UNIJOURN

International Journal of Sustainable Transportation Technology
Vol. 5, No. 2, October, 2022, 55-60

Digital Twin Model Development for Autonomous Tram Localization

Yolanda Tania Mulyadi[1], M Rifqi Rafian Putra[1], Yul Yunazwin Nazaruddin[2,3*], Miranti Indar Mandasari[4]

[1]Department of Engineering Physics, Institut Teknologi Bandung, Bandung, Indonesia
[2]Instrumentation and Control Research Group, Department of Engineering Physics, Institut Teknologi Bandung, Bandung, Indonesia
[3]National Center for Sustainable Transportation Technology, Bandung, Indonesia
[4]Building Physics Research Group, Department of Engineering Physics, Institut Teknologi Bandung, Bandung, Indonesia

**Email: yul@itb.ac.id*

Abstract

The rapid advancement of information technologies led to the rapid growth of various aspects, one of which is autonomous vehicles. Digital twin technology is being frequently developed in autonomous vehicle research, enabling real-time remote monitoring and control of the vehicle's physical assets. This technology can reduce maintenance costs and risks as well as prevent and speed up accident management. This paper proposes a digital twin model for the autonomous tram, one of the vehicles widely explored due to its safety, low emissions, and high capacity. In this research, the proposed digital twin model is utilized to virtually represent the kinematics of the tram prototype in a 2D model from data sent via Message Queuing Telemetry Transport (MQTT) protocol, enabling real-time remote control with low-band consumption. Virtual representation of the tram prototype is gathered via physical sensors and Long Short-Term Memory (LSTM) as the virtual model and controlled by a Stanley controller. The results confirmed that the use of the proposed digital twin model could remotely monitor and control the autonomous tram prototype in real-time conditions.

Keywords

Autonomous tram; Digital twin; Message queuing telemetry transport; Virtual sensor; Stanley controller

1 Introduction

Information technology has developed rapidly over the last few decades. This technology has changed various aspects towards a more effective and cost-efficient business, enabling the industry to grow continuously. In the meantime, Digital twin (DT) is a modern technology that has recently been much examined by many researchers for predictive analysis in various case studies, one of which is in the development of Autonomous Vehicles (AV). With the rise of the industrial era 4.0, the emergence of DT technology brings new applications to AV simulation. This technology enables autonomous vehicles, including trams, to simulate vehicle information in various scenarios, such as missing position information [1], [2]. This approach can potentially reduce AV failure related to position sensor interferences, e.g., if the tram runs under trees, tall buildings, and tunnels.

Digital Twin represents a physical object or assembly using integrated simulations and service data [3]. The DT concept was introduced initially by Michael Grieves to synchronize the physical product and the information contained in the virtual product dynamically, enabling an instantaneous perspective on how the product meets its design goals [4]. Xiong et al. [5] applied a similar concept to the AV case by creating a DT-assisted simulation in a car-following scenario, creating a safe and efficient simulation test implementation.

The most common objective of an autonomous vehicle is to follow the designated trajectory. Previous research has shown several proven algorithms that enable vehicles to follow the trajectory according to the position information [6], [7]. The Stanley controller is one of the simplest path-following algorithms commonly used in autonomous vehicles. This path-following algorithm demonstrates the ability of the controller to track trajectories over steep and wavy terrain [8]. However, the existing path following algorithm requires reliable position data accuracy and is highly dependent on it. The event of missing position information greatly affects its performance.

Therefore, in this paper, a DT model in a path-following scenario is proposed to simulate the missing position information scenario for autonomous trams, one of the

Received July 1, 2022, Revised September 19, 2022, Accepted for publication October 6, 2022

widely explored autonomous vehicles which enable mass transportation in the city [9], [10] to be implemented. To realize the idea mentioned, the main components of the proposed system consist of the following:

(1) Long Short-Term Memory (LSTM) – a neural network model used as a virtual model that generates predictions of vehicle localization data.
(2) Stanley controller – a path-following algorithm that allows the controller to maintain the tram to move in the desired trajectory path.

The tram prototype was used as the research subject implementation in this investigation. This prototype has various localization sensors, such as an Indoor Positioning System (IPS), an Inertial Measurement Unit (IMU), and a wheel encoder. The localization sensor data is then processed via the Unscented Kalman Filter (UKF) estimation algorithm, providing more accurate localization [11], [12]. The DT of this prototype is connected to the physical device through the Message Queuing Telemetry Transport (MQTT) communication network, enabling remote monitoring and control of the autonomous tram [13].

Based on the above explanation, the main objective of this investigation is to develop a digital twin model of the autonomous tram and to enhance the position estimation using a virtual model in the case of any missing position information.

2 Proposed Framework

DT model is proposed using the definition explained in the previous section. As shown in Figure 1, the DT model framework is divided into two parts: physical twin components and digital twin components. The tram of physical twin components acquires physical localization sensor data utilizing the autonomous driving controller. The controller also maintains the tram on a trajectory path using sensor data and sends the data to the digital twin via a wireless router. Then the physical data is used as input to a virtual model as a position prediction model in the scenario of missing position information. The model results are then stored and sent back to the controller.

3 Designed System

The implementation framework of the proposed DT model is shown in Figure 2. The sensor component takes physical data, which is then sent over the wireless network to the digital twin. In the digital twin, the physical data is trained into a virtual model made of LSTM. The virtual model data is then stored and displayed on the Human Machine Interface (HMI). The virtual model is also sent over the wireless network to the autonomous driving controller, which contains the Unscented Kalman Filter (UKF) algorithm to estimate the position by combining heterogeneous sensor data and the Stanley controller algorithm to determine the steer and speed commands to the steering and rear wheel.

In Figure 2, the implementation components include seven parts: sensors, network layer (wireless router and MQTT broker), vehicle DT, data storage and model evaluation, Human Machine Interface (HMI), autonomous driving controller, and actuators. Each of the components is described in detail in the following sections.

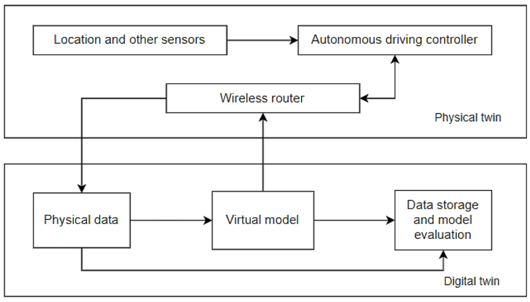

Figure 1 The proposed digital twin framework.

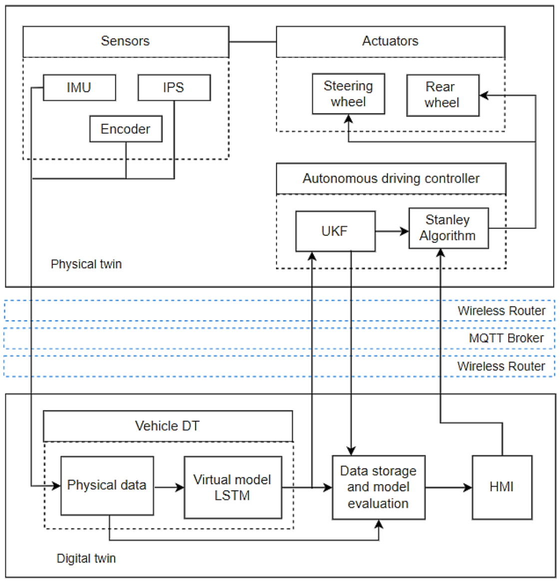

Figure 2 The implementation framework of the DT model. The black dotted lines indicate a group of components in the autonomous tram. The blue dotted lines indicate the network layer between the physical and digital twins. The solid lines indicate the flow of communication data.

3.1 Sensors

Sensors are one of the most crucial components in the system as they are responsible for sensing the condition of the tram in real-time cases. In this work, the tram was equipped with three sensors: Indoor Positioning System (IPS), an Inertial Measurement Unit (IMU), and a wheel encoder. IPS is a sensor system that can detect the position of objects inside a building using radio waves, magnetic fields, and acoustic signals [14], [15]. Further, IPS was used to replace the GNSS because the experimental investigation was carried out indoors. IMU is a sensor device consisting of three main parts: an accelerometer, a gyroscope, and a magnetometer. This sensor was equipped to measure the tram's specific force, angular rate, and orientation [16]. To receive velocity information of the tram, a wheel encoder was attached to the rear wheel of the tram [17]. The rear wheel of the tram is mounted on a rotating rod having grating holes that light can pass through. When the wheel rotates, the grid blocks and transmits light from the source to the sensor, forming pulse waves.

3.2 Network Layer

The network layer is used to connect the physical twin and digital twin. This study uses a wireless router as a transmitter to the internet network. Information is sent through the router to the MQTT broker, a link that allows MQTT clients to communicate.

Figure 3 shows that in MQTT, clients are divided into two components: the sender, known as the publisher, and the recipient of the data, also known as the subscriber. The two components are connected by a third component, the MQTT broker, which directs messages from publishers to subscribers' endpoints on the same topic.

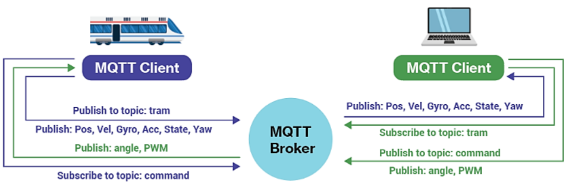

Figure 3 MQTT communication protocol illustration for the autonomous tram.

3.3 Vehicle DT

Vehicle DT consists of physical data received from sensors and a virtual model built by LSTM. A virtual model is built to predict vehicle localization data in the case of missing position information scenarios [1]. LSTM is applied to replace the GNSS with its independence from external conditions. LSTM can model sequential data more accurately than other machine learning algorithms because it has a gating mechanism in each hidden layer, namely input, forget, and output gate, so LSTM can remember essential information from data [18]. The hidden layer has outputs as in the following equations with the layer details represented in Figure 4.

$$c(t) = f_t * c(t-1) + i_t * \tilde{c_t} \tag{1}$$

$$h(t) = \tanh\big(c(t)\big) * o_t \tag{2}$$

where $c(t)$ is the state cell that stores the overall condition of the cell, $\tilde{c_t}$ as the newly processed information, and $h(t)$ is the hidden cell that focuses on storing the most recent state.

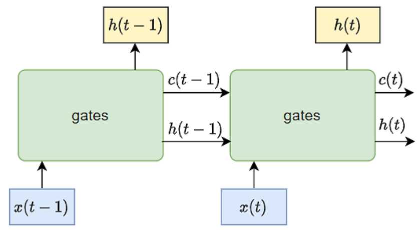

Figure 4 The architecture of LSTM.

The LSTM model is trained using the physical data to build a reliable model. The model architecture is built using LSTM, dropout, and layers with Sigmoid and ReLu activation functions.

3.4 Data Storage and Model Evaluation

LSTM model predictions are then stored in a database together with physical data received from the sensors. The position predictions are evaluated and compared with the actual position measurement. At the same time, the UKF estimation algorithm utilizes the predictions as the correction in the case of missing position information scenarios; then, the result is also stored in this database. All data in the database are visualized to users through a Human Machine Interface (HMI).

3.5 HMI

In implementing this prototype tram system, the user can provide target path input to the tram actuator through the HMI and automatically command the tram to move into the designated path using the Stanley controller algorithm. After the tram is active, each data sent by the physical sensors is processed to the virtual model and displayed on the HMI. The received position data is depicted as a line representing the path formed by the tram based on the localization system used. The

HMI displays the tram's real-time speed, orientation, and acceleration data. This HMI can also display the predictions from the LSTM virtual model compared to the actual IPS measurement results.

3.6 Autonomous Driving Controller

An autonomous driving controller is a controller module that includes UKF and Stanley controller algorithms in this research. UKF algorithm is applied to estimate the tram position in all scenarios. UKF is one of the Kalman filter developments that performs estimation by taking several sigma points and sample points around the current data average based on the covariance [19]. There are two stages in UKF, namely, the prediction and correction stages. In the prediction stage, UKF predicts the desired variable and the covariance matrix, which are written in equations 3 and 4. Both predictions are then corrected in the correction stage using more accurate data, such as IPS or LSTM prediction. The outputs of this stage are represented in equations 5 and 6.

$$\check{\mathbf{x}}_k = \sum_{i=0}^{2N} \alpha^{(i)}\, \check{\mathbf{x}}_k^{(i)} \tag{3}$$

$$\check{\mathbf{P}}_k = \sum_{i=0}^{2N} w^{(i)} \left(\check{\mathbf{x}}_k^{(i)} - \check{\mathbf{x}}_k\right)\left(\check{\mathbf{x}}_k^{(i)} - \check{\mathbf{x}}_k\right)^T + \mathbf{Q}_{k-1} \tag{4}$$

$$\hat{\mathbf{x}}_k = \check{\mathbf{x}}_k + \mathbf{K}_k \left(\hat{y}_k' - \sum_{i=0}^{2N} \alpha^{(i)} \hat{y}_k'^{(i)}\right) \tag{5}$$

$$P_y = \sum_{i=0}^{2N} \alpha^{(i)} \left(\hat{y}_k'^{(i)} - \hat{y}_k'\right)\left(\hat{y}_k'^{(i)} - \hat{y}_k'\right)^T + R_k \tag{6}$$

The position estimation from UKF is used as a reference input for the Stanley controller to determine the next tram control command. Stanley controller is a path-tracking approach algorithm that applies the front axle as the reference point [6].

Figure 5 shows the principle of the Stanley controller in following the target trajectory. This algorithm minimizes orientation error, ψ(t), and cross-track error, e(t). The output of steering angle control is obtained through the following equation,

$$\delta(t) = k_1 * \psi(t) + \tan^{-1}\left(\frac{k_2 * e(t)}{1 + v_f(t)}\right) \tag{7}$$

The steering angle command was sent to the steering wheels as actuators.

3.7 Actuators

The system's actuator is a servo motor for controlling the steering wheel and a dynamo motor for controlling the speed of a tram prototype. The servo motor used in this research is MG996R. This servo can rotate 180 degrees, allowing this servo to represent the steering wheel. For the implementation of this dynamo motor, an 88002-train motor is used to control speed through a pulse width modulation (PWM) control mechanism.

All components of this system are connected through various communication protocols. The communication protocol used in the design of the system is illustrated in Figure 6. This communication involves a Wireless Local Area Network (WLAN), MQTT, Serial, Universal Asynchronous Receiver-Transmitter (UART), and Inter-Integrated Controller (I2C). The application of protocols for each component corresponds to the component's design. For example, WLAN and MQTT are used for long-distance connections without cables so that the implementation can be used remotely. I2C is used to reduce the need for pins used in sensor readings compared to serial, while UART is used to accommodate communication between controllers and IPS.

The design of the entire system is then implemented in the form of a prototype. It will be explained in more detail in the next section.

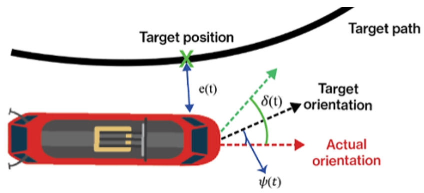

Figure 5 Stanley controller's control law illustration.

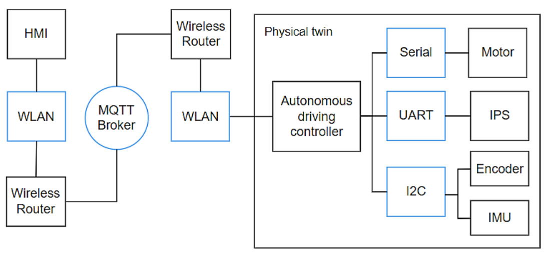

Figure 6 The communication topology of the DT model. The solid black lines indicate components in the autonomous tram. The solid blue lines indicate the communication protocol.

4 System Implementation

According to the proposed system, the DT implementation platform has been developed. Commands to the tram can be given through the HMI that has been designed. Through this HMI, the tram's movement was also visualized in real-time with UKF position estimation with and without the LSTM model correction. All the data from HMI to tram and vice versa are sent through wireless routers and an MQTT broker. Resembling the designed HMI, a track is built inside a room as a ramp for tram movement. The tram is placed on the track with all sensors, actuators, and an autonomous driving controller equipped in the tram's body. The DT testing platform is shown in Figure 7.

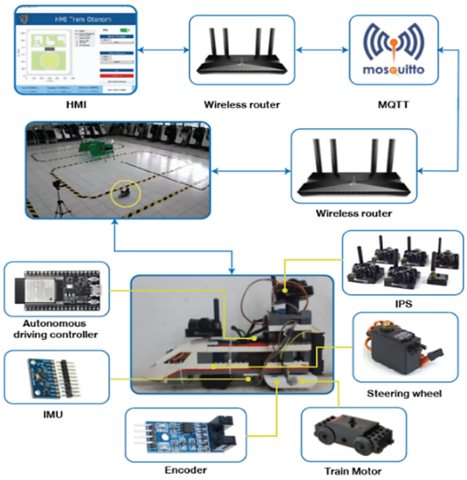

Figure 7 Proposed system implementation.

DT implementation is initialized in this research by providing the tram target trajectory through HMI. The tram is expected to move following the input trajectory. During the testing, the IPS was turned off several times, representing the missing information of GNSS due to environmental disturbances. In this case, the virtual model was evaluated to determine whether it could replace the position data accurately so that the Stanley controller could keep the tram moving in line with the track.

DT implementation was performed 50 times to collect more data in IPS working conditions. The data has been collected was then used for LSTM model training. The model was trained with input data (x train) and output data (y train) represented in equation 8, with the result shown in Table 1.

$$X_{train} = [\boldsymbol{V}, yaw, I_{steer}, IMU_a, gyro_{rpy}];\ Y_{train} = [\Delta x, \Delta y] \quad (8)$$

Table 1 LSTM virtual model training performance

Loss	Accuracy	X position maximum range	Y position maximum range
0.00104	0.932	-13.12 to 15.14	-14.31 to 9.82

The model was evaluated by predicting 1000 timestamps of collected data. Error model predictions were compared to the change of position for each time step, and the result is visualized in Figure 8.

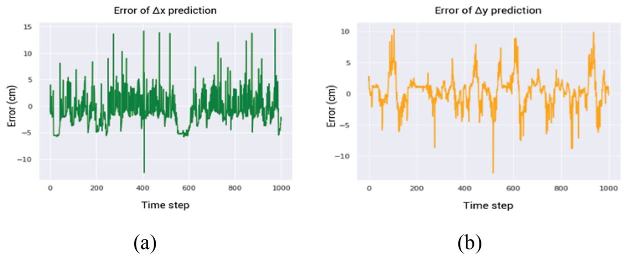

Figure 8 Position prediction error of virtual model on the (a) x and (b) y axes.

The trained model was then implemented in the system. During implementation, the trained LSTM model predicted changes in tram position in real-time and added position data in the previous time step as UKF correction data for missing position information scenarios. The results of the implementation are explained in the next section.

5 Result and Discussion

The experimental investigation was done by giving the target path from the HMI on the digital twin to the tram prototype. The target path is received by the controller and processed using the Stanley controller algorithm into actuator input. The actuator movement is measured by the proposed physical and virtual sensors and then transmitted back to the digital twin. The experiment results are shown in Figure 9, in which the tram prototype moves along the closed area shown as a dark grey box where it is assumed that no signal has been received from the IPS. It can be seen in the figure that there are 4 lines, namely the target line, IPS, localization system with LSTM, and localization system without LSTM. In the experiments, IPS was turned on and became the corrector for the virtual sensor. It causes the IPS line, systems with LSTM virtual model, and without LSTM virtual model to show similar results. It can be seen that the prototype moves along the target path using IPS position data under normal conditions and the LSTM virtual model for the case of loss of position information. Further analysis related to model performance is described in Table 2.

As shown in Table 2, the model's performance is tested by comparing the error during the availability of position information and the virtual model using the Mean Absolute Error (MAE) metric, which describes the average system error in determining the tram position. It can be observed that from these metrics, the virtual model on the proposed digital twin can eliminate system errors without virtual models by 84% on the x-axis and 81% on the y-axis.

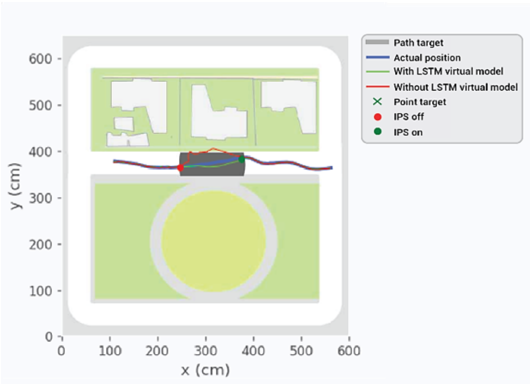

Figure 9 Visualization of position estimation experiment on digital twin model.

Table 2 LSTM virtual model training performance

Position Sensor Availability	Virtual Model Availability	Axis	MAE (cm)
Available	Available	x	1.42
		y	1.25
Not Available	Not Available	x	63.60
		y	63.86
	Available	x	10.90
		y	14.90

6 Conclusion

Research has been carried out to develop a digital twin model for the autonomous tram. The digital twin model has been successfully implemented on the autonomous tram so that it can display physical sensor data along with the estimated position of the sensor and virtual model in the real-time environment. Several tests carried out through the digital twin model show that the virtual sensors system can make the Stanley controller keep the tram running on the specified path. The test results show that the LSTM virtual model can improve positioning accuracy and eliminate most position prediction errors due to the loss of position information by more than 80% based on the MAE metric.

Acknowledgment

This research was funded by the Indonesian Ministry of Education, Culture, Research, and Technology under the Fundamental Research scheme 2022 and partially by the Institut Teknologi Bandung Research Program 2022.

References

[1] M. R. R. Putra, Y. T. Mulyadi, Y. Y. Nazaruddin, and F. A. Maani, “Localization Method for Missing Information in Autonomous Vehicle Using LSTM and UKF Approaches,” *2022 13th Asian Control Conference, Proceedings*, pp. 429–434, 2022.

[2] Y. Nazaruddin, F. Maani, E. R. Muten, P. W. L. Sanjaya, G. Tjahjono, and J. A. Oktavianus, “An Approach for the Localization Method of Autonomous Vehicles in the Event of Missing GNSS Information,” *2021 60th Annual Conference of the Society of Instrument and Control Engineers of Japan*, pp. 881–886, 2021.

[3] A. Fuller, Z. Fan, C. Day, and C. Barlow, “Digital Twin: Enabling Technologies, Challenges and Open Research,” *IEEE Access*, vol. 8, pp. 108952–108971, 2020.

[4] M. Grieves, “Digital Twin: Manufacturing Excellence through Virtual Factory Replication,” *White paper*, vol. 1, pp. 1–7, 2014.

[5] H. Xiong, Z. Wang, G. Wu, and Y. Pan, “Design and Implementation of Digital Twin-Assisted Simulation Method for Autonomous Vehicle in Car-Following Scenario,” *J Sens*, vol. 2022, 2022.

[6] A. AbdElmoniem, A. Osama, M. Abdelaziz, and S. A. Maged, “A path-tracking algorithm using predictive Stanley lateral controller,” *Int J Adv Robot Syst*, vol. 17, no. 6, Dec. 2020.

[7] J. M. Snider, “Automatic Steering Methods for Autonomous Automobile Path Tracking,” *Robotics Institute, Pittsburgh, PA, Tech. Rep. CMU-RITR-09-08*, 2009.

[8] G. M. Hoffmann, C. J. Tomlin, M. Montemerlo, and S. Thrun, “Autonomous automobile trajectory tracking for off-road driving: Controller design, experimental validation and racing,” *Proceedings of the American Control Conference*, pp. 2296–2301, 2007.

[9] V. Ndlovu, P. Newman, V. Ndlovu, and P. Newman, “Leapfrog Technology and How It Applies to Trackless Tram,” *J Transp Technol*, vol. 10, no. 3, pp. 198–213, 2020.

[10] P. Newman *et al.*, “The Trackless Tram: Is it the Transit and City Shaping Catalyst we have been waiting for?,” *J Transp Technol*, vol. 09, no. 01, pp. 31–55, 2020.

[11] A. Giannitrapani, N. Ceccarelli, F. Scortecci, and A. Garulli, “Comparison of EKF and UKF for spacecraft localization via angle measurements,” *IEEE Trans Aerosp Electron Syst*, vol. 47, no. 1, pp. 75–84, 2011.

[12] I. Faruqi, M. B. Waluya, Y. Y. Nazaruddin, and T. A. Tamba, “Train Localization using Unscented Kalman Filter–Based Sensor Fusion,” *IJSTT*, vol. 1, no. 2, pp. 35–41, 2018.

[13] G. Sasikala *et al.*, “IoT real time data acquisition using MQTT protocol,” *J Phys Conf Ser*, vol. 853, no. 1, p. 012003, 2017.

[14] M. Kaluža, K. Beg, and B. Vukelić, “Analysis of an Indoor Positioning Systems,” *Zbornik Veleučilišta u Rijeci*, vol. 5, no. 1, pp. 13–32, 2017.

[15] R. F. Brena, J. P. García-Vázquez, C. E. Galván-Tejada, D. Muñoz-Rodriguez, C. Vargas-Rosales, and J. Fangmeyer, “Evolution of Indoor Positioning Technologies: A Survey,” *J Sens*, vol. 2017, 2017.

[16] J. Zhao, “A Review of Wearable IMU (Inertial-Measurement-Unit)-based Pose Estimation and Drift Reduction Technologies,” *J Phys Conf Ser*, vol. 1087, no. 4, p. 042003, 2018.

[17] R. Petrella, M. Tursini, L. Peretti, and M. Zigliotto, “Speed measurement algorithms for low-resolution incremental encoder equipped drives: A comparative analysis,” *International Aegean Conference on Electrical Machines and Power Electronics and Electromotion ACEMP’07 and Electromotion’07 Joint Conference*, pp. 780–787, 2007.

[18] J. Gonzalez and W. Yu, “Non-linear system modeling using LSTM neural networks,” vol. 51, no. 13, pp. 485–489, 2018.

[19] E. A. Wan and R. van der Merwe, “The unscented Kalman filter for nonlinear estimation,” *IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium, AS-SPCC 2000*, pp. 153–158, 2000.