IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received June 10, 2019, accepted July 6, 2019, date of publication July 16, 2019, date of current version August 6, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2929133

A Survey on Odometry for Autonomous Navigation Systems

SHERIF A. S. MOHAMED[1], MOHAMMAD-HASHEM HAGHBAYAN[1], TOMI WESTERLUND[1], JUKKA HEIKKONEN[1], HANNU TENHUNEN[1,2], AND JUHA PLOSILA[1], (Member, IEEE)

[1]Department of Future Technologies, University of Turku (UTU), 20500 Turku, Finland
[2]Department of Electronic Systems, KTH Royal Institute of Technology, 11428 Stockholm, Sweden

Corresponding author: Sherif A. S. Mohamed (samoha@utu.fi)

This work was supported in part by the Academy of Finland-Funded Research Project under Grant 314048.

ABSTRACT The development of a navigation system is one of the major challenges in building a fully autonomous platform. Full autonomy requires a dependable navigation capability not only in a perfect situation with clear GPS signals but also in situations, where the GPS is unreliable. Therefore, self-contained odometry systems have attracted much attention recently. This paper provides a general and comprehensive overview of the state of the art in the field of self-contained, i.e., GPS denied odometry systems, and identifies the out-coming challenges that demand further research in future. Self-contained odometry methods are categorized into five main types, i.e., wheel, inertial, laser, radar, and visual, where such categorization is based on the type of the sensor data being used for the odometry. Most of the research in the field is focused on analyzing the sensor data exhaustively or partially to extract the vehicle pose. Different combinations and fusions of sensor data in a tightly/loosely coupled manner and with filtering or optimizing fusion method have been investigated. We analyze the advantages and weaknesses of each approach in terms of different evaluation metrics, such as performance, response time, energy efficiency, and accuracy, which can be a useful guideline for researchers and engineers in the field. In the end, some future research challenges in the field are discussed.

INDEX TERMS Self-contained localization, wheel odometry, inertial odoemtry, laser odometry, visual-inertial odometry, filter-based, optimization-based, loosely-coupled, tightly-coupled, GPS-denied.

I. INTRODUCTION

One of the most important challenges that have been raised recently in the field of autonomous system applications is *self-localization*, i.e., to self-allocate the position and orientation of a vehicle/vessel over time. For autonomous navigation, obstacle avoidance and object tracking, a platform must continuously preserve information of its position and pose. The traditional localization technique that has been widely employed in autonomous platforms is the Global Positioning System (GPS). It is a global satellite system that uses radio signals to determine the position and speed of mobile platforms with global coverage. It was developed by the US military in 1973 in order to accurately estimate the position of an intercontinental ballistic missile (ICBM) [1]. In the early 1980s, the GPS became accessible for civilians at a different new carrier frequency [2]. It can provide positioning information, with an accuracy of a few meters, at any time and any point around the Earth [3] and can be used for self-localization [4], [5]. However, the GPS suffers from some problems that make it less reliable to be used for precise self-localization, such as satellite coverage fluctuation, multipath effects, latency, and inaccuracy.

Although advanced GPS systems can at best provide accurate positioning within a few centimeters, it is still not reliable enough for a core navigation system of autonomous platforms, especially for localization of aquatic and aerial vehicles [6]. The strength of a satellite signal varies depending on the place and environment conditions. For example, in forests the signal strength is weaker than urban areas. Moreover, GPS is not suitable for indoor navigation, since radio signals are affected by walls and other objects. All these factors disturb the process of acquisition and tracking of GPS signals at receivers, making self-localization less reliable [7], [8]. *Multipath reception*, where GPS signals arrive at a receiver from more than one satellite [9], [10] or via

The associate editor coordinating the review of this manuscript and approving it for publication was Seung-Hyun Kong.

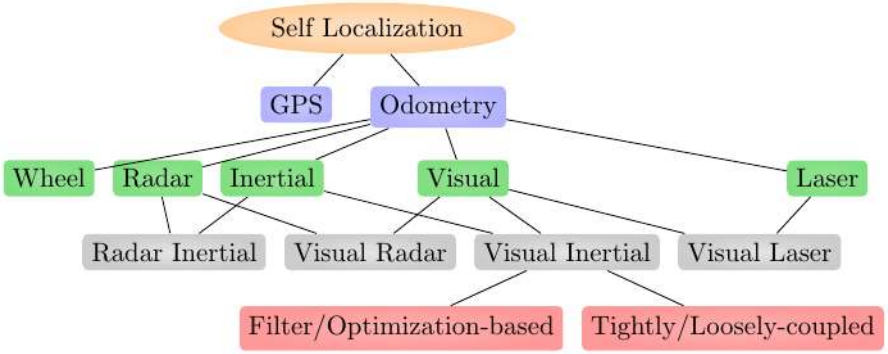

FIGURE 1. General categorization of the localization strategies proposed in literature.

multiple reflective surfaces, is another well-known problem of the GPS. Moreover, the atmospheric condition of the Earth affects the amount of time it takes for a GPS signal to travel from satellites to a device, causing varying *delays* in signal reception [11]. On top of these problems, the GPS can only provide information about the linear velocity of a vehicle; precise localization of a vehicle requires information on both linear and angular velocities. This is especially important for aquatic and aerial vehicles that need such information in three dimensions, in contrast with terrestrial vehicles with two-dimensional navigation. Even though some studies have suggested that autonomous navigation can benefit from using the GPS to perform a set of smart navigation features, such as holding a position and returning to home, a GPS-based navigation system is not a sufficiently reliable or accurate service to be used alone for high-precision self-localization.

Recently, many studies have emerged on *self-contained odometry methods* and simultaneous localization and mapping (SLAM) as a popular example [12]–[14]. Such techniques enable the position and orientation of a vehicle to be calculated based on data obtained from onboard sensors. Unlike the GPS, the proposed techniques do not rely on external assistance of satellite radio signals that are often inconsistent and too noisy for precise localization. Instead, they rely on *odometry* which uses local sensory data to determine the position and orientation of the platform relative to a given starting point.

Usually the SLAM techniques apply an odometry algorithm to obtain the pose of the moving platform where later fed into a global map optimization algorithm, i.e., *loop closure*, to reduce the drift [15] as much as possible based on the history of the pose, i.e., *map*. In other words, SLAM, uses the history of the pose as a global map and when the robot returns to a previously visited area, SLAM techniques reduce the accumulated error caused by odometry by using such history. It should be noted that odometry techniques use local optimization methods, e.g., windowed bundle adjustment, to optimize a part of the map, i.e., *local map*, over the last poses and this results in *local map consistency*. SLAM, on the other hand, is concerned to maintain the *global map consistency* and odometry algorithm can be considered as the first phase of the SLAM that is followed by next steps such as loop closing and global optimization.

Several approaches have been proposed for odometry, however there is a lack of a general survey sorting the accomplished research into appropriate categories and providing a comprehensive overview of the techniques applied in this field of study. In [16], the authors present a brief survey on only camera-based odometry for *resource-constrained* platforms (e.g. micro-aerial vehicles, or MAVs, with very limited processing, memory, and battery resources), focusing on the number and type of cameras mounted on the platforms. In [15], [17], and [18], the authors separately describe some aspects of the vision-based odometry such as the basics, history, and comparison of different proposed techniques in the state-of-the-art.

The aforementioned surveys do not cover all aspects of odometry in a comprehensive, well-categorized, and all-in-one manner which would provide researchers and developers with a valuable resource for comparing different existing solutions. This paper aims at answering the need for such a survey, focusing on a comprehensive categorization of recently proposed self-localization approaches. Figure 1 shows an overall categorization of the self-localization methods discussed in this paper. These include GPS and five basic odometry approaches for GPS-denied navigation, i.e., *wheel*, *inertial*, *radar*, *visual*, and *laser* odometries. A combination of different sensors, i.e., multisensor data fusion, is commonly used in object detection and odometry methods to improve the accuracy and robustness of the system [19], [20]. For example, combining inertial and visual odometries leads to a new type of approach called *visual-inertial* odometry. Correspondingly, a combination of visual and laser, or radar odometries results in *visual-laser* and *visual-radar* odometry. Visual-inertial odometry approaches can also be studied from two specific aspects, i.e., whether they are *filter/optimization* based or *tightly/loosely* coupled. The former aspect is about the main method for data preprocessing, while the latter aspect addresses in which stage data fusion of camera and inertial measurements can be applied. The different odometry

approaches mentioned in Figure 1 are analyzed in the subsequent sections in more detail.

II. WHEEL ODOMETRY

Wheel odometry (WO) is one of the simplest forms of self-contained localization that has been used in many skid-steering robots, such as two- and four-wheel robots. In these vehicles, the right-side and left-side wheels can be operated independently at different speeds and directions. They have had many applications; NASA's Mars Exploration Rovers (MER) are prominent examples of such robotic vehicles [21]. The wheel odometry method is based on wheel encoders that are mounted on a robot to track the number of revolutions each wheel has made. The number of revolutions is integrated into a dynamic model to determine the robot's current position relative to the starting point [22]. Wheel odometry approach suffers from several limitations. For example, it can be applied only to ground vehicles, and not to aerial or aquatic ones. Moreover, it suffers from a *position drift* phenomenon wherein the error in the measurements accumulates over time. Also, wheel odometry systems perform poorly on complex uneven terrains and slippery surfaces due to wheel slippage. Even though wheel odometry is a simple and inexpensive localization technique, it is not suitable for controlling platforms that require a precise and long-term reliable localization system.

III. INERTIAL ODOMETRY

Inertial odometry (IO), or an *inertial navigation system* (INS), is a localization method that uses the measurements from the IMU sensor to determine the position, orientation, altitude, and linear velocity of a vehicle/robot, relative to a given starting point. An IMU sensor is a micro-electro-mechanical system (MEMS) device that mainly consists of a 3-axis accelerometer and a 3-axis gyroscope. The accelerometer measures non-gravitational acceleration whereas the gyroscope measures orientation based on measurement of gravity and magnetism. The small size and low power consumption of these MEMS-based sensors have made them an ideal solution for resource-constrained systems, such as drones and micro-robots. Moreover, navigation systems based on IMUs do not require an external reference to accurately estimate the position of a platform. However, these systems suffer from a drifting issue due to errors originated from different sources e.g., constant errors in gyroscope measurements and accelerometers. These errors, later, lead to an increasing error in the estimated velocity and position [23]. Therefore, inertial odometry systems are inaccurate and unsuitable for applications that require localization for long periods of time. To tackle this problem, different solutions have been proposed. In [24], for instance, a probabilistic approach based on double-integration rotated acceleration using the extended Kalman filter framework (EKF) is presented. Even with such improvements, inertial odometry is not capable enough to be used as the primary navigation method for autonomous

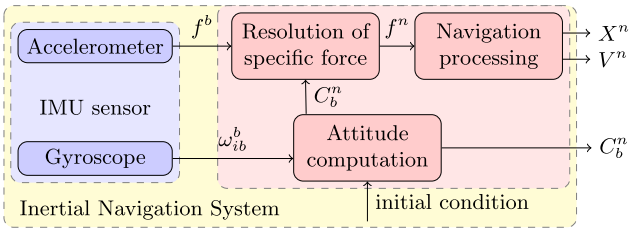

FIGURE 2. The general block diagram of inertial navigation system.

vehicles in GPS-denied environments. Figure 2 illustrates the structure of the inertial navigation system in which the measurements obtained from the IMU sensor are integrated using the dead reckoning method to estimate the current pose.

In Figure 2, f^b and w_{ib}^b denote the linear force and the true angular velocity in the body frame of the IMU sensor, measured by the accelerometer and gyroscope, respectively. The estimated altitude, i.e., roll, pitch, and yaw, of the rigid body, C_b^n, is computed by the Altitude Computation unit. In the Resolution unit the linear force is multiplied by cosine matrix C_b^n to obtain the reference vector f^n in the inertial axes. The navigation processing unit uses the reference vector, f^n in Figure, to compute the position and the velocity of the platform, denoted by X^n and V^n, respectively.

IV. LASER ODOMETRY

Laser odometry (LO), or LiDAR odometry, is an approach for estimating the position and orientation of a platform by tracking laser speckle patterns reflected from surrounding objects. LiDARs are insensitive to ambient lighting and low-texture environments [25]. LiDAR sensors have become smaller and lightweight compared with older versions, thus they can be attached even to a micro aerial vehicle (MAV) [26]. In general, the LiDAR-based sensing process consists of two main parts: laser emission and optical observation. In the laser emission part, coherent and spatial light is emitted from the laser device to the surrounding environment. In the optical observation part, the radiated laser light on an object reflects *laser speckles* on the 2D *observation plane* that is the plane for monitoring the laser reflections based on optical detectors. A 3D image is reconstructed by contrasting different consecutive 2D images.

When the LiDAR scanning rate is higher than the extrinsic motion, the standard iterative closest point (ICP) method [27] is often used to compute a moving object's velocity to address the motion distortion introduced by a single-axis 3D LiDAR [28]. ICP is a general and standard 3D reconstruction algorithm in which the correspondences between the cloud points of two scans are computed iteratively, calculating the transformation function which minimizes the distance between corresponding points. Algorithm 1 shows a formal specification of the ICP method. The inputs of the algorithm are two observed point clouds in two consecutive LiDAR sweeps. Based on the inputs, the initial transformation function is calculated using the common singular-value decomposition (SVD) method [29], and then it gets

Algorithm 1 Standard ICP Algorithm

Inputs: Two point clouds $A = \{a_k\}, B = \{b_k\}$
Outputs: Transformation function $T = \{R, t\}$
Constant: Threshold value d_{max}

1: $T \leftarrow T_{init}$
2: **while** $Error \leq d_{max}$ **do**
3: $\quad M \leftarrow FindClosestPoint(A, T.B)$
4: $\quad Error = \frac{1}{N}\sum_{k=1}^{N} \|m_k - T.b_k\|^2$
5: $\quad T \leftarrow \arg\min \frac{1}{N}\sum_{k=1}^{N} \|m_k - T.b_k\|^2$
6: **end while**

optimized by only considering rational values. A threshold d_{max} is used to avoid violating the assumption of full overlap. However, this threshold introduces a trade-off between convergence and accuracy, as a low threshold value results in a bad convergence. On the other hand, a large threshold value causes incorrect correspondences, which leads to low accuracy results [30].

Another way to reconstruct a 3D surface is by applying the point-to-plane variant of ICP [31] which leverages the advantage of surface normal information by minimizing the sum of the squared distance between a point and its tangent plane for each correspondence to improve performance, i.e., robustness and accuracy. Moreover, Segal *et al.* [30] propose a generalized ICP (GICP) framework by combining the standard ICP and point-to-plane ICP algorithms into a single framework to increase accuracy. In the GICP framework, all measured points are assumed to be drawn from the Gaussian center at the true point, and a maximum likelihood estimation (MLE) is used to iteratively estimate transformation for aligning the scans.

On the other hand, because LiDAR scanning can be relatively slow, other sensors, such as cameras and IMUs [32]–[34], are often used to carry out the velocity measurements. Another approach is to use 2-axis LiDAR scanning without any aid from other sensors by utilizing laser intensity returns to create visual images and match features among images to recover motion. In [26], the authors propose an effective point cloud registration method based on detecting and extracting edges and planar points. This method requires a lower cloud density compared with a method proposed by Anderson and Barfoot [35], where features are extracted from intensity images. For 3D LiDARs (e.g. *Velodyne*), conventional approaches, such as ICP, and feature-based approaches fail to precisely register the point clouds because of vertical sparsity and ring structure issues. To address these problems, the authors in [36] propose a method to efficiently align and register point clouds of a 3D LiDAR using collar line segments (CLS). In this method, point clouds are transformed into line clouds using random generation. Moreover, line clouds are accurately registered using an algorithm based on the LiDAR odometry and mapping (LOAM) method [26].

The main drawback of LiDAR odometry is that it is difficult to implement on a resource-constrained platform, because it applies iterative optical matching among points of two sets, which requires fairly demanding computations [17]. Moreover, getting an accurate scan and correcting the motion distortion from a object, e.g., glass, is very challenging, leading to poor performance [37].

V. RADAR ODOMETRY

Radar odometry (RO) is a technique to estimate the relative motion of a platform by analyzing scans obtained by the onboard radar sensor. Radar, short for radio detection and ranging, is a sensor that uses radio waves to determine the velocity, range and angle of surrounding objects. It is available in two forms: pulse and continuous-wave (CW) radar. A pulse radar system emits short and powerful pulses and receives the echo signals in the silent periods. CW radar, i.e., frequency modulated-continuous wave (FMCW) radar, transmits a steady stream of linearly modulated CW signals. The key difference between the two is that a CW radar sensor, with its continuous signals, is capable of generating high resolution images from the reflected signals, while pulse radar typically suffers from a blind spot on front of the sensor (up 50 meters). CW radar has attracted a lot of attention in the fields of localization and object avoidance due to its beneficial characteristics, such as a low sampling rate, low power consumption, and, more importantly, its minimum target range. To maintain a physically small-size antenna, a combination of FMCW and synthetic aperture radar (SAR) is used to generate the two-dimensional image with high resolution [38], [39].

Radar is a long-range active sensor that is immune against poor weather conditions and can operate in low-texture environments. For these reasons, various approaches have been proposed in the literature to estimate the ego-motion of ground and aerial vehicles based solely on radar measurements. Generally, most of the RO approaches can be split into two main steps, namely feature extraction and tracking.

The first step is to extract important features from the radar scans. In [40], the amplitude gridmap accumulated from the radar scan is transformed into a grayscale image and then interesting points are detected using feature extraction techniques, e.g, SIFT. In [38], the authors use a range-compressed image to estimate the motion of an unmanned aerial aircraft. The Hough transform [41] is used to detect strong scatters by searching for hyperbolas in the image. The main drawback of this method is that it demands a lot of computational power. In [39], they extended their work by using a method called *thresholding* to identify scatters from noise. In [42], the authors propose a method to detect strong and stable scatters in two steps: range-bearing estimation and constant false alarm rate (CFAR) detection. They extract the bearing angle from range-compressed signals generated by two channels by subtracting their phase components. To remove clutters from the received signals and reduce the computational burden, the ordered statistic CFAR is used. Another way is to extract

interesting areas in radar gridmaps using methods such as DBSCAN [43] and MSER [44]. In [45], the authors use a 1D signal (i.e. the power-range spectra) to extract a set of landmarks in radar scans. They first estimate the noise statistics and then scale the power value at each range to extract strong scatters.

The second step in RO is the process of tracking scattered points in radar data that is related to the same observed object in different times. In [39], a recursive-RANSAC algorithm is used to track point scatters from range-compressed images of radar. In [46], the authors propose an algorithm based on a feature descriptor in which extracted features are tracked using the binary annular statistic descriptor (BASD) and Hamming distance. In [47], a scan matching method is deployed to track features. The extracted features are aligned in order to minimize some cost function using matching algorithms such as ICP. In [45], the authors propose an algorithm to perform data association using a feature descriptor (i.e. unray) and relationships between features. Unlike ICP, this approach does not rely heavily on a good initial estimate.

RO measurements are affected by outliers and non-flat terrain. In [39], [48], an outlier rejection scheme is used to remove outliers and improve the RO solution. To overcome the non-flat terrain problem, radar measurements can be also fused with other sensors, such as the IMU [42] and RGB camera [49] to help overcome radar limitations.

VI. VISUAL ODOMETRY

Visual odometry (VO) estimates the position and orientation of a platform by analyzing the variations induced by the motion of a camera on a sequence of images. An example of early research in this field is NASA's Mars exploration program, where visual odometry has been used for estimating the position of rovers in rough terrains [50], [51]. Taking a broader view, VO can be seen as part of structure from motion (SfM) which is a general technique for reconstructing a 3D scene and camera pose using a set of images [15]. SfM can be performed in various ways based on, e.g., the number of cameras that have been employed, the camera calibration status, and the order of images. A 3D scene is reconstructed by computing the OF from key information extracted from two consecutive image frames. The key information (e.g. corners) is extracted using an *image feature detector*, such as *Moravec* [52] and *Harris* [53] corner detectors. The reconstructed scene can be refined using bundle adjustment [54] or another offline optimization method.

There are three different standard techniques to calculate the transformation matrix between two sequential images from two sets of correspondences based on the specification of the point correspondence in two or three dimensions [55].

3D-3D correspondences: In this case, the camera motion *(transformation)* can be computed from two sets of correspondences specified in three dimensions. First, by capturing two stereo image pairs, extract and match feature points between them. Second, triangulate the 3D matched points for each stereo pair. The transformation is computed with an absolute scale by minimizing the L2 distance between the two 3D point sets [15].

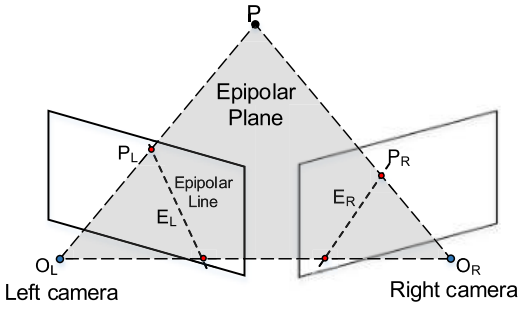

FIGURE 3. Illustration of the epipolar geometry, and epipolar constraint.

2D-2D correspondences: In this method the transformation matrix is calculated using the essential matrix [15]. The essential matrix defines the geometric relationship between two sequential images and it is computed from the 2D feature correspondences using the epipolar constraint. A simple and common approach to compute the essential matrix is by using the Nister five-point algorithm [56]. In this method, a set of five corresponding points are used to determine the relative scale between consecutive frames. Another way to calculate the essential matrix is by using the eight-point algorithm presented by Fischler and Firschein [57]. The major issue in this approach is that it computes the transformation matrix up to an unknown scale factor.

3D-2D correspondences: The main concept of this method is to compute the transformation matrix by minimizing the 2D reprojection error from 2D and 3D correspondences, as shown in Eq. 4:

$$T_t^k = \arg\min_{T_t^k} \sum_i \left\| p_k^i - P_{t-1}^i \right\|^2 \tag{1}$$

where T_{t-1}^t is the transformation matrix from $t-1$ to t, the image measurements are denoted as p^t, and P_{t-1}^i is the reprojection of the 3D features X_{t-1}^i into image I^t. This problem is also known as the *perspective-n-points* (PnP), which estimates the pose of a camera using a set of N number of 3D points. The minimal solution to recover the camera pose requires three 3D-2D correspondences, which is known as the perspective-3-point (P3P) [58]. Motion estimation based on 3D-2D provides better accuracy than 3D-3D due to minimizing the reprojection error of an image instead of the 3D-3D feature position error [56].

VO techniques, as shown in Figure 4, can be categorized based on the key information, position of the camera, and type/number of the camera. The key information, upon which odometry is performed, can be direct raw measurements, i.e., pixels [59]–[66], or indirect image features such as corners and edges [67], [67]–[74], or combination of them, i.e., hybrid information [6], [75], [76]. The camera type/number can be monocular [77]–[79], stereo [80]–[82], RGB-D [83], omnidirectional [6], [64], [84], fisheye [85], [86], or event-based [87]–[90]. The camera pose, in turn,

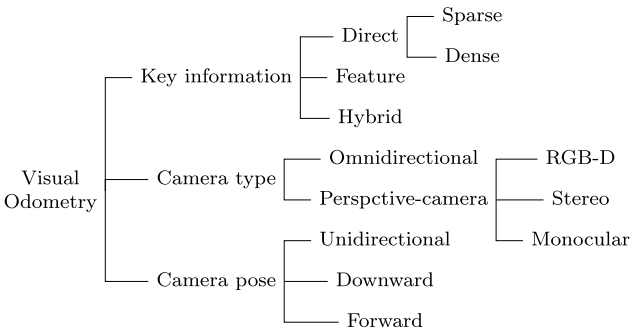

FIGURE 4. General classification of studies in the field of visual odometry.

can be either forward-facing, downward-facing, or hybrid. In the rest of this section, the mentioned VO techniques are elaborated in more detail.

A. KEY INFORMATION

1) DIRECT APPROACHES

In direct approaches, raw visual measurements in terms of *pixels* are used to estimate the position of a vehicle. The change in the appearance of the captured images, i.e., the intensity of image pixels, is analyzed to estimate the pose [91]. The input of the algorithm consists of consecutive images from the camera(s). Based on the captured images, an *optical flow* (OF) algorithm is used to determine the changes among frames. It uses pixel intensity to compute the 2D displacement vector which shows the movement of points between two consecutive frames. In visual odometry, OF algorithms are classified into dense and sparse schemes. In dense OF, all pixels are optimized using various techniques based on a global smoothness assumption [92]. An example of a dense OF algorithm is the Horn-Shunck [93] method which calculates the displacement of each pixel in a frame by solving the brightness constancy that is formulated as follows:

$$BC = \iint [(I_x u + I_y v + I_t)^2 + \alpha^2(\|\nabla u\|^2 + \|\nabla v\|^2)]dxdy \tag{2}$$

where (u, v) are the smoothness constraints and I is the image. On the other hand, sparse algorithms, e.g., Lucas-Kanade [94], exploit the assumption that the flow in an image is locally smooth. Thus, sparse OFs only process some pixels from the whole image by solving the brightness constancy equation based on a template matching technique. For instance, a window of 3×3 pixels around the point gives 9 equations per pixel with two unknowns, formulated as:

$$\begin{bmatrix} I_x(p1) & I_y(p1) \\ I_x(p2) & I_y(p2) \\ \vdots & \vdots \\ I_x(p9) & I_y(p9) \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} I_t(p1) \\ I_t(p2) \\ \vdots \\ I_t(p9) \end{bmatrix} \tag{3}$$

where $\{p1 \ldots p9\}$ represent the nine pixels in the 3×3 window.

However, this solution is sub-optimal, because the number of equations is larger than the number of unknowns. Usually, a least square criterion is applied to simplify these 9 equations into only two equations as follow:

$$\begin{bmatrix} \sum I_x I_x & \sum I_x I_y \\ \sum I_x I_y & \sum I_y I_y \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} \sum I_x I_t \\ \sum I_y I_t \end{bmatrix} \tag{4}$$

2) FEATURE-BASED APPROACHES

Feature-based or *indirect* approaches extract points of interest in each captured image using feature detectors, such as corner or edge detectors. Corners are one of the most unique keypoints as they show a two dimensional intensity change, and therefore these keypoints are well distinguished from the neighboring points [95]. Consequently, several proposed methods are based on corners, for instance, Harris detector [53], SIFT [96], SURF [97], FAST [98], and ORB [99]). Edges in images are areas with strong intensity contrasts. The majority of edge detectors are based on gradient or Laplacian [100]. The Laplacian edge detector, e.g. the Marr–Hildreth algorithm, [101] uses one kernel to search for the zero crossings in the second derivative of the image. Unlike the Laplacian detector, the gradient edge detector, such as the Canny edge detector [102] uses two kernels to detects the edges by looking for the maximum and minimum in the first derivative of the image.

Figure 5 illustrates the general block diagram of this approach. The consecutive images are pre-processed using different feature detection and matching techniques to generate an intermediate representation, i.e., point correspondences. Subsequently, an optimization process is performed by minimizing *geometric error* to calculate the transformation matrix. One of the major advantages of the feature-based method is that it is robust against geometric distortions and brightness inconsistencies [103]. However, it discards valuable information from the captured image by extracting only strong interest points. Moreover, feature extraction and matching processes require lots of computational resources and consequently consumes much energy that is proportional to the number of extracted features. Therefore, only a few

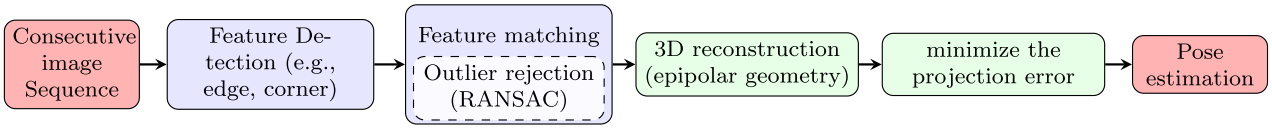

FIGURE 5. The general pipeline of feature-based visual odometry.

features can be maintained in the system to make the approach more feasible for resource-constrained applications, such as drones [104].

3) HYBRID APPROACHES

Feature-based approaches are not robust in low-texture environments, because only a few features can be detected and tracked. On the other hand, direct methods exploit all key information in the images including weak intensity variations, which leads to more robust and efficient results in such environments. However, direct approaches are computationally more demanding. A combination of the direct and feature-based methods, i.e., a *hybrid* approach, is used to tackle these issues. For example, Scaramuzza and Siegwart [6] propose a hybrid VO method to estimate the pose of a ground vehicle. The displacement is estimated using a feature-based method, whereas the orientation is efficiently determined based on a direct method. Similarly, Feng *et al.* present a localization system based on direct and indirect methods [75]. Their algorithm consists of two modules: the feature-based module is used to estimate the pose if there are enough features in the frame; on the other hand, the direct module is used in low-texture environments. In [105], a semi-direct visual odometry (SVO) method has been presented to eliminate the need for costly feature extraction at every frame. This approach uses subpixel feature correspondence to increase accuracy, and performs feature extraction only on selected keyframes. In contrast, Niccola *et al.* [76] propose a hybrid method that combines a feature-based method with semi-dense direct image alignment. In this approach, a direct method is only performed on keyframes, while a feature-based method is performed on frames in between, and the output is used as a prior for direct methods tracking.

B. CAMERA TYPE/NUMBER

Visual odometry can be classified based on the type and number of cameras used in the algorithm. There are six different types of cameras that are mostly used in visual odometry approaches: stereo, monocular, RGB-D, omnidirectional, fisheye, and event-based cameras.

1) STEREO

In a stereo camera setup, platforms are equipped with multiple cameras to easily reconstruct the 3D information from stereo image pairs. The pose can be accurately obtained by extracting and tracking key information between two pairs of stereo images and then applying a motion estimation algorithm, e.g., a maximum likelihood motion algorithm [106]. Figure 3 presents the main elements of stereo odometry. In the figure, a point **P**, that is viewed from two fully calibrated and aligned cameras with their respective centers of projection points O_L and O_R, can be reconstructed with an absolute scale using a triangulation technique and two sets of point correspondences. The main idea here is to search corresponding features in two aligned images along the same 1D epipolar line. Such a technique, also called *epipolar geometry*, reduces the search time by narrowing down the feature search/matching domain from 2D images to 1D epipolar lines. One of the main disadvantages of using stereo cameras for localization is that they require precise extrinsic calibration to provide accurate results. Due to varying conditions, e.g. shocks, vibration, etc., such extrinsic calibration degrades over time and periodic re-calibration is necessary for effective pose estimation. Moreover, stereo cameras typically have a fixed *baseline distance*, i.e., the distance between the two cameras, which affects the accuracy of the depth estimation in different scenarios. In outdoor environments, to efficiently estimate the depth of far objects, a large baseline distance is needed. However, due to the size limitation of the platforms such as cars and UAVs, it is hard to have two cameras with a large baseline distance. On the other hand, to obtain the depth of very close objects, the baseline needs to be ultra-short.

2) MONOCULAR

Monocular setups estimate the position and orientation by analyzing consecutive images from a single camera [78]. Unlike in a stereo camera setup, monocular visual odomtery does not suffer from the baseline issue, and that is why monocular VO has attracted much attention in recent years. However, it requires at least three different frames to reconstruct the 3D information [107]. Moreover, one of the main disadvantages of monocular-based approaches is that the translation vector is computed up to a relative scale, since the transformation *(orientation and translation)* between the first two frames is not fully known. Therefore, the distance between first two camera poses is set to a predefined value [15]. One way of solving this problem is by obtaining additional information about the initial transformation using other sensors, such as IMU and LiDAR [77]. Another way is by relying only on the visual information captured by the camera. For example, in [108] the authors have proposed an algorithm to tackle the scale problem in translation by using vision data, the mounting point of the camera, and the planarity of the road surface. This method is able to continuously resolve the ambiguity in the scale and reduce the scale drift. Furthermore, in [109], a scale recovery algorithm has been proposed using only the vision data from a monocular camera using a deep convolutional neural network (CNN) algorithm. The advantage of this method is that it can recover the scale and eliminate the scale drift from the structure of the whole environment rather from a fixed reference plane as in [108]. However, it is infeasible for real applications due to heavy computational needs of CNN.

Algorithm 2 presents an overview of the method proposed by Scaramuzza and Fraundorfer [15] to estimate motion of a monocular camera based on 2D-2D correspondences. Generally, the algorithm finds a transformation matrix $T_t = \{R_t, t_t\}$ that minimizes the reprojection error of the matched points in two consecutive frames. Features are extracted from current and previous images and then matched to generate a set of correspondence points. The generated correspondences are

Algorithm 2 Monocular Motion Estimation Based on 2D-2D

Inputs: Image sequence $(I_{t-1}, I_t, \dots)$
Outputs: Transformation matrix T

1: $(kp1, kp2) \leftarrow keyPoint(I_{t-1}, I_t)$
2: $M_i \leftarrow match(kp1, kp2)$
3: $E \leftarrow SVD(M_i)$
4: $R_t, t_t \leftarrow decompose(E)$
5: $T_t \leftarrow T_{t-1}T'_t$
6: Repeat from 1

used to compute an essential matrix E utilizing the SVD technique. A minimum of 5 correspondence points are needed to compute the essential matrix. E is decomposed into a rotational matrix R_t and a translation vector t_t. After this, the relative scale is determined for updating the translation vector, and the transformation matrix is computed [108].

3) RGB-D

RGB-D cameras are an optimal solution to provide information about the real depth, compared with the stereo and monocular setups discussed above. A stereo camera scheme performs a costly *epipolar line* search to obtain the depth information, and if stereo cameras are not aligned, an additional warping process is needed to align the epipolar lines of both cameras horizontally. On the other hand, a monocular setup cannot get the depth information about the surroundings in a real scale [109]. Most RGB-D visual odometry approaches utilize feature-based methods which provide more robustness [83], [110]. Also direct-based RGB-D VO approaches have been presented to accurately obtain the pose in low texture environments and to avoid the consumption of computing resources in the feature detection and matching processes [111]. Moreover, in [112], a hybrid scheme using an RGB-D camera has been proposed to leverage the robustness of feature-based methods and consistency of direct methods in low texture environments.

4) OMNIDIRECTIONAL

An omnidirectional camera, also known as 360-camera, is a camera with a 360° *field of view* (FoV) in azimuth and 90° to 140° in elevation [113]. It can include a fisheye lens or a catadioptric optical system [114]. In addition, it can achieve more accurate pose estimation compared with traditional cameras with a small FoV, because it can capture more information from the environment [115]. Moreover, it overcomes the inherent problem of the rotation-translation ambiguity of small FoV cameras [116]. In [6], the authors present a VO system based on a single omnidirectional camera to obtain the position and orientation of vehicles. The system consists of two modules: homography-based and direct-based module. The first module uses a feature-based method to detect and track features from the ground plane. The second module uses a direct method to estimate the rotation of the vehicle. In [79], [117], a monocular SLAM algorithm based on an omnidirectional camera is proposed, which leads to a localization system that is more robust to rotation-only movements.

5) FISHEYE

Cameras with a fisheye lens produce a wide panoramic image with almost 180° from side-to-side. Fisheye cameras help to observe more features in the environments compared with pinhole cameras. However, it can produce some distortions. Therefore, a special distortion model is required to correct the radial distortion [85]. In addition, pixel correspondences in fisheye-stereo cameras lie on epipolar curves. Therefore, traditional disparity search algorithms, e.g., semi-global matching (SGM) [118] cannot be used for fisheye stereo matching. Moreover, disparity matching algorithms for fisheye cameras requires more computational power, because epipolar curves are more expensive to compute. In [86], the authors propose a visual odometry algorithm based on a stereo fisheye camera. A semi-dense direct method is used for image alignment. The epipolar curve distortion induced by the fisheye cameras is tackled by using the plane-sweeping stereo algorithm. A total number of 128 plane hypotheses is used to estimate the depth for every pixel in the image. The plane hypothesis which gives the highest similarity score is used to obtain the depth map. The similarity score is based on *zero-mean normalized cross-correlation* (ZMCC). The main drawback of the plane-sweeping stereo algorithm in the direct framework is that it demands a lot of computational power. In [85], the authors presented a visual odometry algorithm based on a semi-direct method. The plane-sweeping algorithm is applied to a set of extracted features instead of all pixels in the image. Thus, the computational complexity is reduced significantly.

6) EVENT-BASED

Event cameras are bio-inspired vision sensors, e.g, dynamic vision sensors (DVS), that capture changes in intensity asynchronously across all pixels on the camera, also known as *events* [87]. They have outstanding characteristics, such as low latency, high temporal resolution, and high dynamic range (140 dB compared to 60 dB of conventional cameras). In addition, event cameras do not suffer from motion blur because all pixels capture light independently. Therefore, they offer a significant improvement for vision-based localization algorithms, such as VO [88]. In [87], the authors propose an algorithm to compute the optical flow from the event stream using the feature-based method. The algorithm used to extract the features is based on building a polarization map for all pixel in the image. Based on this map, the motion can be easily detected by counting the number of incoming events with expected polarity for each pixel. Kueng *et al.* [88] present a method to estimate the 6-DOF motion using a dynamic and active-pixel vision sensor (DAVIS). Firstly, features are detecting in the grayscale frames and then tracked using the event stream.

C. CAMERA POSE

Based on the position and orientation of the camera, the existing VO localization systems can be divided into three categories: forward-facing [119]–[121], downward-facing [70], [122], and hybrid [74], [123]. A forward-facing setup provides more information, but it is a suboptimal solution for detecting small movements. Moreover, it can be obscured by shadows and surround changes, such as wind and sunlight [124]. On the other hand, localization systems based on downward-facing cameras have been successfully used for positioning in pre-explored environments. However, these systems are inaccurate when vehicles are moving fast, because it is challenging to find good matching points between two consecutive images [122]. Therefore, a hybrid approach, i.e., a combination of the forward-facing and downward-facing camera setups is used to tackle the limitations of each scheme. For instance, Piyathilaka and Munasinghe [123] propose an outdoor localization system based on VO using such a hybrid camera setup for a skid steered robot. The vision data from the downward-facing camera is used to localize the robot at low speeds. At high speeds, on the other hand, the captured images from the downward-facing camera are not used in the VO pipeline, because it is difficult to track features based on them, and hence the data provided by the forward-facing camera is used to localize the vehicle.

Although visual odometry approaches are very successful in localizing platforms in indoor environments, there are still many challenges that need to be tackled to use VO as a precise localization method in outdoor environments. The major challenges of VO-based localization are related to computational complexity, scale ambiguity and image conditions, such as lighting, low-textured regions, and image blurriness [67]. Moreover, it suffers from drifting issues, since it is based on incremental computation of the camera path, leading to gradual accumulation of errors introduced by each new frame over time. To address these challenges, several methods for sensor fusion have been proposed, such visual-laser and visual-inertial odometries.

VII. VISUAL-LASER ODOMETRY

Visual-LiDAR odometry [32]–[34] fuses visual and LiDAR odometry to overcome the limitations of LiDAR, such as motion distortion and non-prominent environments (e.g., *highways*), and visual odometry such as drifting and low-texture environments. In [34], a combination of VISO2 [125] and LOAM in a loosely coupled fashion has been presented. The VISO2 module calculates the transformation between two consecutive LiDAR sweeps to correct the distortion of the laser point cloud. Moreover, the position and orientation of the LiDAR are initialized by combining the results of VISO2 and the last pose calculated by the LOAM module [26]. Thereafter, the optimal state is obtained by extracting shape features from the corrected point cloud and matching them with the point cloud in the map. Unlike the previous approach that uses ICP for 3D data registration which is computationally expensive. Zhuang *et al.* [33] present a bearing angle (BA) model to convert the 3D LiDAR data to a two-dimensional BA image, which is an optimal way for feature extraction and matching. The BA model was originally proposed by Scaramuzza *et al.* [126], who defines the bearing angle as the angle between two cloud points and the laser beams.

VIII. VISUAL-RADAR ODOMETRY

Vision-based localization systems have some challenges, such as the lack of features in the scene, inconsistent feature matching between consecutive frames, and illumination. Moreover, VO methods are not suitable for outdoor applications, since vision sensors are affected by the environmental conditions, such as rain, fog, and snow. One way to overcome these limitations is to combine vision data with measurements from radar, as the radar is immune against these issues. In [49], the authors propose a localization system for UAVs by fusing measurements from five main sensors (i.e., radar, camera, IMU, barometer, and magnetometer) to accurately estimate the forward velocity. All the sensors are fused in a loosely coupled fashion via an extended Kalman filter.

IX. RADAR-INERTIAL ODOMETRY

To achieve accurate motion estimation results, some approaches fuse radar data with IMU measurements in a loosely or tightly coupled manner. In [38], [39], the authors fuse the radar and IMU data in an extended Kalman filter (EKF) to estimate the state of an aircraft. Here IMU measurements are used for statistical prediction, and the range and above-ground level (AGL) estimates are used in the measurement model. In [127], the authors propose a method to combine data from a single radar and measurements from the gyroscope to obtain the forward, sideslip, and angular speeds of a ground platform to overcome challenges in odometry on slippery surfaces.

X. VISUAL-INERTIAL ODOMETRY

Localization methods based on vision are affected significantly by many environmental conditions such as lighting, shadows, blur images, and frame drops. On the other hand, IMU-based methods, although not affected by surroundings, usually deteriorate with time. The limitations from both sides can be overcome by integrating the two methods, resulting in visual-inertial odometry (VIO), which can provide greater accuracy and robustness. As shown in Table 1, VIO can be categorized into two ways, based on how the visual and inertial data are fused: filter-based and optimization-based. Moreover, based on when the measurements are fused it can be categorized into loosely-coupled and tightly-coupled. In addition, there are various camera setups, e.g., monocular, stereo, RGB-D, and omnidirectional cameras; and different methods to extract key information from captured images, such as feature-based, direct, and hybrid approaches.

TABLE 1. Visual-inertial odometry approaches in the state-of-the-art.

Articles	Camera				Key Information			Filter-based	Optimization-based	Fusion	
	Mono	Stereo	RGBD	Omni	Feature	Direct	Hybrid			Loosely	Tightly
[128], [129]		✓					✓	✓			✓
[77], [130]–[135]	✓				✓			✓			✓
[136]–[138]	✓				✓				✓		✓
[139], [140]		✓				✓		✓			✓
[55], [141]–[143]		✓			✓			✓		✓	
[144]				✓	✓			✓			✓
[145]			✓			✓		✓			✓
[146], [147]		✓				✓			✓		✓
[148]	✓						✓		✓		✓
[119], [149]		✓			✓			✓			✓
[83], [150]			✓		✓			✓		✓	
[151]	✓					✓		✓			✓

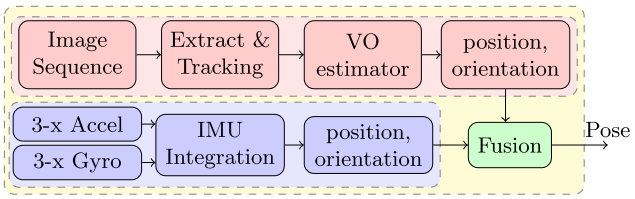

FIGURE 6. A general block diagram of the loosely-coupled framework.

A. LOOSELY-COUPLED APPROACH

In loosely-coupled techniques, position and orientation are determined by blending pose estimation from two standalone subsystems, i.e., a visual odometry module and an IMU module. Estimated data are fused in a delayed stage to refine the position and orientation of a vehicle. Each positioning subsystem is treated as a stand alone pose estimator [152]. One of the main advantages of loosely-coupled framework that it limits the computational complexity by using fixed dimension for the state space. Generally, a common methods to fuse sensor data is the conventional Kalman filter (KF). There are also various methods to fuse sensor data using non-linear optimization techniques that provides better accuracy and robustness [18]. However, these techniques demand more computational power compared with KF, which makes them very challenging to be implemented on resource-constrained systems, such as drones [55]. Loosely-coupled approaches can be categorized into two main branches: 1) wherein pose data, estimated by the VO subsystem, are used as the update step of KF for IMU measurements [55], [141]–[143], and 2) where data obtained by the IMU sensor are integrated as independent measurements into a vision optimizer [152]. Figure 6 shows the general block diagram of the loosely-coupled approach. The VO subsystem consists of two main units: the extraction and tracking unit and the VO estimator unit. Features are extracted and tracked from two consecutive images to produce two sets of correspondences. Then an ego-motion algorithm is applied to obtain the position and orientation. Simultaneously, the IMU subsystem estimates the position and orientation by integrating the measurements from the IMU sensor. The fusion is applied at the last stage of the pipeline to refine the position and orientation estimated by the two subsystems.

In [55], a loosely-coupled stereo VIO, that is implemented based on the error-state kalamn filter (ESKF), has been proposed. A traditional indirect error state estimation method is used to fuse a high rate IMU and relatively low rate visual odometry output, instead of using a direct fusion method which will cause a loss in high dynamic information obtained by IMU sensor. In this approach, the true states are composed of nominal- and error-states. The nominal-state does not take into account the noise. Thus, the accumulated errors will be collected in the error-state and estimated with the ESKF. The error-states are always small and therefore the computation of Jacobian matrix is very easy and fast. Moreover, a *keyframe* concept is used in the VO model to reduce the drift effect. The main advantage of using the keyframe concept is to reduce the system's vulnerability to losing the track scenarios in terms of system stability and performance. The first keyframe is selected during the system initialization phase based on the quality of tracked features. In addition, the feature detection and description are based on ORB rather than using a more robust but slow descriptor such as SIFT and FAST. Basically, ORB is a combination of the FAST feature detector and the BRIEF descriptor [99].

Similarly, in [141], the authors propose a loosely-coupled fusion method based on the indirect feedback KF. The VO subsystem calculates the delta position between two consecutive frames. Moreover, two models are proposed to measure the accumulated delta position in the instant camera coordinate system and in the initial/final time of the accumulation interval. A spurious matches(outliers) can deteriorate the performance of Kalman filter significantly. Hence, a common sigma-check technique is used at the early steps of VO to remove the outliers.

In [142], the authors propose a loosely-coupled filtering framework to integrate noisy measurements from stereo cameras and an IMU sensor to provide more accurate and efficient pose estimation for drones in real-time for indoor-outdoor environment. As the relative measurements depend on both current and past states, an augmented state vector is created by augmenting the current state with copies of previous states. In this approach, the filtering framework is based on an Unscented Kalman filter (UKF) instead of popular Extended kalman filer (EKF)-based framework. UKF is used

to avoid computing the Jacobian matrices, which is proven to be computationally too complex and time consuming for systems like drones. As in most state-of-the-art VO approach, a keyframe-based algorithm is used to avoid temporal drifting. Moreover, the Kanade-Lucas-Tomasi (KLT) [153] feature tracker is used to track features extracted using lightweight corner detector running at a high-rate, e.g., 25 Hz.

In [152], the authors present a loosely coupled approach, in which the motion is estimated based on stereo VO and the absolute gravity is corrected by using an IMU as an inclinometer to obtain the absolute roll and pitch. The framework is based on an EKF and the state of the vehicle is represented by a 7-element vector. A multiscale feature detector, called CenSurE [154], is used to provide more stability in indoor and outdoor environments. Finally, a nonlinear batch optimization based on an incremental sparse bundle adjustment (SBA) is used to reduce the error in the VO subsystem.

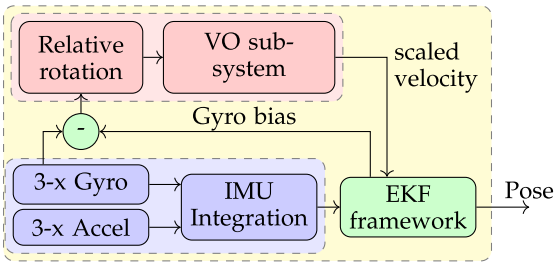

FIGURE 7. System setup for semi-tight coupling of inertial-optical flow based speed recovery presented in [155].

B. SEMI-TIGHTLY COUPLED APPROACH

Semi-tightly coupled approaches fuse the estimated pose provided by the VO subsystem with raw measurements of the IMU sensor to achieve a balanced accuracy and computational complexity. In [155], a semi-tightly coupled navigation system based on a monocular camera has been designed for MAVs. The framework is based on an extended Kalman filter, and consists of two complementary visual modules: a 6-DoF pose estimator and a 3D speed estimator. The 3D speed estimator is used in the initialization phase of the pose estimator. In this method, the state of the filter is composed of 24-element. The setup for the presented semi-tightly coupled framework is shown in Figure 7. The scaled camera velocity, which is used in the EKF update setp, is computed using a first-order quaternion integration to recover the relative rotation between two camera frames. The vision part is based on an eight-point algorithm [156] to reduce dimensionality, in which only eight features with their corresponding OF vectors are used. An off-line calibration of the inter-sensor parameters has a complexity of at least $O(N^2)$, where N is the number of features in an image. Thus, online calibration is proposed using an inertial-optical flow approach to address this issue.

In [157], a semi-tightly coupled approach utilizing an optimization-based framework has been presented. The pre-integrated IMU measurements are fused with the pose measurements from the VO module based on edge alignment. IMU pre-integration is used to avoid the need for determining the global rotation between the world frame and body frame. This is important because the initial altitude is required to be known to determine the global rotation, which is challenging in many applications [158]. In addition, to improve convergence during aggressive motion the incremental rotation is initialized using the gyroscope reading.

C. TIGHTLY-COUPLED APPROACH

Tightly-coupled approaches fuse key information extracted from captured images with raw measurements of the IMU sensor at early stages to achieve better accuracy. Key information can be obtained by extracting and tracking feature points from images using image detector techniques, i.e., corner detectors, or by using pixel intensity of images with OF algorithms. Tightly coupled approaches perform direct and systematic fusion of visual and IMU measurements and usually lead to better results compared with loosely-coupled approaches. This is because the tightly-coupled framework combines the key information for image alignment and the IMU error term into one cost function [159]. Figure 8 shows the general framework for a tightly-coupled VIO solution. Features extracted from the captured images are fused at early stage with the raw measurements from the IMU sensor to obtain more accurate pose estimation. Tightly-coupled approaches can be classified into two general categories, i.e., 1) filter-based [77], [107], [130], [134], [149], [160]–[163] and 2) optimization-based [136], [137], [164] approaches, see Table 1.

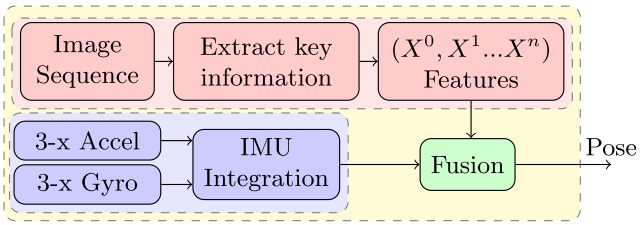

FIGURE 8. General block diagram of tightly-coupled framework.

D. FILTER-BASED APPROACH

Filter-based approaches are among the earliest approaches used to solve VIO and SLAM problems [149]. They consist of two main parts: a prediction step and an update step. Furthermore, they can be viewed as a maximum a posterior (MAP) method, wherein measurements from proprioceptive *(internal state)* sensors, such as IMU sensors, are used to compute the prior distribution of the platform pose, and measurements from exteroceptive *(external state)* sensors, e.g., cameras are used to build the likelihood distribution. In filter-based visual-inertial odometry, the prior distribution *(dynamic model)* of a vehicle is computed by using linear and angular velocities measured from the IMU sensor. This dynamic model is used in the prediction step to predict motion of the vehicle. In addition, key information, such as features,

or pixel intensities extracted from captured images are used as a likelihood distribution *(measurement model)* to update the predictions in the update step. Visual-inertial odometry based on filtering framework can be classified into three categories: extended Kalman filter (EKF) [151], multi-state constraint Kalman filter (MSCKF) [77], [107], [149], and unscented Kalman filter (UKF) [160], [163].

1) EXTENDED KALMAN FILTER

In nonlinear system (like an autonomous vehicle), an approximate nonlinear filtering, such as an EKF, or a particle filter (PF), is used for fusion. An EKF is able to provide an accurate estimate for Gaussian models with limited linearity. On the other hand, a PF is appropriate approach for non-Gaussian and nonlinear systems. However, in the robotics filed, EKFs are used instead of PFs due to their computational efficiency [18]. The EKF framework can be divided into three main steps: state representation, building a measurement model, and finally an update step [165]. Basically, an EKF is a nonlinear version of the general Kalman filter (KF), which performs a first-order linearization around the transition function at each time step. Typically, EKF based VIO determines the position and orientation of a vehicle by determining state propagation from noisy IMU measurements, and correlation from key information that extracted from images captured by a single camera or multiple cameras mounted on vehicle [129]. Although, EKF framework is one of the most popular filtering strategies, it has some disadvantages. First, the EKF is difficult to implement in practice. Second, it is not a very reliable method for highly nonlinear systems [162].

In [151], an EKF-based monocular VIO approach is presented. In this method, the innovation term or measurement residual is composed of pixel intensity errors of images. A robocentric representation approach is employed to improve system consistency by estimating the location of extracted features with respect to current camera position, reducing the effect of nonlinearities significantly [166]. A limited number of features can be integrated into the filter state, and therefore the feature management system is based on heuristic methods and an adaptive Shi-Tomasi corner detector [153] to keep the reliable features only. The proposed system is able to run smoothly with 50 extracted features. However, accuracy drops significantly when less than 20 extracted features are integrated in the filter state. On the other hand, increasing number of tracked features leads to an increase in the complexity of EKF quadratically.

In [129], an EKF-based stereo VIO system based on both point- and line-features is proposed. Line features are used to improve system robustness in low-texture environment, where detection of point features is challenging. In addition, a lightweight filter-based framework is proposed to reduce long-term drift without relying on more complex computational techniques, such as bundle adjustment. The framework is formulated as an EKF update, in which the current sliding window maintained by the filter is relocated to the past keyframe to reduce accumulated drift.

In [139], a VIO approach based on an iterated EKF (iEKF) framework with a fully robocentric formulation and photometric error model has been proposed. The iEKF framework provides simultaneous landmark by iteratively update the per-landmark, and thus provides a full state refinements. In this approach, the local texture of a landmark is taken into account. Therefore, non-corner shaped features, i.e., line features can be extracted and tracked, improving system robustness in challenging scenarios, such as low texture-environments. A fully robocenteric formulation of the states used to reduce observability and nonlinearity issues. The innovation term is derived by projecting the patch into the current image and thus calculating the photometric error for every pixel in a patch. The computational complexity is reduced by applying a QR-decomposition. In addition, an intensity-based scoring technique has been used to select landmarks to be tracked due to the iEKF's limited scalability. The quality score is calculated based on three sub-scores: global quality, local quality, and local visibility.

2) MULTI-STATE CONSTRAINT KALMAN FILTER

The problem of EKFs is that they demand high computational power, and thus they are not suitable for resource-constrained systems, such as drones. An N number of features augmented into the state vector leads to a cubic computational complexity in terms of the number of features: $O(N^3)$ [107]. In contrast, *structureless* methods, such as MSCKF can attain better precision and consistency, due to its less strict probabilistic assumption and delayed linearization [107]. Moreover, the MSCKF framework has complexity linear to the number of landmarks because it marginalizes the 3D feature positions out of the state vector [77]. Therefore, MSCKFs ensure a good compromisation between computational cost and precision.

In [77], an MSCKF-based monocular VIO system is presented. The measurement model exploits the geometric constraints that arise when a static feature is observed in more than two images. Thus, the 3D feature positions are not included in the filter state vector, which leads to computational complexity only linear to the number of extracted features.

One of the main shortcomings of the conventional MSCKF algorithm is that it has incorrect observability properties, which leads to inconsistency in performance. The spurious gain along the direction of unobservable subspaces leads to a large error in estimation. To address this issue, the authors in [130] extended the filter state and noise model from the Euclidean space to Lie group SE(3). Unlike the conventional MSCKF, the observability matrix of the proposed MSCKF-LG is independent of the estimated state. Therefore, the MSCKF-LG algorithm is invariant to the linearization errors, which improves consistency.

In [107], a modified MSCKF algorithm, i.e, MSCKF 2.0 is proposed, which improves consistency and accuracy. In this algorithm, the Jacobian matrix is computed from the first estimates of each state. Moreover, the camera-to-IMU

transformation parameters are included in the filter state vector to ensure correct observability properties. On the other hand, in the conventional MSCKF, these transformation parameters are assumed to be known. One main advantage of the proposed algorithm is that it is capable of estimating the IMU-to-camera parameters online using manual measurements for initialization. Therefore, it can operate in an unknown environment without prior knowledge of the map.

3) UNSCENTED KALMAN FILTER

The UKF is a Bayesian filter which uses a set of sigma points to update the system states. These weighted sigma points are derived from the prior distribution and lie on the covariance contour in the state space. The mean and covariance are determined by propagating weighted sigma points through a nonlinear process [162]. The conventional EKF accomplishes an analytical local linearization. On the other hand, UKFs perform a statistical local linearization, which leads to higher accuracy. Moreover, UKFs perform a third order linearization [161]. Therefore, nonlinear systems based on UKFs show superior performance compared with EKF-based frameworks. Moreover, the UKF avoids computing the Jacobian matrix, which makes it derivative-free [142]. However, one main drawback of using the UKF framework is that it requires more computational power, which makes its implementation in resource-constrained systems difficult.

In [160], the authors have proposed a power-on-and-go localization system based on a UKF framework. The proposed system is able to accurately calibrate the sensor-to-senor transform in the field without relying on a known calibration target. As part of the target-free calibration procedure, an approximate number of 50 features are selected as point landmarks. These features are selected automatically based on their distribution on the first image and the frequency of their appearance in a 10 seconds window. In addition, the scale ambiguity of the monocular SfM is addressed by fixing the directions to three highly distributed features *(anchors)* selected in the first image. Furthermore, a pseudo-measurements techniques, i.e., an unscented quaternion estimator [167] is used to reduce the uncertainty associated with each of the three features.

In [163], the author has utilized an epipolar constraint in a UKF-based framework to estimate the pose of a MAV. The epipolar constraint is deployed because it is easier to accurately track features between two consecutive frames that tacking features over an extended period of time. However, this approach has three main drawbacks. First, the epipolar constraint biases toward the center of tracked features in images. To overcome this issue, the author proposes an algorithm to compute deviations from the epipolar constraint. Second, the scale ambiguity in monocular visual measurements which makes it hard to distinguish between fast camera movements and observing an object that is far away. The author tackles this problem by using an air pressure sensor to measure the airspeed of the MAV. Finally, to minimal the accumulative error, a minimal sampling rate of the image data is used.

In [134], a UKF-based VIO system operating on the Lie group SE(3) is proposed. A Lie group is a group that is a smooth manifold, with the property that the composition and inversion are smooth operations. The main source of nonlinearity is the kinematics of rotation. Generally, the orientation representation is modeled using Euler angles [168] or Quaternions [138]. In this approach, the process model is expressed on the SE(3) space to obtain a unique and global representation of a rigid body pose. The IMU measurements are used as control inputs, while the camera measurements are used during the update step. Using the SE(3) representation causes some specific problems, e.g., tangent spaces in a manifold cannot directly translated. Therefore, the authors have proposed an algorithm based on the concept of parallel transport to move the state covariance on the manifold and handle the measurement update. Moreover, a noise model based on the Lie algebra se(3) is used to keep a minimum representation of the observed noise.

E. OPTIMIZATION-BASED APPROACH

Optimization-based approaches, also known as *smoothing-based* approaches, estimate the pose by jointly optimizing key information extracted from images and inertial measurements from an IMU sensor. Therefore, they outperform filter-based approaches in terms of accuracy [137]. However, carrying out iterative minimization of a least square error function requires more computational resources. These techniques can be divided into three main categories: fixed-lag, full-smoothing, and incremental-smoothing algorithms. Fixed-lag smoothing, or *online optimization*, estimates all states within a given time window and marginalizes old states in order to reduce the computational complexity. However, marginalizing states outside the estimation window leads to inconsistency in performance [136]. Full smoothing, or *batch optimization*, estimates the entire history of the states by solving a large number of linear algebraic equations as a minimization problem [18]. Although batch-optimization frameworks have the highest accuracy compared with other approaches, they become infeasible for real-time applications because the trajectory grows over time. Incremental smoothing leverages the computational cost by identifying and updating only the variables affected by the new measurements [148].

In [137], a fixed-lag optimization-based monocular VIO system has been proposed. The fixed-lag framework only optimizes recent observations and parameters, which leads to a high computational cost reduction. A trade-off between the estimation accuracy and computational cost can be achieved by changing the window size. The proposed framework directly optimizes the noisy inertial measurements and vision data in a single cost function. The cost function consists of the inconsistency in the IMU-to-camera transforms and parameters, such as IMU biases, camera poses, and feature positions. These parameters are incrementally updated as more observations become available. Furthermore, a Harris

corner detector is used with the KLT [169] for feature tracking to reduce the computational cost, which makes the system suitable for real-time applications.

In [164], the authors present a keyframe-based localization algorithm based on batch-optimization framework. The cost function composed of errors of the IMU sensor as well as the 3D landmarks and reprojection errors from stereo cameras. Furthermore, a keyframe paradigm is employed and old states are marginalized to reduce the computational cost. A frame is selected as a keyframe, if the ratio between the area spanned by matched points and the area spanned by all feature points detected in an image is less than 60 percent. Moreover, a customized multi-scale streaming SIMD extension (SSE) optimized Harris corner detector is used to extract features.

One drawback of the batch-optimization framework is that it requires processing of a large amount of data, making it less suitable, even infeasible, for real-time optimization. Moreover, the high rate of inertial measurements increases the number of variables in the optimization which leads to slow operation. To address these issues, in [136], the authors proposed a VIO system based on an incremental-smoothing framework and a preintegration theory. The preintegration technique tackles the high rate of the IMU by combining inertial measurements between two keyframes into a single motion constraint. Often, new measurements have only a local effect on the MAP estimate. Thus, the incremental-smoothing framework leverages the computational complexity by identifying and updating only the variables affected by the new measurements. Moreover, a structureless model is employed to avoid the delay of the vision data processing during incremental smoothing by removing all 3D points from the variables to be estimated.

XI. DISCUSSION AND CONCLUSION

We introduced a comprehensive literature review of self-contained localization approaches in GPS-denied environments. These approaches can be divided into five main categories: wheel odometry, inertial odometry, visual odometry, laser odometry, and radar odometry. Each of the mentioned approaches has some drawbacks that are mainly evaluated under several conditions such as low-texture environment, low lighting, shadow, scale ambiguity, and drifting over time. Therefore, as discussed in the survey, various combinations and fusions of these approaches have been proposed, e.g., visual-laser odometry and visual-inertial odometry.

Wheel odometry is one of the earliest self-contained localization systems, which is used to estimate the position relative to a starting point using wheel encoders. However, wheel odometry suffers from some disadvantages, such as position drift and inaccuracy on uneven terrain and slippery surfaces. Moreover, it can only be used for ground platforms. Inertial odometry tackles these drawbacks by estimating the position and orientation of a vehicle using the measurements from an accelerometer and a gyroscopic sensor. Inertial odometry still suffers from a drifting issue because a constant error in the gyroscope or accelerometer leads to a quadratic error in the velocity and a cubic error growth in the position.

Radar odometry uses an antenna to emit radio signals to measure the velocity and range of objects around the vehicle. The main advantages of a radar system is that it has a wide range coverage and it is immune to environmental conditions, e.g., cloudy weather, and can easily operate at night, which makes it a suitable solution for outdoor applications. However, it can only be used for object detection, since the output resolution is not high enough for object identification. LiDAR, on the other hand, emits laser pulses to detect the objects in the environment. The main advantages of a LiDAR system over a radar system are that LiDAR can detect small objects using a short wavelength and can build an extract 3D monochromatic image of the surrounding objects. However, it has limitations concerning transparent objects (e.g. glass) and challenging weather (e.g. dust, fog, rain, and snow).

Visual odometry estimates the position and orientation by extracting key information from images. The key information can be extracted using direct or indirect techniques. Although visual odometry provides more precise estimation compared with inertial and wheel odometries, it still suffers from some drawbacks. These problems are mainly related to computational complexity and image conditions, such as low lighting, shadow, and low texture environments. Furthermore, there are drifting issues caused by error accumulation as visual odometry is based on relative measurements.

Visual-inertial odometry has been proposed to tackle the shortcomings of inertial and visual odometries. Basically, VIO fuses the visual data captured by single or multiple cameras with the inertial measurements provided by an IMU sensor to determine the position and orientation of a vehicle. The state-of-the-art VIO approaches can be categorized as loosely-coupled and tightly-coupled. A loosely-coupled approach is considered a black box and is usually composed of two standalone pose estimators. The pose obtained by each estimator is refined by fusing them in a delayed stage. One main advantage of loosely-coupled approaches is that they have a fixed dimension state space which bounds the computational load. However, they are suboptimal because the correlation between the internal measurements and vision data is disregarded. On the other hand, tightly-coupled approaches leverage the complementary advantages of IMUs and cameras by jointly fusing their data in an early stage, which leads to more precise estimations. However, this process demands more computational power.

Another way to categorize the VIO approaches is to consider them either filter-based or optimization-based. Filter-based approaches estimate the pose by building a filter on the inertial measurements and key information extracted from captured images. Improved computational efficiency is achieved by delaying the interference processes to the latest stage of the system and marginalizing the past states from the filter. One main drawback of the marginalization is that it leads to consistency issues. Moreover, filter-based approaches produce suboptimal results due to

TABLE 2. Comparison of common localization techniques in the state-of-the-art.

	Real time	Power	accuracy	Energy	Robustness	dimensions
GPS	Soft	low-power	semi-accurate	non-efficient	high	2D
WO	Hard	low-power	non-accurate	efficient	low	2D
IO	Hard	low-power	non-accurate	efficient	high	3D
RO	Hard	low-power	accurate	efficient	high	3D
VO	Firm	high-power	accurate	non-efficient	low	3D
LO	Hard	high-power	accurate	non-efficient	medium	3D
Filter-based VIO	Firm	high-power	accurate	non-efficient	medium	3D
Optimization-based VIO	Soft	high-power	accurate	non-efficient	medium	3D
Loosely-coupled VIO	Firm	high-power	non-accurate	non-efficient	low	3D
Tightly-coupled VIO	Soft	high-power	accurate	non-efficient	medium	3D

linearization errors. The EKF framework is one of the most common filter-based frameworks which mainly consists of a prediction step and an update step. The EKF approach performs a first-order linearization around the current mean and covariance at each time step. Therefore, the EKF is reliable only for systems that have a Gaussian model with limited nonlinearity. Another problem of the EKF-based VIO systems is that their computational complexity increases quadratically with the number of tracked features integrated into the filter state vector. This means in practice that accuracy of the EKF approach is limited. For highly nonlinear systems, to achieve better accuracy, the UKF framework should be used instead of the EKF framework. Furthermore, the UKF approach is suitable for applications that are composed of black box models, since linearization is not required in the propagation of the mean and covariance. In addition, computation of the Jacobian matrices is not needed. However, a major drawback of the UKF is that it demands more computational power than the other filter-based frameworks. This indicates that implementation of the UKF is very challenging in resource-constrained systems, such as drones.

The MSCKF framework provides an alternative way to fuse the visual and inertial measurements. It does this by constraining the measurements through a stochastically cloned pose within a sliding window. The conventional MSCKF suffers from inconsistent state estimation due to the spurious gain along the direction of unobservable subspace. To tackle this issue, various methods have been proposed. For instance, including the camera-to-IMU parameters in filter state vector is a way to ensure the correct observability properties. Another method is by exploiting the filter state and noise from the rigid body motion on the Lie group (SE3) instead of the Euclidean space, which makes the unobservable subspaces invariant to the linearization error.

Optimization-based approaches use nonlinear optimization to directly minimize the errors between the integrated motion obtained from the inertial measurements and camera motion estimated by classic reprojection error minimization. Optimization-based approaches can be categorized into three main types: fixed-lag, full-smoothing, and incremental-smoothing algorithms. Full smoothing outperforms filter-based approaches in terms of accuracy due to its capability of linearizing the current and past states. However, it includes heavy processing and is not therefore well-suited for resource-constrained systems. Computational costs can be reduced, e.g, by using keyframes, sliding window, or incremental smoothing. In fixed-lag approaches, an active window algorithm is used to marginalize the old states, which bounds the computational complexity. However, marginalizing the old states introduces some issues, such as sparsity, inconsistency, and linearization errors. Finally, the incremental-smoothing framework addresses the full-smoothing and fixed-lag issues by leveraging the advantages of the factor graphs to maintain the sparsity level. Computational complexity is reduced by updating only a small subset of variables.

Odometry algorithms can be compared from different perspectives and by defining different evaluation metrics. The evaluation metrics generally are determined based on the expected goals of the algorithm in its use case and limitations of the platform. Each odometry algorithm has its own pros and cons w.r.t. the different evaluation metrics that make the algorithm suitable for a specific use case. An overview of the advantages and disadvantages of the most common self-contained localization methods is shown in Table 2. To illustrate a general comparison among different aspects of the algorithms, we have categorized the algorithms independently based on six evaluation metrics that are depicted in the table and explained as follows:

The first metric is performance. It can be evaluated from different aspects e.g., the order of execution, the amount of data the algorithm needs, and different behaviors of the software w.r.t. the platform. In this survey, to demonstrate a fair and general performance evaluation for different algorithms, we propose three performance categories for an algorithm: hard real-time, firm real-time, and soft real-time, denoting the capability of the algorithm to be used in hard, firm, and soft real-time applications, independently of the platform the algorithm is running on. Hard real-time applications are those which have a strong timing constraint on the worst-case execution time of the tasks. Such applications are very common in mission-critical systems, where failure to meet any deadline might result in loss of life or property. For instance, an autonomous car is a hard real-time system, where running software must not miss any deadline; failure to do so might lead to an accident. Firm real-time applications provide a degree of flexibility for their running tasks to miss some deadlines, as long as the misses are adequately spaced w.r.t.

each other, so that system failures are avoided. However, the performance will degrade, if too many deadlines are missed. For example, unmanned aerial vehicles (UAVs) can use firm real-time algorithms, since the system can survive infrequent task failures. On the other hand, many consecutive deadlines misses could lead to unintended consequences such as crashes. The tasks in soft real-time applications can be executed without any strict deadlines. Such applications are suitable for scenarios, where relatively wide gap can exist between two consecutive executions of odometry.

The second metric is power efficiency for which we propose two categories: high-power demand and low-power demand algorithms. High power demand algorithms require power-hungry sensors and many computational resources to process the sensory data. Low-power algorithms, on the contrary, demand a relatively small amount of computational resources and low power sensors and are suitable for a platform that cannot provide a high level of instantaneous power. Some constraints that directly affect the power resource of a platform are the limited capability of the battery to provide the needed power and the temperature limitations of the sensors and processing units.

The third metric is energy efficiency for which we consider two categories: energy-efficient and non-energy-efficient algorithms. Here energy refers to the amount of odometry energy consumed by the sensors and processing units. In general, an energy-efficient algorithm requires less energy to provide a given service. Energy efficient odometry techniques are crucial for small and light-weight platforms, such as MAVs and small robots, in which only low-capacity batteries can be used. On the other hand, a non-energy-efficient algorithm consumes more energy to provide a given service. Such odometry methods are suitable for platforms such as cars and vessels that can accommodate large-size batteries or fuel systems capable of providing enough power.

The fourth metric is accuracy. We categorize the algorithms into accurate, semi-accurate, and non-accurate algorithms. An accurate method can precisely obtain the position and orientation of a vehicle at any time the system is active. These algorithms are suitable for platforms such as autonomous cars and UAVs which require extremely high-resolution localization within millimeters. Semi-accurate algorithms can provide an accurate pose estimation only for a short period of time, since they are affected by the drift of sensory data over time. Therefore, they are used in applications that require short-term localization, e.g., MAVs which have an average flight time of 15 minutes. Non-accurate algorithms fail to determine the pose accurately during both short- and long-term activity of the system, because they can only obtain the position within a few centimeters and are heavily affected by the sensor drift over time. They are reasonable for applications that do not require precise localization, such as warehouse robots.

The fifth metric is robustness against the lack of illumination and the environmental conditions (e.g., rain, fog, dust, and snow). We identify three levels of robustness, i.e., high, medium, and low. High robust algorithms have an stable outcome under certain amount of noise, for example in different weather conditions, and therefore they are more commonly used in outdoor applications. For example, odometry based on IMU and radar systems is immune to adverse lighting and weather conditions. The outcome of algorithms with a medium level of robustness changes based on the different environmental conditions. However such fluctuation does not result in significant errors and is therefore acceptable. For instance, in GPS systems, atmospheric conditions affect the latency of the received signals from satellites and might negatively affect the localization process, but such errors are tolerable and in some cases the error affects the performance by forcing the system to re-calculate the pose (*re-localization*). Low robust algorithms are typically using sensors, e.g., cameras, which require good lighting conditions, performing poorly in bad weather conditions. Techniques such as RANSAC have been used to eliminate outliers from the estimation process, which may have effect on computational complexity.

Finally, the sixth metric is the dimension of the calculated pose that is the outcome of the algorithm and can be either 2D or 3D. For instance, wheel odometry techniques can estimate the linear displacement of the vehicle by counting the number of revolutions of the wheels, which is 2D. GPS systems are categorized also under the 2D type, since they can only determine the planar motion of the platform, i.e., XYZ. Even though vision sensors can be used to estimate the 3D pose, i.e., translation and orientation of the platform, some methods focus on providing solutions for constrained motion that can be translation [170] or orientation [171]. Such approaches are not suitable for aerial vehicles since those require algorithms that can estimate the 6-DoF,[1] i.e., translation and orientation.

As can be seen in Table 2, except for GPS that needs a connection to a satellite which is slow, all non-visual odometries are hard real-time, since the amount of information to be processed is not large. However, visual odometry, because of the large amount of information the vision sensor provides, is more accurate. Another issue is the energy consumption which has a direct relationship with the amount of data. This means that visual odometry, which demands heavy computations, is not energy efficient compared with the other techniques, such as radar-based odometry. Except for radar-based odometry and GPS, it can be seen that accuracy and robustness have an inverse relationship, i.e., if one is high the other is low. Another fact that can be extracted from the comparison is the relationship between power efficiency, accuracy, and performance. To achieve a more accurate estimation of the pose, all information in the images needs to be fully utilized. For example, increasing the number of extracted features in the model results in a more accurate estimation of the pose. Such processing of the data increases the computational

[1]It should be noted that six degrees of freedom (6-DoF) refers to the freedom of movement of a rigid body in three-dimensional space which is equivalent to 3D.

complexity drastically, which prolongs the response time of the odometry algorithm. This problem can be solved by adding more computational resources, leading to a need for a stronger power supply. This approach is well-suited for large platforms, such as cars and vessels, that can have multiple powerful CPUs and GPUs onboard. On the other hand, in low power algorithms, a less amount of data is being analyzed. This can be achieved by either using sensors that provide low resolution data or processing a small portion of the collected data, potentially affecting the accuracy of the system. These techniques are suitable for resource constrained platforms, e.g., micro aerial vehicles (MAV), which cannot house a powerful CPU or GPU onboard, but can tolerate some degree of inaccuracy.

REFERENCES

[1] P. Srinivas and A. Kumar, "Overview of architecture for GPS-INS integration," in *Proc. Recent Develop. Control, Automat. Power Eng. (RDCAPE)*, Oct. 2017, pp. 433–438.
[2] S. Vatansever and I. Butun, "A broad overview of GPS fundamentals: Now and future," in *Proc. IEEE 7th Annu. Comput. Commun. Workshop Conf. (CCWC)*, Jan. 2017, pp. 1–6.
[3] A. Saha, A. Kumar, and A. K. Sahu, "FPV drone with GPS used for surveillance in remote areas," in *Proc. 3rd Int. Conf. Res. Comput. Intell. Commun. Netw. (ICRCICN)*, Nov. 2017, pp. 62–67.
[4] H. N. Viet, K.-R. Kwon, S.-K. Kwon, E.-J. Lee, S.-H. Lee, and C.-Y. Kim, "Implementation of GPS signal simulation for drone security using MATLAB/simulink," in *Proc. 24th Int. Conf. Electron., Electr. Eng. Comput. (INTERCON)*, Aug. 2017, pp. 1–4.
[5] D. Bender, W. Koch, and D. Cremers, "Map-based drone homing using shortcuts," in *Proc. IEEE Int. Conf. Multisensor Fusion Integr. Intell. Syst. (MFI)*, Nov. 2017, pp. 505–511.
[6] D. Scaramuzza and R. Siegwart, "Appearance-guided monocular omnidirectional visual odometry for outdoor ground vehicles," *IEEE Trans. Robot.*, vol. 24, no. 5, pp. 1015–1026, Oct. 2008.
[7] D. Borio, L. Camoriano, and L. L. Presti, "Impact of GPS acquisition strategy on decision probabilities," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 44, no. 3, pp. 996–1011, Jul. 2008.
[8] M. Mozaffari, A. Broumandan, K. O'Keefe, and G. Lachapelle, "Weak GPS signal acquisition using antenna diversity," in *Proc. Ubiquitous Positioning Indoor Navigat. Location Based Service (UPINLBS)*, 2014, pp. 11–18.
[9] T. Kos, I. Markezic, and J. Pokrajcic, "Effects of multipath reception on GPS positioning performance," in *Proc. ELMAR*, Sep. 2010, pp. 399–402.
[10] S. Miura, S. Hisaka, and S. Kamijo, "GPS multipath detection and rectification using 3D maps," in *Proc. 16th Int. IEEE Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2013, pp. 1528–1534.
[11] L. Bian, "Study on ionospheric delay correction in GPS signal," in *Proc. IEEE 11th Int. Conf. Electron. Meas. Instrum.*, vol. 1, Aug. 2013, pp. 79–83.
[12] M. W. M. G. Dissanayake, P. Newman, S. Clark, H. F. Durrant-Whyte, and M. Csorba, "A solution to the simultaneous localization and map building (SLAM) problem," *IEEE Trans. Robot. Autom.*, vol. 17, no. 3, pp. 229–241, Jun. 2001.
[13] A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse, "MonoSLAM: Real-time single camera SLAM," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 6, pp. 1052–1067, Jun. 2007.
[14] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardós, "ORB-SLAM: A versatile and accurate monocular SLAM system," *IEEE Trans. Robot.*, vol. 31, no. 5, pp. 1147–1163, Oct. 2015.
[15] D. Scaramuzza and F. Fraundorfer, "Visual odometry [tutorial]," *IEEE Robot. Autom. Mag.*, vol. 18, no. 4, pp. 80–92, Dec. 2011.
[16] M. Shan, Y. Bi, H. Qin, J. Li, Z. Gao, F. Lin, and B. M. Chen, "A brief survey of visual odometry for micro aerial vehicles," in *Proc. 42nd Annu. Conf. IEEE Ind. Electron. Soc. (IECON)*, Oct. 2016, pp. 6049–6054.
[17] M. O. A. Aqel, M. H. Marhaban, M. I. Saripan, and N. B. Ismail, "Review of visual odometry: Types, approaches, challenges, and applications," *SpringerPlus*, vol. 5, no. 1, p. 1897, Oct. 2016.
[18] J. Gui, D. Gu, S. Wang, and H. Hu, "A review of visual inertial odometry from filtering and optimisation perspectives," *Adv. Robot.*, vol. 29, no. 20, pp. 1289–1301, 2015.
[19] F. Farahnakian, M.-H. Haghbayan, J. Poikonen, M. Laurinen, P. Nevalainen, and J. Heikkonen, "Object detection based on multi-sensor proposal fusion in maritime environment," in *Proc. 17th IEEE Int. Conf. Mach. Learn. Appl. (ICMLA)*, Orlando, FL, USA, Dec. 2018, pp. 971–976.
[20] M.-H. Haghbayan, F. Farahnakian, J. Poikonen, M. Laurinen, P. Nevalainen, J. Plosila, and J. Heikkonen, "An efficient multi-sensor fusion approach for object detection in maritime environments," in *Proc. 21st Int. Conf. Intell. Transp. Syst. (ITSC)*, Maui, HI, USA, Nov. 2018, pp. 2163–2170.
[21] Y. Cheng, M. Maimone, and L. Matthies, "Visual odometry on the mars exploration rovers," in *Proc. IEEE Int. Conf. Syst., Man Cybern.*, vol. 1, Oct. 2005, pp. 903–910.
[22] K. Yousif, A. Bab-Hadiashar, and R. Hoseinnezhad, "An overview to visual odometry and visual SLAM: Applications to mobile robotics," *Intell. Ind. Syst.*, vol. 1, no. 4, pp. 289–311, Nov. 2015.
[23] S. Du, W. Sun, Y. Gao, and Z. Li, "An investigation on MEMS IMU error mitigation using rotation modulation technique," in *Proc. 27th Int. Tech. Meeting Satell. Division Inst. Navigat. ION GNSS*, vol. 3, Jan. 2014, pp. 1822–1838.
[24] A. Solin, S. Cortes, E. Rahtu, and J. Kannala, "Inertial odometry on handheld smartphones," in *Proc. 21st Int. Conf. Inf. Fusion (FUSION)*, Jul. 2017, pp. 1–5.
[25] R. Saito, K. Watanabe, and I. Nagai, "Laser odometry taking account of the tilt on the laser sensor," in *Proc. 10th Asian Control Conf. (ASCC)*, May/Jun. 2015, pp. 1–4.
[26] J. Zhang and S. Singh, "LOAM: Lidar odometry and mapping in real-time," in *Proc. Conf. Robot. Sci. Syst. (RSS)*, 2014, p. 9.
[27] S. Rusinkiewicz and M. Levoy, "Efficient variants of the ICP algorithm," in *Proc. 3rd Int. Conf. 3-D Digit. Imag. Modeling*, May 2001, pp. 145–152.
[28] F. Moosmann and C. Stiller, "Velodyne SLAM," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2011, pp. 393–398.
[29] S. Du, N. Zheng, S. Ying, Q. You, and Y. Wu, "An extension of the ICP algorithm considering scale factor," in *Proc. IEEE Int. Conf. Image Process.*, vol. 5, Sep./Oct. 2007, pp. V-193–V-196.
[30] A. Segal, D. Hähnel, and S. Thrun, "Generalized-ICP," *Robot., Sci. Syst.*, vol. 2, no. 4, p. 435, 2009.
[31] Y. Chen and G. Medioni, "Object modeling by registration of multiple range images," in *Proc. IEEE Int. Conf. Robot. Autom.*, vol. 3, Apr. 1991, pp. 2724–2729.
[32] M. Achtelik, A. Bachrach, R. He, S. Prentice, and N. Roy, "Stereo vision and laser odometry for autonomous helicopters in GPS-denied indoor environments," *Proc. SPIE*, vol. 7332, Apr. 2009, Art. no. 733219.
[33] Y. Zhuang, S. Yang, X. Li, and W. Wang, "3D-laser-based visual odometry for autonomous mobile robot in outdoor environments," in *Proc. 3rd Int. Conf. Awareness Sci. Technol. (iCAST)*, Sep. 2011, pp. 133–138.
[34] M. Yan, J. Wang, J. Li, and C. Zhang, "Loose coupling visual-lidar odometry by combining VISO2 and LOAM," in *Proc. 36th Chin. Control Conf. (CCC)*, Jul. 2017, pp. 6841–6846.
[35] S. Anderson and T. D. Barfoot, "RANSAC for motion-distorted 3D visual sensors," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Nov. 2013, pp. 2093–2099.
[36] M. Velas, M. Spanel, and A. Herout, "Collar line segments for fast odometry estimation from velodyne point clouds," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, May 2016, pp. 4486–4495.
[37] J. Horn and G. Schmidt, "Continuous localization of a mobile robot based on 3D-laser-range-data, predicted sensor images, and dead-reckoning," *Robot. Auton. Syst.*, vol. 14, nos. 2–3, pp. 99–118, 1995.
[38] E. B. Quist and R. W. Beard, "Radar odometry on fixed-wing small unmanned aircraft," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 52, no. 1, pp. 396–410, Feb. 2016.
[39] E. B. Quist, P. C. Niedfeldt, and R. W. Beard, "Radar odometry with recursive-RANSAC," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 52, no. 4, pp. 1618–1630, Aug. 2016.
[40] J. Callmer, D. Törnqvist, F. Gustafsson, H. Svensson, and P. Carlbom, "Radar SLAM using visual features," *EURASIP J. Adv. Signal Process.*, vol. 2011, p. 71, Sep. 2011.
[41] R. O. Duda and R. E. Hart, "Use of the Hough transformation to detect lines and curves in pictures," *Commun. ACM*, vol. 15, no. 1, pp. 11–15, Jan. 1972. doi: 10.1145/361237.361242.

[42] A. F. Scannapieco, A. Renga, G. Fasano, and A. Moccia, "Experimental analysis of radar odometry by commercial ultralight radar sensor for miniaturized UAS," *J. Intell. Robot. Syst.*, vol. 90, nos. 3–4, pp. 485–503, Sep. 2018.

[43] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proc. 2nd Int. Conf. Knowl. Discovery Data Mining (KDD)*, Portland, OR, USA, 1996, pp. 226–231.

[44] J. Matas, O. Chum, M. Urban, and T. Pajdla, "Robust wide-baseline stereo from maximally stable extremal regions," *Image Vis. Comput.*, vol. 22, no. 10, pp. 761–767, 2004.

[45] S. H. Cen and P. Newman, "Precise ego-motion estimation with millimeter-wave radar under diverse and challenging conditions," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, Brisbane, QLD, Australia, May 2018, pp. 1–8.

[46] F. Schuster, M. Wörner, C. G. Keller, M. Haueis, and C. Curio, "Robust localization based on radar signal clustering," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2016, pp. 839–844.

[47] D. Vivet, P. Checchin, and R. Chapuis, "Localization and mapping using only a rotating FMCW radar sensor," *Sensors*, vol. 13, no. 4, pp. 4527–4552, 2013.

[48] A. F. Scannapieco, M. D. Graziano, G. Fasano, and A. Renga, "Improving radar-based mini-UAS navigation in complex environments with outlier rejection," in *Proc. AIAA Scitech Forum*, 2019, p. 2379.

[49] M. Mostafa, S. Zahran, A. Moussa, N. El-Sheimy, and A. Sesay, "Radar and visual odometry integrated system aided navigation for UAVS in GNSS denied environment," *Sensors*, vol. 18, no. 9, p. 2776, 2018.

[50] H. P. Moravec, "Obstacle avoidance and navigation in the real world by a seeing robot rover," Ph.D. dissertation, Stanford, CA, USA, 1980.

[51] L. Matthies and S. Shafer, "Error modeling in stereo navigation," *IEEE J. Robot. Autom.*, vol. 3, no. 3, pp. 239–248, Jun. 1987.

[52] H. P. Morevec, "Towards automatic visual obstacle avoidance," in *Proc. 5th Int. Joint Conf. Artif. Intell. (IJCAI)*, vol. 2, 1977, p. 584.

[53] C. G. Harris and J. M. Pike, "3D positional integration from image sequences," in *Proc. Alvey Vis. Conf.*, Cambridge, U.K., 1987, pp. 1–4.

[54] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon, "Bundle adjustment—A modern synthesis," in *Proc. Int. Workshop Vis. Algorithms (ICCV)*, 2000, pp. 298–372.

[55] H. Lin and F. Defay, "Loosely coupled stereo inertial odometry on low-cost system," in *Proc. Int. Micro Air Vehicle Conf. Flight Competition (IMAV)*, 2017, pp. 143–148.

[56] D. Nistér, "An efficient solution to the five-point relative pose problem," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 6, pp. 756–770, Jun. 2004.

[57] M. A. Fischler and O. Firschein, Eds., *Readings in Computer Vision: Issues, Problems, Principles, and Paradigms*. San Francisco, CA, USA: Morgan Kaufmann, 1987.

[58] J. Ruppelt and G. F. Trommer, "Stereo-camera visual odometry for outdoor areas and in dark indoor environments," *IEEE Aerosp. Electron. Syst. Mag.*, vol. 31, no. 11, pp. 4–12, Nov. 2016.

[59] D. Nistér, O. Naroditsky, and J. Bergen, "Visual odometry for ground vehicle applications," *J. Field Robot.*, vol. 23, no. 1, pp. 3–20, 2006.

[60] A. Cumani, "Feature localization refinement for improved visual odometry accuracy," *Int. J. Circuits, Syst. Signal Process.*, vol. 5, no. 2, pp. 151–158, Jan. 2011.

[61] H. E. Benseddik. O. A. Djekoune, and M. Belhocine, "Sift and surf performance evaluation for mobile robot-monocular visual odometry," *Int. J. Image Graph.*, vol. 2, no. 1, pp. 70–76, 2014.

[62] O. Naroditsky, X. S. Zhou, J. Gallier, S. I. Roumeliotis, and K. Daniilidis, "Two efficient solutions for visual odometry using directional correspondence," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 4, pp. 818–824, Apr. 2012.

[63] A. Howard, "Real-time stereo visual odometry for autonomous ground vehicles," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Sep. 2008, pp. 3946–3952.

[64] N. M. Suaib, M. H. Marhaban, M. I. Saripan, and S. A. Ahmad, "Performance evaluation of feature detection and feature matching for stereo visual odometry using SIFT and SURF," in *Proc. IEEE REGION Symp.*, Apr. 2014, pp. 200–203.

[65] C. Villanueva-Escudero, J. Villegas-Cortez, A. Zúñiga-López, and C. Avilés-Cruz, "Monocular visual odometry based navigation for a differential mobile robot with Android OS," in *Human-Inspired Computing and Its Applications*, A. Gelbukh, F. C. Espinoza, and S. N. Galicia-Haro, Eds. Springer, 2014, pp. 281–292.

[66] I. Parra, M. A. Sotelo, D. F. Llorca, and M. Ocaña, "Robust visual odometry for vehicle localization in urban environments," *Robotica*, vol. 28, no. 3, pp. 441–452, May 2010.

[67] R. Gonzalez, F. Rodriguez, J. L. Guzman, C. Pradalier, and R. Siegwart, "Control of off-road mobile robots using visual odometry and slip compensation," *Adv. Robot.*, vol. 27, no. 11, pp. 893–906, 2013.

[68] S. Lovegrove, A. J. Davison, and J. Ibañez-Guzmán, "Accurate visual odometry from a rear parking camera," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2011, pp. 788–793.

[69] Y. Yu, C. Pradalier, and G. Zong, "Appearance-based monocular visual odometry for ground vehicles," in *Proc. IEEE/ASME Int. Conf. Adv. Intell. Mechatronics (AIM)*, Jul. 2011, pp. 862–867.

[70] N. Nourani-Vatani, J. Roberts, and M. V. Srinivasan, "Practical visual odometry for car-like vehicles," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2009, pp. 3551–3557.

[71] N. Nourani-Vatani and P. V. K. Borges, "Correlation-based visual odometry for ground vehicles," *J. Field Robot.*, vol. 28, no. 5, pp. 742–768, Sep./Oct. 2011.

[72] C. McManus, P. Furgale, and T. D. Barfoot, "Towards lighting-invariant visual navigation: An appearance-based approach using scanning laser-rangefinders," *Robot. Auton. Syst.*, vol. 61, no. 8, pp. 836–852, 2013.

[73] A. M. Zhang and L. Kleeman, "Robust appearance based visual route following for navigation in large-scale outdoor environments," *Int. J. Robot. Res.*, vol. 28, no. 3, pp. 331–356, 2009.

[74] N. Bellotto, K. Burn, E. Fletcher, and S. Wermter, "Appearance-based localization for mobile robots using digital zoom and visual compass," *Robot. Auton. Syst.*, vol. 56, no. 2, pp. 143–156, 2008.

[75] J. Feng, C. Zhang, B. Sun, and Y. Song, "A fusion algorithm of visual odometry based on feature-based method and direct method," in *Proc. Chin. Automat. Congr. (CAC)*, Oct. 2017, pp. 1854–1859.

[76] N. Krombach, D. Droeschel, and S. Behnke, "Combining feature-based and direct methods for semi-dense real-time stereo visual odometry," in *Intelligent Autonomous Systems* (Advances in Intelligent Systems and Computing), vol. 531. Springer, Jul. 2017, pp. 855–868.

[77] A. I. Mourikis and S. I. Roumeliotis, "A multi-state constraint Kalman filter for vision-aided inertial navigation," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, Apr. 2007, pp. 3565–3572.

[78] J. A. Hesch, D. G. Kottas, S. L. Bowman, and S. I. Roumeliotis, "Consistency analysis and improvement of vision-aided inertial navigation," *IEEE Trans. Robot.*, vol. 30, no. 1, pp. 158–176, Feb. 2014.

[79] J.-P. Tardif, Y. Pavlidis, and K. Daniilidis, "Monocular visual odometry in urban environments using an omnidirectional camera," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Sep. 2008, pp. 2531–2538.

[80] J. Kelly and G. S. Sukhatme, "An experimental study of aerial stereo visual odometry," *IFAC Proc. Volumes*, vol. 40, no. 15, pp. 197–202, 2007.

[81] G. Dubbelman and F. C. A. Groen, "Bias reduction for stereo based motion estimation with applications to large scale visual odometry," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 2222–2229.

[82] N. Sünderhauf and P. Protzel, "Towards using sparse bundle adjustment for robust stereo odometry in outdoor Terrain," in *Proc. Towards Auto. Robotic Syst. (TAROS)*, 2006, pp. 206–213.

[83] A. S. Huang, A. Bachrach, P. Henry, M. Krainin, D. Maturana, D. Fox, and N. Roy, "Visual odometry and mapping for autonomous flight using an RGB-D camera," in *Proc. Int. Symp. Robot. Res.*, 2011, pp. 235–252.

[84] L. Frédéric, "The visual compass: Performance and limitations of an appearance-based method," *J. Field Robot.*, vol. 23, no. 10, pp. 913–941, 2006.

[85] L. Heng and B. Choi, "Semi-direct visual odometry for a fisheye-stereo camera," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2016, pp. 4077–4084.

[86] P. Liu, L. Heng, T. Sattler, A. Geiger, and M. Pollefeys, "Direct visual odometry for a fisheye-stereo camera," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2017, pp. 1746–1752.

[87] A. J. Lee and A. Kim, "Event-based real-time optical flow estimation," in *Proc. 17th Int. Conf. Control, Automat. Syst. (ICCAS)*, Oct. 2017, pp. 787–791.

[88] B. Kueng, E. Mueggler, G. Gallego, and D. Scaramuzza, "Low-latency visual odometry using event-based feature tracks," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2016, pp. 16–23.
[89] A. Censi and D. Scaramuzza, "Low-latency event-based visual odometry," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, May/Jun. 2014, pp. 703–710.
[90] A. Z. Zhu, N. Atanasov, and K. Daniilidis, "Event-based visual inertial odometry," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5816–5824.
[91] D. V. García, L. F. Rojo, A. G. Aparicio, L. P. Castelló, and Ó. R. García, "Visual odometry through appearance- and feature-based method with omnidirectional images," *J. Robot.*, vol. 2012, Jul. 2012, Art. no. 797063.
[92] R. K. Gupta and S.-Y. Cho, "A correlation-based approach for real-time stereo matching," in *Advances in Visual Computing*, G. Bebis, R. Boyle, B. Parvin, D. Koracin, R. Chung, R. Hammound, M. Hussain, T. Kar-Han, R. Crawfis, D. Thalmann, D. Kao, and L. Avila, Eds. Berlin, Germany: Springer, 2010, pp. 129–138.
[93] B. K. P. Horn and B. G. Schunck, "Determining optical flow," *Artif. Intell.*, vol. 17, nos. 1–3, pp. 185–203, Aug. 1981.
[94] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proc. 7th Int. Joint Conf. Artif. Intell. (IJCAI)*, vol. 2. San Francisco, CA, USA: Morgan Kaufmann, 1981, pp. 674–679.
[95] M. El-Gayar, H. Soliman, and N. Meky, "A comparative study of image low level feature extraction algorithms," *Egyptian Informat. J.*, vol. 14, no. 2, pp. 175–181, 2013.
[96] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proc. Int. Conf. Comput. Vis. (ICCV)*, vol. 2, Washington, DC, USA, 1999, pp. 1150–1157.
[97] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (SURF)," *Comput. Vis. Image Understand.*, vol. 110, no. 3, pp. 346–359, Jun. 2008.
[98] E. Rosten and T. Drummond, "Machine learning for high-speed corner detection," in *Proc. 9th Eur. Conf. Comput. Vis. (ECCV)*. Berlin, Germany: Springer-Verlag, 2006, pp. 430–443.
[99] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "Orb: An efficient alternative to SIFT or SURF," in *Proc. Int. Conf. Comput. Vis.*, 2011, pp. 2564–2571.
[100] P. A. Mlsna and J. J. Rodríguez, *Gradient and Laplacian Edge Detection*. Amsterdam, The Netherlands: Elsevier, 2009, pp. 495–524.
[101] D. Marr and E. Hildreth, "Theory of edge detection," *Proc. Roy. Soc. London. B, Biol. Sci.*, vol. 207, pp. 187–217, Feb. 1980.
[102] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-8, no. 6, pp. 679–698, Nov. 1986.
[103] S. Yang and S. Scherer, "Direct monocular odometry using points and lines," *CoRR*, vol. abs/1703.06380, Mar. 2017.
[104] N. Yang, R. Wang, X. Gao, and D. Cremers, "Challenges in monocular visual odometry: Photometric calibration, motion bias, and rolling shutter effect," *IEEE Robot. Autom. Lett.*, vol. 3, no. 4, pp. 2878–2885, Oct. 2018.
[105] C. Forster, M. Pizzoli, and D. Scaramuzza, "SVO: Fast semi-direct monocular visual odometry," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, May/Jun. 2014, pp. 15–22.
[106] D. M. Helmick, Y. Cheng, D. S. Clouse, L. H. Matthies, and S. I. Roumeliotis, "Path following using visual odometry for a mars rover in high-slip environments," in *Proc. IEEE Aerosp. Conf.*, vol. 2, Mar. 2004, pp. 772–789.
[107] M. Li and A. I. Mourikis, "High-precision, consistent EKF-based visual-inertial odometry," *Int. J. Robot. Res.*, vol. 32, no. 6, pp. 690–711, May 2013.
[108] B. M. Kitt, J. Rehder, A. D. Chambers, M. Schonbein, H. Lategahn, and S. Singh, "Monocular visual odometry using a planar road model to solve scale ambiguity," in *Proc. Eur. Conf. Mobile Robots*, 2011, pp. 1–6.
[109] X. Yin, X. Wang, X. Du, and Q. Chen, "Scale recovery for monocular visual odometry using depth estimated with deep convolutional neural fields," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 5871–5879.
[110] D. Belter, M. Nowicki, and P. Skrzypczy ski, "Improving accuracy of feature-based RGB-D SLAM by modeling spatial uncertainty of point features," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, May 2016, pp. 1279–1284.
[111] J. Engel, T. Schöps, and D. Cremers, "LSD-SLAM: Large-scale direct monocular SLAM," in *Computer Vision—ECCV*, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds. Cham, Switzerland: Springer, 2014, pp. 834–849.
[112] H. Wang, X. Wu, Z. Chen, and Y. He, "A novel hybrid visual odometry using an RGB-D camera," in *Proc. 33rd Youth Acad. Annu. Conf. Chin. Assoc. Automat. (YAC)*, May 2018, pp. 47–51.
[113] Z. Zhang, H. Rebecq, C. Forster, and D. Scaramuzza, "Benefit of large field-of-view cameras for visual odometry," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, May 2016, pp. 801–808.
[114] J. P. Barreto and H. Araujo, "Direct least square fitting of paracatadioptric line images," in *Proc. Conf. Comput. Vis. Pattern Recognit. Workshop*, vol. 7, Jun. 2003, p. 78.
[115] J. C. Bazin, C. Demonceaux, P. Vasseur, and I. S. Kweon, "Motion estimation by decoupling rotation and translation in catadioptric vision," *Comput. Vis. Image Understand.*, vol. 114, no. 2, pp. 254–273, 2010.
[116] K. Dannilidis and H. H. Nagel, "The coupling of rotation and translation in motion estimation of planar surfaces," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 1993, pp. 188–193.
[117] D. Caruso, J. Engel, and D. Cremers, "Large-scale direct SLAM for omnidirectional cameras," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep./Oct. 2015, pp. 141–148.
[118] H. Hirschmuller, "Stereo processing by semiglobal matching and mutual information," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 2, pp. 328–341, Feb. 2008.
[119] K. Sun, K. Mohta, B. Pfrommer, M. Watterson, S. Liu, Y. Mulgaonkar, C. J. Taylor, and V. Kumar, "Robust stereo visual inertial odometry for fast autonomous flight," *IEEE Robot. Autom. Lett.*, vol. 3, no. 2, pp. 965–972, Apr. 2018.
[120] N. Onkarappa and A. D. Sappa, "An empirical study on optical flow accuracy depending on vehicle speed," in *Proc. IEEE Intell. Vehicles Symp.*, Jun. 2012, pp. 1138–1143.
[121] J. Campbell, R. Sukthankar, I. Nourbakhsh, and A. Pahwa, "A robust visual odometry and precipice detection system using consumer-grade monocular vision," in *Proc. IEEE Int. Conf. Robot. Automat.*, Apr. 2005, pp. 3421–3427.
[122] M. Dille, B. Grocholsky, and S. Singh, "Outdoor downward-facing optical flow odometry with commodity sensors," in *Proc. Results 7th Int. Conf. Field Service Robot. (FSR)*, A. Howard, K. Iagnemma, and A. Kelly, Eds. Cambridge, MA, USA: Springer, 2009.
[123] L. Piyathilaka and R. Munasinghe, "Multi-camera visual odometry for skid steered field robot," in *Proc. 5th Int. Conf. Inf. Automat. Sustainability*, Dec. 2010, pp. 189–194.
[124] L. Piyathilaka and R. Munasinghe, "An experimental study on using visual odometry for short-run self localization of field robot," in *Proc. 5th Int. Conf. Inf. Automat. Sustainability*, Dec. 2010, pp. 150–155.
[125] A. Geiger, J. Ziegler, and C. Stiller, "StereoScan: Dense 3D reconstruction in real-time," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2011, pp. 963–968.
[126] D. Scaramuzza, A. Martinelli, and R. Siegwart, "Precise bearing angle measurement based on omnidirectional conic sensor and defocusing," in *Proc. Eur. Conf. Mobile Robots (ECMR)*, 2005, pp. 38–43.
[127] R. Ghabcheloo and S. Siddiqui, "Complete odometry estimation of a vehicle using single automotive radar and a gyroscope," in *Proc. 26th Medit. Conf. Control Automat. (MED)*, Zadar, Croatia, Jun. 2018, pp. 855–860.
[128] K. Mohta, M. Watterson, Y. Mulgaonkar, S. Liu, C. Qu, A. Makineni, K. Saulnier, K. Sun, A. Zhu, J. Delmerico, K. Karydis, N. Atanasov, G. Loianno, D. Scaramuzza, K. Daniilidis, C. J. Taylor, and V. Kumar, "Fast, autonomous flight in GPS-denied and cluttered environments," *J. Field Robot.*, vol. 35, no. 1, pp. 101–120, 2018.
[129] F. Zheng, G. Tsai, Z. Zhang, S. Liu, C. Chu, and H. Hu, "Trifo-VIO: Robust and efficient stereo visual inertial odometry using points and lines," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Systems (IROS)*, Madrid, Spain, Oct. 2018, pp. 3686–3693.
[130] S. Heo and C. G. Park, "Consistent EKF-based visual-inertial odometry on matrix Lie group," *IEEE Sensors J.*, vol. 18, no. 9, pp. 3780–3788, May 2018.
[131] D. Caruso, A. Eudes, M. Sanfourche, D. Vissiere, and G. Le Besnerais, "An inverse square root filter for robust indoor/outdoor magneto-visual-inertial odometry," in *Proc. Int. Conf. Indoor Positioning Indoor Navigat. (IPIN)*, Sep. 2017, pp. 1–8.
[132] K. Wu, T. Zhang, D. Su, S. Huang, and G. Dissanayake, "An invariant-EKF VINS algorithm for improving consistency," in *Proc. IEEE Int. Conf. Intell. Robots Syst.*, Sep. 2017, pp. 1578–1585.
[133] A. Beauvisage and N. Aouf, "Multimodal visual-inertial odometry for navigation in cold and low contrast environment," in *Proc. Eur. Conf. Mobile Robots (ECMR)*, Sep. 2017, pp. 1–6.

[134] G. Loianno, M. Watterson, and V. Kumar, "Visual inertial odometry for quadrotors on SE(3)," in *Proc. IEEE Int. Conf. Robot. Automat.*, May 2016, pp. 1544–1551.

[135] K. J. Wu, A. Ahmed, G. A. Georgiou, and S. I. Roumeliotis, "A square root inverse filter for efficient vision-aided inertial navigation on mobile devices," *Robot., Sci. Syst.*, vol. 2, Jul. 2015.

[136] C. Forster, L. Carlone, F. Dellaert, and D. Scaramuzza, "On-manifold preintegration for real-time visual–inertial odometry," *IEEE Trans. Robot.*, vol. 33, no. 1, pp. 1–21, Feb. 2017.

[137] E. Hong and J. Lim, "Visual inertial odometry using coupled nonlinear optimization," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2017, pp. 6879–6885.

[138] Z. Yang and S. Shen, "Monocular visual–inertial state estimation with online initialization and camera–IMU extrinsic calibration," *IEEE Trans. Autom. Sci. Eng.*, vol. 14, no. 1, pp. 39–51, Jan. 2017.

[139] M. Bloesch, M. Burri, S. Omari, M. Hutter, and R. Siegwart, "Iterated extended Kalman filter based visual-inertial odometry using direct photometric feedback," *Int. J. Robot. Res.*, vol. 36, no. 10, pp. 1053–1072, 2017.

[140] M. Schwaab, D. Plaia, D. Gaida, and Y. Manoli, "Tightly coupled fusion of direct stereo visual odometry and inertial sensor measurements using an iterated information filter," in *Proc. DGON Inertial Sensors Syst. (ISS)*, Sep. 2017, pp. 1–20.

[141] S. Sirtkaya, B. Seymen, and A. A. Alatan, "Loosely coupled Kalman filtering for fusion of visual odometry and inertial navigation," in *Proc. 16th Int. Conf. Inf. Fusion*, Jul. 2013, pp. 219–226.

[142] S. Shen, Y. Mulgaonkar, N. Michael, and V. Kumar, "Multi-sensor fusion for robust autonomous flight in indoor and outdoor environments with a rotorcraft MAV," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May/Jun. 2014, pp. 4974–4981.

[143] D. Scaramuzza *et al.*, "Vision-controlled micro flying robots: From system design to autonomous navigation and mapping in GPS-denied environments," *IEEE Robot. Autom. Mag.*, vol. 21, no. 3, pp. 26–40, Sep. 2014.

[144] M. Ramezani, K. Khoshelham, and L. Kneip, "Omnidirectional visual-inertial odometry using multi-state constraint Kalman filter," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2017, pp. 1317–1323.

[145] D. Caruso, M. Sanfourche, G. Le Besnerais, and D. Vissière, "Infrastructureless indoor navigation with an hybrid magneto-inertial and depth sensor system," in *Proc. Int. Conf. Indoor Positioning Indoor Navigat. (IPIN)*, vol. 1, Oct. 2016, pp. 1–8.

[146] V. Usenko, J. Engel, J. Stückler, and D. Cremers, "Direct visual-inertial odometry with stereo cameras," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, May 2016, pp. 1885–1892.

[147] Y. Ling, T. Liu, and S. Shen, "Aggressive quadrotor flight using dense visual-inertial fusion," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, May 2016, pp. 1499–1506.

[148] C. Forster, L. Carlone, F. Dellaert, and D. Scaramuzza, "IMU preintegration on manifold for efficient visual-inertial maximum-a-posteriori estimation," in *Robotics: Science and Systems XI*. Rome, Italy: Sapienza Univ. of Rome, 2015.

[149] F. Pang, Z. Chen, L. Pu, and T. Wang, "Depth enhanced visual-inertial odometry based on multi-state constraint Kalman filter," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2017, pp. 1761–1767.

[150] L. Tang, S. Yang, N. Cheng, and Q. Li, "Toward autonomous navigation using an RGB-D camera for flight in unknown indoor environments," in *Proc. IEEE Chin. Guid., Navigat. Control Conf. (CGNCC)*, Aug. 2014, pp. 2007–2012.

[151] M. Bloesch, S. Omari, M. Hutter, and R. Siegwart, "Robust visual inertial odometry using a direct EKF-based approach," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep./Oct. 2015, pp. 298–304.

[152] K. Konolige, M. Agrawal, and J. Solà, "Large-scale visual odometry for rough Terrain," in *Robotics Research*, M. Kaneko and Y. Nakamura, Eds. Berlin, Germany: Springer, 2011, pp. 201–212.

[153] J. Shi and C. Tomasi, "Good features to track," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 1994, pp. 593–600.

[154] M. Agrawal, K. Konolige, and M. R. Blas, "CenSurE: Center surround extremas for realtime feature detection and matching," in *Computer Vision—ECCV* (Lecture Notes in Computer Science), vol. 5305. 2008, pp. 102–115.

[155] S. Weiss, M. W. Achtelik, S. Lynen, M. Chli, and R. Siegwart, "Real-time onboard visual-inertial state estimation and self-calibration of MAVs in unknown environments," in *Proc. IEEE Int. Conf. Robot. Automat.* Saint Paul, MN, USA: IEEE, May 2012, pp. 957–964.

[156] Y. Ma, S. Soatto, J. Kosecka, and S. S. Sastry, *An Invitation to 3-D Vision: From Images to Geometric Models*. New York, NY, USA: Springer-Verlag, 2003.

[157] Y. Ling, M. Kuse, and S. Shen, "Edge alignment-based visual–inertial fusion for tracking of aggressive motions," *Auton. Robots*, vol. 42, no. 3, pp. 513–528, Mar. 2018.

[158] T. Lupton and S. Sukkarieh, "Visual-inertial-aided navigation for high-dynamic motion in built environments without initial conditions," *IEEE Trans. Robot.*, vol. 28, no. 1, pp. 61–76, Feb. 2012.

[159] S. Shen, N. Michael, and V. Kumar, "Tightly-coupled monocular visual-inertial fusion for autonomous flight of rotorcraft MAVs," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, May 2015, pp. 5303–5310.

[160] J. Kelly and G. S. Sukhatme, "Visual-inertial sensor fusion: Localization, mapping and sensor-to-sensor self-calibration," *Int. J. Robot. Res.*, vol. 30, no. 1, pp. 56–79, 2011.

[161] R. van der Merwe and E. A. Wan, "Sigma-point Kalman filters for integrated navigation," in *Proc. 60th Annu. Meeting Inst. Navigat.*, 2004, pp. 641–654.

[162] S. J. Julier and J. K. Uhlmann, "New extension of the Kalman filter to nonlinear systems," *Proc. SPIE*, vol. 3068, pp. 182–193, Jul. 1997.

[163] C. N. Taylor, "Fusion of inertial, vision, and air pressure sensors for MAV navigation," in *Proc. IEEE Int. Conf. Multisensor Fusion Integr. Intell. Syst.*, Aug. 2008, pp. 475–480.

[164] S. Leutenegger, S. Lynen, M. Bosse, R. Siegwart, and P. Furgale, "Keyframe-based visual-inertial odometry using nonlinear optimization," *Int. J. Robot. Res.*, vol. 34, no. 3, pp. 314–334, 2015.

[165] D. Simon, *Optimal State Estimation: Kalman, H Infinity, and Nonlinear Approaches*. New York, NY, USA: Wiley, 2006.

[166] J. Civera, O. G. Grasa, A. J. Davison, and J. M. M. Montiel, "1-point RANSAC for EKF-based structure from motion," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2009, pp. 3498–3504.

[167] J. L. Crassidis and F. L. Markley, "Unscented filtering for spacecraft attitude estimation," in *Proc. AIAA Guid., Navigat. Control Conf. (GNC)*, 2003.

[168] S. Shen, Y. Mulgaonkar, N. Michael, and V. Kumar, "Vision-based state estimation and trajectory control towards high-speed flight with a quadrotor," in *Proc. Robot., Sci. Syst. IX*. Berlin, Germany: Technische Univ. Berlin, Jun. 2013.

[169] J. K. Suhr, "Kanade-lucas-tomasi (KLT) feature tracker," Yonsei Univ., Seoul, Republic of Korea, Tech. Rep. EEE6503, 2009.

[170] D. Weikersdorfer, R. Hoffmann, and J. Conradt, "Simultaneous localization and mapping for event-based vision systems," in *Proc. 9th Int. Conf. Comput. Vis. Syst. (ICVS)*, St. Petersburg, Russia, Jul. 2013, pp. 133–142.

[171] M. Cook, L. Gugelmann, F. Jug, C. Krautz, and A. Steger, "Interacting maps for fast visual interpretation," in *Proc. Int. Joint Conf. Neural Netw.*, Jul./Aug. 2011, pp. 770–776.

SHERIF A. S. MOHAMED received the B.A. degree in electrical, electronics, and communication engineering from Ain Shams University, Egypt, in 2011, and the M.S. degree in electronics and information engineering from Kunsan National University, South Korea, in 2016. He is currently pursuing the Ph.D. dergee from the University of Turku, Finland. His research interests include vision-based navigation algorithms for autonomous vehicles, embedded systems, swam intelligence, and machine learning.

MOHAMMAD-HASHEM HAGHBAYAN received the B.A. degree in computer engineering from the Ferdowsi University of Mashhad, the M.S. degree in computer architecture from the University of Tehran, Iran, and the Ph.D. degree (Hons.) from the University of Turku, Finland. Since 2018, he has been a Lecturer and holding a Postdoctoral position at the University of Turku. His research interests include high-performance energy-efficient architectures for autonomous systems and artificial intelligence. He has several years of experience working in industry and designing IP cores as well as developing research tools.

TOMI WESTERLUND joined the Department of Future Technologies, University of Turku, as a Senior Researcher, in 2008, where he became a University Research Fellow, in 2015, and an Adjunct Professor (docent) of embedded electronics education and applications, in 2017. Since 2013, he has been a Visiting Scholar with Fudan University, Shanghai, China. He is also with the Finnish Centre of Excellence in Research of Sustainable Space (Academy of Finland). He is the responsible Researcher of the nanosatellite technology research group at the department. In addition to the nanosatellite technology, his current research interests include the Internet of Things (IoT) in smart agriculture, smart cities, as well as unmanned autonomous platforms (aerial, ground, and surface). In all these application areas, the core research interests include energy efficiency, dependability, interoperability, and autonomous operation.

JUKKA HEIKKONEN has been a Professor of computer science with the University of Turku, Finland, since 2009. His current research as the Head of the Algorithms and Computational Intelligent (ACI) research group is related to data analytics, machine learning, and autonomous systems. He has worked at top level research laboratories and at the Center of Excellences in Finland and international organizations (European Commission, Japan), and has led many international and national research projects. He has authored more than 150 scientific articles.

HANNU TENHUNEN received the Diploma degree from the Helsinki University of Technology, Finland, in 1982, and the Ph.D. degree from Cornell University, Ithaca, NY, USA, in 1986. He received an Honorary Doctorate from Tallinn Technical University. He was a Full Professor, an Invited Professor, or a Visiting Honorary Professor in Finland (TUT, UTU), Sweden (KTH), USA (Cornel U), France (INPG), China (Fudan and Beijing Jiatong Universities), and Hong Kong (The Chinese University of Hong Kong). He is currently a Professor with the Electronic Systems Laboratory, Royal Institute of Technology (KTH), and a Professor with the University of Turku (UTU). He has contributed over 850 international publications with an H-index 41. He has served on the technical program committee of all major conferences in his research areas. He has contributed numerous invited papers to various journals. He has nine international patents granted in multiple countries. He is a member of the Academy of Engineering Science of Finland. He has been the General Chair, the Vice-Chair, or a member of the steering committee of multiple conferences in his core competence areas. He has been one of the founding editorial board members of three scientific journals. He has been a quest editor of multiple special issues of scientific journals or books.

JUHA PLOSILA (M'06) received the Ph.D. degree in electronics and communication technology from the University of Turku (UTU), Finland, in 1999, where he is currently a Professor (Full) of autonomous systems and robotics with the Department of Future Technologies. He is also the Head of the EIT Digital Master Programme in Embedded Systems at the EIT Digital Master School (European Institute of Innovation and Technology) and represents UTU in the Node Strategy Committee of the EIT Digital Helsinki/Finland node. He has a strong research background in adaptive multi-processing systems and platforms and their design, including specification, development, and verification of self-aware multi-agent monitoring and control architectures for massively parallel systems, machine learning and evolutionary computing-based approaches, as well as application of heterogeneous energy-efficient architectures to new computational challenges in the cyber–physical systems and the Internet of Things domains, with a recent focus on fog/edge computing (edge intelligence) and autonomous multi-drone systems.