Multiple-Sensors-Fusion for Localization of a Container-Truck using a Fast Particle Filter

Rini N. Fatimah, Adi Suparyanto, Yul Yunazwin Nazaruddin, and Augie Widyotriatmo

***Abstract*—This paper presents a particle filter approach to localize a container truck system using indoor global positioning system (iGPS), radio frequency identification (RFID) and inertial measurement unit (IMU) that consists of accelerometer, gyroscope, and magnetometer. A fast particle filter conducted by the properties of Gaussian probability density function is proposed to perform fast and low-cost computation. The prediction, particle weighting, and estimation stages are used to estimate the angle and the position of the container-truck. The effectiveness and the applicability of the proposed fusion algorithm for localization of a container-truck using the multiple sensors of iGPS, RFID, and IMU are shown by experimental results.**

***Index Terms*—Particle filter, localization, sensor fusion, container truck**

I. Introduction

AUTOMATED container-trucks is an important development in logistic industry. They offer unmanned operation, so the productivity and efficiency are increased and, moreover, human error is reduced. To build the automated container terminal, there are three main components: quay gantry crane, automated guided vehicles (AGVs), and automated stacking crane or automated rubber-tired-gantry [1]. This paper focuses on AGVs, especially the container-trucks. AGVs are used to transport containers that are moved from the loading/unloading area to the temporary storage area. Localization systems which determine the position of AGVs are required. The control system of AGVs needs position data to determine direction and movement ensuring that the AGVs can follow their desired trajectories or paths.

In mobile robot localization systems, relative position sensors are widely utilized. The types of sensors include encoder and inertial sensor. These sensors measure truck's linear and angular velocity or acceleration. By integrating the sensor measurements, the position and orientation are estimated [2]. The relative position sensors have a high update rate or, in other words, by utilizing the relative position sensors, the sampling frequency of the position data is high. However, integration of relative sensors in obtaining position data is not suitable for long-term measurement because it suffers from huge drifting bias due to accumulated error during the integration. Therefore, it needs external references for correction [3], [4]. In this case, the absolute position sensors. The absolute position sensors can be provided by camera, GPS, iGPS, beacon, landmark, and model mathcing between the measurement of wall edges with a known map. The method using absolute position sensors provides accurate long-term measurement but takes longer update rate than that using the relative position sensors.

Sensor fusion is a technique to merge both types of positioning sensors in order to provide high accuracy estimation and high update rate. Sensor fusion techniques for mobile robot localization have been developed using different kinds of sensors. In [5], author used encoders, compass sensor, laser range finder, and omni-directional camera to obtain mobile robots state based on an Extended Kalman Filter. In [3], the extended Kalman filter (EKF) is used to localize a mobile robot equipped with an encoder, compass, IMU, and GPS. In [6] and [7], an IMU, wheel encoder, and GPS are integrated using Kalman Filter to obtain three-dimensional (3D) outdoor localization system. In [8], low pass filter and Kalman filter were used to reduce noise of input signal and combine sensor data from odometer, IMU, active beacon to obtain precise navigation system. IMU and GPS were also fused in [9] using intelligent fusion based on adaptive neuro-fuzzy inference system (ANFIS). In [10], particle filter algorithm to improve performance over the established nonlinear filtering approach (e.g. EKF) is proposed. Kalman filter has been known as the fast recursive linear computational estimation. It employs the Gaussian distribution in describing the uncertainties of sensors. The EKF method extends the estimation method by the capabilities to handle with the nonlinearities of the model. Particle filter employs Bayesian filter in projecting and updating the sensor data. The eminence of this particle filter is that it can cope with the nonlinearity of the model as well as to handle any probability distribution of the uncertainties. The ultimate deficiency of the particle filter is the high cost computation, where a lot of

Manuscript received June 19, 2018. This work was supported by the Ministry of Research, Technology and Higher Education of Republic of Indonesia under the Decentralized Research Program on Excellent Research University, Bandung Institute of Technology, Bandung, Indonesia, 2018

R.N. Fatimah is with the Engineering Physics Study Program, Faculty of Industrial Technology, Bandung Institute of Technology, Bandung, Indonesia (e-mail: rini.rnf@gmail.com)

A. Suparyanto, is with the Engineering Physics Study Program, Faculty of Industrial Technology, Bandung Institute of Technology, Bandung, Indonesia (e-mail: adisuparyanto68@gmail.com)

Y.Y. Nazaruddin is with Instrumentation and Control Research Group, Faculty of Industrial Technology, Bandung Institute of Technology, Bandung, Indonesia (e-mail: yul@tf.itb.ac.id)

A. Widyotriatmo is is with Instrumentation and Control Research Group, Faculty of Industrial Technology, Bandung Institute of Technology, Bandung, Indonesia (e-mail: augie@tf.itb.ac.id)

particles and their distribution functions need to be employed in the calculation

In this paper, we mainly discuss the implementation of particle filter to estimate container truck's position. We propose a particle filter algorithm with Gaussian distribution. The probability density function of the sensor measurement errors and noises are assumed to be Gaussian. This method provides a fast and low cost computation particle filter. A research in [11] states that, under the Gaussian assumption, Gaussian particle filter is asymtotically optimal in the number of particle, hence, has much improved performance and versatility over filter using other distributions. Besides, Gaussian assumption makes the lower complexity than known particle filter.

The sensor fusion technique utilizes accelerometer, gyroscope, magnetometer, iGPS, and radio frequency identification (RFID). Both of relative and absolute position sensor may provide wrong measurement data because of environmental changes. RFID is used because this sensor had resilienceness to the environmental changes [12]. Particle filter is used to estimate the orientation angle and the position of the truck. Gyroscope and magnetometer are fused to estimate the orientation angle of the truck. There are two conditions to estimate the position. First, the condition where RFID is not detected. Then, the position is estimated by fusing accelerometer and iGPS measurement data. Second condition is where the RFID is detected. Then, the position is estimated by fusing the RFID and iGPS measurement data.

This paper is organized as follows: Section II derives the kinematic model of container truck and the Gaussian probability density function multiplication; Section III presents the particle filter algorithm; Section IV shows the experimental result followed by conclusions in Section V.

II. METHODS

A. Kinematic Model

The measurement data of IMU (consisted of accelerometer, gyroscope, and magnetometer), iGPS, and RFID are fused using the particle filter algorithm. Various serial communication types, i.e., universal asynchronous receiver/transmitter (UART), serial peripheral interface (SPI), and inter-integrated circuit (I2C), from sensors to a node microcontroller in the container-truck is used. Then, the data from the node microcontroller is parsed to the node microcontroller in the control room (CR) via wireless communication, using MQTT protocol. The node in the CR is connected to a personal computer (PC) that provides user interface. Fig.1 shows the system design of multiple sensor fusion in this research. The aim of the algorithm is to estimate the container truck's position and orientation angle. System design of multiple sensor fusion involves several components such as sensor, microcontroller, communication protocol and also processing software.

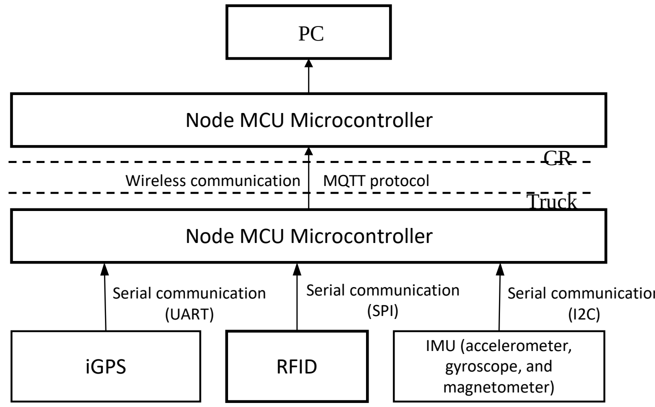

Fig. 1. System Design of Multiple Sensor Fusion

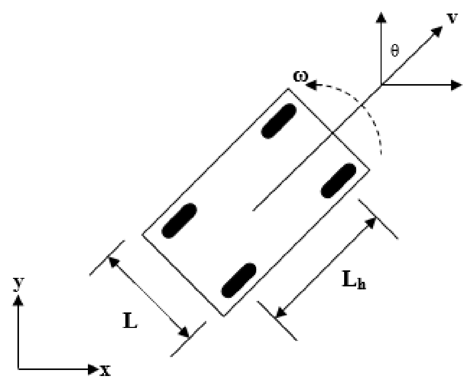

Fig. 2. Kinematic Model of Container truck

To implement the sensor fusion technique, we start with the kinematic model of the container truck rigid body. Fig.2 shows container truck rigid body model schematic, where x, y are the coordinate system denoting the position of the truck in Cartesian coordinate, θ is the orientation, R denotes the radius of driven wheels, L denotes the distance between the wheels, and v and ω are the linear and rotational velocities of the container truck rigid body.

The kinematic model of container truck rigid body in Fig.2 follows the nonholonomic systems which contains the derivative with respect to time of x, y, and θ as

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \sin\theta & 0 \\ \cos\theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{v} \\ \boldsymbol{\omega} \end{bmatrix}. \tag{1}$$

B. Gaussian Probability Density Function

Gaussian distribution with mean μ dan variance σ^2 is utilized as the uncertainties model of the sensors. Let p be the random variables of measurement. The probability density function $f(p)$ of the Gaussian distribution fulfills the equation (2).

$$f(p) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left[-\frac{(p-\mu)^2}{2\sigma^2}\right],\ -\infty < p < \infty \tag{2}$$

Let $f(p)$ and $g(p)$ be the Gaussian probability density function with arbitrary means μ_f and μ_g and standard deviations σ_f and σ_g,which are

$$f(p) = \frac{1}{\sqrt{2\pi}\sigma_f} e^{-\frac{(p-\mu_f)^2}{2\sigma_f^2}} \text{ and } g(p) = \frac{1}{\sqrt{2\pi}\sigma_g} e^{-\frac{(p-\mu_g)^2}{2\sigma_g^2}}.$$

The multiplication between *f(p)* and *g(p)* is illustrated in Fig.3. Their multiplication is

$$f(p)\,g(p) = \frac{1}{2\pi\sigma_f\sigma_g} e^{-\left(\frac{(p-\mu_f)^2}{2\sigma_f^2}+\frac{(p-\mu_g)^2}{2\sigma_g^2}\right)}. \tag{3}$$

Examine the term in the exponent, we have

$$\gamma = \frac{(p-\mu_f)^2}{2\sigma_f^2} + \frac{(p-\mu_g)^2}{2\sigma_g^2} \tag{4}$$

Expanding the two quadratics and collecting terms in (4) in the powers of p gives

$$\gamma = \frac{(\sigma_f^2+\sigma_g^2)p^2 - 2(\mu_f\sigma_g^2+\mu_g\sigma_f^2)p + \mu_f^2\sigma_g^2 + \mu_g^2\sigma_f^2}{2\sigma_f^2\sigma_g^2}. \tag{5}$$

Then, restructuring (5) yields

$$\gamma = \frac{p^2 - 2\frac{(\mu_f\sigma_g^2+\mu_g\sigma_f^2)}{(\sigma_f^2+\sigma_g^2)}p + \frac{\mu_f^2\sigma_g^2+\mu_g^2\sigma_f^2}{(\sigma_f^2+\sigma_g^2)}}{2\frac{\sigma_f^2\sigma_g^2}{(\sigma_f^2+\sigma_g^2)}}. \tag{6}$$

Defining new variable ϵ equation (7)

$$\epsilon = \frac{\left(\frac{\mu_f\sigma_g^2+\mu_g\sigma_f^2}{\sigma_f^2+\sigma_g^2}\right)^2 - \left(\frac{\mu_f\sigma_g^2+\mu_g\sigma_f^2}{\sigma_f^2+\sigma_g^2}\right)^2}{2\frac{\sigma_f^2\sigma_g^2}{(\sigma_f^2+\sigma_g^2)}} = 0, \tag{7}$$

Then,

$$\gamma = \frac{p^2 - 2\frac{(\mu_f\sigma_g^2+\mu_g\sigma_f^2)}{(\sigma_f^2+\sigma_g^2)}p + \left(\frac{\mu_f\sigma_g^2+\mu_g\sigma_f^2}{\sigma_f^2+\sigma_g^2}\right)^2}{2\frac{\sigma_f^2\sigma_g^2}{(\sigma_f^2+\sigma_g^2)}} + \frac{\frac{\mu_f^2\sigma_g^2+\mu_g^2\sigma_f^2}{(\sigma_f^2+\sigma_g^2)} - \left(\frac{\mu_f\sigma_g^2+\mu_g\sigma_f^2}{\sigma_f^2+\sigma_g^2}\right)^2}{2\frac{\sigma_f^2\sigma_g^2}{(\sigma_f^2+\sigma_g^2)}} \tag{8}$$

We define the combined standard deviation σ_{fg} and the combined mean μ_{fg} from the multiplication of the probability functions of f and g, where

$$\sigma_{fg} = \sqrt{\frac{\sigma_f^2\sigma_g^2}{(\sigma_f^2+\sigma_g^2)}} \quad\text{and}\quad \mu_{fg} = \frac{(\mu_f\sigma_g^2+\mu_g\sigma_f^2)}{(\sigma_f^2+\sigma_g^2)}. \tag{9}$$

After some manipulations, (8) reduces to

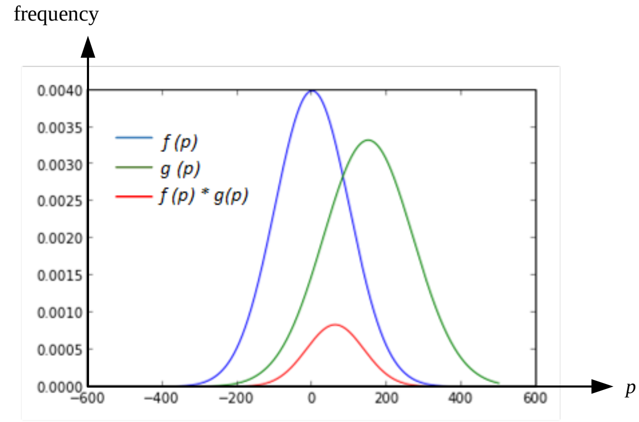

Fig. 3. Two Gaussian Probability Density Function Multiplication

$$\gamma = \frac{\left(p - \frac{\mu_f\sigma_g^2+\mu_g\sigma_f^2}{\sigma_f^2+\sigma_g^2}\right)^2}{2\frac{\sigma_f^2\sigma_g^2}{(\sigma_f^2+\sigma_g^2)}} + \frac{(\mu_f-\mu_g)^2}{2(\sigma_f^2+\sigma_g^2)}$$
$$= \frac{(p-\mu_{fg})^2}{2\sigma_{fg}^2} + \frac{(\mu_f-\mu_g)^2}{2(\sigma_f^2+\sigma_g^2)}. \tag{10}$$

Substituting (10) into (3) gives

$$f(p)g(p) = \frac{1}{2\pi\sigma_f\sigma_g} exp\left[-\frac{(p-\mu_{fg})^2}{2\sigma_{fg}^2}\right] exp\left[-\frac{(\mu_f-\mu_g)^2}{2(\sigma_f^2+\sigma_g^2)}\right] \tag{11}$$

Defining $\delta = \frac{(\mu_f-\mu_g)^2}{2(\sigma_f^2+\sigma_g^2)}$, we have

$$f(p)g(p) = \frac{1}{2\pi\sigma_f\sigma_g} exp\left[-\frac{(p-\mu_{fg})^2}{2\sigma_{fg}^2}\right] e^{-\delta} \tag{12}$$

Normalization is needed to cancel δ. If $f(p)g(p)$ is written by $F(p)$ and its normalization is written by $\widehat{F(p)}$, where

$$\widehat{F(p)} = \frac{F(p)}{\int_{-\infty}^{\infty} F(p)dp} \tag{13}$$

$$\int_{-\infty}^{\infty} F(p)dp = \int_{-\infty}^{\infty} \frac{1}{2\pi\sigma_f\sigma_g} exp\left[-\frac{(p-\mu_{fg})^2}{2\sigma_{fg}^2}\right] \exp(-\delta)\ dp \tag{14}$$

$$\int_{-\infty}^{\infty} F(p)dp = \frac{e^{-\delta}}{2\pi\sigma_f\sigma_g} \int_{-\infty}^{\infty} \frac{1}{2\pi\sigma_f\sigma_g} e^{-\frac{(p-\mu_{fg})^2}{2\sigma_{fg}^2}}\ dp \tag{15}$$

$$\int_{-\infty}^{\infty} F(p)dp = \frac{e^{-\delta}}{\sqrt{2}\pi\sigma_f\sigma_g}\sigma_{fg} \tag{16}$$

we obtain

$$\widehat{F(p)} = \frac{1}{\sigma_{fg}\sqrt{2\pi}} e^{\frac{-(p-\mu_{fg})^2}{2\sigma_{fg}}} \quad (17)$$

Equation (17) shows the result of normalized multiplication of two Gaussian probability density functions, *f*(*p*) and *g*(*p*). It can be seen that the product yields the combined mean μ_{fg} and standard deviation σ_{fg}. The new combined values are used in the calculation of the particle filter algorithm.

III. PARTICLE FILTER ALGORITHM

The particle filter algorithm is used to estimate position and orientation angle of container truck from the fusion of the iGPS, IMU, and RFID. The fusion process consists of the angle estimation and position estimation. The algorithms and procedures in determining the angle and position estimation, respectively, are shown.

A. *Orientation Angle Estimation*

For the orientation angle estimation using particle filter, it needs three stages: prediction, particle weighting, and estimation. Fig.4 shows the flow chart in determining the angle estimation. The orientation angle estimation fuses the data from magnetometer and the gyroscope.

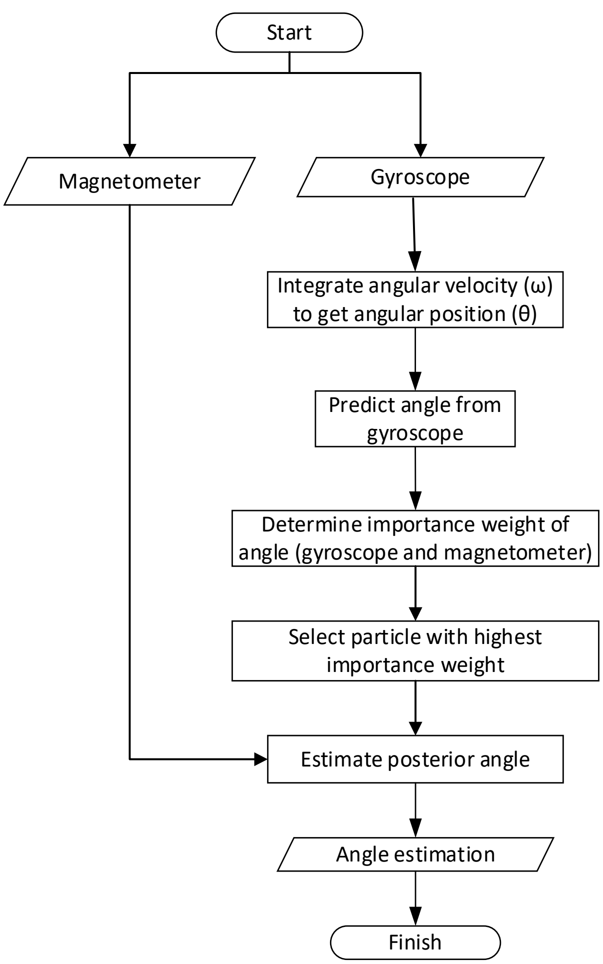

Fig. 4. Angle Estimation Flow Chart

Prediction Stage

The prediction stage is used to predict the orientation angle of container truck. At the prediction stage, a set number of particle is generated around the gyroscope measurement value. The output of gyroscope is the angular velocity. The prediction step for gyroscope fulfills the following equations:

$$\theta_{t+1} = \theta_t + \dot{\theta}_t \, dt + \theta_e, \quad (18)$$

$$\theta_e = \sigma_\theta \text{ rand } (n), \quad (19)$$

where θ_e represents the gyroscope measurement error which is assumed to be Gaussian with zero mean, the function rand*(n)* in (19) represents the function that generates random number, and σ_θ^2 is the gyroscope variance.

Particle Weighting Stage

The particle weighting stage is used to determine the weight of each particle. The weight represents particle's probability to estimate posterior angle. If the sensor measurement error is assumed to be Gaussian, then the function for determining the weight of each angle particle is

$$W_{t+1}^i\left(\theta_{t+1}^i, \bar{\theta}_{t+1}\right) = \frac{1}{\sigma_\theta\sqrt{2\pi}} \exp\left(\frac{(\theta_{t+1}^i - \bar{\theta}_{t+1})^2}{2(\sigma_\theta)^2}\right), \quad (20)$$

where θ_{t+1}^i is the predicted angle for the *i-th* particle at time *t* + 1, σ_θ^2 is the gyroscope variance, and $\bar{\theta}_{t+1}$ is the magnetometer sensor measurement sampled at time *t* + 1. After that, we find the biggest weight $W_{t+1}^{max,\theta}$ of angle particle which is represented by (21).

$$\hat{\theta} = \theta_{t+1}(W_{t+1}^{max,\theta}). \quad (21)$$

Estimation Stage

After we get particle with the biggest weight, we perform angle estimation by multiplying the Gaussian probability density function as the result of the prediction stage and the Gaussian probability function of magnetometer sensor measurement. If the probability density function as the result of prediction stage is represented by $f(\theta)$, we have

$$f(\theta) = \frac{1}{\sigma_\theta\sqrt{2\pi}} \exp\left(-\frac{(\theta-\mu_\theta)^2}{2\sigma_\theta^2}\right), \quad (22)$$

where σ_θ^2 represents the gyroscope variance and μ_θ represents the value which is resulted on the prediction stage. The Gaussian probability density function of magnetometer measurement is represented by $f(\theta|z)$

$$f(\theta|z) = \frac{1}{\sigma_z\sqrt{2\pi}} \exp\left(-\frac{(\theta-\mu_z)^2}{2\sigma_z^2}\right), \quad (23)$$

where σ_z^2 is magnetometer variance and μ_z is angle value of magnetometer sensor measurement. The two Gaussian probability function is multiplied, then the estimated probability density function of the orientation angle $f_{est}(\theta)$ is

$$f_{est}(\theta) = \frac{1}{\sigma_z \sigma_\theta \sqrt{2\pi}} \exp\left(-\frac{(\theta-\mu_z)^2}{2\sigma_z^2} - \frac{(\theta-\mu_q)^2}{2\sigma_\theta^2}\right). \quad (24)$$

Based on (17), the normalized of (24) is

$$\widehat{f_{est}}(\theta) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-(\theta-\mu_\theta)^2}{2\sigma}} \quad (25)$$

Therefore the estimated angle value is

$$\theta_{est} = \mu_\theta = \frac{\sigma_z^2\hat{\theta} + \sigma_\theta^2 z_\theta}{\sigma_\theta^2 + \sigma_z^2}. \quad (26)$$

Based on (26), to estimate the orientation angle, we need a predicted value with the biggest weight ($\hat{\theta}$), magnetometer measurement value (z), gyroscope variance (σ_θ^2), and magnetometer variance (σ_z^2). The output of this stage is estimated angle which is used to calculate acceleration of accelerometer in *x* and *y* direction.

B. *Position Estimation*

As in the orientation angle estimation, position estimation using particle filter also consists of three stages: the prediction, particle weighting, and estimation. Fig.5 shows the flow chart of the position estimation.

Prediction Stage

The prediction stage is used to predict position of container truck. There are two conditions of prediction stages due to the existence of RFID. The first condition is where the RFID is not detected. A set number of particles is generated around the accelerometer measurement value. The output of accelerometer is the linear acceleration. The prediction stage for accelerometer fulfills the following equations:

$$\begin{bmatrix} x_{t+1} \\ y_{t+1} \end{bmatrix} = \begin{bmatrix} x_t \\ y_t \end{bmatrix} + \begin{bmatrix} \dot{x}_t \\ \dot{y}_t \end{bmatrix} dt + \frac{1}{2}\begin{bmatrix} a\sin\theta_{est} \\ a\cos\theta_{est} \end{bmatrix} dt^2 + \begin{bmatrix} x_e \\ y_e \end{bmatrix}, \quad (27)$$

$$x_e = \sigma_x \operatorname{rand}(n), \quad (28)$$
$$y_e = \sigma_y \operatorname{rand}(n), \quad (29)$$

where x_e and y_e represent accelerometer measurement errors in *x* and *y* directions, respectively. The errors are assumed to be Gaussian with zero mean. The function rand(*n*) in (28) and (29) generates random number. The accelerometer variances in *x* and *y* directions are represented by σ_x^2 and σ_y^2. a is the accelerometer measurement value and θ_{est} is the estimated orientation angle.

For the second condition where RFID is detected, a set number of particle is generated around the RFID measurement value. The output of RFID is the *x* and *y* coordinate. The prediction stage for RFID fulfills

$$\begin{bmatrix} x_{t+1} \\ y_{t+1} \end{bmatrix} = \begin{bmatrix} x_t \\ y_t \end{bmatrix} + \begin{bmatrix} x_e \\ y_e \end{bmatrix}, \quad (30)$$

where x_e and y_e represent RFID measurement errors in *x* and *y* directions which are assumed to be Gaussian.

Particle Weighting Stage

The function for determining the weight of each position particle is

$$W_{t+1}^i\left(q_{t+1}^i, z_{t+1}\right) = \frac{1}{\sigma_q\sqrt{2\pi}} \exp\left(\frac{(q_{t+1}^i - z_{t+1})^2}{2(\sigma_q)^2}\right), \quad (31)$$

where q_{t+1}^i is the position of the *i-th* particle at $t + 1$, z_{t+1} is the position measurement data, and σ_q is its sensor variance.

$$z_{t+1} = [\bar{x}_{t+1} \;\; y_{t+1}] \quad (32)$$

The position weighting is the multiplication of the weights of *x* and *y*, which is

$$W_{t+1}^i\left(q_{t+1}^i, z_{t+1}\right) = \frac{1}{\sigma_x\sqrt{2\pi}} \exp\left(\frac{(x_{t+1}^i - \bar{x}_{t+1})^2}{2(\sigma_x)^2}\right) \frac{1}{\sigma_y\sqrt{2\pi}} \exp\left(\frac{(y_{t+1}^i - \bar{y}_{t+1})^2}{2(\sigma_y)^2}\right) \quad (33)$$

where x_{t+1}^i and y_{t+1}^i are predicted *x* and *y* coordinate of the *i*-th particle. For the first condition, σ_x and σ_y are the iGPS sensor variance, $\bar{x}_{t+1}$ and $\bar{y}_{t+1}$ are the iGPS measurement value. For the second condition, σ_x and σ_y are the RFID sensor variance, $\bar{x}_{t+1}$ and $\bar{y}_{t+1}$ are the RFID measurement value. After that, we find the biggest weight $W_{t+1}^{max,q}$ of position particle as

$$\hat{q} = q_{t+1}(W_{t+1}^{max,q}). \quad (34)$$

Estimation Stage

The same procedures as in the angle estimation, we do multiplication of two Gaussian probability density functions. Therefore, we have position estimation which is represented as

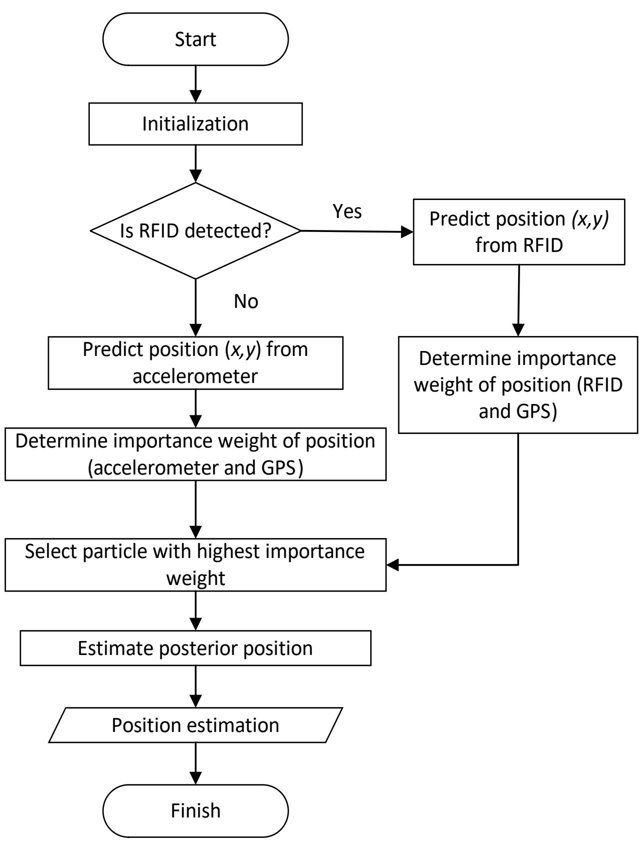

Fig. 5. Position estimation flow chart.

$$q_{est} = \mu = \frac{\sigma_z^2 \hat{q} + \sigma_q^2 z_q}{\sigma_q^2 + \sigma_z^2} \tag{35}$$

In this stage, we have two conditions. For the first condition, σ_z^2 is the iGPS sensor variance, $\hat{q}$ is the predicted value with the biggest weight, σ_q^2 is accelerometer sensor variance, and z_q is the iGPS sensor measurement value. For the second condition, σ_z^2 is the iGPS sensor variance, $\hat{q}$ is predicted value with the biggest weight, σ_q^2 is the RFID sensor variance, and z_q is the iGPS sensor measurement value.

IV. EXPERIMENTAL RESULTS

Localization system using sensor fusion technique, implementing fast and low cost computation is implemented on straight path and circular path of a miniature container truck. The sensor fusion provides the estimated value of the container truck's position close to the true value or reference. This result agrees with the concept that the estimated value made by the IMU sensor is corrected and updated after the iGPS sending the measurement data update.

Fig. 6 shows the performance of sensor fusion in straight path. In this test, the vehicle moves along a circular path. The initial value of the position is at x = 219.5cm and y = 182cm. In this experiment, there is no data loss from the iGPS, thus the data from the sensor fusion technique and that from the iGPS only has minor difference.

In Fig. 7, the container truck performs circular path. The sensor fusion technique is capable to determine the position of the vehicle at any time. The sensor fusion technique can provide container truck's position data even when the iGPS suffers a data loss. This case occurs in the position of x = 19.5 cm and y = 140 cm up to x = 19.5 cm and y = 98 cm in Fig. 6. In that point, the iGPS does not provide data update at all and only send the same data as the previous, therefore there is an empty area. However, by using sensor fusion technique the area can be estimated. The data is derived from the IMU and the previous iGPS data. The sensor fusion technique is also capable to eliminating relative sensor errors by using RFID sensors as absolute sensors. Fig. 7 shows that every time the RFID tag is passed, the estimated position is always repaired. For example, at x = 19.5cm and y = 182cm, the RFID is installed and the identification code of RFID is defined at the coordinate x = 19.5cm and y = 182cm. The RFID has a small variance so that the RFID has a big weight in the positioning of the moment. Although other sensors detect positions far enough to the reference, but if the RFID is detected, the position estimation is fixed by the sensor data from the RFID.

V. CONCLUSION

Sensor fusion technique using the proposed fast and low-cost particle filter algorithm for container-truck localization has been discussed. The method improves the performance of the individual sensors. The accumulated errors from IMU, the loss of signals of iGPS, and the inaccuracy of RFID sensors are not seen after implementing the proposed particle filter. The calculation of the particle filter is fast and is low cost computation since the new probability function of particles, which combine probability density functions of sensors, is derived in a closed-form solution. Experimental results show the data fused from the iGPS, RFID sensors, and IMU provide a fast and precision measurement of the miniature container truck trailer.

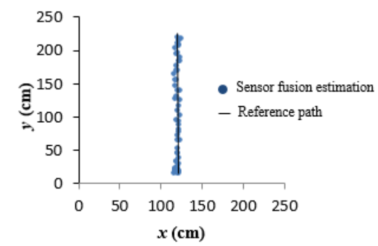

Fig. 6. Straight Path Tracking Result Using Sensor Fusion

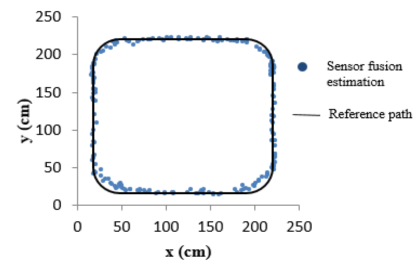

Fig. 7. Circular Path Tracking Result Using Sensor Fusion

REFERENCES

[1] J. J. Van, "Control of automated container terminals," Delft University of Technology, Delft, 2004.

[2] H. J. V. d. Hardt, D. Wolf and R. Husson, "The dead reckoning localization system of the wheeled mobile robot ROMANE," *Proc. 1996 IEEE/SICE/RSJ International Conference on Multisensor Fusion and Integration for Intelligent Systems*, Washington DC, 1996.

[3] E. A. Khatib, M. A. Jaradat, M. Abdel-Hafez and M. Roigari, "Multiple sensor fusion for mobile robot localization and navigation using Extended Kalman Filter," *Proc. 2015 10th International Symposium on Mechatronics and It's Application (ISMA)*, Sharjah, 2015.

[4] J. Z. Sasiadek dan P. Hartana, "Sensor fusion for dead-reckoning mobile robot navigation," *IFAC Workshop on Mobile Robot Technology*, Jejudo Island, 2001.

[5] T. T. Hoang, P. M. Duong, N. T. T. Van, D. A. Viet dan T. Q. Vinh, "Multi-sensor perceptual system for mobile robot and sensor fusion-based localization," *Proc. Int. Conf. on Control, Automation and Information Sciences* (*ICCAIS*), Hanoi, 2012.

[6] E. North, J. Georgy, M. Tarbouci, U. Iqbal dan A. Noureldin, "Enhanced mobile robot outdoor localization using INS/GPS integration," *Proc. Inte. Conf. on Computer Engineering & Systems*, Cairo, 2009.

[7] I. A. S. R. Senevirathna, K. L. U. Sirisena, H. G. N. Sandamali, M. B. Pillai dan A. M. Harsha, "Simulation of mobile robot navigation with sensor fusion on an uneven path," *Proc. Int. Conf. on Circuits, Power, and Computing Technologies (ICCPCT)*, Nagercoil, 2014.

[8] T. Lee, J. Shin dan D. Cho, "Position estimation for mobile robot using in-plane 3-axis IMU and active beacon," *Proc. IEEE International Symposium on Industrial Electronics*, Seoul, 2009.

[9] M. A. K. Jaradat and M. F. Abdel-Hafez, "Enhanced, delay dependent , intelligent fusion for INS/GPS navigation system," *IEEE Sensors Journal*, vol. 14, no. 5, pp. 1545-1554, 2014.

[10] G. G. Rigatos, "Particle filtering for state estimation in nonlinear industrial systems," *IEEE Trans. Instrumentation and Measurement*, vol. 58, no. 11, pp. 3885-3900, 2009.

[11] J. H. Kotecha dan P. M. Djuric, "Gaussian Particle Filter," *IEEE Trans. on Signal Processing*, vol. 51, no. 10, pp. 2592-2601, 2013.

[12] B. S. Choi, "Mobile robot localization in indoor environment using RFID and sonar fusion system," *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, USA, 2009.