Article

A Railway Lidar Point Cloud Reconstruction Based on Target Detection and Trajectory Filtering

Hao Liu [1,*], Lianbi Yao [1], Zhengwen Xu [1], Xianzheng Fan [2], Xiongfeng Jiao [2] and Panpan Sun [2]

[1] College of Surveying and Geo-informatics, Tongji University, Shanghai 200092, China
[2] China Railway Shanghai Design Institute Group Corporation Limited, Shanghai 200070, China
* Correspondence: 2032020@tongji.edu.cn

Abstract: The traditional railway survey adopts a manual observation method, such as a total station measuring system. This method has high precision, but the amount of data is small, and the measurement efficiency is low. Manual measurement cannot meet the requirements of dynamic continuous high-precision holographic measurement during railway outages. Mobile laser scanning is a mobile mapping system based mainly on a laser scanner, inertial measurement unit (IMU) and panoramic camera. Mobile laser scanning has the advantages of high efficiency, high precision and automation. However, integrating inertial navigation data and mobile laser scanning data to obtain real 3D information about railways has always been an urgent problem to be solved. Therefore, a point cloud reconstruction method is proposed based on trajectory filtering for a mobile laser scanning system. This paper corrects the odometer data by identifying railway feature points through deep learning and uses Rauch–Tung–Striebel (RTS) filtering to optimize the trajectory results. Combined with the railway experimental track data, the maximum difference in the east and north coordinate direction can be controlled within 7 cm, and the average elevation error is 2.39 cm. This paper applies a multi-sensor integrated mobile detection system to railway detection. It is of great significance to the healthy development of the intelligent railway system.

Keywords: mobile laser scanning; point cloud; feature recognition; deep learning; inertial measurement unit; multi-sensor

1. Introduction

In recent years, with the continuous advancement of China's strategy of strengthening China's transportation, the scale and speed of railway tunnel construction have made remarkable achievements. By the end of 2020, 16,798 railway tunnels had been built and operated across the country, with a total length of about 19,630 km, and the total mileage of the built subway tunnels exceeds 6000 km. Railway operating enterprises attach great importance to tunnel safety and invest a lot of manpower and material resources to carry out disease investigation, deformation monitoring, operation and maintenance inspection, etc. [1–4]. In different stages of rail transit construction, operation and maintenance, comprehensive testing with high precision, high reliability and high efficiency are necessary to ensure operational safety.

With the continuous growth of railway operating mileage and its proportion to public transportation, the operation and maintenance costs are high, and the demand for efficient inspection and engineering monitoring during the operation period is also increasing rapidly. The traditional method uses manual inspection, total station and level to monitor the structural deformation, but the operation efficiency is difficult to meet the detection needs of the normal maintenance and maintenance of the super-large-scale network in the future [5]. In the context of the rapid expansion of railways, the operation and maintenance department urgently needs an efficient, precise and comprehensive inspection technology to replace the traditional inspection method in order to inspect the operating tunnel during the skylight time.

In recent years, non-contact detection methods based on laser scanning have been rapidly developed. As technology has advanced, efficient and intelligent comprehensive detection technologies have been developed for the rail transit industry. In particular, laser scanning technology has the advantages of miniaturization of equipment, measurement and imaging, no need for lighting, and low operating cost. In recent years, it has been widely used in the construction and operation of railways and subways [6–11]. Shanghai, Nanjing, Shenzhen, Guangzhou, Tianjin, Wuhan and other cities have generally applied laser scanning technology for regular convergence and deformation surveys and structural disease surveys. The railway mobile laser measurement system used in this study has a fast measurement speed and richer results, and it is 5–10 times more efficient than traditional measurement methods.

For complex scenes such as railways, relevant research institutes at home and abroad have developed various types of orbital mobile laser scanning systems. Sturari et al. [12] proposed a new method to mix visual and point cloud information, using a hybrid of the Felix robot's multi-view camera and linear laser. This method is effective for railway safety and infrastructure change detection. Stein et al. [13] used template matching and spatial clustering to process the laser scanning data of the railway and obtained the rail detection results through quantitative and qualitative experiments. The GRP5000 developed by Amberg [14] in Switzerland is equipped with a PROFILER 6012 3D scanner, which integrates a tilt sensor, an odometer and a displacement sensor. During the mobile scanning process, the prism installed on the car body is tracked and measured by the total station to obtain the absolute coordinates of the track. The point cloud results can be used for track scanning, central axis extraction, gap analysis, boundary and deformation detection. The VMX-RAIL mobile laser measurement system of Riegl, Austria [15] uses three high-speed laser scanners to reduce scanning shadows and achieve 130 km/h high-speed synchronous acquisition of point clouds and images. It supports third-party software for cross-section measurement, limit measurement, asset survey, etc. Li [16] et al. measured the high-precision 3D curves of railway tracks using a rail trolley equipped with a laser-assisted inertial navigation system (INS)/odometer system. A model for measuring the irregularity of the track curve was established, and the actual track irregularity was measured. SUN [17] et al. proposed a tunnel environment disease and deformation detection system with laser scanning as the main sensor. This system provided functions such as tunnel structure deformation and disease extraction. CHEN [18] proposed a track extraction algorithm based on generalized neighborhood height differences. Based on the track extraction with the last track, it can accurately extract the track information of the entire railway.

Most of the current research on POS (position and orientation system) focuses on different types of combined navigation and odometer combined methods, filtering and smoothing models, and precise single-point positioning technology. However, the error characteristics of the odometer in the actual harsh environment are less involved, and there is no mature error suppression or elimination method [19–21]. Because of the special environment of railway tunnels with poor positioning, domestic and foreign researchers have done a lot of research. Tsujimura et al. [22] proposed a localization scheme for tunnel robots that rely on fixed orbits for navigation. This method utilizes a magnetic flux position measurement system installed on the ground and underground tunnel robots to locate the robot. However, this positioning scheme is only suitable for robots with fixed orbits, and it is difficult to achieve high dynamic continuous positioning. JING et al. [23] achieved accurate 3D reconstruction of objects using a tunnel robot to obtain depth and true-color images. JING et al. used an improved iterative closest point to obtain single-point positions and assembled dense point cloud reconstruction.

In the above research on orbital mobile laser scanning systems, these systems are costly in terms of hardware and software. Because of the large volume and weight of measuring equipment, workers spend more time on installation and operation. The software offers only a single mode of data processing, and the software cannot be customized to the complex Chinese railway environment. There is no corresponding filtering method for the

odometer and INS data fusion in the POS calculation. At the same time, there is a lack of a method to reduce the accumulation of odometer errors. Therefore, how to study a low-cost, high-precision, and efficient railway mobile measurement system is an urgent problem to be solved.

In this study, a mobile laser scanning system is used for data collection, and ortho-photo images are used to realize the automatic identification of feature points and mileage correction. The trajectory data are optimized by the Kalman filter (KF) to obtain high-precision trajectory data.

In Section 2, we introduce the hardware and software design of the railway mobile laser scanning system. Section 3 introduces the generation method of orthophoto, the method of mileage and pose correction and the method of high-precision point cloud generation. Section 4 evaluates the accuracy of the railway laser scanning results. In Sections 5 and 6, future research prospects and recommendations, as well as the final findings of this paper, are discussed.

2. Railway Mobile Laser Scanning System

In order to achieve the purpose of three-dimensional absolute coordinate measurement, it is necessary to integrate the laser scanner and the position and attitude system into the system platform. Furthermore, the instrument platform is mounted on the mobile carrier to perform fast dynamic scanning during the carrier movement. Therefore, it is necessary to design a set of railway mobile laser scanning devices in-house. The self-developed system integrates two-dimensional cross-section laser scanners, INS, panoramic cameras, and a tablet into a mobile carrier that can run on the track.

Considering the ease of installation and disassembly of the platform, the occlusion of the camera's field of view, the occlusion of the Global Navigation Satellite System (GNSS) antenna, and the stability of sensor installation, a multi-sensor integration box is designed. The platform integration design is shown in Figure 1, which mainly includes: GNSS antennas, panoramic camera, laser scanner, tablet, and IMU. 10 mm aluminum alloy plates splice the main structure, and each plate has a weight reduction design. The outside is bent and welded by stainless steel plates to reduce exposed screws. The raw materials of each component must meet the requirements of lightweight and high rigidity. The data of scanner, INS, panoramic camera, and mileage are collected through a tablet, which adopts a capacitive touch screen and embedded wall-mounted installation. The present invention completes the startup of the device, the expansion of the power supply, and the transmission and downloading of data through the control panel, thereby simplifying the data transmission process. In the present invention, the scanner is fixed in the instrument case through the threads on both sides, which is convenient for installation and disassembly. A threaded interface is designed on the base of the instrument, which is suitable for various rail cars and official vehicles. The platform reserves total station and prism installation interfaces, and corresponding workbenches can be added according to subsequent needs.

The laser scanner selected in this paper is the German Z+F 9012 section scanner, which adopts the principle of laser ranging, emits laser pulses to the target surface, and records information such as distance, angle, and reflection intensity. To achieve fast and high-density acquisition of a large number of point cloud data of objects, the scanning rate of the system exceeds 1 million points per second, the maximum scanning speed is 200 rpm, and the distance resolution is 0.1 mm.

The inertial navigation system used in this article is the XD300A-DGI, which consists of a cost-effective closed-loop fiber optic gyroscope, an accelerometer, and a high-end GNSS receiving board. The bias stability of the inertial accelerometer is 5×10^{-5} g, and the bias stability of the gyro is $0.02°/h$. The IMU consists of three accelerometers and three fiber optic gyroscopes. It is responsible for measuring the acceleration and angular velocity of the carrier and sending this information to the information processing circuit. The information processing circuit utilizes the acceleration and angular velocity measured by the inertial measurement unit for navigation and settlement. At the same time, using the

satellite navigation information received by the GNSS board as a reference, comprehensive navigation is performed to correct the navigation error of the inertial navigation.
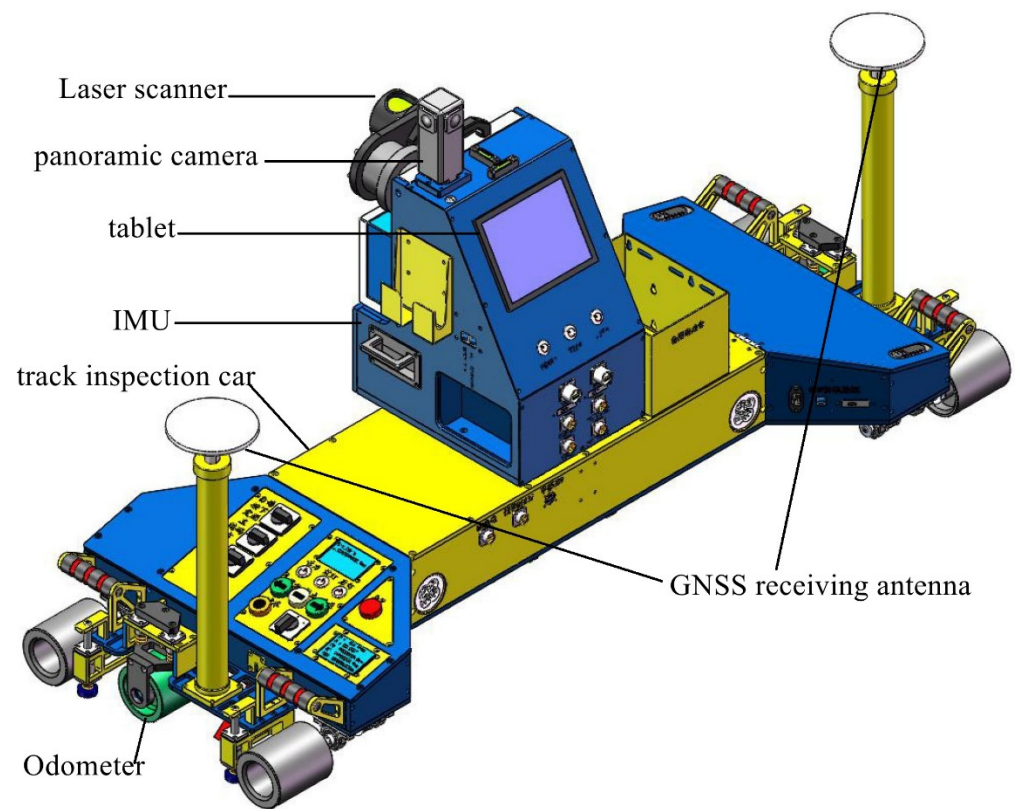

Figure 1. Schematic diagram of the mobile railway inspection vehicle, including a laser scanner, IMU, GNSS antenna, a tablet, an odometer, and the panoramic camera.

3. Methodology

This paper divides the process of target detection and trajectory filtering into four steps: orthophoto generation, mileage localization method based on You Only Look Once (YOLO), location update and RTS trajectory filtering, and high-precision 3D point cloud reconstruction. The calculation flow chart is shown in Figure 2.

3.1. Orthophoto Generation

The laser sensor emits a laser beam with a specific transmission power, and the laser beam is reflected and scattered when it reaches the surface of the object and is received by the laser sensor [24]. The received echo power is obtained after processing by the internal system. The receiver converts the received power into voltage and digitally outputs an integer value, which is the intensity value. According to the scan lines, raster grayscale images are generated by performing projection expansion. The intensity value of the Z+F 9012 series laser scanner is stored in a 16-bit data length, and the acquired point cloud intensity range can be exported as [0, 1] or [0, 65,525] and other different grayscale ranges.
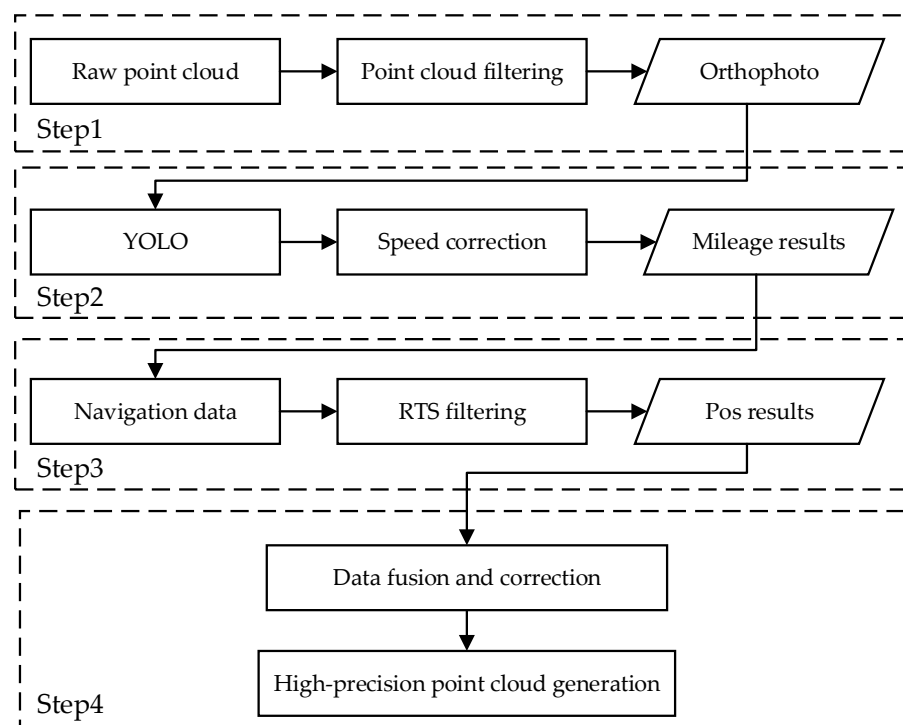
Figure 2. Flow chart of data processing of railway mobile laser scanning.

In order to highlight the difference in the point cloud intensity values of railway scanning feature objects, it is necessary to enhance the acquired point cloud intensity values. The full-scale grayscale histogram stretching method is adopted. The original intensity range is mapped to [0, 1] through a linear change. Let Q be the laser point set, and the original intensity value of each point is $W(i)$, $i \in Q$. Get the min and max in the raw intensity as follows:

$$W_{min} = \min_{i \in P} W(i) \tag{1}$$

$$W_{max} = \max_{i \in P} W(i) \tag{2}$$

Linearly map it to 0 and 255, respectively, to get the stretched strength value $W_{stretch}(i)$:

$$W_{stretch}(i) = \frac{255}{W_{max} - W_{min}} * (W(i) - W_{min}) \tag{3}$$

The original scanned point cloud has a small number of intensity minimum or maximum points. Only using this method to stretch the point cloud intensity value has certain defects. In order to solve this defect, on the basis of stretching the full-level grayscale histogram, an intensity value stretching method considering the number of point clouds is adopted. The minimum number of point clouds is used as a threshold to reduce the influence of intensity noise points. As a result, the intensity of the point cloud is firstly counted, the number of point clouds corresponding to each intensity value in the intensity value is obtained, and the histogram curve is drawn. As shown in Figure 3, the minimum and maximum strengths with points greater than the threshold are counted as the minimum and maximum strengths of strength stretching. The intensity image is shown in Figure 4.
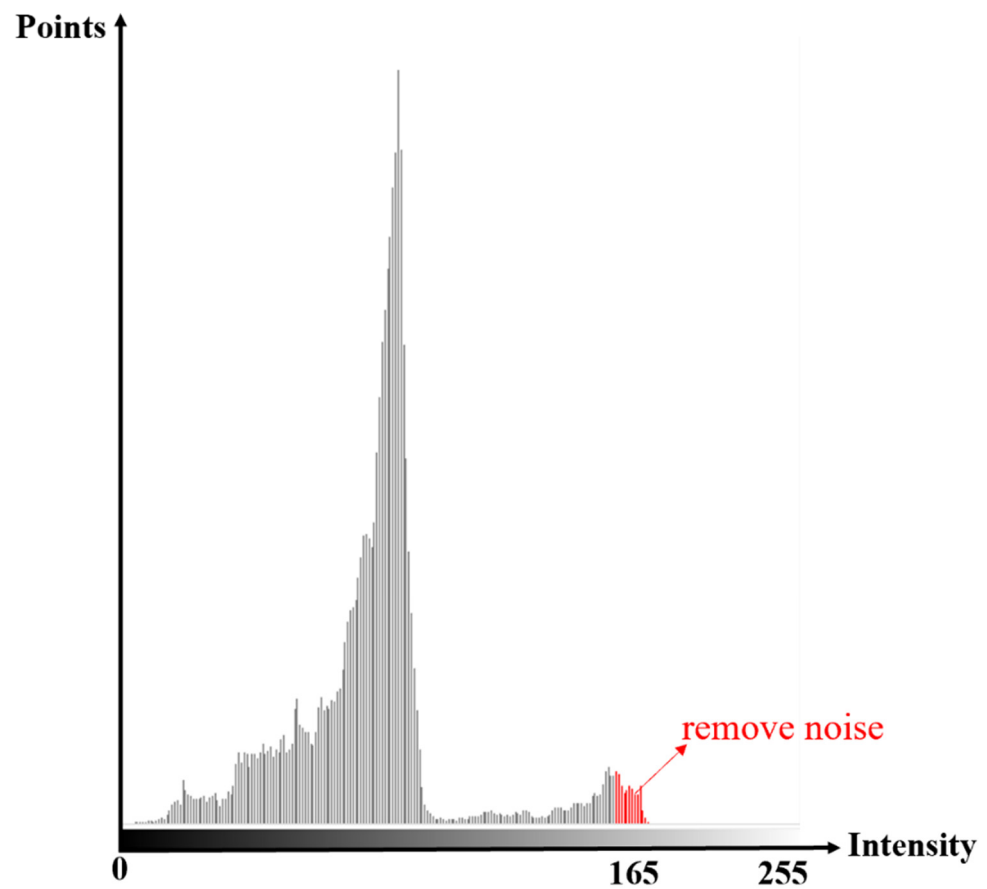
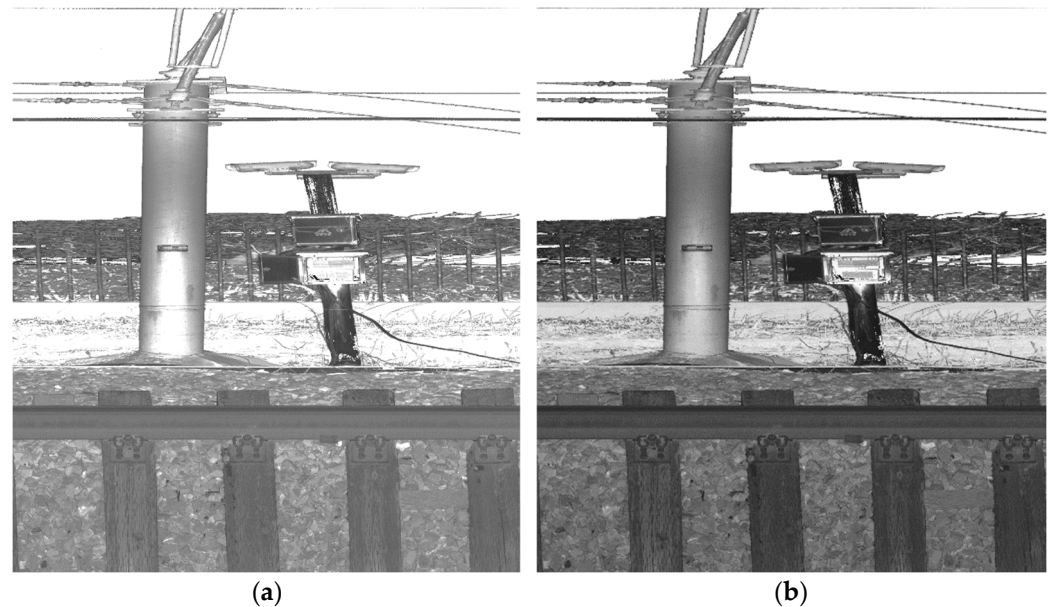**Figure 3.** Removal of unusual parts of the intensity values (red parts) by intensity histogram filtering.

(**a**)　　　　　　　　　　　　　　　　　(**b**)

Figure 4. Intensity image generated from partial railway point cloud data: (**a**) Before intensity filtering; (**b**) After intensity filtering.

A depth image, also known as a range image, refers to an image that uses the distance from the image collector to each point in the scene as a pixel value. The laser scanning system realizes the calculation of the depth image by comparing the time delay or phase of the transmitted signal and the received signal.

Firstly, the discrete laser point cloud data are resampled by the regular grid to obtain a digital depth (distance) matrix. Resampling is a method to determine the pixel value for any unknown point based on the four adjacent known pixel points. The pixel value of the unknown point is obtained by a first-order linear calculation in the horizontal and vertical directions. Different depth values are grayscale quantized and stretched. Taking this as the pixel value, we get a depth image that only contains the distance information. In this imaging process, interpolation errors are generated during resampling. For fewer errors, adaptive filtering of the resulting depth image is required.

Adaptive filtering is the process of dividing an image into sub-blocks and detecting the noise by applying it to the pixels in each sub-block. After adaptive filtering, the noise in the original image is suppressed, the main features of the image become more obvious, and the depth image is shown in Figure 5.
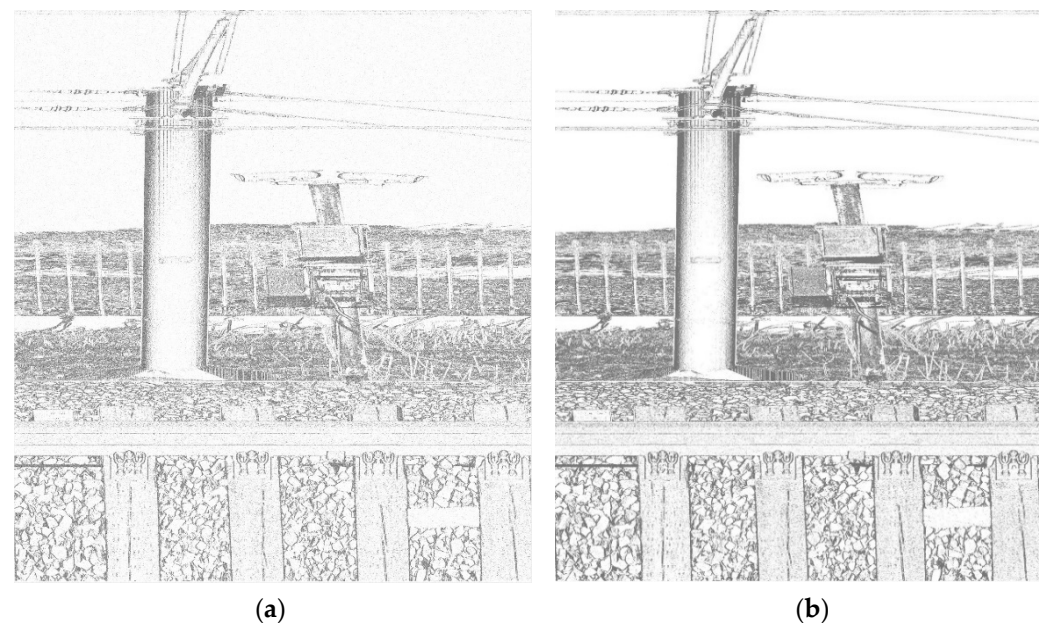

(**a**)　　　　　　　　　　　　　　　(**b**)

Figure 5. Depth image generated from partial railway point cloud data: (**a**) Before adaptive filtering; (**b**) After adaptive filtering.

The point cloud intensity filtering eliminates the influence of some external factors on the intensity value so that the point cloud intensity value is only related to the target reflectivity, which can improve detection accuracy. On the other hand, the distribution range characteristics of point cloud laser intensity values have been clarified by scanning various objects. The processing of point cloud depth images is mainly to enhance the expression of characteristic structures through image processing. Adaptive filtered images are used for target classification and identification based on geometric and spatial features of the structure.

3.2. Mileage Correction

3.2.1. YOLO Target Detection

At present, the method of mileage value acquisition is mainly based on an odometer. But the accuracy of the odometer is easily affected by complex environments such as wheel slippage, detection speed, and rail wear, resulting in cumulative errors in measurement results [25]. In this study, the deep learning YOLOv3 target detection algorithm is used to extract regular feature points such as fasteners in the orthophoto. The method of equal-spaced correction is used to reduce the cumulative error effect caused by abnormal mileage values.

As one of the representative detection networks in the field of object detection, the YOLOv3 model is widely used in the field of rapid detection of small objects. YOLOv3 is a target detection algorithm with both accuracy and speed. It inherits and integrates the network structure of other target detection algorithms so that its overall performance has reached a high level. Its network structure is shown in Figure 6 [26–28]. As YOLOv4, YOLOv5, and others, have also been proposed one after another, it has been proved that the YOLO algorithm still has great potential for improvement [29,30].
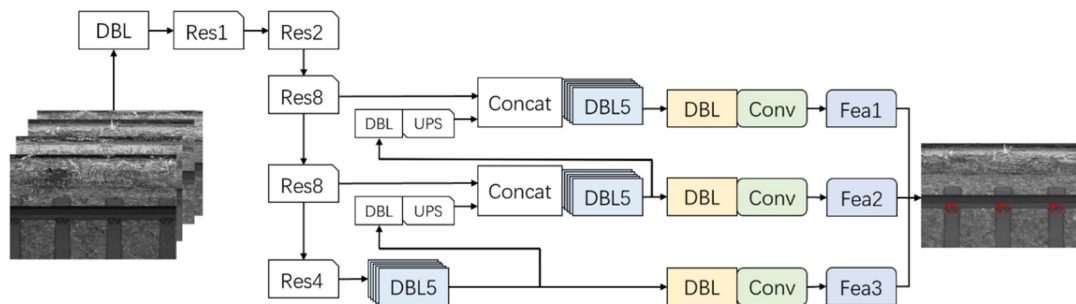

Figure 6. YOLOv3 network structure.

Since the track fasteners to be identified are small object targets, the input of the network is modified from $416 \times 416 \times 3$ to $533 \times 533 \times 3$, and the resolution improvement is beneficial to retain more useful information. The multi-scale fusion operation of the feature pyramid is realized through multiple convolutions and up-sampling functions between the scale feature maps.YOLOv3 not only deepens the network depth through a large number of residual structures but also uses various methods such as feature fusion, multi-scale prediction, and bounding box regression to achieve a great improvement in detection speed and accuracy.

The relationship between the YOLOv3 target box and the prior box is shown in Figure 7. YOLOv3 does not directly predict the position and size of the target but obtains the offset of the grid cell of the feature map by regression to determine the target's center position. The size of the target is determined according to the scale factor of the target box size relative to the prior box.

The regression formula is:

$$b_x = \sigma(t_x) + c_x \tag{4}$$

$$b_y = \sigma(t_y) + c_y \tag{5}$$

$$b_w = p_w e^{t_w} \tag{6}$$

$$b_h = p_h e^{t_h} \tag{7}$$

In the formula, (c_x, c_y) represents the center coordinates of the preset bounding box on the feature map; (p_w, p_h) represents the width and height of the prior box on the feature map; (t_x, t_y) represents the offset of the prediction center relative to the grid point; (t_w, t_h) represents the predicted bounding box width and height scaling ratio; (b_w, b_h, b_x, b_y) represents the final predicted target bounding box; σ represents the use of an activation function to scale the prediction offset into the interval 0 to 1.

In order to improve the speed of model training, the K-means method is used to perform cluster analysis of the data set, and the initial prediction frame parameters that are closer to the detection target in the orthophoto image are obtained. During the training process, in order to prevent overfitting caused by too many iterations, a loss value protection function is added to improve the stability of model training.
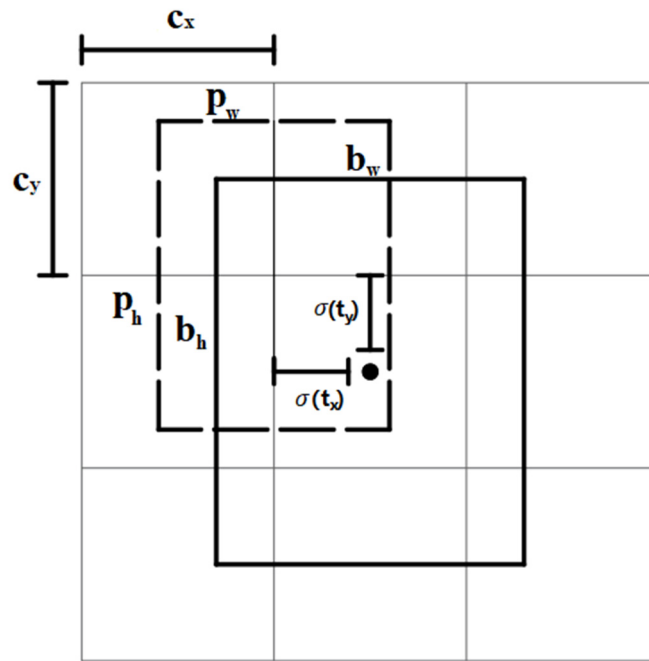
Figure 7. Bounding boxes with dimension priors and location prediction.

3.2.2. Performance Evaluation Indicators

The loss function is an important indicator for evaluating the model's performance. During the model's training process, the parameters in the network are continuously adjusted, and the value of the loss function loss is optimized to minimize the model's value to complete the model's training. The loss function of YOLOv3 consists of three parts, including confidence loss (L_{conf}), class loss error (L_{cla}) and localization loss (L_{loc}), $\lambda_1, \lambda_2, \lambda_3$ are balance weight coefficients. The loss function can be specifically expressed by the following formula:

$$L(O, o, P, p, l, g) = \lambda_1 L_{conf} + \lambda_2 L_{cla} + \lambda_3 L_{loc} \tag{8}$$

The target confidence loss (L_{conf}) is expressed as:

$$L_f(o, p) = -\sum (o_i \ln(p'_i) + (1 - o_i) \ln(1 - p'_i)) \tag{9}$$

$$p'_i = Sigmoid(p_i) \tag{10}$$

In the formula, p_i represents the confidence score, and o_i represents the ratio of the predicted box to the ground-truth box.

The target class loss (L_{cla}) is expressed as:

$$L_{cla}(O, P) = -\sum_{i \in Pos} \sum_{j \in cla} (O_{ij} \ln(P'_{ij}) + (1 - O_{ij}) \ln(1 - P'_{ij})) \tag{11}$$

$$P'_{ij} = Sigmoid(P_{ij}) \tag{12}$$

In the formula, O_{ij} represents the binary cross-entropy loss, and P_{ij} represents the probability of the existence of the j-th object in the predicted object bounding box.

The target localization loss is expressed as:

$$L_{loc}(l, g) = \sum_{i \in Pos} \sum_{m \in \{x,y,w,h\}} (l_i^m - g_i^m)^2 \tag{13}$$

In the formula, l_i^m represents the value of the coordinate offset processed by the activation function, and g_i^m represents the coordinate offset between the real frame and the default frame.

The algorithm test in this paper uses Mean Average Precision (mAP), Recall, and Precision to evaluate the performance of the algorithm. The evaluation formulas are:

$$mAP = \frac{\sum_{i=0}^{k} AP}{m} \tag{14}$$

In the formula, AP represents the accuracy rate corresponding to each target, and m represents the total number of target object categories.

$$Recall = \frac{TP}{TP + FN} \tag{15}$$

$$Precision = \frac{TP}{TP + FP} \tag{16}$$

In the formula, TP is the positive detection rate, that is, the calibration is a positive sample, and the actual detection is also a positive sample; FP is the false detection rate, that is, the calibration is a negative sample, and the actual detection is a positive sample. FN is the missed detection rate; it is calibrated as a positive sample, and the actual detection is a negative sample.

The target detection system filters out useless target boxes below the threshold according to the settings of the confidence threshold and the intersect over the union threshold. If the threshold is too high, it may cause some real targets to be filtered out and missed. If the threshold is set too low, some targets for model error detection will not be filtered out and will be treated as normal targets. This paper selects the best confidence threshold through multiple tests. When the confidence threshold is 0.50, and the intersect over union threshold is 0.55, more targets are correctly detected, and no false targets occur. Finally, we use the above thresholds for model training to ensure that more correct targets are detected.

3.2.3. Mileage Constraints and Positioning

Due to the complex environment of mobile scanning, the YOLO target detection algorithm has certain errors, and only deep learning cannot extract all the coordinate information of feature points. Therefore, this paper first adopts the method of distance clustering to segment the target set of fasteners identified by deep learning. If the number of targets in the clustering set is less than three, the set is not calculated. This method can effectively eliminate the feature information that is wrongly identified and count the number of missing feature points. Secondly, the missing fastener feature points are complemented using Lagrangian interpolation, and their coordinate values can be expressed as:

$$y(t) = \sum_{\substack{j=0 \\ j \neq i}}^{n} \left(\prod_{i=0}^{n} \left(\frac{p - p_i}{p_j - p_i} \right) \right) y_j \tag{17}$$

In the formula, t is the missing feature node; p_0, p_1, p_2, ..., p_n are the feature points obtained by clustering. y_0, y_1, y_2, ..., y_n are the abscissa values in the pixel coordinate system corresponding to the cluster feature points.

According to the obtained abscissa information of the feature points and the starting and ending mileage information of the measurement interval, the accurate mileage value corresponding to each feature point can be obtained, which solves the problem that the image coordinates do not match the actual mileage in the mobile measurement. The main calculation process is as follows:

(1) Mileage constraint: Calculate the total length D of the interval according to the information of the starting and ending mileage plates. According to the target detection

result, the pixel interval y_i of the feature point is obtained. Combined with the change value y of the abscissa of the start and end pixels, the interval value d_i of each feature point is calculated. We obtain the mileage value between the feature points according to the pixel value interpolation method:

$$d_i = d_{i-1} + \frac{y_i}{y} * D \tag{18}$$

In the formula, D represents the length of the known start and end points, and d_i represents the mileage value of the feature points identified by target detection. y represents the pixel interval between start and end points, y_i represents the pixel interval of consecutive feature points.

(2) Initial positioning: Starting from the initial feature point mileage, iteratively accumulate the interval values of each feature point to obtain the mileage value of each feature point. According to the position of the pixel in the pixel interval m_i, we obtain the mileage value of the pixel point s'_i:

$$s'_i = d_{i-1} + \frac{m_i}{y_i} * d_i \tag{19}$$

In the formula, s'_i represents the initial mileage of any pixels, and m_i represents the pixel values within adjacent feature points.

(3) Speed analysis: According to the scanning frequency f of the laser scanner, the time interval T_i of the scanning line is calculated. The corresponding speed v'_i is obtained based on the finite element method, and the mean filtering method is used to further optimize the speed value. The influence of the noise in the system on the speed value is weakened, and the speed value v_i of the moving carrier in each period is obtained.

$$v'_i = (s'_i - s'_{i-1})/T_i \tag{20}$$

$$v_i = g(v'_i, w) \tag{21}$$

In the formula, g represents the mean filter, and w represents the window size of mean filtering. T_i represents the time interval between adjacent scan lines, v'_i represents the initial velocity value, and v_i represents the filtered velocity value.

(4) Mileage correction: Correct the initial mileage value according to the optimized speed value to obtain the final mileage value s_n.

$$s_n = \sum_{i=0}^{n} M_i = \sum_{i=0}^{n} v_i T \tag{22}$$

In the formula, M_i represents the mileage value of adjacent pixels after filtering, and s_n represents the mileage value of the target point after filtering.

3.3. Inertial Navigation Data Processing Model

3.3.1. Location Update Algorithm

Dead reckoning is a technology that uses the known position, combined with the moving speed and bearing, to estimate the existing position [31]. Because it does not need to receive external signals, it is suitable for estimating the position of the mobile measurement system in the position without a GNSS signal. Furthermore, we can form an inertial navigation system with the data of the GNSS navigation data to overcome the shortcoming that the accumulated error will increase with time.

When the inertial navigation data are used as the original data for position update, the attitude update adopts the quaternion algorithm, and the formula is as follows:

$$Q(q_0, q_1, q_2, q_3) = q_0 + q_1 i + q_2 j + q_3 k \tag{23}$$

In the formula, q_0, q_1, q_2, q_3 are real numbers, i, j, k are both mutually orthogonal unit vectors and imaginary units $\sqrt{-1}$.

The projection of the carrier measured by the accelerometer relative to the inertial frame into the local coordinate system, namely:

$$f_{ib}^e(t_{k+1}) \approx \frac{1}{2}(C_b^e(t_k) + C_b^e(t_{k+1}))f_{ib}^b(t_{k+1}) \tag{24}$$

In the formula, $f_{ib}^b(t_{k+1})$ represents the specific force output of the accelerometer, and $f_{ib}^e(t_{k+1})$ represents the specific force projected to the local coordinate system. $C_b^e(t_{k+1})$ and $C_b^e(t_k)$ represent the pose matrices of two adjacent moments, respectively.

Using rectangular integration for the velocity differential equation gives:

$$
\begin{aligned}
\mathbf{v}^e(t_{k+1}) &= \mathbf{v}^e(t_k) + \int_{t_k}^{t_{k+1}}(\mathbf{f}_{ib}^e(t) - 2\mathbf{\Omega}_{ie}^e\mathbf{v}^e(t) + \mathbf{g}^e(t))dt \\
&\approx \mathbf{v}^e(t_k) + (\mathbf{f}_{ib}^e(t_{k+1}) - 2\mathbf{\Omega}_{ie}^e\mathbf{v}^e(t_k) + \mathbf{g}^e(t_k))\Delta t
\end{aligned} \tag{25}
$$

In the formula, $\mathbf{v}^e(t_{k+1})$ represents the velocity of the motion carrier obtained in the current epoch, $\mathbf{v}^e(t)$ represents the velocity of the motion carrier in the previous epoch, Δt represents the epoch interval, $\mathbf{\Omega}_{ie}^e$ is an obliquely symmetric matrix formed by the angular velocity of the carrier rotating relative to the local geographic coordinate system, and $\mathbf{g}^e$ represents the local gravity vector at the location of the carrier.

It is easy to derive from the position differential equation:

$$\mathbf{r}^e(t_{k+1}) = \mathbf{r}^e(t_k) + \int_{t_k}^{t_{k+1}} \mathbf{v}^e(t)dt \tag{26}$$

Usually using trapezoidal integration to solve the above equation, we get:

$$\mathbf{r}^e(t_{k+1}) \approx \mathbf{r}^e(t_k) + (\mathbf{v}^e(t_k) + \mathbf{v}^e(t_{k+1}))\frac{\Delta t}{2} \tag{27}$$

In the formula, $\mathbf{r}^e(t_k)$ and $\mathbf{r}^e(t_{k+1})$ represent the position of the motion carrier at adjacent times.

For the case of fusion calculation of odometer and inertial navigation data, it is necessary to use vehicle non-integrity constraints to assist in the calculation. For the convenience of research, the odometer measurement coordinate system is established, the y-axis is in the ground plane in contact with the carrier wheels and points to the front of the vehicle body, the z-axis is perpendicular to the ground plane and is positive, and the x-axis points to the right. Assuming that the carrier trolley does not drift, derail, and shake up and down when the vehicle runs as normal on the railway, the sky and lateral velocities can be set to 0 as constraints. According to the above definition, the speed output of the odometer in the navigation coordinate system can be obtained:

$$v_D^m = \begin{bmatrix} 0 & v_D & 0 \end{bmatrix}^T \tag{28}$$

$$v_D^n = C_b^n \cdot v_D^m \tag{29}$$

In the formula, v_D represents the forward speed of the odometer, v_D^m represents the speed vector of the odometer in the odometer coordinate system, and v_D^n represents the speed vector of the odometer in the navigation coordinate system.

The components of the odometer in the east, north, and sky directions of the navigation system are:

$$
\begin{cases}
v_{ODE} = -v_{OD}cos\theta sin\varphi \\
v_{ODN} = v_{OD}cos\theta cos\varphi \\
v_{ODU} = v_{OD}sin\theta
\end{cases} \tag{30}
$$

3.3.2. Installation Declination Error

In the inertial navigation system, the velocity measured by the odometer is usually projected to the inertial navigation coordinate system. In practical applications, the odome-

ter coordinate system and the inertial navigation coordinate system often do not coincide, and an installation error needs to be calibrated and compensated, as shown in Figure 8.
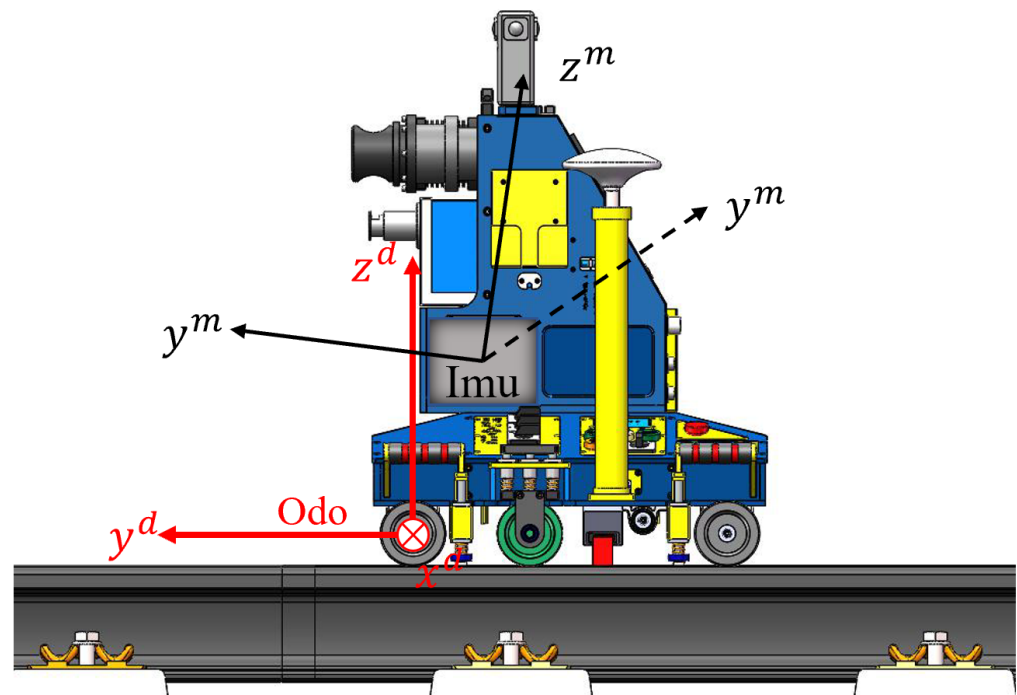

Figure 8. Installation declination error of the odometer coordinate system and the inertial navigation coordinate system.

Assuming that the transformation matrix between the odometer coordinate system and the inertial navigation coordinate system is C_b^d, α, β, γ represent the pitch angle, roll angle, and heading angle of the odometer coordinate system relative to the inertial navigation coordinate system. The attitude matrix is:

$$
\begin{aligned}
C_b^d &= \begin{bmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \\ C_{31} & C_{32} & C_{33} \end{bmatrix} \\
&= \begin{bmatrix} \cos\beta\cos\gamma - \sin\alpha\sin\beta\sin\gamma & \cos\beta\sin\gamma + \sin\alpha\sin\beta\cos\gamma & -\sin\beta\cos\alpha \\ -\cos\alpha\sin\gamma & \cos\alpha\cos\gamma & \sin\alpha \\ \sin\beta\cos\gamma + \sin\alpha\cos\beta\sin\gamma & \sin\beta\sin\gamma - \sin\alpha\cos\beta\cos\gamma & \cos\alpha\cos\beta \end{bmatrix}
\end{aligned}
\tag{31}
$$

In the case that the three installation error angles are all small angles, the above attitude matrix can be simplified as follows:

$$
C_b^d = \begin{bmatrix} 1 & \gamma & -\beta \\ -\gamma & 1 & \alpha \\ \beta & -\alpha & 1 \end{bmatrix}
\tag{32}
$$

Let $\boldsymbol{\varphi} = \begin{bmatrix} \alpha & \beta & \gamma \end{bmatrix}^{\mathrm{T}}$, then:

$$
C_d^b = \left(C_b^d \right)^{\mathrm{T}} = (\boldsymbol{I} + \boldsymbol{\varphi} \times)
\tag{33}
$$

In the formula, I is the identity matrix; $\varphi \times$ represents the cross-product antisymmetric matrix constructed by each component of φ.

The projection of v^{d} on the inertial navigation coordinate system is:

$$v^{\mathrm{b}} = C_{\mathrm{d}}^{\mathrm{b}} v^{\mathrm{d}} = (I + \varphi \times) v^{\mathrm{d}} = v^{\mathrm{d}} + (\varphi \times) v^{\mathrm{d}} \tag{34}$$

Arrange to get:

$$v^{\mathrm{b}} - v^{\mathrm{d}} = \left(-v^{\mathrm{d}} \times\right) \varphi = \begin{bmatrix} 0 & 0 & -v_D \\ 0 & 0 & 0 \\ v_D & 0 & 0 \end{bmatrix} \varphi \tag{35}$$

The speed of the vehicle in the navigation system can be obtained from the inertial navigation system output and the attitude transformation matrix $C_{\mathrm{n}}^{\mathrm{b}}$ is obtained from the inertial navigation output. Taking the quantity on the left side of the equal sign in the formula as the measured value and φ as the estimated value, the corresponding state can be estimated by the recursive least squares method.

3.3.3. RTS Filtering

A Kalman filter is an estimation method. Kalman filtering can estimate parameters in the system in real-time, such as continuously changing position and velocity, among others. The estimator is updated with a series of noise-contaminated observations, which must be a function of the parameter to be estimated, but may not contain enough information to determine a unique estimate over a given time.

After obtaining adequate information, the Kalman filter uses prior knowledge, such as deterministic and statistical properties of system parameters, as well as observations to obtain the optimal estimate, which is a Bayesian estimate. On the basis of the provided initial estimates, the Kalman filter performs a weighted average of the prior values and the new values obtained from the latest observation data through recursive operations to obtain the latest state estimates [32]. In order to achieve the optimal weighting of data, the Kalman filter has estimation uncertainty and can give the correlation measure between estimation errors of different parameters. This is achieved with step-by-step iterations of parameter estimation, which also incorporates the uncertainty of observations due to noise.

Since the 3D point cloud generation based on multi-sensor fusion uses the original data to estimate the trajectory position error as the observation update of the Kalman filter, when there is an observation update, the error of the position estimation and its covariance is tiny. However, due to the existence of residual systematic errors, the position measurement estimation error and its covariance between two observation updates will gradually increase with time. In order to obtain a high-accuracy position estimate in the whole measurement process, it is necessary to make full use of all observations to update the constraint error through an appropriate algorithm, which is a typically fixed interval smoothing problem. In the fixed interval smoothing algorithm, the commonly used methods mainly include the forward and backward Kalman filter smoothing algorithm and the RTS smoothing algorithm. The two algorithms are equivalent in theory, and the RTS smoothing algorithm is relatively simple to implement. In this paper, the RTS smoothing algorithm is used to process the data in the orbit measurement interval [33–35].

The RTS smoother consists of a typical forward Kalman filter and a reverse smoother. After each system propagation and observation update, the system state vector, error covariance matrix, and state transition matrix are recorded. After the data reaches the end, the data is smoothed from the endpoint to the start point in the reverse direction. The calculation flow of the RTS smoothing algorithm is shown in Figure 9:
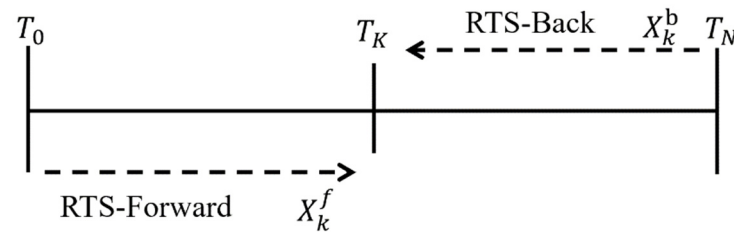
Figure 9. RTS smoothing algorithm flow.

The forward filtering of the RTS smoothing algorithm is a typical Kalman filter, and the smooth estimated value of the state vector is a weighted combination of two filtering estimated values. The calculation process is as follows:

$$X_{t,f}^{-} = F_t * X_{t-1,f} \tag{36}$$

$$P_{t,f}^{-} = F_t * P_{t-1,f} * F_t^{T} + \Gamma_t Q_t \Gamma_t^{T} \tag{37}$$

$$K_{t,f} = P_{t,f}^{-} * H_t^{T} * \left[H_t * P_{t,f}^{-} * H_t^{T} + R_t \right]^{-1} \tag{38}$$

$$X_{t,f} = X_{t,f}^{-} + K_t * \left[Z_t - H_t * X_{t,f}^{-} \right] \tag{39}$$

$$P_{t,f} = \left[I - K_t * H_t \right] * P_{t,f}^{-} \tag{40}$$

In the formula, $X_{t,f}$ and $P_{t,f}$ represent the optimal filter estimation value and its covariance matrix of the forward filtering state vector at time t, $X_{t,f}^{-}$ and $P_{t,f}^{-}$ represents the optimal one-step predicted value of the state vector and its covariance matrix at time t of the forward filtering, H_t is the observation matrix at time t, and $K_{t,f}$ is the optimal gain matrix of the forward filtering at time t. F_t represents the system state transition matrix, which can be obtained from the system error matrix A_t, and Γ_t is the discrete noise matrix at time t of the forward filter.

The inverse smoothing process can be expressed as:

$$K_{t,b} = P_{t,f} * F_t^{T} * \left(P_{t+1,f}^{-} \right)^{-1} \tag{41}$$

$$X_{t,b} = X_{t,f} + K_{t,b} * \left[X_{t+1,b} - X_{t+1,f}^{-} \right] \tag{42}$$

$$P_{t,b} = P_{t,f} + K_{t,b} * \left[P_{t+1,b} - P_{t+1,f}^{-} \right] * K_{t,b}^{T} \tag{43}$$

In the formula, $X_{t,b}$ and $P_{t,b}$ are the optimal smoothing estimation value and its error covariance matrix of the RTS smoothing algorithm at time t. $K_{t,b}$ is the optimal smoothing gain matrix of the RTS smoother at time t.

3.4. High-Precision Point Cloud Generation

One of the keys to multi-source data fusion of mobile scanning systems is time registration. First, the field data acquisition achieves zero-time identity through the synchronous control of integrated acquisition software. In the post-processing process, the data matching of the adjacent moments is carried out through the phase time interval to realize the consistency of the data, as shown in Figure 10.
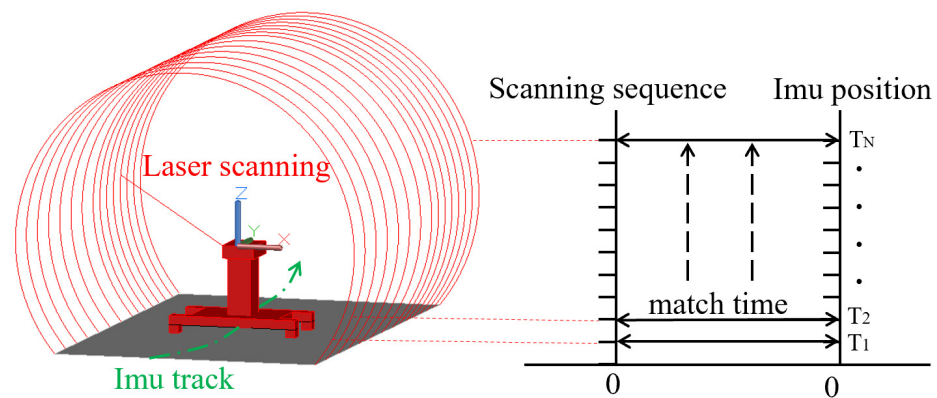
Figure 10. Fixed-time synchronization of laser scanning data and IMU data.

After the time synchronization between the sensors is completed, the relevant sensor data under the same time reference can be obtained, and the spatial calibration of the system hardware can unify the relevant sensor data on the same spatial reference. According to the transformation relationship between the coordinate systems and the model parameters, the scan data can be converted from the scanner coordinate system to the vehicle body coordinate system by using the calibration parameters of the system. Combined with the position and attitude data of the vehicle body coordinate system with the GNSS antenna phase center as the origin of the coordinate system calculated by the INS, the scanned data can be converted from the vehicle body coordinate system to the data in the absolute coordinate system. Finally, the 3D point cloud data based on the absolute coordinate system is obtained by completing the fusion processing of multi-source data.

The point cloud calculation formula for converting the point cloud data from the scanner coordinate system to the absolute coordinate system is:

$$\begin{bmatrix} X_C \\ Y_C \\ Z_C \end{bmatrix} = \begin{bmatrix} \Delta X_{BC} \\ \Delta Y_{BC} \\ \Delta Z_{BC} \end{bmatrix} + R_{BC}(\begin{bmatrix} \Delta X_{QB} \\ \Delta Y_{QB} \\ \Delta Z_{QB} \end{bmatrix} + R_{MB}\begin{bmatrix} x_Q \\ y_Q \\ z_Q \end{bmatrix}) \tag{44}$$

In the formula, $\left(x_Q, y_Q, z_Q\right)^T$ is the section coordinates based on the scanner coordinate system after the original Z+F 9012 scan data is parsed; the original data is recorded by the inertial navigation system. Combined with the body trajectory position parameters $\left(\Delta X_{BC}, \Delta Y_{BC}, \Delta Z_{BC}\right)^T$, the rotation matrix R_{BC} is provided by the combination of three attitude angles (yaw, pitch, roll); The translation matrix $\left(\Delta X_{QB}, \Delta Y_{QB}, \Delta Z_{QB}\right)^T$ and rotation matrix R_{MB} are obtained from the system calibration obtained from the scanner coordinate system to the vehicle body coordinate system.

Through the above calculation formula, the original data obtained by the scanner can be used to post-process the data of the inertial navigation system, combined with the transformation principle of the coordinate system, and finally, the 3D point cloud data based on the absolute coordinate system can be obtained.

4. Experimental Results

In order to verify the feasibility and stability of the railway mobile scanning system, in this study, the data acquisition and processing analysis test was carried out at the track comprehensive test site of the Jiading Campus of Tongji University, as shown in Figure 11. The total mileage of the test area is about 200 meters, and multiple control points are arranged in the middle of the test site as known points. In this study, multi-sensor integration was carried out at the test site, and a base station was established in an open area. The above-mentioned railway mobile laser scanning hardware integration system is used to collect and record the data of the scanner and the inertial navigation system and complete time synchronization for multiple data sources. The trajectory data is filtered and

corrected by using the method proposed in this paper. The accuracy is evaluated with the Inertial Explorer (IE) solution data. The commercial software meets the needs of mobile measurements for high accuracy positioning and attitude processing. Finally, multiple data are fused to obtain the railway's 3D continuous point cloud data. Furthermore, we compare the known points measured by the total station with the coordinate points extracted from the point cloud. The accuracy of the 3D point cloud was verified through the above experiment.

Figure 11. Mobile laser scanning system in experiment environment.

4.1. Target Detection and Mileage Calibration Accuracy

To the best of our knowledge, there is no standard dataset for railway fastener training and testing. Our previous study collected thousands of kilometers of mobile laser scanning data from Chinese railways. Then, the position of the fasteners in the laser image was marked by the tool Label-Image. Finally, a railway fasteners dataset was established. We randomly selected 925 images as the training data and designated the other 322 images as the testing data. After obtaining the training model, we performed target detection on the laser-scanned images for this experiment. At the same time, the positions of the fasteners were manually marked as the true value data to verify the target detection accuracy.

The precision rate of rail fasteners extracted by YOLOV3 is above 91.3%, and the recall rate is above 89.6%. The recognition results of some fasteners are shown in Figure 12. Using the method of feature point completion, the missing fasteners are interpolated and completed, and the center point position information of all fasteners in the interval is obtained.

Since the electric carrier's moving speed remains unchanged, the initial mileage positioning result and the carrier running speed in this interval are calculated according to the mileage constraint and the positioning algorithm. In order to avoid the influence of factors such as motion vibration and track irregularity, velocity filtering is used to optimize the results. Calculate the final interval mileage positioning result according to the optimized speed value.

The comparison results between the actual measured value and the mileage positioning algorithm are shown in Table 1. The mileage positioning error before filtering is within 9 cm, the mileage positioning error after filtering is about within 6 cm, and the standard error is 0.04. The accuracy of mileage positioning is high, which can better reduce the influence of the cumulative error of the odometer and meet the accuracy requirements of the trajectory solution.
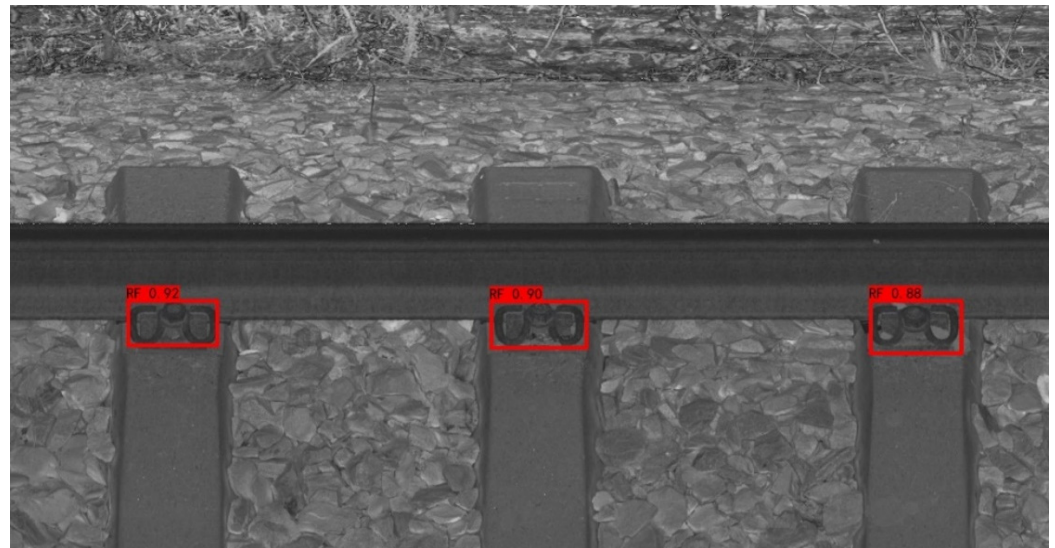
Figure 12. Feature point recognition results.

Table 1. Mileage positioning accuracy.

Truth Value (m)	Before Filtering (m)	After Filtering (m)	Error Before Filtering (m)	Error After Filtering (m)
YK660 + 0.97	YK0660 + 0.94	YK660 + 0.98	0.03	−0.01
YK680 + 0.23	YK680 + 0.19	YK680 + 0.22	0.04	0.01
YK700 + 0.69	YK700 + 0.60	YK700 + 0.63	0.09	0.06
YK720 + 0.45	YK720 + 0.54	YK720 + 0.50	−0.09	−0.05
YK740 + 0.28	YK740 + 0.33	YK740 + 0.27	−0.05	0.01
YK760 + 0.01	YK760 + 0.08	YK760 + 0.02	−0.07	−0.01
YK780 + 0.98	YK780 + 0.9	YK780 + 0.94	0.08	0.04
YK800 + 0.97	YK800 + 0.94	YK800 + 0.98	0.03	−0.01
Standard Deviation			0.07	0.04

4.2. Trajectory Correction Results

Before the on-board test, the installation deviation angle of the inertial navigation system was pre-calibrated by using the odometer information and the reference position point, and the heading angle (α) = −0.4579° and the heading angle (γ) = 0.5457° were obtained. The calibration results show that the installation deviation angle meets the requirements of a small angle. After adding the installation declination error, the influence of the system error caused by the installation can be better corrected. At the same time, the divergence trend of the inertial navigation trajectory over time is also suppressed, as shown in Figure 13.

In order to facilitate the analysis, the deviations in the east, north, and sky directions of the original estimated data, the RTS smoothing filter, and the ground truth data were calculated, respectively, as shown in Figures 14–16. The solid green line represents the RTS, and the KF error is represented by the solid red line. It can be seen from the schematic diagram that under the same conditions, the error of RTS in the north direction is the smallest, and the error in the east direction is relatively unstable. The instability of the inertial navigation system and the vibration of the system may cause direction anomalies. Except for a single outlier in the east direction, the error is below 5 cm in all other directions. Experiments show that the RTS algorithm has a better correction effect than the KF algorithm.
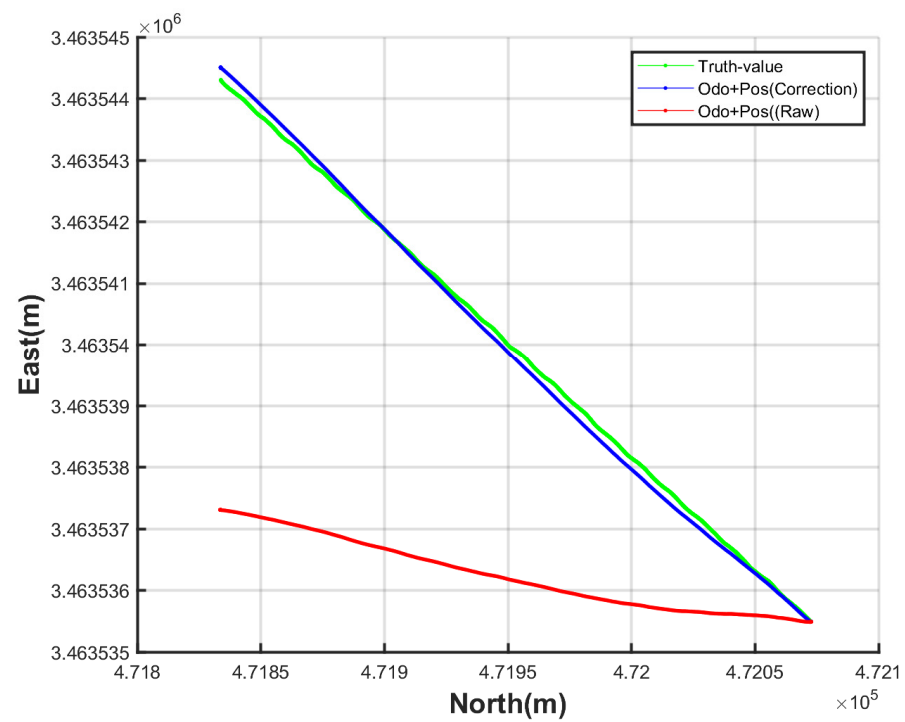
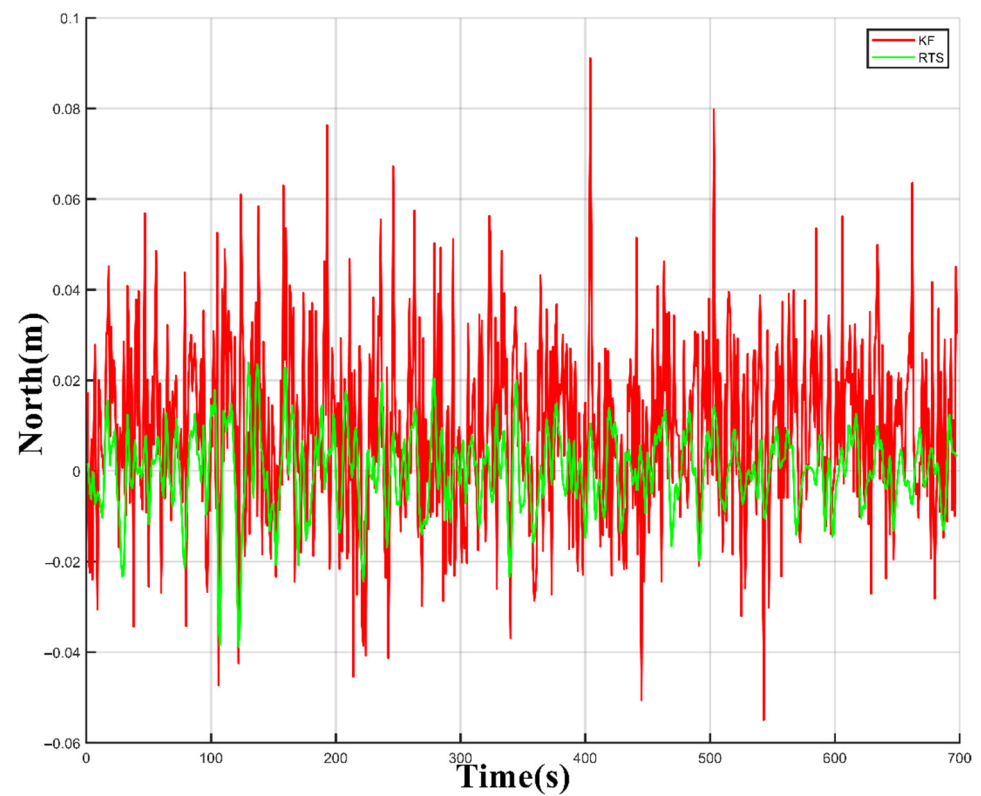**Figure 13.** The influence of installation declination on the trajectory.

Figure 14. Comparison of northward error between the KF and RTS.
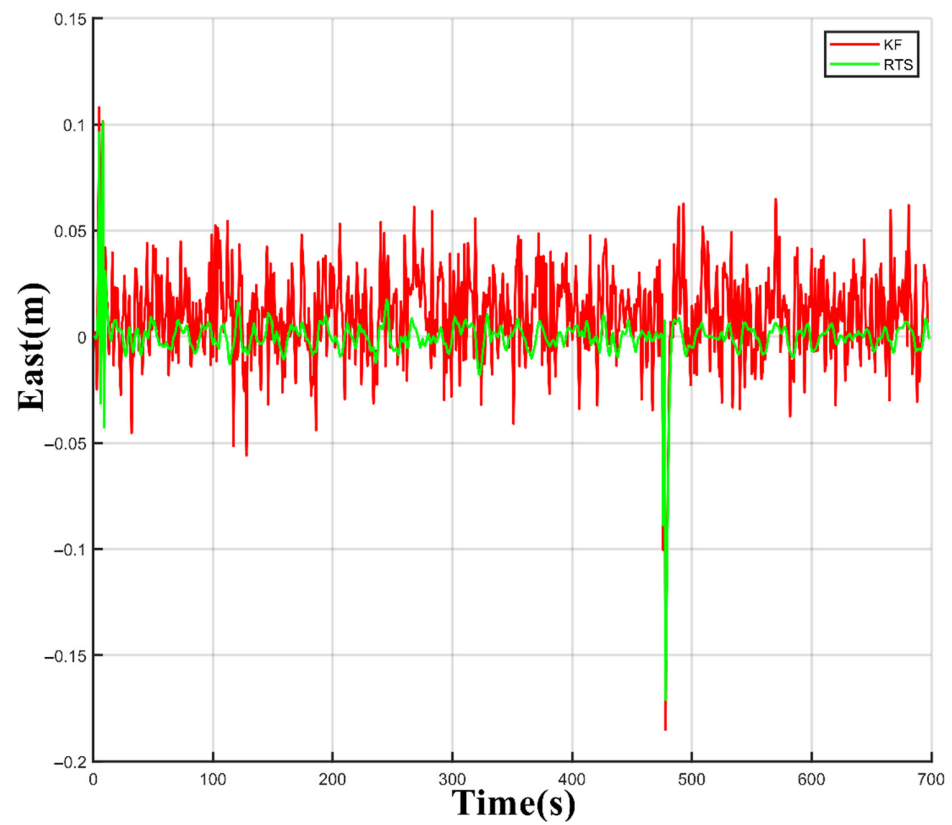
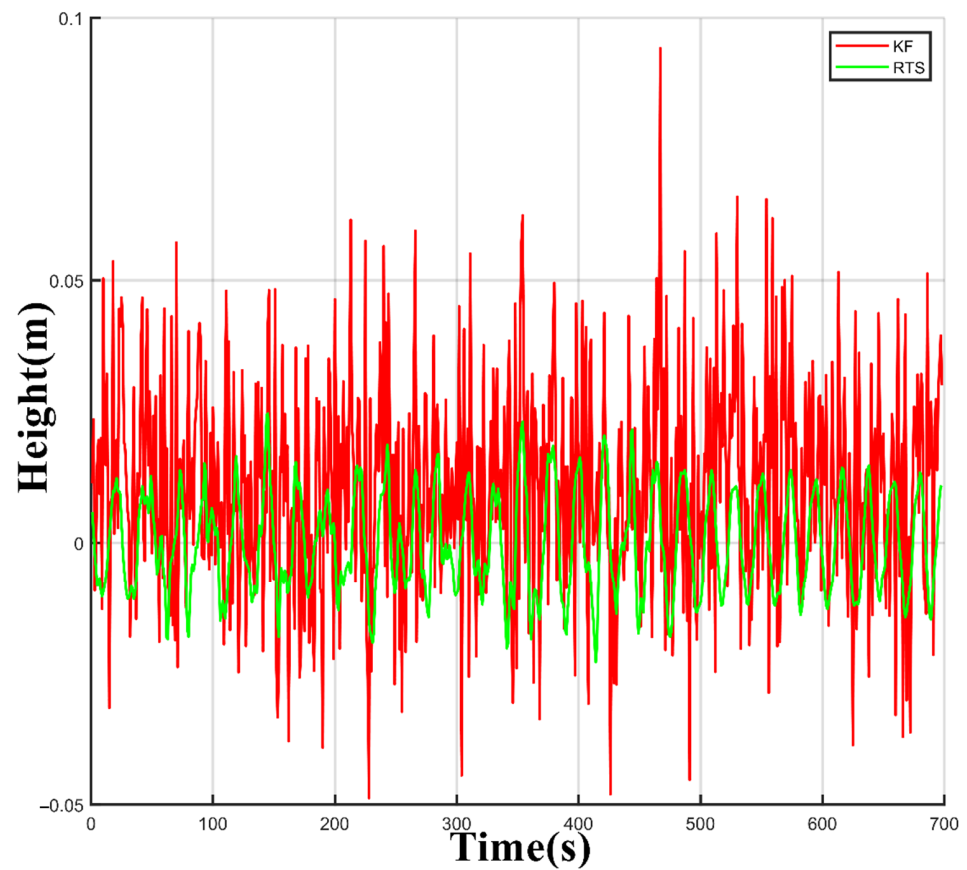**Figure 15.** Comparison of eastward error between the KF and RTS.

Figure 16. Comparison of elevation error between the KF and RTS.

4.3. 3D Point Cloud Generation Accuracy

After the data collection is completed, the trajectory data is filtered and smoothed to calculate the positioning and attitude information of the trolley trajectory on the railway track. Combining the solution trajectory, vehicle system calibration parameters, INS information, and scanning point cloud synchronization information for the calculation, the conversion from the relative point cloud to the geographic coordinate system is realized. The 3D point cloud data of the railway track in the Shanghai urban construction coordinate system is obtained, as shown in Figure 17.
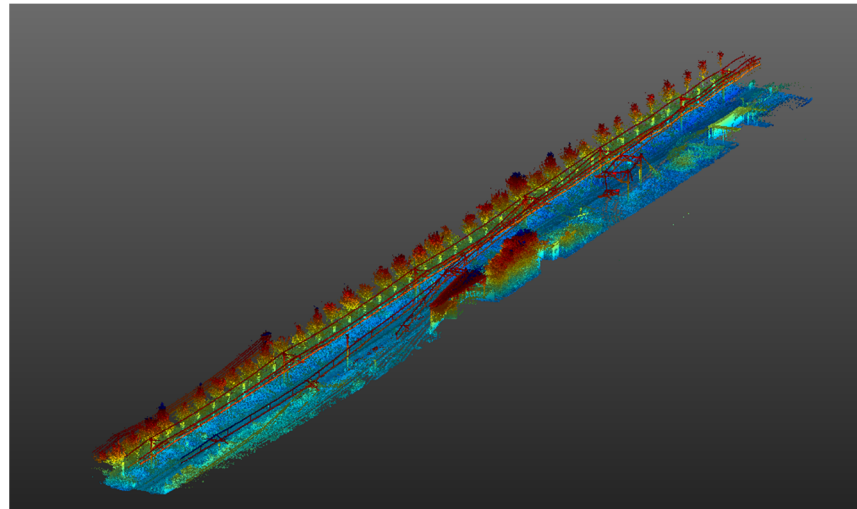

Figure 17. 3D point cloud effect of the railway.

After obtaining the 3D point cloud of the railway track and its surroundings, in order to test the actual accuracy of the 3D point cloud generated by the system, this paper uses the absolute accuracy index of the scanned point cloud to evaluate that accuracy as shown in Figure 18. For absolute accuracy, this paper selects 23 identification points and uses the method of total station measurement to measure the coordinates, and the measurement accuracy can reach the millimeter level. By comparing the coordinates of the identified points in the corresponding 3D point cloud, the corresponding coordinate difference and statistical difference are obtained. Table 2 lists the coordinate errors of 23 orbital points. The average error of the absolute value in the north coordinate direction of the identification point is 0.86 cm, the maximum difference is 4.6 cm, and the standard deviation is 1.65. The average error of the absolute value in the east coordinate direction is 1.34 cm, the maximum difference is 6.9 cm, and the standard deviation is 2.95. The average error of the absolute value in the elevation coordinate direction is 2.39 cm, the maximum difference is 6.33 cm, and the standard deviation is 1.77. The point error of the system is 1.6 cm, which proves that the system's overall accuracy is high.

Table 2. Point cloud accuracy.

Point Name	ΔN (cm)	ΔE (cm)	ΔH (cm)
JD01	1.20	−0.50	−0.17
JD02	1.00	−6.90	1.13
JD03	−1.80	−1.60	1.73
JD04	0.40	−3.20	0.03
JD05	−0.30	−6.80	0.03
JD06	−0.50	3.10	5.63
JD07	1.30	−1.20	3.73
JD08	−0.30	−4.55	3.78

Table 2. *Cont.*

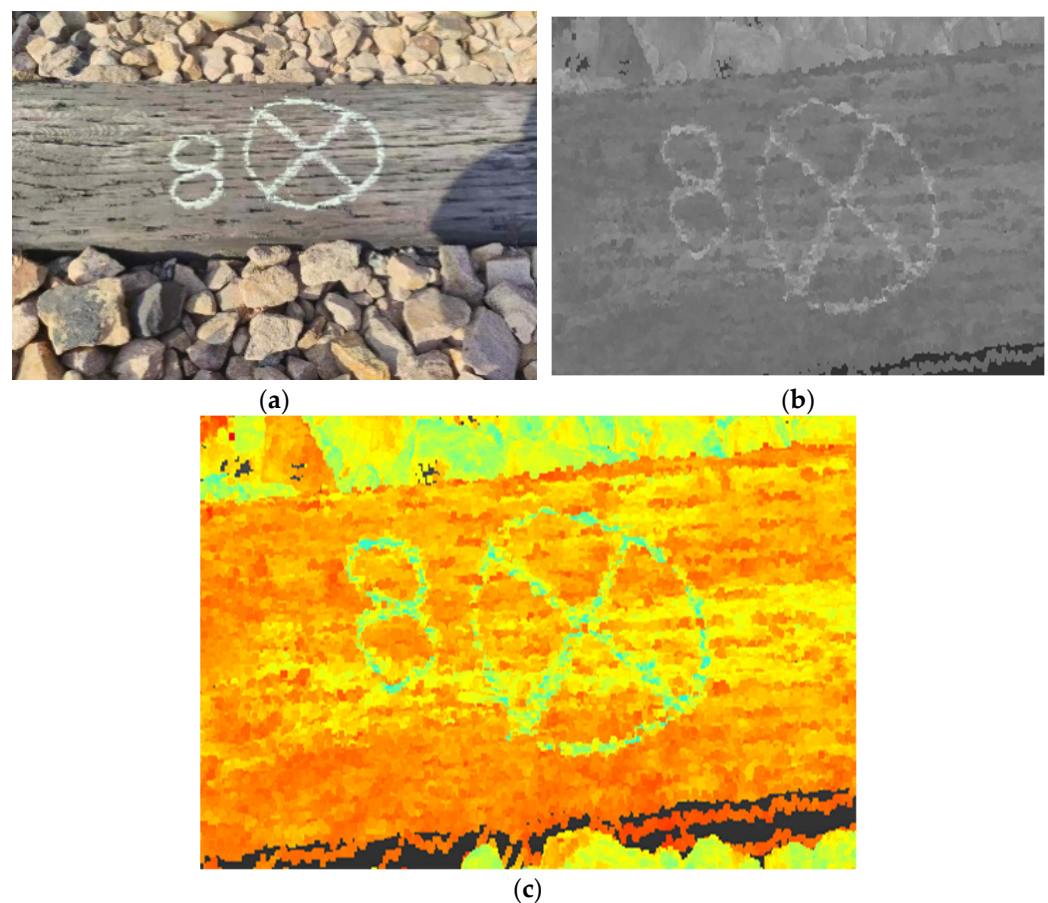| Point Name	ΔN (cm)	ΔE (cm)	ΔH (cm)
JD09	−1.50	1.10	1.23
JD10	0.80	−2.85	3.13
JD11	0.30	−5.10	3.38
JD12	−1.50	2.35	0.48
JD13	1.30	−0.60	3.98
JD14	1.60	−3.40	3.43
JD15	−0.65	−1.40	3.53
JD16	−0.30	2.70	1.93
JD17	1.75	1.45	3.13
JD18	3.10	−3.60	6.33
JD19	2.10	−2.30	0.83
JD20	1.20	2.10	0.43
JD21	2.10	0.70	1.83
JD22	4.60	1.60	3.23
JD23	3.90	−2.00	2.03
Average Deviation	0.86	−1.34	2.38
Standard Deviation	1.65	2.95	1.77

Figure 18. Control point signs in different images: (**a**) Real scene image; (**b**) Grayscale image; (**c**) Intensity scale image.

5. Discussion

In this study, the deep learning method is used to extract the track feature points, the Lagrange interpolation is used to complete the missing feature points, and the speed filtering method is used to optimize the mileage in segments. In an interval of 20 meters, the mileage accuracy is controlled within 1 decimeter so as to reduce the influence of the

cumulative error caused by the odometer as much as possible. In the traditional method, it is necessary to reduce the influence of the cumulative error caused by the slippage of the odometer by increasing the fixing device between the trolley and the track. Still, this method will gradually increase the error over time and reduce the work efficiency. At present, the target detection algorithm used in this paper is YOLOv3, and more advanced target detection algorithms such as YOLOv7 can be used in the future to obtain better recognition accuracy and speed of track feature points.

In this paper, the RTS smoothing algorithm is used to smooth the dead reckoning results, and the experiments show that smoothing has a great effect on improving the accuracy of post-processing. Due to the complex and changeable railway environment, the GNSS signal is often blocked. At this time, it is necessary to use an appropriate method to compensate for the drift of the inertial navigation system. In addition to the smoothing algorithm proposed in this paper, the commonly used method of dynamic zero-speed correction can only slow down the drift speed of the error but cannot completely limit the drift of the error. Therefore, in order to obtain better trajectory results in the subterranean environment, it is necessary to use higher-precision IMU in the future and continue to study and optimize the proposed IMU trajectory error model to improve the practicability of the system.

In the future, we will add visual sensors and simultaneous localization and mapping (SLAM) algorithms to optimize the trajectory to adapt to complex railway environments such as mountain tunnels. At the same time, the conversion relationship between the measurement coordinates and point cloud coordinates based on the control points is studied, and the trajectory data is further corrected by segmental fitting to obtain a higher-precision railway three-dimensional point cloud.

6. Conclusions

In this study, we adopt a mileage correction algorithm based on visual feature points and use deep learning to detect objects in the generated orthophoto. Then, RTS filtering is used to optimize the dead reckoning trajectory, and the real motion trajectory of the mobile carrier is restored. The multi-source data fusion method is used to obtain the railway point cloud data in the absolute coordinate system.

The experimental results show that the mileage correction method has good positioning accuracy and stability, and the mileage positioning error is within about 6 cm, which provides accurate mileage results for trajectory calculation. The 3D point cloud data is generated by combining the trajectory generated by the RTS filtering algorithm and the scanning data. The maximum errors of the 23 coordinate points measured are all within 7 cm. The average error of the point position is 1.60 cm, and the average error of the elevation is 2.39 cm.

Different from the traditional method that uses manual detection of railways, this paper proposes a new method using the combination of calibrated mileage and inertial navigation data. The mileage is calibrated segmentally using visual feature points, which avoids multiple stops of the mobile laser scanning system for calibration. It reduces the influence of the trajectory error increasing with time in the traditional method. There is no relevant research on this method yet.

The measurement method combined with laser scanning and inertial navigation can quickly obtain the real 3D scene point cloud in the absolute coordinate system. The speed and amount of information at the measuring point far exceed traditional methods. In the future, according to the point cloud data, we can complete the extraction of railway elements, including the track centerline, contact wire, and catenary wire. Moreover, 3D models of railway buildings can be built from point cloud data. At the same time, high-precision point cloud data can help us to complete the information of infrastructure along the route quickly. It is of great significance in railway digital survey and design, intelligent operation and maintenance, and digital transformation.

References

1. Lee, J.; Choi, H.; Park, D.; Chung, Y.; Kim, H.-Y.; Yoon, S. Fault detection and diagnosis of railway point machines by sound analysis. *Sensors* **2016**, *16*, 549. [CrossRef] [PubMed]
2. Liu, S.; Lu, Q.; Li, H.; Wang, Y. Estimation of moisture content in railway subgrade by ground penetrating radar. *Remote Sens.* **2020**, *12*, 2912. [CrossRef]
3. Jang, J.; Shin, M.; Lim, S.; Park, J.; Kim, J.; Paik, J. Intelligent image-based railway inspection system using deep learning-based object detection and weber contrast-based image comparison. *Sensors* **2019**, *19*, 4738. [CrossRef] [PubMed]
4. Huang, Q.; Wang, Y.; Luzi, G.; Crosetto, M.; Monserrat, O.; Jiang, J.; Zhao, H.; Ding, Y. Ground-based radar interferometry for monitoring the dynamic performance of a multitrack steel truss high-speed railway bridge. *Remote Sens.* **2020**, *12*, 2594. [CrossRef]
5. Specht, C.; Wilk, A.; Koc, W.; Karwowski, K.; Dąbrowski, P.; Specht, M.; Grulkowski, S.; Chrostowski, P.; Szmagliński, J.; Czaplewski, K.; et al. Verification of GNSS measurements of the railway track using standard techniques for determining coordinates. *Remote Sens.* **2020**, *12*, 2874. [CrossRef]
6. Zou, R.; Fan, X.; Qian, C.; Ye, W.; Zhao, P.; Tang, J.; Liu, H. An Efficient and accurate method for different configurations railway extraction based on mobile laser scanning. *Remote Sens.* **2019**, *11*, 2929. [CrossRef]
7. Elberink, S.O.; Khoshelham, K. Automatic extraction of railroad centerlines from mobile laser scanning data. *Remote Sens.* **2015**, *7*, 5565–5583. [CrossRef]
8. Du, L.; Zhong, R.; Sun, H.; Liu, Y.; Wu, Q. Cross-section positioning based on a dynamic MLS tunnel monitoring system. *Photogramm. Rec.* **2019**, *34*, 244–265. [CrossRef]
9. Han, Y.; Sun, H.; Zhong, R. Three-dimensional linear restoration of a tunnel based on measured track and uncontrolled mobile laser scanning. *Sensors* **2021**, *21*, 3815. [CrossRef]
10. Hruboš, M.; Janota, A. 3D surface modeling based on data from the mobile measurement platform. In Proceedings of the 2014 IEEE 12th International Symposium on Applied Machine Intelligence and Informatics (SAMI), Herl'any, Slovakia, 23–25 January 2014; pp. 39–43.
11. Zhou, S.; Song, W. Deep learning-based roadway crack classification using laser-scanned range images: A comparative study on hyperparameter selection. *Autom. Constr.* **2020**, *114*, 103171. [CrossRef]
12. Sturari, M.; Paolanti, M.; Frontoni, E.; Mancini, A.; Zingaretti, P. Robotic platform for deep change detection for rail safety and security. In Proceedings of the 2017 European Conference on Mobile Robots (ECMR), Paris, France, 6–8 September 2017; pp. 1–6.
13. Stein, D.; Spindler, M.; Lauer, M. Model-based rail detection in mobile laser scanning data. In Proceedings of the 2016 IEEE Intelligent Vehicles Symposium (IV), Gothenburg, Sweden, 19–22 June 2016; pp. 654–661.
14. Cui, H.; Ren, X.; Mao, Q.; Hu, Q.; Wang, W. Shield subway tunnel deformation detection based on mobile laser scanning. *Autom. Constr.* **2019**, *106*, 102889. [CrossRef]
15. Boavida, J.; Oliveira, A.; Santos, B. Precise long tunnel survey using the Riegl VMX-250 mobile laser scanning system. In Proceedings of the 2012 RIEGL International Airborne and Mobile User Conference, Orlando, FL, USA, 28 February–1 March 2012.
16. Li, Q.; Chen, Z.; Hu, Q.; Zhang, L. Laser-aided INS and odometer navigation system for subway track irregularity measurement. *J. Surv. Eng.* **2017**, *143*, 04017014. [CrossRef]
17. Sun, H.; Xu, Z.; Yao, L.; Zhong, R.; Du, L.; Wu, H. Tunnel monitoring and measuring system using mobile laser scanning: Design and deployment. *Remote Sens.* **2020**, *12*, 730. [CrossRef]
18. Chen, C.; Zhang, T.; Kan, Y.; Li, S.; Jin, G. A rail extraction algorithm based on the generalized neighborhood height difference from mobile laser scanning data. In Proceedings of the SPIE Future Sensing Technologies, Online, 9–13 November 2020; pp. 105–124.
19. Bobkowka, K.; Nykiel, G.; Tysiąc, P. DMI measurements impact on a position estimation with lack of GNSS signals during Mobile Mapping. *J. Phys. Conf. Ser.* **2017**, *870*, 12010. [CrossRef]
20. Specht, C.; Koc, W.; Chrostowski, P.; Szmagliński, J. Accuracy assessment of mobile satellite measurements in relation to the geometrical layout of rail tracks. *Metrol. Meas. Syst.* **2019**, 309–321.
21. Kaartinen, H.; Kukko, A.; Hyyppä, J.; Jaakkola, A. Benchmarking mobile laser scanning systems using a permanent test field. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2012**, *39*, B5.
22. Tsujimura, T.; Manabe, T. Electromagnetic system navigating tunneling robots. *Robot. Auton. Syst.* **2002**, *40*, 229–238. [CrossRef]

23. Jing, N.; Ma, X.; Guo, W.; Wang, M. 3D Reconstruction of underground tunnel using depth-camera-based inspection robot. *Sens. Mater.* **2019**, *31*, 2719. [CrossRef]

24. Pfeifer, N.; Briese, C. Laser scanning–principles and applications. In Proceedings of the GeoSiberia 2007-International Exhibition and Scientific Congress, Novosibirsk, Russia, 25 April 2007; p. cp-59-00077.

25. Boronahin, A.; Larionov, D.Y.; Podgornaya, L.; Shalymov, R.; Filatov, Y.V.; Bokhman, E. Specialized navigation system for rail track diagnostics. In Proceedings of the 2016 IEEE NW Russia Young Researchers in Electrical and Electronic Engineering Conference, St. Petersburg, Russia, 2–3 February 2016; pp. 401–403.

26. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767 2018.

27. Zhao, L.; Li, S. Object detection algorithm based on improved YOLOv3. *Electronics* **2020**, *9*, 537. [CrossRef]

28. Yi, Z.; Yongliang, S.; Jun, Z. An improved tiny-yolov3 pedestrian detection algorithm. *Optik* **2019**, *183*, 17–23. [CrossRef]

29. Yu, J.; Zhang, W. Face Mask Wearing Detection Algorithm Based on Improved YOLO-v4. *Sensors* **2021**, *21*, 3263. [CrossRef] [PubMed]

30. Yang, G.; Feng, W.; Jin, J.; Lei, Q.; Li, X.; Gui, G.; Wang, W. Face mask recognition system with YOLOV5 based on image recognition. In Proceedings of the 2020 IEEE 6th International Conference on Computer and Communications (ICCC), Chengdu, China, 11–14 December 2020; pp. 1398–1404.

31. Randell, C.; Djiallis, C.; Muller, H. Personal position measurement using dead reckoning. In Proceedings of the Seventh IEEE International Symposium on Wearable Computers, White Plains, NY, USA, 21–23 October 2003; p. 166.

32. Li, Q.; Li, R.; Ji, K.; Dai, W. Kalman filter and its application. In Proceedings of the 2015 8th International Conference on Intelligent Networks and Intelligent Systems (ICINIS), Tianjin, China, 1–3 November 2015; pp. 74–77.

33. Wang, Y. Position estimation using extended Kalman filter and RTS-smoother in a GPS receiver. In Proceedings of the 2012 5th international congress on image and signal processing, Chongqing, China, 16–18 October 2012; pp. 1718–1721.

34. Murata, M.; Kashino, K. Normalized unscented Kalman filter and normalized unscented RTS smoother for nonlinear state-space model identification. In Proceedings of the 2013 American Control Conference, Washington, DC, USA, 17–19 June 2013; pp. 5462–5467.

35. Sun, M.; Gao, Y.; Jiao, Z.; Xu, Y.; Zhuang, Y.; Qian, P. R-T-S assisted kalman filtering for robot localization using UWB measurement. *Mob. Netw. Appl.* **2022**, *27*, 1–10. [CrossRef]