Intelligent navigation algorithm of plant phenotype detection robot based on dynamic credibility evaluation

Wei Lu[1*], Mengjie Zeng[2], Huanhuan Qin[1]

(1. *College of Artificial Intelligence, Nanjing Agricultural University, Nanjing 210031, Jiangsu, China*;
2. *College of Energy and Power Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, Jiangsu, China*)

Abstract: Due to the non-standardization and complexity of the farmland environment, Global Navigation Satellite System (GNSS) navigation signal may be affected by the tree shade, and visual navigation is susceptible to winged insect and mud, which makes the navigation information of the plant phenotype detection robot unreliable. To solve this problem, this study proposed a multi-sensor information fusion intelligent navigation algorithm based on dynamic credibility evaluation. First, three navigation methods were studied: GNSS and Inertial Navigation System (INS) deep coupling navigation, depth image-based visual navigation, and maize image sequence navigation. Then a credibility evaluation model based on a deep belief network was established, which used dynamically updated credibility to intelligently fuse navigation results to reduce data fusion errors and enhance navigation reliability. At last, the algorithm was loaded on the plant phenotype detection robot for experimental testing in the field. The result shows that the navigation error is 2.7 cm and the navigation effect of the multi-sensor information fusion method is better than that of the single navigation method in the case of multiple disturbances. The multi-sensor information fusion method proposed in this study uses the credibility model of the deep belief network to perform navigation information fusion, which can effectively solve the problem of reliable navigation of the plant phenotype detection robot in the complex environment of farmland, and has important application prospects.

Keywords: plant phenotype detection, robot, dynamic credibility evaluation, intelligent navigation, multi-sensor information fusion

DOI: 10.25165/j.ijabe.20211406.6615

1　Introduction

The combination of intelligent navigation technology and agricultural machinery has improved agricultural farming methods. In terms of agricultural machinery navigation, Global Navigation Satellite System (GNSS) navigation[1], inertial navigation[2], visual navigation[3], Light Detection And Ranging (LiDAR)[4], and other navigation methods are popular nowadays. Many related studies have confirmed the feasibility of these algorithms in farmland[5]. The GNSS navigation method has stable positioning data and accurate navigation in a wide and standardized area[6,7], but its signal is susceptible to the interference of the surrounding environment in a complex environment, which affects the positioning effect[8]. In addition, the Inertial Navigation System (INS) has a delay in the actual operation process[9,10], and there is accumulated navigation in long-term operation, so the GNSS system and the INS system are often used in combination. Miguel et al.[11] conducted localization methods for autonomous vehicles to improve laser information to enhance the accuracy of LiDAR and

GNSS/INS combined positioning system and make up for part of the outdoor environment limitations of GNSS/INS combined navigation. Although it can improve positioning accuracy but significantly increases the cost. Because the hardware of visual navigation is low cost and the algorithm is diversified, the excellent visual algorithm can achieve high recognition accuracy in a suitable environment. Kovacs et al.[12] proposed a space hopping robot navigation method based on monocular vision, which can realize the accurate estimation of the hopping distance. However, the actual farmland operating environment is complex and unpredictable, such as uneven field surface, vibration, winged insect, and dust, etc., which affect the acquisition of visual information. Therefore, an auxiliary algorithm is generally added to the visual algorithm in the practical application to ensure the stability of the actual work. Ma et al.[13] proposed an autonomous navigation algorithm for wolfberry orchard based on visual cues and fuzzy control to achieve high precision, good robustness, and real-time performance, which made up for errors caused by single visual navigation that may be based on jitter, etc. LiDAR is widely used in farmland navigation because it can scan the surrounding environment to obtain distance information quickly and conveniently with little computation, but the cost is relatively high. Velasquez et al.[14] developed the maize crop reactive navigation system based on the front LiDAR sensor combined with H∞ robust controller, which experimental results showed that the proposed navigation system can control the robot's displacement between crop rows. In addition to conventional navigation methods, new navigation methods are also being studied. Huang et al.[15] proposed an evaluation positioning and orientation system based on sound signals. Experiments have obtained high positioning and

orientation accuracy in the greenhouse, indicating that this spread-spectrum sound-based system has the potential to be deployed on small distributed greenhouse robots, but the application in outdoor fields requires further research and application verification.

As a matter of fact, in the actual complex conditions of the agricultural field, all the sensors have the possibility of short-time huge error or even failure. The accuracy of the GNSS navigation technology is affected by agricultural machine tilt angle, meteorological conditions, and satellite information intensity. The agricultural machine tilt is randomly caused by changeable cropland bump and soil fluidity. And the stability of the GNSS signal is vulnerable to meteorological conditions and tree shades. What's more, it may be temporarily interrupted in the case of sudden thunderstorms and dense shade[16], etc. So, the inertial navigation technology is used to compensate GNSS because of its short-time high precision navigation capability, but the navigation accuracy is easily affected by the shaking caused by the strong vibration of agricultural machinery and the bumpy field surface. The visual navigation method is suitable for any complex and changeable environment based on powerful algorithms with low cost but the sensor may be covered by wing insects and bugs which will lead to navigation error or even failure. So, the information of the short-time failure sensors used for field navigation should be ignored during the period of failure, otherwise, it will lead to huge errors.

Be aimed at hereat, multi-sensor information fusion methods for agricultural machinery navigation in farmland environments have been proposed aiming at improving navigation accuracy using low-cost sensors and information fusion algorithms. Mahmud et al.[17] integrated the camera and GPS receiver on a small autonomous field mobile detection robot independently developed to obtain autonomous full-coverage crop field information. The experiment showed that the fusion navigation technology can realize the reconnaissance mission of crops and soil and minimize the impact of robots on farmland. Jasinski et al.[18] proposed the concept of autonomous robots for seeding and wide-row planting and based on the data of multiple sensors (camera, position, and distance) to realize autonomous walking and job detection in farmland. Alberto-Rodriguez et al.[19] mentioned a smart tractor equipped with sensors and cameras, which intelligently established the most effective route based on terrain and obstacles. Aghi et al.[20] proposed a low-cost, low-power local exercise planner, which was based on RGB-D cameras, low-range hardware, and double-layer controls. Combined with the deep learning model, it can intelligently switch the use of algorithms to realize the autonomous and accurate navigation of the exerciser in the vineyard. The existed intelligent navigation research, as described in Reference [21], shows that the multi-sensor combined with an information fusion algorithm can improve navigation precision effectively at a low hardware cost. Unfortunately, most of the existed multi-sensor information fusion navigation algorithms have fixed weight values and thresholds which lead to poor work quality or even danger in the unstructured agricultural field.

To overcome the short-time dynamic error of different sensors caused by complex environments in farmland, this study proposes an intelligent navigation algorithm based on dynamic reliability assessment which is also verified by a plant phenotype detection robot in the maize field. The method combines a variety of navigation sensors such as GNSS, INS, RGB-D imaging by adjusting the weight values dynamically according to the credibility

of each sensor to make up for the defects of every single sensor detection, aiming at realizing the accurate navigation of farmland operation.

2   Materials and methods

2.1   Framework of platform and algorithm

The platform used in the study for algorithm study is a plant phenotypic detection robot with a variable structure developed by the laboratory of the authors' team, which has a unique "zero turning radius" design that can be applied in the narrow workspace and variable crop rows flexibly, shown in Figure 1a. The depth camera (Intel RealSense D435, 1280×720 pixels resolution, 0.28-10.00 m effective range, made by Intel Corporation, USA), Huatest CGI-410 integrated navigation system (Network RTK coupled with INS, Precision 10 mm, made by Shanghai Huace Navigation Technology Ltd., China) and an industrial computer (64-bit, Win 10 operating system, 16 GB memory, dual-core processor Intel Core i5-4200H) were installed in the robot, as shown in Figure 1.

a. Plant phenotypic detection robot

b. Intel D435   c. GNSS signal receiver

Figure 1   The platform and sensors of plant phenotypic detection robot

Due to intermittent failures such as short-term loss of GNSS signal sources, unstable INS affected by jitter, or short-term camera signal failure caused by strong light and other reasons will occur during the long-term operation of the sensors, which directly lead to large navigation error or even failure. For this matter, an intelligent navigation technology based on dynamic credibility evaluation was proposed for improving the stability and robustness of the traditional navigation methods.

Firstly, GNSS/INS integrated navigation based on deep coupling was employed to overcome short-time satellite signal loss caused by tree shade or severe weather. Depth images of crops were applied for crop row navigation to reduce green weed interference. Matching feature points of the image sequence was used to reduce the calculation amount for fast navigation.

Second, the credibility of each sensor was calculated by the credibility model based on the deep belief network, then input to the multi-sensor information fusion model together with the navigation information of all the sensors. To reduce or even zero the weight of the error navigation information source to avoid great deviation and make the data fusion result more reliable. The overall frame diagram is shown in Figure 2.
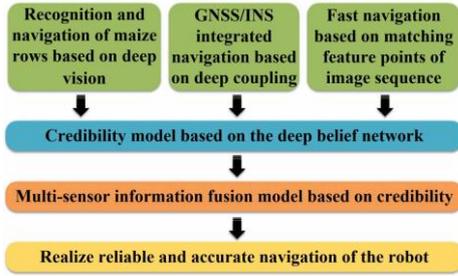
Note: GNSS: Global Navigation Satellite System; INS: Inertial Navigation System.

Figure 2   The overall frame diagram of intelligent navigation algorithm

2.2   GNSS/INS integrated navigation based on deep coupling

Because of parallel rows of maize seedlings in the field, although row distance varies with maize varieties and planting strategies, navigation along the maize seeding row is the main task during phenotypic detection in the field.   So, the popular navigation approach of GNSS/INS integrated navigation was employed.   To make full use of the navigation performance of GNSS and INS, the deep coupling method was utilized to improve the integrated navigation of GNSS/INS compared with the traditional loose coupling method[22].   The principle diagram of the deep coupling method is shown in Figure 3.

As shown in Figure 3, the satellite receiver receives the signal through the antenna, and then the satellite ephemeris calculates the pseudo-range and doppler frequency shift of these original signals.   The velocity and position of the robot are calculated by the INS solving unit to predict the values of the pseudo-range and Doppler frequency shift combined with the values calculated by the satellite ephemeris.   The predicted values and the measured values of pseudo-range and Doppler frequency shift are subtracted to obtain the calculation error after the data fusion filter.   Finally, the optimal estimate values of velocity and position are obtained by subtracting the calculated errors of INS from the calculated values of INS.

Note: "+" and "−" on the line in the figure indicate the positive and negative of the signal, and "+" in the circle indicates the signal superimposition process.

Figure 3   Schematic diagram of the deep coupling

To achieve a stable and reliable filtering effect, the Kalman filter was selected as the data fusion filter.   The filtering algorithm can realize the optimal estimation of discrete-time linear systems which is widely used in many fields of signal processing.   Kalman filtering has two main steps for data fusion filtering including prediction and correction.   Firstly, the measurement prediction quantity of the next moment is derived through the system state equation according to the previous state quantity, and then the actual measured value is used to adjust the predicted value.

In the deep-coupled navigation system[23], the equation of state of the system is

$$x_k = \boldsymbol{\Psi}_{k/k-1}\boldsymbol{x}_{k-1} + \boldsymbol{B}_{k/k-1}\boldsymbol{e}_{k-1} + \boldsymbol{w}_{k-1} \tag{1}$$

where, $\boldsymbol{x}_k$ is the state vector for the k state; $\boldsymbol{\Psi}_{k/k-1}$ is the state transition matrix; $\boldsymbol{B}_{k/k-1}$ presents the input relational matrix; $\boldsymbol{w}_{k-1}$ is the noise vector; $\boldsymbol{e}_{k-1}$ is the external input vector.   For the deep coupling system, the value $\boldsymbol{e}_{k-1}$ is generally negligible and considered to be 0.   The measurement equation of the system is

$$y_k = \boldsymbol{H}_k\boldsymbol{x}_k + \boldsymbol{v}_k \tag{2}$$

where, y_k represents the observation and measurement of the system; $\boldsymbol{H}_k$ refers to the observation matrix of the system; $\boldsymbol{v}_k$ represents the measurement noise vector.   In this system, the observed measurement is obtained by subtracting the predicted value of pseudo-range and Doppler frequency shift from that of measured residual value, shown as the following equations:

$$y_k = [\Delta\rho^i \quad \cdots \quad \Delta\rho^j \quad \Delta f^i \quad \cdots \quad \Delta f^j]^T \tag{3}$$

$$\Delta\rho^i = \boldsymbol{u}_1^i\Delta x + \boldsymbol{u}_2^i\Delta y + \boldsymbol{u}_3^i\Delta z + \Delta t + \varepsilon_\rho^i \tag{4}$$

$$\Delta f^i = \boldsymbol{u}_1^i\Delta\dot{x} + \boldsymbol{u}_2^i\Delta\dot{y} + \boldsymbol{u}_3^i\Delta\dot{z} + \Delta f_i + \Delta f_c + \varepsilon_f^i \tag{5}$$

where, $\boldsymbol{u}_1^i$, $\boldsymbol{u}_2^i$, and $\boldsymbol{u}_3^i$ are the projections of the unit vector of the i-th satellite relative to the user position vector on the x-axis, y-axis, and z-axis, respectively.   The specific values are as follows:

$$\boldsymbol{u}_1^i = \frac{x-x^i}{[(x-x^i)^2 + (y-y^i)^2 + (z-z^i)^2]} \tag{6}$$

$$\boldsymbol{u}_2^i = \frac{y-y^i}{[(x-x^i)^2 + (y-y^i)^2 + (z-z^i)^2]} \tag{7}$$

$$\boldsymbol{u}_3^i = \frac{z-z^i}{[(x-x^i)^2 + (y-y^i)^2 + (z-z^i)^2]} \tag{8}$$

The measurement matrix $\boldsymbol{H}_k$ can be obtained from the above equation.

$$\boldsymbol{H}_k = \begin{bmatrix} \boldsymbol{u}_1^i & \boldsymbol{u}_2^i & \boldsymbol{u}_3^i & 0_{1\times3} & 0_{1\times9} & 1 & 0 & 0 \\ & & & \vdots & & & & \\ \boldsymbol{u}_1^j & \boldsymbol{u}_2^j & \boldsymbol{u}_3^j & \ldots & 0_{1\times9} & 1 & 0 & 0 \\ 0_{1\times3} & \boldsymbol{u}_1^i & \boldsymbol{u}_2^i & \boldsymbol{u}_3^i & 0_{1\times9} & 0 & 1 & 1 \\ & & & \vdots & & & & \\ 0_{1\times3} & \boldsymbol{u}_1^j & \boldsymbol{u}_2^j & \boldsymbol{u}_3^j & 0_{1\times9} & 0 & 1 & 1 \end{bmatrix} \tag{9}$$

2.3   Line recognition and navigation of maize crops based on depth vision

Based on the depth image obtained by the depth camera, the maize plants can be effectively distinguished in a pure green environment or an environment with a wide variety of weeds[24].   First, the acquired depth image and color image are converted into the same coordinate system, and then the maize plant images are segmented by edge features after denoising processing.   Then the target maize seedlings are used to create a fitting navigation line, and the actual navigation angle will be obtained by conversion and calculation at the same time to realize visual navigation.   The

complete algorithm flow chart is shown in Figure 4.
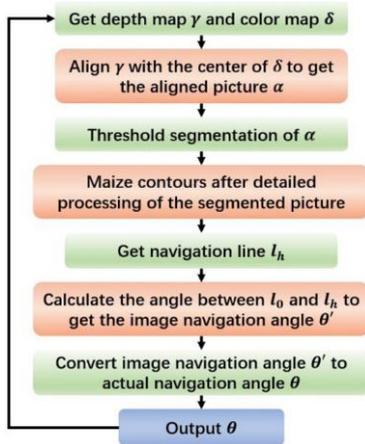
Figure 4   Flow chart of RGB-D algorithm

2.3.1   Align

Since the imaging center points of the depth images and the RGB images are different, so deviation or even errors will be caused if the depth image is directly used for navigation processing[25].   Therefore, aligning the process of depth image and RGB image is needed to transfer them into the same coordinate system before navigation angle calculation.   First, restore the two-dimensional coordinate pixel point $P_{u,v}^d$ to the spatial point $\dot{P}_{dc}$ in the depth coordinate system.

$$\dot{P}_{dc} = Z\boldsymbol{K}_d^{-1}P_{u,v}^d \tag{10}$$

where, $\boldsymbol{K}_d$ is the depth camera parameter matrix; Z is the depth value.   Then obtain the Euclidean transformation matrix transformed from $\dot{P}_{dc}$ to space point $\dot{P}_{cc}$ in the color map coordinate system.

$$\dot{P}_{cc} = \boldsymbol{T}_{w2c}P_w = \boldsymbol{T}_{w2c}\boldsymbol{T}_{w2d}^{-1}\dot{P}_{dc} \tag{11}$$

$$\boldsymbol{T}_{d2c} = \boldsymbol{T}_{w2c}\boldsymbol{T}_{w2d}^{-1} = \begin{bmatrix} R_{w2c}R_{w2d}^{-1} & t_{w2c} - R_{w2c}R_{w2d}^{-1}t_{w2d} \\ 0 & 1 \end{bmatrix} \tag{12}$$

where, $\boldsymbol{T}_{w2c}$ is the external parameter matrix of the world coordinate system to the color coordinate system; $\boldsymbol{T}_{w2d}$ is the external parameter matrix of the world coordinate system to depth coordinate system; $\boldsymbol{T}_{d2c}$ is the external parameter matrix of depth coordinate system to color coordinate system.   Finally, map $\dot{P}_{cc}$ to the color plane $Z=1$ to complete the alignment.

$$P_{u,v}^c = K_c(\dot{P}_{cc}/z) \tag{13}$$

where, K_c is the parameter moment of the depth camera; $\dot{P}_{cc}/z$ is normalized by the z-axis, that is, the three-axis coordinates of the point are divided by the z value.

The depth information is displayed in a pseudo-color because it is inconvenient for naked-eye observation.   In this study, a clear indoor outline was selected for illustration because the alignment outline of maize seedlings in the field is less obvious.   The depth information displayed on the color image is shown in Figure 5, which matches well with the color image.

Figure 5   Image example after aligning the color map with the depth map

2.3.2   Denoising

The depth camera was installed directly below the plant phenotype detection robot for acquiring depth images.   Since the image is affected by external factors such as hardware equipment and environmental conditions during image acquisition, filters such as grayscale, Butterworth filtering, Average filtering, Median filtering, and Homomorphic filtering were applied for evaluating the filtering effects.

The filtering effect of different filter various a lot shown in Figure 6.   The classic Butterworth high pass filter can magnify the detailed edge information, but darken the whole image overall, moreover, the edge information in the dark area is weakened, shown in Figure 6c.   The average filter can denoise the image effectively but blur the image, the edge features were reduced, shown in Figure 6d.   Homomorphic filtering is a method to compress the image brightness range and enhance the image contrast at the same time in the frequency domain which can solve the problem of additive noise flexibly.   In Figure 6e, it can enhance the edge information in the dark area distinctly but darken the whole image and weaken the edge information in the bright area.   Median filtering is a nonlinear technique that can protect the edge details of the image well while removing the noise and is suitable for eliminating the random noise in the target image.   It is used in many fields widely, e.g., image enhancing and restoration.   As shown in Figure 6f, compared with the grayscale image, the processed image has an obvious noise elimination effect, and the edge information is refined without loss, and the edge details in the dark image can be completely retained.   So, the Median Filter was selected for denoising by comparing the denoising effects of the above algorithms.

a. Original   b. Grayscale   c. Butterworth filtering   d. Average filtering   e. Homomorphic filtering   f. Median filtering
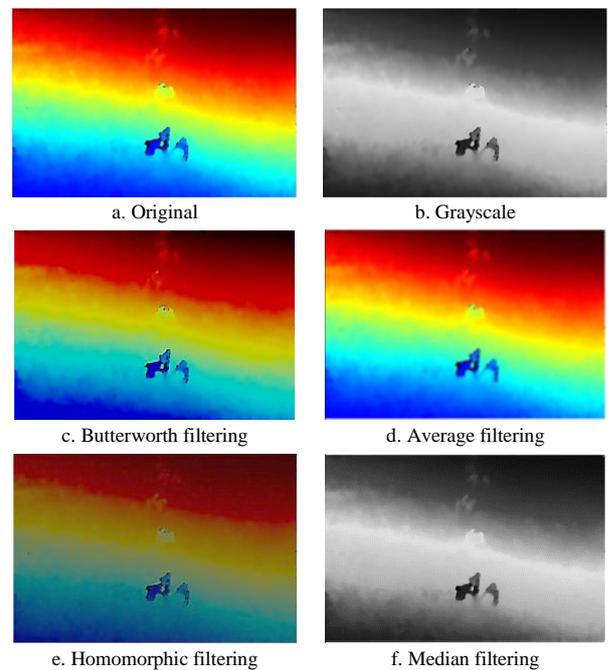
Figure 6   Comparison of different filtering methods

2.3.3   Navigation line extraction

After the image denoising, the appropriate method should be selected to obtain the seedling features.   Due to the depth images of seedlings are significantly different from that of weeds and other backgrounds compared with a color image, the navigation line is extracted using a depth image[26,27].   The edge extraction algorithms such as Sobel, Roberts, Prewitt, Canny, and Log were selected for comparing extraction effects.   The processing effect comparison of classical edge operators such as Sobel edge

detection operator[28], Roberts edge detection operator[29], Prewitt edge detection operator[30], Log edge detection operator[31], and Canny edge detection operator[32] is shown in Figure 7.   Among these detection operators, the processing results of Sobel edge operator, Roberts edge operator, and Prewitt edge operator show good continuity of edge features, less noise in the figure, and obvious and detailed edge features, but less obvious edge information in the original figure is lost and the edge information is not complete; Canny edge operator extracted a large range of edge information because the algorithm smoothed the image in its calculation, shown in Figure 7e.   The content of edge information detected by the Log edge operator has more noise points, but is relatively complete with good continuity, in Figure 7f.   So, the Log operator was selected for navigation line extraction and followed by further filtering to obtain the optimal effect.

a. Original

b. Sobel

c. Roberts

d. Prewitt

e. Canny

f. Log
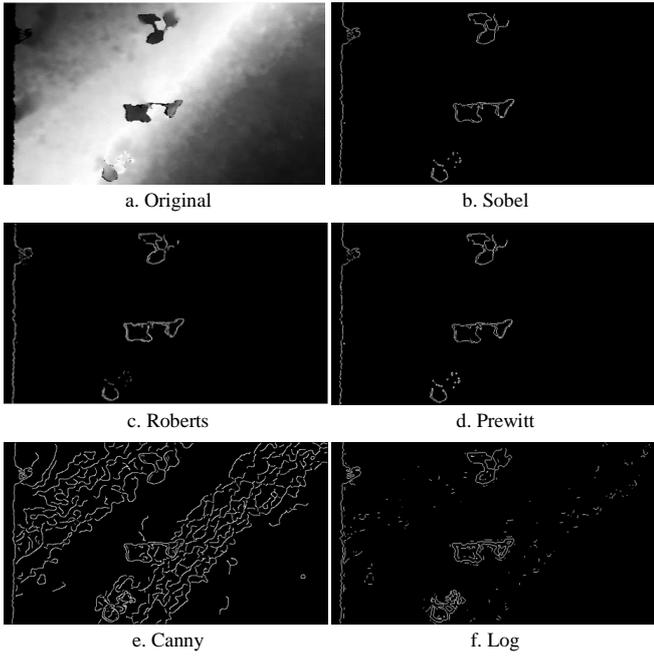
Figure 7   Comparison of different edge operators

LOG edge extraction is mainly divided into three steps, filtering, enhancement, and detection[31].   First, use a two-dimensional Gaussian filter $G(x, y)$ to smoothly convolve (*) the image $\alpha(x, y)$ to obtain a smooth image $I(x, y)$.

$$G(x,y) = \frac{1}{2\pi\sigma^2}\exp(-\frac{x^2+y^2}{2\sigma^2}) \tag{14}$$

$$I(x,y) = G(x,y)*\alpha(x,y) \tag{15}$$

Then use Laplacian ∇^2 to obtain the second-order directional derivative image $N(x, y)$ of $I(x, y)$.

$$\begin{aligned} N(x,y) &= \nabla^2\{I(x,y)\} \\ &= \nabla^2[G(x,y)*\alpha(x,y)] \\ &= [\nabla^2 G(x,y)]*\alpha(x,y) \end{aligned} \tag{16}$$

where, $\nabla^2 G(x, y)$ is the LOG edge, extraction operator.

Finally, use $\nabla^2 G(x, y)$ to perform a convolution operation on the original gray image, and extract the zero-crossing point as the edge point.   Since the LOG operator is sensitive to noise, supplementary filtering is usually used in the process to solve the problem of accompanying noise in the extraction of edges.

After obtaining the complete outlines of the maize seedlings, the center points are fitted into a line to create the target navigation line.   Part test results are shown in Figure 8, which shows good accuracy.

a. Test chart 1

b. Test chart 2
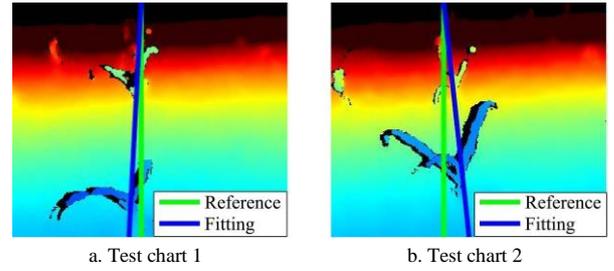
Figure 8   Examples of Depth visual navigation recognition diagram

2.3.4   Deviation angle calculation

Because of the visual angle distortion caused by the camera, there was a visual error between the deviation angle in the image with the actual deviation angle, so the image deviation angle could not be identified as the deviation angle directly.   To solve the problem, the proportional substitution method was adopted to convert the direction angle θ' of the image into the actual direction angle θ.   The basic principle is shown in Figure 9.
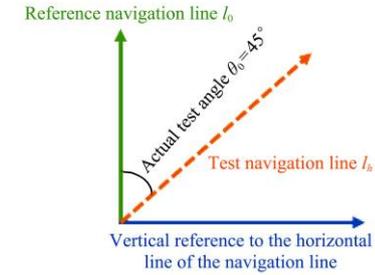
a. Schematic diagram of actual deviation angle
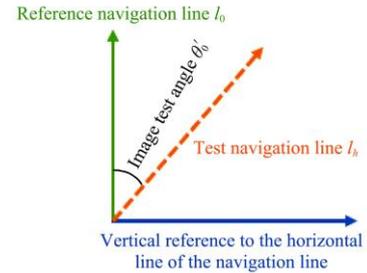
b. Schematic diagram of image deviation angle

Figure 9   Schematic diagram of angle conversion

According to the principle of the diagram, the direction angle conversion equation can be obtained as:

$$\theta = \theta'\cdot k \tag{17}$$

where, k is the conversion proportional coefficient.

The test method is applied to obtain the conversion proportional coefficient k.   Fix the test angle firstly and then get the measurement value many times and calculate the average value θ_0' θ_0' to obtain a reliable fixed conversion proportionality coefficient k according to Equation (18). During testing, the actual angle $\theta_0=45°$ was selected.

$$k = \frac{\theta_0}{\theta_0'} \tag{18}$$

2.4   Image sequence navigation

For the complex farmland environment, a large error of navigation angle may occur only based on depth visual navigation because of high weeds, low maize seedling and camera shaking while image sampling.   To reduce the probability error mentioned above, a navigation algorithm based on image sequence matching is proposed.   Extract the depth images from continuous frames of the video stream, followed by identifying the same feature points of

a seedling, and then calculate the displacement and direction of the robot to realize visual navigation.   The complete algorithm flow chart is shown in Figure 10.
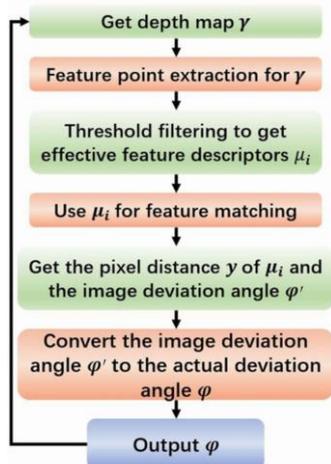
Figure 10   Image sequence navigation algorithm

2.4.1   Feature point extraction

Many computer vision algorithms use feature detection as the initial step, so there are a lot of feature detectors.   Different feature detectors vary greatly in feature detection type, computational complexity, and repeatability.   Currently, several popular feature detectors such as SIFT algorithm[33], SURF algorithm[34], ORB algorithm[35], and so on are used in different domains.   Therefore, the same two consecutive images were employed to evaluate the performance of these algorithms to select the suitable one for navigation.   The comparison results are shown below in Table 1 and Figure 11.

Table 1   Comparison of three feature matching methods

Type	Match time/s	Image distance	Deviation angle/(°)
SURF	0. 1398	15. 1235	6. 7345
SIFT	8. 0654	11. 7763	5. 3856
ORB	0. 1918	13. 535	7. 7117

In terms of time SIFT algorithm takes the most time consuming, more than 8 s, followed by ORB and SURF algorithms only using 0.1918 s and 0.1398 s.   For the performance, SIFT algorithm has the best matching precision than that of other algorithms.   The Deviation angles of SIFT, SURF, and ORB algorithms are 5.3856°, 6.7345°, and 7.7117°, respectively.   So SURF algorithm was selected on comprehensive consideration of time consumption and matching precision.

a. ORB match result

b. SIFT match result

c. SURF match result

Note: Corresponding matched feature points connected in the two images of different frames. More parallel connection lines indicate higher accurate point detection.
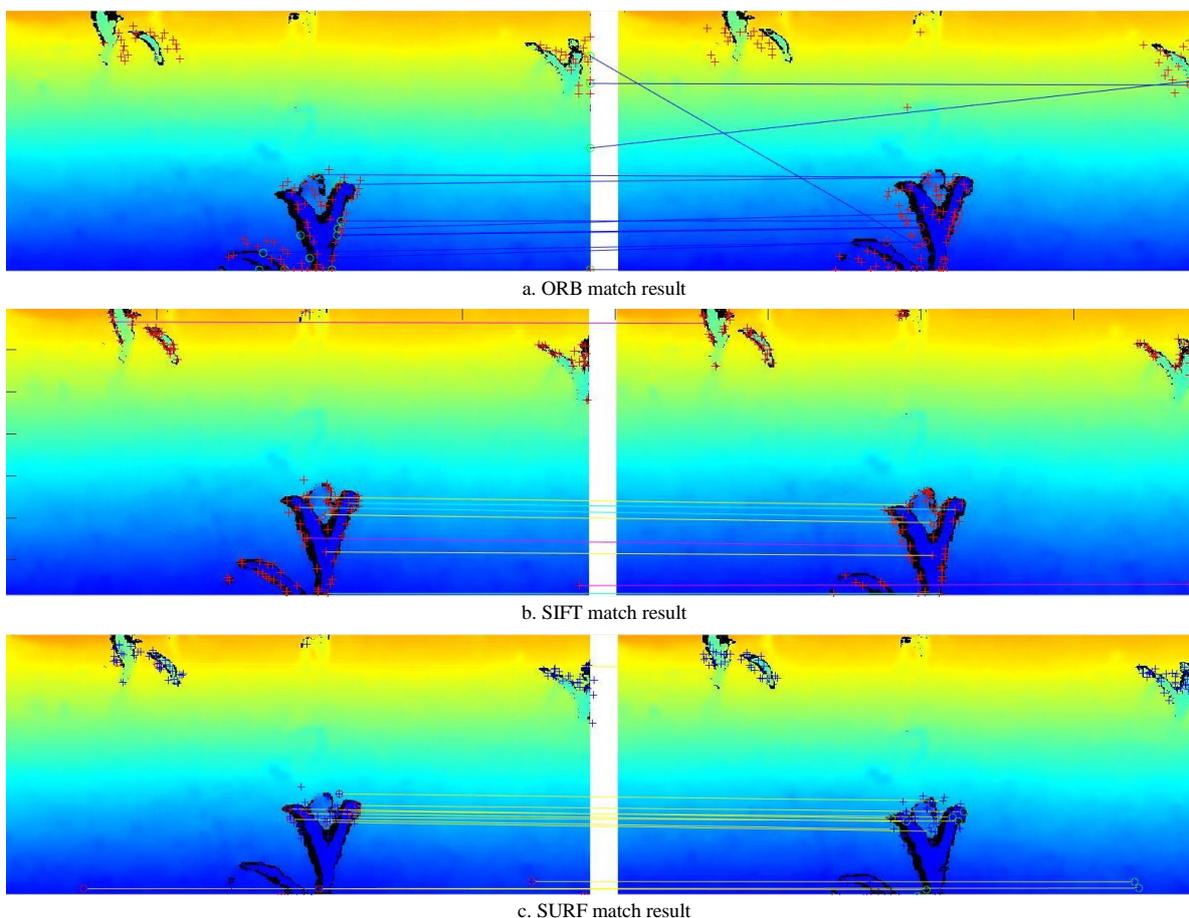
Figure 11   Match results of three feature matching methods

2.4.2   Deflection angle calculation

The SURF feature descriptor is used to describe the feature points, then threshold matching is performed with the feature descriptor of the processed image of the previous frame, and a dozen groups of feature points with the closest features are retained as the effective feature point pair of this algorithm.   According to the coordinates of the effective feature point pairs, as shown in Figure 12, the image distance y and the image deviation angle φ' of the two images can be calculated.   The plus or minus sign of the calculated value can be used to indicate the steering direction is left or right to realize positioning and navigation.

For a pair of feature points $T(x_1, y_1)$ and $T'(x_2, y_2)$ that are successfully matched in two consecutive images, the image distance y represents the moving distance and φ' represents the

deviation angle of movement.

$$y = |T(x_1, y_1) - T'(x_2, y_2)| \qquad (19)$$

$$\varphi' = \arctan \frac{y_1 - y_2}{x_1 - x_2} \qquad (20)$$

When the value φ' is positive, it indicates that the robot body is steering to the right side, and the negative value shows the robot is steering to the left side; if the value φ' is 0, it indicates that the body is moving stably without deviation.   So, the calculated value φ' can be used as the steering angle φ of the robot (Figure 13).
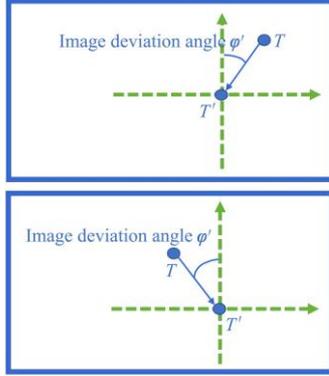
Note: $T(x_1, y_1)$ and $T'(x_2, y_2)$ are a pair of feature points that are successfully matched in two consecutive images.

Figure 12   Plane diagram of image deflection angle

a. Vehicle body diagram   b. Image and vehicle angle
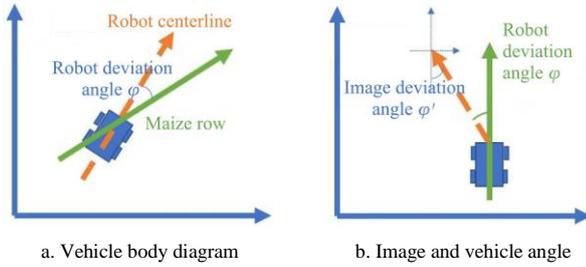
Figure 13   Conversion of vehicle migration angle and image migration angle

2.5 Dynamic prediction of sensor reliability

Because of the complexity and changeability of field working conditions such as uneven ground, shaded by a tree or insect, tall seedlings, or weeds, the sensors may occur short-duration failure during working.   To avoid short time failure of sensors, the Deep Belief Network (DBN) was introduced to calculate the credibility of each sensor, as shown in Figure 14, for further sensor information fusion.
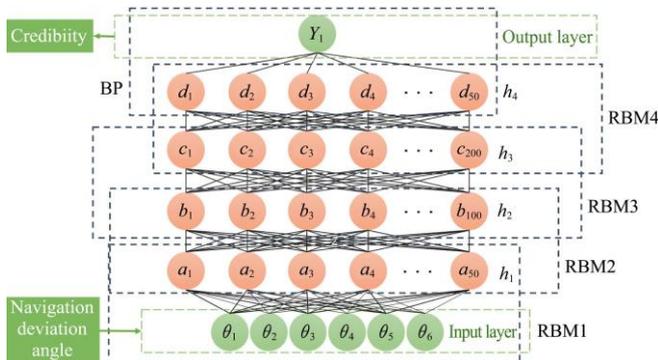
Note: RBM: Restricted Boltzmann Machine; BP: Back Propagation; DBN: Deep Belief Network, the same as below.

Figure 14   The structure of the DBN model used for credibility

2.5.1 Deep belief network model

The main component of the Deep Belief Network is the Restricted Boltzmann Machine (RBM)[36,37].   As shown in Figure 15, a purely restricted Boltzmann machine can be regarded as an undirected graph model, where H is the hidden layer and can be regarded as a feature extractor; W is a bidirectional connection matrix; V is visible Floor.
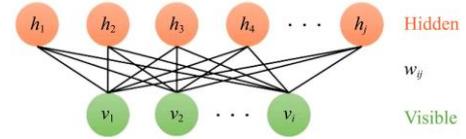
Note: Wij represents the directed connection matrix between the explicit layer node i and the hidden layer node j.

Figure 15   The simple schematic diagram of RBM model structure

Suppose a restricted Boltzmann machine has m explicit layer nodes and n hidden layer nodes, and the vector v represents the state of the explicit layer node, and the vector h represents the hidden layer node[38].   The energy definition of the system is

$$E(v,h|\vartheta) = -\sum_{i=1}^{m} a_i v_i - \sum_{j=1}^{n} b_j h_j - \sum_{i=1}^{m}\sum_{j=1}^{n} v_i W_{ij} h_j \qquad (21)$$

In the above equation, $\vartheta = \{a_i, b_j, W_{ij}\}$ are the parameters that restrict the Boltzmann machine, and they are all real numbers.   Among them, the variable a_i is the bias of the explicit layer node i, the variable b_j is the bias of the hidden layer node j, and the variable W_{ij} represents the directed connection matrix between the explicit layer node i and the hidden layer node j.   The joint probability distribution of (v, h) can be obtained based on the above energy equation as follows:

$$P(v,h|\vartheta) = \frac{e^{-E(v,h|\vartheta)}}{Z(\vartheta)} \qquad (22)$$

$$Z(\vartheta) = -\sum_{v,h} e^{-E(v,h|\vartheta)} \qquad (23)$$

where, $Z(\vartheta)$ is the normalization factor, or the partition function.

When using a restricted Boltzmann machine, in general, the log-likelihood of the restricted Boltzmann machine for this data is the most direct evaluation criterion for the learning effect.   However, due to the normalization factor $Z(\vartheta)$, it is not possible to solve the log-likelihood directly, so other methods need to be selected to evaluate the learning effect of the RBM, to know the reliability of the neuron.   And the reconstruction error method was adopted to evaluate RBM because it has little calculation overhead, is a simple algorithm, easy to implement, and can evaluate the likelihood of limiting the Boltzmann machine.   The steps are as follows (Where $\|v\|$ is the first or second-order paradigm):

1) Initialization error: $E_r = 0$;

2) Cycle (all $v^{(t)}$, $t \in \{1, 2, 3, …, T\}$);

3) Calculate the conditional probability distribution $P(h,v|\vartheta)$, and extract $h \in \{0, 1\}$ from $P(h,v|\vartheta)$;

4) Calculate the conditional probability distribution $P(v',h|\vartheta)$, and extract $v' \in \{0, 1\}$ from $P(v',h|\vartheta)$;

5) $E_r^{(t)} = \|v' - v^{(t)}\|$;

6) Perform the above operations on all the input training samples, and finally return the total reconstruction error $Er = \sum_{t=1}^{T} Er^{(t)}$.

A deep belief network can be trained by both supervised and unsupervised methods[39].   Unsupervised training has low training data requirements but low precision.   Supervised training has good precision but the training dataset is difficult to obtain.   To

give consideration to the accuracy and facilitate the acquisition of the data training set simultaneously, a navigation dataset tagging method based on the variance of continuously measured data was proposed. The navigation data used was obtained during field testing. For the collected data of each sensor, the latest data with the length of *n* are defined as a group by the sliding window, then the variance of this group is calculated as the variance of the *n*-th data according to Equation (24), in which the variance from the first data to the (n-1)th data is negligible for calculation.

$$\omega^2 = \frac{(\theta_1 - m)^2 + (\theta_2 - m)^2 + \ldots + (\theta_n - m)^2}{n} \qquad (24)$$

where, ω^2 is the value of the variance; θ_i refers to the *i*-th consecutive navigation data returned by the navigation system; *n* is the amount of data selected by variance evaluation and *m* represents the average of this set of data.

The variance calculated by each sensor is normalized and processed as the output of DBN training data, as shown in Figure 16, and the training set is constructed.
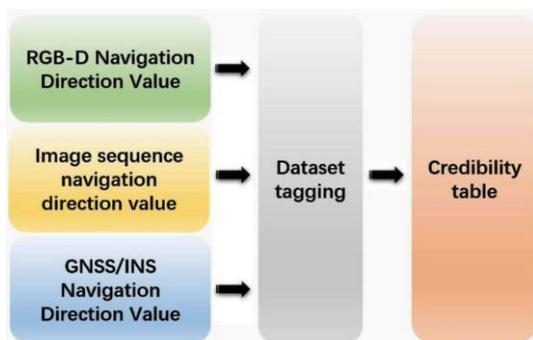
Figure 16   DBN training data acquisition flow chart

The main construction steps of the above DBN-based credibility model are as follows:

1) Establish RBM;

2) Choose a training method and convert the RBM into a neural network;

3) Obtain the hidden layer *h* transformed by RBM and use it to train the next layer;

4) Train all RBMs to meet the minimum reconstruction error;

5) Create a new multi-layer neural network and configure the results obtained by RBM initialization into the network;

6) Retrain the entire multilayer neural network to get the final model.

After completing the establishment of the above-mentioned DBN credibility model, 2000 training datasets and 1000 testing datasets were employed for model training and evaluation. The results show that the percentage of data with an error of less than 5% was 88.2% and 86.9%, respectively. The average time consumption is about 0.4 ms. The actual operation results show that it can meet the needs of field navigation of the plant phenotype detection robot, which proves the credibility model is reasonable and effective.

2.5.2   Multi-sensor information fusion model

The specific flow chart of the information fusion algorithm based on the credibility model is shown in Figure 17.

GNSS/INS combined navigation data, depth vision navigation data, and image sequence navigation data input to the above DBN credibility model and the real-time credibility η_i, $i \in (1, 3)$ of these three kinds of data can be obtained respectively. Different treatments are taken on the data according to the value of different credibility. The data flow channel will be directly closed when the

credibility is less than 50%, and the data flow will be temporarily masked. When the credibility value is greater than the specified threshold, the data flow channel will be immediately restored and the real-time monitoring will be switched on to ensure data reliability. When the credibility is within the credible range, it is substituted into Equation (25) to obtain the final navigation data τ of information fusion.

$$\tau = \frac{\theta_1 \cdot \eta_1}{\eta_1 + \eta_2 + \eta_3} + \frac{\theta_2 \cdot \eta_2}{\eta_1 + \eta_2 + \eta_3} + \frac{\theta_3 \cdot \eta_3}{\eta_1 + \eta_2 + \eta_3} \qquad (25)$$
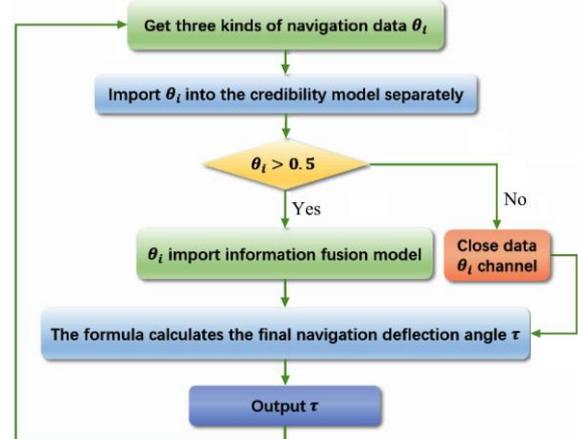
Figure 17   Three kinds of navigation information fusion flowchart

3   Experiments

The experimental platform was a plant phenotype detection variable structure robot developed by the lab and the experiment was carried out in a meadow of Nanjing Agricultural University. In this study, the computer employed for the maize seedling line navigation was configured as follows: 64-bit Windows 10 operating system, 16 GB memory, and a two-core processor Intel(R) Core(TM) i5-4200H central processing unit (CPU) @ 2.80 GHz. The proposed algorithm was tested on MATLAB 2018b platform. In the experiment, the robot traveled along the lines of the maize seedling rows and each navigation method such as GNSS/INS, depth image navigation, image sequence navigation, and the multi-sensor information fusion navigation was tested for comparing the performance. The experimental phenotyping robot and travel route are shown in Figure 18.
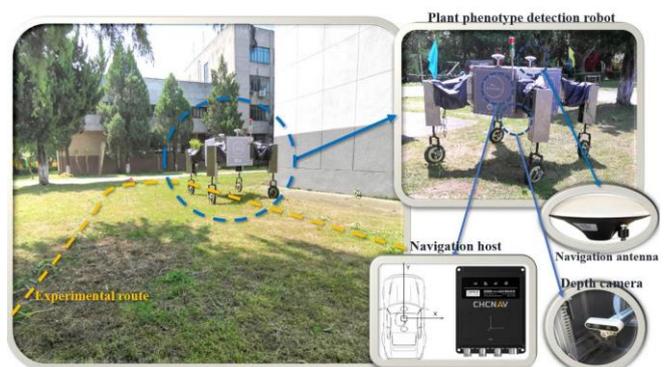
Figure 18   Experimental phenotyping robot and travel route

During the experiment, the camera was installed directly under the center of the robot for sampling the image of seedlings under the robot and at the same time to avoid uneven illumination caused by direct sunlight on plants, as shown in Figure 19a. Figure 19b shows the installation location of the signal receiver of the GNSS system. One receiver is responsible for orientation and the other for positioning. The two receivers were installed on the same

horizontal line with a certain distance for good location accuracy. Moreover, the camera was located in the center of the two receivers which is the physical center of the robot for matching the centers of different navigation devices, shown in Figure 19c. The moving trajectory such as the straight line and turning line of the phenotyping robot is shown in Figure 20.

Three experiments were carried out for testing the navigation effects under different working conditions. The first experiment tested the navigation effects of each navigation approach and that of the multi-sensor information fusion method. The second experiment was conducted to evaluate the navigation effect of the multi-sensor fusion method by intermittent sheltering of antenna signals with metal plates for creating GNSS navigation discontinuous failure. The third experiment aimed to assess the navigation effect of the multi-sensor fusion method by blocking the camera occasionally for making camera failure.

The experimental results are shown in Figure 21, and the corresponding credibility data is shown in Figure 22. Combined with the data in Table 2, it can be seen that the maximum error of only visual navigation is 8.3 cm, and the average error is 4.13 cm; the maximum error of only GNSS/INS integrated navigation is 6.2 cm, and the average error is 3.78 cm. In comparison, the maximum error of the proposed multi-sensor information fusion navigation is 4.9 cm, the average error is only 2.7 cm, which is better than that of the above single navigation algorithm.

In the application of the proposed multi-sensor information fusion navigation approach, the short-term signal lost efficacy caused by the occlusion of some sensors by surrounding trees or other influences was artificially simulated. Figure 21b and Figure 21c indicate that the usage of multi-sensor information fusion navigation can compensate for the short-term signal interruption caused by one navigation signal temporarily failure.

a. Depth camera   b. GNSS receiver   c. The center

Figure 19   Partial details of the plant phenotype detection robot

a. First line   b. Bend   c. Second line
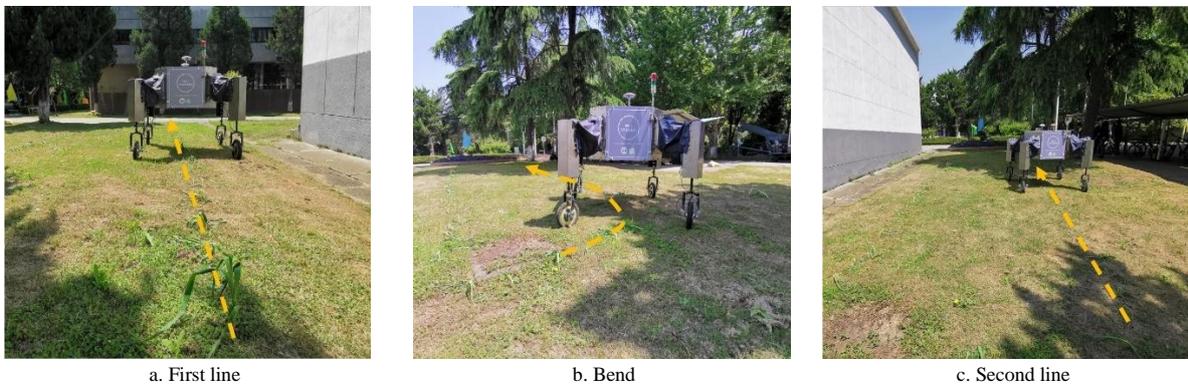
Figure 20   Detailed map of the experimental route

a. Comparison of multiple experiments and tests   b. GNSS signal occlusion experiment   c. Visual signal occlusion experiment
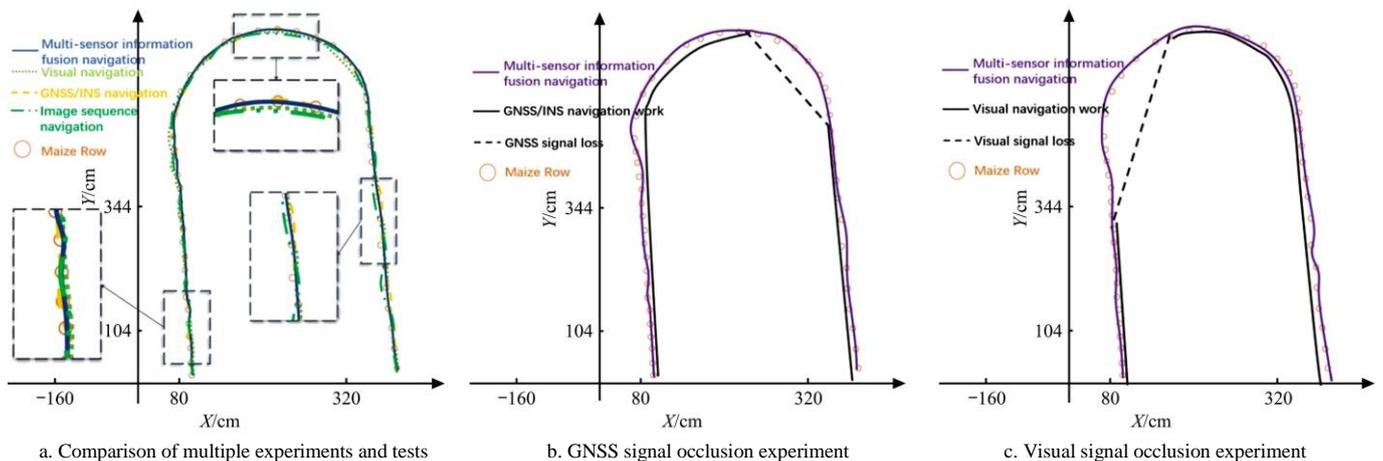
Figure 21   Error analysis diagram of experimental results of three methods
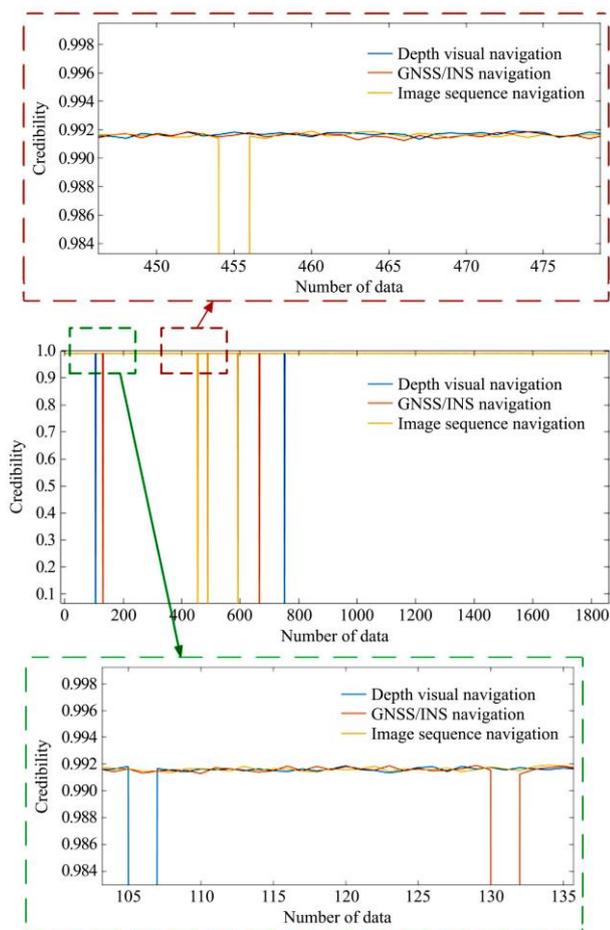
a. Multi-sensor information fusion experiment
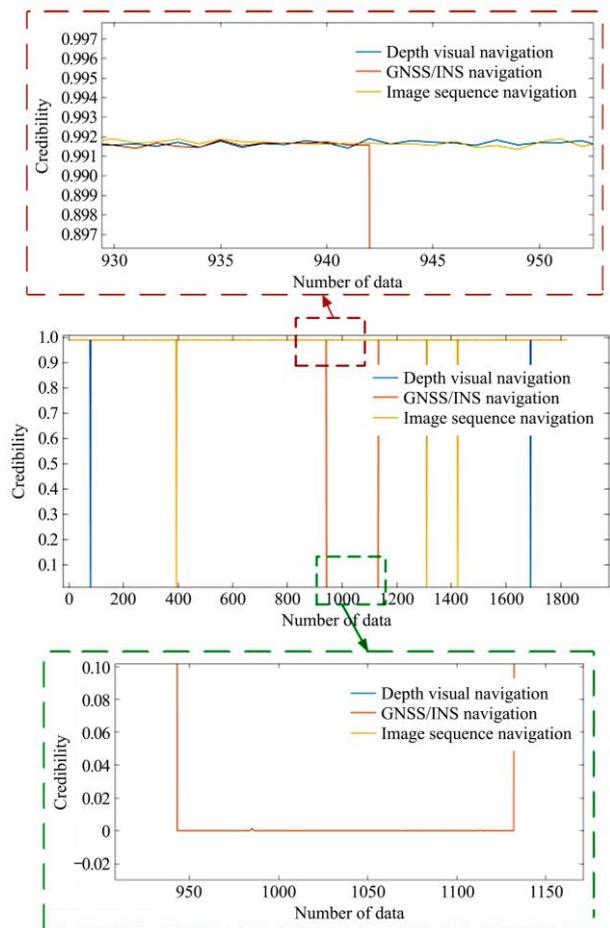
b. GNSS signal occlusion experiment
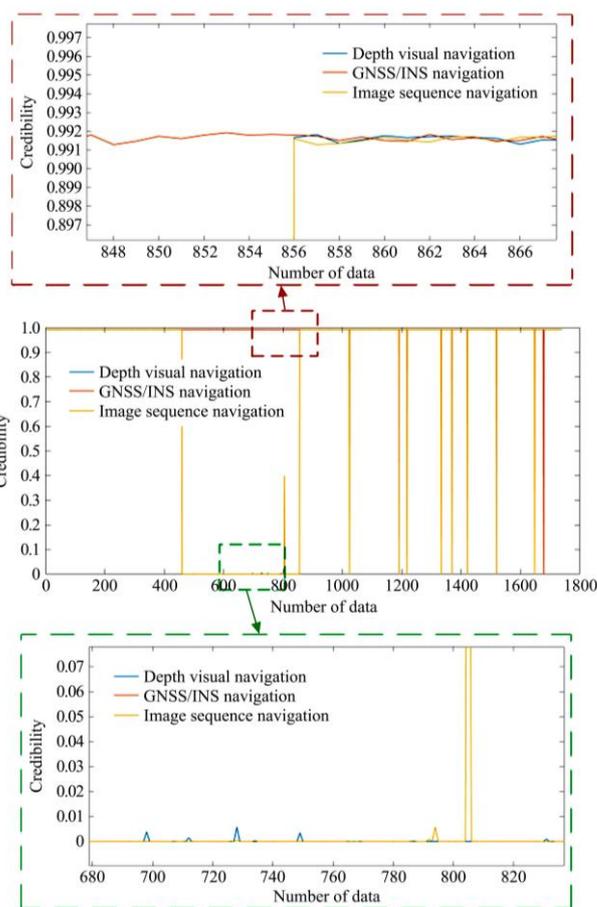
c. Visual signal occlusion experiment

Figure 22   Credibility analysis diagram of experimental results of three methods

Table 2   Comparison of different navigation results

Type	Maximum error/cm	Positioning time/ms	Average error/cm
GNSS/INS navigation	10.0	108.000	3.78
Depth visual navigation	8.3	40.325	4.13
Image sequence navigation	12.6	35.504	8.83
Multi-sensor information fusion navigation	4.9	186.000	2.70

In Figure 22b and Figure 22c, it can be seen clearly that the evaluation of the reliability of the sensors made by the dynamic evaluation unit works well and the sensor failure can be reflected. The comparative experimental results prove that the proposed multi-sensor information fusion navigation algorithm is reasonable and effective, which can meet the needs of autonomous navigation operations of the plant phenotype detection robot in the maize field.

4   Conclusions

This study aimed at improving the reliability of autonomous navigation of a plant phenotype detection robot in the field for maize seedling phenotyping detection in complicated working conditions.   Aiming at the intermittent, short-term, and uncertain failure of a single sensor unit, an intelligent navigation algorithm based on multi-sensor information fusion is proposed.   The robot visual navigation method based on depth images can realize maize seedling row extraction for real-time visual navigation in the complex farmland environment, with an average time consumption of 40 ms; frame detection on the video stream to achieve position matching of consecutive frames, thereby obtaining the robot movement displacement, with the average matching time is

35.5 ms; real-time acquisition of GNSS/INS integrated navigation data, and navigation parameter conversion to achieve reliable navigation, under normal circumstances, the navigation accuracy is within 5 cm. The reliability model based on DBN was tested on the testing set, and the accuracy reaches 86.9%. Experimental results demonstrate that using the DBN reliability model, the navigation error can be reduced to less than 2.7 cm by ignoring big error sensor information dynamically in the complicated farmland environment such as tree shading, uneven field, high weeds, etc., compared to the big error of 30 cm of the traditional fixed equal weight navigation model, which demonstrates good reliability and robustness of the proposed method. The navigation data refresh rate is 50 times per second using small memory footprint consumption.

The above three navigation information were intelligently fused using the credibility model to obtain an intelligent navigation algorithm with a time cost increment of about 78 ms and more computational resource consumption compared with a single sensor to achieve higher navigation accuracy in the complicated field condition.

The proposed intelligent navigation algorithm of a plant phenotype detection robot based on dynamic credibility evaluation can judge and select reliable sensor information and ignore large error data dynamically to improve the reliability and navigation accuracy compared with the existed methods[40]. Therefore, the proposed intelligent navigation method based on multi-sensor information fusion in the paper can meet real-time high reliable robot navigation in the complicated agricultural field for plant phenotyping detection and monitoring[41], which has important practical application prospects.

Acknowledgements

[References]

[1] Pini M, Marucco G, Falco G, Nicola M, De Wilde W. Experimental testbed and methodology for the assessment of RTK GNSS receivers used in precision agriculture. IEEE Access, 2020; 8(1): 14690–14703.

[2] Yang J, Du T, Liu X, Niu B, Guo L. Method and implementation of a bioinspired polarization-based attitude and heading reference system by integration of polarization compass and inertial sensors. IEEE Transactions on Industrial Electronics, 2020; 67(11): 9802–9812.

[3] Lu W, Zeng M J, Wang L, Luo H, Mukherjee S, Huang X H, et al. Navigation algorithm based on the boundary line of tillage soil combined with guided filtering and improved anti-noise morphology. Sensors, 2019; 19(18): 3918. doi: 10.3390/s19183918.

[4] Bolourian N, Hammad A. LiDAR-equipped UAV path planning considering potential locations of defects for bridge inspection. Automation in Construction, 2020; 117(16): 103250. doi: 10.1016/j.autcon.2020.103250.

[5] Zhu J P, Cen H Y, He L W, He Y. Development and performance evaluation of a multi-rotor unmanned aircraft system for agricultural monitoring. Smart Agriculture, 2019; 1(1): 43–52. (in Chinese)

[6] Li S C, Zhang M, Cao R Y, Ji Y H, Zhang Z Q, Li H, et al. Development of the automatic navigation system for combine harvester based on GNSS. Int J Agric & Biol Eng, 2021; 14(5): 163–171.

[7] Zhou H, Hu L, Luo X W, Tang L M, Du P, Mao T, et al. Design and test of

[8] Tayebi A, Gomez J, Fernández M, de Adana F S, Gutiérrez O. Low-cost experimental application of real-time kinematic positioning for increasing the benefits in cereal crops. Int J Agric & Biol Eng, 2021; 14(3): 194–199.

[9] Huang Z, Ono M, Shiigi T, Suzuki T, Habaragamuwa H, Nakanishi H, et al. Is spread spectrum sound a robust local positioning system for a quadcopter operating in a greenhouse? Chemical Engineering Transactions, 2017; 58(1): 829–834.

[10] Tientadakul R, Nakanishi H, Shiigi T, Huang Z, Jacky Tsay L W, Kondo N. Indoor navigation system by combining ultrasonic wave TOA and inertial measurement. 2020 59th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE), 2020; pp.1690–1695. doi: 10.23919/SICE48898.2020.9240233.

[11] de Miguel M Á, García F, Armingol J M. Improved LiDAR probabilistic localization for autonomous vehicles using GNSS. Sensors, 2020; 20(11): 3145. doi: 10.3390/s20113145.

[12] Kovacs G, Kunii Y, Maeda T, Hashimoto H. Trajectory estimation and position correction for hopping robot navigation using monocular camera. ROBOMECH Journal, 2020; 7(1): 25. doi:10.1186/s40648-020-00172-3.

[13] Ma Y, Zhang W Q, Qureshi W S, Gao C, Zhang C L, Li W. Autonomous navigation for a wolfberry picking robot using visual cues and fuzzy control. Information Processing in Agriculture, 2021; 8(1): 15–26. doi: 10.1016/j.inpa.2020.04.005.

[14] Velasquez A E B, Higuti V A H, Guerrero H B, Gasparino M V, Magalhaes D V, Aroca R V, et al. Reactive navigation system based on H∞ control system and LiDAR readings on corn crops. Precision Agriculture, 2020; 21(2): 349–368.

[15] Huang Z, Tsay L W J, Zhao X, Fukuda H, Shiigi T, Nakanishi H, et al. Position and orientation measurement system using spread spectrum sound for greenhouse robots. Biosystems Engineering, 2020; 198(1): 50–62.

[16] Elghamrawy H, Karaim M, Tamazin M, Noureldin A. Experimental evaluation of the impact of different types of jamming signals on commercial GNSS receivers. Applied Sciences, 2020; 10(12): doi: 4240. 10.3390/app10124240.

[17] Mahmud M S A, Abidin M S Z, Mohamed Z, Rahman M K I A, Buyamin S. Enhanced probabilistic roadmap for robot navigation in virtual greenhouse environment. Modeling, Design and Simulation of Systems, 2017; 752(1): 172–182.

[18] Jasinski M, Maczak J, Radkowski S, Korczak S, Rogacki R, Mac J, et al. Autonomous agricultural robot-conception of inertial navigation system. Challenges in Automation, Robotics and Measurement Techniques, Springer, 2016; 440(1): 669–679.

[19] Alberto-Rodríguez A, Neri-Muñoz M, Fernández, J R, Márquez-Vera M A, Ramos-Velasco L E, Díaz-Parra O, et al. Review of control on agricultural robot tractors. International Journal of Combinatorial Optimization Problems and Informatics, 2020; 11(3): 9–20.

[20] Aghi D, Mazzia V, Chiaberge M. Local motion planner for autonomous navigation in vineyards with a RGB-D camera-based algorithm and deep learning synergy. Machines, 2020; 8(2): 27. doi: 10.3390/machines8020027.

[21] Kolar P, Benavidez P, Jamshidi M. Survey of data fusion techniques for laser and vision based sensor integration for autonomous navigation. Sensors, 2020; 20(8): 2180. doi: 10.3390/s20082180.

[22] He W, Lian B W, Zhang L L. GNSS/INS ultra-tightly integrated tracking algorithm assisted by fuzzy control. Journal of Northwestern Polytechnical University, 2016; 34(1): 98–105. (in Chinese)

[23] Yang G C, Li Q, He W. A new method of initial alignment assisted by pseudo-satellites. Navigation Positioning & Timing, 2017; 4(3): 17–21. (in Chinese)

[24] Jin X, Li R S, Ji J T, Yuan Y W, Li M Y. Obstacle avoidance transplanting method based on Kinect visual processing. Int J Agric & Biol Eng, 2021; 14(5): 72–78.

[25] Li HY, Liu H, Cao N, Peng Y, Xie SR, Luo, J, et al. Real-time RGB-D image stitching using multiple Kinects for improved field of view. International Journal of Advanced Robotic Systems, 2017; 14(2): 172988141769556. doi: 10.1177/1729881417695560.

[26] Hasan S M A, Ko K. Depth edge detection by image-based smoothing and orphological operations. Journal of Computational Design and Engineering, 2016; 3(1):191–197.

[27] Paulo F, Zhao Z, Jithin M, Nusrat J, John S. Distinguishing volunteer corn from soybean at seedling stage using images and machine learning. Smart

laser-controlled paddy field levelling-beater. Int J Agric & Biol Eng, 2020; 13(1): 57–65.

Agriculture, 2020; 2(3): 61–74.

[28] Qu Y D, Cui C S, Chen S B, Li J Q.   A fast subpixel edge detection method using Sobel-Zernike moments operator.   Image and Vision Computing, 2005; 23(1): 11–17.

[29] Grafova L, Kasparova M, Kakawand S, Prochazka A, Dostalova T.   Study of edge detection task in dental panoramic radiographs. Dentomaxillofacial Radiology, 2013; 42(7): 720120391.   doi: 10.1259/ dmfr.20120391.

[30] Li K, He F Z, Yu H P.   Robust visual tracking based on convolutional features with illumination and occlusion handing.   Journal of Computer Science and Technology, 2018; 33(1): 223–236.

[31] Rosten E, Drummond T.   Fusing points and lines for high performance tracking.   In: 10th IEEE International Conference on Computer Vision (ICCV 05), Beijing: IEEE, 2005; 1: 1508–1515.   doi: 10.1109/ ICCV.2005.104.

[32] Rosten E, Drummond T.   Machine learning for high-speed corner detection.   In: Computer Vision – ECCV 2006, Springer, 2006; 3951(1):430–443.

[33] Mikolajczyk K, Schmid C.   A performance evaluation of local descriptors. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2005; 27(10): 1615–1630.

[34] Lindeberg T.   Image matching using generalized scale-space interest points.

Journal of Mathematical Imaging and Vision, 2015; 52(1): 3–36.

[35] Stanhope T P, Adamchuk V I.   Feature-based visual tracking for agricultural implements.   IFAC PapersOnline, 2016; 49(16): 359-364.

[36] Hinton G E, Osindero S, The Y W.   A fast learning algorithm for deep belief nets.   Neural Computation, 2006; 18(7): 1527–1554.

[37] Mohamed A R, Dahl G E, Hinton G.   Acoustic modeling using deep belief networks.   IEEE Transactions on Audio Speech and Language Processing, 2012; 20(1): 14–22.

[38] Li H W, Xu B S, Du C H, Yang Y.   Performance prediction and power density maximization of a proton exchange membrane fuel cell based on deep belief network.   Journal of Power Sources, 2020; 461(1): 228154. doi: 10.1016/j.jpowsour.2020.228154.

[39] Liu W B, Wang Z D, Liu X H, Zeng N Y, Liu Y R, Alsaadi F E.   A survey of deep neural network architectures and their applications. Neurocomputing, 2017; 234(1): 11–26.

[40] Xu J, Yang G, Sun Y, Picek S.   A multi-sensor information fusion method based on factor graph for integrated navigation system.   IEEE Access, 2021; 9(1): 12044–12054.

[41] Shamshiri R R, Weltzien C, Hameed I A, Yule I J, Grift T E, Balasundram S K, et al.   Research and development in agricultural robotics: A perspective of digital farming.   Int J Agric & Biol Eng, 2018; 11(4): 1–14.