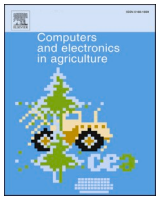
Computers and Electronics in Agriculture 215 (2023) 108409

Contents lists available at ScienceDirect

Computers and Electronics in Agriculture

journal homepage: www.elsevier.com/locate/compag



Remote safety system for a robot tractor using a monocular camera and a YOLO-based method

Sixun Chen [a], Noboru Noguchi [b,*]

[a] Graduate School of Agriculture, Hokkaido University, Sapporo, Japan
[b] Research Faculty of Agriculture, Hokkaido University, Sapporo, Japan

ARTICLE INFO

Keywords:
Robot tractor
Remote control
Monocular camera
Deep learning
Obstacle detection

ABSTRACT

Japanese agriculture faces serious problems caused by labor shortages due to both a decline in the agricultural population and the aging of the remaining agricultural population. Japanese government is actively promoting the use of agricultural robots to address labor shortages issues. According to the *Safety Assurance Guidelines for Agricultural Machinery Autonomous Navigation* issued by the Ministry of Agriculture, Forestry, and Fisheries of Japan. Human monitoring is necessary during the work of agricultural robots. A remote safety system was developed to support the monitoring of a multi-robot tractor. It uses a monocular camera as an image-input device and a deep-learning method YOLO model as a detector for humans and tractors. The system acquires image data at the robot's local end and sends it to the remote end for an image analysis; it then calculates the relative position of the detected target to the robot tractor. The Q-Q plot and *t*-test were employed to enhance the accuracy of human positioning. The safety results of the system's analysis are sent back to the tractor for execution and to generate an alert to the human monitor. Human positioning with a relative error of 2.6% at 15 m was obtained. Every target was correctly detected in the 2022–2023 field experiment. These results demonstrate that the remote safety system can support the human monitoring of a robot tractor.

1. Introduction

Japanese agriculture faces serious problems caused by labor shortages due to both a decline in the agricultural population and the aging of the remaining agricultural population. Ministry of Agriculture, Forestry, and Fisheries (MAFF) of Japan conducts a regular census of the country's agriculture and forestry industries, and in 2020 the census revealed that the number of agriculture management entities had decreased by 20 % compared to 2015. In addition, among the personnel engaged mainly in agriculture, 70 % were aged over 65 years. Together, Japan's aging population and low birth rate have produced a serious labor shortage that is being felt in many industries, and it has not been easy to attract new farmers. This problem is a serious concern in other countries too (e.g., China and Korea).

It has been expected that agriculture-robot technology can address these agricultural population issues, with robot safety being the foremost requirement for widespread adoption. For example, researchers at Japan's Hokkaido University developed a robot tractor that can do farm work within a 5 cm error (Noguchi et al., 1997). One of the essential requirements for such agriculture robots is the ability to work safely. The detection of obstacles by an agriculture robot is an aspect of safe operation, and it was shown that lasers are an economical and practical device for this purpose (Kise et al. 2005). However, a two-dimensional (2D) laser cannot detect an obstacle whose height is lower than the laser's scanning plane. Guo et al. (2002) developed a safety alert system that uses two ultrasonic sensors, and this system can detect the position

Abbreviations: AI, artificial intelligence; ANN, artificial neural network; CLAS, centimeter-level augmentation service; CNN, convolutional neural network; CSPNet, Cross Stage Partial Network; EV, electric vehicle; GIS, geographic information system; GNSS, global navigation satellite system; GPS, global positioning system; IMU, inertial measurement unit; ISO, International Organization for Standardization; LiDAR, light detection and ranging; mAP, mean average precision; MEC, Multi-Access Edge Computing; MS COCO, Microsoft Common Objects in Context dataset; NUC, next unit of computing; PASCAL VOC, pattern analysis statistical modeling and computational learning visual object classes; PNP, perspective-n-point method; Q-Q plot, Quantile-Quantile plot; R-CNN, region-based convolutional neural networks; R-FCN, region-based fully convolutional network; RTK, real-time kinematics; CVM, support vector machine; TCP, transmission control protocol; TOF, time of flight; YOLO, you only look once.

* Corresponding author.
E-mail address: noguchi@agr.hokudai.ac.jp (N. Noguchi).

https://doi.org/10.1016/j.compag.2023.108409
Received 21 May 2023; Received in revised form 1 November 2023; Accepted 2 November 2023
Available online 29 November 2023

of a moving object in the vicinity of agricultural machinery and generate a warning signal to ensure the operator's safety. The sensors have good stability, but they cannot recognize visual information and thus cannot perform more complex tasks, such as identifying obstacle types.

An outdoor localization with a 3D-laser scanner was proposed by Yang et al. (2018) to solve the problem of poor localization accuracy in GPS (global positioning system)-denied environments. Three-dimensional laser scanning has precise positioning capabilities, but Li and Ibanez-Guzman (2020) also provided a review of current automotive light detection and ranging (LiDAR) systems, identifying their high cost as a major constraint or challenge. Yang and Noguchi (2012) created a system with omni-directional stereovision provided by two multi-lens-based high-definition (HD) omni-directional cameras to detect humans. The system can recognize humans and has an error less than 0.5 m, but the cameras are rather expensive and thus not suitable for use by most farmers. Each of the above-described systems has its advantages and disadvantages, and it was suggested that the use of fusion sensors as an alternative can better cope with the problems caused by the systems' disadvantages (Castanedo, 2013). The fusion method of spatial information from LiDAR and machine vision was described by Sun et al. (2021). The proposed method is able to achieve a balance between detection accuracy and detection speed.

With the increasing labor shortage in various regions, the demand for large-scale uses of agricultural robots is also increasing. A multi-robot tractor system for conducting agriculture field work was developed by Zhang and Noguchi (2017). Their system can improve work efficiency by having multiple robots working together. A key point in the development of the multiple-robot system is that when multiple robots work together in one field, a single human is sufficient to monitor all of the robots. This method has saved labor but is not as efficient as having multiple robots working in different fields. If multiple robots work separately in different fields, there is no risk of collision between the robots, but more human monitors are needed to monitor the operations of the robots based on the current conditions (Noguchi, 2000). According to the *Safety Assurance Guidelines for Agricultural Machinery Autonomous Navigation* issued by the Ministry of Agriculture, Forestry, and Fisheries of Japan. Human monitoring is necessary during the work of agricultural robots. For agricultural robots working in different fields, the deployment of a human monitor for each field is not as efficient as desired. Developing a remote-control system to monitor all of the robots active in various fields from just one monitoring room would be an effective way to save labor costs and improve the robots' efficiency. Albiero et al., 2022 also suggests a roadmap for the Agricultural Robotics Research Community (ARRC) to optimize agricultural operations by using multiple robot tractors with lower power instead of a single large machine, thus aiming to enhance logistics, operational geometry, and energy efficiency through robotics.

Convolutional neural networks (CNNs) have been widely used in many fields (Alzubaidi et al., 2021). In the field of autonomous driving, CNNs have been applied to object detection, lane detection, and pedestrian recognition, enabling vehicles to perceive their environment and make decisions based on real-time data (Bojarski et al., 2016; Arnold et al., 2019). Object-detection techniques using deep CNNs are also helpful in ensuring the safe operations of robot tractors. Deep-learning calculations generally require a high-performance personal computer (PC), and unlike self-driving vehicles driven in an urban setting, robot tractors that operate mainly in farmland must deal with high temperatures (Oh et al., 2020), high vibration (Prasad et al., 1995), sand, and mud. Harsh operating environments can adversely affect the long-term use of high-performance PCs. The use of embedded systems instead of high-performance PCs for deploying these artificial intelligence (AI) models is a common choice. Zhang et al. (2022) deployed a deep-learning model for strawberry detection on a Jetson Nano computer (Cass, 2020), but the capacity of the embedded device is much lower than that obtained with a PC, thus resulting in a much lower speed of strawberry detection on the Jetson Nano compared to the PC. This lower speed poses a danger if it is applied for real-time human detection.

To enable the stable, cost-effective, and efficient monitoring of multiple tractors in operation, we have developed a remote safety system for robot tractors. The system uses a monocular camera installed on each tractor to collect visual data from the front of the robot tractor. The visual data are transmitted to 'the cloud' via internet for analysis, and instructions from the system are sent back to the robot tractors for execution. Moorehead et al. (2012) developed a system can also control autonomous tractor in remote end for orchard maintenance. Compared to this system, our newly developed remote safety system has several advantages. (1) Compared to communication through a local area network, communication over the Internet enables remote monitoring of autonomous tractor robots located at greater physical distances. A single human operator can monitor multiple robot tractors working simultaneously at vastly different locations, thus increasing efficiency and providing labor-savings. (2) Differentiated from Geometric Detector, Appearance Classifier or other traditional machine learning method, deep-learning obstacle-detection methods are closer to human thinking and logic and have better credibility. (3) Our primary data processing and computations are conducted remotely. The majority of the human operator's work is carried out in a remote-control monitoring room in a controlled environment, relying solely on signals to control the robot. This improves the system's stability and makes it easier to maintain and update. (4) Compared to depth cameras and 3D-LiDAR, monocular cameras and 2D-LiDAR have the advantages of low cost and easy maintenance. Furthermore, the method of remote data processing eliminates the need to install high-performance computers on each individual working tractor. All these advantages will help ordinary farmers make use of these new technologies (Noguchi and Barawid, 2011).

The remainder of this article is organized as follows. Section 2 introduces the system architecture, the obstacle detection method YOLOv5s, the obstacle positioning method perspective-n-point (PNP), the method of analyzing and correcting errors using Q-Q plots and t-tests, and the principles underlying the division of stopping and deceleration zones. Section 3 presents the results of our experiment testing the system, and Section 4 provides a summary.

2. Materials and methods

We next describe the main hardware components of the robot tractors used with the new system, the detection of obstacles, and the positioning method.

2.1. Experimental hardware

The hardware of the system shown in Fig. 1 consists of two main parts: one for the robot tractor and one for the remote control of the robot tractor. The Vehicle Robotics laboratory of Hokkaido University developed the robot tractor used in this study, i.e., the Kubota MR1000A (Crawler) (Table 1).

A global navigation satellite system (GNSS) and inertial measurement unit (IMU) sensors are used to obtain information about the robot's location and attitude. The GNSS provides centimeter-level augmentation service (CLAS) to enhance the positioning accuracy. The camera (PixPro, Kodak) is connected to a 'next unit of computing' (NUC) and is used as an input device for capturing images with a resolution of 1280 × 720 pixels. The NUC is a data relay station between the remote-end computer and the robot tractor controller. A compact, lightweight 2D LiDAR sensor (UTM-30LX, Hokuyo, Osaka, Japan) directly connected to the robot controller is used to detect obstacles other than people and tractors, and its use can also prevent accidents due to a network delay or missed detection.

For the remote-control part of the system, a workstation (ThinkPad P71, Lenovo) equipped with an Intel Xeon E3-1535 M V6 processor with 16 GB of RAM and an NVIDIA Quadro P4000 graphic card and a 16-GB

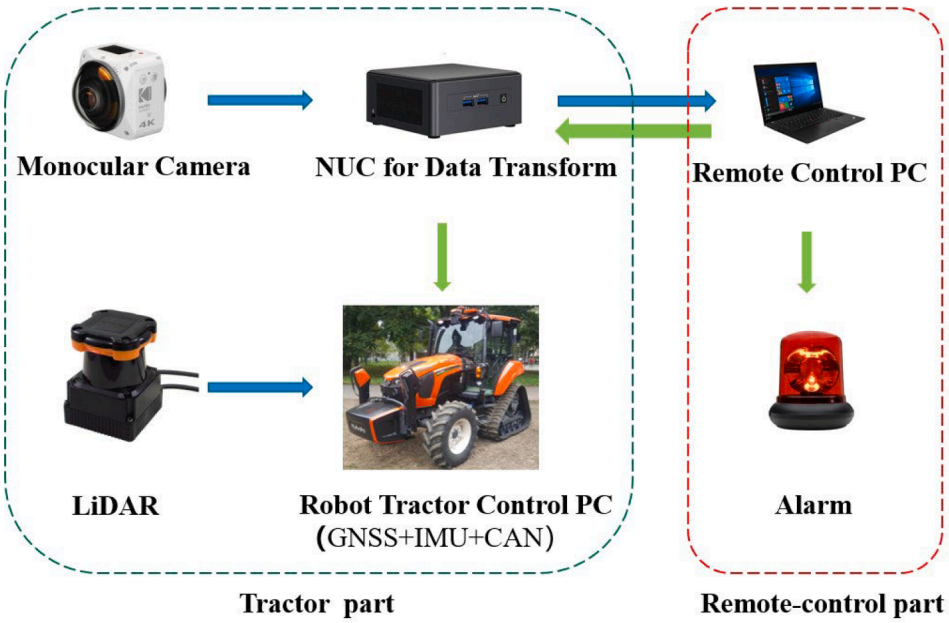


Fig. 1. Overview of the remote safety system.

Table 1
Specifications of the system.

Tractor part	
Platform	MR1000A, Kubota, Japan
Image sensor	KODAK PIXPRO 4KVR360 (lens B)
Spatial resolution	1280 × 720 pixels
Maximum frame rate	32 FPS
LiDAR sensor	UTM-30LX, Hokuyo, Osaka, Japan
CPU of NUC	Intel Core i5-10210U, up to 4.2 GHz
IMU	VN100, VectorNav, US
GNSS	Alloy, Trimble, US
Remote-control part	
Remote control PC	ThinkPad P71, Lenovo, China
CPU	Intel Xeon E3-1535 M V6, 4.2 GHz
Memory	DDR4 64 GB
Graphic card	NVIDIA Quadro P4000,8GB memory
Alarm	handmade; Arduino

RAM PC is used to process the images to determine whether or not there is an obstacle and to calculate the positions of obstacles. An alarm (handmade; Arduino) alerts the operator when an obstacle is detected.

2.2. Remote safety system workflow

The new system's remote safety system workflow as shown in Fig. 2 can be described as follows. (1) If the camera captures a picture, the camera will transmit the raw image data to the NUC in the tractor via a universal serial bus (USB) cable. The NUC sends the image data to the workstation in the remote monitoring room through the internet by connected wireless Wi-Fi. The transmission control protocol (TCP) is used for image transmission.

(2) After the image data are received, the workstation will input the received raw image data into the CNN-based image analysis program, and the program will analyze the raw image by a YOLOv5s neural network. If an object (e.g., a human or a tractor) is detected, the program will assign a serial number to the detected object and mark it in the image in the form of a bounding box. Next, the coordinates of the object in the image are transformed to the world coordinate system, and the distance between the object and the tractor is calculated to generate a safety index. Table 2 summarizes the safety index details. When more than one object is detected simultaneously, the sizes of all safety indexes

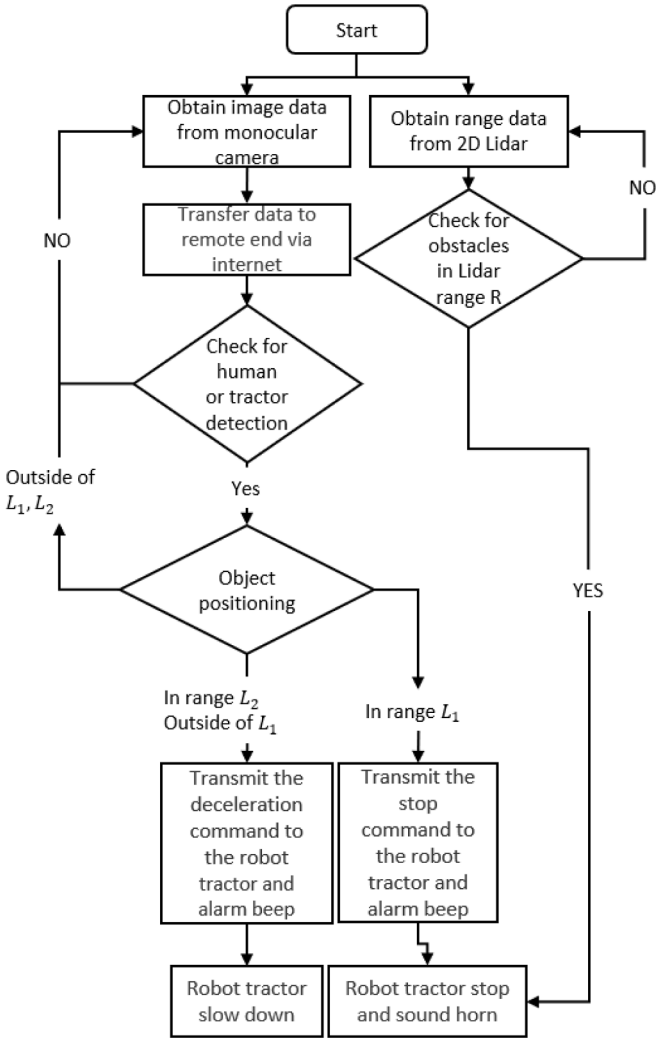


Fig. 2. Flow chart of the remote safety system.

Table 2
Definition of the safety index.

Safety index	0	1	2	3
Safety condition	Safe	Safe	Attention	Stop
Tractor Condition	Normal	Normal	Slow down and beep	Stop and beep
Obstacle position	No detection	$> (L_1 + L_2) and > R$	$> L_1 and < (L_1 + L_2)$	$< L_1 or < R$
Alarm	Standby	Standby	Beep	Beep

in each frame are compared, and the largest safety index is output. Each safety index is transmitted back to the robot tractor NUC via the internet.

Finally, the robot tractor's NUC transmits the received safety index via USB cable to the robot tractor's control computer, which is directly connected to the LiDAR sensor and accepts the safety index from both the LiDAR sensor and the NUC. The control computer prioritizes the execution of the tractor's action corresponding to the safety index with the larger number, and it sends the command results to the tractor via a controller area network (CANBUS) according to the safety conditions described in Table 2. The ranges of L_1, L_2, R, and W are shown in Fig. 3. Where L_1 is the visual stopping zone range, L_2 is the visual deceleration zone range, R is the LiDAR stopping zone range, and W is the visual zone width.

2.3. Obstacle detection method

Compared to autonomous vehicles operating in cities, robotic tractors generally have fewer detection targets, and we thus chose the two most frequent obstacles in farmland as detection targets: humans and tractors. Other possible obstacles are detected using a LiDAR sensor. Human detection is a challenge that robot tractors must overcome before being applied to real-world scenarios because human safety is most important issue.

Regarding traditional approaches applied to human detection, from the classifiers point of view there are several classification algorithms used to perform pedestrian detection, most of which are applied in a supervised approach, e.g., support vector machine (SVM), artificial neural network (ANN), or boosting algorithms (Brunetti et al., 2018). As deep-learning approaches there are some mainstream object detection architectures, including 'you only look once' (YOLO) (Redmon et al., 2016), R-FCN (Dai et al., 2016), R-CNN (Girshick et al., 2014), Faster R-CNN (Ren et al., 2015), Mask R-CNN (He et al., 2017), and SSD (Liu et al., 2016),which are all CNN-based. Generally, there is no specific guideline on which model researchers or practitioners should use. The choice of model varies depending on factors such as memory requirements, accuracy, and time cost, and the choice should be determined based on the specific detection task.

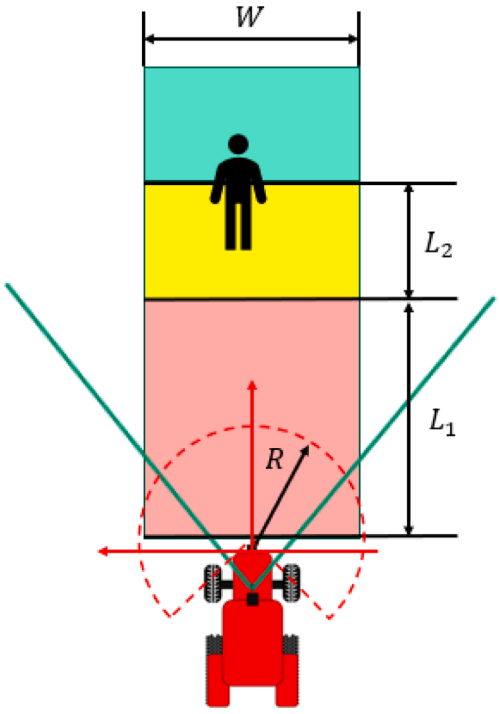


Fig. 3. Segmentation of the system's detection area. In this study, we used 8 m for L_1, 4 m for L_2, 5.5 m for R, and 3.2 m for W, where L_1 is the visual stopping zone range, L_2 is the visual deceleration zone range, R is the LiDAR stopping zone range, and W is the visual zone width.

For the detector, we trained three models in order to compare their performance and select the best model. The system initially used YOLOv3 in the Darknet framework as the detection model (Redmon and Farhadi, 2018) and trained a Faster R-CNN as a comparison, and it was then updated to YOLOv5s in the PyTorch framework as the detection model. It is worth noting that there are different types of object detection algorithms, such as one-stage algorithms like YOLO and two-stage algorithms like Faster R-CNN. The Faster R-CNN algorithm first generates a set of object proposals and then classifies them, whereas YOLO directly divides the image into grids and predicts the category for each grid as shown in Fig. 4a, thereby significantly improving the speed of detection. This feature makes YOLO very suitable to be used in practical projects. YOLO's pre-trained weights have good detection for human targets, but there is no detection for robot tractors. We increased the dataset size to address this problem. The experiments conducted in the present study used YOLOv5s as the detector.

The training environment used for the detector in this study was the Windows 10 operating system, with an Nvidia Geforce RTX 3070 (8G) GPU, a Darknet framework for YOLOv3, a Pytorch framework for YOLOv5, and Faster R-CNN (ResNet) (He et al., 2016). The training data set can be divided into two categories: human data from the MS COCO (Microsoft Common Objects in Context) dataset (Lin et al., 2014), and tractor and human data from the internet and video recordings of robot tractors. We selected 4,366 human datasets from the open-source MS COCO dataset. The self-collected datasets were obtained from the video recordings of robot tractors owned by the VeBots Laboratory of Hokkaido University and online tractor images as shown in Fig. 4b, which consisted of 1,430 items labeled with tractors and humans. Data augmentation is a proven and effective method for improving a model's performance that aims to increase the dataset size by transforming the existing data (Shorten and Khoshgoftaar, 2019). We enlarged the self-labeled dataset by rotating, inverting, and scaling. In addition to these traditional data augmentation methods, we used the newer method of mosaic data augmentation (Bochkovskiy et al., 2020), which reduces the reliance on a large batch size by stitching four random images into one image as shown in Fig. 4c; the enhanced image includes the information of all four images. The detection of small targets is usually poorer than the detection of large targets, and stitching four images into one image expands the number of small targets in the dataset, thereby improving the detection performance of the model for small targets. We supplemented the number of self-labeled datasets to 5,400 by using the data augmentation method. Before training the Faster R-CNN, it is necessary to convert the dataset in COCO format to the PASCAL VOC (pattern

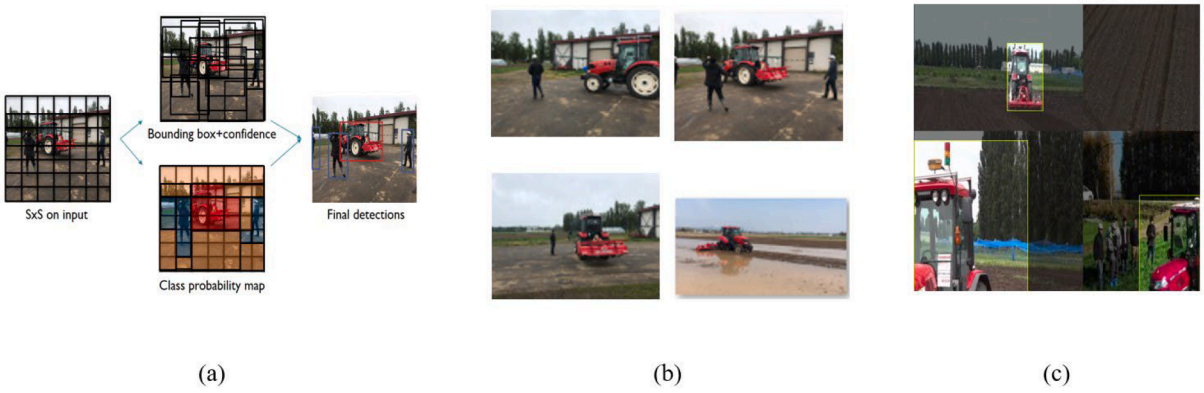


Fig. 4. Training of a CNN model. **(a)** YOLO detection flowchart. YOLO directly divides the image into grids and predicts the category for each grid. **(b)** Self-collected datasets, which consisted of 1,430 items labeled with tractors and humans. **(c)** Data augmentation, rotating, inverting, scaling and mosaic data augmentation was used to expand the dataset.

analysis statistical modeling and computational learning visual object classes) format. The detection results of the remote safety system are depicted in Fig. 5.

2.4. Obstacle positioning method

We used a monocular camera to collect images, as shown in Fig. 6. The camera was mounted on the pan-tilt unit at the top of the robot tractor at 2.6 m above the ground, with a 30° angle. The images captured by the camera in real-time, with a resolution of 1280 × 720 pixels, are transmitted to the NUC inside the tractor. Subsequently, the images will be transmitted over the internet to a remote high-performance processor for further processing.

The perspective projection model for cameras can be expressed as follows:

$$Z_C \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = K[R|T] \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \tag{1}$$

where $P_i = [X_w\ Y_w\ Z_w\ 1]^T$ is the homogenous world point, $p_i = [x\ y\ 1]^T$ is the corresponding homogeneous image point, K is the matrix of the intrinsic camera parameters, Z_C is a scale factor for the image point, and $[R|T]$ is the matrix of the extrinsic parameters. The matrix of intrinsic and extrinsic camera parameters is:

$$K = \begin{bmatrix} f_x & & c_x \\ & f_y & c_y \\ & & 1 \end{bmatrix} \tag{2}$$

where f_x and f_y represent the focal length in terms of pixels, and c_x and c_y represent the principal point of the camera.

$$[R|T] = \begin{bmatrix} R_{11} & R_{12} & R_{13} & T_1 \\ R_{21} & R_{22} & R_{23} & T_2 \\ R_{31} & R_{32} & R_{33} & T_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3}$$

where R is the rotation matrix and T is the translation matrix.

According to Zhang's method (Zhang, 2000), we can calibrate the camera by using a checkerboard to obtain the intrinsic K, extrinsic $[R|T]$, and distortion coefficients.

The conversion equation for the pixel coordinates to the tractor's world coordinates is:

$$Z_C \begin{matrix} x \\ y \\ z \end{matrix} = \begin{pmatrix} f_x & 0 & c_x & 0 \\ 0 & f_y & c_y & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} [R|T] \begin{pmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{pmatrix} \tag{4}$$

Equation (4) can be extended as:

Fig. 5. The remote safety system's detection results. The purple bounding box is showing detection result of human and green bounding box is showing detection result of tractor. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Fig. 6. Camera position and angle. The camera was mounted on the pan-tilt unit at the top of the robot tractor at 2.6 m above the ground, with a 30° angle.

$$\begin{cases} Z_C{*}x = X_w{*}(f_x{*}R_{11} + c_x{*}R_{31}) + X_w{*}(f_x{*}R_{12} + \\ c_x{*}R_{32}) + Z_w{*}(f_x{*}R_{13} + c_x{*}R_{33}) + f_x{*}T_1 + c_x{*}T_3 \\ Z_C{*}y = X_w{*}\left(f_y{*}R_{21} + c_y{*}R_{31}\right) + Y_w{*}(f_y{*}R_{22} + \\ c_y{*}R_{32}) + Z_w{*}(f_y{*}R_{23} + c_y{*}R_{33}) + f_y{*}T_2 + c_y{*}T_3 \\ Z_C = X_w{*}R_{31} + Y_w{*}R_{32} + Z_w{*}R_{33} + T_3 \end{cases} \quad (5)$$

According to the perspective-n-point (PNP) method (Fischler and Bolles, 1981), we only need to know the world coordinates and the pixel coordinates of a feature point, the intrinsic camera parameters, and the camera distortion coefficients to find the world coordinates of the camera. We set four control points (C_1, C_2, C_3, and C_4) on the ground as shown in Fig. 7. The control points' coordinates and the distance between the camera and each control points are obtained from the real-time kinematics (RTK) global positioning system (GPS). The pixel coordinates are obtained by imaging the control point in the tractor camera, and the world coordinates of the camera relative to the plane XY can be restored.

Since the points that we want to measure must be located on the plane of contact with the ground, we can assume that $Z_w = 0$ and substitute it into Eq. (5), which can be simplified to:

$$\begin{cases} Z_C{*}x = X_w{*}(f_x{*}R_{11} + c_x{*}R_{31}) + X_w{*}(f_x{*}R_{12} + \\ c_x{*}R_{32}) + f_x{*}T_1 + c_x{*}T_3 \\ Z_C{*}y = X_w{*}\left(f_y{*}R_{21} + c_y{*}R_{31}\right) + Y_w{*}(f_y{*}R_{22} + \\ c_y{*}R_{32}) + + f_y{*}T_2 + c_y{*}T_3 \\ Z_C = X_w{*}R_{31} + Y_w{*}R_{32} + T_3 \end{cases} \quad (6)$$

The ternary equation with the three unknowns X_w, Y_w, and Z_C is obtained and can be solved. The obtained solution is the world coordinates $P_i = [X_i\ Y_i\ 0\ 1]^T$ of the pixel point p_i and the distance Z_{ci} between the camera and p_i. According to the world coordinates and the pre-measured distance between the control point for the tractor, we can calculate the distance D_i of P_i for the tractor. It should be noted that in this study we used the component D_{xi} of D_i in the X-axis direction as the actual distance.

In addition to visual positioning, our system also utilizes 2D-LiDAR positioning as a local safety device. Compared to cameras, LiDAR can perform more accurate distance measurements, is less susceptible to factors like vibration and lighting conditions, and does not rely on data learning. It can detect the distance to any obstacle that reflects laser beams. LiDAR employs the time of flight (TOF) measurement method. A laser is emitted from the transmitter of the rangefinder, which illuminates the target and reflects to the receiver. The distance to the target d is calculated by measuring the time t that it takes for the laser to travel to the target and back, using the speed of light c.

The calculation formula is as follows:

$$d = c{*}\frac{t}{2} \quad (7)$$

The Hokuyo UTM-30LX LiDAR that we used has a maximum scanning range of 270°, a maximum distance of 30 m, and a scanning time of 25 ms.

2.5. Input data correction

The bounding box of the object in an image is detected using YOLOv5s. If an object is detected, we can obtain the pixel coordinates of each bounding box. For the location estimation, we used a single pixel $p_i(x_i, y_i)$ which was located at the center of that bounding box's bottom edge to represent each detection. The actual pixel position where the target is located will have deviation ε_p from the pixel position of p_i as

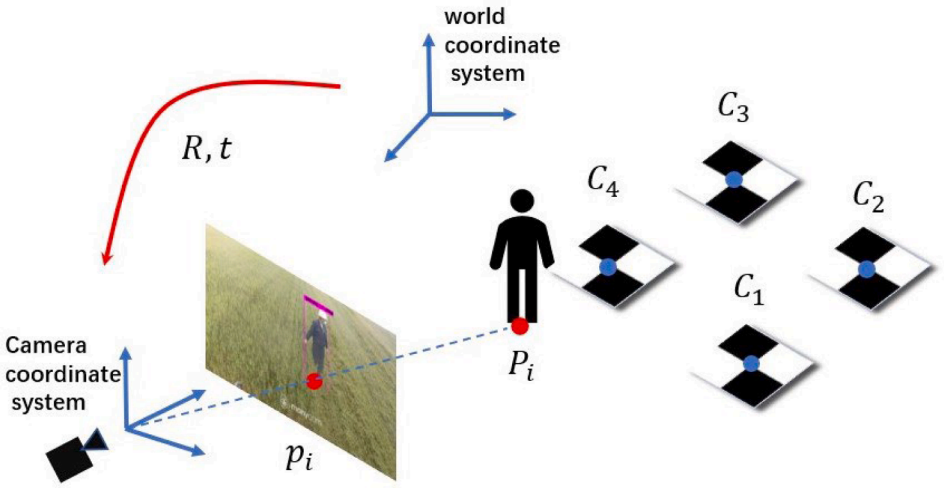


Fig. 7. The perspective-n-point (PNP) method. The center point of the camera projecting to XY, where these four control points are located, is the origin of the world coordinate system; the tractor's forward direction is the X-axis positive direction, the Z-axis directs to the center of the camera lens, and the Y-axis can be derived by the right-hand rule.

shown in Fig. 8. Due to the imaging principle of a monocular camera, this deviation may cause more significant positioning errors at greater distances. The deviation of the object in the same position may be different in different frames. This phenomenon exists in many detection algorithms that use the bounding box to mark objects.

If the ground truth has a pixel coordinate in the Y-direction of y_g, we define ε_p as follows:

$$\varepsilon_p = y_i - y_g \tag{8}$$

To investigate the probability distribution of this deviation in the YOLOv5s algorithm, we counted the deviation of the detected YOLOv5s predicting the bounding box of the target from the actual value in 1-m increments within a range of 15 m. All of the actual values were marked manually in a subjective manner. Since the markers beyond the range of 15 m will produce significant errors due to small-pixel-value floating, and the objects that are far away from the tractor are not part of this study object of interest, only the distribution within the 15-m range was thus considered. A total of 158 ground truths were labeled, and to compare the differences in ε_p between each distance range, we performed a *t*-test (Daniel and Cross, 2018) for each adjacent distance range. The *t*-test requires that the data obey a normal distribution, and we made a Q-Q plot (quantile–quantile plot) (Chambers, 2018) for each interval as shown left56197500in Fig. 9.

From the Q-Q plot, we can find that the probability distributions of all deviations on the 12 intervals obey a normal distribution. For each interval with the adjacent intervals, we applied a two-sample *t*-test; $\overline{x}$ and σ represent the sample mean and standard deviation of ε in each range, respectively. The null hypothesis is that two populations have an equal mean, and the alternative hypothesis is that the two means are not equal:

$$H_0 : \mu_1 = \mu_2 \, H_1 : \mu_1 \neq \mu_2 \tag{9}$$

For equal variance is assumed: In this case the test statistic *t*:

$$t = \frac{(\overline{x}_1 - \overline{x}_2)}{\sigma_p \sqrt{\frac{1}{n_1} + \frac{1}{n_2}}} \tag{10}$$

where $\overline{x}_1$ and $\overline{x}_2$ are sample means, n_1 and n_2 are sample sizes, ${\sigma_1}^2$ and ${\sigma_2}^2$ are sample variances and pooled variance σ_p is used:

$$\sigma_p = \sqrt{\frac{(n_1 - 1){\sigma_1}^2 + (n_2 - 1){\sigma_2}^2}{n_1 + n_2 - 2}} \tag{11}$$

Degrees of freedom (DF) *v*:

$$v = n_1 + n_2 - 2 \tag{12}$$

For equal variance is not assumed: In this case the usual two sample t-statistic no longer has a t-distribution and approximate test statistic, t' is used:

$$t' = \frac{(\overline{x}_1 - \overline{x}_2)}{\sqrt{\frac{{\sigma_1}^2}{n_1} + \frac{{\sigma_2}^2}{n_2}}} \tag{13}$$

The t-distribution with v' (DF) is used to approximate the distribution of t' where:

$$v' = \frac{\left({\sigma_1}^2/n_1 + {\sigma_2}^2/n_2\right)^2}{\frac{({\sigma_1}^2/n_1)^2}{n_1 - 1} + \frac{({\sigma_2}^2/n_2)^2}{n_2 - 1}} \tag{14}$$

If a p-value reported from a *t*-test is < 0.05, we considered the result statistically significant, and if a p-value was > 0.05, the result was considered insignificant. As shown in Table 3, the p-values of the

Fig. 8. Deviation of the bounding box and ground truth. (a) Deviation in the close range. (b) Deviation in the long range. The purple box is the predicted box of the YOLO network, and the red line is the ground truth of the position where the obstacle is standing. We selected the point at the center of the line connecting both feet as the location on the human body to draw the red line. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

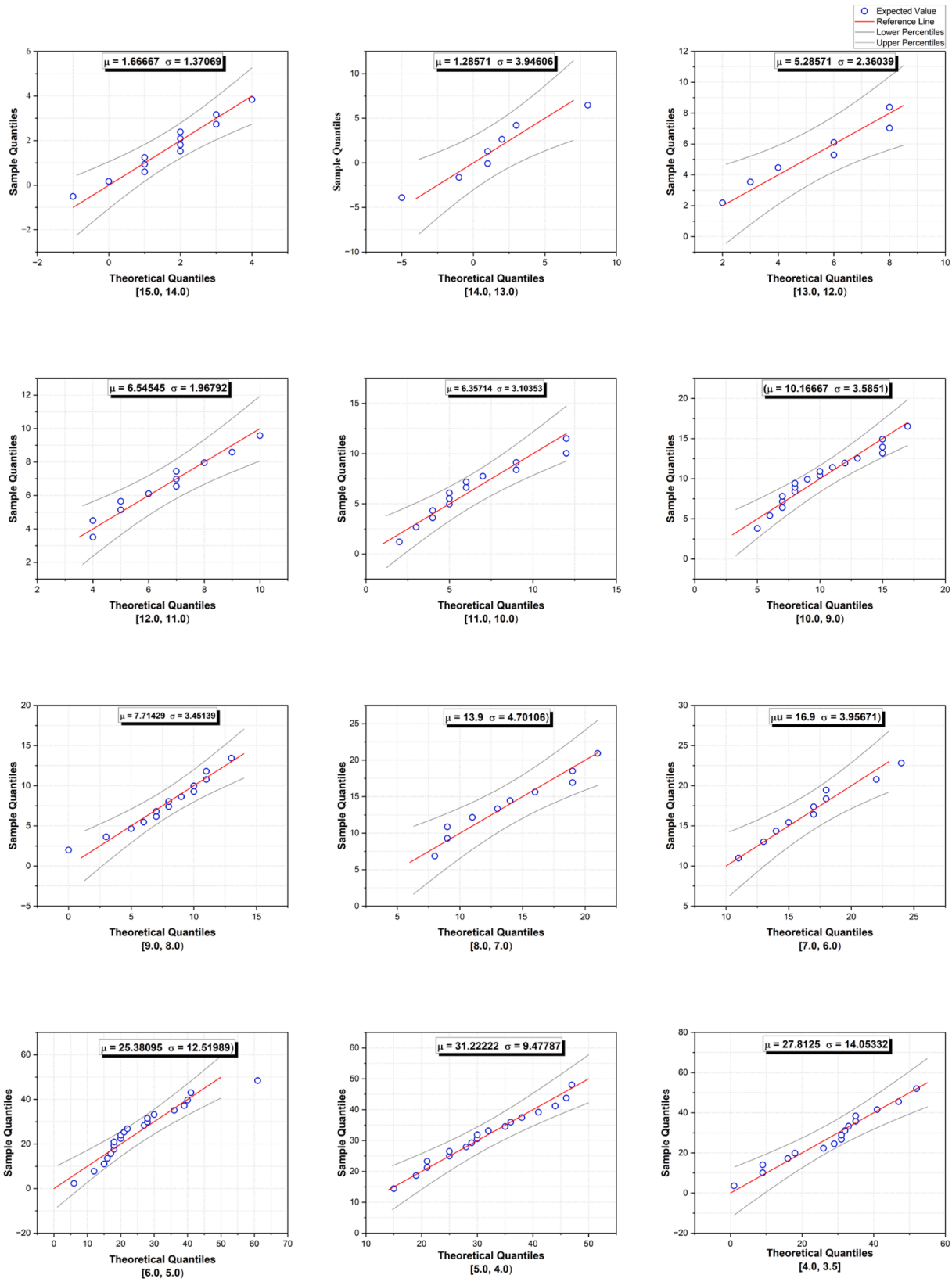


Fig. 9. Q-Q plot for each interval of the 15-m range. The points will fall on the 45° reference line if the data in each range are normally distributed.

adjacent intervals at 13 m, 10 m, 8 m, and 6 m are all < 0.05, and we thus consider that there were significant differences in the probability distributions of the regional deviation on both sides of the above ranges. The merged intervals as shown in Fig. 10 within the 15-m range can be found in Table 4.

After the statistical analyses, we observed that in most cases, ε_p is positive. This is because, at this camera angle, the size of the bounding box assigned by YOLO to the detected person is often larger than the

Table 3
Probability distribution for ε_p in each interval of the 15-m range.

Interval (m)	n	$\bar{x}$	σ	t'	v'	P
[15.0, 14.0)	12	1.7	1.37	0.25	6.86	0.812
[14.0, 13.0)	7	1.3	3.90	−2.30	9.81	**0.045**
[13.0, 12.0)	7	5.3	2.36	−1.18	11.17	0.264
[12.0, 11.0)	11	6.55	1.97	0.18	22.16	0.855
[11.0, 10.0)	14	6.36	3.10	−3.22	29.60	**0.003**
[10.0, 9.0)	18	10.2	3.59	1.96	28.58	0.060
[9.0, 8.0)	14	7.71	3.45	−3.54	15.66	**0.003**
[8.0, 7.0)	10	13.9	4.70	−1.54	17.50	0.141
[7.0, 6.0)	10	16.9	3.96	−2.82	26.66	**0.009**
[6.0, 5.0)	21	25.4	12.52	−1.66	36.49	0.106
[5.0, 4.0)	18	31.2	9.48	0.82	25.85	0.420
[4.0, 3.5]	16	27.8	14.05	N/A	N/A	N/A

[footnote] n: the number of detections; t': the t-value with reference to the next adjacent interval; v': the degrees of freedom; P: the p-value with reference to the next adjacent interval. Only the data assuming unequal variances is listed, and the differences between the two hypotheses are consistent based on the calculations.

person's actual size, and this value increases as the distance between the camera and the person decreases. For the points to be measured in the pixel ranges of these five intervals, we apply the pixel coordinate correction in different intervals, assuming that the pixel coordinates of the YOLO predicted the point of an object falling in interval i is $p_c(x_c, y_c)$. The distance is then calculated using the pixel points representing the object as $p_c(x_c, y_c - \overline{\varepsilon}_i)$, where $\overline{\varepsilon}_i$ is obtained by calculating the regression equation of sample deviation and y_c in distance interval i.

Regarding the positioning correction of tractor targets, due to the significantly larger size of tractors compared to humans, using a single pixel to represent the tractor's position would result in significant errors in localization. For now, no corrections are being made to the calculations for the positions of the other tractors, and currently, this model occasionally produces some cases where it detects only the tractor but not the attached implements or separates the calculations for the implements and the tractor, which also increases the positioning error. We thus plan to employ a keypoint detection method to predict the tractor's position. To determine the angle of the tractor robot from the wheels and calculate the nearest distance, it will be necessary to reannotate and retrain the collected data. This is also one of the tasks for future work.

2.6. Safety zone

In this study, the base tractor we used is the Kubota MR1000A crawler-type tractor with a total length of 4,505 mm, a width of 1,980 mm, and a total height of 2,725 mm. The commonly used operating width for rotary operations is 2,600 mm or 2,800 mm. For safety purposes, when operated remotely, the maximum specified speed can be set to 10 km/h, and when a specified rotation count of 2,600 mm is selected, the actual maximum vehicle speed is 8.8 km/h. In the deceleration zone, if a target is detected and the speed exceeds 3.6 km/h, the robot will decelerate to 3.6 km/h. This section requires confirmation of the ranges for the visual stopping zone, the visual deceleration zone, and the LiDAR stopping zone.

Referring to the performance evaluation criteria for reducing pedestrian collision injuries set by Japan's Ministry of Land, Infrastructure, Transport and Tourism (MLIT), it is necessary to avoid collisions with pedestrians crossing in front when they are moving at a speed of 5 km/h. The following four points were therefore considered when specifying the range for each zone in this study: (1) A visual stopping zone serves as the first layer of the safety zone, and when the robot is traveling at its maximum speed (8.8 km/h), braking based solely on visual detection should ensure the avoidance of collisions with obstacles within the visual stopping zone.

(2) A LiDAR stopping zone serves as the second layer of the safety zone, and when the robot is traveling at its maximum speed (8.8 km/h), braking based solely on LiDAR detection should ensure the avoidance of collisions with pedestrians crossing at a speed of 5 km/h in the vicinity of the robot.

(3) To ensure the operational efficiency of the robot tractor, braking

Table 4
Probability distribution for ε_p in the merged intervals of the 15-m range.

Interval (m)	n	$\bar{x}$	σ
[15.0, 13.0)	19	1.53	2.52
[13.0, 10.0)	32	6.19	2.57
[10.0, 8.0)	32	9.09	3.68
[8.0, 6.0)	20	15.4	4.50
[6.0, 3.5)	55	28	12.13

[footnote] n: the number of detections; $\bar{x}$: the mean average; σ: standard deviation.

Fig. 10. Merged range intervals with similar probability distributions. The merging of the range intervals with similar probability distributions produced five intervals.

based solely on visual detection should minimize the impact of objects that are not on the operational path.

(4) The robot tractor can decelerate to 3.6 km/h after passing over a deceleration zone at its maximum speed (8.8 km/h).

We conducted field experiments at Hokkaido University's farm to measure the braking time and braking distance of the robot tractor under various braking modes as shown in Fig. 11.

$$S_d = S_1 - S_2 \tag{15}$$

The braking time, defined as time t_d from the vehicle's horn blast (automatically activated when an obstacle is detected in the stopping zone) to the time at which the vehicle comes to a complete stop was measured in each experimental group three times, and the results are shown in Table 5.

According to principle (1) outlined above, the visual stopping zone should be greater than the braking distance at maximum speed, i.e., $L_1 > 7.4$ m. We thus used 8 m as the value of L_1. According to principle (3), visual detection should minimize the impact of objects that are not on the operational path. The commonly used operating width for rotary operations is 2.6 m or 2.8 m (i.e., 2.8 m for W). We set the value of W as 3.2 m. The visual stopping zone was thus an 8 m × 3.2 m rectangular area in front of the tractor as shown in Fig. 3, section 2.2. According to principle (4), after setting the stopping safety zone as an 8 m × 3.2 m rectangular area, we determined that at 8 m in front of the robot tractor, a deceleration distance of 3.6 m is required to reduce the speed from 8.8 km/h to 3.6 km/h. We thus set the L_2 value as 4 m as shown in Fig. 3, section 2.2.

According to principle (2), as the final safety zone, braking based solely on LiDAR detection should ensure the avoidance of collisions with pedestrians crossing at a speed of 5 km/h in the vicinity of the robot. Considering (*i*) the robot tractor as a rectangular rigid body with a total length of 4,505 mm and a width of 1,980 mm and (*ii*) the LiDAR scanning range as a sector (adjustable-angle) specified as a semi-circle of 180° in front of the robot with a scanning radius R, Fig. 12 depicts the position relationship between the tractor and pedestrians crossing at 5 km/h. The breaking distance S_d is defined as:

Based on the Pythagorean theorem, the LiDAR scanning radius R is thus calculated as:

Fig. 11. Experiment for measuring the braking time and braking distance. We assumed an 8-m zone in front of the tractor as the visual stopping safety zone, and beyond 8 m was the safe zone. A test operator stood at the boundary of the 8-m zone judged by the system and recorded the tractor's stopping position S_2 relative to the operator's position. Subsequently, the robot tractor was remotely controlled to travel from a distant point towards the operator at its maximum speed of 8.8 km/h. After automatic stopping based on visual positioning, the position S_1 where the tractor came to a halt was recorded.

Table 5
The braking time and distance for each braking method at maximum speed.

Breaking type	$\overline{t_d}$(s)	$\overline{S_d}$(m)
Visual breaking	**3.5**	**7.4**
LiDAR breaking	2.1	3.7

[footnote] $\overline{t_d}$: the average barking time; $\overline{S_d}$: the average barking distance.

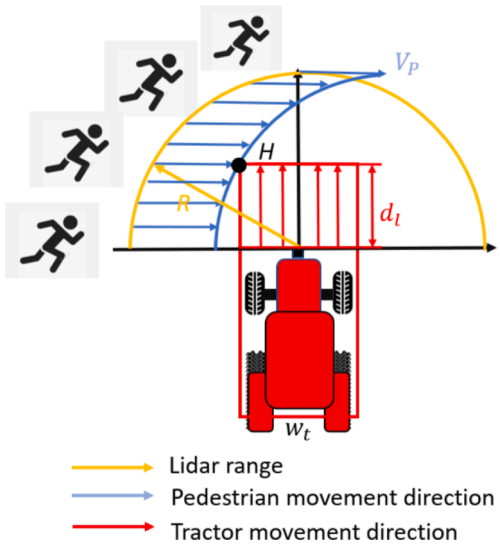


Fig. 12. The position relationship between the tractor and pedestrians crossing at 5 km/h. Assuming that the initial position of the tractor is the origin of the coordinate system with the LiDAR position. At the coordinates of point $H(-\frac{w_t}{2}, \overline{d_l})$, a pedestrian who is crossing at 5 km/h (V_P) would collide with the tractor.

$$R > \sqrt[2]{\left(-\frac{w_t}{2} - V_P * \overline{t_l}\right)^2 + \left(\overline{d_l}\right)^2} \tag{16}$$

where $\overline{d_l}$ is the average LiDAR detection barking distance, $\overline{t_l}$ is the average LiDAR detection barking time, w_t is the tractor width. We calculated that $R > 5.47$ m and set the value of R as 5.5 m as shown in Fig. 3, section 2.2.

3. Experiment and discussion

3.1. Multi-robot remote monitoring experiment

left1619250In October 2021 we set up a remote monitoring room at the Iwamizawa City Data Center as shown in Fig. 14c and monitored two robot tractors at the Hokkaido University farm (37 km away) as shown in Fig. 14b and another two robot tractors at the Iwamizawa Nishiyauchi farm (7 km away) as shown in Fig. 14a as the multi-robot remote monitoring experiment as shown in Fig. 13.

The operation and monitoring of the four tractors are carried out by a single person in the middle, while the students on both sides are responsible only for observing and preventing any unexpected situations as shown in Fig. 14c.

The four robot tractors (Kubota MR1000A crawler-type, Kubota MR1000A wheel-type, Yanmar EG105 crawler-type, Yanmar EG83 wheel-type) started from their respective starting points and went along the farm road, arrived at the appropriate field and moved within the field via two paths, and then returned to the starting point the same way. We let obstacles randomly invade the tractors' paths during this time. Every target was detected correctly in the experiment, and all robot tractors acted correctly according to the safety index. Such field

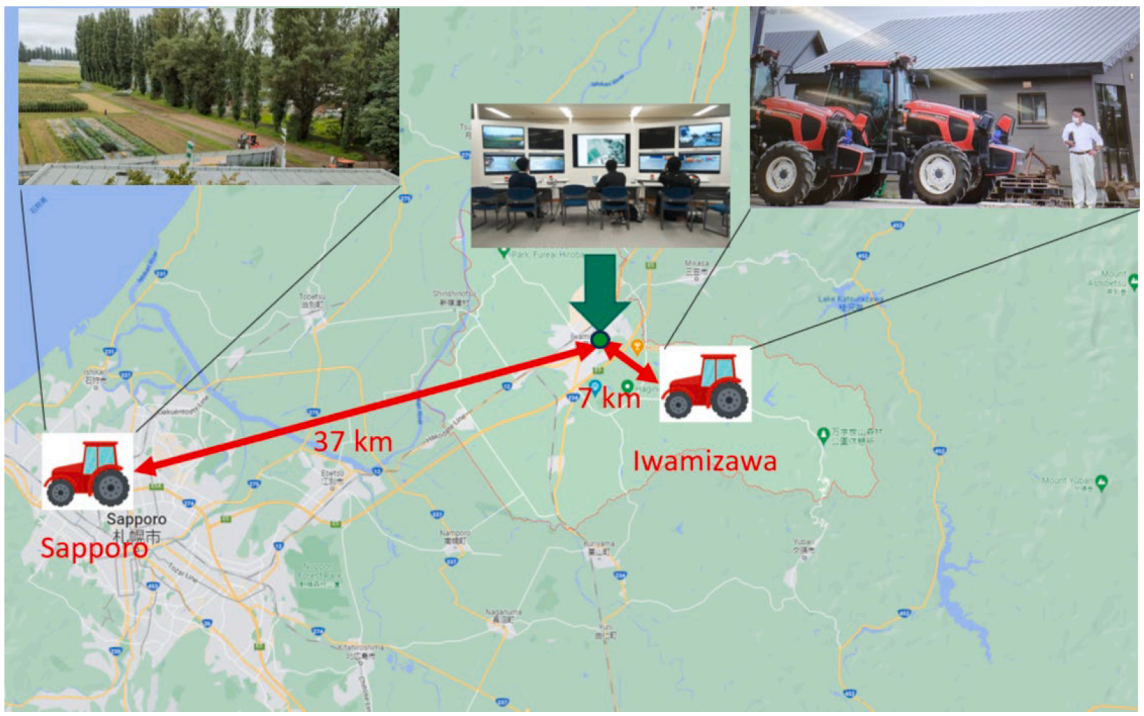


Fig. 13. Remote monitoring experiment map in Hokkaido. Remote monitoring room at the Iwamizawa City Data Center. Two robot tractors at the Hokkaido University farm. Two robot tractors at Iwamizawa Nishiyauchi farm.

Fig. 14. The remote monitoring experiment. **(a)** The Iwamizawa field running experiment. **(b)** The Sapporo field running experiment. **(c)** The Iwamizawa City remote monitoring room. In the remote monitoring room, eight screens on the left and right sides are responsible for displaying the image information transmitted from the front and rear cameras of the four robot tractors. The large screen in the middle shows the positions of the four robot tractors on a map using geographic information system (GIS) software.

experiments were conducted multiple times during 2022–2023 in Tsurunuma valley, Iwamizawa Nishiyauchi farm, the Hokkaido University farm, and Noto farm on the tractor or electric vehicle (EV) developed by Yamasaki and Noguchi (2023). The monitoring locations during the experiment, the robot positions, the distances between each location, the number of robots monitored simultaneously, the instances of obstacle intrusion, and the successful detection of obstacles are presented in Table 6. All obstacles were correctly detected.

The results of this remote monitoring experiment demonstrated that (*i*) the remote safety system can accurately and stably assist operators in monitoring and controlling multiple robot tractors working at different locations, and (*ii*) the system can be used on various agriculture robot platforms.

3.2. The detector model's performance

Since the primary goal of the remote safety system is to detect obstacles approaching the tractor, we focused on the detection performance of the model for large and medium-sized targets, as well as the accuracy and detection speed of the model. We used the current mainstream mean average precision (mAP) as the evaluation metric of the detection model in this study. The mAP metric is the average value of the AP (average precision) for all categories (person and tractor).

These metrics are calculated as follows:

$$mAP = \frac{1}{c}\sum_{i=0}^{c} AP_i \tag{17}$$

$$AP = \int_0^1 Precision(Recall)dR \tag{18}$$

$$Precision = \frac{TP}{TP + FP} \tag{19}$$

$$Recall = \frac{TP}{TP + FN} \tag{20}$$

Table 6
Field experiments for robot tractor.

Time	Monitoring location	Robot location	*D* (km)	*N*	T_o	T_s
2022-05-13	Iwamizawa	Iwamizawa	7	1	8	8
2022-05-19	Iwamizawa	Iwamizawa	7	2	8	8
2022-06-08	Iwamizawa	Sapporo/Tsurunuma	37/29	2	9	9
2022-06-21	Iwamizawa	Sapporo/ Tsurunuma/ Iwamizawa	37/ 29/7	3	12	12
2022-07-06	Iwamizawa	Sapporo/ Tsurunuma/ Iwamizawa	37/ 29/7	3	12	12
2022-07-12	Iwamizawa	Sapporo/ Tsurunuma/ Iwamizawa	37/ 29/7	3	12	12
2022-08-02	Tsurunuma	Tsurunuma	2.8	2	6	6
2022-08-03	Tsurunuma	Tsurunuma	2.8	2	8	8
2022.-08-08	Sapporo	Sapporo	1	1	4	4
2022-08-24	Tsurunuma	Tsurunuma	2.8	2	10	10
2022-08-26	Tsurunuma	Tsurunuma	2.8	2	8	8
2022-09-02	Sapporo	Sapporo	0.3	2	8	8
2022-09-21	Sapporo	Sapporo/ Tsurunuma	1/37	3	12	12
2022-10-02	Sapporo	Sapporo	0.3	2	6	6
2022-10-21	Sapporo	Sapporo	0.3	2	4	4
2022-11-09	Sapporo	Sapporo	0.3	1	9	9
2023-04-10	Sapporo	Noto	730	1	6	6
2023-05-22	Tsurunuma	Sapporo/ Tsurunuma	56/ 2.8	2	6	6
2023-05-24	Sapporo	Tsurunuma	56	2	6	6
2023-05-26	Sapporo	Sapporo	1	2	6	6
2023-06-09	Sapporo	Sapporo/ Tsurunuma/ Iwamizawa	1/56/ 37	4	14	14
2023-06-15	Sapporo	Sapporo	0.3	1	4	4

[footnote] D: the distance between monitoring room and robot location; *N*: number of robots; T_o: times of obstacle intrusion; T_s: times of successful detection.

where *TP* represent the number of correctly detected objects (true positives), *FP* represent the number of falsely detected objects (false positives), and *FN* represent the number of missed objects (false negatives). *Precision*(*Recall*) means the precision-recall curve. *c* is the number of classes, which is two in this study.

Compared to the YOLOv3 model, YOLOv5 uses the Cross Stage Partial Network (CSPNet) as its backbone network, which enhances the feature extraction capabilities based on the original model. YOLOv5 also optimizes the bounding box loss, classification loss, and object confidence loss, contributing to the improved model performance. YOLOv5s is the fastest but least accurate version among the YOLOv5 models. In the present study, we aimed to maximize the model's runtime speed while satisfying minimum accuracy requirements, and we thus chose YOLOv5s as our primary test model. Table 7 shows that the Faster RCNN model had the highest accuracy, but with a low frame rate. In practice, the speed of the analysis is an important indicator, and small-scale targets generally cause missed detection. Although a mAP value of 87.3 % may not be an exceptionally high number, we observe from the results in Table 6 that this model achieved a 100 % accuracy rate in multiple field experiments. This is because the test dataset contains many small-scale targets, and this model demonstrates a very high detection rate for targets of varying sizes within the robot's forward region of interest (ROI) during real-world usage. We conducted tests using 500 images featuring either target tractors or humans within the tractor's stopping or deceleration zone. The detection rate achieved using the YOLOv5s model was 0.98. A comparison of the three models revealed that even though the Faster RCNN model has higher accuracy, its lower frame rate may cause issues in practical applications. The accuracy of YOLOv5s and YOLOv3 is slightly lower, but while meeting the real-world usage requirements YOLOv5s has a faster speed, making it perform better in real-world scenarios. Therefore, when considering factors such as detection performance, accuracy, and speed, we believe that the YOLOv5s model holds an advantage over the others and is capable of serving as the detection head for robots in real-world applications.

3.3. Object positioning results

The actual distances were measured in spaces between 3.5 m and 15 m by the RTK-GPS as shown in Fig. 15. The difference between the actual and predicted distances was recorded for all points at 0.5-m intervals. Each group of experiments was conducted five times, and we used the average of the results. The error results are shown in Fig. 16. Further objects tend to have a larger error, meaning that the error variance is more prominent. The root mean square (RMS) error of the measured distance was 0.403 m and had an average relative error of 2.6 % at 15 m; the maximum error occurs at the farthest point of 15 m, which is 0.77 m. The maximum relative error remains at around 5 %, which does not significantly impact the safety system's judgment within the ROI; we thus consider it an acceptable value. We concluded that the remote safety system can correctly detect and accurately locate targets.

3.4. Multi-target detection results

We counted the results of a 30-sec video of the detection of multiple people that was recorded in a field experiment on the Iwamizawa City farm. We compared the actual condition of all characters within 15 m of the tractor appearing in the video and the safety conditions predicted by the safety system, as shown in Fig. 17. The solid line in the figure represents the real safety condition, and the dashed line represents the predicted safety condition. At 3–8 sec, 12 sec, and 27–30 sec, an error in the estimation of the safety condition occurred due to the effective occlusion generated by the previous person, and no prediction bounding box or wrong bounding box position was generated for these person as shown in Fig. 18.

It should be noted that there were also some small targets beyond 15 m from the tractor that were not detected. This was due to the lower detection ability of YOLO for small-scale targets compared to large-scale targets, and it will cause missed detections due to occlusion, for which the assistance of LiDAR is necessary for completing the safety evaluation. From the results, it can be seen that the system is prone to missed detections or false alarms due to occlusions. Fortunately, since the

Table 7
Results of object detection on the test set.

Detector	Backbone	*mAP*(%)	AP_p(%)	AP_t(%)	FPS
YOLOv3	Darknet53	87.4	85.4	89.3	18
YOLOv5s	CSPDarknet	87.3	84.7	89.9	**32**
Faster R-CNN	Resnet50	**89.7**	**88.3**	**91.0**	8

[footnote] AP_p : average accuracy of person class. AP_t: average accuracy of tractor class.

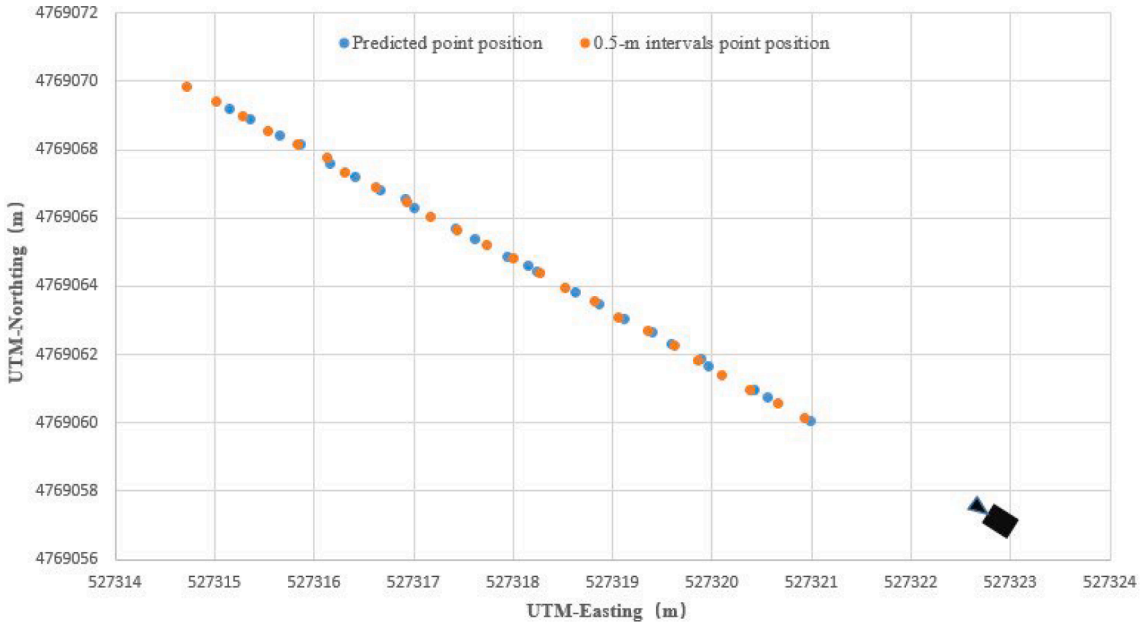


Fig. 15. Position of the range point. Blue points are the coordinates predicted by the system. Orange points are the coordinate of points measured at 0.5-m intervals using RTK-GPS. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

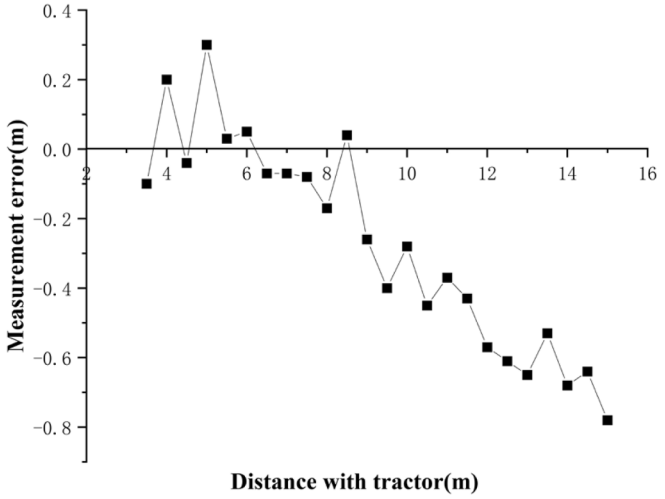


Fig. 16. Measurement error in the X-direction. The distance error tends to increase proportionally to the distance in our method. This is due to the imaging characteristics of monocular cameras and the method we used to calculate the target position by using the bounding box position.

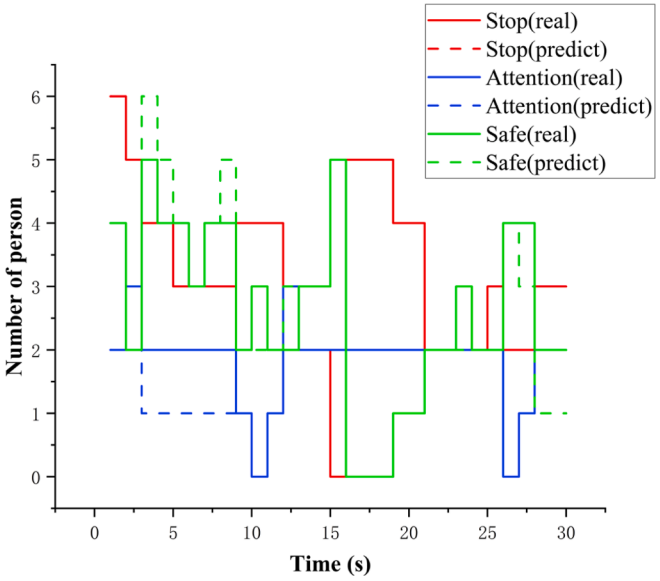


Fig. 17. Real or predict safety conditions for each character, safety condition according to Table 2. If the same-color solid line and dashed line do not overlap at a certain frame, it means that there is a prediction error. The *dashed line above the solid line* indicates a false positive, while the *dashed line below the solid line* indicates a false negative.

occluding objects are always behind the missed or falsely detected objects, although the judgment results of the occluded objects may be affected, the final safety judgment of the robot tractor is generally not affected.

3.5. The remote performance

We tested the robot tractors' braking distance at different speeds. The experiment was conducted at the Hokkaido University farm as shown in Fig. 19.

The average braking distance at different speeds is shown in Table 8. The results indicate that remote transmission affects robot-tractor braking due to network delays. The increase in vehicle speed increases this effect, but it is acceptable for the robot tractor, which always runs at a low speed (normally 3 km/h). Additionally, the robot tractor will slow down before a potential obstacle enters its path, ensuring safety during the tractor's regular operation under remote control. When only LiDAR is used as a safety device, setting the LiDAR detection range too far can lead to issues where the tractor stops, for example, when detecting corn stalks while passing through a cornfield. When only a camera is used as a safety device, network delays may pose some security risks for remote control; thus, linkage with locally connected security devices (e.g., LiDAR) is essential.

Because visual braking relies on transmitting images captured by the robot's camera to a monitoring station via a mobile Wi-Fi network for remote processing and then sending back the image-processing results to the robot for execution, both the braking distance and the braking time can be influenced by the speed and stability of the network

Fig. 18. Missed detection due to effective occlusion. For the three characters on the upper right side of the road, only two bounding boxes were generated.

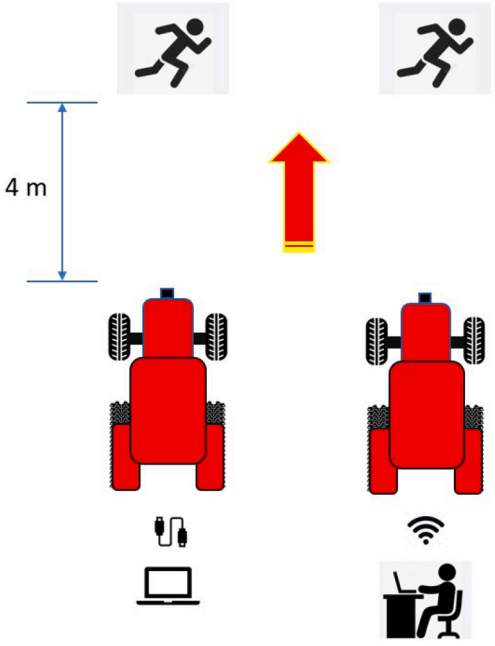


Fig. 19. Remote-control performance test experiment. A person stood 4 m in front of the tractor and invaded the tractor route under the network connection or the cable connection to the tractor control computer, respectively (with the LiDAR off), and we measured and recorded the braking distance of the tractor. Each respective test was conducted five times.

Table 8
Results of the remote-control performance test.

Control Method	Robot speed (km/h)	Braking distance (m)
Edge	2	1.05
	3.5	1.17
	5	1.68
remote	2	1.17
	3.5	1.41
	5	2.09

communication. We conducted tests using two different networks to assess their impact on remote monitoring. One network was a standard LTE mobile router available for commercial lease which had relatively unstable network connectivity, a minimum download speed of 20 Mbps, and an upload speed of 5 Mbps. On the day of testing, the network exhibited a download speed of 317 Mbps and an upload speed of 20 Mbps. The other network was a 5G mobile router provided by NTT East Japan, which had more stable network connectivity. The 5G network utilizes a technology known as Multi-Access Edge Computing (MEC). The delay generated during signal transmission by the tractor comprises latency in the wireless segment and latency in the wired segment. MEC minimizes the latency in the wireless segment by locating servers closer to the end devices in terms of physical distance. In the experiments, the server used was situated in a building approximately 400 m away from the tractor. On the day of testing, this network demonstrated a download speed of 264 Mbps and an upload speed of 126 Mbps. Under different network conditions, we conducted tests to determine the time and distance required for the robot to come to a complete stop at its maximum speed while solely visual braking was relied upon during remote control. The testing methodology was consistent with the procedures outlined above in Section 2.6.

The test results are shown in Table 9. It can be observed that under the three different network environments, the impact on braking time is minimal. However, the upload speed does have some influence on the braking distance. We believe that this is because an insufficient upload speed can lead to delays when transmitting the images captured by the tractor robot's front camera to the remote end, resulting in a delay in receiving commands from the remote monitoring end. Note that the command transmission requires very little network bandwidth, and thus the download speed has a minimal impact on the tractor's braking distance. The braking time, in contrast, is controlled primarily by the tractor robot's local program and is not significantly affected by the network environment. In summary, for remotely controlled tractor robots using this approach, the safety strategy should be determined based on the minimum upload speed according to the specific network environment they are in. Additionally, lower network latency and upload speeds contribute to the improvement of the robot's safety.

Table 9
The braking time and distance for each network environment at maximum speed.

Network type	$\bar{t}$(s)	$\bar{d}$(m)	Download (Mbps)	Upload (Mbps)
5G	3.5	7.4	264	**126**
LTE	**3.6**	**10.2**	**317**	20
LTE	3.5	11	30	10

[footnote] $\bar{t}$: the average barking time, $\bar{d}$: the average barking distance.

4. Conclusions

We developed a remote safety system that uses a monocular camera and 2D-LiDAR to obtain data and a YOLOv5s model as a detector with a detection mAP value of 87.3 %. This system can perform an image analysis at 32 FPS on a Quadro P4000 GPU-based workstation. Obstacle distances are predicted with an average relative error of 2.6 %, maximum error of 0.77 m at 15 m, and the remote control of multiple robot tractors according to the safety index can be performed. This novel system is not restricted by physical distance and is low cost, with high stability and resilience to environmental factors. The system achieved a 100 % success rate in responding to obstacle intrusions in 2022–2023 field experiments. The present experimental results demonstrate that the new system can assist an individual operator with the remote monitoring of robot tractors, thus saving labor costs and improving efficiency in agricultural applications. However, this system also has its deficiencies, including limited precision in tractor positioning, a restricted range of visual localization objects (only humans and other tractors), and susceptibility to network environmental constraints. We plan to address these issues in our future research.

CRediT authorship contribution statement

Sixun Chen: Methodology, Data curation, Investigation, Formal analysis, Software, Validation, Visualization, Writing - original draft, Writing - review & editing. **Noboru Noguchi:** Conceptualization, Supervision, Project administration, Funding acquisition, Resources, Writing - review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

This research is supported by the Research Promotion Program for Enhancing Innovative Creation funded by the Bio-oriented Technology Research Advancement Institution (BRAIN). The project is named “Development of a smart grapevine cultivation system using electric robots.”

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.compag.2023.108409.

References

Albiero, D., Pontin Garcia, A., Kiyoshi Umezu, C., Leme de Paulo, R., 2022. Swarm robots in mechanized agricultural operations: a review about challenges for research. Comput. Electron. Agric. 193, 106608. https://doi.org/10.1016/j.compag.2021.106608.

Alzubaidi, L., Zhang, J., Humaidi, A.J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M.A., Al-Amidie, M., Farhan, L., 2021. Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. Journal of Big Data 8 (1), 53. https://doi.org/10.1186/s40537-021-00444-8.

Arnold, E., Al-Jarrah, O.Y., Dianati, M., Fallah, S., Oxtoby, D., Mouzakitis, A., 2019. A survey on 3d object detection methods for autonomous driving applications. IEEE Trans. Intell. Transp. Syst. 20 (10), 3782–3795. https://doi.org/10.1109/TITS.2019.2892405.

Brunetti, A., Buongiorno, D., Trotta, G.F., Bevilacqua, V., 2018. Computer vision and deep learning techniques for pedestrian detection and tracking: A survey. Neurocomputing 300, 17–33. https://doi.org/10.1016/j.neucom.2018.01.092.

Bochkovskiy, A., Wang, C. Y., Liao, H. Y. M., 2020. YOLOv4: Optimal speed and accuracy of object detection. arXiv preprint arXiv: 2004.10934.

Bojarski, M., Del Testa, D., Dworakowski, D., Firner, B., Flepp, B., Goyal, P., et al., 2016. End to end learning for self-driving cars. arXiv preprint arXiv: 1604.07316.

Cass, S., 2020. Nvidia makes it easy to embed AI: The Jetson nano packs a lot of machine-learning power into DIY projects - [Hands on]. IEEE Spectr 57 (7), 14–16. https://doi.org/10.1109/MSPEC.2020.9126102.

Chambers, J.M., 2018. Graphical methods for data analysis. CRC Press. https://doi.org/10.1201/9781351072304.

Dai, J., Li, Y., He, K., Sun, J., 2016. R-FCN: Object detection via region-based fully convolutional networks. Adv. Neural Inf. Proces. Syst. 29, 379–387. https://doi.org/10.48550/arXiv.1605.06409.

Daniel, W.W., Cross, C.L., 2018. Biostatistics: a foundation for analysis in the health sciences. Wiley.

Fischler, M.A., Bolles, R.C., 1981. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM 24 (6), 381–395. https://doi.org/10.1145/358669.358692.

Castanedo, F., 2013. A review of data fusion techniques. The Scientific World Journal, *2013*, 1–19. https://doi.org/10.1155/2013/704504.

Li, Y., Ibanez-Guzman, J., 2020. Lidar for autonomous driving: The principles, challenges, and trends for automotive lidar and perception systems. IEEE Signal Processing Magazine 37 (4), 50–61. https://doi.org/10.1109/MSP.2020.2973615.

Guo, L., Zhang, Q., Han, S., 2002. Agricultural machinery safety alert system using ultrasonic sensors. J. Agric. Saf. Health 8 (4), 385–396. https://doi.org/10.13031/2013.10219.

Girshick, R., Donahue, J., Darrell, T., Malik, J.. Rich feature hierarchies for accurate object detection and semantic segmentation. https://doi.org/10.1109/CVPR.2014.81.

He, K., Gkioxari, G., Dollár, P., Girshick, R.. Mask r-cnn. https://doi.org/10.48550/arXiv.1703.06870.

He, K., Zhang, X., Ren, S., Sun, J.. Deep residual learning for image recognition. https://doi.org/10.48550/arXiv.1512.03385.

Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Zitnick, C. L., 2014. Microsoft coco: Common objects in context. In Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13. pp. 740-755. Springer International Publishing. https://doi.org/10.48550/arXiv.1405.0312.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., Berg, A.C., 2016. Ssd: Single shot multibox detector, 14. Springer International Publishing, pp. 21–37. https://doi.org/10.48550/arXiv.1512.02325.

Moorehead, S. J., Wellington, C. K., Gilmore, B. J., Vallespi, C., 2012, October. Automating orchards: A system of autonomous tractors for orchard maintenance. In Proceedings of the IEEE international conference of intelligent robots and systems, workshop on agricultural robotics.

Noguchi, N., Barawid, O.C., 2011. Robot farming system using multiple robot tractors in japan agriculture. IFAC Proceedings Volumes 44 (1), 633–637. https://doi.org/10.3182/20110828-6-IT-1002.03838.

Noguchi, N., Ishii, K., Terao, H., 1997. Development of an agricultural mobile robot using a geomagnetic direction sensor and image sensors. J. Agric. Eng. Res. 67 (1), 1–15. https://doi.org/10.1006/jaer.1996.0138.

Noguchi, N., 2000. Engineering challenges in agricultural mobile robot towards information agriculture. In Proceedings of the XIV Memorial CIGR World Congress, 2000, pp. 147-154.

Oh, J., Choi, K., Son, G.-H., Park, Y.-J., Kang, Y.-S., Kim, Y.-J., 2020. Flow analysis inside tractor cabin for determining air conditioner vent location. Comput. Electron. Agric. 169, 105199. https://doi.org/10.1016/j.compag.2019.105199.

Prasad, N., Tewari, V.K., Yadav, R., 1995. Tractor ride vibration—A review. Journal of Terramechanics 32 (4), 205–219. https://doi.org/10.1016/0022-4898(95)00017-8.

Redmon, J., Divvala, S., Girshick, R., Farhadi, A.. You only look once: Unified, real-time object. https://doi.org/10.1109/CVPR.2016.91.

Redmon, J. and Farhadi, A., 2018. YOLOv3: An incremental improvement. arXiv preprint arXiv: 1804.02767.

Ren, S., He, K., Girshick, R., Sun, J., 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. Adv. Neural Inf. Proces. Syst. 28, 91–99. https://doi.org/10.48550/arXiv.1506.01497.

Shorten, C., Khoshgoftaar, T.M., 2019. A survey on image data augmentation for deep learning. Journal of Big Data 6 (1), 1–48. https://doi.org/10.1186/s40537-019-0197-0.

Sun, B., Li, W., Liu, H., Yan, J., Gao, S., Feng, P., 2021. Obstacle detection of intelligent vehicle based on fusion of LiDAR and machine vision. Eng. Lett. 29 (2), 722–730.

Yamasaki, Y., Noguchi, N., 2023. Research on autonomous driving technology for a robot vehicle in mountainous farmland using the Quasi-Zenith Satellite System. Smart Agricultural Technology 3. https://doi.org/10.1016/j.atech.2022.100141.

Yang, L., Noguchi, N., 2012. Human detection for a robot tractor using omni-directional stereo vision. Computers and Electronics in Agriculture, 89, 116–125. https://doi.org/10.1016/j.compag.2012.08.011.

Yang, Q., Qu, D., Xu, F., Zou, F., He, G., Sun, M., 2018. Mobile robot motion control and autonomous navigation in GPS-denied outdoor environments using 3D laser scanning. Assembly Automation 39 (3), 469–478. https://doi.org/10.1108/AA-02-2018-029.

Zhang, Z., 2000. A flexible new technique for camera calibration. IEEE Trans. Pattern Anal. Mach. Intell. 22 (11), 1330–1334. https://doi.org/10.1109/34.888718.
Zhang, Y., Yu, J., Chen, Y., Yang, W., Zhang, W., He, Y., 2022. Real-time strawberry detection using deep neural networks on embedded system (rtsd-net): An edge AI application. Comput. Electron. Agric. 192,106586. https://doi.org/10.1016/j.compag.2021.106586.
Zhang, C., Noguchi, N., 2017. Development of a multi-robot tractor system for agriculture field work. Comput. Electron. Agric. 142, 79–90. https://doi.org/10.1016/j.compag.2017.08.017.