

Article

Wheel Odometry with Deep Learning-Based Error Prediction Model for Vehicle Localization

Ke He, Haitao Ding *, Nan Xu and Konghui Guo

State Key Laboratory of Automotive Simulation and Control, Jilin University, Changchun 130025, China; heke20@mails.jlu.edu.cn (K.H.)

* Correspondence: dinght@jlu.edu.cn

Abstract: Wheel odometry is a simple and low-cost localization technique that can be used for localization in GNSS-deprived environments; however, its measurement accuracy is affected by many factors, such as wheel slip, wear, and tire pressure changes, resulting in unpredictable and variable errors, which in turn affect positioning performance. To improve the localization performance of wheel odometry, this study developed a wheel odometry error prediction model based on a transformer neural network to learn the measurement uncertainty of wheel odometry and accurately predict the odometry error. Driving condition characteristics including features describing road types, road conditions, and vehicle driving operations were considered, and models both with and without driving condition characteristics were compared and analyzed. Tests were performed on a public dataset and an experimental vehicle. The experimental results demonstrate that the proposed model can predict the odometry error with higher accuracy, stability, and reliability than the LSTM and WhONet models under multiple challenging and longer GNSS outage driving conditions. At the same time, the transformer model's overall performance can be improved in longer GNSS outage driving conditions by considering the driving condition characteristics. Tests on the experimental vehicle demonstrate the model's generalization capability and the improved positioning performance of dead reckoning when using the proposed model. This study explored the possibility of applying a transformer model to wheel odometry and provides a new solution for using deep learning in localization.

Keywords: autonomous driving; localization; wheel odometry; deep learning; transformer model



Citation: He, K.; Ding, H.; Xu, N.; Guo, K. Wheel Odometry with Deep Learning-Based Error Prediction Model for Vehicle Localization. *Appl. Sci.* **2023**, *13*, 5588. <https://doi.org/10.3390/app13095588>

Academic Editors: Yi Han and Xiaojun Tan

Received: 7 April 2023
Revised: 25 April 2023
Accepted: 28 April 2023
Published: 30 April 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Autonomous driving has become a broad research topic in recent years, and it has great potential to enhance driving safety and improve transport efficiency [1]. An autonomous driving system includes many modules, such as decision, planning, and control modules [2], which require accurate knowledge of the vehicle's position to perform correct driving decisions and actions [3]. For example, an error of a few decimeters may cause the vehicle to position itself in the wrong lane, leading to incorrect driving decisions, and thus to traffic accidents. Therefore, self-driving cars require robust localization systems with decimeter-level or even centimeter-level accuracy [4].

Global navigation satellite systems (GNSS) [5] are the most commonly used method for vehicle localization. A GNSS is suitable for open areas but is less reliable in obscured environments involving structures such as tall buildings and viaducts due to signal occlusion, multipath errors, and other factors. A GNSS can reach the centimeter level in open areas with real-time kinematic (RTK) [6] technology; however, the effect of multipath errors on localization estimates persists, so GNSS alone cannot provide consistent and reliable localization accuracy. A GNSS is usually integrated with an IMU [7] to form a localization system. IMUs consist of accelerometers and gyroscopes, which can measure vehicle acceleration and angular velocity. This information can be further used to calculate vehicle position relative to its initial position in a process called dead reckoning. However,

IMU measurements are affected by multiple noises that are amplified during the multiple integrations of acceleration to displacement and angular velocity to attitude angle, grow exponentially, and accumulate over time.

Wheel odometry measures the number of pulses per unit of time at the wheel through wheel encoders to calculate wheel speed and travel distance. Due to the necessity of an anti-lock brake system (ABS), wheel encoder sensors are mounted on the vehicle, making wheel odometry a localization technology that can be used universally. Compared to IMU's accelerometer sensors, wheel odometry requires fewer integration steps to determine the vehicle position, reducing errors in the integration process and providing a better positioning solution. Wheel odometry was initially widely used in mobile robotics and was then gradually applied to vehicles to complete localization tasks. Thrun, S et al. [8] improved the problem of poor estimation of vehicle attitude and IMU bias when GNSS signals are weak or down, which occurs in common GNSS/IMU fusion algorithms, by considering wheel odometry. Funk, N et al. [9] complemented visual-inertial odometry with wheel odometry for automotive applications. In addition, wheel odometry can be the primary localization algorithm in some special cases, such as low light conditions, low speed driving, and parking scenes [10,11].

The accuracy of wheel odometry-based localization is affected by many factors; on the one hand, it is affected by model parameters, such as tire diameter. Although these can be calibrated during installation, they can change dynamically when the vehicle is driven due to certain factors, such as wheel wear and load changes, which can affect the odometry output. On the other hand, the measurement accuracy of odometry is affected by wheel slip. In addition, unevenness on the road surface, such as bumps and potholes, interferes with the odometry results. Therefore, the output errors of wheel odometry have high uncertainty, and accurately predicting the wheel odometry error is key to improving wheel odometry-based localization accuracy.

Fazekas, M et al. proposed an off-line iterative estimation algorithm in [12] and an online estimation method in [13] in which Kalman filter and the least squares algorithm are performed in an iterative loop for wheel circumference estimation of autonomous vehicles to improve wheel odometry's accuracy. They further estimated the wheel circumferences recursively to improve the wheel speed estimation with a nonlinear least squares method in [14]. The study [15] calibrated the parameters of the wheel odometry model with Gauss-Newton regression and Kalman filter. Welte A et al. [16] presented a method able to accurately calibrate the model parameters by using a Rauch-Tung-Striebel smoothing scheme which enables to obtain state estimates close to the ground truth. In summary, these studies used traditional state estimation methods to improve wheel odometry's accuracy. However, these methods rely on precise parameter adjustment, which lacks adaptability to various driving conditions. On the other hand, the wheels are affected by many factors of road conditions and their own parameters so that it is difficult to obtain an accurate model under the action of multiple factors using traditional state estimation methods. Deep learning is a data-driven approach that can learn complex nonlinear properties and uncertainty from data. Researchers have proposed deep learning-based techniques to discover the error drift characteristics of IMUs over time for better GPS/INS combined navigation solutions [17–20]. Several deep learning techniques have also been applied to wheel odometry: an LSTM model was proposed to learn the uncertainty of wheel odometry in [21], and a WhONet model was proposed in [22], which further improved the performance compared to the LSTM model.

The transformer model [23] emerged in the field of natural language processing (NLP), and it was first used for machine translation tasks to achieve SOTA effects. A transformer model can fully utilize the GPU resources for parallel computation to accelerate the training speed. Compared with traditional models such as RNN and LSTM that deal with time series, transformer models do not rely on past hidden states to capture the dependence on previous information but learns the association of information at each moment in a time series as a whole, avoiding the risk of losing past information and reducing the performance degradation caused by long-term dependence.

This study aims to explore a new low-cost localization method that combines wheel odometry and deep learning techniques. The main contributions of this paper are as follows:

1. A transformer-based error prediction model is developed to learn the measurement uncertainty of wheel odometry, and the end-to-end correspondence from the wheel speed to the travel distance error is established. Tests on a publicly available dataset and an experimental vehicle are performed to verify that the proposed model can accurately predict the output error of wheel odometry and improve the accuracy, stability, and reliability compared to LSTM and WhONet under various driving conditions. In addition, the model trained using the public dataset is transferred to the experimental vehicle to verify the model's generalization capability and the improved positioning performance of dead reckoning by combining the model.
2. Driving condition characteristics, including features describing road types, road conditions, and vehicle driving operations, are considered. Models with different inputs are developed and compared, one with and the other without driving condition characteristics. The tests demonstrate that the model that considers driving characteristics has better adaptability to longer GNSS outage driving conditions.

The rest of the paper is organized as follows: Section 2 describes the dead reckoning method based on wheel odometry; Section 3 presents the Transformer model and the selected input features; Section 4 defines the dataset and metrics used to train the model and evaluate its performance, as well as the model parameters, in addition to showing the process of model parameter tuning; Section 5 demonstrates the results based on a publicly available dataset and when the model is applied to real vehicles, followed by a relevant discussion; finally, the entire paper is concluded in Section 6.

2. Dead Reckoning Based on Wheel Odometry

The dead reckoning (DR) system relies on odometry, which measures the distance traveled by the vehicle, and the angular velocity obtained from the chassis or a gyroscope in IMUs, to estimate the relative position forward from the current position.

The wheel odometry is based on wheel encoders to track the number of revolutions each wheel has made, and each wheel's measured angular velocity can be derived. Then, the distance traveled by each wheel per unit of time is calculated by combining the wheel radius. According to wheel odometry, the vehicle's measured travel distance from moment $t - 1$ to moment t can be expressed as:

$$x_{mea} = \int_{t-1}^t (w_{rl}r + w_{rr}r)/2 dt \quad (1)$$

where w_{rl} and w_{rr} are the measured left and right rear wheel angular velocities, respectively, and r is the nominal wheel radius.

Many factors influence the measurement accuracy of wheel odometry. First, the measurement of wheel angular velocity contains errors; second, some factors, such as the vehicle load changes, change the wheel radius dynamic; third, some aspects, such as wheel slip and road potholes, lead to the wheel rolling distance not being equal to the actual distance the vehicle travels.

The vehicle's real travel distance from moment $t - 1$ to moment t can be expressed as:

$$x_{real} = \int_{t-1}^t ((w_{rl} + \varepsilon_{w_{rl}})(r + \varepsilon_{rl}) + (w_{rr} + \varepsilon_{w_{rr}})(r + \varepsilon_{rr})) dt/2 + \varepsilon_s \quad (2)$$

where $\varepsilon_{w_{rl}}$ is the error of the actual left rear wheel angular velocity compared with w_{rl} , $\varepsilon_{w_{rr}}$ is the error of the real right rear wheel angular velocity compared with w_{rr} , ε_{rl} is the error between r and the real left rear wheel radius during vehicle driving, ε_{rr} is the error between

r and the real right rear wheel radius during vehicle driving, and ε_s is the error between the rolling distance and the actual travel distance.

Let x_{bias} be the error of the travel distance:

$$x_{bias} = x_{mea} - x_{real} \quad (3)$$

where x_{mea} is the measured travel distance.

In practice, the point coordinates measured by RTK-GPS have centimeter-level accuracy and can be regarded as the true value of the localization coordinates. Suppose p_1, p_2, \dots, p_k are the recorded points from moment $t - 1$ to moment t , then x_{real} can be expressed as:

$$x_{real} = \sum_{i=1}^{k-1} |p_{i+1} - p_i| \quad (4)$$

where $|p_{i+1} - p_i|$ is the distance between p_i and p_{i+1} on the Earth's surface, which can be calculated according to the Vincenty formula given in [24].

Suppose the vehicle's travel distance in a short time Δt is Δs and the vehicle's heading angle is ψ , the transverse and longitudinal displacements Δx and Δy are:

$$\Delta x = \Delta s \cos \psi \quad (5)$$

$$\Delta y = \Delta s \sin \psi \quad (6)$$

The relative displacements between two moments t_1 and t_2 can be regarded as accumulating relative displacements in a very short time Δt . Therefore, the transverse displacement x_{12} and the longitudinal displacement y_{12} from moment t_1 to moment t_2 can be expressed as:

$$x_{12} = \sum_{i=1}^n \Delta s_i \cos \psi_i \quad (7)$$

$$y_{12} = \sum_{i=1}^n \Delta s_i \sin \psi_i \quad (8)$$

According to Equations (5)–(8), after obtaining the travel distance between adjacent moments with the wheel odometry, the vehicle's relative displacement between two moments can be calculated by combining the vehicle's heading angle. Figure 1 shows the role of the wheel odometry error prediction model in the localization system. With the established model, the travel distance error can be predicted accurately, and the travel distance output by the odometry can be corrected so that a more accurate position of the next moment can be obtained after dead reckoning.

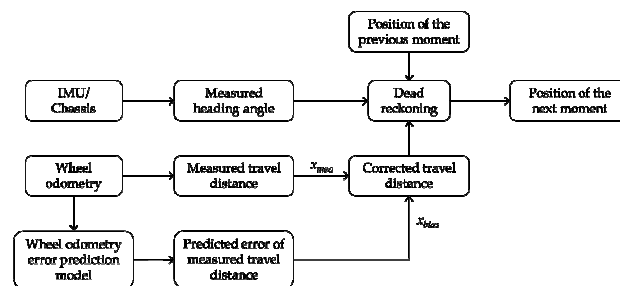


Figure 1. Role of the wheel odometry error prediction model in the localization system.

3. Transformer Model

As the output of wheel odometry is affected by many factors, the measurement error has high uncertainty and keeps changing and accumulating over time. Transformer is a deep learning model that processes time series, and the use of Transformer is studied in this

paper to learn the error characteristics of wheel odometry from time series for more accurate localization. In this section, a Transformer-based error prediction model is established to predict the error of the wheel odometry output.

The Transformer model is based on encoder/decoder architecture. The original input is translated into the model input containing location information through embedding and location coding. The encoder consists of a stack of N identical layers with two sub-layers. The first sub-layer is a multi-head self-attention layer used to extract the features of different attention heads of the model. The second sub-layer is a fully connected feed-forward network, which can enhance the nonlinear representation ability of the model. Both sub-layers use residual connection and normalization. Residual connection further enhances the fitting ability of the model, and normalization can avoid the problem of the slow convergence of the model caused by the parameters being too large or too small. The decoder is also composed of N identical layers stacked. In addition to the two sub-layers in each encoder layer, the decoder inserts a third sub-layer, which is used to perform a multi-head attention calculation on the output of the last encoder. Similar to the encoder, the decoder uses residual connection and normalization for each sub-layer. The first self-attention sub-layer of the decoder is designed in a masked form to ensure that the prediction of position i depends only on the known output at positions less than i . The output of the decoder is converted to the required dimension through a linear layer, and the value with the highest probability calculated through the softmax layer is the final output of the model.

Considering that different driving conditions may influence the wheel states and affect the odometry output, two different Transformer error prediction models are designed. The first model contains four-wheel wheel speed as the input to the encoder, and the second model considers the driving condition features and four-wheel wheel speed as the input to the encoder. Both models output the predicted error sequence. The Transformer model that does not consider the driving conditions is referred to as Transformer-NDC for convenience, and the Transformer model that considers the driving conditions is referred to as Transformer-DC for convenience. The structures of the two models are shown in Figure 2.

Referring to the different driving conditions in the public dataset (IO-VNBD) [25] and common driving condition features, a total of 29 features describing different driving conditions were selected, as shown in Table 1. Features 1–9 describe different road types, features 10–15 describe different road conditions, and features 16–29 describe various vehicle driving operations. The data subsets are preprocessed in combination with the selected features. The feature value takes the value of 1 when the corresponding driving condition of the data subset contains the related feature; otherwise, it takes the value of 0.

Table 1. Selected characteristics describing different driving conditions.

Serial Number	Features	Serial Number	Features	Serial Number	Features	Serial Number	Features
1	Roundabout	9	Valleys	17	Sharp left and right turns	25	Bumps
2	Mountain roads	10	Rain	18	Swift maneuvers	26	Zig-zag driving
3	Country roads	11	Dirt roads	19	Slipping	27	Approximate straight-line motion
4	Expressway	12	Gravel roads	20	Successive left and right turns	28	Stationary
5	Town-center driving	13	Wet roads	21	Varying acceleration within a short time	29	Parking
6	Inner-city driving	14	Mud roads	22	U-turns		
7	Winding roads	15	Potholes	23	Reverse driving		
8	Residential roads	16	Hard brake	24	Drifts		

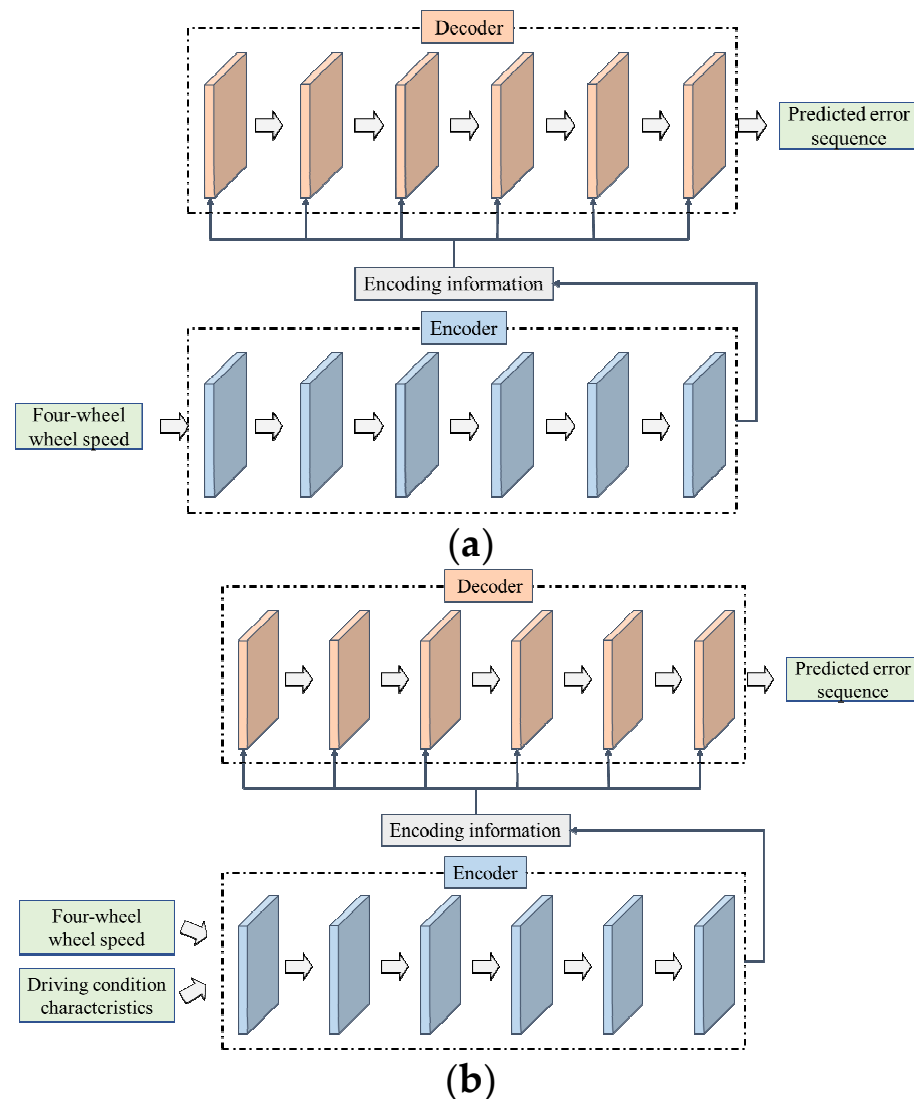


Figure 2. Two different Transformer models. (a) Transformer model that does not consider the characteristics of driving conditions. (b) Transformer model that considers the characteristics of driving conditions.

4. Experimental Setup

The model is trained using the Pytorch framework commonly used for deep learning. The dataset for training and testing is the IO-VNB dataset [25], which records driving data with a total length of 5700 km and a total driving time of 98 h in multiple driving conditions. This dataset captures multiple signals such as longitudinal vehicle acceleration, yaw rate, heading angle, GPS coordinates (latitude and longitude), and wheel speed from each timestamp of the vehicle ECU at a sampling interval of 10 Hz.

4.1. Training

To facilitate model comparison, the model's training set is consistent with the training set in [22], as shown in Table 2, which is characterized by about 1590 min of drive time over a total distance of 1165 km. The model training parameters are shown in Table 3. The model uses the Adam optimizer, where $\beta_1 = 0.9$, $\beta_2 = 0.98$, $\epsilon = 10^{-9}$. During training, the learning rate varies according to the following equation:

$$lrate = d_{\text{model}}^{-0.5} \cdot \min(step_num^{-0.5}, step_num \cdot warmup_steps^{-1.5}) \quad (9)$$

where $warmup_steps$ is a constant taking the value of 100. Equation (9) means that the learning rate increases linearly in the first $warmup_steps$ step and decreases proportionally to the reciprocal of the square root of the number of steps $step_num$. The loss function used in training is the mean absolute error loss function.

Table 2. IO-VNB data subsets used for the Transformer model training.

Serial Number	IO-VNB Data Subset	Serial Number	IO-VNB Data Subset
1	V-S1	18	V-Vta27
2	V-S2	19	V-Vta28
3	V-S3c	20	V-Vta29
4	V-S4	21	V-Vta30
5	V-St1	22	V-Vtb1
6	V-M	23	V-Vtb2
7	V-Y2	24	V-Vtb5
8	V-Vta2	25	V-Vtb9
9	V-Vta8	26	V-Vw4
10	V-Vta9	27	V-Vw5
11	V-Vta10	28	V-Vw14b
12	V-Vta13	29	V-Vw14c
13	V-Vta16	30	V-Vfa01
14	V-Vta17	31	V-Vfa02
15	V-Vta20	32	V-Vfb01a
16	V-Vta21	33	V-Vfb01b
17	V-Vta22		

Table 3. Training parameters of Transformer-NDC and Transformer-DC.

Parameters	Transformer-NDC	Transformer-DC
Number of encoders and decoders N	6	6
Output dimension d_{model} of the sublayers and the embedding layer	512	256
Layer number d_{ff} of the intermediate layer in fully connected feed-forward networks	32	64
Number of attention heads h	8	8
Batch size	32	32
Epochs	100	100
Time step (sliding window)	11 s	11 s
Random inactivation ratio <i>Dropout</i>	0	0
Activation function	ReLU	ReLU

4.2. Testing

Two evaluation metrics, the cumulative root mean square error (CRSE) and the cumulative true error (CTE), are listed in [22]. The CRSE describes the cumulative root mean square of the prediction error per 1 s within the total duration N_t . This metric ignores the negative sign of the estimation error and thus provides a better understanding of the performance of the positioning technique. CTE measures the sum of the prediction errors per 1 s within the total duration N_t . The CTE metric mainly reflects whether the estimated value is higher or lower than the actual value. Compared with the CRSE, the CTE is less realistic when comparing the performances of positioning techniques. Therefore, the CRSE is used to evaluate the model performance in this paper. The CRSE is defined as follows:

$$CRSE = \sum_{t=1}^{N_t} \sqrt{e_{pred}^2} \quad (10)$$

where t is the sampling period and e_{pred} is the prediction error.

Mean (μ): the average value of the CRSE for all series to reveal the average prediction accuracy of the model in each driving condition.

$$\mu = \frac{1}{N_s} \sum_{i=1}^{N_s} \text{CRSE} \quad (11)$$

where N_s is the total number of sequences in each driving condition.

Standard deviation (σ): this metric indicates the variation in the CRSE in all series. It demonstrates the stability of the model.

$$\sigma = \sqrt{\frac{\sum (\text{CRSE}_i - \mu)^2}{N_s}} \quad (12)$$

Maximum (max): the maximum CRSE value among all sequences evaluated in each driving condition. It indicates the model's reliability, which means the model cannot be applied to odometry error correction if the maximum CRSE is too large.

Minimum (min): the minimum CRSE value of all sequences evaluated in each driving condition. It indicates the model's highest possible accuracy.

The data subsets for performance evaluation are shown in Table 4. The test was first performed on the V-Vw12 dataset, which depicts near-straight-line driving on a highway, to assess the model's performance under relatively easy driving conditions. Nevertheless, the highway scenario can be considered challenging due to the large distances covered per second. Since more challenging driving conditions, such as wet roads, roundabouts, and hard brake, are pretty challenging for the localization algorithm [26] and introduce high uncertainty to wheel odometry measurements, the model's performance was simultaneously evaluated under challenging driving conditions. The data subsets used in these tests were divided into a number of test sequences of 10 s lengths each that predicted the changes in the subsequent 1 s. In addition, the data subsets were split into several test sequences of 30, 60, 120, or 180 s depending on the outage driving condition being evaluated to test the model's performance in longer-term GNSS outage driving conditions.

Table 4. IO-VNB data subsets for performance evaluation.

Driving Condition	IO-VNB Data Subset	Total Travel Time, Distance Traveled, Speed and Acceleration
Motorway	V-Vw12	1.75 min, 2.64 km, 82.6 to 97.4 km/h, -0.06 to 0.07 g
Sharp cornering and successive left and right turns	V-Vw6	2.1 min, 1.08 km, 3.3 to 40.7 km/h, -0.34 to 0.26 g
	V-Vw7	2.8 min, 1.23 km, 0.4 to 42.2 km/h, -0.37 to 0.37 g
	V-Vw8	2.7 min, 1.12 km, 0.0 to 46.4 km/h, -0.37 to 0.27 g
Wet road	V-Vtb8	1.2 min, 1.35 km, 60.9 to 76.5 km/h, -0.35 to 0.08 g
	V-Vtb11	0.7 min, 0.84 km, 65.1 to 75.3 km/h, -0.05 to 0.12 g
	V-Vtb13	2.1 min, 0.99 km, 7.5 to 43.3 km/h, -0.31 to 0.22 g
Quick changes in vehicle's acceleration	V-Vfb02e	1.6 min, 1.52 km, 37.4 to 73.9 km/h, -0.24 to 0.19 g
	V-Vta12	1.1 min, 1.27 km, 44.7 to 85.3 km/h, -0.44 to 0.13 g
Roundabout	V-Vta11	1.0 min, 0.92 km, 26.8 to 97.7 km/h, -0.45 to 0.15 g
	V-Vfb02d	1.5 min, 0.84 km, 0.0 to 57.3 km/h, -0.33 to 0.31 g
Hard brake	V-Vw16b	2.0 min, 1.99 km, 1.3 to 86.3 km/h, -0.75 to 0.29 g
	V-Vw17	0.5 min, 0.54 km, 31.5 to 72.7 km/h, -0.8 to 0.19 g
Longer GNSS outage	V-Vtb3	13.8 min, 0.71 km, 0.0 to 37.5 km/h, -0.23 to 0.33 g
	V-Vfb02a	59.9 min, 96.5 km, 0.0 to 122.3 km/h, -0.5 to 0.37 g
	V-Vfb02b	18.3 min, 7.69 km, 0.0 to 84.3 km/h, -0.5 to 0.35 g

4.3. Model's Parameter Adjustment Process

Taking Transformer-NDC as an example, the parameter adjustment process is shown in this section. Taking the test on the V-Vfb02d dataset as an example, the four rightmost

columns of Table 5 list the model's corresponding performance on the four evaluation metrics. The parameter values listed for the baseline model are the final parameter values selected.

Table 5. Parameter adjustment process of Transformer-NDC (parameter values not listed are the same as the corresponding parameter values of the baseline model).

	N	d_{model}	d_{ff}	h	$Dropout$	max	min	μ	σ
baseline	6	512	32	8	0	0.36	0.03	0.09	0.07
A				4		0.42	0.03	0.12	0.06
				16		0.33	0.03	0.09	0.07
B	4					0.38	0.03	0.12	0.06
	8					0.42	0.03	0.09	0.08
C		1024				0.51	0.03	0.09	0.10
		256				0.51	0.03	0.09	0.10
D			16			0.43	0.03	0.11	0.06
			64			0.85	0.03	0.25	0.20
E					0.1	1.13	0.04	0.14	0.20
					0.2	0.94	0.04	0.20	0.18

The five sections A–E listed in Table 5 test the effect of changing the corresponding parameters on the results. A changes h with respect to the baseline, and it can be seen that the mean and maximum error increase when h decreases to 4, indicating a decrease in the accuracy and reliability of the model. When h increases to 16, although the maximum error is slightly reduced, the number of model parameters increases and the model's computational efficiency decreases, so this parameter value is not chosen. B changes N with respect to the baseline, and the maximum error increases when N decreases to 4 and increases to 8, indicating a decrease in the model's reliability, and the mean error increases when N decreases to 4, indicating a decrease in the model's accuracy. C changes d_{model} relative to the baseline, and when d_{model} decreases to 256 and increases to 1024, the maximum error increases significantly, indicating a decrease in the model's reliability, and the standard deviation increases showing that the model stability decreases. D changes d_{ff} with respect to the baseline, and it can be seen that the mean and maximum error increase when d_{ff} decreases to 16. When d_{ff} increases to 64, there is a large increase in the maximum value, mean, and standard deviation of error, indicating that the reliability, accuracy, and stability of the model decrease significantly. E changes $Dropout$ relative to the baseline, and when $Dropout$ increases to 0.1 and 0.2, the maximum value, mean, and standard deviation of error increase significantly, indicating that the reliability, accuracy, and stability of the model decrease significantly.

5. Results and Discussion

5.1. Tests on the Dataset

In this section, the proposed wheel odometry error prediction model based on Transformer was compared with the LSTM model proposed in [21] and the WhONet model proposed in [22] to evaluate the model performance using the metrics described in Section 4.2.

5.1.1. Tests in Challenging Driving Conditions

Motorway

The test results of the motorway scenario are displayed in Table 6 and Figure 3. It can be seen that the four metrics of Transformer-DC and Transformer-NDC were equal and were lower than those of LSTM and WhONet. Compared with LSTM and WhONet, both Transformer models showed 76% and 52% error reductions in the max metric, 83% and 69% error reductions in the μ metric, and 92% and 85% error reductions in the min metric, respectively, demonstrating better positioning reliability and positioning accuracy. All the

models had a low standard deviation below 0.1 m and had good stability in this driving condition. On the whole, all four models show good performance in this driving condition. Compared with the LSTM and WhONet, the proposed model significantly improves the average and maximum prediction accuracy.

Table 6. Test results for the motorway scenario.

IO-VNB Dataset	Performance Metrics	Model Error (m)			
		CRSE			
		LSTM	WhONet	Transformer-NDC	Transformer-DC
V-Vw12	max	0.68	0.33	0.16	0.16
	min	0.25	0.13	0.02	0.02
	μ	0.46	0.26	0.08	0.08
	σ	0.08	0.04	0.03	0.03

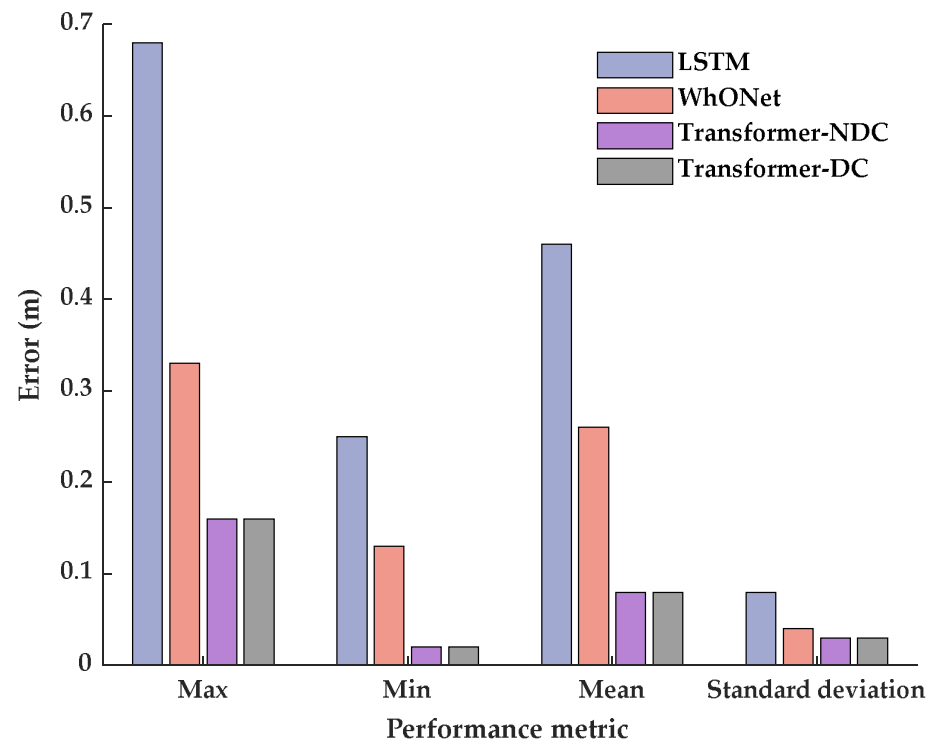


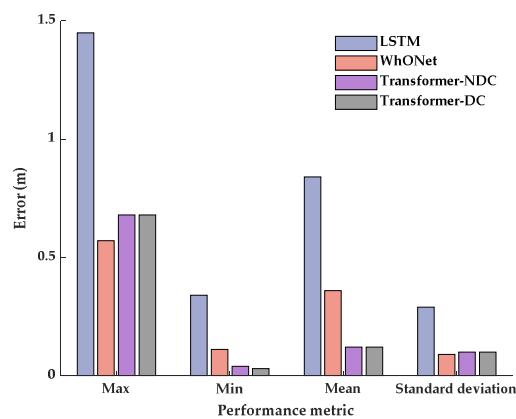
Figure 3. Test results for the motorway scenario.

Sharp Cornering and Successive Left and Right Turns

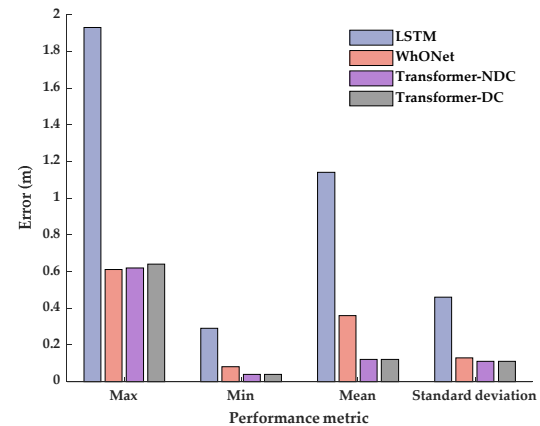
Table 7 and Figure 4 illustrate the test results for the driving condition with sharp cornering and successive left and right turns. The four metrics of Transformer-DC and Transformer-NDC on all three data subsets were approximately equal. Compared to LSTM, the two proposed models show considerable performance improvements, specifically up to 76%, 93%, 90%, and 80% on the four metrics. Compared to WhONet, the performances on the max and standard deviation metrics were identical; the minimum and mean errors were reduced by up to 73% and 68%, respectively. In summary, the proposed models improve the reliability, highest possible accuracy, average accuracy, and standard deviation compared to LSTM, and they improve the highest possible and average accuracy compared to WhONet under this driving condition.

Table 7. Test results for the driving condition with sharp cornering and successive left and right turns.

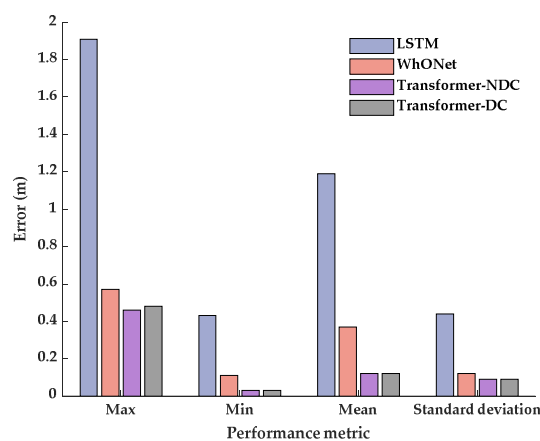
IO-VNB Dataset	Performance Metrics	Model Error (m)			
		CRSE			
		LSTM	WhONet	Transformer-NDC	Transformer-DC
V-Vw6	max	1.45	0.57	0.68	0.68
	min	0.34	0.11	0.04	0.03
	μ	0.84	0.36	0.12	0.12
	σ	0.29	0.09	0.10	0.10
V-Vw7	max	1.93	0.61	0.62	0.64
	min	0.29	0.08	0.04	0.04
	μ	1.14	0.36	0.12	0.12
	σ	0.46	0.13	0.11	0.11
V-Vw8	max	1.91	0.57	0.46	0.48
	min	0.43	0.11	0.03	0.03
	μ	1.19	0.37	0.12	0.12
	σ	0.44	0.12	0.09	0.09



(a)



(b)



(c)

Figure 4. Test results for the driving condition with sharp cornering and successive left and right turns. (a) V-Vw6; (b) V-Vw7; (c) V-Vw8.

Wet Road

The test results for the wet road scenario are shown in Table 8 and Figure 5. Both Transformer-DC and Transformer-NDC performed approximately the same on the three data subsets. Compared to LSTM, the two proposed models show considerable perfor-

mance improvements, specifically up to 75%, 90%, 90%, and 85% on the four metrics. Compared to WhONet, although there was only a slight improvement in the model's performance in the four metrics on the V-Vtb8 and V-Vtb11 data subsets, the performances on the max, mean, and standard deviation metrics were improved dramatically by 75%, 67%, 84%, and 78%, respectively, on the V-Vtb13 data subset.

Table 8. Test results for the wet road scenario.

IO-VNB Dataset	Performance Metrics	Model Error (m)			
		CRSE			
		LSTM	WhONet	Transformer-NDC	Transformer-DC
V-Vtb8	max	0.48	0.19	0.14	0.14
	min	0.23	0.08	0.07	0.07
	μ	0.33	0.14	0.10	0.09
	σ	0.04	0.02	0.01	0.01
V-Vtb11	max	0.37	0.14	0.12	0.13
	min	0.24	0.05	0.05	0.05
	μ	0.31	0.11	0.09	0.09
	σ	0.03	0.02	0.02	0.02
V-Vtb13	max	2.33	2.33	0.59	0.58
	min	0.20	0.06	0.02	0.02
	μ	1.24	0.75	0.12	0.12
	σ	0.73	0.49	0.11	0.11

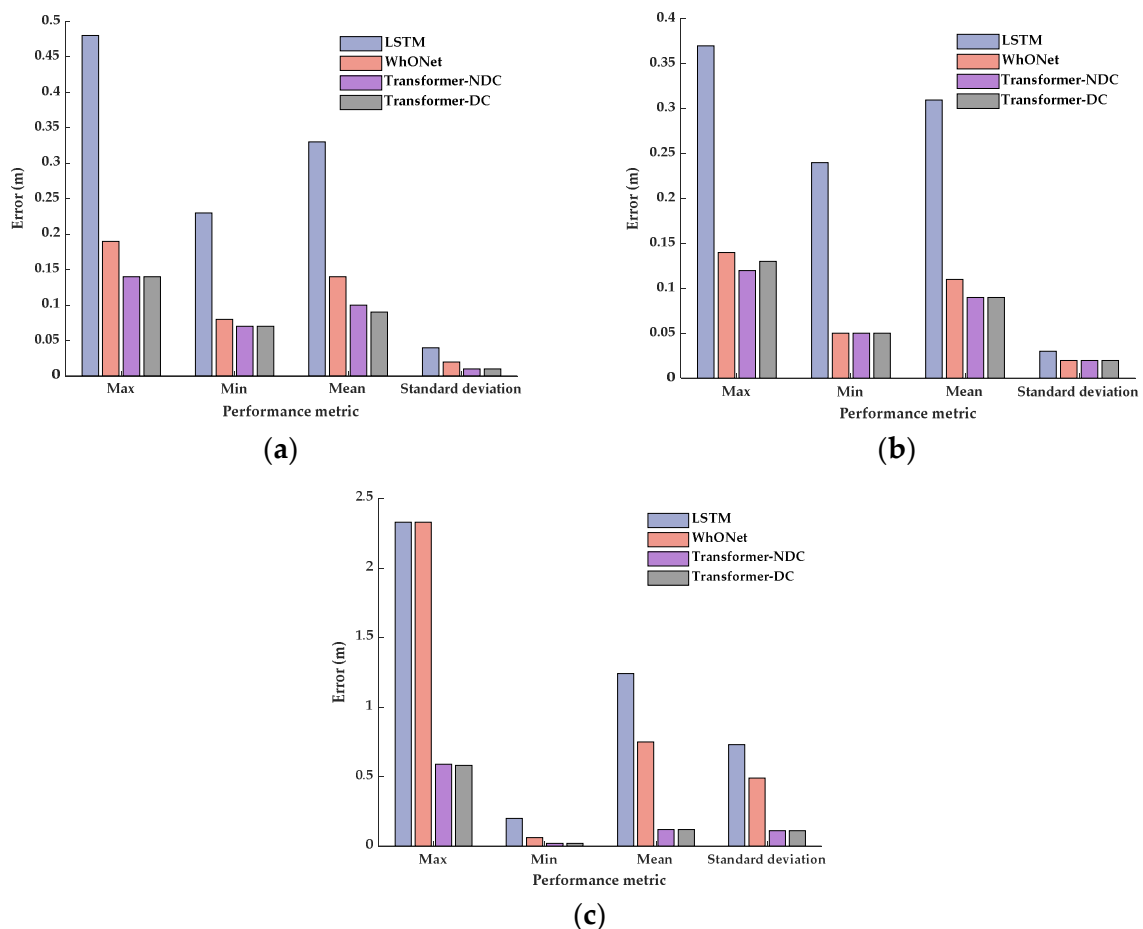


Figure 5. Test results for the wet road scenario. (a) V-Vtb8; (b) V-Vtb11; (c) V-Vtb13.

Quick Changes in Vehicle's Acceleration

Table 9 and Figure 6 demonstrate the test results for the driving condition with quick changes in the vehicle's acceleration. Both Transformer-DC and Transformer-NDC had almost the same performance on the two data subsets. On the V-Vfb02e dataset, compared to LSTM, the two proposed models show 57%, 94%, 86%, and 72% improvements on the four metrics; compared to WhONet, the two proposed models show 55%, 82%, 82%, and 78% improvements on the four metrics. On the V-Vta12 dataset, compared to LSTM, the two proposed models show 54%, 92%, 83%, and 64% improvements on the four metrics; compared to WhONet, the two proposed models show 54%, 64%, 74%, and 59% improvements on the four metrics. In summary, the proposed models significantly improved the reliability, highest possible accuracy, average accuracy, and stability compared to LSTM and WhONet in this driving condition.

Table 9. Test results for the driving condition with quick changes in vehicle acceleration.

IO-VNB Dataset	Performance Metrics	Model Error (m)			
		CRSE			
		LSTM	WhONet	Transformer-NDC	Transformer-DC
V-Vfb02e	max	2.95	2.82	1.27	1.28
	min	0.54	0.17	0.03	0.03
	μ	1.43	1.14	0.20	0.20
	σ	0.65	0.81	0.18	0.18
V-Vta12	max	2.30	2.27	1.05	1.05
	min	0.65	0.14	0.05	0.05
	μ	1.37	0.89	0.23	0.23
	σ	0.55	0.49	0.20	0.20

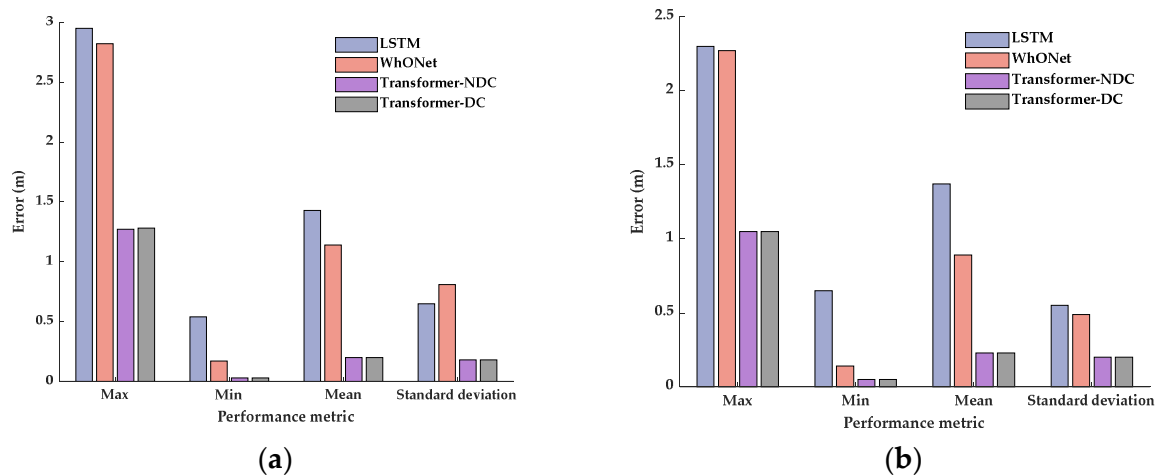


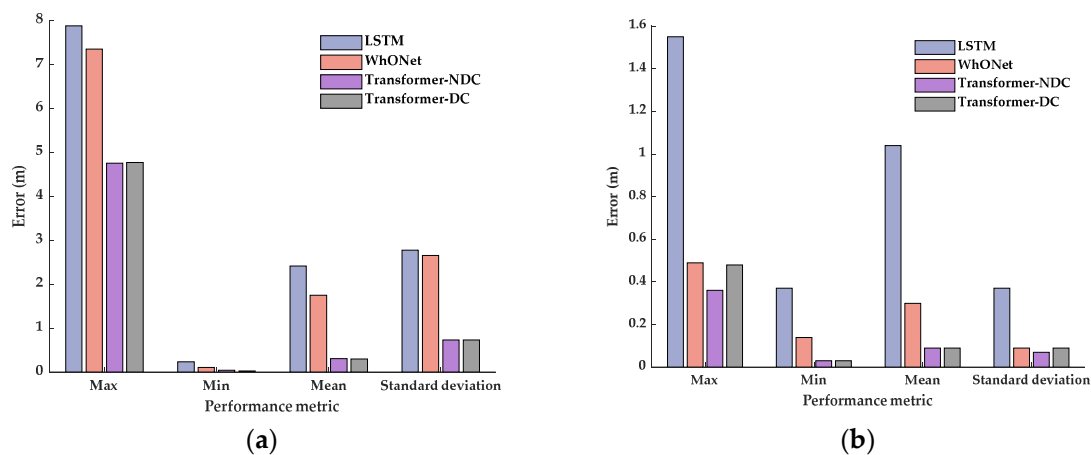
Figure 6. Test results for the driving condition with quick changes in vehicle acceleration. (a) V-Vfb02e; (b) V-Vta12.

Roundabout

Table 10 and Figure 7 list the test results in the roundabout scenario. Both Transformer-DC and Transformer-NDC had approximately the same performance on the two data subsets. The proposed models outperform LSTM. Specifically, the models can enhance the performance by up to 77%, 92%, 91%, and 81% on all four metrics. Compared to WhONet, the proposed models show up to 35%, 79%, 83%, and 73% improvements on all four metrics. In summary, the proposed models significantly improved the reliability, highest possible accuracy, average accuracy, and stability compared to LSTM and WhONet in this driving condition.

Table 10. Test results for the roundabout scenario.

IO-VNB Dataset	Performance Metrics	Model Error (m)			
		CRSE			
		LSTM	WhONet	Transformer-NDC	Transformer-DC
V-Vta11	max	7.88	7.35	4.76	4.77
	min	0.24	0.11	0.04	0.03
	μ	2.42	1.75	0.31	0.30
	σ	2.78	2.66	0.73	0.73
V-Vfb02d	max	1.55	0.49	0.36	0.48
	min	0.37	0.14	0.03	0.03
	μ	1.04	0.30	0.09	0.09
	σ	0.37	0.09	0.07	0.09

**Figure 7.** Test results for the roundabout scenario. (a) V-Vta11; (b) V-Vfb02d.

Hard Brake

Test results for the hard brake driving condition are displayed in Table 11 and Figure 8. The results show that Transformer-DC and Transformer-NDC exhibited almost identical performance on the two data subsets. The four metrics of the proposed models are reduced by up to 96%, 93%, 97%, and 95%, respectively, compared to LSTM. Compared to WhONet, the two proposed models show up to 84%, 95%, 92%, and 77% improvements in the four metrics. It is thus proven that the accuracy, stability, and reliability of the transformer-based model are significantly improved compared with LSTM and WhONet under this challenging driving condition.

Table 11. Test results for the hard brake driving condition.

IO-VNB Dataset	Performance Metrics	Model Error (m)			
		CRSE			
		LSTM	WhONet	Transformer-NDC	Transformer-DC
V-Vw16b	max	2.93	0.88	0.36	0.36
	min	0.10	0.10	0.01	0.01
	μ	1.49	0.36	0.06	0.06
	σ	0.95	0.22	0.05	0.05
V-Vw17	max	3.51	0.87	0.14	0.15
	min	0.41	0.61	0.03	0.03
	μ	2.13	0.78	0.06	0.06
	σ	0.01	0.01	0.03	0.03

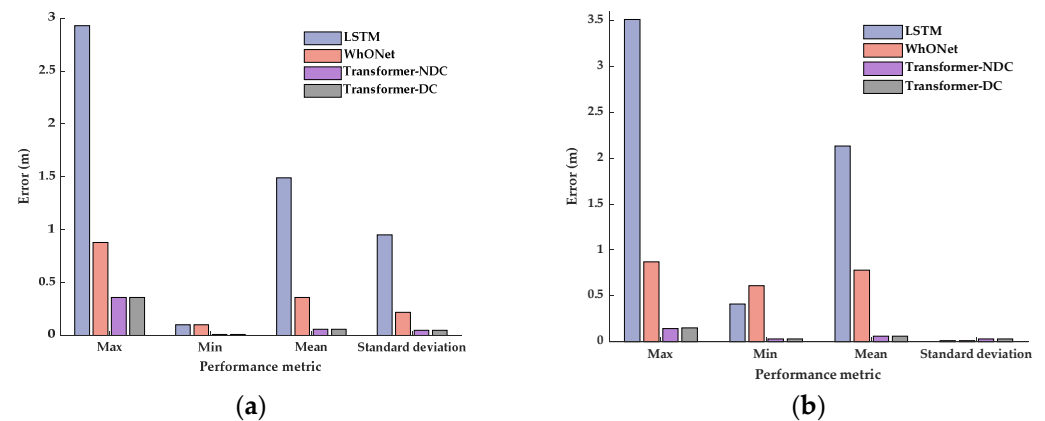


Figure 8. Test results for the hard brake driving condition. (a) V-Vw16b; (b) V-Vw17.

In summary, the proposed model has higher accuracy, stability, and reliability under various challenging driving conditions where it is difficult to measure odometry signals accurately compared with LSTM and WhONet. Meanwhile, the Transformer models with and without the driving condition characteristics can obtain approximately the same performance under these challenging driving conditions by adjusting the parameters.

5.1.2. Tests in Longer GNSS Outage Driving Conditions

Since WhONet has better performance than LSTM, as shown in Tables 6–11, in this section, the proposed model was compared further with WhONet to evaluate their performance over four longer-term GNSS outages of 30 s, 60 s, 120 s, and 180 s using the corresponding data subsets in Table 4, and the evaluation results are shown in Tables 12–15.

30 s GNSS Outage

Test results on the 30 s GNSS outage driving condition are displayed in Table 12 and Figure 9. The results show that Transformer-NDC demonstrated significantly better performance than the WhONet model. Specifically, the errors on the four metrics were reduced by up to 50%, 81%, 66%, and 60%, respectively, demonstrating that the Transformer model has higher accuracy and better stability and reliability than the WhONet model in this driving condition. In addition, the performance of Transformer-DC on the max metric is improved compared to Transformer-NDC, which proves that considering the driving condition characteristics boosts the model's reliability.

Table 12. Results under the 30 s GNSS outage driving condition.

IO-VNB Dataset	Performance Metrics	Model Error (m)		
		CRSE		
		WhONet	Transformer-NDC	Transformer-DC
V-Vtb3	max	1.35	1.35	0.93
	min	0.11	0.05	0.04
	μ	0.67	0.25	0.24
	σ	0.20	0.17	0.14
V-Vfb02a	max	8.57	4.31	3.65
	min	0.28	0.07	0.08
	μ	1.01	0.34	0.34
	σ	0.83	0.33	0.44
V-Vfb02b	max	1.70	1.31	1.27
	min	0.37	0.07	0.06
	μ	1.04	0.36	0.35
	σ	0.30	0.22	0.22

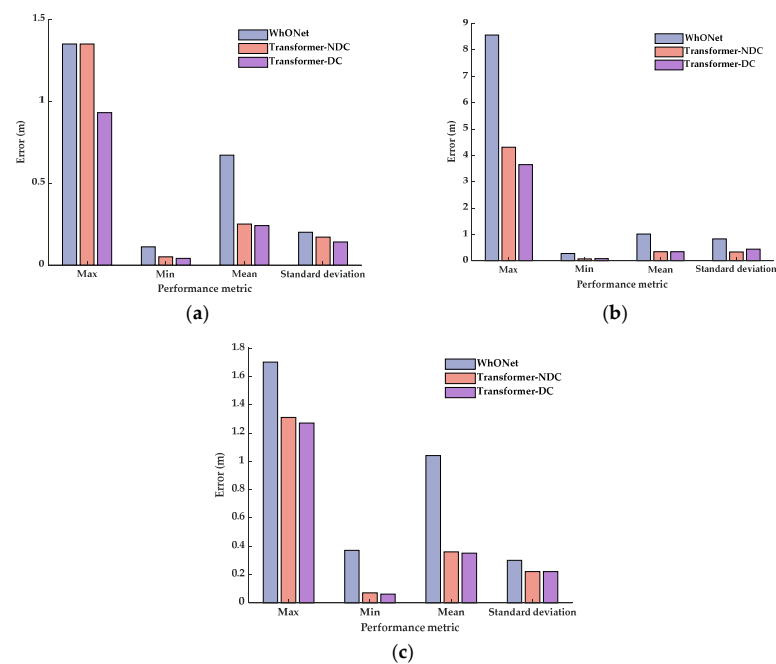


Figure 9. Test results under the 30 s GNSS outage driving condition. (a) V-Vtb3; (b) V-Vfb02a; (c) V-Vfb02b.

60 s GNSS Outage

Table 13 and Figure 10 illustrate the test results under the 60 s GNSS outage driving condition. It can be seen that compared with the WhONet model, Transformer-NDC improved the performance enormously. Specifically, the errors were reduced on the four metrics greatly by up to 25%, 82%, 66%, and 42% on the V-Vfb02a and V-Vfb02b data subsets; on the V-Vtb3 data subset, although the errors on the max and μ metrics were slightly increased, the mean error was decreased tremendously by 62%. Additionally, Transformer-DC improved the performance on the four metrics by up to 32%, 82%, 67%, and 23% on the three datasets compared with the WhONet model, and it reduced the maximum error by up to 33% compared to Transformer-NDC. In summary, the proposed models have higher accuracy and better stability and reliability in error prediction compared to WhONet, and considering the driving condition characteristics can enhance the model's reliability.

Table 13. Results under the 60 s GNSS outage driving condition.

IO-VNB Dataset	Performance Metrics	Model Error (m)		
		CRSE		
		WhONet	Transformer-NDC	Transformer-DC
V-Vtb3	max	2.29	2.54	1.71
	min	0.29	0.16	0.12
	μ	1.31	0.50	0.47
	σ	0.25	0.34	0.27
V-Vfb02a	max	9.62	7.24	6.55
	min	0.87	0.19	0.19
	μ	2.02	0.68	0.68
	σ	1.11	0.64	0.86
V-Vfb02b	max	2.94	2.41	2.40
	min	1.10	0.20	0.20
	μ	2.08	0.71	0.69
	σ	0.48	0.38	0.38

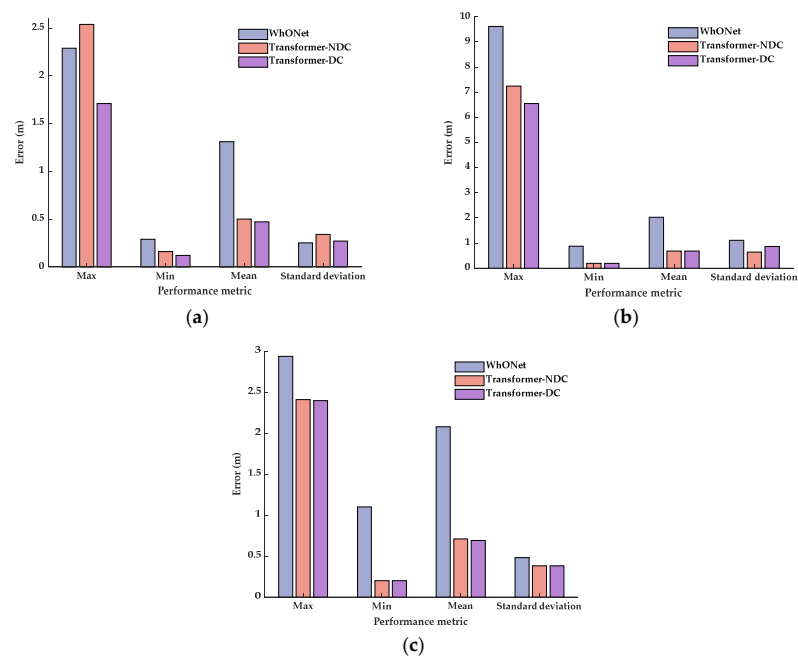


Figure 10. Test results under the 60 s GNSS outage driving condition. (a) V-Vtb3; (b) V-Vfb02a; (c) V-Vfb02b.

120 s GNSS Outage

The test results under the 120 s GNSS outage driving condition are listed in Table 14 and Figure 11. Compared with the WhONet model, except for the slight increment on the σ metric on the V-Vtb3 data subset, both Transformer models improved the performance on the four metrics. Specifically, Transformer-NDC reduced the errors on the four metrics by up to 50%, 81%, 67%, and 35%, respectively, and the Transformer-DC reduced the errors on the four metrics by up to 50%, 84%, 71%, and 35%, respectively, proving that the proposed models have higher accuracy and better stability and reliability compared with the WhONet model in this driving condition. In addition, the values on the four metrics of Transformer-DC are reduced compared to Transformer-NDC, demonstrating that considering the driving condition characteristics improves the model's accuracy, stability, and reliability.

Table 14. Results under the 120 s GNSS outage driving condition.

IO-VNB Dataset	Performance Metrics	Model Error (m)		
		CRSE		
		WhONet	Transformer-NDC	Transformer-DC
V-Vtb3	max	4.01	3.64	2.80
	min	1.75	0.34	0.30
	μ	2.62	0.99	0.95
	σ	0.19	0.61	0.49
V-Vfb02a	max	11.75	7.56	7.30
	min	2.05	0.49	0.38
	μ	4.07	1.35	1.19
	σ	1.41	1.16	1.08
V-Vfb02b	max	5.76	2.88	2.86
	min	2.52	0.48	0.41
	μ	4.15	1.42	1.38
	σ	0.88	0.57	0.57

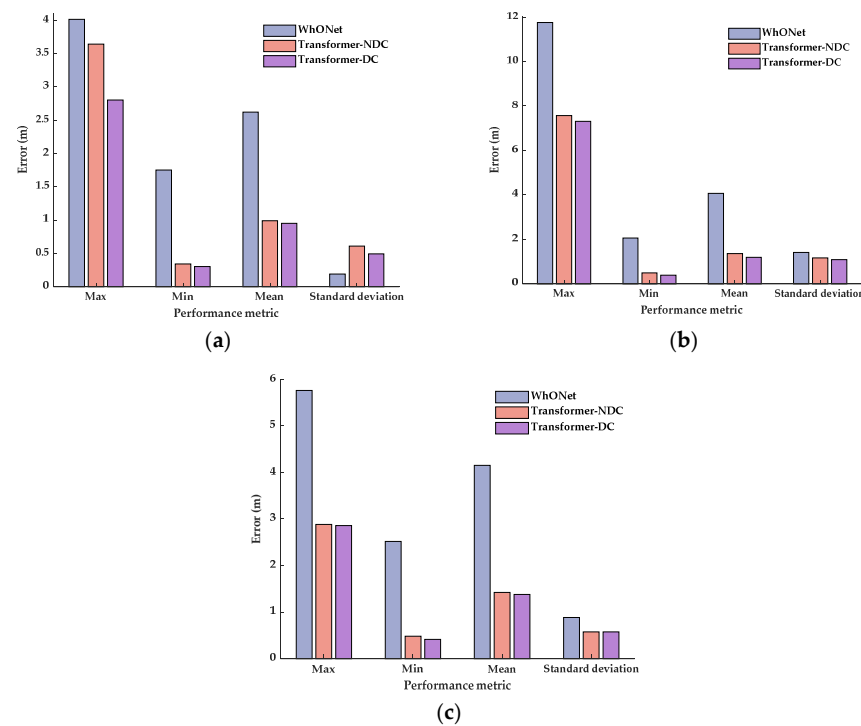


Figure 11. Test results under the 120 s GNSS outage driving condition. (a) V-Vtb3; (b) V-Vfb02a; (c) V-Vfb02b.

180 s GNSS Outage

Table 15 and Figure 12 show the test results under the 180 s GNSS outage driving condition. Compared with WhONet, Transformer-NDC improved the performance on the max, min, and μ metrics by up to 51%, 79%, and 66%, respectively, exhibiting that the Transformer model has higher accuracy and better reliability than the WhONet model in this driving condition. By considering the driving condition characteristics, Transformer-DC reduced the error on the four metrics on the V-Vtb3 and V-Vfb02a data subsets and the error on the max and μ metrics on the V-Vfb02b data subset, which proves that considering the driving condition characteristics improves the model's accuracy, stability, and reliability.

Table 15. Results under the 180 s GNSS outage driving condition.

IO-VNB Dataset	Performance Metrics	Model Error (m)		
		CRSE		
		WhONet	Transformer-NDC	Transformer-DC
V-Vtb3	max	4.97	4.31	3.83
	min	2.81	0.64	0.51
	μ	3.93	1.44	1.39
	σ	0.21	0.77	0.66
V-Vfb02a	max	13.02	9.90	8.54
	min	3.67	0.76	0.64
	μ	6.08	2.03	1.80
	σ	1.55	1.68	1.62
V-Vfb02b	max	8.09	3.99	3.68
	min	4.04	0.83	0.89
	μ	6.23	2.13	2.06
	σ	0.77	0.72	0.73

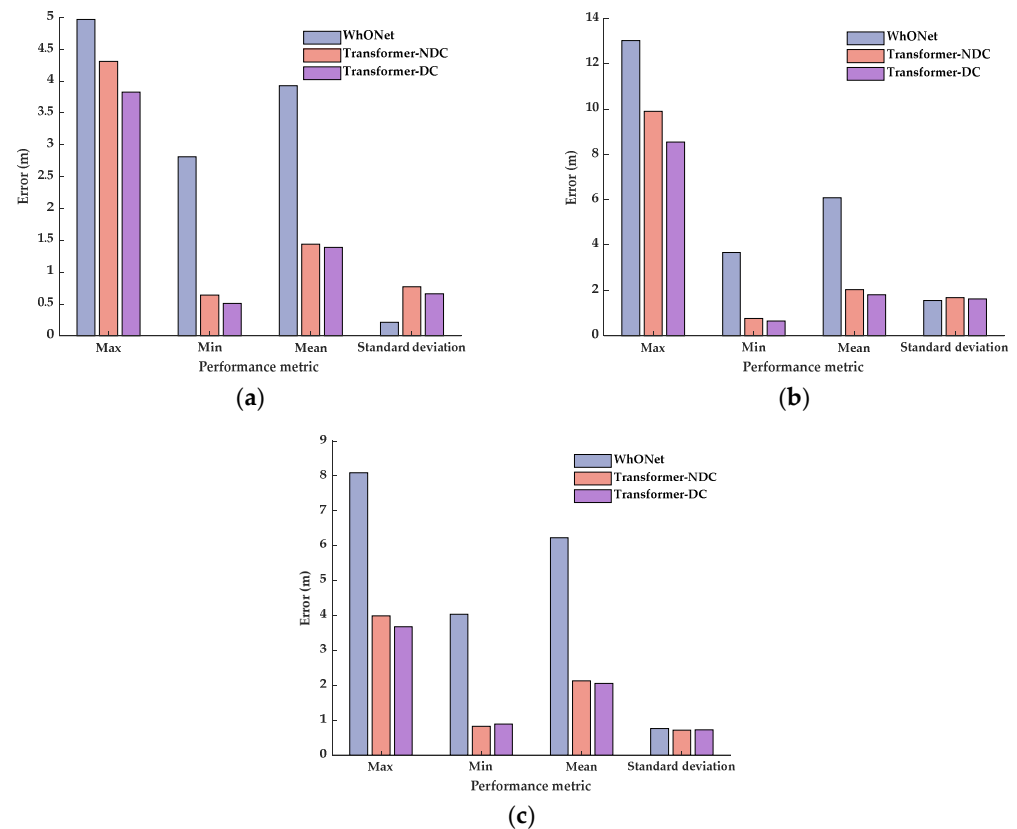


Figure 12. Test results under the 180 s GNSS outage driving condition. (a) V-Vtb3; (b) V-Vfb02a; (c) V-Vfb02b.

In conclusion, the proposed Transformer models can accurately predict odometry errors under different longer GNSS outage driving conditions and have higher accuracy, stability, and reliability than the WhONet model. Additionally, Transformer-DC, which considers the driving condition characteristics, can better adapt to different driving conditions than Transformer-NDC, and this improves the model's overall performance.

From the tests in Section 5.1, it can be seen that whether considering the driving condition characteristics or not, the proposed Transformer model can predict odometry errors with higher accuracy, stability, and reliability than LSTM and WhONet under various driving conditions including challenging and longer GNSS outage driving conditions, which may cause the wheel odometry output to become unpredictable and to accumulate errors. The results showcase the superiority of the model and its great potential in terms of improving positioning performance. Although the Transformer models with and without the driving condition characteristics can perform roughly equally well under challenging driving conditions, the Transformer model's overall performance can be improved under longer GNSS outage driving conditions by considering the driving condition characteristics. It can be inferred that considering the driving condition characteristics can improve the reliability and stability of long-term prediction by enhancing the adaptability to various driving conditions, while this is not so evident in short-term prediction.

5.2. Tests on the Experimental Vehicle

In this section, Transformer-NDC is taken as an example, and the model's generalization capability and improved dead reckoning performance based on wheel odometry combined with the model was verified on the experimental vehicle, which is shown in Figure 13. RTK equipment was loaded on the vehicle to provide centimeter-level-accuracy GPS coordinates that can be used as true values to verify the localization accuracy.

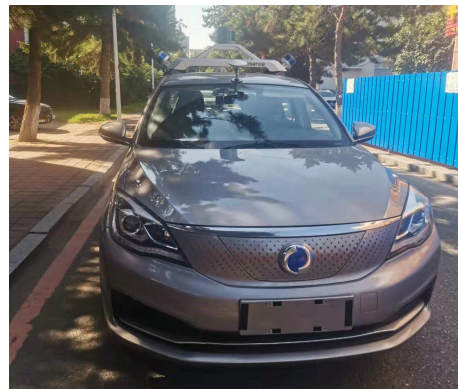


Figure 13. Experimental vehicle.

To verify the generalization capability of the model, the model trained on the public dataset was directly applied to this experimental vehicle. The experiment was conducted in a region of Changchun, and the model's performance on the real vehicle data was compared with that on the V-Vta8 dataset under similar driving conditions, as is shown in Table 16. It can be seen that the corresponding maximum and mean metrics of CRSE increase after the model transfers, but not much, and there is a slight decrease in the minimum and standard deviation, so the model still has good model prediction function after the model is transferred.

Table 16. Error comparison of the experimental vehicle and the corresponding data subset.

Performance Metrics	Model Error (m) CRSE	
	V-Vta8	Experimental Vehicle
max	1.02	1.46
min	0.05	0.02
μ	0.33	0.60
σ	0.21	0.20

The vehicle driving route during the experiment is shown in Figure 14, and the RTK-GPS coordinates, the positioning coordinates obtained by dead reckoning alone, and the positioning coordinates obtained by dead reckoning combined with the odometry error prediction model were recorded. It can be seen from the enlarged parts in the figure that the green curve and the blue curve overlap approximately in most cases, which is due to the fact that the vehicle driving along the lane is the most dominant driving behavior, and the odometry output error mainly affects the longitudinal positioning error of the vehicle, which cannot be directly reflected in this figure.

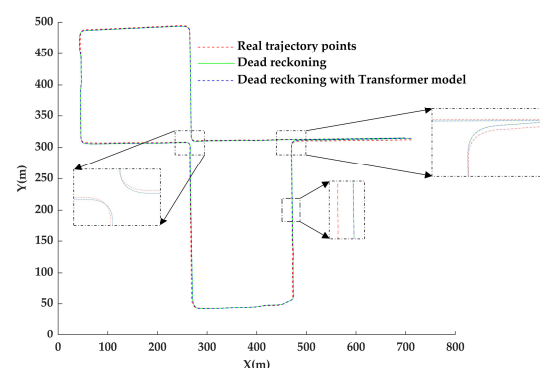


Figure 14. Localization trajectory comparison.

Figure 15 compares the error of the original dead reckoning with the error of the dead reckoning after compensation using the error prediction model proposed in this paper. In Figure 15b,c, the error after compensation using the error prediction model is significantly lower compared to the original dead reckoning, and in Figure 5a, although the error of the original dead reckoning is relatively lower than the error after combining with the prediction model compensation between the first 1000 points and the 4500th to 5000th points, which is caused by the deep learning-based model's uncertainty, both are small and less than 0.5 m, and the error after combining with the prediction model compensation is lower than the original dead reckoning at other points most of the time. It is thus proven that the proposed error prediction model can effectively compensate for the output error of the wheel odometry in the x and y directions and overall, thus improving the accuracy of the dead reckoning and the performance of the positioning system.

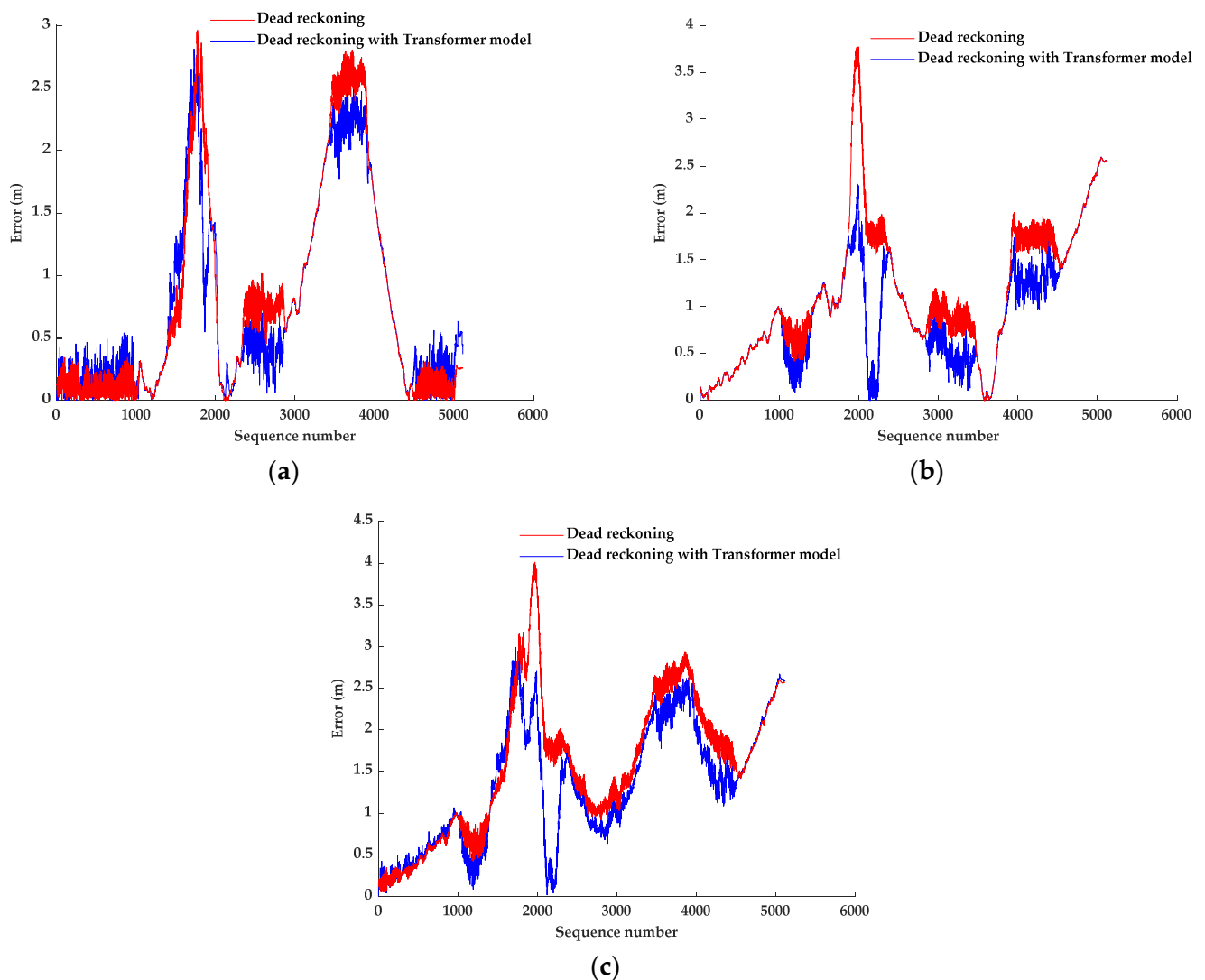


Figure 15. Error of the dead reckoning and the dead reckoning with the Transformer model relative to the real trajectory. (a) Error in the x coordinate; (b) Error in the y coordinate; (c) The total error.

Table 17 compares the performance indexes of the original dead reckoning error and the error after error compensation with the model. It can be seen that the maximum value of the positioning error is significantly reduced, and the mean and standard deviation of the error are also reduced after the error is compensated for by the error prediction model,

indicating that the proposed error prediction model can effectively predict the odometry error and improve the reliability, stability, and accuracy of odometry positioning.

Table 17. Comparison between original dead reckoning and dead reckoning with Transformer model.

Performance Metrics (m)	Dead Reckoning	Dead Reckoning with Transformer Model
max	4.01	2.99
min	0.06	0
μ	1.58	1.34
σ	0.84	0.74

In summary, the model performance when transferring the model trained on the public dataset to the real vehicle application declined compared to the performance on the dataset, but it is still effective at predicting the odometry error and improving the accuracy and stability of localization. Predictably, if the model is trained using the data collected from the real vehicle and then applied to the real vehicle, the accuracy will be higher, but the workload simultaneously increases. This experiment validates the generalization capability of the model and provides a basis for the application and generalization of the model to multiple vehicle types.

6. Conclusions

In this study, a Transformer-based wheel odometry error prediction model is established to accurately predict the accumulated and changing travel distance errors during vehicle driving so as to improve the positioning performance of wheel odometry. The study and its findings can be summarized as follows:

1. Two different Transformer error prediction models are designed. Both models contain four-wheel wheel speed as the input. One considers the driving condition features, which contain multiple features describing different road types, road conditions, and driving operations, as the input, and the other does not.
2. The performance of the proposed model is evaluated and compared with LSTM and WhONet under challenging driving conditions and longer-term GNSS outage driving conditions. The results demonstrate that the proposed model has higher accuracy, stability, and reliability under various challenging driving conditions and longer-term GNSS outage driving conditions. Specifically, the model performs up to 96%, 94%, 97%, and 95% better than LSTM and up to 84%, 95%, 92%, and 78% better than WhONet on the max, min, mean, and standard deviation metrics under various challenging conditions. In addition, Transformer-NDC performs up to 51%, 84%, 71%, and 60% better than WhONet on the max, min, mean, and standard deviation metrics under longer-term GNSS outage driving conditions. Although the Transformer models with and without the driving condition characteristics achieve approximately the same performance under challenging driving conditions, Transformer-DC, which considers the driving condition characteristics, can better adapt to driving conditions with longer-term GNSS outage than Transformer-NDC and improve the model's overall performance.
3. Tests on an experimental vehicle are conducted to verify the model's generalization capability and the improved dead reckoning positioning performance by combining the model. In particular, a model trained on a public dataset is directly applied to the experimental vehicle, and the model's performance on the experimental vehicle is compared with that on a data subset with the similar driving condition. The results show that although the model's performance declines slightly, the mean and standard deviation of the model's errors are only 0.6 m and 0.2 m, which demonstrates that the model is still effective at predicting the odometry error and improving the accuracy and stability of localization.

This study explored the possibility of applying the Transformer model to wheel odometry, providing a new solution for the application of deep learning to localization. Although the test on the experimental vehicle validated the model's generalization capability and provides a basis for the application and generalization of the model to multiple vehicle types, in future work, more study is needed to further develop and explore related techniques to enhance the generalization of the Transformer model to multiple vehicle types and to improve the model accuracy after transferring so that the efficiency of model utilization can be improved, in addition, the related techniques to improve the performance of the Transformer model and the use of other deep-learning models to localization can be studied.

Author Contributions: Conceptualization, H.D.; data curation, K.H.; investigation, K.H.; methodology, K.H.; project administration, N.X. and K.G.; resources, H.D. and N.X.; software, K.H.; supervision, K.G.; validation, K.H.; writing—original draft, K.H.; writing—review and editing, K.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Jilin Province major science and technology special projects of China (grant number 20220301033GX) and the National Natural Science Foundation of China (grant number U1864206).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

1. Yurtsever, E.; Lambert, J.; Carballo, A.; Takeda, K. A survey of autonomous driving: Common practices and emerging technologies. *IEEE Access* **2020**, *8*, 58443–58469. [\[CrossRef\]](#)
2. Badue, C.; Guidolini, R.; Carneiro, R.V.; Azevedo, P.; Cardoso, V.B.; Forechi, A.; Jesus, L.; Berriel, R.; Paixao, T.M.; Mutz, F.; et al. Self-driving cars: A survey. *Expert Syst. Appl.* **2021**, *165*, 113816. [\[CrossRef\]](#)
3. Viana, K.; Zubizarreta, A.; Diez, M. A Reconfigurable Framework for Vehicle Localization in Urban Areas. *Sensors* **2022**, *22*, 2595. [\[CrossRef\]](#) [\[PubMed\]](#)
4. Zhang, Y.; Wang, L.; Jiang, X.; Zeng, Y.; Dai, Y. An efficient LiDAR-based localization method for self-driving cars in dynamic environments. *Robotica* **2022**, *40*, 38–55. [\[CrossRef\]](#)
5. Gyagenda, N.; Hatilima, J.V.; Roth, H.; Zhmud, V. A review of GNSS-independent UAV navigation techniques. *Robot. Auton. Syst.* **2022**, *152*, 104069. [\[CrossRef\]](#)
6. Zhang, J.; Wen, W.; Huang, F.; Wang, Y.; Chen, X.; Hsu, L.-T. GNSS-RTK Adaptively Integrated with LiDAR/IMU Odometry for Continuously Global Positioning in Urban Canyons. *Appl. Sci.* **2022**, *12*, 5193. [\[CrossRef\]](#)
7. Lyu, P.; Bai, S.; Lai, J.; Wang, B.; Sun, X.; Huang, K. Optimal time difference-based TDCP-GPS/IMU navigation using graph optimization. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 9514710. [\[CrossRef\]](#)
8. Thrun, S.; Montemerlo, M.; Dahlkamp, H.; Stavens, D.; Aron, A.; Diebel, J.; Fong, P.; Gale, J.; Halpenny, M.; Hoffmann, G.; et al. Stanley: The robot that won the DARPA Grand Challenge. *J. Field Robot.* **2006**, *23*, 661–692. [\[CrossRef\]](#)
9. Funk, N.; Alatur, N.; Deuber, R. *Autonomous Electric Race Car Design*; EVS30 Symposium: Stuttgart, Germany, 2017.
10. Schwesinger, U.; Bürki, M.; Timpner, J.; Rottmann, S.; Wolf, L.; Paz, L.M.; Grimm, H.; Posner, I.; Newman, P.; Häne, C.; et al. Automated valet parking and charging for e-mobility. In Proceedings of the 2016 IEEE Intelligent Vehicles Symposium (IV), Gothenburg, Sweden, 19–22 June 2016; pp. 157–164.
11. Brunker, A.; Wohlgemuth, T.; Frey, M.; Gauterin, F. Odometry 2.0: A slip-adaptive EIF-based four-wheel-odometry model for parking. *IEEE Trans. Intell. Veh.* **2018**, *4*, 114–126. [\[CrossRef\]](#)
12. Fazekas, M.; Németh, B.; Gáspár, P. Iterative parameter identification method of a vehicle odometry model. *IFAC-PapersOnLine* **2019**, *52*, 199–204. [\[CrossRef\]](#)
13. Fazekas, M.; Németh, B.; Gáspár, P.; Sename, O. Vehicle odometry model identification considering dynamic load transfers. In Proceedings of the 2020 28th Mediterranean Conference on Control and Automation (MED), Saint-Raphaël, France, 15–18 September 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 19–24.
14. Fazekas, M.; Gáspár, P.; Németh, B. Velocity Estimation via Wheel Circumference Identification. *Period. Polytech. Transp. Eng.* **2021**, *49*, 250–260. [\[CrossRef\]](#)
15. Fazekas, M.; Gáspár, P.; Németh, B. Calibration and improvement of an odometry model with dynamic wheel and lateral dynamics integration. *Sensors* **2021**, *21*, 337. [\[CrossRef\]](#) [\[PubMed\]](#)

16. Welte, A.; Xu, P.; Bonnifait, P. Four-wheeled dead-reckoning model calibration using RTS smoothing. In Proceedings of the International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 20–24 May 2019; pp. 312–318.
17. Malleswaran, M.; Vaidehi, V.; Manjula, S.; Deborah, S.A. Performance comparison of HONNs and FFNNs in GPS and INS integration for vehicular navigation. In Proceedings of the 2011 International Conference on Recent Trends in Information Technology (ICRTIT), Chennai, India, 3–5 June 2011; pp. 223–228.
18. Malleswaran, M.; Vaidehi, V.; Saravanaselvan, A.; Mohankumar, M. Performance Analysis of Various Artificial Intelligent Neural Networks For Gps/Ins Integration. *Appl. Artif. Intell.* **2013**, *27*, 367–407. [[CrossRef](#)]
19. Dai, H.-F.; Bian, H.-W.; Wang, R.-Y.; Ma, H. An INS/GNSS integrated navigation in GNSS denied environment using recurrent neural network. *Def. Technol.* **2019**, *16*, 334–340. [[CrossRef](#)]
20. Fang, W.; Jiang, J.; Lu, S.; Gong, Y.; Tao, Y.; Tang, Y.; Yan, P.; Luo, H.; Liu, J. A LSTM Algorithm Estimating Pseudo Measurements for Aiding INS during GNSS Signal Outages. *Remote Sens.* **2020**, *12*, 256. [[CrossRef](#)]
21. Onyekpe, U.; Palade, V.; Kanarachos, S.; Christopoulos, S.R.G. Learning uncertainties in wheel odometry for vehicular localisation in GNSS deprived environments. In Proceedings of the 2020 19th IEEE International Conference on Machine Learning and Applications (ICMLA), Miami, FL, USA, 14–17 December 2020.
22. Onyekpe, U.; Palade, V.; Herath, A.; Kanarachos, S.; Fitzpatrick, M.E. WhONet: Wheel Odometry neural Network for vehicular localisation in GNSS-deprived environments. *Eng. Appl. Artif. Intell.* **2021**, *105*, 104421. [[CrossRef](#)]
23. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, *30*.
24. Onyekpe, U.; Kanarachos, S.; Palade, V.; Christopoulos, S.G. Vehicular Localisation at High and Low Estimation Rates during GNSS Outages: A Deep Learning Approach. In *Deep Learning Applications*; Wani, V.P.M.A., Khoshgoftaar, T., Eds.; Springer: Singapore, 2020; Volume 2, pp. 229–248.
25. Onyekpe, U.; Palade, V.; Kanarachos, S.; Szkolnik, A. IOVNBD: Inertial and Odometry Benchmark Dataset for Ground Vehicle Positioning. *arXiv* **2020**, arXiv:2005.01701.
26. Onyekpe, U.; Palade, V.; Kanarachos, S. Learning to Localise Automated Vehicles in Challenging Environments Using Inertial Navigation Systems (INS). *Appl. Sci.* **2021**, *11*, 1270. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.