2015 International Conference on Indoor Positioning and Indoor Navigation (IPIN), 13-16 October 2015, Banff, Alberta, Canada

Sensor Fusion for Indoor Navigation and Tracking of Automated Guided Vehicles

Risang Gatot Yudanto, Frederik Petré
Flanders Make
Celestijnenlaan 300, B-3001 Leuven, Belgium
risanggatot.yudanto@flandersmake.be; frederik.petre@flandersmake.be

***Abstract*—A novel sensor fusion approach between a wireless Ultra-Wideband (UWB) Indoor Positioning System (IPS) and an Inertial Navigation System (INS) is presented for real-time indoor navigation and tracking of Automated Guided Vehicles (AGVs) and Mobile Robots in factories and warehouses. AGV applications are particularly challenging since they require very low position errors and very high position update rates to be able to track the highly dynamic movements in real-time. The sensor fusion algorithm consists of two main blocks: the first is a delay compensation block that compensates the delayed results of position estimation coming from the IPS, while the second block is a multi-rate Extended Kalman Filter (EKF) that combines dynamic models of the movements, delay-compensated position measurements from the IPS and measurements coming from the INS. To validate the proposed sensor fusion approach, extensive tests have been performed on a real-time platform using a combination of an Ubisense UWB IPS and an INS based on two low-cost MEMS accelerometers and one gyroscope mounted on the Flanders Make Badminton Robot performing highly dynamic movements. The results show that the proposed sensor fusion approach is able to deliver a 1 kHz position update rate in real-time operation with a position error standard deviation of 3.7 cm for linear movements and 1.7 degrees for rotational movements. These are significant performance improvements compared to a stand-alone Ubisense UWB indoor positioning system or a stand-alone inertial navigation system.**

Keywords*—Wireless indoor positioning; Indoor navigation and tracking; Sensor fusion; Ultra Wideband (UWB); Inertial Measurement Unit (IMU); Inertial Navigation System (INS); Extended Kalman Filter (EKF); Automated Guided Vehicle (AGV).

I. INTRODUCTION

Wireless indoor positioning, which allows to identify and locate process-critical assets in real-time, is gaining a lot of momentum and seems to be on the verge of a breakthrough, with many potential applications in e.g. factory automation, warehousing and logistics, healthcare, safety, security and smart buildings. Indoor navigation and tracking of Automated Guided Vehicles (AGVs) in factories and warehouses is a particularly challenging application, since these vehicles not only move at high speeds, requiring high position update rates (at least 100 Hz, preferably 1 kHz), but also regularly perform complex maneuvers (e.g. picking up a load), requiring low position errors (at most 10 cm, preferably 1 cm). Existing state-of-the-art radio-frequency (RF) indoor positioning technologies, like the Ubisense [1] Ultra Wide-Band (UWB) indoor positioning system, achieve 10 Hz position update rates and 30cm position errors for static objects under best case operating conditions, thus falling short in meeting the challenging requirements of AGV applications.

To adequately fulfill the requirements of AGV applications, in this paper we propose a novel Kalman Filter (KF) based sensor fusion method that optimally combines the position measurements from an Ubisense UWB indoor positioning system with the measurements from a low-cost Inertial Navigation System (INS) including MEMS accelerometers and gyroscopes. Compared to a stand-alone Ubisense indoor positioning system, the proposed sensor fusion approach allows to significantly increase the position update rate while at the same time reduce the position error. Compared to a stand-alone inertial navigation system, it guarantees long term stability of the position error by ridding integration drift. The sensor fusion method has been validated in real-time on the Flanders Make Badminton Robot performing highly dynamic linear and rotational movements. The Badminton Robot [2] which is able to play badminton against a human player with a real racket and a real shuttlecock, is an adequate proxy for an AGV exhibiting even more demanding requirements.

Some previous works also apply sensor fusion using different sensors or between an absolute and a relative positioning system for navigation and tracking of a mobile robot or vehicle. For example, [3] uses fusion of single-image localization and dead-reckoning data for continuous localization of a mobile robot, while [4] uses only medium-end Inertial Measurement Units (IMUs) and fuzzy-logic rules for mobile robot tracking. In [5], the fusion of an optical relative sensor and a wireless absolute localization system is proposed, while [6] proposes a modular framework for hybrid positioning using Bayesian Recursive Estimation algorithms also combined with available a priori knowledge such as map information. Recently, an integrated indoor navigation system for ground vehicles that fuses inertial sensors, light detection and ranging (LiDAR) sensors, received signal strength (RSS) observations in wireless local area networks (WLANs), odometry, and predefined occupancy floor maps has been proposed [7]. Finally, in [8] a sensor fusion between Time of Arrival based UWB positioning and IMUs leads to a good position and orientation accuracy.

This paper is organized as follows. Section II defines the system requirements and details the system architecture. Section III describes the proposed sensor fusion algorithm in general and the delay compensation block as well as the Extended Kalman Filter design in particular. Section IV discusses the experimental validation results with the Flanders

978-1-4673-8402-5/15/$31.00 @2015 IEEE

Make Badminton Robot. Finally, Section V summarizes our main conclusions and proposes some ideas for future work.

II. System Requirements and Architecture

A vehicle as illustrated in Figure 1 must perform auto navigation tasks and need to be tracked in real-time. The available sensor measurements in the vehicle are: two accelerometers on the left and right side of the vehicle, one gyroscope in the center of the vehicle and two UWB tags on the front and back of the vehicle. The main states to be estimated are position, (x, y), orientation (θ), linear velocities (v^t, v^n), angular velocity (ω), linear accelerations (a^t, a^n) and angular acceleration (a^ω).

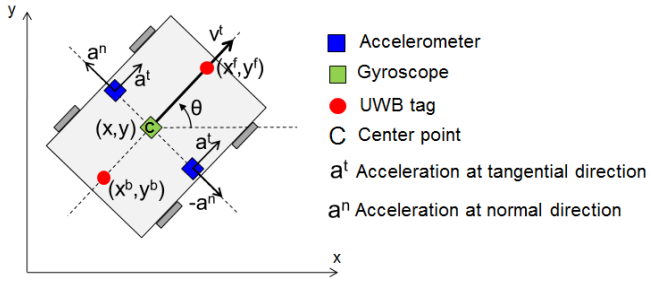

Figure 1: Illustration of sensor placement on an automatic guided vehicle.

A. System Requirements

Due to the complex and accurate maneuvers performed by the AGV during operation in factories and warehouses (e.g. picking up a load); and the dynamic nature of the AGV moving at high speeds, the system requirements for its localization and tracking system are very stringent, as summarized in Table 1.

	Required
2D Position error	1 – 10 cm
2D Orientation error	1 – 2 degrees
Update rate	100 Hz - 1 kHz
End-to-end delay	4 – 8 ms

Table 1: System Requirements for AGV localization and tracking inside factories and warehouses.

B. Experimental Results from Existing Systems

On the one hand, we have selected the commercially available Ubisense system as the IR-UWB technology platform to be used for indoor positioning of AGVs. This choice is based on the fact that Ubisense is market and technology leader for indoor positioning applications in factory automation and has been widely used in many applications in industry and research. The Ubisense system uses a combination of Time Difference of Arrival (TDoA) and Angle of Arrival (AoA) techniques to perform localization.

We have performed measurements with the Ubisense system using the Flanders Make Badminton Robot as an adequate proxy for an AGV. The experimental results, which have been presented in [9], are summarized in Table 2. It can be concluded that a stand-alone Ubisense system does not meet the requirements for wireless indoor navigation and tracking of AGVs; not in terms of position accuracy, nor position update rate and end-to-end delay.

	Measured
2D Position error	5 cm to > 50 cm (static) Up to 1.3 m (dynamic)
2D Orientation error	0.65 degrees to > 5 degrees (static) Up to 108 degrees (dynamic)
Update rate	10 Hz
End-to-end delay	184 ms

Table 2: Summary of measurement results using stand-alone Ubisense system.

On the other hand, it is also possible to use an Inertial Navigation System (INS), which is based on several Inertial Measurement Units (IMUs) such as accelerometers and gyroscopes, in order to localize and track the AGV. Unlike a wireless indoor positioning system, this approach does not require fixed infrastructure in terms of fixed base stations with a known reference position and is able to deliver the required high update rate. However the position measurements are drifting over time due to integration of noisy IMUs signals, especially when a low-cost Micro Electro-Mechanical Systems (MEMS) based INS is used. Experimental results from our study on using only low-cost IMU for online trajectory estimation is presented in [10].

A stand-alone wireless IPS using Ubisense system and a stand-alone INS each have their own advantages and disadvantages as summarized in Table 3.

Criteria	*Wireless IPS*	*Low-cost INS*
Short-term precision	Low	High
Long-term accuracy	High	Low
Update rate	Low	High
End-to-end delay	Large	Small

Table 3: Comparison between a stand-alone wireless Indoor Positioning System (IPS) and a stand-alone low-cost Inertial Navigation System (INS) for indoor navigation and tracking.

To benefit from the advantages of both systems, in this paper, we propose a novel sensor fusion method that optimally combines the position measurements from the wireless IPS with the measurements from the INS to obtain an improved estimation of position, velocity and acceleration. In this way, the performance of the overall localization system is dramatically improved, especially in terms of precision, update rate and end-to-end delay.

C. Sensor Fusion System Architecture

The system architecture for indoor navigation and tracking using sensor fusion between the Ubisense indoor positioning system and the inertial navigation system, comprising several IMUs is depicted in Figure 2. On the vehicle (Mobile Platform) it consists of two accelerometers, one gyroscope, and at least two Ubisense tags that transmit UWB signals. On the fixed world (Fixed Platform) four Ubisense base stations with a fixed and known reference position receive these signals and compute the tags' positions using TDoA and AoA algorithms.

The computed tags' positions are then sent to the fixed observer in the fixed platform that computes the vehicle position and orientation. Then it sends this information to the sensor fusion engine in the mobile platform through a custom low-latency wireless module that has been previously developed in Flanders Make [11]. In the meantime, the measurements from the accelerometers and gyroscope are sampled at a high rate (≥1 kHz) and the results are taken by the sensor fusion engine.

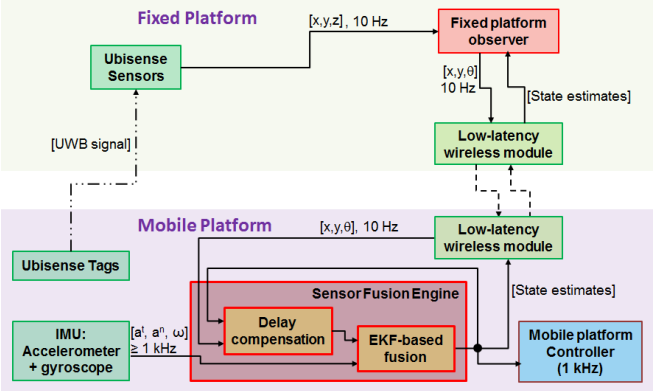

Figure 2: Functional block diagram of sensor fusion framework.

The outputs of sensor fusion engine are the state estimates including position, velocity and acceleration. The outputs need to be obtained at a high rate (at least 100 Hz; preferably 1 kHz) as they are used by the mobile platform controller, which is also running at the same high rate.

III. Sensor Fusion Design

From the AGV scheme illustrated in Figure 1, the position of the center and the orientation of the vehicle from measured positions of two UWB tags placed on the vehicle, one on the front (x^f, y^f) and one on the back (x^b, y^b), with both tags located at equidistance from the center can be calculated using equations expressed in (1), (2) and (3),:

$$x = \frac{x^f - x^b}{2} \tag{1}$$

$$y = \frac{y^f - y^b}{2} \tag{2}$$

$$\theta = \tan^{-1}\left(\frac{y^f - y^b}{x^f - x^b}\right) \tag{3}$$

with x and y represents measured vehicle's center position while θ is measured orientation.

On the other hand, it is also possible to estimate the states of the vehicle using measurement coming from IMUs. In generic form, the relation between position, orientation, velocity and acceleration can be expressed in the following equations:

$$x = x_0 + T \cdot v^x + \tfrac{1}{2} \cdot T^2 \cdot a^x \tag{4}$$

$$y = y_0 + T \cdot v^y + \tfrac{1}{2} \cdot T^2 \cdot a^y \tag{5}$$

$$\theta = \theta_0 + T \cdot \omega + \tfrac{1}{2} \cdot T^2 \cdot a^\omega \tag{6}$$

$$v^x = v_0^x + T \cdot a^x \tag{7}$$

$$v^y = v_0^y + T \cdot a^y \tag{8}$$

$$\omega = \omega_0 + T \cdot a^\omega \tag{9}$$

with T is integration time, x_0, y_0 and θ_0, are initial position and orientation, v_0^x and v_0^y are initial linear velocities in x and y directions, ω_0 is initial angular velocity, a^x and a^y are linear accelerations in x and y directions, while a^ω is angular acceleration.

The output of gyroscope is angular velocity (ω), while the output of the accelerometers are linear accelerations in tangential direction (a^t) and in normal direction (a^n). A conversion from tangential-normal (t-n) into x-y directions is needed and as illustrated in the AGV scheme in Figure 1, it can be expressed in (10) and (11):

$$a^x = a^t \cdot \cos\theta - a^n \cdot \sin\theta \tag{10}$$

$$a^y = a^t \cdot \sin\theta + a^n \cdot \cos\theta \tag{11}$$

The same concept can also be applied to define relationship between linear velocities in x-y with velocities in t-n directions.

$$v^x = v^t \cdot \cos\theta - v^n \cdot \sin\theta \tag{12}$$

$$v^y = v^t \cdot \sin\theta + v^n \cdot \cos\theta \tag{13}$$

With IMUs placement as depicted in Figure 1, where the gyroscope is in the center of the vehicle and two identical accelerometers placed at the right and left side of the vehicle at equidistance from the center, the linear accelerations a^t and a^n of the vehicle can be computed as the following:

$$a^t = \frac{a^{t\_R} + a^{t\_L}}{2} \tag{14}$$

$$a^n = \frac{a^{n\_R} + a^{n\_L}}{2} \tag{15}$$

with a^{t_R} and a^{t_L} are measured accelerations in tangential direction of right and left accelerometers while a^{n_R} and a^{n_L} are measured accelerations in normal direction of right and left accelerometers respectively.

As illustrated in the functional block diagram in Figure 2, the computation of vehicle's center position and orientation from two UWB tags positions expressed in equations (1), (2) and (3) is performed in the fixed platform observer. The results x, y and θ are then forwarded to the sensor fusion engine in the mobile platform.

Due to the significant delay of position measurements coming from IPS, a dedicated delay compensation is needed beside the Extended Kalman Filter itself. Therefore, the sensor fusion engine consists of two main blocks: a delay compensation block and a multi-rate Extended Kalman Filter (EKF) block.

A. *Delay Compensation*

Delay compensation block compensates the delay of the position estimates coming from the wireless indoor positioning system. The delay of the position coming from wireless IPS need to be compensated in order to achieve good accuracy in real-time when tracking high dynamic movements of the vehicle. From Table 2, it can be seen that the end-to-end delay of the Ubisense system amounts to 184 ms, which is quite large. This delay introduces an additional error in the positioning due to the dynamics.

Every time the position measurements from IPS is received in the sensor fusion engine at time *k*, they represent the position at *k-d* with *d = 184* ms. The delay compensation predicts the actual position at time *k* using position measurements coming from wireless IPS received at time *k* as initial position (which represents the position at time *k-d*) plus the integration of measurements from IMUs from time *k-d* until *k-1*. The previous state estimates from time *k-d* until *k-1* are buffered and used in the delay compensation to reconstruct the trajectory between time *k-d* until *k-1* in order to have accurate prediction at time *k*.

Functional block diagram to illustrate the process inside delay compensation block is depicted in Figure 3.

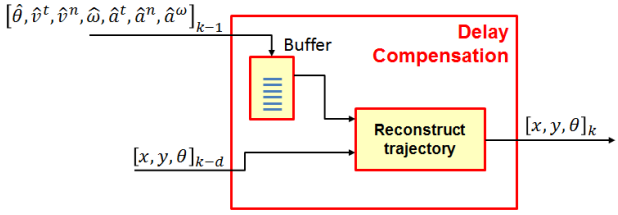

Figure 3: Functional block diagram of delay compensation block.

Mathematically, the computation inside the reconstruct trajectory block depicted in Figure 3 can be expressed in equation (16):

$$\begin{bmatrix} x_k \\ y_k \\ \theta_k \end{bmatrix} = \begin{bmatrix} x_{k-d} \\ y_{k-d} \\ \theta_{k-d} \end{bmatrix} + \sum_{n=(k-d)}^{n=k-1} \begin{bmatrix} T \cdot \hat{v}_n^x + \frac{1}{2} \cdot T^2 \cdot \hat{a}_n^x \\ T \cdot \hat{v}_n^y + \frac{1}{2} \cdot T^2 \cdot \hat{a}_n^y \\ T \cdot \hat{\omega}_n + \frac{1}{2} \cdot T^2 \cdot \hat{a}_n^\omega \end{bmatrix} \quad (16)$$

with T is the sampling time, while x_k, y_k, θ_k represents predicted position and orientation at time *k*, while x_{k-d}, y_{k-d}, θ_{k-d} are the delayed position and orientation received at mobile platform at time *k*. Velocity and acceleration in *x-y* directions $\hat{v}^x, \hat{v}^y, \hat{a}^x, \hat{a}^y$ can be represented in the form of *t-n* directions using the relations described in equations (10), (11), (12) and (13). Therefore, the required state estimates in delay compensation block are $\hat{\theta}$, $\hat{v}^t, \hat{v}^n$, $\hat{\omega}, \hat{a}^t, \hat{a}^n$, $\hat{a}^\omega$ which are taken from the output of EKF block.

Note that, it is also possible to directly use the raw angular velocity from gyroscope (ω) and raw linear accelerations from accelerometers (a^t, a^n) in the delay compensation block instead of using the state estimate output from EKF ($\hat{\omega}, \hat{a}^t, \hat{a}^n$). Nevertheless, the output of EKF is less noisy than the raw data because it is basically the filtered IMUs output, therefore the delay compensation block uses the state estimate output from EKF ($\hat{\omega}, \hat{a}^t, \hat{a}^n$) instead of the raw IMUs data.

B. *Extended Kalman Filter*

The second block is a multi-rate Extended Kalman Filter (EKF) that optimally combines the delay compensated position estimates from the indoor positioning system with measurements from the inertial navigation system as well as dynamic models of the movements. The use of a non-linear EKF has been chosen since the dynamic models of the movements are non-linear in nature.

A multi-rate EKF has been designed since the Ubisense tag localization update rate (~10 Hz) is much slower than the sampling rate of the inertial measurement units (≥ 1 kHz). The EKF uses one set of prediction models to predict all the state estimates $(\hat{x}, \hat{y}, \hat{\theta}, \hat{v}^t, \hat{v}^n, \hat{\omega}, \hat{a}^t, \hat{a}^n, \hat{a}^\omega)$ and two sets of measurement models. In case the Ubisense measurement is not available, the EKF uses the first set of measurement models that only incorporate IMUs measurements. This process runs at a high sampling rate. In case the Ubisense measurement is available, the EKF switches to the second set of measurement models that also incorporate the measured position from the Ubisense system next to the measurements from the IMUs. The latter process runs at a lower rate, i.e. the same rate as the Ubisense update rate. However, the state estimates are still available at the high rate since the same prediction models are always used in both cases.

The prediction model of the EKF is implemented in continuous time in Kalman-Bucy form [12] [13] while the model of measurement update is implemented in discrete time. A generic prediction model in time derivative form in continuous time based on system dynamic model can be expressed in (17):

$$\dot{\hat{\mathbf{x}}}(t) = \mathbf{F}(t) \cdot \hat{\mathbf{x}}(t) + \mathbf{B}(t) \cdot \mathbf{u}(t) + \mathbf{w}(t) \quad (17)$$

with $\hat{\mathbf{x}}(t)$ is a column vector of the predicted state estimates, $\mathbf{F}(t)$ is the dynamic coefficient matrix, $\mathbf{u}(t)$ is the input vector, which is zero in this case because the system is only an observer and there is no control input used in the state estimation, while $\mathbf{w}(t)$ is process noise, which is assumed to be Gaussian random variable with zero mean and covariance $\mathbf{Q}$, $\mathbf{w}(t) \sim N(0, \mathbf{Q})$.

The prediction model in our case can be expressed in (18):

$$\begin{bmatrix} \dot{\hat{x}}(t) \\ \dot{\hat{y}}(t) \\ \dot{\hat{\theta}}(t) \\ \dot{\hat{v}}^t(t) \\ \dot{\hat{v}}^n(t) \\ \dot{\hat{\omega}}(t) \\ \dot{\hat{a}}^t(t) \\ \dot{\hat{a}}^n(t) \\ \dot{\hat{a}}^\omega(t) \end{bmatrix} = \mathbf{F}(t) \cdot \begin{bmatrix} \hat{x}(t) \\ \hat{y}(t) \\ \hat{\theta}(t) \\ \hat{v}^t(t) \\ \hat{v}^n(t) \\ \hat{\omega}(t) \\ \hat{a}^t(t) \\ \hat{a}^n(t) \\ \hat{a}^\omega(t) \end{bmatrix} + \mathbf{w}(t) \tag{18}$$

with the dynamic coefficient matrix *F* is expressed in (19):

$$\mathbf{F}(t) = \begin{bmatrix} 0 & 0 & 0 & \cos\hat{\theta}(t) & -\sin\hat{\theta}(t) & 0 & dt\cdot\cos\hat{\theta}(t) & -dt\cdot\sin\hat{\theta}(t) & 0 \\ 0 & 0 & 0 & \sin\hat{\theta}(t) & \cos\hat{\theta}(t) & 0 & dt\cdot\sin\hat{\theta}(t) & dt\cdot\cos\hat{\theta}(t) & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{19}$$

On the other hand, the generic measurement model in discrete time at time *k* can be expressed in (20):

$$\mathbf{z}_k = \mathbf{H} \cdot \mathbf{x}_k + \mathbf{v}_k \tag{20}$$

with $\mathbf{z}_k$ is a column vector of state measurements, $\mathbf{H}$ is measurement sensitivity matrix and $\mathbf{v}_k$ is measurement noise, which is assumed to be Gaussian random variable with zero mean and covariance $\mathbf{R}$, $\mathbf{v}_k \sim N(0, \mathbf{R})$.

The first measurement models expressed in (21), (22) and (23) are used when only IMUs measurements (ω, a^t, a^n) are available. These models run at high sampling rate.

$$\begin{bmatrix} \omega \\ a^t \\ a^n \end{bmatrix} = \mathbf{H} \cdot \begin{bmatrix} x_k \\ y_k \\ \theta_k \\ v_k^t \\ v_k^n \\ \omega_k \\ a_k^t \\ a_k^n \\ a_k^\omega \end{bmatrix} + \mathbf{v}_k \tag{21}$$

$$\mathbf{H} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \tag{22}$$

The measurement noise **R** can be represented in (23):

$$\mathbf{R} = \begin{bmatrix} \sigma_\omega^2 & 0 & 0 \\ 0 & \sigma_{a^t}^2 & 0 \\ 0 & 0 & \sigma_{a^n}^2 \end{bmatrix} \tag{23}$$

with $\sigma_\omega^2, \sigma_{a^t}^2$ and $\sigma_{a^n}^2$ represents the noise variance of IMUs measurement output: ω, a^t and a^n. These values can be obtained from measured sensor signal.

The second measurement models expressed in (24), (25) and (26) are used when x, y and θ measurements from indoor positioning system are also available beside the IMUs output. These models run at low rate.

$$\begin{bmatrix} x \\ y \\ \theta \\ \omega \\ a^t \\ a^n \end{bmatrix} = \mathbf{H} \cdot \begin{bmatrix} x_k \\ y_k \\ \theta_k \\ v_k^t \\ v_k^n \\ \omega_k \\ a_k^t \\ a_k^n \\ a_k^\omega \end{bmatrix} + \mathbf{v}_k \tag{24}$$

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \tag{25}$$

The measurement noise **R** can be represented in (26):

$$\mathbf{R} = \begin{bmatrix} \sigma_x^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & \sigma_y^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma_\theta^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma_\omega^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sigma_{a^t}^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & \sigma_{a^n}^2 \end{bmatrix} \tag{26}$$

with σ_x^2, σ_y^2 and σ_θ^2 represents the noise variance of delay-compensated indoor positioning measurement output: x, y and θ.

During initialization, the value of **Q** can be initialized as expressed in (27). Since there are two sets of measurement models, the process noise **Q** also need to be defined for each of them. For the case when only IMUs measurements (ω, a^t, a^n) are available, Q_{66}, Q_{77} and Q_{88} can be initialized using small

values while the others need to be large enough (e.g. 10^{-2}). On the other hand, in the case when IPS measurements (x, y, θ) are also available besides the IMUs, Q_{11}, Q_{22} and Q_{33} can also be initialized with small values (e.g. 10^{-8}). Small values for $\mathbf{Q}$ represents small plant noise for the estimated state. This is usually the case when direct measurement of the estimated state is available to be observed, which means the prediction model has small error.

$$\mathbf{Q} = \begin{bmatrix} Q_{11} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & Q_{22} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & Q_{33} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & Q_{44} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & Q_{55} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & Q_{66} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & Q_{77} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & Q_{88} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & Q_{99} \end{bmatrix} \quad (27)$$

Prediction of the state estimates at time t_k can be computed by solving differential equation expressed in (18) for $\hat{\mathbf{x}}(t_k)$ from initial state estimate $\hat{\mathbf{x}}(t_{k-1})$. A fourth order Runge-Kutta method has been implemented to perform this computation.

The error covariance of the predicted state estimates can be expressed in (28):

$$\dot{\mathbf{P}}(t) = \mathbf{F}^{[1]}(t) \cdot \mathbf{P}(t) + \mathbf{P}(t) \cdot \mathbf{F}^{[1]^T}(t) + \mathbf{Q} \quad (28)$$

with $\mathbf{F}^{[1]}(t)$ is first order Jacobian matrix of the prediction model. The use of Jacobian matrix $\mathbf{F}^{[1]}(t)$ is necessary since $\dot{\hat{\mathbf{x}}}(t)$ is non-linear.

$$\mathbf{F}^{[1]}(t) = \frac{\partial \dot{\hat{\mathbf{x}}}(t)}{\partial \hat{\mathbf{x}}(t)} \quad (29)$$

Fourth order Runge-Kutta method is also used to compute the error covariance of the predicted state estimates at time t_k, $\mathbf{P}(t_k)$ from initial value $\mathbf{P}(t_{k-1})$.

State estimate measurement update can be computed using equation (30):

$$\hat{\mathbf{x}}_k(+) = \hat{\mathbf{x}}_k(-) + \overline{\mathbf{K}}_k \cdot \left[\mathbf{z}_k - \mathbf{H} \cdot \hat{\mathbf{x}}_k(-)\right] \quad (30)$$

with $\hat{\mathbf{x}}_k(+)$ is state estimate after measurement update while $\hat{\mathbf{x}}_k(-)$ is state estimate before measurement update, which is equals to the output of state estimate prediction $\hat{\mathbf{x}}(t_k)$.

Calculation of error covariance of the state measurement update is implemented using Joseph form [14] expressed in (31):

$$\mathbf{P}_k(+) = \left[\mathbf{I} - \overline{\mathbf{K}}_k \cdot \mathbf{H}\right] \cdot \mathbf{P}_k(-) \cdot \left[\mathbf{I} - \overline{\mathbf{K}}_k \cdot \mathbf{H}\right]^T + \overline{\mathbf{K}}_k \cdot \mathbf{R} \cdot \overline{\mathbf{K}}_k^T \quad (31)$$

with $\mathbf{I}$ is identity matrix, $\mathbf{P}_k(+)$ is error covariance after measurement update while $\mathbf{P}_k(-)$ is error covariance before measurement update, which is equals to the error covariance of prediction $\mathbf{P}(t_k)$. The Joseph form for computing $\mathbf{P}_k(+)$ in equation (31) has been chosen in order to overcome ill-conditioning problem.

$\overline{\mathbf{K}}_k$ in equations (30) and (31) is the Kalman gain matrix which can be computed using the following equation:

$$\overline{\mathbf{K}}_k = \mathbf{P}_k(-) \cdot \mathbf{H}^T \cdot \left[\mathbf{H} \cdot \mathbf{P}_k(-) \cdot \mathbf{H}^T + \mathbf{R}\right]^{-1} \quad (32)$$

In this work, the values of initial state estimates $\hat{\mathbf{x}}(t_0)$ are taken from the measurement when the mobile platform is in stand-still position while the error covariance of state estimates $\mathbf{P}(t_0)$ is initialized with a [9x9] diagonal matrix with diagonal elements equals to 10^{-2}.

IV. Experimental Validation

The sensor fusion algorithm has been implemented on Triphase Real Time Target (Intel Core Duo with 2.26 GHz clock speed) [15], and is able to run at 1 kHz rate. As a consequence, the achieved end-to-end delay is 1 ms. All the validation experiments have been performed using this platform.

Experimental validation of the proposed sensor fusion system has been performed using the Flanders Make Badminton Robot as the test platform. This robot has three degrees of freedom on three different axes: one linear axis, and two rotational axes mimicking the shoulder and elbow movement. Each axis is equipped with a highly accurate encoder that has been used as the reference measurement providing the ground truth for the tests.

The set-up consists of the electro-mechanical part of the Badminton Robot instrumented with two MEMS accelerometers, one MEMS gyroscope, and two Ubisense tags. This set-up accurately emulates the dynamic movements of automated guided vehicles and wheeled mobile robots. The low-cost MEMS inertial sensors used in this set-up are tri-axis accelerometers from Freescale (model MMA 7361) and a single axis gyroscope from Analog Devices (model ADXRS 620).

In order to better analyze the performance of the sensor fusion algorithm, the experiments are divided into two separate test cases: a first test case for an oscillating linear movement and a second test case for an oscillating rotary movement. Both cases use the highly accurate encoder signal from the Badminton Robot as the reference: the linear encoder on the linear motor for linear movement validation and the rotary encoder (shoulder movement) on the robot's arm for rotational movement validation.

A. *Validation for Linear Movement*

A picture of the set-up for linear movement validation is shown in Figure 4. The position measured by the linear encoder is used as the ground truth by first taking the average of the measured position from Ubisense before the robot starts to move as the initial position. The output of the sensor fusion is then compared with the measured position of the robot from the encoder.

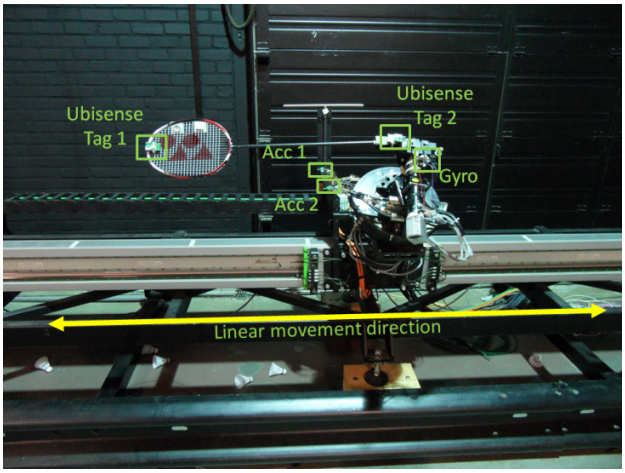

Figure 4: Test setup for linear movement validation of the sensor fusion using Flanders Make Badminton Robot.

In this test case, the robot moves with a maximum speed of 4.5 m/s and a maximum acceleration of 30 m/s^2. The estimated position and the position error for the linear movement is depicted in Figure 5. It can be seen how positioning using Ubisense-only is delayed while the position estimate from the sensor fusion approach compensates this delay and provides more accurate results. With respect to using only Ubisense, the sensor fusion approach reduces the position error standard deviation from 34 cm to 3.7 cm, which is almost a factor of 10 improvement. The maximum error, which is mainly caused by the end-to-end delay, is also reduced by a factor 6.5 from 1.3 m to 0.2 m.

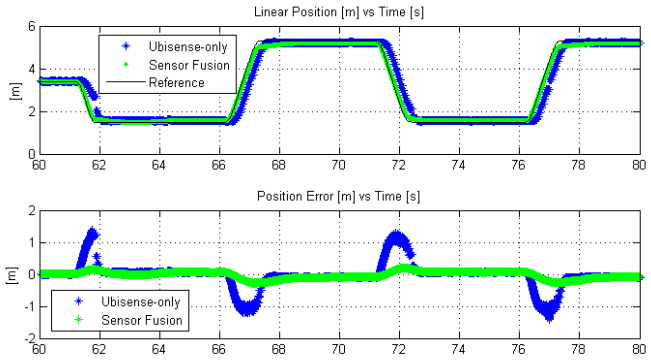

Figure 5: Estimated position (top) and position error (bottom) from linear movement validation.

B. *Validation for Rotational Movement*

A picture of this set-up is shown in Figure 6. The orientation measured by rotary encoder on the robot's shoulder is used as the ground truth by first taking the average of the measured orientation from Ubisense as the initial orientation before the robot's arm starts to move. The output of the sensor fusion is then compared with the measured orientation of the robot from the encoder.

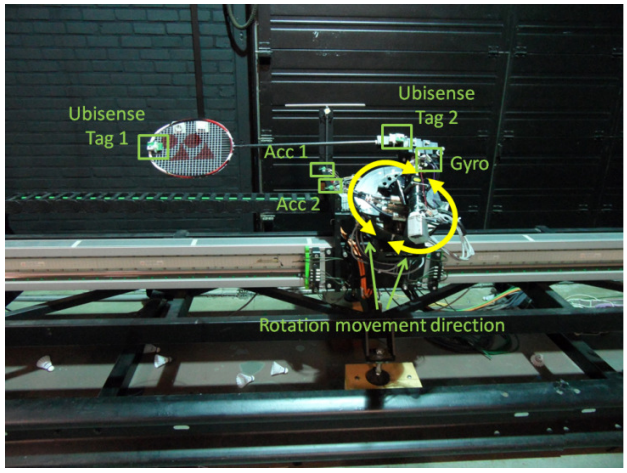

Figure 6: Test setup for rotational movement validation of the sensor fusion using Flanders Make Badminton Robot.

In this test case, the robot's arm moves with a maximum angular velocity of 300˚/s and a maximum angular acceleration of 1000˚/s^2. The result of the estimated orientation from the validation test is depicted in Figure 7. Here, it can be seen also how sensor fusion compensates the delay from Ubisense-only positioning and provides more accurate results. With respect to using only Ubisense, the sensor fusion approach reduces the orientation error standard deviation from 18˚ to 1.7˚, which is around a factor of 10 improvement. The maximum error is also reduced by a factor 13.5 from 108˚ to 8˚.

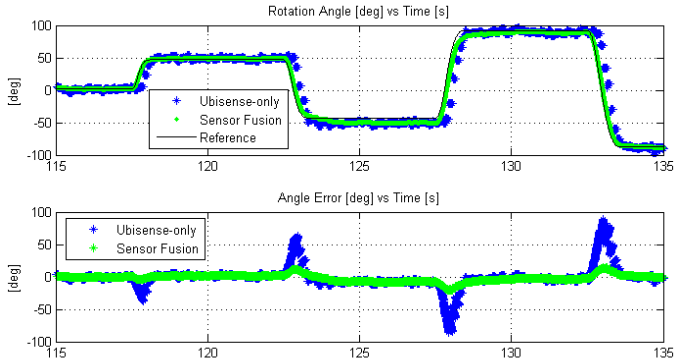

Figure 7: Estimated orientation angle (top) and orientation error (bottom) from rotational movement validation.

V. CONCLUSION

Using sensor fusion between the Ubisense indoor positioning system and an inertial navigation system clearly improves the performance of stand-alone Ubisense-only or INS-only positioning. The results are summarized in Table 4.

Criteria	*Ubisense-only*	*Sensor Fusion*
Update rate	10 Hz	1 kHz
End-to-end delay	184 ms	1 ms
Position error (1σ)	34 cm	3.7 cm
Max position error	1.3 m	0.2 m
Orientation error (1σ)	18˚	1.7˚
Max orientation error	108˚	8˚

Table 4: Summary of sensor fusion results from experimental validation.

The proposed sensor fusion approach gives a factor 10 improvement in both linear and rotation movement precision compared with Ubisense-only positioning and without position drift as in the case of low-cost INS-only positioning.

Nevertheless, there is still room for improvement in the proposed sensor fusion, which is part of our future works. For example, each position measurement coming from each UWB tag can be included in the measurement model and apply sequential update each time the measurement from each tag is available. The measurement from each accelerometer on the right and left side can also be included separately in the measurement model in order to have better estimate of angular acceleration. Due to the drift, IMUs bias and sensitivity can also be included in the state estimates so that they can be compensated when they drift.

REFERENCES

[1] www.ubisense.net

[2] J. Stoev, S. Gillijns, A. Bartic, W. Symens, "Badminton Playing Robot – A Multi-Disciplinary Test Case in Mechatronics", 5th IFAC Symposium on Mechatronic Systems, Cambridge, Massachusetts, USA, September 13-15, 2010.

[3] J. Hom, G. Schmidt, "Continuous Localization for Long-Range Indoor Navigation of Mobile Robots", IEEE International Conference on Robotics and Automation, 1995

[4] L. Ojeda, J. Borenstein, "FLEXnav: Fuzzy Logic Expert Rule-Based Position Estimation for Mobile Robots on Rugged Terrain", IEEE International Conference on Robotics and Automation, 2002

[5] L. Klingbeil, M. Romanovas, P. Schneider, M. Traechtler, Y. Manoli, "A Modular and Mobile System for Indoor Localization", International Conference on Indoor Positioning and Indoor Navigation IPIN, 2010

[6] J. C. F Michel, M. Christmann, M. Fiegert, P. Gulden, M. Vossiek, "Multisensor Based Indoor Vehicle Localization System for Production and Logistic", IEEE International Conference on Multisensor Fusion and Integration, Heidelberg, Germany, September 2006

[7] M. M. Atia, S. Liu, H. Nematallah, T. B. Karamat, A. Noureldin, "Integrated Indoor Navigation System for Ground Vehicles With Automatic 3-D Alignment and Position Initialization", IEEE Transaction on Vehicular Technology, vol 64, no 4, April 2015.

[8] M. Kok, J. D. Hol, T. B. Schön, "Indoor Positioning Using Ultrawideband and Inertial Measurements", IEEE Transaction on Vehicular Technology, vol 64, no 4, April 2015.

[9] R. Yudanto, M. L. Ruiz de Arbulo, F. Petré, "Multi-tag IR-UWB Wireless Positioning of a Badminton Robot", 14th Mechatronics Forum International Conference, Karlstad, Sweden, June 16-18, 2014

[10] R. Yudanto, A. P. Ompusunggu, A. Bey-Temsamani, "On Improving Low-Cost IMU Performance for Online Trajectory Estimation", SPIE International Conference on Cyber Physical Systems, Barcelona, Spain, May 4-6, 2015

[11] M. Paselli, F. Petré, O. Rousseaux, G. Meynants, B. Gyselinckx, M. Engels, "A high-performance wireless sensor node for industrial control applications", International Conference on Systems, 2008.

[12] R. E. Kalman, "A new approach to linear filtering and prediction problems", ASME Journal of Basic Engineering, Vol. 82, pp. 34-45, 1960.

[13] R. E. Kalman, R. S. Bucy, "New results in linear linear filtering and prediction theory", ASME Journal of Basic Engineering, Series D, Vol. 83, pp. 95-108, 1961.

[14] R. S. Bucy, P. D. Joseph, *Filtering for Stochastic Processes, with Applications to Guidance*, Wiley, New York, 1968.

[15] http://www.triphase.be/products/RTT/