sensors

MDPI

Article

ACK-MSCKF: Tightly-Coupled Ackermann Multi-State Constraint Kalman Filter for Autonomous Vehicle Localization

Fangwu Ma, Jinzhu Shi *, Yu Yang, Jinhang Li and Kai Dai

State Key Laboratory of Automotive Simulation and Control, Jilin University, Changchun 130022, China; mikema@jlu.edu.cn (F.M.); yangyu17@mails.jlu.edu.cn (Y.Y.); jinhang18@mails.jlu.edu.cn (J.L.); daikai18@mails.jlu.edu.cn (K.D.)

* Correspondence: shijz16@mails.jlu.edu.cn

Received: 8 October 2019; Accepted: 3 November 2019; Published: 5 November 2019

Abstract: Visual-Inertial Odometry (VIO) is subjected to additional unobservable directions under the special motions of ground vehicles, resulting in larger pose estimation errors. To address this problem, a tightly-coupled Ackermann visual-inertial odometry (ACK-MSCKF) is proposed to fuse Ackermann error state measurements and the Stereo Multi-State Constraint Kalman Filter (S-MSCKF) with a tightly-coupled filter-based mechanism. In contrast with S-MSCKF, in which the inertial measurement unit (IMU) propagates the vehicle motion and then the propagation is corrected by stereo visual measurements, we successively update the propagation with Ackermann error state measurements and visual measurements after the process model and state augmentation. This way, additional constraints from the Ackermann measurements are exploited to improve the pose estimation accuracy. Both qualitative and quantitative experimental results evaluated under real-world datasets from an Ackermann steering vehicle lead to the following demonstration: ACK-MSCKF can significantly improve the pose estimation accuracy of S-MSCKF under the special motions of autonomous vehicles, and keep accurate and robust pose estimation available under different vehicle driving cycles and environmental conditions. This paper accompanies the source code for the robotics community.

Keywords: autonomous vehicles; visual–inertial odometry; Ackermann error state measurements; sensor fusion

1. Introduction

Providing a real-time accurate vehicle position is one of the key technologies to enhance the active safety of advanced driver assistance systems (ADASs) and achieve autonomous driving [1]. Wheel odometry is a fundamental method in most autonomous vehicles, owing to low cost and high reliability. It uses the information from the proprioceptive sensors, including the drive motor encoder sensor and steering wheel angle sensor, to estimate the position relative to a starting location with the method of dead-reckoning, based on the kinematics model. It exhibits the characteristic of high positioning accuracy in short-term operation. However, it is sensitive to systematic and non-systematic errors [2]. The accumulated errors will result in a drift in position estimation gradually over time. Therefore, it is necessary to fuse data from other sensors to improve positioning accuracy [3]. The fusion of the Global Navigation Satellite System (GNSS) and the Inertial Navigation System (INS) can provide high-accuracy global positioning [4]. However, the performance degrades quickly in GPS-denied environments. The positioning technology based on the Light Detection and Ranging (LIDAR) odometry [5,6] has the advantages of algorithm maturity, high precision and real-time performance, but its application in vehicles is limited due to the high cost of LIDAR.

On the other hand, visual odometry (VO), which adopts visual sensors, is gaining significant interest by the robotics community owing to the abundant perceptive information, small size and low cost [7–9]. A detailed review of the VO methods can be found in the literature [10–12]. While VO algorithms undergo a lack of robustness when confronted with motion blur, low texture scenes, illumination changes, sharp turns, partial or full occlusions and the influence of dynamic objects [13], visual-inertial odometry (VIO) [14–17] algorithms can significantly improve the performance by fusing low-cost IMUs, which attract much attention in the field of micro air vehicles (MAVs). For vehicle positioning, Ramezani and Khoshelham [18] propose a stereo Multi-State Constraint Kalman Filter (Stereo-MSCKF) using MSCKF [19], which shows a promising potential to provide positioning with sub-meter accuracy in short periods of GNSS signal loss. By far, VIO algorithms mainly consist of EKF-based methods, including MSF [20], ROVIO [21], MSCKF [19], Stereo-MSCKF [18], S-MSCKF [22], R-VIO [23] and optimization-based methods including the Open Keyframe-based Visual-Inertial SLAM (OKVIS) [24], VINS-Mono [25], PL-VIO [26] and Basalt [27]. Delmerico and Scaramuzza [28] performed a thorough evaluation of open source VIO, suggesting that EKF-based algorithms display better efficient performance, whereas optimization-based algorithms provide higher accuracy. Therefore, the trade-off between efficiency and accuracy in practical applications should be considered in order to choose an appropriate off-the-shelf VIO algorithm.

However, VIO algorithms provide enormous challenges for autonomous vehicle localization. Firstly, the outdoor traffic scene is completely uncontrolled. The scene can be cluttered, and thus contains a number of moving objects, such as pedestrians and vehicles. Therefore, VIO must be able to work precisely and robustly in such challenging situations to ensure the safety of pedestrian and passenger. Moreover, the on-board computer is an embedded system with limited computational resources. Hence, VIO should possess low computational complexity for the convenience of in-vehicle deployment [29]. Finally and most importantly, autonomous vehicles usually move on an approximately planar environment, and the motion is constrained to be of either constant local acceleration or no rotation. Under these special motions, VIO is subjected to additional unobservable directions [30], which indeed leads to larger positioning errors.

To address the above problems, there are some studies to integrate VIO algorithms with additional sources of information. VINS-RGBD [31] is proposed to fuse RGB-D and IMU data for small ground rescue robots, which overcomes the problem of scale unobservability mentioned in [30]. However, the use of the RGB-D sensor does not apply to the outdoor traffic scene in direct sunlight [32] for autonomous vehicles. He et al. [26] propose a tightly-coupled monocular VIO system, PL-VIO, by integrating more structured line features. The PL-VIO improves positioning accuracy, especially in uncontrolled outdoor scenarios. Nevertheless, the line detection and matching were the bottlenecks in the efficiency of the system. Zheng and Liu [33] propose an SE(2)-constrained pose parameterization optimization-based VIO for ground vehicles, which could obtain better accuracy under sharp-turn motion. Wu et al. [34] prove that the VIO scale, or two additional degrees of freedom (DOF) of global orientation, could become unobservable when ground mobile robots move with constant acceleration, or no rotation, respectively. They propose to fuse wheel odometry measurements into VIO, which significantly improves positioning accuracy under special motions. Dang et al. [35] propose a tightly-coupled VIO to fuse wheel encoder measurements by considering wheel slippage, which achieves great improvements in positioning accuracy. Huai and Huang [23] propose a robocentric VIO (R-VIO) by reformulating the system with respect to the IMU frame, and show that this system can resolve the scale unobservability issue under special motions without using additional sensors.

Although the above tightly-coupled VIO systems achieve high accuracy and robustness, they are sensitive to initialization [36] and require a higher computational burden due to the iterative optimal backend. Moreover, these methods are based upon differential drive robots or hand-held equipment. In contrast to the wide utilization of inertial measurements, KO-Fusion [37] is proposed to integrate Mecanum wheel motion constraint and the RGB-D sensor for small ground rescue robots, which improves the robustness under the fast and loopy motion.

However, the use of the RGB-D sensor and Mecanum wheels does not directly apply to autonomous vehicles which usually run on the asphalt road in the outdoor high-light traffic scene. Moreover, its efficient performance should be further optimized for the convenience of in-vehicle deployment. As the on-road and off-road autonomous vehicles in the outdoor environment are mostly Ackermann steering robots [38], it is necessary to demonstrate in further research works whether the above methods are applicable to Ackermann steering robots.

Therefore, the purpose of this paper is to effectively utilize the wheel odometry and steering model of Ackermann steering robots to improve the pose estimation accuracy under special motions of autonomous vehicles, and to keep pose estimation available with accuracy and robustness under different vehicle driving cycles and environmental conditions. In view of the efficient performance, an open source EKF-based VIO is selected as the base of our work, i.e., S-MSCKF [22], which follows the same fundamental principle as Stereo-MSCKF [18]. We propose a tightly-coupled Ackermann Multi-State Constraint Kalman Filter, termed ACK-MSCKF, to fuse Ackermann error state measurements and S-MSCKF with tightly-coupled EKF-based mechanism. The main contributions of this paper are highlighted as follows:

(1) Conducting the formulation and implementation of a tightly-coupled visual-inertial odometry system, ACK-MSCKF, which significantly improves the pose estimation accuracy under the special motions of autonomous vehicles, and can allow the position and orientation estimation to remain accurate and robust under different vehicle driving cycles and environmental conditions.

(2) The rotation extrinsic parameters calibration method between vehicle coordinates and VIO coordinates is presented with a performance comparison.

(3) Facilitating the reproducibility of related research with the source code [39] of ACK-MSCKF publicly available.

(4) The performance of ACK-MSCKF is compared with the state-of-the-art open source stereo VIO, S-MSCKF and OKVIS [24] on the real-world datasets acquired by our experimental vehicle with both qualitative and quantitative analysis results.

The remainder of this paper is structured as follows: The formulation and implementation of the proposed approach are both introduced in Section 2, and then Section 3 exhibits corresponding experiments and results. Besides, the experimental results are demonstrated in Section 4. Finally, our conclusions are drawn in Section 5.

2. The Proposed Approach

The basic principle of ACK-MSCKF is shown in Figure 1. The process model in which the state, including parameters to be estimated, is propagated using IMU measurements. Then, the state is augmented with the new camera state of the left camera. Finally, the state is successively updated by Ackermann measurements and visual measurements.

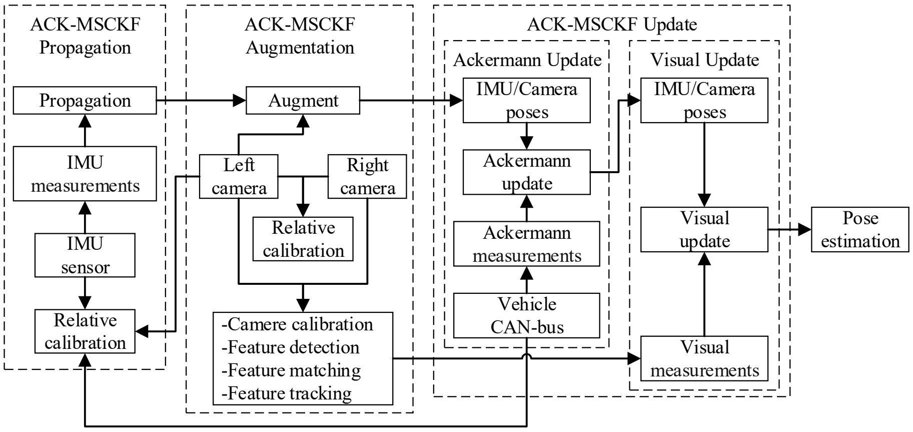


Figure 1. The basic workflow of Ackermann visual-inertial odometry (ACK-MSCKF).

2.1. Coordinate Systems and Notations

(1) Vehicle Coordinate System {B}. The {B} is fixed with the vehicle body. The origin of {B} lies on the automotive rear axle center. The x-axis and y-axis points to forward and left, respectively, following the right-hand rule.

(2) World Coordinate System {W}. The {W} has the same origin and axes as the {B} at the initial time.

(3) VIO Coordinate System {I}. The origin of {I} lies on the center of the stereo camera, which is installed in the front of the vehicle. The x-axis and y-axis points to upward and right, respectively, following the right-hand rule.

(4) Camera Coordinate System {C}. The origin of {C} lies on the center of the left camera. The x-axis and y-axis points to right and downward, respectively, following the right-hand rule.

(5) Inertial Coordinate System {G}. The origin of {G} is the same as that of {I} at the time of VIO initialized. The axes of {G} are obtained by calculation at the time of VIO initialized, and its z-axis is aligned with earth gravity.

In this paper, we adopt the JPL [40,41] quaternion conventions which are used in S-MSCKF to derive ACK-MSCKF and the Hamilton [42] quaternion conventions to implement rotation extrinsic parameters calibration.

2.2. Process Model and State Augmentation

The IMU state at the sampling time of IMU t_k is defined as,

$$\mathbf{X}_{I_k} = \begin{bmatrix} {}^{I_k}_{G}\mathbf{q}^{\mathrm{T}} & \mathbf{b}^{\mathrm{T}}_{g,k} & {}^{G}\mathbf{v}^{\mathrm{T}}_{I_k} & \mathbf{b}^{\mathrm{T}}_{a,k} & {}^{G}\mathbf{p}^{\mathrm{T}}_{I_k} & {}^{C}_{I}\mathbf{q}^{\mathrm{T}} & {}^{I}\mathbf{p}^{\mathrm{T}}_{C} \end{bmatrix}^{\mathrm{T}} \tag{1}$$

where ${}^{I_k}_{G}\mathbf{q}$ denotes the rotation quaternion from {G} to {I},${}^{G}v_{I_k}$ and ${}^{G}\mathbf{p}_{I_k}$ are the velocity and position of {I} in {G}, ${}^{C}_{I}\mathbf{q}$ and ${}^{I}\mathbf{p}_{C}$ both denote the exterior parameters between {I} and {C}. The symbols $\mathbf{b}_{g,k}$ and $\mathbf{b}_{a,k}$ are the IMU's gyroscope and accelerometer biases, respectively.

The camera state at the sampling time of camera t_j is defined as,

$$\mathbf{X}_{C_j} = \begin{bmatrix} {}^{C_j}_{G}\mathbf{q}^{\mathrm{T}} & {}^{G}\mathbf{p}^{\mathrm{T}}_{C_j} \end{bmatrix}^{\mathrm{T}} \tag{2}$$

where ${}^{C_j}_{G}\mathbf{q}$ denotes the rotation quaternion from {G} to {C} and ${}^{G}\mathbf{p}_{C_j}$ is the position of {C} in {G}.

According to Equation (1) and Equation (2), the full state vector with N camera states at time $t_k \in \left[\begin{matrix} t_j & t_{j+1} \end{matrix}\right)$ is given by,

$$\mathbf{X}_{k} = \begin{bmatrix} \mathbf{X}^{\mathrm{T}}_{I_k} & {}^{C_{j-N+1}}_{G}\mathbf{q}^{\mathrm{T}} & {}^{G}\mathbf{p}^{\mathrm{T}}_{C_{j-N+1}} & {}^{C_{j-N+2}}_{G}\mathbf{q}^{\mathrm{T}} & {}^{G}\mathbf{p}^{\mathrm{T}}_{C_{j-N+1}} & \cdots & {}^{C_j}_{G}\mathbf{q}^{\mathrm{T}} & {}^{G}\mathbf{p}^{\mathrm{T}}_{C_j} \end{bmatrix}^{\mathrm{T}} \tag{3}$$

Following Equation (3), the full error state $\widetilde{\mathbf{X}}_k \in \mathbb{R}^{21+6N}$ at time t_k is defined as,

$$\widetilde{\mathbf{X}}_{k} = \begin{bmatrix} \widetilde{\mathbf{X}}^{\mathrm{T}}_{I_k} & {}^{C_{j-N+1}}_{G}\widetilde{\boldsymbol{\theta}}^{\mathrm{T}} & {}^{G}\widetilde{\mathbf{p}}^{\mathrm{T}}_{C_{j-N+1}} & {}^{C_{j-N+2}}_{G}\widetilde{\boldsymbol{\theta}}^{\mathrm{T}} & {}^{G}\widetilde{\mathbf{p}}^{\mathrm{T}}_{C_{j-N+2}} & \cdots & {}^{C_j}_{G}\widetilde{\boldsymbol{\theta}}^{\mathrm{T}} & {}^{G}\widetilde{\mathbf{p}}^{\mathrm{T}}_{C_j} \end{bmatrix}^{\mathrm{T}} \tag{4}$$

where,

$$\widetilde{\mathbf{X}}_{I_k} = \begin{bmatrix} {}^{I_k}_{G}\widetilde{\boldsymbol{\theta}}^{\mathrm{T}} & \widetilde{\mathbf{b}}^{\mathrm{T}}_{g,k} & {}^{G}\widetilde{\mathbf{v}}^{\mathrm{T}}_{I_k} & \widetilde{\mathbf{b}}^{\mathrm{T}}_{a,k} & {}^{G}\widetilde{\mathbf{p}}^{\mathrm{T}}_{I_k} & {}^{C}_{I}\widetilde{\boldsymbol{\theta}}^{\mathrm{T}} & {}^{I}\widetilde{\mathbf{p}}^{\mathrm{T}}_{C} \end{bmatrix}^{\mathrm{T}} \tag{5}$$

For modeling the process model and state augmentation, we follow [22].

2.3. Measurement Model

2.3.1. Vehicle Relative Rotation and Translation Estimation

From the geometric constraints depicted in Figure 2, the estimation of vehicle relative rotation and translation can be given by the following equation:

$$ {}^{B_{j-1}}_{B_j}\mathbf{T} = {}^{C}_{B}\mathbf{T}^{-1}\,{}^{G}_{C_{j-1}}\mathbf{T}^{-1}\,{}^{G}_{C_j}\mathbf{T}\,{}^{C}_{B}\mathbf{T} \tag{6} $$

Equation (6) can be represented as,

$$ \begin{aligned} C\left({}^{B_{j-1}}_{B_j}\mathbf{q}\right) &= C\left({}^{C}_{B}\mathbf{q}\right)^{\mathrm{T}} C\left({}^{C_{j-1}}_{G}\mathbf{q}\right) C\left({}^{C_j}_{G}\mathbf{q}\right)^{\mathrm{T}} C\left({}^{C}_{B}\mathbf{q}\right) \\ {}^{B_{j-1}}_{B_j}\mathbf{p} &= C\left({}^{C}_{B}\mathbf{q}\right)^{\mathrm{T}}\left[C\left({}^{C_{j-1}}_{G}\mathbf{q}\right)\left[C\left({}^{C_j}_{G}\mathbf{q}\right)^{\mathrm{T}}\,{}^{C}_{B}\mathbf{p} + {}^{G}_{C_j}\mathbf{p} - {}^{G}_{C_{j-1}}\mathbf{p}\right] - {}^{C}_{B}\mathbf{p}\right] \end{aligned} \tag{7} $$

where,

$$ \begin{aligned} C\left({}^{C}_{B}\mathbf{q}\right)^{\mathrm{T}} &= C\left({}^{B}_{I}\mathbf{q}\right) C\left({}^{C}_{I}\mathbf{q}\right)^{\mathrm{T}} \\ {}^{C}_{B}\mathbf{p} &= {}^{C}_{I}\mathbf{p} + {}^{I}_{B}\mathbf{p} \end{aligned} \tag{8} $$

where $C(\cdot)$ denotes the function converting the quaternion to the rotation matrix [41]. $C\left({}^{B}_{I}\mathbf{q}\right)$ and ${}^{I}_{B}\mathbf{p}$ denote rotation extrinsic parameters and translation extrinsic parameters between {B} and {I}, respectively. $C\left({}^{C}_{I}\mathbf{q}\right)$ and ${}^{C}_{I}\mathbf{p}$ denote rotation extrinsic parameters and translation extrinsic parameters between {C} and {I}, respectively. $C\left({}^{B_{j-1}}_{B_j}\mathbf{q}\right)$ and ${}^{B_{j-1}}_{B_j}\mathbf{p}$ are vehicle relative rotation and translation estimation from time t_j to t_{j-1}.

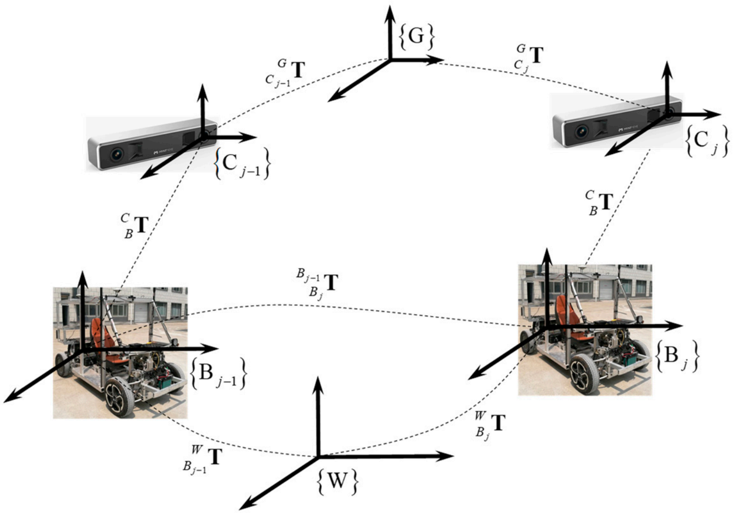


Figure 2. Geometric relation between {B} and {C} frames from time t_{j-1} to time t_j. The symbol ${}^{W}_{B_{j-1}}\mathbf{T}$ denotes the vehicle pose at time t_{j-1} in {W}, ${}^{W}_{B_j}\mathbf{T}$ denotes the vehicle pose at time t_j in {W},${}^{B_{j-1}}_{B_j}\mathbf{T}$ represents the homogeneous coordinate transformation from time t_j to time t_{j-1}, ${}^{C}_{B}\mathbf{T}$ represents the homogeneous coordinate transformation from {B} to {C}, ${}^{G}_{C_{j-1}}\mathbf{T}$ denotes the camera pose at time t_{j-1} in {G} and finally ${}^{G}_{C_j}\mathbf{T}$ denotes the camera pose at time t_j in {G}.

2.3.2. Ackermann Error State Measurements

After the state was augmented with the new camera state, the state is successively updated by Ackermann measurements and visual measurements. Here, the tightly-coupled Ackermann error state measurement model is present as the primary contribution of this paper.

Based on Equation (7), the measurement residual $\mathbf{r}^{B_j}$ can be defined as,

$$\mathbf{r}^{B_j} = \begin{pmatrix} \mathbf{r}^B_{\theta_{j-1,j}} \\ \mathbf{r}^B_{v_j} \end{pmatrix} = \begin{pmatrix} \mathbf{z}^B_{\theta_{j-1,j}} \cdot \mathrm{C}\left({}^{B_{j-1}}_{B_j}\mathbf{q}\right)^{-1} \\ \mathbf{z}^B_{v_j} - {}^{B_{j-1}}_{B_j}\mathbf{p}/\Delta t_{j-1,j} \end{pmatrix} \tag{9}$$

where $\mathbf{r}^B_{\theta_{j-1,j}}$ and $\mathbf{r}^B_{v_j}$ are the Ackermann steering measurement residual from time t_{j-1} to time t_j and the velocity measurement residual at time t_j respectively, and $\mathbf{z}^B_{\theta_{j-1,j}}$ and $\mathbf{z}^B_{v_j}$ are the Ackermann steering measurement and velocity measurement, respectively. The camera sampling period $\Delta t_{j-1,j} = t_j - t_{j-1}$. We assume that the velocity measurement $\mathbf{z}^B_{v_j}$ is constant for low-speed motion from time t_{j-1} to time t_j.

Considering the existence of nonholonomic constraint for a ground wheeled vehicle [43], the vehicle velocity in the plane perpendicular to the moving direction is considered as zero mean with Gaussian white noise. So, the velocity measurement $\mathbf{z}^B_{v_j}$ can be represented as,

$$\mathbf{z}^B_{v_j} = \begin{bmatrix} v^B_{x,j} & 0 & 0 \end{bmatrix}^{\mathrm{T}} \tag{10}$$

where $v^B_{x,j}$ is the vehicle longitudinal velocity at time t_j.

The Ackermann steering measurement $\mathbf{z}^B_{\theta_{j-1,j}}$ can be derived by the Ackermann steering geometry which is used in car-like wheeled mobile robots, and a closer look at the Ackermann steering geometry is illustrated in Figure 3. It is used to avoid wheel slippage by rotating the inner wheel with a larger angle than the outer wheel. As shown in Figure 3, the Instantaneous Center of Rotation (ICR) is located at the intersection of the extension cord of the automotive rear axle and the perpendicular bisectors of the front wheels [44].

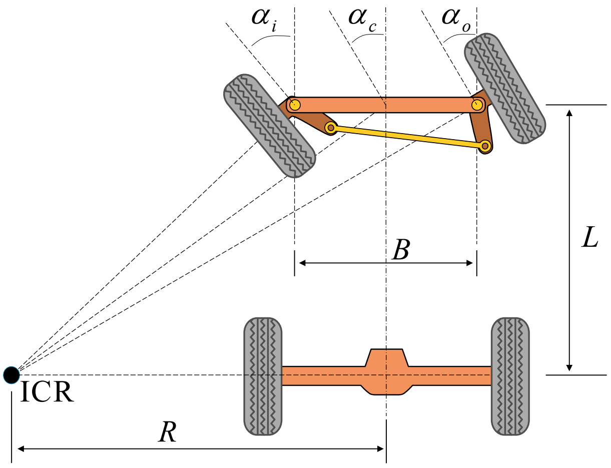


Figure 3. Ackermann steering geometry used in car-like wheeled mobile robots. B is the distance between the king pins, L is the wheel base (distance between the two axles), α_i is the steering angle of the inner wheel from straight ahead, α_c is the steering angle of the virtual wheel at the midpoint between the front wheels, α_o is the steering angle of the outer wheel from straight ahead and R is the steering radius (distance between the Instantaneous Center of Rotation (ICR) and the centerline of the vehicle). The sign of α_i, α_c, α_o and R follows the rule that a left turn assumes a positive value and a right turn a negative value in the coordinate system {B}.

It is assumed that vehicles follow the approximate planar motion, and there are no wheel slippage and sideslip angle for low-speed motion [45]. Therefore, the equation of Ackerman steering geometry then can be derived as follows,

$$\begin{gathered} \frac{L}{\tan|\alpha_o|} - \frac{L}{\tan|\alpha_c|} = \frac{B}{2} \\ R = \frac{L}{\tan\alpha_c} \end{gathered} \tag{11}$$

According to Equation (11), the kinematic relation between R and α_o is shown in:

$$R = \begin{cases} L/\tan\alpha_o - B/2 \text{ for } \alpha_o > 0 \\ L/\tan\alpha_o + B/2 \text{ for } \alpha_o < 0 \end{cases} \tag{12}$$

$$\alpha_o = \delta/\eta \tag{13}$$

where the steering wheel angle δ can be acquired from the vehicle CAN-bus. η is the angular transmission ratio obtained from the offline calibration in this paper, $\eta = 17$.

By knowing the vehicle longitudinal velocity v_x^B, R and the vehicle CAN-bus sampling time t_τ, the change of yaw angle $\phi_{j-1,j}^B$ from time t_{j-1} to time t_j is obtained [46]:

$$\phi_{j-1,j}^B = \int_{\tau\in[j-1,j]} v_{x,\tau}^B / R_\tau d\tau \tag{14}$$

Considering the nonholonomic constraint and the approximate planar motion assumption, the roll and pitch angle are considered as a zero mean with Gaussian white noise,

$$\theta_{j-1,j}^B = \begin{bmatrix} 0 & 0 & \phi_{j-1,j}^B \end{bmatrix}^{\mathrm{T}} \tag{15}$$

By using the Rodrigues rotation formula, the Ackermann steering measurement $\mathbf{z}_{\theta_{j-1,j}}^B$ can be represented by the exponential map of $\theta_{j-1,j}^B$ [42]:

$$\mathbf{z}_{\theta_{j-1,j}}^B = \exp\left(\left(\theta_{j-1,j}^B\right)^\wedge\right) \tag{16}$$

where the operator $(\cdot)^\wedge$ maps a vector to a skew symmetric matrix.

By linearizing Equation (9) at current error state $\widetilde{\mathbf{X}}_k$, $\mathbf{r}^{B_j}$ can be approximated as,

$$\mathbf{r}^{B_j} = \mathbf{H}^{B_j}\widetilde{\mathbf{X}}_k + \mathbf{n}^{B_j} \tag{17}$$

where the Ackermann error state measurement Jacobian matrix $\mathbf{H}^{B_j}$ can be derived as,

$$\mathbf{H}^{B_j} = \begin{bmatrix} 0_{3\times15} & \frac{\partial \mathbf{r}_{\theta_{j-1,j}}^B}{\partial {}_I^C\widetilde{\boldsymbol{\theta}}} & 0_{3\times3} & 0_{3\times(6N-6)} & \frac{\partial \mathbf{r}_{\theta_{j-1,j}}^B}{\partial {}_G^{C_j}\widetilde{\boldsymbol{\theta}}} & 0_{3\times3} \\ 0_{3\times15} & \frac{\partial \mathbf{r}_{v_j}^B}{\partial {}_I^C\widetilde{\boldsymbol{\theta}}} & \frac{\partial \mathbf{r}_{v_j}^B}{\partial {}^I\widetilde{\mathbf{p}}_C} & 0_{3\times(6N-6)} & \frac{\partial \mathbf{r}_{v_j}^B}{\partial {}_G^{C_j}\widetilde{\boldsymbol{\theta}}} & \frac{\partial \mathbf{r}_{v_j}^B}{\partial {}^G\widetilde{\mathbf{p}}_{C_j}} \end{bmatrix} \tag{18}$$

where

$$\begin{aligned}
\frac{\partial \mathbf{r}_{\theta_{j-1,j}}^B}{\partial {}_G^{C_j}\widetilde{\boldsymbol{\theta}}} &= -C\left({}_B^C\mathbf{q}\right)^{\mathrm{T}} C\left({}_G^{C_{j-1}}\mathbf{q}\right) C\left({}_G^{C_j}\mathbf{q}\right)^{\mathrm{T}} \\
\frac{\partial \mathbf{r}_{\theta_{j-1,j}}^B}{\partial {}_I^C\widetilde{\boldsymbol{\theta}}} &= C\left({}_B^C\mathbf{q}\right)^{\mathrm{T}} C\left({}_G^{C_{j-1}}\mathbf{q}\right) C\left({}_G^{C_j}\mathbf{q}\right)^{\mathrm{T}} - C\left({}_B^C\mathbf{q}\right)^{\mathrm{T}} \\
\frac{\partial \mathbf{r}_{v_j}^B}{\partial {}_G^{C_j}\widetilde{\boldsymbol{\theta}}} &= -C\left({}_B^C\mathbf{q}\right)^{\mathrm{T}} C\left({}_G^{C_{j-1}}\mathbf{q}\right) C\left({}_G^{C_j}\mathbf{q}\right)^{\mathrm{T}} \left\lfloor {}_B^C\mathbf{p}\times \right\rfloor / \Delta t_{j-1,j} \\
\frac{\partial \mathbf{r}_{v_j}^B}{\partial {}^G\widetilde{\mathbf{p}}_{C_j}} &= C\left({}_B^C\mathbf{q}\right)^{\mathrm{T}} C\left({}_G^{C_{j-1}}\mathbf{q}\right) / \Delta t_{j-1,j} \\
\frac{\partial \mathbf{r}_{v_j}^B}{\partial {}_I^C\widetilde{\boldsymbol{\theta}}} &= C\left({}_B^C\mathbf{q}\right)^{\mathrm{T}} \left\lfloor C\left({}_G^{C_{j-1}}\mathbf{q}\right)\left({}_{C_j}^G\mathbf{p} - {}_{C_{j-1}}^G\mathbf{p}\right)\times \right\rfloor / \Delta t_{j-1,j} \\
\frac{\partial \mathbf{r}_{v_j}^B}{\partial {}^I\widetilde{\mathbf{p}}_C} &= \left(C\left({}_B^C\mathbf{q}\right)^{\mathrm{T}} C\left({}_G^{C_{j-1}}\mathbf{q}\right) C\left({}_G^{C_j}\mathbf{q}\right)^{\mathrm{T}} - C\left({}_B^C\mathbf{q}\right)^{\mathrm{T}}\right) / \Delta t_{j-1,j}
\end{aligned} \tag{19}$$

Meanwhile, $\mathbf{n}^{B_j}$ in Equation (17) denotes the noise of the measurement, and the measurement covariance matrix of $\mathbf{n}^{B_j}$ is given by,

$$\mathbf{S}^{B_j} = \begin{bmatrix} \mathbf{S}^B_{\theta_{j-1,j}} & 0_{3\times3} \\ 0_{3\times3} & \mathbf{S}^B_{v_j} \end{bmatrix} \tag{20}$$

where $\mathbf{S}^B_{v_j}$= $\mathrm{diag}\left\{ \begin{matrix} \sigma^2_{v^B_x} & \sigma^2_{v^B_y} & \sigma^2_{v^B_z} \end{matrix} \right\}$, $\mathbf{S}^B_{\theta_{j-1,j}}$= $\mathrm{diag}\left\{ \begin{matrix} \sigma^2_{\theta^B_{roll}} & \sigma^2_{\theta^B_{pitch}} & \mathbf{S}^B_{\phi_{j-1,j}} \end{matrix} \right\}$, and $\mathbf{S}^B_{\phi_{j-1,j}}$ can be given by the following recursive formula,

$$\mathbf{S}^B_{\phi_{j-1,\tau}} = \mathbf{S}^B_{\phi_{j-1,\tau-1}} + \mathbf{V}_{\tau-1}\mathbf{M}\mathbf{V}^{\mathrm{T}}_{\tau-1} \tag{21}$$

where the input Jacobian matrix $V_{\tau-1}$ can be obtained according to Equation (14),

$$\begin{array}{l} \phi^B_{j-1,\tau} = \phi^B_{j-1,\tau-1} + \Delta t_{\tau-1,\tau} v^B_{x,\tau-1} / R_{\tau-1} \\ \mathbf{V}_{\tau-1} = \dfrac{\partial \phi^B_{j-1,\tau}}{\partial \begin{bmatrix} v^B_{x,\tau-1} & \delta_{\tau-1} \end{bmatrix}^{\mathrm{T}}} = \begin{bmatrix} V^{\phi}_{v^B_{x,\tau-1}} & V^{\phi}_{\delta_{\tau-1}} \end{bmatrix} \end{array} \tag{22}$$

where $\Delta t_{\tau-1,\tau} = t_\tau - t_{\tau-1}$ is the vehicle CAN-bus sampling period, $V^{\phi}_{v^B_{x,\tau-1}}$ and $V^{\phi}_{\delta_{\tau-1}}$ can be derived as,

$$\begin{array}{ll} V^{\phi}_{v^B_{x,\tau-1}} & = \frac{\Delta t_{\tau-1,\tau}}{R_{\tau-1}} \\ V^{\phi}_{\delta_{\tau-1}} & = \frac{L v^B_{x,\tau-1} \Delta t_{\tau-1,\tau} \left(\tan^2(\delta_{\tau-1}/\eta) + 1 \right)}{\eta (R_{\tau-1} \tan(\delta_{\tau-1}/\eta))^2} \end{array} \tag{23}$$

and the input noise $\mathbf{M}$ is given by,

$$\mathbf{M} = \begin{bmatrix} \sigma^2_{v^B_x} & 0 \\ 0 & \sigma^2_{\delta} \end{bmatrix} \tag{24}$$

where $\sigma_{v^B_x}$ and σ_δ are the standard deviations of vehicle longitudinal velocity and steering wheel angle, respectively. Meanwhile, $\sigma_{v^B_y}$ and $\sigma_{v^B_z}$ are the white-noise standard deviations of vehicle lateral velocity and vertical velocity, respectively. Moreover, $\sigma_{\theta^B_{roll}}$ and $\sigma_{\theta^B_{pitch}}$ are the white-noise standard deviations of vehicle roll angle and pitch angle, respectively. In this paper, $\sigma_{v^B_x} = 0.3$, $\sigma_{v^B_y} = 0.3$, $\sigma_{v^B_z} = 5$, σ_δ= 17, $\sigma_{\theta^B_{roll}} = 10 \cdot \sqrt{\mathbf{S}^B_{\phi_{j-1,j}}}$, $\sigma_{\theta^B_{pitch}} = 10 \cdot \sqrt{\mathbf{S}^B_{\phi_{j-1,j}}}$.

Finally, the Kalman gain, updated state vector and covariance matrix at time t_j can be obtained as follows,

$$\begin{array}{l} \mathbf{K} = \mathbf{P}_{k|k} \left(\mathbf{H}^{B_j} \right)^{\mathrm{T}} \left(\mathbf{H}^{B_j} \mathbf{P}_{k|k} \left(\mathbf{H}^{B_j} \right)^{\mathrm{T}} + \mathbf{S}^{B_j} \right)^{-1} \\ \mathbf{X}^+_k = \mathbf{X}_k + \mathbf{K}\mathbf{r}^{B_j} \\ \mathbf{P}^+_{k|k} = (\mathbf{I}_\xi - \mathbf{K}\mathbf{H}^{B_j}) \mathbf{P}_{k|k} (\mathbf{I}_\xi - \mathbf{K}\mathbf{H}^{B_j})^{\mathrm{T}} + \mathbf{K}\mathbf{S}^{B_j}\mathbf{K}^{\mathrm{T}} \end{array} \tag{25}$$

where $\mathbf{I}_\xi \in \mathbb{R}^{(21+6N)\times(21+6N)}$ represents the identity matrix.

2.3.3. Visual Measurements

In the visual front-end, the FAST feature detector and KLT optical flow tracker are used. The configuration of the visual front-end and the visual feature measurement model in this paper are the same as S-MSCKF. Please refer to [22] for further details.

2.4. Extrinsic Parameters Calibration

Accurate extrinsic parameters can greatly affect the accuracy and reliability of overall multi-sensors' fusion system performances. In this paper, the rotation extrinsic parameters $C\left({}^{C}_{I}\mathbf{q}\right)$ and translation extrinsic parameters ${}^{C}_{I}\mathbf{p}$ between {C} and {I} are calibrated by using the Kalibr visual-inertial calibration toolbox [47]. The translation extrinsic parameters ${}^{I}_{B}\mathbf{p}$ between {B} and {I} are obtained by the vehicle design model, and we adopt the singular value decomposition (SVD) [48] rotation calibration and weighted average quaternion method [49] to calibrate rotation extrinsic parameters $C\left({}^{B}_{I}\mathbf{q}\right)$ between {B} and {I}. A dual-antenna Spatial NAV982-RG inertial navigation system with real-time kinematic (GPS-RTK) is used to assist calibration. The GPS-RTK is fixed just above the automotive rear axle center and has the same axes as the {B}. Assuming that the installation error can be ignored, rotation extrinsic parameters $C\left({}^{B}_{I}\mathbf{q}\right)$ can be obtained by aligning two rotation sequences from GPS-RTK and frame {I}. The following equation holds for any γ,

$$ {}^{B_{\gamma-1}}_{B_\gamma}\mathbf{q} \otimes {}^{B}_{I}\mathbf{q} = {}^{B}_{I}\mathbf{q} \otimes {}^{I_{\gamma-1}}_{I_\gamma}\mathbf{q} \tag{26} $$

where ${}^{B_{\gamma-1}}_{B_\gamma}\mathbf{q}$ and ${}^{I_{\gamma-1}}_{I_\gamma}\mathbf{q}$ are unit quaternions of GPS-RTK rotation measurements and gyroscope integral measurements of frame {I} from time t_γ to time $t_{\gamma-1}$ respectively, and $\otimes$ is the quaternion multiplication operator.

Equation (26) can be represented as,

$$ \left[\Omega_L\left({}^{B_{\gamma-1}}_{B_\gamma}\mathbf{q}\right) - \Omega_R\left({}^{I_{\gamma-1}}_{I_\gamma}\mathbf{q}\right)\right] \cdot {}^{B}_{I}\mathbf{q} = \Omega_\gamma^{\gamma-1} \cdot {}^{B}_{I}\mathbf{q} = 0 \tag{27} $$

where

$$ \begin{aligned} \Omega_L\left({}^{B_{\gamma-1}}_{B_\gamma}\mathbf{q}\right) &= \begin{bmatrix} {}^{B_{\gamma-1}}_{B_\gamma}\mathbf{q}_w & -{}^{B_{\gamma-1}}_{B_\gamma}\mathbf{q}_{xyz}^{\mathrm{T}} \\ {}^{B_{\gamma-1}}_{B_\gamma}\mathbf{q}_{xyz} & {}^{B_{\gamma-1}}_{B_\gamma}\mathbf{q}_w\mathbf{I}_3 + \left\lfloor {}^{B_{\gamma-1}}_{B_\gamma}\mathbf{q}_{xyz}\times \right\rfloor \end{bmatrix} \\ \Omega_R\left({}^{I_{\gamma-1}}_{I_\gamma}\mathbf{q}\right) &= \begin{bmatrix} {}^{I_{\gamma-1}}_{I_\gamma}\mathbf{q}_w & -{}^{I_{\gamma-1}}_{I_\gamma}\mathbf{q}_{xyz}^{\mathrm{T}} \\ {}^{I_{\gamma-1}}_{I_\gamma}\mathbf{q}_{xyz} & {}^{I_{\gamma-1}}_{I_\gamma}\mathbf{q}_w\mathbf{I}_3 - \left\lfloor {}^{I_{\gamma-1}}_{I_\gamma}\mathbf{q}_{xyz}\times \right\rfloor \end{bmatrix} \end{aligned} \tag{28} $$

where $\Omega_L\left({}^{B_{\gamma-1}}_{B_\gamma}\mathbf{q}\right)$ and $\Omega_R\left({}^{I_{\gamma-1}}_{I_\gamma}\mathbf{q}\right)$ are matrix representations for left and right Hamilton [42] quaternion multiplication, and $\mathbf{q}_w$ and $\mathbf{q}_{xyz}$ are the real part and imaginary part of a quaternion. respectively.

By collecting a period of rotation sequences, the overconstrained linear system can be represented as,

$$ \begin{bmatrix} \Omega_1^0 \\ \Omega_2^1 \\ \vdots \\ \Omega_\gamma^{\gamma-1} \end{bmatrix} {}^{B}_{I}\mathbf{q} = \Omega_\gamma \cdot {}^{B}_{I}\mathbf{q} = 0 \tag{29} $$

where time t_γ keeps going until the rotation calibration is completed.

Equation (29) can be solved by using SVD. Then, the right unit singular vector corresponding to the smallest singular value of Ω_γ is the solution. We set a threshold of iterations N_s, and the rotation calibration process terminates if $\gamma > N_s$.

After n rounds of rotation calibration with random sampling of rotation sequences, the weighted average quaternion [50] is utilized to obtain rotation extrinsic parameters ${}^{B}_{I}\mathbf{q}$,

$$ {}^{B}_{I}\mathbf{q} = \underset{\mathbf{q}\in\mathbb{S}^3}{\mathrm{argmax}}\,\mathbf{q}^T\mathbf{W}\mathbf{q} \tag{30} $$

where $\mathbb{S}^3$ denotes the unit 3-sphere, and $\mathbf{W}$ is given by,

$$\mathbf{W} = \sum_{i=1}^{n} {}^{\mathrm{sec}}\sigma_i \cdot {}^{B}_{I}\mathbf{q}_i \cdot {}^{B}_{I}\mathbf{q}_i^T \tag{31}$$

where ${}^{B}_{I}\mathbf{q}_i$ denotes rotation calibration solution in the i^{th} round, ${}^{\mathrm{sec}}\sigma_i$ is the second smallest singular value of Ω_γ in the i^{th} round.

Equation (30) can be solved by using SVD, the rotation extrinsic parameters ${}^{B}_{I}\mathbf{q}$ are the eigenvector of $\mathbf{W}$ corresponding to the maximum eigenvalue.

3. Experiments and Results

The experimental vehicle setup and experimental scenarios are first introduced in this section. Secondly, the experimental results are exhibited for different vehicle driving cycles.

3.1. Experimental Vehicle Setup

To the best of our knowledge, it would be impractical to get public datasets collected by Ackermann steering vehicles in an outdoor environment, which contains both vehicle CAN-bus data, visual data, IMU data and ground truth. The public datasets used for VO and VIO are known as the EuRoC MAV Dataset [51], RGBD SLAM Dataset [52], KITTI Dataset [53] and PennCOSYVIO Dataset [54]. The EuRoC MAV Dataset is mainly suitable for MAVs applications; the RGBD SLAM dataset only contains RGBD camera data; the PennCOSYVIO Dataset is collected by handheld devices; and the KITTI Dataset is acquired by an Ackermann steering vehicle. However, the data does not contain time synchronous vehicle CAN-bus data. Therefore, the performance of the proposed method is evaluated by collecting real-world data.

The experimental vehicle is an Ackermann steering vehicle with dual in-wheel motors drive designed by our research group. As shown in Figure 4, the experimental vehicle can be remote controlled by using the smart phone application (APP) developed by our research group. Telematics BOX (T-BOX) is mainly used for communicating with the smart phone APP and Vehicle Control Unit (VCU) to realize the display and remote control of the experimental vehicle. The datasets were collected in the outdoor environment of the test field. We used the MYNTEYE-S1030 stereo camera (MYNTAI Inc., Beijing, China) with a focal length of 8 mm as the visual-inertial sensor. The camera with 720 pixels × 480 pixels resolution and 30 Hz frame rate is equipped with a time-synchronized IMU. The acquisition frequency of the IMU is 200 Hz. The vehicle longitudinal speed and steering wheel angle were acquired from the vehicle CAN-bus with sampling frequency 150 Hz by using the Controller Area Network (CAN) interface card.

A dual-antenna Spatial NAV982-RG inertial navigation system was used, which combines the FindCM positioning service from Qianxun Spatial Intelligence Inc. for centimeter-level positioning of the vehicle as the experimental ground truth.

The data acquisition was performed on a domestic mini-computer (Shenzhen Yingchi Technology Co., Ltd, Shenzhen, China) running on an Intel Core i3-5005U CPU with 2.00 GHz, 4 GB RAM (Intel, Santa Clara, CA, USA). The algorithm evaluation was performed on a Lenovo laptop (Lenovo, Beijing, China) running on an Intel Core i5-8300H CPU with 2.30 GHz, 8 GB RAM (Intel, Santa Clara, CA, USA). The data acquisition software and entire algorithm were implemented in C++ programming language (Free Software Foundation, Inc., Boston, Mass., USA) using ROS Kinetic (Open Source Robotics Foundation, Inc., Mountain View, CA, USA). We evaluated the trajectories using the methods provided by [55].

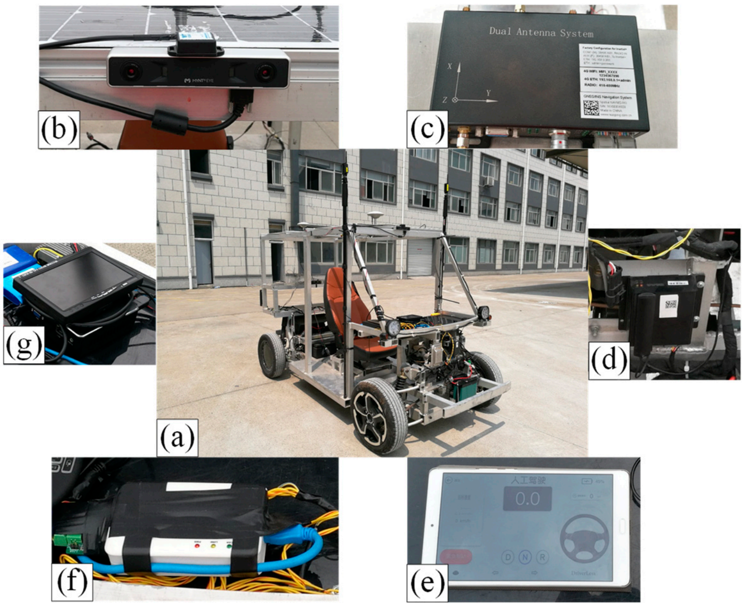


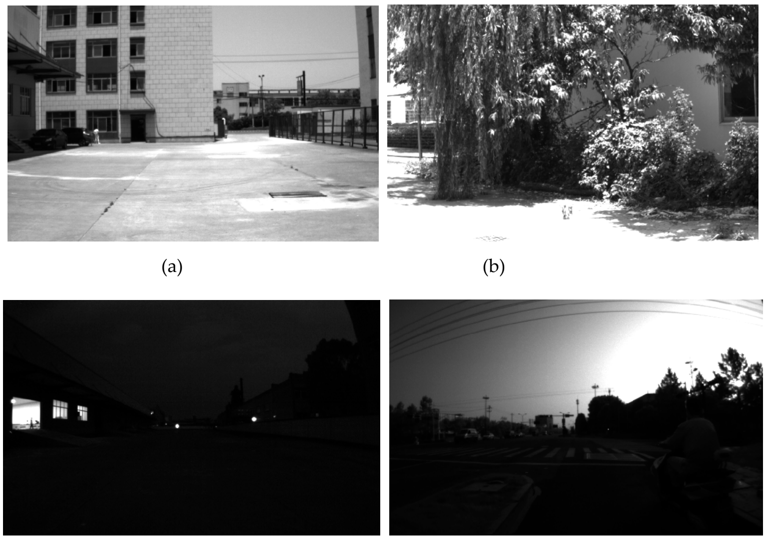
Figure 4. Experimental vehicle setup: (**a**) The Ackermann steering vehicle built by our research group; (**b**) MYNTEYE-S1030; (**c**) dual-antenna Spatial NAV982-RG; (**d**) telematic box ((communicating with the smart phone application (APP) and Vehicle Control Unit (VCU)); (**e**) smart phone APP; (**f**) Controller Area Network (CAN) interface card; (**g**) mini-computer for the data acquisition.

3.2. Experimental Results

Taking into account the different vehicle driving cycles and visually distinctive environmental features, in order to evaluate the algorithm more comprehensively, five different experimental datasets were acquired; i.e., the AM_01 dataset, AM_02 dataset, AM_03 dataset, AM_04 dataset and AM_05 dataset. AM_01 and AM_02 were respectively acquired under the straight line driving cycle and slalom line driving cycle by traveling around the building in a test field counterclockwise for one circle in the forenoon. In both of these datasets, there exist strong changes of illumination against the sunlight and motion blur at the sharp turns. AM_03 was acquired under the low-light environment at dusk which poses a great challenge for the VIO visual front-end. AM_04 with the dataset including road dynamic objects, such as other vehicles and pedestrians, was acquired under a long-distance driving cycle on the trafficway. AM_05 was acquired under a free driving cycle to verify the extrinsic parameters calibration method proposed in Section 2.4. The details of the experimental datasets are shown in Table 1, and Figure 5 shows several acquired images in the AM_01, AM_02, AM_03 and AM_04 datasets.

Table 1. List of our experimental datasets.

Dataset	Vehicle Driving Cycles	Environmental Condition	Duration (s)	Data Bulk			
				Vehicle CAN-Bus	Stereo Images	IMU	Ground Truth
AM_01	Straight	High-light	151	24,171	4532	30,164	30,244
AM_02	Slalom	Motion blur	170	27,135	5099	33,933	34,025
AM_03	Free	Low-light	131	17,387	3930	26,258	13,111
AM_04	Free	Dynamic objects	493	65,147	14,801	98,651	49,291
AM_05	Free	High-light	238	38,020	7136	47,439	47,583



(a) (b)

(c) (d)

Figure 5. Acquired images in the experimental datasets. (**a**) AM_01; (**b**) AM_02; (**c**) AM_03; (**d**) AM_04.

3.2.1. Extrinsic Parameters Calibration Performance

The original S-MSCKF was utilized to evaluate the extrinsic parameters calibration performance on the AM_05 dataset. S-MSCKF with the calibration method proposed in Section 2.4 is named S-MSCKF(s), and S-MSCKF with its rotation extrinsic parameter manually measured is named S-MSCKF(m).

Figure 6 shows that the performance from S-MSCKF(s) and S-MSCKF(m) on AM_05 dataset. For quantitative evaluation, we chose the absolute translation error (ATE) and the relative error (RE) as the evaluation metric. The open-source package [55] offers an off-the-shelf interface to perform a comprehensive evaluation of VIO. A yaw-only transformation with only the first state was used for trajectory alignment according to this package. Figure 6a,b show the top-view and side-view of the estimated trajectory, respectively. As presented in Figure 6c, the boxplots of the relative translation error are expressed in percentages by dividing the error with the sub-trajectory length. The boxplots of the relative yaw error are displayed in Figure 6d.

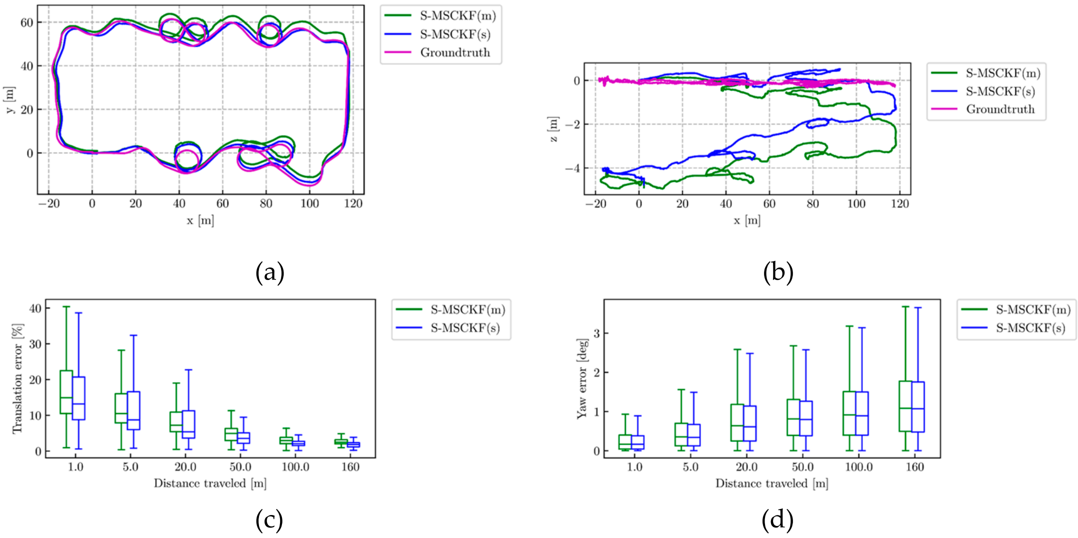


Figure 6. The performance from the Stereo Multi-State Constraint Kalman Filter (S-MSCKF)(s) (blue line), S-MSCKF(m) (green line) and ground truth (magenta line) on the AM_05 dataset. (**a**) Top-view of estimated trajectory; (**b**) Side-view of estimated trajectory; (**c**) The relative translation error; (**d**) The relative yaw error. Relative errors are measured over different sub-trajectories of length {1, 5, 20, 50, 100, 160} m.

From Figure 6a, it can be seen that the estimated trajectory of S-MSCKF(s) aligns to the ground truth better than S-MSCKF(m). Obviously from Figure 6c, S-MSCKF(s) performs better than S-MSCKF(m) on relative translation error. Meanwhile, Figure 6b illustrates both methods exhibit large scale drift along the gravitational direction.

The absolute translation error on the AM_05 dataset is shown in Table 2, and the relative translation error and relative yaw error on the AM_05 dataset are shown in Table 3.

Table 2. Absolute translation error on AM_05 dataset.

Methods	RMSE (m)
S-MSCKF(s)	3.05
S-MSCKF(m)	4.21

Table 3. The relative error on AM_05 dataset.

Sub-trajectory Length (m)	Relative Translation Error (m)		Relative Yaw Error (deg)	
	S-MSCKF(s)	**S-MSCKF(m)**	**S-MSCKF(s)**	**S-MSCKF(m)**
1	0.17	0.19	0.25	0.27
5	0.61	0.67	0.46	0.48
20	1.56	1.76	0.76	0.79
50	1.94	2.39	0.91	0.93
100	2.20	2.96	1.00	1.02
160	2.97	4.14	1.18	1.20

It can be seen from the quantitative results in Tables 1 and 2 that the absolute translation error of S-MSCKF(s) is reduced by 27.55% of S-MSCKF(m). The relative translation error and relative yaw error of S-MSCKF(s) are less than S-MSCKF(m) over all sub-trajectory lengths. By comprehensive analysis of Tables 1 and 2, S-MSCKF(s) performs better than S-MSCKF(m), indicating that the proposed extrinsic parameters' calibration method can significantly improve the position accuracy.

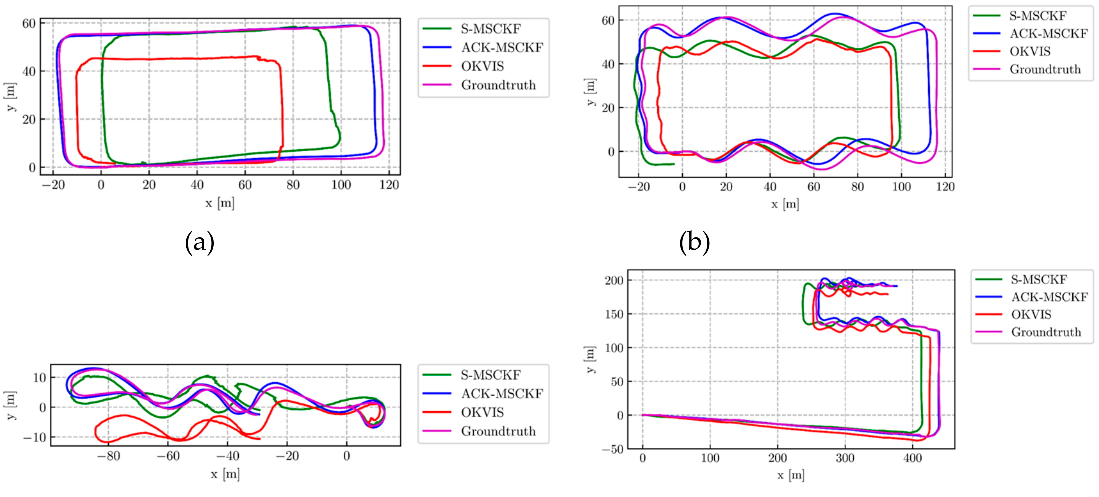
3.2.2. ACK-MSCKF Performance

The performance of ACK-MSCKF is compared with state-of-the-art open source stereo VIO, S-MSCKF and OKVIS on the AM_01, AM_02, AM_03 and AM_04 datasets acquired by our experimental vehicle. The extrinsic parameters obtained by our proposed calibration method are used for all of these algorithms.

Figures 7 and 8 show the trajectory estimation results in a top view and aligned to Google Earth, respectively.

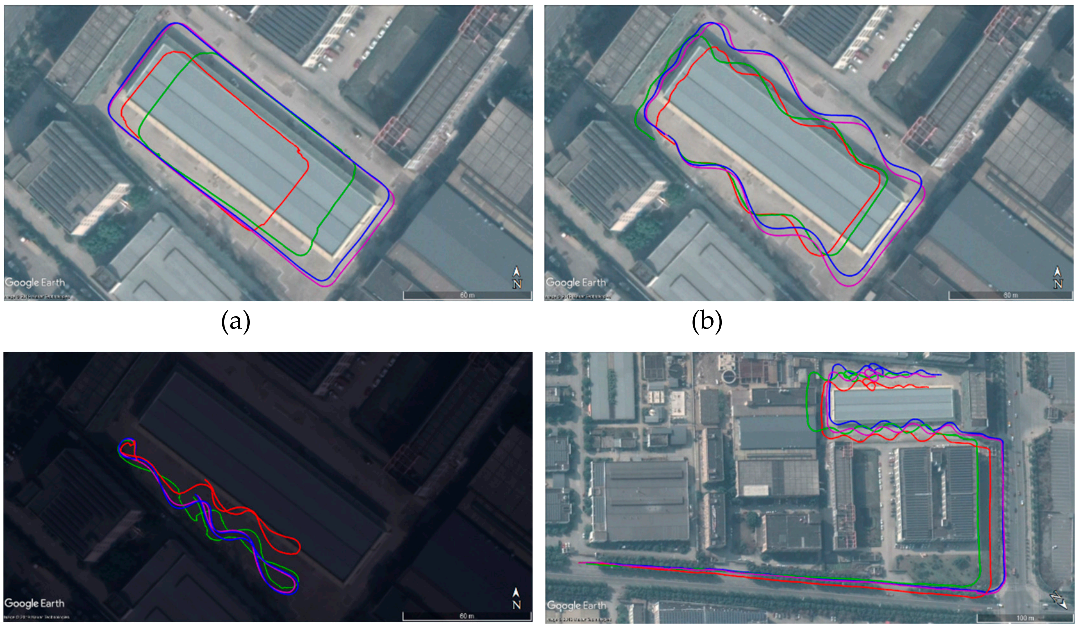
From Figure 7 we can see that the estimated trajectories of our proposed ACK-MSCKF algorithm align to the ground truth better than others. Meanwhile, it can be seen that the position estimation results of S-MSCKF and OKVIS have larger scale drift, especially in the straight line driving cycle and slalom line driving cycle (Figure 7a,b). Figure 7c suggests that OKVIS has larger y-axis error in the low-light environment.

By fusing Ackermann error state measurements and S-MSCKF, it can be seen intuitively that the accuracy of the trajectory estimation is significantly improved.



(a) (b) (c) (d)

Figure 7. The top-views of estimated trajectory from ACK-MSCKF (blue line), S-MSCKF (green line), Open Keyframe-based Visual-Inertial SLAM (OKVIS) (red line) and ground truth (magenta line) of the dataset: (**a**) AM_01; (**b**) AM_02; (**c**) AM_03; (**d**) AM_04.

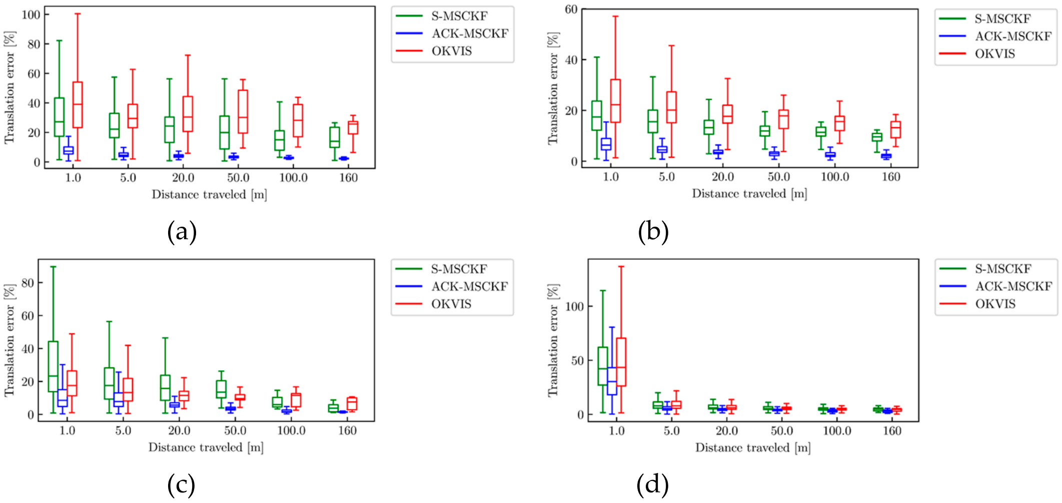


(a) (b) (c) (d)

Figure 8. The estimated trajectories aligned to Google Earth from ACK-MSCKF (blue line), S-MSCKF (green line), OKVIS (red line) and ground truth (magenta line) of the dataset: (**a**) AM_01; (**b**) AM_02; (**c**) AM_03; (**d**) AM_04.

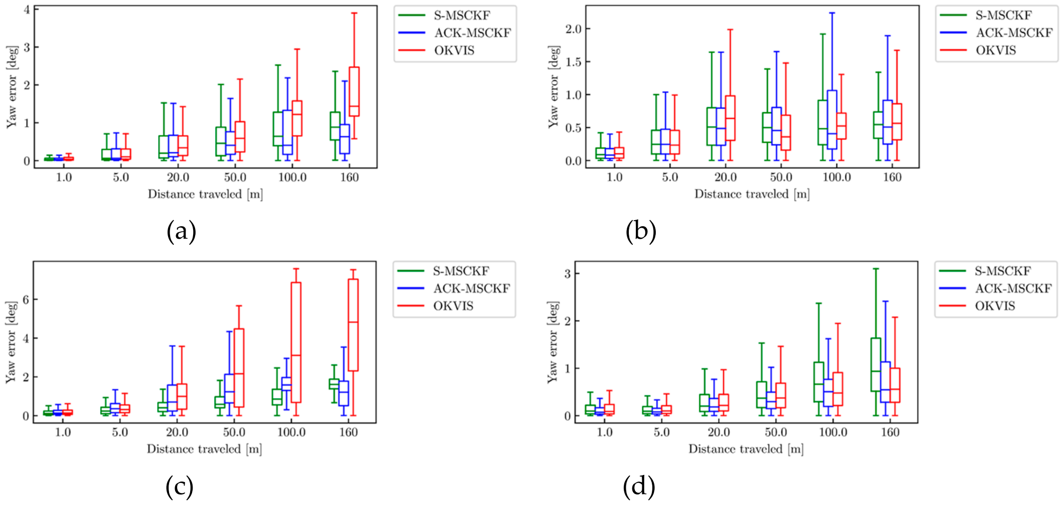
Figure 8 shows that the trajectory of ground truth (magenta line) is highly identical with the outline of our test field in Google Earth. Therefore, the ground truth has high credibility to evaluate the performance of our algorithm.

The boxplots of the relative translation error are shown in Figure 9. Figure 10 presents the boxplots of the relative yaw error.



(a) (b) (c) (d)

Figure 9. The relative translation error from ACK-MSCKF (blue line), S-MSCKF (green line) and OKVIS (red line) of the dataset: (**a**) AM_01; (**b**) AM_02; (**c**) AM_03; (**d**) AM_04.



(a) (b) (c) (d)

Figure 10. The relative yaw error from ACK-MSCKF (blue line), S-MSCKF (green line) and OKVIS (red line) of the dataset: (**a**) AM_01; (**b**) AM_02; (**c**) AM_03; (**d**) AM_04.

Figure 9 demonstrates that the relative translation error of OKVIS on AM_01, AM_02 and AM_04 datasets is at the largest within the sub-trajectory length of 5 m, while S-MSCKF has the largest relative translation error within the sub-trajectory length of 50 m on the AM_03 dataset. Meanwhile, there exists larger scale drift, especially in the straight line driving cycle and slalom line driving cycle (Figure 9a,b).

It can be seen from Figure 10 that OKVIS has the largest relative yaw error within the sub-trajectory length of 20 m on all of datasets. The relative yaw error of OKVIS increases significantly along with the sub-trajectory length increases, especially in the straight line driving cycle and low-light environment (Figure 10a,c). S-MSCKF performs best on the relative yaw error within the sub-trajectory length of 100 m on the AM_03 dataset.

It can be concluded from Figures 9 and 10 that the relative translation error of the ACK-MSCKF algorithm is significantly reduced, and the relative yaw error is close to S-MSCKF within the sub-trajectory length of 5 m.

The absolute translation error along the whole trajectory of each experimental dataset is shown in Table 4, and the overall relative translation error and relative yaw error on all of the real-world datasets are reflected in Table 5.

Table 4. Absolute translation error along the whole trajectory of each experimental dataset.

Methods	AM_01 (m)	AM_02 (m)	AM_03 (m)	AM_04 (m)
ACK-MSCKF	2.25	3.43	1.57	5.13
S-MSCKF	15.24	12.26	5.23	21.66
OKVIS	24.62	13.01	9.68	13.63

Table 5. The overall relative error on all of the real-world datasets.

Sub-trajectory Length (m)	Relative Translation Error (m)			Relative Yaw Error (deg)		
	ACK-MSCKF	S-MSCKF	OKVIS	ACK-MSCKF	S-MSCKF	OKVIS
1	0.25	0.41	0.46	0.14	0.15	0.36
5	0.34	0.75	0.80	0.21	0.20	0.41
20	0.97	2.48	2.65	0.43	0.39	0.68
50	1.85	4.99	5.88	0.59	0.55	0.87
100	2.99	7.70	9.99	0.74	0.78	1.11
160	4.18	10.41	12.94	0.80	1.04	1.33

It can be seen from the quantitative results in Table 4 that the absolute translation error of ACK-MSCKF is the least on all of the datasets. There exists a larger absolute translation error, especially under the straight line driving cycle and outdoor long-distance driving cycle on the road. By analyzing the quantitative results in Table 5, it can be obtained that the relative translation error of S-MSCKF and OKVIS increases significantly along with the sub-trajectory length increases. At the same time, it illustrates that the relative yaw error of S-MSCKF is less than OKVIS over different sub-trajectory lengths.

By comprehensive analysis of Tables 4 and 5, we can see that the relative translation errors of ACK-MSCKF in different sub-trajectory lengths are significantly reduced, and it is 4.18 m in the sub-trajectory length of 160 m, which is reduced respectively by 59.85% and 67.70% under the same condition of S-MSCKF and OKVIS. Meanwhile, the relative yaw error of ACK-MSCKF is close to S-MSCKF. The least relative yaw error from ACK-MSCKF is 0.80 deg in the sub-trajectory length of 160 m, which is reduced respectively by 23.08% and 39.85% under the same condition of S-MSCKF and OKVIS.

4. Discussion

From the experimental results, this can all be summarized as the following:

Accurate extrinsic parameters calibration greatly affects the accuracy and reliability of overall system performances. The extrinsic parameters calibration method proposed in this paper can significantly improve the position accuracy with the absolute translation error reduced by 27.55%.

S-MSCKF and OKVIS confront the problem of scale unobservability under special motions of autonomous vehicles. The reasons can be explained as follows. Visual features in outdoor scenes are mostly far away relative to the stereo baseline, making the stereo camera degenerated into a monocular one. Besides, the scale unobservability under special motions of ground vehicles attached to monocular VIO cannot be neglected.

ACK-MSCKF can achieve accurate and robust pose estimation under different vehicle driving cycles and environmental conditions, which mainly benefits from the improvement of observability under special motions of autonomous vehicles by fusing Ackermann error state measurements. The relative translation error of ACK-MSCKF is reduced respectively by 59.85% and 67.70% of S-MSCKF and OKVIS in the sub-trajectory length of 160 m. Moreover, the relative yaw error is reduced respectively by 23.08% and 39.85% under the same condition.

5. Conclusions

In conclusion, this paper proposes a tightly-coupled Ackermann visual-inertial odometry as ACK-MSCKF. The Ackermann steering geometry and Ackermann error state measurement model are illustrated in detail. Moreover, the rotation extrinsic parameters calibration method is presented. The performance is evaluated on real-world datasets acquired by an Ackermann steering vehicle. The experimental results demonstrate that the proposed ACK-MSCKF improves the pose estimation accuracy of S-MSCKF under special motions of autonomous vehicles, and can maintain the accuracy and robustness of pose estimation available under different vehicle driving cycles and environmental conditions. This work shows the guiding significance in the study of the high-precision pose estimation of Ackermann steering vehicles through an in-vehicle proprioceptive sensor and visual inertial sensor with a tightly-coupled filter-based mechanism.

However, the limitation of this work derives from the assumption that vehicles follow approximate planar motion without wheel slippage and sideslip angle for low-speed operation. In the future, we will further study the Ackermann error state measurement model-aided VIO under non-planar motion, and use incremental optimization-based algorithms [56–59] to further improve the pose estimation accuracy of autonomous vehicles.

Author Contributions: Conceptualization, F.M. and J.S.; Data curation, J.S.; Funding acquisition, F.M.; Investigation, J.S.; Methodology, J.S.; Project administration, F.M.; Resources, F.M. and K.D.; Software, J.S.; Supervision, F.M.; Validation, J.S.; Visualization, J.S.; Writing-original draft, J.S.; Writing-review & editing, F.M., J.S., Y.Y. and J.L.

Funding: This research was funded by the Jilin Province Key Technology and Development Program (Grant No. 20190302077GX), the National Natural Science Foundation of China (Grant No. 51705185) and the National Key Technologies R&D Program of China (Grant No. 2017YFC0601604).

Acknowledgments: We would like to thank the support received from Linhe Ge and Shouren Zhong. We acknowledge http://www.shenlanxueyuan.com/ for providing professional training and technical exchange about the latest algorithms in VIO.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Lee, K.; Li, S.E.; Kum, D. Synthesis of Robust Lane Keeping Systems: Impact of Controller and Design Parameters on System Performance. *IEEE Trans. Intell. Transport. Syst.* **2018**, *20*, 3129–3141. [CrossRef]
2. Lee, K.; Chung, W.; Yoo, K. Kinematic parameter calibration of a car-like mobile robot to improve odometry accuracy. *Mechatronics* **2010**, *20*, 582–595. [CrossRef]
3. Toledo, J.; Piñeiro, J.; Arnay, R.; Acosta, D.; Acosta, L. Improving odometric accuracy for an autonomous electric cart. *Sensors* **2018**, *18*, 200. [CrossRef]
4. Groves, P.D. *Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems*; Artech House: Norwood, MA, USA, 2013.
5. Xue, H.; Fu, H.; Dai, B. IMU-Aided High-Frequency Lidar Odometry for Autonomous Driving. *Appl. Sci.* **2019**, *9*, 1506. [CrossRef]
6. Zhang, J.; Singh, S. Visual-lidar odometry and mapping: Low-drift, robust, and fast. In Proceedings of the 2015 IEEE International Conference on Robotics and Automation (ICRA), Seattle, WA, USA, 26–30 May 2015; pp. 2174–2181.
7. Mur-Artal, R.; Tardós, J.D. Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras. *IEEE Trans. Robot.* **2017**, *33*, 1255–1262. [CrossRef]
8. Engel, J.; Koltun, V.; Cremers, D. Direct sparse odometry. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *40*, 611–625. [CrossRef]
9. Forster, C.; Zhang, Z.; Gassner, M.; Werlberger, M.; Scaramuzza, D. SVO: Semidirect visual odometry for monocular and multicamera systems. *IEEE Trans. Robot.* **2017**, *33*, 249–265. [CrossRef]
10. Younes, G.; Asmar, D.; Shammas, E.; Zelek, J. Keyframe-based monocular SLAM: Design, survey, and future directions. *Robot. Auton. Syst.* **2017**, *98*, 67–88. [CrossRef]

11. Aqel, M.O.A.; Marhaban, M.H.; Saripan, M.I.; Ismail, N.B. Review of visual odometry: Types, approaches, challenges, and applications. *Springerplus* **2016**, *5*, 1897. [CrossRef]
12. Saputra, M.R.U.; Markham, A.; Trigoni, N. Visual SLAM and Structure from Motion in Dynamic Environments: A Survey. *ACM Comput. Surv.* **2018**, *51*, 37. [CrossRef]
13. Aladem, M.; Rawashdeh, S. Lightweight visual odometry for autonomous mobile robots. *Sensors* **2018**, *18*, 2837. [CrossRef] [PubMed]
14. Huang, G. Visual-inertial navigation: A concise review. *arXiv* **2019**, arXiv:1906.02650.
15. Gui, J.; Gu, D.; Wang, S.; Hu, H. A review of visual inertial odometry from filtering and optimisation perspectives. *Adv. Robot.* **2015**, *29*, 1289–1301. [CrossRef]
16. Mur-Artal, R.; TardÓs, J.D. Visual-Inertial Monocular SLAM with Map Reuse. *IEEE Robot. Autom. Lett.* **2017**, *2*, 796–803. [CrossRef]
17. Von Stumberg, L.; Usenko, V.; Cremers, D. Direct sparse visual-inertial odometry using dynamic marginalization. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, Australia, 21–25 May 2018; pp. 2510–2517.
18. Ramezani, M.; Khoshelham, K. Vehicle Positioning in GNSS-Deprived Urban Areas by Stereo Visual-Inertial Odometry. *IEEE Trans. Intell. Veh.* **2018**, *3*, 208–217. [CrossRef]
19. Mourikis, A.I.; Roumeliotis, S.I. A multi-state constraint Kalman filter for vision-aided inertial navigation. In Proceedings of the 2007 IEEE International Conference on Robotics and Automation, Roma, Italy, 10–14 April 2007; pp. 3565–3572.
20. Lynen, S.; Achtelik, M.W.; Weiss, S.; Chli, M.; Siegwart, R. A robust and modular multi-sensor fusion approach applied to MAV navigation. In Proceedings of the 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, Tokyo, Japan, 3–7 November 2013; pp. 3923–3929.
21. Bloesch, M.; Omari, S.; Hutter, M.; Siegwart, R. Robust visual inertial odometry using a direct EKF-based approach. In Proceedings of the 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Hamburg, Germany, 28 September–2 October 2015; pp. 298–304.
22. Sun, K.; Mohta, K.; Pfrommer, B.; Watterson, M.; Liu, S.; Mulgaonkar, Y.; Taylor, C.J.; Kumar, V. Robust stereo visual inertial odometry for fast autonomous flight. *IEEE Robot. Autom. Lett.* **2018**, *3*, 965–972. [CrossRef]
23. Huai, Z.; Huang, G. Robocentric visual-inertial odometry. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018; pp. 6319–6326.
24. Leutenegger, S.; Lynen, S.; Bosse, M.; Siegwart, R.; Furgale, P. Keyframe-based visual–inertial odometry using nonlinear optimization. *Int. J. Robot. Res.* **2014**, *34*, 314–334. [CrossRef]
25. Qin, T.; Li, P.; Shen, S. VINS-Mono: A Robust and Versatile Monocular Visual-Inertial State Estimator. *IEEE Trans. Robot.* **2018**, *34*, 1004–1020. [CrossRef]
26. He, Y.; Zhao, J.; Guo, Y.; He, W.; Yuan, K. Pl-vio: Tightly-coupled monocular visual-inertial odometry using point and line features. *Sensors* **2018**, *18*, 1159. [CrossRef]
27. Usenko, V.; Demmel, N.; Schubert, D.; Stückler, J.; Cremers, D. Visual-Inertial Mapping with Non-Linear Factor Recovery. *arXiv* **2019**, arXiv:1904.06504.
28. Delmerico, J.; Scaramuzza, D. A benchmark comparison of monocular visual-inertial odometry algorithms for flying robots. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, Australia, 21–25 May 2018; pp. 2502–2509.
29. Wu, M.; Lam, S.; Srikanthan, T. A Framework for Fast and Robust Visual Odometry. *IEEE Trans. Intell. Transport. Syst.* **2017**, *18*, 3433–3448. [CrossRef]
30. Wu, K.J.; Roumeliotis, S.I. *Unobservable Directions of VINS under Special Motions*; Department of Computer Science & Engineering, University of Minnesota: Minneapolis, MN, USA, 2016.
31. Shan, Z.; Li, R.; Schwertfeger, S. RGBD-Inertial Trajectory Estimation and Mapping for Ground Robots. *Sensors* **2019**, *19*, 2251. [CrossRef] [PubMed]
32. Schmid, K.; Tomic, T.; Ruess, F.; Hirschmüller, H.; Suppa, M. Stereo vision based indoor/outdoor navigation for flying robots. In Proceedings of the 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, Tokyo, Japan, 3–7 November 2013; pp. 3955–3962.
33. Zheng, F.; Liu, Y.H. SE(2)-Constrained Visual Inertial Fusion for Ground Vehicles. *IEEE Sens. J.* **2018**, *18*, 9699–9707. [CrossRef]
34. Wu, K.J.; Guo, C.X.; Georgiou, G.; Roumeliotis, S.I. Vins on wheels. In Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA), Singapore, 29 May–3 June 2017; pp. 5155–5162.

35. Dang, Z.; Wang, T.; Pang, F. Tightly-coupled Data Fusion of VINS and Odometer Based on Wheel Slip Estimation. In Proceedings of the 2018 IEEE International Conference on Robotics and Biomimetics (ROBIO), Kuala Lumpur, Malaysia, 12–15 December 2018; pp. 1613–1619.
36. Hong, E.; Lim, J. Visual-Inertial Odometry with Robust Initialization and Online Scale Estimation. *Sensors* **2018**, *18*, 4287. [CrossRef] [PubMed]
37. Houseago, C.; Bloesch, M.; Leutenegger, S. KO-Fusion: Dense Visual SLAM with Tightly-Coupled Kinematic and Odometric Tracking. In Proceedings of the 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 20–24 May 2019.
38. Huskić, G.; Buck, S.; Zell, A. GeRoNa: Generic Robot Navigation. *J. Intell. Robot. Syst.* **2018**, *95*, 419–442. [CrossRef]
39. ACK-MSCKF ROS Package. Available online: https://github.com/qdensh/ACK-MSCKF (accessed on 8 October 2019).
40. Breckenridge, W.G. Quaternions proposed standard conventions. *Jet Propuls. Lab. Pasadena CA Interoff. Memo. IOM* **1999**, 343–379.
41. Trawny, N.; Roumeliotis, S.I. Indirect Kalman filter for 3D attitude estimation. *Univ. Minn. Dept. Comp. Sci. Eng. Tech. Rep* **2005**, *2*, 2005. [CrossRef]
42. Sola, J. Quaternion kinematics for the error-state Kalman filter. *arXiv* **2017**, arXiv:1711.02508.
43. Scaramuzza, D.; Fraundorfer, F.; Pollefeys, M.; Siegwart, R. Absolute scale in structure from motion from a single vehicle mounted camera by exploiting nonholonomic constraints. In Proceedings of the 2009 IEEE 12th International Conference on Computer Vision, Kyoto, Japan, 29 September–2 October 2009; IEEE: Piscataway, NJ, USA, 2009; pp. 1413–1419.
44. Kevin, M.L.; Park, F.C. *Modern Robotics: Mechanics, Planning, and Control*; Cambridge University Press: Cambridge, UK, 2017.
45. Rajamani, R. *Vehicle Dynamics and Control*, 2nd ed.; Springer: New York, NY, USA, 2012.
46. Marín, L.; Vallés, M.; Soriano, Á.; Valera, Á.; Albertos, P. Multi sensor fusion framework for indoor-outdoor localization of limited resource mobile robots. *Sensors* **2013**, *13*, 14133–14160. [CrossRef]
47. Furgale, P.; Rehder, J.; Siegwart, R. Unified temporal and spatial calibration for multi-sensor systems. In Proceedings of the 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, Tokyo, Japan, 3–7 November 2013; pp. 1280–1286.
48. Choi, S.; Park, J.; Yu, W. Simplified epipolar geometry for real-time monocular visual odometry on roads. *Int. J. Control Autom. Syst.* **2015**, *13*, 1454–1464. [CrossRef]
49. Yang, Z.; Shen, S. Monocular visual-inertial state estimation with online initialization and camera-IMU extrinsic calibration. *IEEE Trans. Autom. Sci. Eng.* **2016**, *14*, 39–51. [CrossRef]
50. Markley, F.L.; Cheng, Y.; Crassidis, J.L.; Oshman, Y. Averaging Quaternions. *J. Guid. Control Dyn.* **2007**, *30*, 1193–1197. [CrossRef]
51. Burri, M.; Nikolic, J.; Gohl, P.; Schneider, T.; Rehder, J.; Omari, S.; Achtelik, M.W.; Siegwart, R. The EuRoC micro aerial vehicle datasets. *Int. J. Robot. Res.* **2016**, *35*, 1157–1163. [CrossRef]
52. Sturm, J.; Engelhard, N.; Endres, F.; Burgard, W.; Cremers, D. A benchmark for the evaluation of RGB-D SLAM systems. In Proceedings of the 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, Vilamoura, Portugal, 7–12 October 2012; pp. 573–580.
53. Geiger, A.; Lenz, P.; Urtasun, R. Are we ready for autonomous driving? the kitti vision benchmark suite. In Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Providence, RI, USA, 16–21 June 2012; pp. 3354–3361.
54. Pfrommer, B.; Sanket, N.; Daniilidis, K.; Cleveland, J. Penncosyvio: A challenging visual inertial odometry benchmark. In Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA), Singapore, 29 May–3 June 2017; pp. 3847–3854.
55. Zhang, Z.; Scaramuzza, D. A tutorial on quantitative trajectory evaluation for visual (-inertial) odometry. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018; pp. 7244–7251.
56. Kaess, M.; Ranganathan, A.; Dellaert, F. ISAM: Incremental Smoothing and Mapping. *IEEE Trans. Robot.* **2008**, *24*, 1365–1378. [CrossRef]
57. Kaess, M.; Johannsson, H.; Roberts, R.; Ila, V.; Leonard, J.J.; Dellaert, F. ISAM2: Incremental smoothing and mapping using the Bayes tree. *Int. J. Robot. Res.* **2012**, *31*, 216–235. [CrossRef]

58. Liu, H.; Chen, M.; Zhang, G.; Bao, H.; Bao, Y. Ice-ba: Incremental, consistent and efficient bundle adjustment for visual-inertial slam. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 1974–1982.
59. Li, X.; Wang, Y.; Khoshelham, K. Comparative analysis of robust extended Kalman filter and incremental smoothing for UWB/PDR fusion positioning in NLOS environments. *Acta Geod. Geophys.* **2019**, *54*, 157–179. [CrossRef]

© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).