SEAMLESS REALTIME LANE LEVEL VEHICULAR NAVIGATION IN GNSS CHALLENGING ENVIRONMENTS USING A RTK GNSS/IMU/VINS INTEGRATION SCHEME

Kai-Wei Chiang [1], Cheng-Xian Lin [1]*, Syun Tsai [1], Chi-Hsin Huang [1], Meng-Lun Tsai [2]
[1] Dept. of Geomatics, National Cheng Kung University, Tainan, Taiwan - kwchiang@geomatics.ncku.edu.tw, e9971217@gmail.com, dino920135@gmail.com, windstorm@geomatics.ncku.edu.tw
[2] High Definition Map Research Center, Dept. of Geomatics, National Cheng Kung University, Tainan, Taiwan - taurusbryant@geomatics.ncku.edu.tw

KEY WORDS: Global Navigation Satellite System, Inertial Navigation System, Sensor Integration, GNSS High-Challenge, Visual Odometry

ABSTRACT:
Outdoor positioning requires a reliable solution that can work in environments where satellite signals are often blocked or degraded. Global Navigation Satellite System (GNSS) is a common choice, but it may not provide accurate results for land vehicles. To address this challenge, this research proposes a multi-sensor integrated system for vehicle navigation that combines GNSS with other sensors. The system uses Extended Kalman Filter (EKF) to fuse the data from different sources and improve the navigation performance. The algorithm targets to provide seamless navigation for urban environments as well as various indoor environments fields with INS/GNSS/VIO aiding integrated solutions. The experimental vehicle of this research is equipped with a tactical-grade inertial sensing measurement unit (IMU) as the test system, a self-designed and assembled visual platform, which includes a camera with a time synchronization protocol and a low-cost IMU. Also, both indoor experimental fields and outdoor urban scenarios with different high challenging were tested to verify the developed algorithm. To evaluate the performance of the proposed real-time navigation system, we use a high-accuracy navigation-grade system as a reference, which provides a stable and reliable trajectory. The result indicates that using the GNSS RTK solution with VIO aiding integration scheme reduced the RMS errors in long outage (450 sec, 1812 m) by 87.4% and 79.9% in position and velocity error, respectively. In urban scenario, the along-track/cross-track maximum errors can achieve 1.4 m / 1.5 m. Overall, these contribute to the development of real-time navigation systems for self-driving vehicle in the future.

1. INTRODUCTION

Global Navigation Satellite System (GNSS) is the user-preferred solution for the requirement of outdoor positioning. However, in the land vehicle application environment, the satellite signal is often obscured and denied, which further reduces the accuracy of the satellite positioning solution. Therefore, it is imperative to develop automotive solutions that can be applied to the hostile satellite signals environment. The Inertial Navigation System (INS)/GNSS integration solution is widely used in the positioning and navigation applications of land vehicles. This research proposes a multi-sensor integrated system for vehicle navigation adopting an Extended Kalman Filter (EKF). The proposed algorithm aims to provide seamless navigation for urban environments as well as various indoor environments with INS/GNSS/VIO aiding integrated solutions. This research focuses on the integration of visual-inertial odometry (VIO) and INS/GNSS system in Real Time Kinematic (RTK) mode.

The major contributions of adopting VIO into INS/GNSS are:

1. VIO provides continuous measurement and uses image and IMU constraints to reduce the accumulation of errors of the main receiver IMU.
2. VIO brings higher performance to an IMU/GNSS system that uses a MEMS-level IMU.
3. Compared to lidar, VIO can reduce the overall cost of future autonomous driving payloads.
4. The integration scheme becomes more suitable for navigation in unknown or GNSS-denied environments.

This research contributes to the performance enhancement of traditional positioning algorithms aided by cameras and inertial sensors and is dedicated to achieving accuracy at the level of "which lane".

2. RELATED WORKS

Visual Inertial Odometry (VIO), also known as Visual Inertial Navigation System (VINS), is developed from Visual Odometry (VO) (Zhang & Ye, 2020). The problem of estimating displacement using visual information alone was first proposed in the 1980s, and the term VO was proposed (Nistér et al., n.d.). VIO makes VO more robust by adding a complementary IMU with the camera. Compared with GPS, VIO can still provide pose estimation indoors and does not suffer from the urban canyon scenario effect that degrades accuracy with near urban high buildings. Compared with LiDAR, the sensors required for VIO are relatively light, inexpensive, and have the means to provide higher frequency pose estimation. A major limitation of the monocular visual odometry method is the inability to determine the scale factor. This problem is inherent to the monocular approach and cannot be solved without additional information. The image frames processed by monocular VO easily lose the depth information of the environment. Although this process recovers three-dimensional information, it does not yield the actual physical scale of the depth recovery. It is necessary to provide benchmarks for scale recovery through other external sources such as GNSS. The accuracy of triangulation based on visual features depends on the frame-to-frame displacement. If the camera rotates approximately, the triangulation algorithm deteriorates and causes feature point tracking to fail. By comparing the advantages and disadvantages of monocular visual and IMU, the fusion of camera and IMU has good complementary characteristics (Aqel et al., 2016). The steps for fusion IMU and VO can be described briefly as follow. First, by aligning the pose sequence estimated by the IMU with the pose sequence estimated by the camera, we can estimate the actual scale of the camera/IMU platform trajectory. Second, the IMU can predict the image frame pose and the feature point location in the next image frame, which enhances the feature tracking

* Corresponding author

algorithm's speed and robustness against fast rotation. Finally, by using the current gravity vector from the IMU accelerometer can transform the estimated position into the navigation coordinate system required for actual navigation. The VIO process can be decomposed into several parts, including internal and external orientation parameter calibration, feature extraction, data pre-processing, initialization and alignment, and nonlinear optimization, which will be explained in the following sections. Due to the slight error in the production process of the camera lens, as well as the difference in focal length and lens shape. The difference between the captured image and the real world is called lens distortion, including radiation distortion and tangential distortion. Using the camera calibration technology, the corresponding internal orientation parameters can be accurately determined. Then the original image with deformation can be corrected so that the relationship between the image point and the corresponding real-world object point satisfy the collinearity condition. In this research, calibrating the visual platform of the camera with the Kalibr toolbox (Qin & Shen, 2018). The following equations represent the radial distortion and tangential distortion of the distortion point, respectively.

$$\begin{cases} x_{radial} = x(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) \\ y_{radial} = y(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) \end{cases} \quad (1)$$

$$\begin{cases} x_{tangential} = x + [2p_1 xy + p_2(r^2 + 2x^2)] \\ y_{tangential} = y + [p_1(r^2 + 2y^2) + 2p_2 xy] \end{cases} \quad (2)$$

where

k_1, k_2, k_3 = The radial distortion coefficients.

p_1, p_2 = The tangential distortion coefficients.

As well as camera, IMU calibration is also important. IMU calibration is performed using the Allen variance technique to characterize the random measurement and process noise of the accelerometer and gyroscope. This method requires collecting static data for a long duration to obtain reliable results. In this study, more than 10 hours of data were. Then the imu_utils toolbox were used to estimate random walk and bias instability. These coefficients are essential for modelling the IMU errors and improving the accuracy of the sensor fusion algorithm. After the camera and IMU are calibrated individually, continue to use the Kalibr toolbox to calibrate the visual platform designed in this research. After receiving the data, consider the calibrated camera interior orientation parameters and the random walk, bias instability of the IMU, and input the image and IMU data into the Kalibr toolbox to estimate the exterior orientation parameters (Qin & Shen, 2018). These are the rotation and translation relationships between the camera and the IMU. One of the key steps in visual odometry (VO) and visual-inertial odometry (VIO) is feature extraction, which builds the correspondence between images in a sequence. A common method for feature extraction is corner detection, which identifies salient points in different images. However, storing the features as gray value matrices is not robust enough for complex environments. Therefore, more advanced algorithms have been developed to extract local image features that are invariant to scale, rotation and illumination changes. Such as Harris corner (Harris & Stephens, n.d.), FAST corner (Rosten & Drummond, 2006), etc. However, corners are not robust enough for most applications, and more stable local image features are needed, such as the Scale-Invariant Feature Transform (SIFT) (Lowe, 2004), Speeded Up Robust Features (SURF) (Bay et al., 2006), Oriented FAST and Rotated BRIEF (ORB) (Rublee et al., 2011) and so on. Feature points using feature extraction for images are composed of key-points and descriptors. The ORB is one of the most classic feature representatives, which is composed of the key point FAST and the descriptor BRIEF (Calonder et al., n.d.). The feature point method, ORB, estimates camera motion using algorithms such as triangulation, epipolar geometry, bundle adjustment (BA). Its high calculation accuracy is particularly prominent in Simultaneous Localization and Mapping (SLAM) applications. On the other hand, there is another method for extracting image feature points, the optical flow method (Lucas & Kanade, n.d.). The optical retention method is different from the feature point method in that only calculate key-points without descriptors. This reduces the computational cost of feature point matching, which is advantageous for real-time applications. In the literature, many VO and VIO frameworks have been developed. Especially in the VIO framework, some of them can solve the state by optimization and others can use a filtering process. The earliest monocular VO framework is Mono-SLAM, which is based on EKF. Since the number of feature points in the state of VO is much larger than the number of poses, and the system state dimension will increase significantly over time, it will make Mono-SLAM only suitable for sparse feature point environments (Davison et al., 2007). Parallel Tracking and Mapping (PTAM) was the first successful application of nonlinear optimization in VO (Klein & Murray, 2007). PTAM proposed the keyframe mechanism, which reduces the number of images that need to be optimized and ensures the real-time performance of the algorithm, which is the design basis of many current VO open-source frameworks. ORB-SLAM2 is also a framework for calculating keyframes based on the bundle adjustment method, which retains the advantages of fast detection of FAST features (Mur-Artal & Tardos, 2016). At the same time, the image pyramid algorithm is used to ensure scale invariance and the gray centroid is used to provide orientation description to ensure rotation invariance. However, its calculation can be time-consuming and resource-consuming. Different from feature extraction method, the direct method, which can build sparse, semi-dense, and dense point maps based on feature points to achieve the purpose of tracking. Based on the direct method, the typical VO representation of the semi-dense direct method is proposed as Large-Scale Direct SLAM (LSD-SLAM) (Engel et al., n.d.). It uses the image gradient to approximate the optical flow transform and effectively reduces the computational time-consuming issue by ignoring the pixel information in the region where the grayscale change is not obvious or the depth information is difficult to estimate. However, Semidirect Visual Odometry (SVO) is a VO framework based on the sparse direct method (Forster et al., 2014). SVO first detects the key points in the image based on the FAST corner detector and then estimates the camera pose by tracking the pixels within the 4×4 pixel range around the key points. The biggest advantage of SVO is lightweight. Subsequently, a sparse version of the direct method VO algorithm is proposed, which is Direct Sparse Odometry (DSO). DSO performs pose estimation and construction based on pixel points with large changes in image gray intensity and corrects the accumulated error through the relationship between geometric and photometric perspectives (Engel et al., 2016). This advantage of DSO is that it has good robustness under the weak texture. The IMU/camera VIO framework based on Kalman filtering uses the IMU positioning data to assist in determining the depth information of image feature points, which effectively improves the scale uncertainty of monocular VO. The Multi-State Constraint Kalman Filter (MSCKF) is a filter-based VIO algorithm (Mourikis & Roumeliotis, 2007). MSCKF uses EKF to fuse IMU and visual information, which can adapt to more intense motion and texture loss for a certain period of time, and maintain higher robustness. Robust Visual Inertial Odometry (ROVIO) is a visual state estimator based on EKF, which takes image photometric error as EKF innovation, and decomposes feature points into direction vectors and distance representation

to improve robustness (Bloesch et al., 2015). Open Keyframe-based Visual-Inertial SLAM (OKVIS) is a VIO algorithm that uses a sliding window of keyframe poses for nonlinear (Leutenegger et al., n.d.). Older keyframes or newer redundant image frames are marginalized before the state estimator to keep the window lightweight. Similar to OKVIS, VINS-Mono tracks robust corner features based on a nonlinear optimized sliding window estimator (Qin et al., n.d.). Processing pre-integration of IMU measurements, and 4DOF pose graph optimization are proposed. The overall VINS-Mono architecture can be divided into the following three parts:

1. Preprocessing, using the optical flow method to track the feature points of the sequence images, and pre-integrating the measurement data of the IMU at the same time.
2. Parameter initialization, compare the estimated pose of the camera with the pre-integrated pose of the IMU, and complete the initialization of the velocity vector, the gravity vector, and the scale factor, as well as the calibration of the gyro bias.
3. BA estimation optimization, based on the sliding window algorithm to marginalize distant keyframes to ensure the real-time performance of VIO estimation.

VI-DSO is a method for estimating camera pose and sparse scene geometry by minimizing photometric and IMU measurement errors, as well as the optimization-based method (von Stumberg et al., 2018). It can track any pixel with a sufficiently large intensity gradient, but it does not easily maintain the global map to affect accuracy. Finally, ORB-SLAM3 is an optimization-based tightly coupled VIO method, which can navigate for a long term with weak visual information with high robustness (Campos et al., 2021). However, it can be time-consuming and resource-consuming when computing, as the same ORB-SLAM2. In this study, the review of the state-of-the-art keyframe-based and filter-based VO confirms that without the intervention of the IMU, the scale issue cannot support long-term and real-time stable operation. Further, the filtering-based and optimization-based VIO frameworks are reviewed. VINS-Mono performs excellent and is stable in the real-time application, while outperforming other types of VIO frameworks in accuracy and robustness. For these reasons, the VIO algorithm used in this research is designed based on VINS-Mono, as shown in Figure 1.

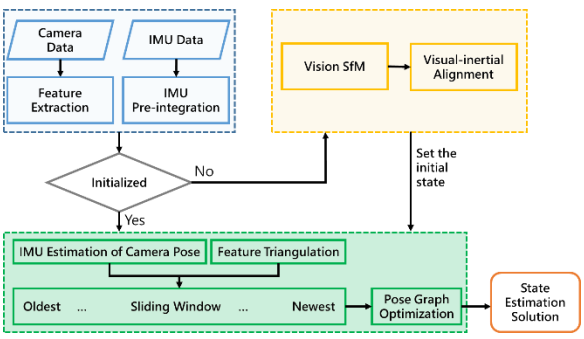

Figure 1 The flowchart of VINS-Mono process

This visual inertial odometry starts with measurement preprocessing, where features are extracted for tracking, and pre-integration of IMU measurements between two consecutive frames is performed. The initialization process provides all necessary values, including attitude, velocity, gravity vector, gyroscope bias, and 3D feature position, for guiding the subsequent nonlinear optimization-based VIO. And the scale of the visual odometry can be recovered by aligning the IMU measurements with the vision-only structure from motion (SfM) (Schönberger & Frahm, n.d.). Finally, the pose graph optimization module accepts the geometrically validated sliding-window relocalization results and performs the global optimization to eliminate drift to obtain state estimation solution.

3. METHODOLOGY

In this study, we analysed the performance of traditional approaches including INS/GNSS/Camera, and vision navigation by using the proposed visual platform sensors.

3.1 Coordinate Frames

We now define notations and frame definitions that we use throughout this paper. Figure 2 depicts the coordinate frames used in this study. The navigation frame (N) is a locally defined frame and in this study, the north–east–down (NED) frame is chosen, the origin of which is located at the center of a platform for which the *x*-axis and *y*-axis are aligned with the Earth's north and east directions, respectively. The *z*-axis is orthogonal to a reference ellipsoid that points downward the center of the Earth. In the vehicle coordinate system (V), the *x*-axis, *y*-axis, and *z*-axis point in the forward direction, to the right, and in the downward direction of the vehicle, respectively. We assumed that the transformation matrix (R_b^v) between the V frame and the IMU sensor frame (B) would be an identity matrix and that the effect of the lever-arm between the IMU sensor and the vehicle center would be ignorable. The transformation matrix between two frames is denoted by quaternion $q_{c_1}^{c_2}$. Finally, in the camera coordinate system (C), the *x*-axis, *y*-axis, and *z*-axis represent the left to right axis of image, top to bottom axis of image direction, and optical axis of camera, respectively. To estimate the 3D coordinates of the features in each image, it requires transformation that contains extrinsic and intrinsic parameters.

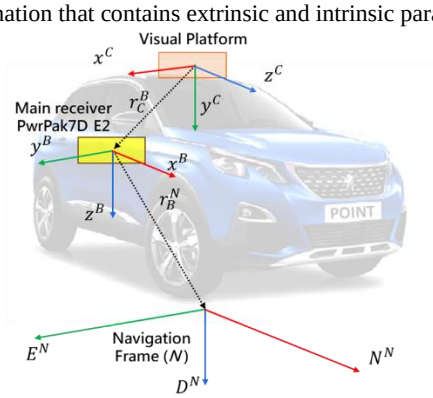

Figure 2 Geometric relationship between each component.

3.2 INS/GNSS/Camera (VINS) Integration Scheme

In this section, the measurement state content and fusion strategy provided by VIO are discussed in detail and a checking mechanism is designed to ensure that all measurements provided by auxiliary sensors meet predetermined criteria before going into the fusion algorithm. In addition, the proposed model for VINS velocity update is developed to improve some important issue areas in traditional positioning. The concept of the proposed INS/GNSS/Camera integration scheme.

3.2.1 Visual Inertial Odometry Integrated Algorithm

The VIO applied in this research is VINS-Mono (Qin, T. et al. 2018), which uses only one camera and one low-cost IMU with the self-developed and assembled vision platform. This method has been proven to be a robust and versatile monocular visual-inertial state estimation. The state vector contains:

$$x_{vins} = \left[r_{b_k}^N, v_{b_k}^N, q_{b_k}^N, b_a, b_g, p_c^b, q_c^b, \lambda_0, \lambda_1, \ldots \lambda_m\right], k \in [0, n] \quad (3)$$

where
k = It represents the kth image.
p_c^b = The translation between camera frame and IMU.
λ_l = It is the inverse distance of the lth feature from its first observation.
m = The total number of features in the sliding window.

This equation also shows that, to get a good estimation of this state, the premise is the joint calibration (images and IMU) that obtain good interior orientation parameters (IOPs) and exterior orientation parameters (EOPs), and then a good initialization through the specific movement. Thanks to the fact that the features of most outdoor scenarios and indoor parking lot scenarios are repeatable and abundant so the extracting and tracking procedure by the VINS algorithm is not difficult. This not only provides a robust position but also limits drift caused by IMU. In this algorithm, estimator initialization is carried out firstly the scale of the visual odometry can be recovered by aligning the IMU measurements with the vision structure from motion. Mathematically, given extrinsic parameters (p_c^b, q_c^b) between the camera and the IMU, the translate poses from the camera to IMU sensor frame are derived by:

$$q_{b_k}^{c0} = q_{b_k}^{c0} \otimes (\mathrm{q_c^b})^{-1} \quad (4)$$

$$\mathrm{s\bar{p}_{b_k}^{c_0}} = \mathrm{s\bar{p}_{c_k}^{c_0}} - \mathrm{R_{b_k}^{c_0} p_c^b} \quad (5)$$

where
s = The scale obtains by visual-inertial alignment.
c_0 = It represents the first camera frame is used as the reference frame for SfM.
q_{c0}^N = The derived refined gravity vector.

In this research, using provide the heading angle to coincide local level frame to n-frame. In this algorithm state estimation process, the IMU is applied to accurately predict the pose of the image frame and the position of the feature points in successive frames. It is worth mentioning that this method improves the matching speed of the feature tracking algorithm and the robustness against fast camera rotation to support real-time applications. For the establishment of this visual inertial odometry, the roll angle and pitch angle are completely observable, and the cumulative drift is only for four degrees of freedom (x, y, z, heading angle), so the pose graph is only for these four degrees of freedom optimizations. Finally, the corresponding velocity $v_{b_k}^N$ is obtained and further utilized in the proposed EKF using Equation (6). Since in-vehicle navigation applications usually do not match the premise of loop closure scenarios, the loop closure thread is not used in this research.

$$\delta Z_{vins} = H_{vins}\delta x_{vins} + \varepsilon_{vins} \quad (6)$$

$$H_{vins} = [0_{3\times3} \quad I_{3\times3} \quad 0_{3\times3} \quad 0_{3\times3} \quad 0_{3\times3} \quad 0_{3\times3} \quad 0_{3\times3}] \quad (7)$$

Where
δZ_{vins} = The vector of residual of velocity between current solution and VINS solution.
H_{vins} = The design matrix derived from the linearization process to connect the navigation states and measurement values.
ε_{vins} = The VINS position and velocity error in the n-frame.

On the other hand, for the VIO aiding application of this research, in terms of visual aid information processing, the measured coordinate vector and rotation matrix are transformed from the visual platform to the main receiver (Pwrpak7D-E2); this is the coordinate vector and the rotation matrix from the c-frame to the b-frame (lever-arm and boresight). And then calculated aiding data converted to the n-frame for this research to integrate the navigation application of the system.

3.2.2 GNSS Measurement Condition Assessment

This VIO innovation sequence can indicate the quality of the GNSS signal based on the VIO innovation sequence. By setting a threshold for the VIO innovation sequence, the EKF can reject the GNSS measurements that are affected by multipath or non-line-of-sight (NLOS), and only update the EKF with reliable measurements. On the other hand, if the velocity derived from the VIO is authorized, then the difference between the GNSS velocity and the velocity derived from the VIO ($\Delta Velocity$) must also be smaller than the velocity threshold, which means that the GNSS position is consistent with the VIO position and can be trusted. Conversely, it can be derived for possible effects, such as multipath, NLOS, and other sources, which means that the GNSS position is affected by some errors and should be rejected. The condition assessment of the design is shown in Equation (8).

$$\begin{cases} if\ Velocity_{inno} \le \theta_{velocity} \\ if(VIO)\ \&\ \Delta Velocity \le \theta_{velocity} \end{cases} is\ authorized. \quad (8)$$

3.2.3 VIO Scale Coefficient Estimation

As described before, the visual-inertial odometry (VIO) system that combines the camera and the inertial measurement unit (IMU) can recover the scale factor in real time. However, due to the limitations of the IMU, there may be some drift in the scale estimation over long distances. With each duration during the run, it must extract the INS/GNSS solution in the open sky area and calculate its scale coefficient. And at the same time meet the requirement of satisfaction below the designed GNSS Standard Deviation (STD) threshold. The VIO can then be estimated using this scaling coefficient by the following Equation:

$$s_a = \frac{\|d^{INS/GNSS}\|}{\|d^{VIO}\|} \quad (9)$$

where
s_a = The estimated scale.
$\|d^{INS/GNSS}\|$ = The distance is derived from the INS/GNSS position solution.
$\|d^{VIO}\|$ = The distance is derived from the VIO position solution.
This scale coefficient estimation method can be applied into real-time applications. Therefore, the scale estimation mechanism designed in this research is shown in Figure 3. The flowchart of this work can be explained as follows.

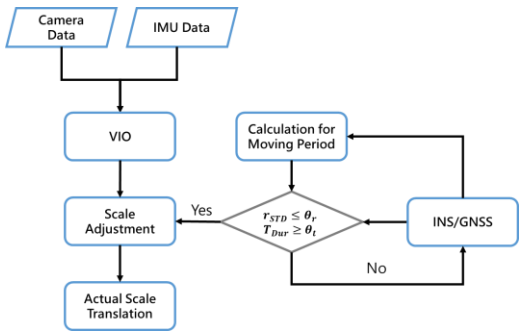

Figure 3 Flowchart of INS/GNSS-based VIO scale adjustment

The INS/GNSS solution while moving in open sky areas is used for the standard deviation empirical threshold of position θ_r, and the design period threshold θ_t. For the following INS/GNSS solutions, their position r_{STD} will be compared to θ_r and the duration T_{Dur} must exceed θ_t. If these conditions are met, the

scale coefficients calculated from the INS/GNSS previous stable solutions will be used to update the scaling adjustment process.

3.2.4 VIO Velocity Update

The proposed fusion algorithm employs measurements from the VIO, such as velocity measurements. These measurements with different drift behaviors could reduce the drift issue in the INS-only case and enhance the detection and rejection of contaminated measurements. Consequently, the velocity update is calculated based on the difference between the two frame positions and the time interval. The velocity of VIO-derived can be written as follows:

$$v_{VIO}^{n} = \frac{\hat{r}_{b_k}^{n} - \hat{r}_{b_{k-1}}^{n}}{t_k - t_{k-1}} \quad (10)$$

where

v_{VIO}^{n} = The velocity information in the navigation frame.

$\hat{r}_{b_k}^{n}$ = The position of VIO at the time kth frame.

Assuming that the measurements of velocity are independent and uncorrelated, which covariance matrix can be defined as:

$$R_{velocity} = J_{velocity} P^{VIO} J_{velocity}{}^{T} \quad (11)$$

where

P^{VIO} = The VIO covariance matrix.

$R_{velocity}$ = The measurement covariance matrices of velocity.

3.2.5 The proposed INS/GNSS/Camera Integration Scheme

The proposed integration scheme is shown in Figure 4. Navigation states from GNSS and INS are fused through an Extended Kalman Filter (EKF), while high output rates and continuous INS trigger prediction equations each epoch. Utilizing camera/IMU source measurements for updating the EKF, these will be described separately. Inputs are the sensor measurements and motion constrain, output is the navigation state, including the position, velocity, and attitude of the filtered solution. In this scheme, IMU measurements are processed using INS mechanization to provide a navigation solution in position, velocity, and attitude in the navigation frame. GNSS provides the absolute position as the primary survey update. For the camera, the VINS-Mono method constructed with several characteristics was chosen to process the data and provide measurement updates in this research. It is including image feature optical flow tracking, IMU data pre-integration, vision-only SfM, sliding window-based nonlinear optimization to achieve tight coupling, and 4DoF pose graph optimization, so on. To provide the velocity measurements as a minor update.

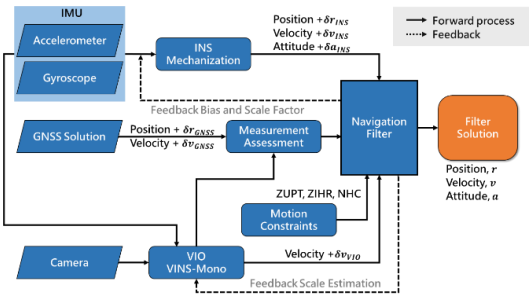

Figure 4 Flowchart of INS/GNSS/Camera fusion algorithm

In general, velocity is one of the important states for land vehicle applications. Velocity measurements could provide information for estimating IMU error state (such as bias) in EKF. It is clearer to identify the state of the vehicle based on the velocity of the vehicle. For example, if the velocities in the three directions are below a certain threshold, it can be considered static to allow the application of motion constraints, such as zero velocity update (ZUPT) or zero integrated heading rate (ZIHR) in EKF. This information also allows users to identify or check for outliers in other measurements. This velocity update assists with overall navigation solutions, especially in GNSS-hostile environments.

4. EXPERIMENT

The experiment setups were show as Figure 5 The test system data was obtained using a SPAN (Synchronized Position Attitude Navigation) system from NovAtel, model Pwrpak7D-E2, which consists of a MEMS-based tactical-grade IMU, Epson G370, and an OEM7 GNSS receiver. The visual platform is composed of a low-cost MEMS-based consumer-grade IMU and an industrial camera. The reference data system is from iMAR model iNAV-RQH-10018 which provides high-precision and reliable IMU measurements at data rates up to 300 Hz. The test system and the reference system are secured with aluminium extrusions and hardwood plywood boards to ensure the lever arm. Finally, the reference trajectories generated by commercial software (NovAtel Inertial Explorer®) that use double differential and smooth mode tightly coupled carrier phase measurements for INS/GNSS integration processing are regarded as true values.

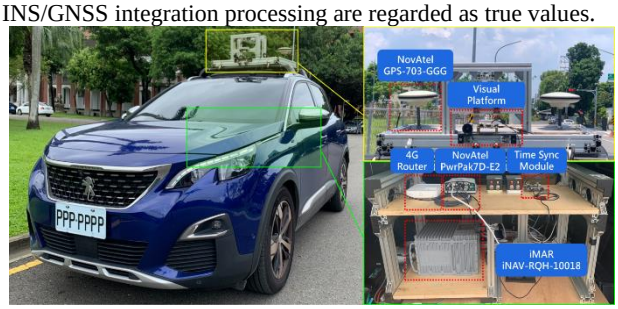

Figure 5 Experimental setup, including the mounting of the visual platform and reference system.

Physical Characteristics		iNAV-RQH-10018 (reference)	Epson G370 (testing)
Output Rate		$300Hz$	$100Hz$
Gyro	Bias Instability	$0.002\ °/hr$	$0.8\ °/hr$
	Random Walk Noise	$0.0015\ °/\sqrt{hr}$	$0.06\ °/\sqrt{hr}$
Accel	Bias Instability	$10\ \mu g$	$12\ \mu g$
	Random Walk Noise	$8\ \mu g/\sqrt{Hz}$	$15\ m/s/\sqrt{hr}$

Table 1 Specification of the RQH-10018 and EPSON G370

To ensure that the benchmarks for each test experiment are the same, an experimental platform was designed with support from Free CAD software for the Basler acA1300-75gc camera and OpenRTK330 IMU. The relative relationship between the original hole size and the design of the two sensors into this software for implementation. Considering the tolerance unit and size of the 3D printer, the results of the 3D printing design platform can be placed and fitted on the aluminium extrusion of the roof rack, as shown in Figure 6. In this way, it can be ensured that the relative relationship (distance and angle) of the camera and the IMU changes very little during installation.

Figure 6 3D Printed design platform and fixed sensors

Basler acA1300-75gc	
Resolution (H x V)	1280 x 1024 Pixels

Focal length	8 mm
FOV (mm)	Mono8, RGB8 BGR8, YCbCr (YUV)
Frame Rate	Up to 88 fps
Sensor Type	CMOS
Interface	Ethernet
Synchronization	Via hardware trigger Via software trigger
Power Requirements	Typical 3.3W when using Power over Ethernet
Conformity	IEEE 802.3af (PoE) IEEE 1588 (PTP-v2)

Table 2 Camera and lens specifications

	Bias Instability	Random Walk Noise	Range	Output
Gyroscope	2 °/hr	0.2 °/$\sqrt{\text{hr}}$	± 400 °/s	100 Hz
Accelerometer	20 μg	24 m/s/$\sqrt{\text{hr}}$	± 8 g	

Table 3 OPENRTK330LI EVB specifications

The first experiment sight was in Tainan city, Taiwan. The scenario contains a long outage in Hai-An underground parking lot. The performance of the proposed multi-sensor integration scheme in the Hai-an underground parking lot is shown in Figure 7, where the red line represents the reference, the green line represents the result of the proposed multi-sensor integration scheme without the VIO aiding, the blue line represents the proposed the multi-sensor integration scheme with the VIO aiding, and the orange line represents the two systems using the same GNSS results (RTK), and the purple box represents the height of the indoor.

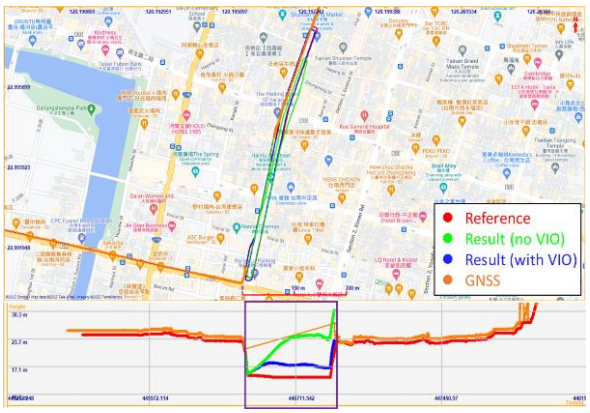

Figure 7 Test trajectory in the underground parking lot.

The proposed multi-sensor integration algorithm is statistically analysed and evaluated with and without the aiding of VIO. Table 4 and Table 5 are the PVA analysis of the proposed multi-sensor integration without and with VIO aiding, sequentially. For the position analysis, with the VIO aiding, the maximum error in the horizontal direction was reduced from more than 178 m to less than 18 m and the RMS error was reduced from 79 m to 10 m. The along-track and cross-track maximum errors were drastically reduced from 177 m to 17.8 m and 125 m to 10 m, respectively. In particular, the position error of the horizontal maximum per distance travelled was reduced from 9.7% to 0.1 %, and the position error of the horizontal maximum per time travelled was reduced from 0.394 m/s to 0.039 m/s as well. The along-track, cross-track errors and the comparison between the proposed multi-sensor integrate algorithm with or without the VIO aiding scheme in the Hai-an underground parking lot are shown in Figure 8 and Figure 9. In particular, the ability of the VIO-aiding algorithm to reduce errors is demonstrated in the horizontal position and velocity, along-track, and cross-track directions as well. VIO aiding for the multi-sensor integration scheme has better performance improvement in indoor environments.

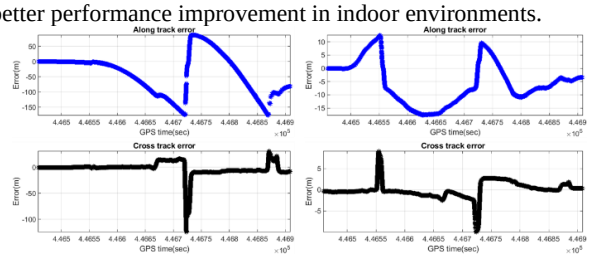

Figure 8 Along/cross track error of **without** (left) and **with** (right) the VIO aiding in the underground parking lot.

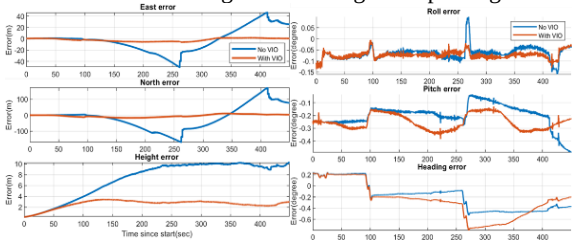

Figure 9 Error analysis plots in the **underground parking lot**.

Positional Error Analysis						Outage:	100 %
Unit: meter	E	N	U	H	3D	Along Track	Cross Track
Mean	-2.04	-14.54	7.238	59.346	60.224	-39.541	-2.415
Max	-50.587	171.389	10.261	177.519	177.765	176.955	124.41
RMSE	20.877	75.933	7.957	**78.75**	79.151	77.45	14.248

Velocity Error Analysis					
	E	N	U	H	3D
Mean	0.068	0.232	0.046	1.155	1.16
Max	0.727	2.986	0.104	3.05	3.05
RMSE	0.391	1.434	0.051	**1.486**	1.487

Attitude Error Analysis			
Unit: degree	Roll	Pitch	Heading
Mean	-0.066	-0.198	-0.19
Max	-0.161	-0.491	-0.531
RMSE	0.071	0.216	0.313

Distance travelled (DT):	1812 meters	Time traveled (TT):	450 s
HPE Max / DT:	**9.7 %**	HPE Max / TT:	0.394 m/s (0.766 nm/h)

Table 4 PVA analysis **without** VIO aiding in **underground parking lot**.

Positional Error Analysis						Outage:	100 %
Unit: meter	E	N	U	H	3D	Along Track	Cross Track
Mean	-1.908	-3.787	2.573	8.557	9.024	-5.582	-0.011
Max	-6.086	-17.071	3.425	17.819	18.109	17.775	9.959
RMSE	3.256	9.344	2.678	**9.895**	10.251	9.699	1.959

Velocity Error Analysis					
	E	N	U	H	3D
Mean	0.009	-0.024	0.014	0.259	0.261
Max	0.262	0.585	0.055	0.619	0.619
RMSE	0.111	0.278	0.024	**0.299**	0.3

Attitude Error Analysis			
Unit: degree	Roll	Pitch	Heading
Mean	-0.073	-0.253	-0.266
Max	-0.137	-0.386	-0.799
RMSE	0.075	0.259	0.401

Distance travelled (DT):	1812 meters	Time traveled (TT):	450 s
Horizontal Max DT:	**0.1 %**	Horizontal Max TT:	0.039 m/s (0.076nm/h)

Table 5 PVA analysis **with** VIO aiding in **underground parking lot**.

The second experiment sight was in Taipei City, Taiwan. The performance of the proposed multi-sensor integration scheme in the Taipei urban area at the experimental site is shown in Figure 10, where the red line represents the reference, the blue line represents the proposed multi-sensor integration scheme with the VIO aiding, and the orange line represents the two systems using the same GNSS results (RTK). It is worth noting that executing navigation under urban canyons and under the elevated road over-trusts poor GNSS results causing errors accumulation, as shown in Figure 10 (A) and (B), respectively. (A) is near the Taipei Railway Station, under the elevated road, and near there are more high buildings.

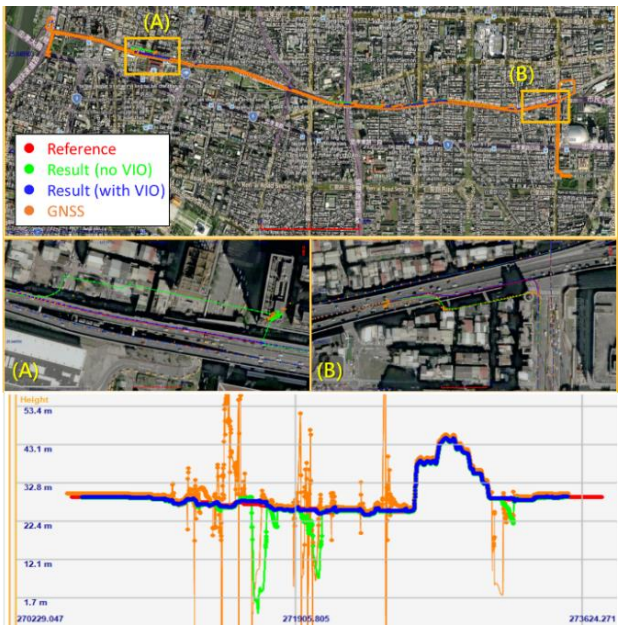

Figure 10 Test trajectory in **urban area**.

In the second scenario, the proposed algorithm is tested in the outdoor Taipei urban area. Table 6 and Table 7are the PVA analysis of the proposed multi-sensor integration without and with VIO aiding, sequentially. For the position analysis, with the VIO aiding, the maximum error in the horizontal direction was reduced from more than 38.4 m to less than 1.5 m. The along-track maximum errors were reduced from 16.7 to 1.4 m, and the cross-track maximum errors were reduced from 37.3 to 1.5 m. It is worth noting that the probability of the along-track improvement from 86.3 % to 100 %, and the probability of the cross-track improvement from 80.1 % to 100 % as well. As well as due to the GNSS and VIO-aiding conditions, the maximum error in the height is greatly reduced by 29.7 m to 1.1 m, refer to the elevation position error analysis plot, as shown in Figure 12.

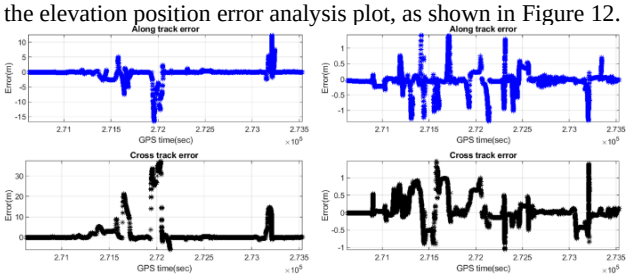

Figure 11 Along/cross track error **without** (left) and **with** (right) the VIO aiding in the Taipei urban area.

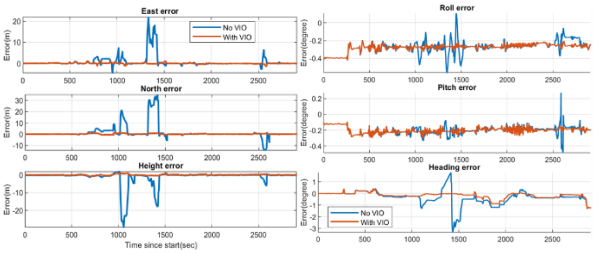

Figure 12 Error analysis plots of urban area

Positional Error Analysis						Epochs:	2478
Unit: meter	E	N	U	H	3D	Along Track	Cross Track
Mean	0.21	-0.08	0.166	0.261	0.36	0.074	-0.001
Max	1.624	-2.071	4.758	**4.289**	5.428	**1.656**	**2.063**
STD	0.155	0.21	0.53	0.225	0.546	0.215	0.259
RMSE	0.261	0.225	0.556	0.344	0.654	0.227	0.259

Velocity Error Analysis					
	E	N	U	H	3D
Mean	0.001	0.001	0.006	0.016	0.02
Max	-0.142	-0.52	-0.154	0.523	0.524
STD	0.015	0.023	0.013	0.023	0.024
RMSE	0.015	0.023	0.015	0.028	0.031

Attitude Error Analysis			
Unit: degree	Roll	Pitch	Heading
Mean	-0.098	-0.25	0.174
Max	-0.269	-0.383	1.047
STD	0.057	0.021	0.159
RMSE	0.113	0.251	0.236
Which Road	**100 %**	Which Lane	**99.2 %**

Table 6 PVA without the VIO aiding in the **urban area**.

Positional Error Analysis						Epochs:	2478
Unit: meter	E	N	U	H	3D	Along Track	Cross Track
Mean	0.197	-0.063	0.129	0.247	0.31	0.065	-0.02
Max	0.868	1.137	1.449	**1.356**	1.731	**0.87**	**1.138**
STD	0.102	0.146	0.194	0.118	0.181	0.168	0.204
RMSE	0.222	0.159	0.232	0.273	0.359	0.181	0.205

Velocity Error Analysis					
	E	N	U	H	3D
Mean	0.001	0.001	0.008	0.014	0.018
Max	-0.142	-0.107	0.157	0.17	0.19
STD	0.014	0.016	0.009	0.016	0.017
RMSE	0.014	0.016	0.012	0.022	0.025

Attitude Error Analysis			
Unit: degree	Roll	Pitch	Heading
Mean	-0.098	-0.251	0.163
Max	-0.275	-0.311	1.034
STD	0.059	0.023	0.137
RMSE	0.114	0.252	0.213
Which Road	**100 %**	Which Lane	**100 %**

Table 7 PVA analysis **with** the VIO aiding in the **urban area**.

5. CONCLUSION

This research proposes a navigation approach based on the visual integration structure using a camera and a low-cost IMU, which integrates INS, GNSS, and VIO respectively. For land vehicle positioning systems, to achieve the level of "which lane", it must be able to provide 1.5 meters of along/cross-track direction accuracy as a prerequisite. The characteristics of VIO composed of low-cost IMU are used to further solve the problems of drift and algorithm divergence of traditional VO in the travelled distance. The fundamental concept is to update existing velocity measurements based on INS/GNSS when VIO conditions are detected as non-stationary. And, built condition evaluation of GNSS measurements to verify the measurement results before the EKF update process. Protecting the navigation state for inaccurate measurement, whether GNSS or VIO. This research discusses the integration of GNSS RTK measurements with INS by using the same GNSS receiver, both with and without visual-inertial odometry aiding to validate the proposed algorithm, respectively. GNSS outage and GNSS-hostile scenario are discussed and selected separately. Show that the proposed strategy to integrate multi-sensor has great potential to achieve lane-level accuracy (1.5 m). It is worth mentioning that compared with the traditional INS/GNSS integration, the maximum error is greatly reduced, and the performance is stable and reliable. Especially, for GNSS solutions that suffer from multipath interference or NLOS contamination. The improvement of the VIO aiding scheme reveals the ability to reduce 96.1 % of horizontal direction maximum error and 94.9 % of horizontal direction RMS error in the cross-track direction. As well as, the ability to reduce 90.2 % and 96.8 % maximum error for the along-track and cross-track, respectively.

REFERENCES

1. Aqel, M. O. A., Marhaban, M. H., Saripan, M. I., & Ismail, N. B. (2016). Review of visual odometry: types, approaches, challenges, and applications. SpringerPlus, 5(1), 1–26. https://doi.org/10.1186/S40064-016-3573-7/TABLES/3
2. Bay, H., Tuytelaars, T., & Van Gool, L. (2006). SURF: Speeded up robust features. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 3951 LNCS, 404–417. https://doi.org/10.1007/11744023_32/COVER
3. Bloesch, M., Omari, S., Hutter, M., & Siegwart, R. (2015). Robust visual inertial odometry using a direct EKF-based approach. IEEE International Conference on Intelligent Robots and Systems, 2015-December, 298–304. https://doi.org/10.1109/IROS.2015.7353389
4. Calonder, M., Lepetit, V., Strecha, C., & Fua, P. (n.d.). BRIEF: Binary Robust Independent Elementary Features ⋆.
5. Campos, C., Elvira, R., Rodriguez, J. J. G., Montiel, J. M. M., & Tardos, J. D. (2021). ORB-SLAM3: An Accurate Open-Source Library for Visual, Visual-Inertial, and Multimap SLAM. IEEE Transactions on Robotics, 37(6), 1874–1890. https://doi.org/10.1109/TRO.2021.3075644
6. Davison, A. J., Reid, I. D., Molton, N. D., & Stasse, O. (2007). MonoSLAM: Real-time single camera SLAM. IEEE Transactions on Pattern Analysis and Machine Intelligence, 29(6), 1052–1067. https://doi.org/10.1109/TPAMI.2007.1049
7. Engel, J., Koltun, V., & Cremers, D. (2016). Direct Sparse Odometry. IEEE Transactions on Pattern Analysis and Machine Intelligence, 40(3), 611–625. https://doi.org/10.1109/TPAMI.2017.2658577
8. Engel, J., Schöps, T., & Cremers, D. (n.d.). LSD-SLAM: Large-Scale Direct Monocular SLAM.
9. Forster, C., Pizzoli, M., & Scaramuzza, D. (2014). SVO: Fast semi-direct monocular visual odometry. Proceedings - IEEE International Conference on Robotics and Automation, 15–22. https://doi.org/10.1109/ICRA.2014.6906584
10. Harris, C., & Stephens, M. (n.d.). A COMBINED CORNER AND EDGE DETECTOR. https://doi.org/10.5244/C.2.23
11. Klein, G., & Murray, D. (2007). Parallel tracking and mapping for small AR workspaces. 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality, ISMAR, 225–234. https://doi.org/10.1109/ISMAR.2007.4538852
12. Leutenegger, S., Furgale, P., Rabaud, V., Chli, M., Konolige, K., & Siegwart, R. (n.d.). Keyframe-Based Visual-Inertial SLAM Using Nonlinear Optimization.
13. Lowe, D. G. (2004). Accepted for publication in the. International Journal of Computer Vision.
14. Lucas, B. D., & Kanade, T. (n.d.). An Iterative Image Registration Technique with an Application to Stereo Vision.
15. Mourikis, A. I., & Roumeliotis, S. I. (2007). A multi-state constraint Kalman filter for vision-aided inertial navigation. Proceedings - IEEE International Conference on Robotics and Automation, 3565–3572. https://doi.org/10.1109/ROBOT.2007.364024
16. Mur-Artal, R., & Tardos, J. D. (2016). ORB-SLAM2: an Open-Source SLAM System for Monocular, Stereo and RGB-D Cameras. IEEE Transactions on Robotics, 33(5), 1255–1262. https://doi.org/10.1109/TRO.2017.2705103
17. Nistér, D., Naroditsky, O., & Bergen, J. (n.d.). Visual Odometry.
18. Qin, T., Li, P., & Shen, S. (n.d.). VINS-Mono: A Robust and Versatile Monocular Visual-Inertial State Estimator. Retrieved April 20, 2023, from https://github.com/HKUST-Aerial-Robotics/VINS-Mobile
19. Qin, T., & Shen, S. (2018). Online Temporal Calibration for Monocular Visual-Inertial Systems. IEEE International Conference on Intelligent Robots and Systems, 3662–3669. https://doi.org/10.1109/IROS.2018.8593603
20. Rosten, E., & Drummond, T. (2006). Machine learning for high-speed corner detection. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 3951 LNCS, 430–443. https://doi.org/10.1007/11744023_34/COVER
21. Rublee, E., Rabaud, V., Konolige, K., & Bradski, G. (2011). ORB: An efficient alternative to SIFT or SURF. Proceedings of the IEEE International Conference on Computer Vision, 2564–2571. https://doi.org/10.1109/ICCV.2011.6126544
22. Schönberger, J. L., & Frahm, J.-M. (n.d.). Structure-from-Motion Revisited. Retrieved April 20, 2023, from https://github.com/colmap/colmap.
23. von Stumberg, L., Usenko, V., & Cremers, D. (2018). Direct Sparse Visual-Inertial Odometry using Dynamic Marginalization. Proceedings - IEEE International Conference on Robotics and Automation, 2510–2517. https://doi.org/10.1109/ICRA.2018.8462905
24. Zhang, H., & Ye, C. (2020). Plane-Aided Visual-Inertial Odometry for 6-DOF Pose Estimation of a Robotic Navigation Aid. IEEE Access, 8, 90042–90051. https://doi.org/10.1109/ACCESS.2020.2994299