally by training on sequences 00 − 08 and testing on sequences 09 and 10 that were not used in the training as was the case for Shamwell et al. (2020) and Zhou et al. (2017). These results are shown in Table 4. SelfVIO significantly outperforms SFMLearner on both KITTI sequences 09 and 10. We also evaluate ORB-SLAM, OKVIS, and ROVIO using the full-resolution input images to show the effect of reducing the input size.

5.2.2. Visual–inertial odometry

The authors of VINet (Clark et al., 2017) provide the errors in boxplots compared to several state-of-the-art approaches for 100–500 m on the KITTI odometry dataset. We reproduced the median, first quartile, and third quartile from Clark et al. (2017) and Shamwell et al. (2020) to the best of our ability and included them in Table 3. SelfVIO outperforms VIOLearner and VINet for longer trajectories (100, 200, 300, 400, 500 m) on KITTI sequence 10. Although SelfVIO (LSTM) is slightly outperformed

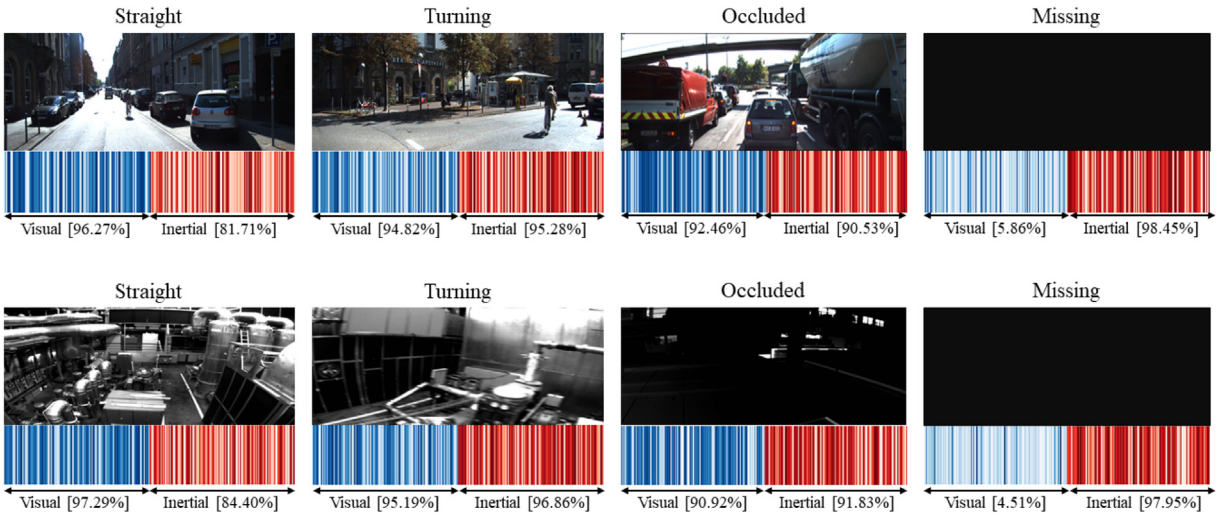


Fig. 5. Visualization of the adaptive fusion under different conditions. The weights and mean percentage activation of visual and inertial features shown at the bottom of each frame reflect the ego-motion dynamics: top: KITTI dataset, bottom: EuRoC dataset. Visual features dominate over inertial features during straight motions, whereas they diminish in importance when faced with a lack of salient visual signals. In the cases of turning and occlusion, the importance of inertial features increases to compensate for the lost visual features due to the reduced overlap between the consecutive frames.

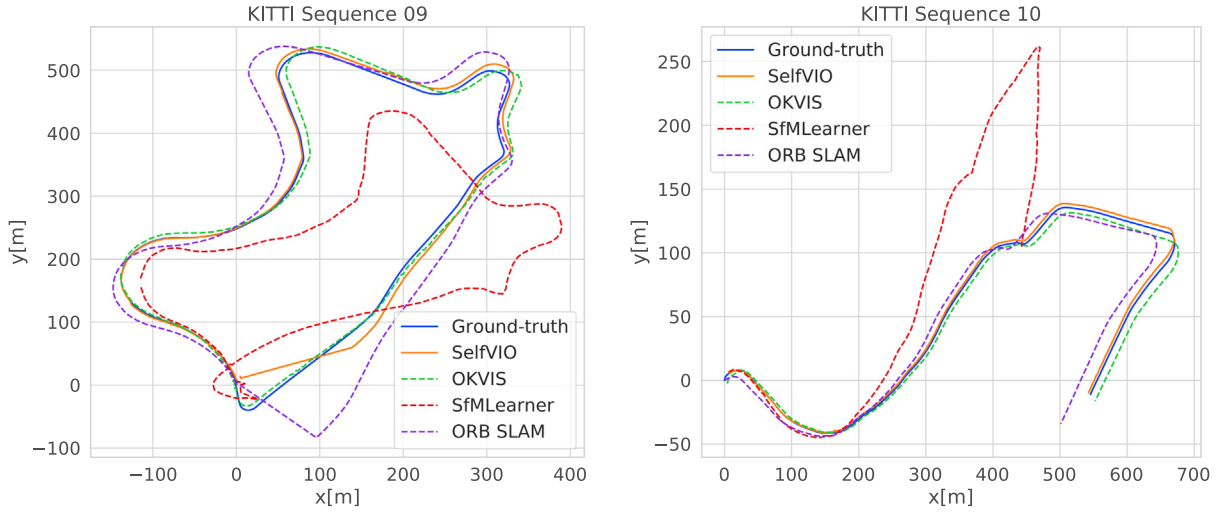


Fig. 6. Sample trajectories comparing the proposed unsupervised learning method SelfVIO with monocular versions of ORB SLAM, OKVIS, SfMLearner, and the ground truth in meter scale on KITTI sequences 09 and 10. SelfVIO shows a better odometry estimation in terms of both rotational and translational motions.

by SelfVIO, it still performs better than VIOLearner and VINET, which shows CNN architecture in SelfVIO increases the estimation performance. It should again be noted that our network can implicitly learn camera-IMU extrinsic calibration from the data. We also compare SelfVIO against the traditional state-of-the-art VIO and include a custom EKF with VISO2 as in VINET (Clark et al., 2017).

We successfully run SelfVIO on the KITTI odometry sequences 09 and 10 and include the results in Table 4 and Fig. 6. SelfVIO outperforms OKVIS and ROVIO on KITTI sequences 09 and 10. However, both OKVIS and ROVIO require tight synchronization between the IMU measurements and the images that KITTI does not provide. Although OKVIS and ROVIO allow for temporal offsets that can be provided in the calibration file, they need either an external calibration process or the online calibration enabled. Besides, OKVIS struggles with the scale when the vehicle moves at a constant velocity. These are most likely the reasons for the poor performance of both approaches on KITTI. Additionally, the acceleration in the KITTI dataset is minimal, which causes a significant drift for the monocular versions of OKVIS and ROVIO. These also highlight a strength of SelfVIO in that it can compensate for loosely temporally synchronized sensors without explicitly estimating their temporal offsets, showing the effectiveness of LSTM in the sensor fusion. Furthermore, we evaluate ORB-SLAM, OKVIS, and ROVIO using the full-resolution images to show the impact of reducing the image resolution (see Table 4). Although higher resolution improves the odometry performance, OKVIS and ROVIO heavily suffer from loose synchronization, and ORB-SLAM is prone to large drifts without loop closure.

In addition to evaluating with relative error over the entire trajectory, we also evaluated SelfVIO RGB using RMSE over five frame snippets as was done in Mahjourian et al. (2018), Shamwell et al. (2020) and Zhou et al. (2017) for their similar monocular approaches. As shown in Table 5, SelfVIO surpasses RMSE performance of SFMLearner, Mahjourian et al. and VIOLearner on KITTI trajectories 09 and 10.

The results on the EuRoC sequences are shown in Table 6 and sample trajectory plots are shown in Fig. 8. SelfVIO produces the most accurate trajectories for many of the sequences, even without explicit loop closing. We additionally evaluate the benchmark

Table 6
Absolute translation errors (RMSE) in meters for all trials in the EuRoC MAV dataset, using monocular versions of all the compared methods. Errors have been computed after the estimated trajectories were aligned with the ground-truth trajectory using the method in Umeyama (1991). The top performing algorithm on each platform and dataset is highlighted in bold. The results for the benchmark methods on full resolution images are reported independently from Delmerico and Scaramuzza (2018). We show the sequences used in the training and testing of the learning-based methods in separate columns.

	Test sequences							Training sequences			
	MH01 (E)	MH03 (M)	MH05 (D)	V101 (E)	V102 (M)	V201 (E)	V203 (D)	MH02 (E)	MH04 (D)	V103 (D)	V202 (M)
OKVIS	0.23	0.33	0.59	0.12	0.26	0.17	0.37	0.30	0.44	0.33	0.21
ROVIO	0.29	0.35	0.69	0.13	0.13	0.17	0.19	0.33	0.63	0.18	0.19
VINSMONO	0.35	0.24	0.46	0.10	0.14	0.11	0.26	0.21	0.29	0.17	0.12
CC (Ranjan et al., 2019) (VO)	0.91	1.13	x	0.69	1.03	0.83	x	0.75	1.18	1.10	0.91
SelfVIO	0.19	0.21	**0.29**	0.08	**0.09**	0.11	**0.11**	0.15	**0.16**	**0.10**	**0.08**
SVOMSF[a]	0.14	0.48	0.51	0.40	0.63	0.20	x	0.20	1.38	x	0.37
MSCKF[a]	0.42	0.23	0.48	0.34	0.20	0.10	1.13	0.45	0.37	0.67	0.16
OKVIS[a]	**0.16**	0.24	0.47	0.09	0.20	0.13	0.29	0.22	0.34	0.24	0.16
ROVIO[a]	0.21	0.25	0.52	0.10	0.10	0.12	0.14	0.25	0.49	0.14	0.14
VINSMONO[a]	0.27	**0.13**	0.35	**0.07**	0.10	**0.08**	0.21	**0.12**	0.23	0.13	**0.08**

[a]Full resolution input image (752 × 480).

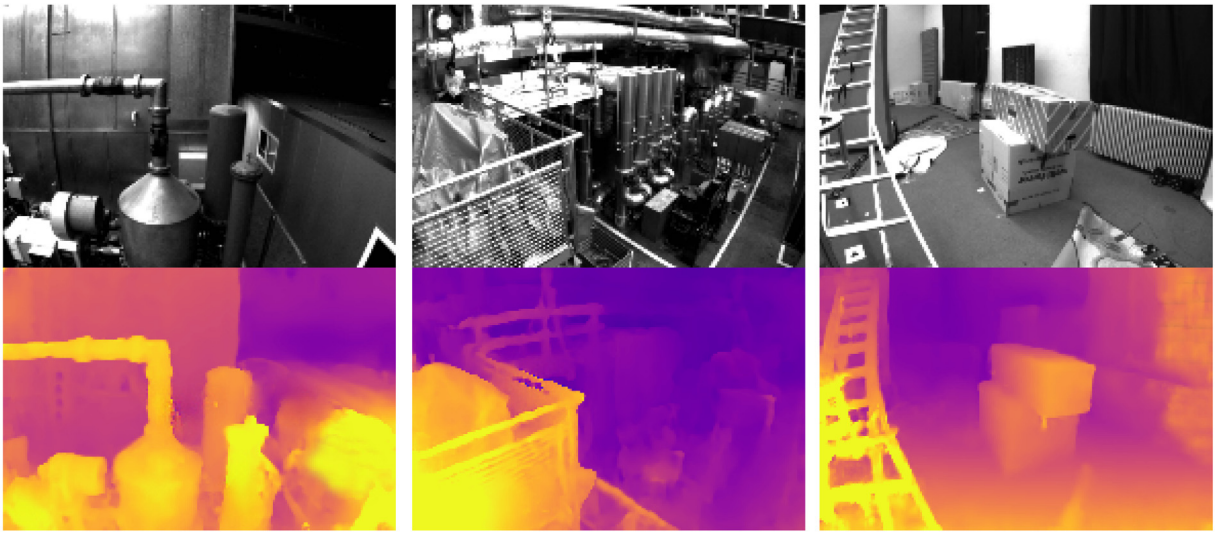


Fig. 7. Qualitative results for monocular depth map estimation on the EuRoC dataset. MAV frames and the corresponding depth maps reconstructed by SelfVIO.

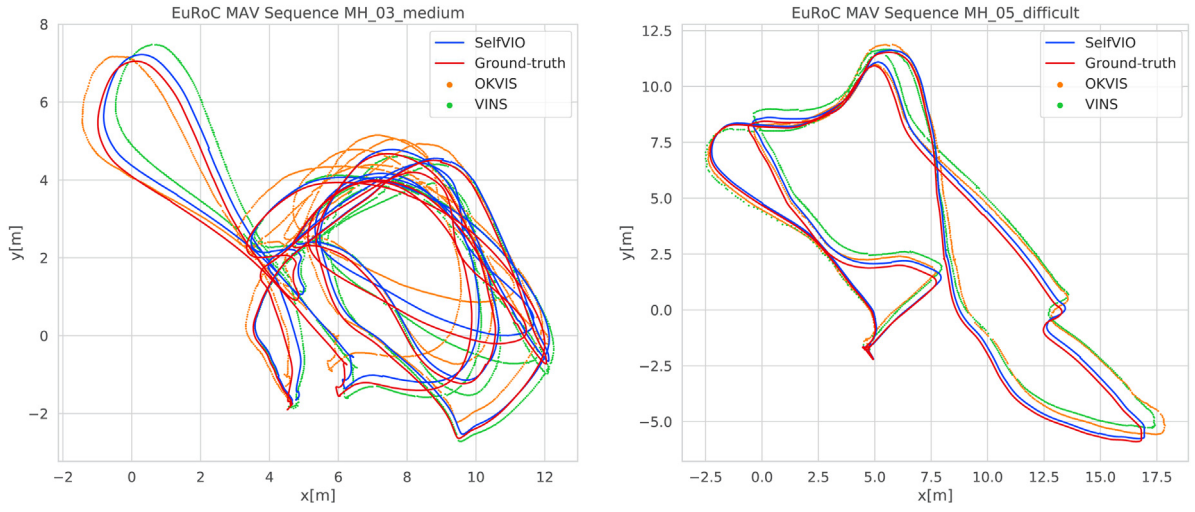
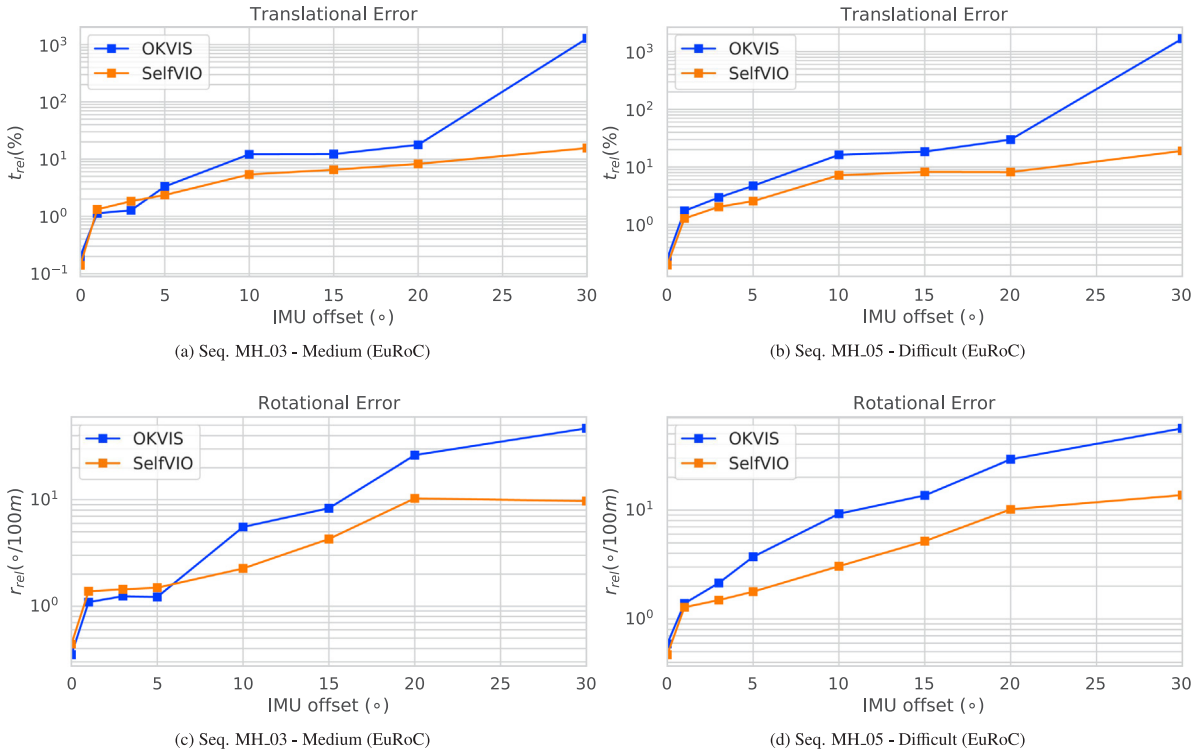


Fig. 8. Sample trajectories comparing the proposed unsupervised learning method SelfVIO with monocular OKVIS and VINS, and the ground truth in meter scale on MH_03 and MH_05 sequences of EuRoC dataset. SelfVIO shows a better odometry estimation in terms of both rotational and translational motions.

methods on the EuRoC dataset using full-resolution input images to show the impact of reducing the image resolution (see Table 6). The full-resolution benchmark results are reported independently from Delmerico and Scaramuzza (2018), which uses a quad-core IntelCore i7-4810MQ CPU with multi-threading at 2.80 GHz, and 32 GB of RAM. We use the recommended parameter settings by



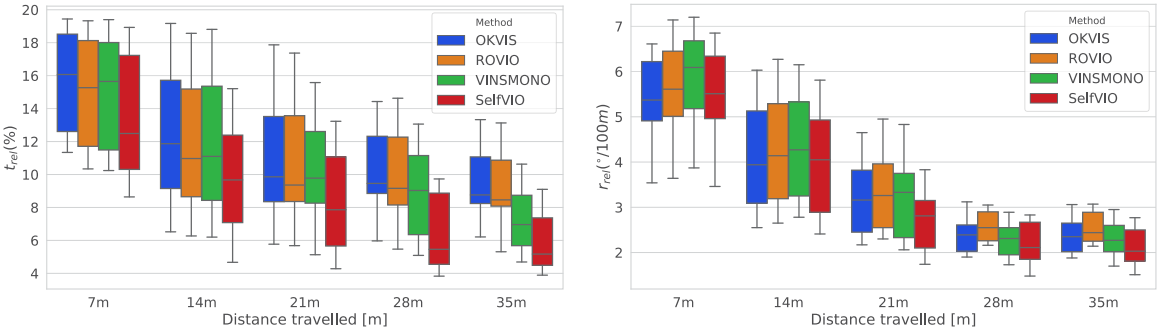
(a) Seq. MH_03 - Medium (EuRoC)

(b) Seq. MH_05 - Difficult (EuRoC)

(c) Seq. MH_03 - Medium (EuRoC)

(d) Seq. MH_05 - Difficult (EuRoC)

Fig. 9. Results on SelfVIO and monocular OKVIS trajectory estimation on the EuRoC sequences MH_03 (left column) and MH_05 (right column) given the induced IMU orientation offset. Measurement errors are shown for each sequence with translational error percentage (top row) and rotational error in degrees per 100 m (bottom row) on lengths 25–100 m. In contrast to SelfVIO, after 20–30°, OKVIS exhibits catastrophic failure in translation and orientation estimation.



(a) Translational relative pose error

(b) Rotational relative pose error

Fig. 10. Boxplot summarizing the relative pose error statistics with respect to the distance travelled for the monocular VIO pipelines on EuRoC dataset over all sequences. Errors are computed over trajectory segments of lengths {7, 14, 21, 28, 35} m. We evaluate the monocular versions of the compared methods.

the authors of each algorithm across all experiments, including the results for full-resolution inputs. Although we use the same powerful hardware for all the compared methods, it is worth noting that SelfVIO needs a powerful GPU as a deep learning-based method. The benchmark methods in Table 6 are lightweight VIO methods, which are mostly designed for onboard platforms with low memory and low computational power. Unlike evaluation methods used in the supervised learning-based methods, we also evaluate SelfVIO on the sequences used for the training to show the effectiveness of the supervisory signal as SelfVIO does not incorporate ground-truth pose and depth maps. The test sequences are also shown in separate columns in Table 6, which are never used during the training. We also evaluate state-of-the-art unsupervised VO method CC (Ranjan et al., 2019) on the EuRoC dataset to show a benchmark result for the unsupervised learning-based methods. In our experiments, CC (Ranjan et al., 2019) failed in MH05 (D) and V203 (D) sequences that contain sharp motions and variable brightness across frames. In Fig. 10, we show statistics for the relative translation and rotation error accumulated over trajectory segments of lengths {7, 14, 21, 28, 35} m over all sequences for each platform-algorithm combination, which is well-suited for measuring the drift of an odometry system. These evaluation distances were chosen based on the length of the shortest trajectory in the EuRoC dataset, VR_02 sequence with 36 m.

To provide an objective comparison to the existing related methods in the literature, we use the following methods for evaluation described earlier in Section 2:

- MSCKF (Mourikis & Roumeliotis, 2007) — multistate constraint EKF,
- SVO+MSF (Faessler et al., 2016) — a loosely coupled configuration of a visual odometry pose estimator (Forster et al., 2017) and an EKF for visual–inertial fusion (Lynen, Achtelik, Weiss, Chli, & Siegwart, 2013),

Table 7
Sensitivity of the compared methods against miscalibrated input. We report the relative rotational and translational errors for translational and (b) temporal offsets spanning a range of several orders of magnitude on EuRoC dataset, using monocular versions of the compared methods.

(a)

	Translational offset					
	0.05 m		0.15 m		0.30 m	
	t_{rel} (%)	r_{rel} (°)	t_{rel} (%)	r_{rel} (°)	t_{rel} (%)	r_{rel} (°)
OKVIS	5.21 ± 1.95	5.16 ± 1.15	20.39 ± 5.06	14.54 ± 2.41	71.53 ± 15.92	65.48 ± 5.62
VINS-Mono	2.63 ± 1.07	8.45 ± 1.57	15.28 ± 4.13	18.14 ± 2.71	32.47 ± 8.86	71.31 ± 7.49
SelfVIO	$\mathbf{1.68 \pm 1.14}$	$\mathbf{2.53 \pm 1.05}$	$\mathbf{10.72 \pm 3.93}$	$\mathbf{9.21 \pm 2.13}$	$\mathbf{25.18 \pm 4.35}$	$\mathbf{51.37 \pm 3.71}$

(b)

	Temporal offset					
	15 ms		30 ms		60 ms	
	t_{rel} (%)	r_{rel} (°)	t_{rel} (%)	r_{rel} (°)	t_{rel} (%)	r_{rel} (°)
OKVIS	18.68 ± 2.78	7.72 ± 1.46	24.03 ± 4.89	18.67 ± 3.79	59.73 ± 10.87	84.63 ± 6.59
VINS-Mono	10.42 ± 1.63	9.23 ± 1.87	18.81 ± 3.19	22.19 ± 3.27	24.51 ± 6.51	90.27 ± 8.61
SelfVIO	$\mathbf{5.43 \pm 1.08}$	$\mathbf{3.63 \pm 1.10}$	$\mathbf{14.31 \pm 3.57}$	$\mathbf{13.29 \pm 2.85}$	$\mathbf{19.37 \pm 5.16}$	$\mathbf{56.26 \pm 4.71}$

- t_{rel}: average translational RMSE drift (%) on length of 100 m–800 m.
- r_{rel}: average rotational RMSE drift (°/100 m) on length of 100 m–800 m.

- OKVIS (Leutenegger et al., 2015) — a keyframe optimization-based method using landmark reprojection errors,
- ROVIO (Bloesch et al., 2017) - an EKF with tracking of both 3D landmarks and image patch features, and
- VINS-Mono (Qin et al., 2018) — a nonlinear-optimization-based sliding window estimator using preintegrated IMU factors.

As we are interested in evaluating the odometry performance of the methods, no loop closure is performed. In difficult sequences (marked with D), the continuous inconsistency in brightness between the images causes failures in feature matching for the filter-based approaches, which can result in divergence of the filter. On the easy sequences (marked with E), although OKVIS and VINSMONO slightly outperform the other methods, the accuracies of SVOMSF, ROVIO and SelfVIO approaches are similar except that MSCKF has a larger error in the machine hall datasets which may be caused by the larger scene depth compared to the Vicon room datasets.

As shown in Fig. 9, orientation offsets within a realistic range of less than 10 degrees show low numbers of errors and great robustness of SelfVIO to sensor implementation with high degrees of miscalibration. Furthermore, offsets within a range of less than 30 degrees display a modestly sloped plateau that suggests successful learning of calibration. In contrast, OKVIS shows surprising robustness to rotation errors under 20 degrees but is unable to handle orientation offsets around the 30 degree mark, where error measures appear to drastically increase. This is plausibly expected because deviations of this magnitude result in a large dimension shift, and unsurprisingly, OKVIS appears unable to compensate. Furthermore, we evaluate SelfVIO, OKVIS, and VINS-Mono on miscalibrated data subject to various translational and temporal offsets between visual and inertial sensors.

Table 7 shows that VINS-Mono and OKVIS perform poorly as the translation and time offsets increase, which is due to their need for tight synchronization. Although large temporal and translational offsets cause a significant increase in rotational and translational errors for all the compared methods, SelfVIO achieves the smallest relative translational and rotational errors under various temporal and translational offsets, which indicates the robustness of SelfVIO against loose temporal and spatial calibration. OKVIS fails to track in sequences MH02-05 when the time offset is set to be 90 ms. We have also tested larger time offsets such as 120 ms, but neither OKVIS nor VINS-Mono provides reasonable estimates. It is worth noting that although OKVIS and VINS-Mono have online calibration modules, SelfVIO is an unsupervised VIO method without any additional online calibration module or additional training on a miscalibrated dataset. Thus, we evaluate the robustness of the compared methods subject to various translational and temporal offsets without enabling the online calibration modules.

By explicitly modelling the sensor fusion process, we demonstrate the strong correlation between the odometry features and motion dynamics. Fig. 5 illustrates that features extracted from visual and inertial measurements are complementary in various conditions. The contribution of inertial features increases in the presence of fast rotation. In contrast, visual features are highly active during large translations, which provides insight into the underlying strengths of each sensor modality.

6. Conclusion

In this work, we presented our SelfVIO architecture and demonstrated superior performance against state-of-the-art VO, VIO, and even VSLAM approaches. Despite using only monocular source–target image pairs, SelfVIO surpasses state-of-the-art depth and motion estimation performances of both traditional and learning-based approaches such as VO, VIO and VSLAM that use sequences of images, keyframe based bundle adjustment, and full bundle adjustment and loop closure. This is enabled by a novel adversarial training and visual–inertial sensor fusion technique embedded in our end-to-end trainable deep visual–inertial architecture. Even when IMU data are not provided, SelfVIO with RGB data outperforms deep monocular approaches in the same domain. In future work, we plan to develop a stereo version of SelfVIO that could utilize the disparity map.

Code availability

All code was implemented in Python using the deep learning framework Tensorflow. We provide the code, trained models, and scripts to reproduce the experiments of this paper at https://github.com/yasinalm/SelfVIO. All source code is provided under the MIT license.

Acknowledgments

This work is supported in part by NIST grant 70NANB17H185 and UKRI EP/S030832/1 ACE-OPS. M.T. thanks TUBITAK for the

2232 International Outstanding Researcher Fellowship and ULAK-BIM for High Performance and Grid Computing Center (TRUBA resources). Y.A. would like to thank the Ministry of National Education in Turkey for their funding and support.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., et al. (2016). TensorFlow: A system for large-scale machine learning. In *Proceedings of the 12th USENIX conference on operating systems design and implementation* (pp. 265–283). USA: USENIX Association.

Aleotti, F., Tosi, F., Poggi, M., & Mattoccia, S. (2019). Generative adversarial networks for unsupervised monocular depth prediction. In *Computer vision – ECCV 2018 workshops* (pp. 337–354). Cham: Springer International Publishing, http://dx.doi.org/10.1007/978-3-030-11009-3_20.

Almalioglu, Y., Ozyoruk, K. B., Gokce, A., Incetan, K., Gokceler, G. I., Simsek, M. A., et al. (2020). EndoL2H: deep super-resolution for capsule endoscopy. *IEEE Transactions on Medical Imaging*, *39*(12), 4297–4309. http://dx.doi.org/10.1109/TMI.2020.3016744.

Almalioglu, Y., Santamaria-Navarro, A., Morrell, B., & Agha-Mohammadi, A.-A. (2021). Unsupervised deep persistent monocular visual odometry and depth estimation in extreme environments. In *2021 IEEE/RSJ international conference on intelligent robots and systems (IROS)* (pp. 3534–3541). http://dx.doi.org/10.1109/IROS51168.2021.9636555.

Almalioglu, Y., Saputra, M. R. U., d Gusmão, P. P. B., Markham, A., & Trigoni, N. (2019). GANVO: Unsupervised deep monocular visual odometry and depth estimation with generative adversarial networks. In *2019 international conference on robotics and automation (ICRA)* (pp. 5474–5480). http://dx.doi.org/10.1109/ICRA.2019.8793512.

Almalioglu, Y., Turan, M., Lu, C. X., Trigoni, N., & Markham, A. (2021). Milli-RIO: Ego-motion estimation with low-cost millimetre-wave radar. *IEEE Sensors Journal*, *21*(3), 3314–3323. http://dx.doi.org/10.1109/JSEN.2020.3023243.

Artetxe, M., Labaka, G., & Agirre, E. (2018). Unsupervised statistical machine translation. In *Proceedings of the 2018 conference on empirical methods in natural language processing* (pp. 3632–3642). Brussels, Belgium: Association for Computational Linguistics, http://dx.doi.org/10.18653/v1/D18-1399.

Asvadi, A., Garrote, L., Premebida, C., Peixoto, P., & J. Nunes, U. (2018). Multimodal vehicle detection: Fusing 3D-LIDAR and color camera data. *Pattern Recognition Letters*, *115*, 20–29. http://dx.doi.org/10.1016/j.patrec.2017.09.038.

Bloesch, M., Burri, M., Omari, S., Hutter, M., & Siegwart, R. (2017). Iterated extended Kalman filter based visual-inertial odometry using direct photometric feedback. *International Journal of Robotics Research*, *36*(10), 1053–1072. http://dx.doi.org/10.1177/0278364917728574.

Bousmalis, K., Silberman, N., Dohan, D., Erhan, D., & Krishnan, D. (2017). Unsupervised pixel-level domain adaptation with generative adversarial networks. In *2017 IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 95–104). http://dx.doi.org/10.1109/CVPR.2017.18.

Burri, M., Nikolic, J., Gohl, P., Schneider, T., Rehder, J., Omari, S., et al. (2016). The EuRoC micro aerial vehicle datasets. *International Journal of Robotics Research*, *35*(10), 1157–1163. http://dx.doi.org/10.1177/0278364915620033.

Chen, C., Rosa, S., Miao, Y., Lu, C. X., Wu, W., Markham, A., et al. (2019). Selective sensor fusion for neural visual-inertial odometry. In *2019 IEEE/CVF conference on computer vision and pattern recognition (CVPR)* (pp. 10534–10543). http://dx.doi.org/10.1109/CVPR.2019.01079.

Clark, R., Bloesch, M., Czarnowski, J., Leutenegger, S., & Davison, A. J. (2018). Learning to solve nonlinear least squares for monocular stereo. In *Computer vision – ECCV 2018* (pp. 291–306). Cham: Springer International Publishing, http://dx.doi.org/10.1007/978-3-030-01237-3_18.

Clark, R., Wang, S., Wen, H., Markham, A., & Trigoni, N. (2017). VINet: Visual-inertial odometry as a sequence-to-sequence learning problem. In *Proceedings of the thirty-first AAAI conference on artificial intelligence* (pp. 3995–4001). San Francisco, California, USA: AAAI Press.

Concha, A., Loianno, G., Kumar, V., & Civera, J. (2016). Visual-inertial direct SLAM. In *2016 IEEE international conference on robotics and automation (ICRA)* (pp. 1331–1338). IEEE, http://dx.doi.org/10.1109/ICRA.2016.7487266.

Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., et al. (2016). The cityscapes dataset for semantic urban scene understanding. In *2016 IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 3213–3223). http://dx.doi.org/10.1109/CVPR.2016.350.

Costante, G., & Mancini, M. (2020). Uncertainty estimation for data-driven visual odometry. *IEEE Transactions on Robotics*, *36*(6), 1738–1757. http://dx.doi.org/10.1109/TRO.2020.3001674.

Delmerico, J., & Scaramuzza, D. (2018). A benchmark comparison of monocular visual-inertial odometry algorithms for flying robots. In *2018 IEEE international conference on robotics and automation (ICRA)* (pp. 2502–2509). IEEE, http://dx.doi.org/10.1109/ICRA.2018.8460664.

Dosovitskiy, A., & Brox, T. (2016). Generating images with perceptual similarity metrics based on deep networks. In *Proceedings of the 30th international conference on neural information processing systems* (pp. 658–666). Red Hook, NY, USA: Curran Associates Inc..

Eigen, D., Puhrsch, C., & Fergus, R. (2014). Depth map prediction from a single image using a multi-scale deep network. *Advances in Neural Information Processing Systems*, *27*, 2366–2374.

Engel, J., Schöps, T., & Cremers, D. (2014). LSD-SLAM: Large-scale direct monocular SLAM. In *Computer vision – ECCV 2014* (pp. 834–849). Cham: Springer, Springer International Publishing, http://dx.doi.org/10.1007/978-3-319-10605-2_54.

Engel, J., Stückler, J., & Cremers, D. (2015). Large-scale direct SLAM with stereo cameras. In *2015 IEEE/RSJ international conference on intelligent robots and systems (IROS)* (pp. 1935–1942). IEEE, http://dx.doi.org/10.1109/IROS.2015.7353631.

Eustice, R. M., Singh, H., & Leonard, J. J. (2006). Exactly sparse delayed-state filters for view-based SLAM. *IEEE Transactions on Robotics*, *22*(6), 1100–1114. http://dx.doi.org/10.1109/TRO.2006.886264.

Faessler, M., Fontana, F., Forster, C., Mueggler, E., Pizzoli, M., & Scaramuzza, D. (2016). Autonomous, vision-based flight and live dense 3D mapping with a quadrotor micro aerial vehicle. *Journal of Field Robotics*, *33*(4), 431–450. http://dx.doi.org/10.1002/rob.21581.

Forster, C., Carlone, L., Dellaert, F., & Scaramuzza, D. (2015). IMU preintegration on manifold for efficient visual-inertial maximum-a-posteriori estimation. In *Robotics: Science and systems (RSS)* (pp. 1–20). Rome, Italy: http://dx.doi.org/10.15607/RSS.2015.XI.006.

Forster, C., Zhang, Z., Gassner, M., Werlberger, M., & Scaramuzza, D. (2017). SVO: Semidirect visual odometry for monocular and multicamera systems. *IEEE Transactions on Robotics*, *33*(2), 249–265. http://dx.doi.org/10.1109/TRO.2016.2623335.

Fortun, D., Bouthemy, P., & Kervrann, C. (2015). Optical flow modeling and computation: A survey. *Computer Vision and Image Understanding*, *134*, 1–21. http://dx.doi.org/10.1016/j.cviu.2015.02.008.

Fraundorfer, F., & Scaramuzza, D. (2012). Visual odometry : Part II: Matching, robustness, optimization, and applications. *IEEE Robotics & Automation Magazine*, *19*(2), 78–90. http://dx.doi.org/10.1109/MRA.2012.2182810.

Furgale, P., Rehder, J., & Siegwart, R. (2013). Unified temporal and spatial calibration for multi-sensor systems. In *2013 IEEE/RSJ international conference on intelligent robots and systems* (pp. 1280–1286). IEEE, http://dx.doi.org/10.1109/IROS.2013.6696514.

Gao, Q., Liu, J., & Ju, Z. (2020). Robust real-time hand detection and localization for space human–robot interaction based on deep learning. *Neurocomputing*, *390*, 198–206. http://dx.doi.org/10.1016/j.neucom.2019.02.066.

García, J., Molina, J. M., & Trincado, J. (2020). Real evaluation for designing sensor fusion in UAV platforms. *Information Fusion*, *63*, 136–152. http://dx.doi.org/10.1016/j.inffus.2020.06.003.

Garg, R., Kumar, B. V., Carneiro, G., & Reid, I. (2016). Unsupervised CNN for single view depth estimation: Geometry to the rescue. In *Computer vision – ECCV 2016* (pp. 740–756). Cham: Springer, Springer International Publishing, http://dx.doi.org/10.1007/978-3-319-46484-8_45.

Gatys, L., Ecker, A. S., & Bethge, M. (2015). Texture synthesis using convolutional neural networks. *Advances in Neural Information Processing Systems*, *28*, 262–270.

Geiger, A., Lenz, P., Stiller, C., & Urtasun, R. (2013). Vision meets robotics: The KITTI dataset. *International Journal of Robotics Research*, *32*(11), 1231–1237. http://dx.doi.org/10.1177/0278364913491297.

Geiger, A., Lenz, P., & Urtasun, R. (2012). Are we ready for autonomous driving? The KITTI vision benchmark suite. In *2012 IEEE conference on computer vision and pattern recognition* (pp. 3354–3361). IEEE, http://dx.doi.org/10.1109/CVPR.2012.6248074.

Godard, C., Aodha, O. M., & Brostow, G. J. (2017). Unsupervised monocular depth estimation with left-right consistency. In *2017 IEEE conference on computer vision and pattern recognition (CVPR), no. 6* (pp. 6602–6611). http://dx.doi.org/10.1109/CVPR.2017.699.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., et al. (2014). Generative adversarial nets. *Advances in Neural Information Processing Systems*, *27*, 2672–2680.

Guizilini, V., Ambrus, R., Pillai, S., Raventos, A., & Gaidon, A. 3D packing for self-supervised monocular depth estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 2485–2494).

He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2020). Mask R-CNN. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *42*(2), 386–397. http://dx.doi.org/10.1109/TPAMI.2018.2844175.

Hinton, G. E., & Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *Science*, *313*(5786), 504–507. http://dx.doi.org/10.1126/science.1127647.

Huang, L., Fu, Q., He, M., Jiang, D., & Hao, Z. (2021). Detection algorithm of safety helmet wearing based on deep learning. *Concurrency Computations: Practice and Experience*, *33*(13), Article e6234. http://dx.doi.org/10.1002/cpe.6234.

İncetan, K., Celik, I. O., Obeid, A., Gokceler, G. I., Ozyoruk, K. B., Almalioglu, Y., et al. (2021). VR-caps: A virtual environment for capsule endoscopy. *Medical Image Analysis*, *70*, Article 101990. http://dx.doi.org/10.1016/j.media.2021.101990.

Isola, P., Zhu, J., Zhou, T., & Efros, A. A. (2017). Image-to-image translation with conditional adversarial networks. In *2017 IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 5967–5976). http://dx.doi.org/10.1109/CVPR.2017.632.

Jaderberg, M., Simonyan, K., Zisserman, A., & Kavukcuoglu, K. (2015). Spatial transformer networks. *Advances in Neural Information Processing Systems*, *28*, 2017–2025.

Jiang, M.-x., Deng, C., Shan, J.-s., Wang, Y.-y., Jia, Y.-j., & Sun, X. (2019). Hierarchical multi-modal fusion FCN with attention model for RGB-D tracking. *Information Fusion*, *50*, 1–8. http://dx.doi.org/10.1016/j.inffus.2018.09.014.

Johnson, J., Alahi, A., & Fei-Fei, L. (2016). Perceptual losses for real-time style transfer and super-resolution. In *Computer vision – ECCV 2016* (pp. 694–711). Cham: Springer, Springer International Publishing, http://dx.doi.org/10.1007/978-3-319-46475-6_43.

Jones, E. S., & Soatto, S. (2011). Visual-inertial navigation, mapping and localization: A scalable real-time causal approach. *International Journal of Robotics Research*, *30*(4), 407–430. http://dx.doi.org/10.1177/0278364910388963.

Kendall, A., Grimes, M., & Cipolla, R. (2015). PoseNet: A convolutional network for real-time 6-DOF camera relocalization. In *Proceedings of the IEEE international conference on computer vision* (pp. 2938–2946).

Kingma, D., & Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980. arXiv:1412.6980.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. *Communications of the ACM*, *60*(6), 84–90. http://dx.doi.org/10.1145/3065386.

Kumar, A. C., Bhandarkar, S. M., & Prasad, M. (2018). Monocular depth prediction using generative adversarial networks. In *2018 IEEE/CVF conference on computer vision and pattern recognition workshops (CVPRW)* (pp. 413–4138). http://dx.doi.org/10.1109/CVPRW.2018.00068.

Laina, I., Rupprecht, C., Belagiannis, V., Tombari, F., & Navab, N. (2016). Deeper depth prediction with fully convolutional residual networks. In *2016 fourth international conference on 3d vision (3DV)* (pp. 239–248). IEEE, http://dx.doi.org/10.1109/3DV.2016.32.

Larsen, A. B. L., Sønderby, S. K., Larochelle, H., & Winther, O. (2016). Autoencoding beyond pixels using a learned similarity metric. In *International conference on machine learning* (pp. 1558–1566). PMLR.

Leutenegger, S., Lynen, S., Bosse, M., Siegwart, R., & Furgale, P. (2015). Keyframe-based visual–inertial odometry using nonlinear optimization. *International Journal of Robotics Research*, *34*(3), 314–334. http://dx.doi.org/10.1177/0278364914554813.

Li, J., Besada, J. A., Bernardos, A. M., Tarrío, P., & Casar, J. R. (2017). A novel system for object pose estimation using fused vision and inertial data. *Information Fusion*, *33*, 15–28. http://dx.doi.org/10.1016/j.inffus.2016.04.006.

Li, M., & Mourikis, A. I. (2013). High-precision, consistent EKF-based visual-inertial odometry. *International Journal of Robotics Research*, *32*(6), 690–711. http://dx.doi.org/10.1177/0278364913481251.

Li, B., Shen, C., Dai, Y., van den Hengel, A., & He, M. (2015). Depth and surface normal estimation from monocular images using regression on deep features and hierarchical CRFs. In *2015 IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 1119–1127). http://dx.doi.org/10.1109/CVPR.2015.7298715.

Li, C., & Wand, M. (2016). Precomputed real-time texture synthesis with Markovian generative adversarial networks. In *Computer vision – ECCV 2016* (pp. 702–716). Cham: Springer, Springer International Publishing, http://dx.doi.org/10.1007/978-3-319-46487-9_43.

Li, R., Wang, S., Long, Z., & Gu, D. (2018). UnDeepVO: monocular visual odometry through unsupervised deep learning. In *2018 IEEE international conference on robotics and automation (ICRA)* (pp. 7286–7291). IEEE, http://dx.doi.org/10.1109/ICRA.2018.8461251.

Liu, F., Shen, C., Lin, G., & Reid, I. (2016). Learning depth from single monocular images using deep convolutional neural fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *38*(10), 2024–2039. http://dx.doi.org/10.1109/TPAMI.2015.2505283.

Long, J., Shelhamer, E., & Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In *2015 IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 3431–3440). http://dx.doi.org/10.1109/CVPR.2015.7298965.

Lundquist, C., & Schön, T. B. (2011). Joint ego-motion and road geometry estimation. *Information Fusion*, *12*(4), 253–263. http://dx.doi.org/10.1016/j.inffus.2010.06.007.

Luo, W., Xing, J., Milan, A., Zhang, X., Liu, W., & Kim, T.-K. (2021). Multiple object tracking: A literature review. *Artificial Intelligence*, *293*, Article 103448. http://dx.doi.org/10.1016/j.artint.2020.103448.

Lupton, T., & Sukkarieh, S. (2012). Visual-inertial-aided navigation for high-dynamic motion in built environments without initial conditions. *IEEE Transactions on Robotics*, *28*(1), 61–76. http://dx.doi.org/10.1109/TRO.2011.2170332.

Lynen, S., Achtelik, M. W., Weiss, S., Chli, M., & Siegwart, R. (2013). A robust and modular multi-sensor fusion approach applied to MAV navigation. In *2013 IEEE/RSJ international conference on intelligent robots and systems* (pp. 3923–3929). IEEE, http://dx.doi.org/10.1109/IROS.2013.6696917.

Mahjourian, R., Wicke, M., & Angelova, A. (2018). Unsupervised learning of depth and ego-motion from monocular video using 3D geometric constraints. In *2018 IEEE/CVF conference on computer vision and pattern recognition* (pp. 5667–5675). http://dx.doi.org/10.1109/CVPR.2018.00594.

Meister, S., Hur, J., & Roth, S. (2018). UnFlow: Unsupervised learning of optical flow with a bidirectional census loss. In *Proceedings of the AAAI conference on artificial intelligence, Vol. 32* (1).

Mourikis, A. I., & Roumeliotis, S. I. (2007). A multi-state constraint Kalman filter for vision-aided inertial navigation. In *Proceedings 2007 IEEE international conference on robotics and automation* (pp. 3565–3572). IEEE, http://dx.doi.org/10.1109/ROBOT.2007.364024.

Muller, P., & Savakis, A. (2017). Flowdometry: an optical flow and deep learning based approach to visual odometry. In *2017 IEEE winter conference on applications of computer vision (WACV)* (pp. 624–631). IEEE, http://dx.doi.org/10.1109/WACV.2017.75.

Mur-Artal, R., Montiel, J. M. M., & Tardós, J. D. (2015). ORB-SLAM: A versatile and accurate monocular SLAM system. *IEEE Transactions on Robotics*, *31*(5), 1147–1163. http://dx.doi.org/10.1109/TRO.2015.2463671.

Mur-Artal, R., & Tardós, J. D. (2017a). ORB-SLAM2: An open-source SLAM system for monocular, stereo, and RGB-D cameras. *IEEE Transactions on Robotics*, *33*(5), 1255–1262. http://dx.doi.org/10.1109/TRO.2017.2705103.

Mur-Artal, R., & Tardós, J. D. (2017b). Visual-inertial monocular SLAM with map reuse. *IEEE Robotics and Automation Letters*, *2*(2), 796–803. http://dx.doi.org/10.1109/LRA.2017.2653359.

Ozyoruk, K. B., Gokceler, G. I., Bobrow, T. L., Coskun, G., Incetan, K., Almalioglu, Y., et al. (2021). EndoSLAM dataset and an unsupervised monocular visual odometry and depth estimation approach for endoscopic videos. *Medical Image Analysis*, *71*, Article 102058. http://dx.doi.org/10.1016/j.media.2021.102058.

Pilzer, A., Xu, D., Puscas, M., Ricci, E., & Sebe, N. (2018). Unsupervised adversarial depth estimation using cycled generative networks. In *2018 international conference on 3d vision (3DV)* (pp. 587–595). IEEE, http://dx.doi.org/10.1109/3DV.2018.00073.

Qin, T., Li, P., & Shen, S. (2018). VINS-Mono: A robust and versatile monocular visual-inertial state estimator. *IEEE Transactions on Robotics*, *34*(4), 1004–1020. http://dx.doi.org/10.1109/TRO.2018.2853729.

Qin, T., & Shen, S. (2018). Online temporal calibration for monocular visual-inertial systems. In *2018 IEEE/RSJ international conference on intelligent robots and systems (IROS)* (pp. 3662–3669). IEEE, http://dx.doi.org/10.1109/IROS.2018.8593603.

Radford, A., Metz, L., & Chintala, S. (2016). Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv:1511.06434 [cs]. arXiv:1511.06434.

Rajan, K., & Saffiotti, A. (2017). Towards a science of integrated AI and robotics. *Artificial Intelligence*, *247*, 1–9. http://dx.doi.org/10.1016/j.artint.2017.03.003.

Ranjan, A., Jampani, V., Balles, L., Kim, K., Sun, D., Wulff, J., et al. (2019). Competitive collaboration: Joint unsupervised learning of depth, camera motion, optical flow and motion segmentation. In *2019 IEEE/CVF conference on computer vision and pattern recognition (CVPR)* (pp. 12232–12241). http://dx.doi.org/10.1109/CVPR.2019.01252.

Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention – MICCAI 2015* (pp. 234–241). Cham: Springer, Springer International Publishing, http://dx.doi.org/10.1007/978-3-319-24574-4_28.

Saputra, M. R. U., de Gusmao, P. P. B., Lu, C. X., Almalioglu, Y., Rosa, S., Chen, C., et al. (2020). DeepTIO: A deep thermal-inertial odometry with visual hallucination. *IEEE Robotics and Automation Letters*, *5*(2), 1672–1679. http://dx.doi.org/10.1109/LRA.2020.2969170.

Shamwell, E. J., Leung, S., & Nothwang, W. D. (2018). Vision-aided absolute trajectory estimation using an unsupervised deep network with online error correction. In *2018 IEEE/RSJ international conference on intelligent robots and systems (IROS)* (pp. 2524–2531). IEEE, http://dx.doi.org/10.1109/IROS.2018.8593573.

Shamwell, E. J., Lindgren, K., Leung, S., & Nothwang, W. D. (2020). Unsupervised deep visual-inertial odometry with online error correction for RGB-D imagery. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *42*(10), 2478–2493. http://dx.doi.org/10.1109/TPAMI.2019.2909895.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, *15*(1), 1929–1958.

Turan, M., Almalioglu, Y., Araujo, H., Konukoglu, E., & Sitti, M. (2018). Deep EndoVO: A recurrent convolutional neural network (RCNN) based visual odometry approach for endoscopic capsule robots. *Neurocomputing*, *275*, 1861–1870. http://dx.doi.org/10.1016/j.neucom.2017.10.014.

Turan, M., Almalioglu, Y., Gilbert, H. B., Mahmood, F., Durr, N. J., Araujo, H., et al. (2019). Learning to navigate endoscopic capsule robots. *IEEE Robotics and Automation Letters*, *4*(3), 3075–3082. http://dx.doi.org/10.1109/LRA.2019.2924846.

Turan, M., Almalioglu, Y., Gilbert, H. B., Sari, A. E., Soylu, U., & Sitti, M. (2018). Endo-VMFuseNet: A deep visual-magnetic sensor fusion approach for endoscopic capsule robots. In *2018 IEEE international conference on robotics and automation (ICRA)* (pp. 5386–5392). http://dx.doi.org/10.1109/ICRA.2018.8461129.

Umeyama, S. (1991). Least-squares estimation of transformation parameters between two point patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *13*(4), 376–380. http://dx.doi.org/10.1109/34.88573.

Ummenhofer, B., Zhou, H., Uhrig, J., Mayer, N., Ilg, E., Dosovitskiy, A., et al. (2017). DeMoN: Depth and motion network for learning monocular stereo. In *2017 IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 5622–5631). http://dx.doi.org/10.1109/CVPR.2017.596.

Usenko, V., Engel, J., Stückler, J., & Cremers, D. (2016). Direct visual-inertial odometry with stereo cameras. In *2016 IEEE international conference on robotics and automation (ICRA)* (pp. 1885–1892). IEEE, http://dx.doi.org/10.1109/ICRA.2016.7487335.

Vankadari, M., Kumar, S., Majumder, A., & Das, K. (2019). Unsupervised learning of monocular depth and ego-motion using conditional patchgans. In *Proceedings of the twenty-eighth international joint conference on artificial intelligence* (pp. 5677–5684). Macao, China: International Joint Conferences on Artificial Intelligence Organization, http://dx.doi.org/10.24963/ijcai.2019/787.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., et al. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, *30*, 5998–6008.

Wang, S., Clark, R., Wen, H., & Trigoni, N. (2017). DeepVO: towards end-to-end visual odometry with deep recurrent convolutional neural networks. In *2017 IEEE international conference on robotics and automation (ICRA)* (pp. 2043–2050). IEEE, http://dx.doi.org/10.1109/ICRA.2017.7989236.

Wang, X., Fouhey, D. F., & Gupta, A. (2015). Designing deep networks for surface normal estimation. In *2015 IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 539–547). http://dx.doi.org/10.1109/CVPR.2015.7298652.

Weiss, S., Achtelik, M. W., Lynen, S., Chli, M., & Siegwart, R. (2012). Real-time onboard visual-inertial state estimation and self-calibration of MAVs in unknown environments. In *2012 IEEE international conference on robotics and automation* (pp. 957–964). IEEE, http://dx.doi.org/10.1109/ICRA.2012.6225147.

Wood, A. T. A. (1994). Simulation of the von Mises Fisher distribution. *Communications in Statistics. Simulation and Computation*, *23*(1), 157–164. http://dx.doi.org/10.1080/03610919408813161.

Wu, Z., Wu, X., Zhang, X., Wang, S., & Ju, L. (2019). Spatial correspondence with generative adversarial network: learning depth from monocular videos. In *2019 IEEE/CVF international conference on computer vision (ICCV)* (pp. 7493–7503). http://dx.doi.org/10.1109/ICCV.2019.00759.

Wulff, J., & Black, M. J. (2019). Temporal interpolation as an unsupervised pretraining task for optical flow estimation. In *Pattern recognition* (pp. 567–582). Cham: Springer, Springer International Publishing, http://dx.doi.org/10.1007/978-3-030-12939-2_39.

Xie, Y., Xu, Z., Zhang, J., Wang, Z., & Ji, S. (2022). Self-supervised learning of graph neural networks: A unified review. arXiv:2102.10757 [cs]. arXiv:2102.10757.

Yang, Y., Geneva, P., Eckenhoff, K., & Huang, G. (2019). Degenerate motion analysis for aided INS with online spatial and temporal sensor calibration. *IEEE Robotics and Automation Letters*, *4*(2), 2070–2077. http://dx.doi.org/10.1109/LRA.2019.2893803.

Yin, Z., & Shi, J. (2018). GeoNet: unsupervised learning of dense depth, optical flow and camera pose. In *2018 IEEE/CVF conference on computer vision and pattern recognition* (pp. 1983–1992). http://dx.doi.org/10.1109/CVPR.2018.00212.

Yu, J. J., Harley, A. W., & Derpanis, K. G. (2016). Back to basics: Unsupervised learning of optical flow via brightness constancy and motion smoothness. In *Computer vision – ECCV 2016 workshops* (pp. 3–10). Cham: Springer, Springer International Publishing, http://dx.doi.org/10.1007/978-3-319-49409-8_1.

Zhan, H., Garg, R., Weerasekera, C. S., Li, K., Agarwal, H., & Reid, I. M. (2018). Unsupervised learning of monocular depth estimation and visual odometry with deep feature reconstruction. In *2018 IEEE/CVF conference on computer vision and pattern recognition* (pp. 340–349). http://dx.doi.org/10.1109/CVPR.2018.00043.

Zhou, T., Brown, M., Snavely, N., & Lowe, D. G. (2017). Unsupervised learning of depth and ego-motion from video. In *2017 IEEE conference on computer vision and pattern recognition (CVPR), no. 6* (pp. 6612–6619). http://dx.doi.org/10.1109/CVPR.2017.700.

Zhou, T., Tulsiani, S., Sun, W., Malik, J., & Efros, A. A. (2016). View synthesis by appearance flow. In *Computer vision – ECCV 2016* (pp. 286–301). Cham: Springer, Springer International Publishing, http://dx.doi.org/10.1007/978-3-319-46493-0_18.

Zhu, J.-Y., Krähenbühl, P., Shechtman, E., & Efros, A. A. (2016). Generative visual manipulation on the natural image manifold. In *Computer vision – ECCV 2016* (pp. 597–613). Cham: Springer, Springer International Publishing, http://dx.doi.org/10.1007/978-3-319-46454-1_36.

Zou, Y., Luo, Z., & Huang, J.-B. (2018). DF-Net: Unsupervised joint learning of depth and flow using cross-task consistency. In *Computer vision – ECCV 2018* (pp. 38–55). Cham: Springer International Publishing, http://dx.doi.org/10.1007/978-3-030-01228-1_3.