



OPEN

Study on a risk model for prediction and avoidance of unmanned environmental hazard

Chengqun Qiu^{1,2✉}, Shuai Zhang¹, Jie Ji³, Yuan Zhong¹, Hui Zhang¹, Shiqiang Zhao¹ & Mingyu Meng⁴

Comprehensive research is conducted on the design and control of the unmanned systems for electric vehicles. The environmental risk prediction and avoidance system is divided into the prediction part and the avoidance part. The prediction part is divided into environmental perception, environmental risk assessment, and risk prediction. In the avoidance part, according to the risk prediction results, a conservative driving strategy based on speed limit is adopted. Additionally, the core function is achieved through the target detection technology based on deep learning algorithm and the data conclusion based on deep learning method. Moreover, the location of bounding box is further optimized to improve the accuracy of SSD target detection method based on solving the problem of imbalanced sample categories. Software such as MATLAB and CarSim are applied in the system. Bleu-1 was 67.1, bleu-2 was 45.1, bleu-3 was 29.9 and bleu-4 was 21.1. Experiments were carried out on the database flickr30k by designing the algorithm. Bleu-1 was 72.3, bleu-2 was 51.8, bleu-3 was 37.1 and bleu-4 was 25.1. From the comparison results of the simulations of unmanned vehicles with or without a system, it can provide effective safety guarantee for unmanned driving.

List of symbols

C	Real license plate size
d	Euclidean geometric distance
P	Frame storage coordinates
M_1	Camera internal parameter matrix
H	Camera head height
h	Camera height
v_0, f_y	Internal parameters provided for calibration
σ	Estimated uncertainty
x_e	Estimated boundary box position and standard deviation
σ_t	Adjustable parameter for variable voting
P_y	Target of P on the Y-axis
L_{reg}	Bounding box regression
t_{xip}, t_{yi}	Deviation between the prediction and the anchor boundary frame
x_p, y_i	Bounding box position in the axle
C'	Size of the object in the picture
fps	Frame rate
P'	Frame storage coordinates
M_2	Camera position matrix
f	Camera focal length
g	Position parameter
v	Pixel height coordinate of the target in the image
c	Category confidence prediction
x_g	Real boundary box position on the ground
Q	Istance from the camera to the nearest point below

¹Jiangsu Province Intelligent Optoelectronic Devices and Measurement-Control Engineering Research Center, Yancheng Teachers University, Yancheng 224007, China. ²School of Automotive and Traffic Engineering, Jiangsu University, Zhenjiang 212013, China. ³Anhui Institute of Optics and Fine Mechanics, Chinese Academy of Sciences, Hefei 230031, China. ⁴Interdisciplinary Graduate School of Science & Engineering, Tokyo Institute of Technology, Yokohama 2268502, Japan. ✉email: ujsqcq@126.com

D	Kullback–Leibler divergence
L_{cls}	Classification loss
t_{xi}^*, t_{yi}^*	Deviation between the real and the anchor boundary frame
x_i^*, y_i^*	True box position in the axle

In the automobile industry, unmanned driving technology has attracted a great deal of attention in recent years. It can fundamentally change the automobile industry and traffic systems. On the other hand, it can also alleviate the problems of accidents, pollution, and congestion of existing vehicles and the traffic¹.

The commercialization of the unmanned driving should take safety as the premise and realize the importance of the safe unmanned driving in the complex driving environment^{2–4}, which is the theme of this paper.

Anti-collision technology is one of the key points of unmanned research. Many achievements have been made in the development of anti-collision technology, such as sensor information fusion, anti-collision research, and anti-collision warning strategy^{4–6}. However, it is still a long distance from being completely practical considering the influence of multiple working conditions.

Some scholars have concluded the problems as follows:

Limited information fusion. At present, the research on sensor fusion is only the fusion between two or three kinds of sensors, and the information of fusion cannot cover the working conditions overall^{5–7}. In order to adapt to the actual driving conditions, it is necessary to fuse the information of various sensors and other data sources.

Multiple road conditions studies are incomplete. No overall consideration is given to factors such as road environment, weather conditions, the influence of personnel in the environment, and the fastest response speed of vehicles^{8–11}.

The early warning strategy needs improving. The present study basically takes distance as the evaluation index. However, for the actual traffic situation, the process from safety to danger is a gradual change, and multiple evaluation indexes should be used^{9–13}.

To solve the problems above, this paper adopts the idea of dynamic risk assessment based on the historical data of the environment and predicts the risk by priority based on the results of environmental risk assessment^{14–17}. The integration of the booming internet big data industry and electronic information engineering technology makes the risk assessment of traffic environment no longer rely on manual rule setting and machine vision recognition, but can realize joint modeling and statistical analysis by using navigation applications and data from the transportation department 10. Moreover, it is possible to dynamically assess the risk of the environment based on historical circumstances and reapply the assessment results to the risk prediction of specific objectives in the environment^{18–20}. Therefore, this train of thought has high practical significance and application value.

Target detection is the leading technology of hazard prediction. The current target detection is mainly aimed at pedestrian, traffic sign, or obstacle^{21–23}. In 2019, It proposed an improved SSD_ARC algorithm for key target detection tasks in driving scenarios^{24–27}. This method can realize fast multi-objective recognition, semantic annotation, and positioning box selection. Although it provides a general recognition framework, it does not involve the risk of identifying the environment itself. VAR system (TTV system) is to have a portable system which can be used for any type of LCD system and help the referees to count the ball better during the game to have a more qualified game²⁸. The AGV (automated guided vehicle) is a system which typically made up of vehicle chassis, embedded controller, motors, drivers, navigation and collision avoidance sensors, communication device and batteries, some of which have load transfer devices^{29–33}. By contrast, this system makes up for this omission by adopting the idea of the priority of the big data risk conclusion model and supplement of target detection, which has high practical significance and application value^{34–47}. On this basis, this paper proposes an improved SSD method by two steps:

First, the positive sample and negative sample imbalance of SSD are improved by FL loss function.

The second is to improve the common boundary box selection and matching in the target detection algorithm.

This paper presents an environmental hazard prediction and avoidance system. The system is divided into two parts: the first part is the prediction part, which is divided into three levels, including environmental perception, environmental risk model, and target detection. The second part is the avoidance part. According to the results of hazard prediction, conservative driving strategy based on speed limit is adopted. By using this system, vehicles can slow down in high-risk areas or traffic complex environments and increase their speed when the risk is low.

The core of safe driving lies in avoiding danger. However, avoiding danger will inevitably affect driving speed and comfort, especially avoiding environmental danger, which is mainly accomplished by implementing a defensive driving strategy. Therefore, the most important part of this paper is the prediction of whether to use the defensive driving strategy. The prediction section first identifies the environment, such as identifying the intersection, lane, parking lot, and pedestrian crossing near the primary and secondary school campus, and then evaluates the risk and gives the forecast target and priority according to the historical data. Finally, the risk index of the target is predicted separately and evaluated synthetically.

At present, the visual algorithm of environment perception can complete the task of environment recognition^{42,48–60}. By combining LBS positioning and other methods, the environmental information can be preset in advance, and the recognition speed and accuracy of the visual algorithm can be improved at the same time. Environmental risk assessment algorithm uses deep learning technology, with the help of Internet open traffic accident database, the comprehensive analysis of traffic accident affected factors in order to rank the dangerous objectives in the environment. It is feasible to practice the idea, but predicting priorities will take a lot of testing to finalize. Besides, under unknown circumstances, the hazard prediction algorithm has been realized in the conflicts of people and cars. The risk prediction is still being explored and the risk error of traffic environment under different time, weather, and other factors need further correcting. The structure of thesis is divided into four aspects:

Classification of driving environment	Type of driving environment
Intersections	Three branches
Intersections	Four branches
Intersections	Multiple branches
Roads	Expressway
Roads	Main road
Roads	Secondary trunk road
Roads	Branches

Table 1. This table mainly describes the driving environment classification and driving environment type. The driving environment classification includes intersection and road. The driving environment type is divided into different branches and roads. Allowing for two points above, so as to obtain the judgment of the driving environment.

Section 1: This paper briefly introduces the development of automobile safety and anti-collision technology and explains the importance of anti-collision technology to the driverless. Although the environmental hazard prediction and avoidance system has not been developed, the significance and prospect of this system are expounded.

Section 2: There are four kinds of target detection methods commonly used in unmanned driving. In this design, we will focus on the risk prediction based on deep learning.

Section 3: Hardware and software design of environmental hazard prediction and avoidance system.

Section 4: The main function of the environmental hazard prediction avoidance system is to avoid the risk. With MATLAB, CarSim software for simulation, we can eventually obtain the experimental results to prove the feasibility of the system design implementation.

System model

The realization of the environment awareness system mainly includes four steps: Firstly, it can input the positioning data to the system with BDS/GPS satellite positioning, LBS positioning of WIFI and base stations. Secondly, it can also use the electronic compass module to achieve position refinement. Moreover, the environment pre-judgment's realizing is based on position and machine vision. Finally, the environmental data is output. Among them, the methods of satellite positioning and electronic compass positioning are quite mature, but how to achieve environmental judgment and corresponding risk assessment on this basis is the key problem to be solved by the environmental awareness system. In this paper, a risk model is established based on location, accident data, and a target detection algorithm through depth learning, which is proposed to realize environmental judgment.

Risk model based on location and accident data. According to the location information provided by the satellite and the electronic compass, it is possible to make judgments on the types of nearby environment. There are six categories of judgments: residential land, industrial land, public facilities land, commercial building land, transportation facilities land, and road land.

Since most driverless vehicles using this system are running, making detailed perception based on machine vision more necessary, which can be divided into two types: intersection and road. Driving environment types are shown in Table 1. Intersections can be divided into three types: three branches, four branches and, multiple branches. And roads can be divided into four types: expressways, main roads, secondary roads, and branch roads. Allowing for two points above, so as to obtain the judgment of the driving environment.

Based on the environment-aware data and the type of the nearby environment, the specific name of the nearby environment can be obtained. At the same time, the system can carry out Internet communication and obtain real-time traffic and weather conditions. Allowing for the above information, environmental risk can be judged from the following three aspects. First of all, risk judgment is ultimately to judge risk types and risk objectives. Secondly, they can be summarized as car-car conflict risk, car-person conflict risk, car-object conflict risk, and vehicle control risk. Eventually, risk objectives are visual objectives such as vehicles, pedestrians, bicycles and electric vehicles. Moreover, risk itself is divided into real risk and hidden risk, and real risk is the possibility of collision between risk target and vehicle on site. The hidden risks are difficult to confirm due to various reasons. But there is still the possibility of collision.

Location based. Risks based on nearby environment types are shown in Table 2. The types of nearby environment can be divided into several major categories such as residential, industrial, public facilities, commercial and transportation facilities environment. And the traffic facilities environment refers to bus stations, railway stations, airports, subway stations and other passenger transport hubs.

Time based. Through the analysis of traffic data on roads, we can sort out the traffic flow of roads at different times. Generally speaking, large traffic volume and complicated traffic environment in each environment represent greater risks, such as working days, holidays and rush hours, could affect the traffic flow. For example, whether there is a road design for non-motor vehicle isolation design and the type of environment in which it is located all affect the complexity of traffic.

Type of nearby environment	Main risk types	Major risk objectives
Reside	Car-person conflict	Non-motor vehicle, pedestrians
Industry	Car-car conflict/vehicle control	Vehicle
Public facilities	Car-person conflict	Non-motor vehicle, pedestrians
Commerce	Car-person conflict	Non-motor vehicle, pedestrians
Means of transportation	Car-car conflict/car-person conflict	Vehicle, non-motor vehicle

Table 2. This table mainly describes the types of accessories, main risk types and main risk objectives. The nearby types are divided into residence, industrial land, etc., the main risk types are divided into vehicle person conflict, vehicle control, etc., and the main risk objectives are divided into non motor vehicles, pedestrians, etc.

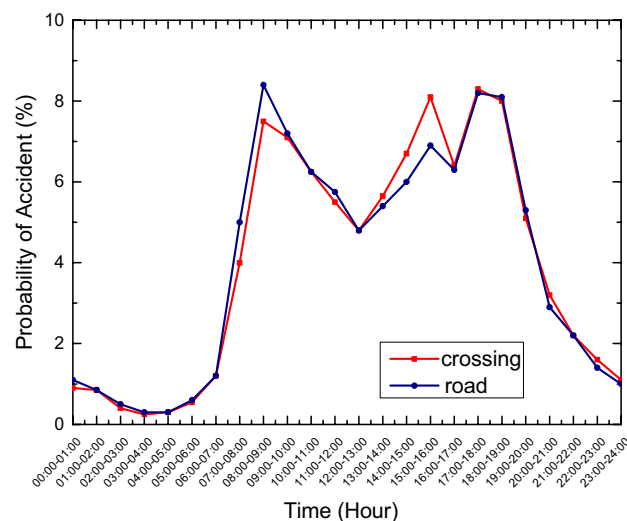


Figure 1. The accident incidence rate in hours of crossing and road in shanghai are fitted by linear relationship. It can be seen from the figure that the value is high from 6 a.m. to 8 p.m.

For working days, the risk of car-car conflict is more significant in most environments, and the same environment type is different in specific environment. For public facilities, full-time educational facilities have a significant risk of collision between people and vehicles after school. Besides, cultural facilities such as public libraries and museums have a higher risk during holidays, and medical facilities have different situations. For the business environment, different commercial districts have different traffic time distributions.

Therefore, this part of the program needs to input the current time. The specific risk type and priority of risk objectives will be determined through database queries.

Based on the scene. Due to possible errors in location and database, the system will confirm and supplement the on-site target detection. First, targets such as pedestrians and vehicles are detected ahead. Compared with the above judgment results, the existing results are marked, or unexpected obstacles appear on site. This part is identified through machine vision to add the non-existent results and avoid the omission of risk targets.

Since the risk targets are divided into real risks and possible risks, the existing risk targets such as vehicles and pedestrians are identified by the completed target detection algorithm, and risk weighting under space-time conditions is evaluated based on historical information.

Figures 1 and 2 show the accident rate in hours and a week respectively based on the data from Shanghai. The appropriate model is generated with specific data.

The accident rate varies greatly with time, and vice versa. Weighted value table based on accident ratio is shown in Table 3. With the average value of 1 for each type, the accident data of the whole year are used for statistics, and the weighted value of each time period is re-weighted.

Risks based on accident types are shown in Table 4. From the analysis of accident data, we can know the accident type, accident vehicle, weather, time period and other information of the accident according to the location. The accident type is the main sequence, which can be divided into rear-end, reversing door switch, traffic signal violation, non-yielding, other accidents and other types. The correlation between each accident type and risk target and risk type can be sorted. Also, the cause and party of the accident under this environment can be known at the same time.

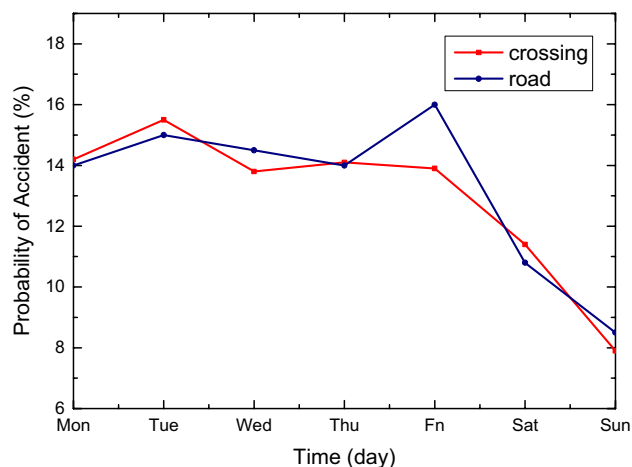


Figure 2. The accident incidence rate in weeks of crossing and road in shanghai are fitted by linear relationship. It can be seen from the figure that the overall trend is downward.

Day	Time								
	Type	01:00	02:00	08:00	09:00	10:00	11:00	12:00	18:00
Monday	Road	0.22	0.20	1.11	2.05	1.91	1.74	1.47	2.30
	Crossing	0.21	0.19	1.32	2.26	1.91	1.69	1.52	2.17
Tuesday	Road	0.24	0.22	1.21	2.24	2.09	1.90	1.60	2.51
	Crossing	0.28	0.21	1.42	2.42	2.04	1.81	1.63	2.33
Wednesday	Road	0.22	0.20	1.09	2.02	1.89	1.72	1.45	2.27
	Crossing	0.27	0.20	1.36	2.33	1.96	1.74	1.57	2.23
Thursday	Road	0.22	0.21	1.11	2.06	1.93	1.76	1.48	2.31
	Crossing	0.27	0.20	1.34	2.29	1.93	1.71	1.54	2.20
Friday	Road	0.22	0.20	1.08	2.01	1.87	1.71	1.44	2.25
	Crossing	0.29	0.22	1.46	2.50	2.11	1.87	1.68	2.40
Saturday	Road	0.17	0.16	0.87	1.60	1.50	1.37	1.15	1.80
	Crossing	0.20	0.14	0.99	1.68	1.42	1.26	1.13	1.62
Sunday	Road	0.13	0.12	0.63	1.16	1.09	0.99	0.83	1.31
	Crossing	0.16	0.12	0.81	1.38	1.17	1.03	0.93	1.33

Table 3. The table shows the probability of road and cross accidents from 1 a.m. to 6 p.m. from Monday to Sunday.

Accident type	Main risk types	Main causes of accidents
Rear-end	Car-car conflict/vehicle control	In front of the car brake/ground skid/visual blind area
Retrograde	Car-car/car-person conflict	A blind spot/sudden rush
Astern	Car-car/car-object conflict	A blind spot/sudden rush
Violation of traffic signal	Car-car/car-person conflict	The signal lamp is shielded/blind/the other party suddenly rushes out
Not yielding	Car-car/car-person conflict	Too fast

Table 4. The table reflects different types of accidents, main risks and main causes of accidents. The accident types are divided into back-end, retrograde, etc., the main risk types are vehicle conflict and vehicle control, etc., and the main causes of accidents are ground anti-skid, blind spots, etc.

According to the matching of current accident location, weather and time, the priority of high-risk accident objects is increased. According to the causes of accidents in the current environment, the hidden risks are added and sorted. Therefore, the final program will output the risk target list data with priority and hidden risks.

Target detection method based on depth learning. The models of deep learning trained are applied to identify and detect the sequence of captured images^{58,61–69}. And the algorithm is used to calculate the direction speed of the target and the distance to provide data for the next step.

The velocity prediction is realized by moving the European distance of the target center point between adjacent Dayton. In short, there is a correspondence between the speeds of the real world image. If the target speed in the real world is fast, the speed in the adjacent pictures will be the same. Therefore, the speed can be obtained by finding the corresponding relation between the real speed and the video image speed. According to the shooting time of adjacent images, the frame rate, the moving distance of the target center and the moving speed in the images can be calculated. Because speed is affected by distance and time, but time is the same for real world and images, the converted distance is the most critical. What's more, the conversion relation can be obtained by using the real size and the image size. For unmanned video images, the license plate can be selected for objects of general size. With the help of the license plates width and the actual width in the image, the conversion ratio is obtained, thus obtaining the real distance and the real speed. And the relative velocity estimation formula of the target is as follows.

Ratio of image to real world:

$$Scale = \frac{C}{C'} \tag{1}$$

Actual speed size:

$$Speed = d \times fps \times scale \tag{2}$$

where C is the real license plate size, C' is the size of the object in the picture, d is the euclidean geometric distance of the target moving in the image determined by the displacement of the center point, and fps is the frame rate.

Since the velocity is vector, the velocity direction of the target should be obtained in addition to scalar. Firstly, image sequence groups within a period of time should be screened out. Secondly, the object center of the same target should be locked, and the moving direction of the object center in the sequence group should be determined to obtain the direction of instantaneous velocity.

For distance calculation, visual distortion and other issues should be considered first when CMOS sensors are used. The correction of matrix and camera internal parameters could be obtained by using Matlab camera calibration toolbox and calibration function of OpenCV library. The details will not be described here due to the length of the paper.

The system applies a fixed device to perform a single visual distance algorithm. Through conversions from real-world to camera coordinate, camera to image coordinate conversion and image to frame storage coordinate, the conversion from real world to frame storage coordinate is realized:

$$z_v \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & u_0 \\ 0 & s_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{3}$$

where (X, Y, Z) is the real world coordinate system, (X_v, Y_v, Z_v) is the camera coordinate system, (x_p, y_p) is the image coordinate, (s_x, s_y) is the unit of dividing pixels by millimeters, (u_0, v_0) is the origin of the fixed frame storage coordinate system, set any position to (u, v) , R is the 3×3 rotation matrix, T is the 3×1 translation matrix, and f is the camera focal length. It can be simplified again:

$$z_v \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & u_0 & 0 \\ 0 & f_y & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{4}$$

Finally reduced to:

$$Z_v \times P = M_1 \cdot M_2 \cdot P' \tag{5}$$

where P is the frame storage coordinates, P' is the real world coordinates, M_1 is the camera internal parameter matrix, M_2 is a camera position matrix.

Take the real situation as a profile of the Y-axis, set the P as the target, and project P_y on the Y-axis. After deduction, the distance formula can be obtained:

$$d = h / \tan \left(\arctan \frac{Q}{h} + \arctan \frac{H}{2f} + \arctan \frac{P_y - y_0}{f} \right) \tag{6}$$

Take Q as the distance from the camera to the nearest point below, h as the camera height, H as the camera head height, (x_0, y_0) is the midpoint coordinate of the image. Making coordinate system conversion:

$$d = h / \tan \left(\arctan \frac{Q}{h} + \arctan \frac{v}{2f_y} + \arctan \frac{v - v_0}{f_y} \right) \tag{7}$$

where v is the pixel height coordinate of the target in the image, and v_0 and f_y are internal parameters provided for calibration.

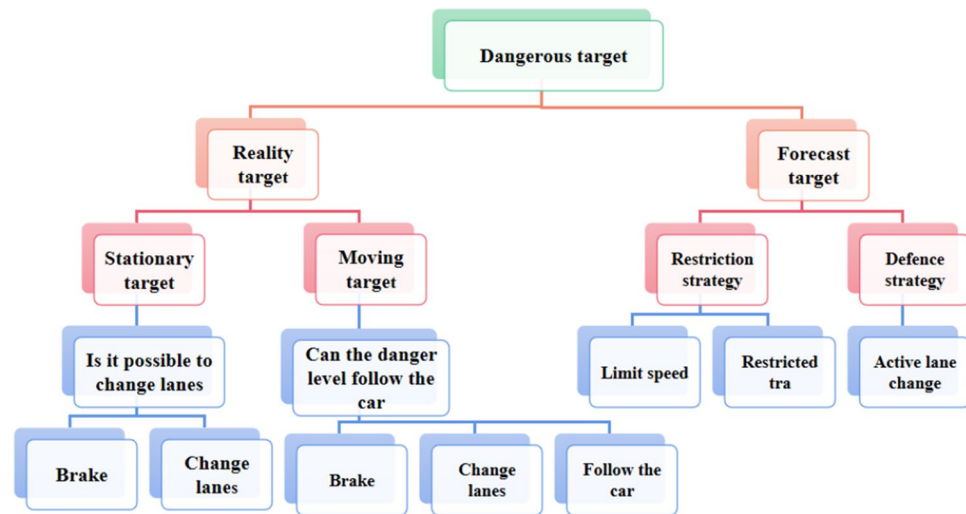


Figure 3. The risk model is used to give the prediction target and risk based on location and accident date, evaluate the target risk index, and then deal with it accordingly.

At the same time, according to its own speed calculation, some risk targets have been or will be on the collision path, and this kind of realistic risk targets are marked as the highest priority. In addition, the priority is arranged in turn according to the speed and distance of the target.

System construction. The prediction part first recognizes and perceives the environment, such as identifying intersections, lanes, parking lots, crosswalks, the vicinity of primary and secondary schools, etc., which is a risk model based on location and accident data. Secondly, the risk is evaluated according to historical data, that is, the risk model is used to give the prediction target and risk based on location and accident data. Finally, the target detection method based on deep learning is intended to detect the target and evaluate the risk index of the target. In a word, the system needs to solve the problems of "what is the current environment", "is there any risk in the environment", "what kind of risk is there", "the degree of danger of various risks" and "how to avoid it".

The trajectory of the risk target is predicted and tracked, and the braking distance is taken as the safe range for estimation. For hidden risks, the risk of ground skidding caused by weather will increase the braking distance, while the risk of line-of-sight problem assumes that objects with the same speed as the vehicle are located in the center of the shielding range, and estimates the safety index.

The parameters affecting the hazard value include the vehicle speed, braking performance, wet skid degree of the road surface and the direction of the risk target speed. Therefore, the hazard value should be obtained through comprehensive consideration of these parameters. According to relevant documents, when emergency braking is used to avoid collision, deceleration greater than 5 m/s^2 can be considered dangerous, 2 to 5 m/s^2 is critical danger, and below 2 m/s^2 —it can be considered safe. However, the road conditions will lead to a decrease in braking performance, which is reflected in the deceleration under the maximum braking effect, referred to as the maximum deceleration. Besides, the braking deceleration of any object should be less than the maximum, especially for objects already in the field of view. It should be considered as appropriate even for predicted objects that do not appear in the field of view. Therefore, the critical dangerous deceleration should also give priority to the environmental ground friction coefficient. Figure 3 demonstrates the internal process shown in the flow chart of circumvention algorithm.

Construction and demonstration of algorithm model based on deep learning

Algorithm construction of environmental perception and hazard prediction. The convolution neural network (CNN) and recursive neural network (RNN) are used to complete the task of environmental perception. The model based on LSTM variable gives different weights to different features. It can not only adapt to complex background, but also can deal with multiple targets. In addition, the end-to-end method of expression model proposed by Northwest University of science and technology in 2018 can be fully described.

Hazard prediction is divided into two parts: Target Detection and Hazard Degree prediction, in which target detection is the application scenario of deep learning. Compared with the traditional algorithm, the algorithm based on deep learning has obvious advantages in detection accuracy and efficiency. An improved SSD based target detection algorithm is proposed in this paper.

Extracting the feature information of important objects in traffic scene is the beginning of the work. Based on the supervised learning method, the attribute set is trained by multi-label classification, and the attribute prediction is carried out by training the deep convolution neural network corresponding to the loss function.

The supplementary description of environmental perception belongs to the category of image semantic recognition, and the method used belongs to the 'end-to-end'.

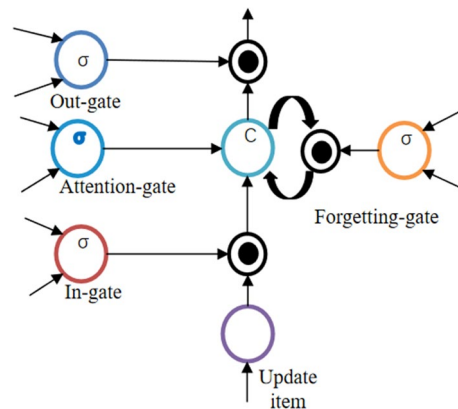


Figure 4. The storage unit regularly balances the content, and the replacement is controlled by four gates. Function based weight unit generation in gating phase. In addition, the hidden layer state and image features of the last node extracted by CNN are input to the unit, and the stimulation function is analyzed by machine vision.

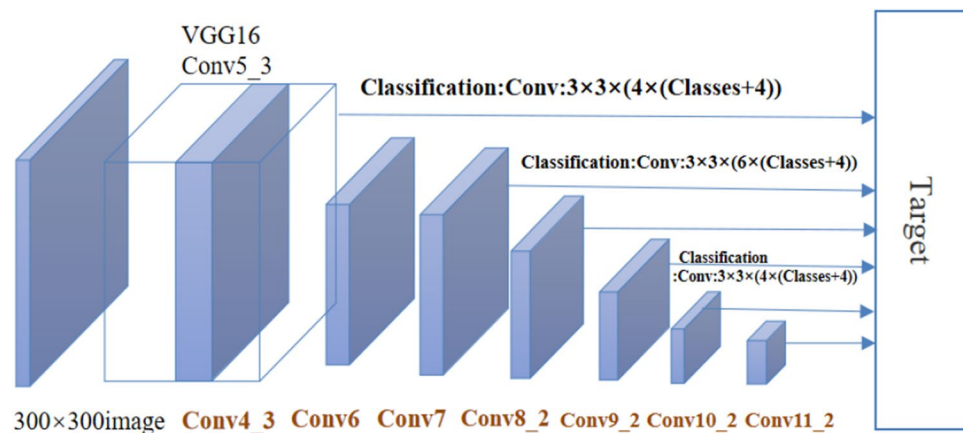


Figure 5. Using vgg-16 as the basic network, its extraction layer in the second half is also predicted, and detection is performed not only on other function diagrams, but also on the bottom convex 4–3 and 7 function points, so as to achieve compatibility with small goals.

The work of feature extraction is completed by CNN classification model. After classification, it is represented by LSTM, which is an RNN variant model. It is particularly important to note that the LSTM model is submitted not only to the extracted image features, but also to relevant information such as color, focus range of attention, location, and so on.. The feature of this method lies in dividing attention by color and weighting attention regions appropriately. However, the so-called color attention weight is to detect areas with relatively concentrated areas or large color changes of the same color in the image, especially for red and colors with significant contrast. By the way, the detection is realized by RGB color coding.

The description of model. LSTM is a special form of RNN network, whose structure has a storage unit for storing some events with certain intervals and delays in the training process. The storage unit shown in Fig. 4 regularly balances the content, and the trade-off is controlled by four gates. A feature-based weight unit is generated during the gate control phase. Besides, the hidden layer state of the previous node and the image features extracted by CNN are input to the unit, and the stimulation features are analyzed by machine vision.

During the encoding phase, pictures and labels exist as vectors in the hidden layer state. Each image extracts features with a trained VGG16 model. At the same time, the label vector is input into the LSTM model through matrix transformation. During the decoding phase, the maximum probability is obtained by multiplying the feature layer of the last layer of the hidden layer by the seventh layer of the fully connected layer. After the comparison, the output model considers the description to be the best match.

Theoretical framework of SSD. In the initial SSD paper, the following structure is presented. SSD is detected using the feature pyramid structure, which uses the characteristic Feature SAMP of conv 4–3, 6–2, 7, 7–2, 8–2, 9–2. At the same time, position regression and Softmax classification are performed. Figure 5 dem-

onstrates that SSD can use VGG-16 as the basic network. The feature extraction layer in the second half is also predicted. In addition, the detection is performed not only on the additional feature maps, but also on the underlying conv4-3 and 7-feature positions to achieve compatibility with small goals.

There are three core design concepts of SSD, as follows:

- There are two kinds of feature maps with multi-scale feature mapping: large feature mapping corresponds to small target and large target is responsible for small feature mapping.
- The feature map is extracted directly by convolution so that a large feature graph can be obtained with a relatively small convolution kernel.
- Setting a priori box each cell generates a priori box with different size, length and width. As the baseline of the bounding box, a priori frame generates multiple a priori frames in different ways during the training process.

Taking VGG16 as the basic model, the SSD transforms the fully-connected layer into $3 \times 33 \times 3$ convolution layer CONV6 and $1 \times 11 \times 1$ convolution layer CONV7, and pool5 from 2×2 to 3×3 . Then the FC8 and drain layers are replaced by a series of convolution layers, and fine-tuned using the detection set. The Conv4 layer with a size of 38×38 in VGG16 will serve as the first feature map for detection. However the layer data is too large to be normalized instead.

Five feature graphs were extracted from the new layers, namely Conv7, Conv8_2, Conv9_2, Conv10_2 and Conv11_2, and six original layers of conv4 are added. Their sizes are (38, 38), (19, 19), (10, 10), (5, 5), (3, 3), (1, 1), (38, 38), (19, 19), (10, 10), (5, 5), (3, 3), (1, 1). They have different priorities, including size, length and width. What's more, as the size of the feature map increases, the previous box size decreases.

The results are obtained by convoluting the feature graph: Category Confidence and bounding box position, each using $3 \times 33 \times 3$ convolution is completed, the essence of SSD is dense sampling.

Algorithm training and improvement. *Training.* Prior box matching. Before work, the prior frame matching with the target or part of the target is retrieved, and the matched boundary frame will enter the prediction phase. The first step of prior frame matching is to confirm Before work at least one frame to be identified. If it has a corresponding target, it becomes a positive sample, otherwise it will be a negative sample. Secondly, if there is a target matching degree greater than the threshold (generally 0.5) for the remaining negative sample, the sample will become a positive sample. Moreover, targets may have multiple prior frames that are not necessarily perfectly matched, but one prior frame cannot correspond to multiple targets.

Loss function. The loss function can be understood as the weighted sum of confidence and position error:

$$L(x, c, l, g) = \frac{1}{N} [L_{conf}(x, c) + \alpha L_{loc}(x, l, g)] \quad (8)$$

where N is the number of positive samples, $x_{ij}^p \in \{1, 0\}$ is used as an indication parameter, and when $x_{ij}^p = 1$, the (i -th) prior box matches the (j -th) target with category p . c is the category confidence prediction. And L is the predicted value of the position, which is the position of the boundary of the target selected in the prior frame, and g represents its position parameter. The position error in the loss function only considers positive samples, which is defined by smooth L_1 loss as follows:

$$L(x, l, g) = \sum_{i \in Pos} \sum_{m \in \{cx, cy, w, h\}} x_{ij}^k smooth_{L_1}(l_i^m - \hat{g}_j^m) \quad (9)$$

$$\hat{g}_j^{cx} = \frac{(g_j^{cx} - d_i^{cx})}{d_i^{cx}} \quad (10)$$

$$\hat{g}_j^{cy} = \frac{(g_j^{cy} - d_i^{cy})}{d_i^{cy}} \quad (11)$$

$$\hat{g}_j^w = \log \frac{(g_j^w)}{d_i^w} \quad (12)$$

$$\hat{g}_j^h = \log \frac{(g_j^h)}{d_i^h} \quad (13)$$

$$smooth_{L_1}(x) = \begin{cases} 0.5x^2 & \text{if } |x| < 1 \\ |x| - 0.5 & \text{otherwise} \end{cases} \quad (14)$$

The parameters are as follows:

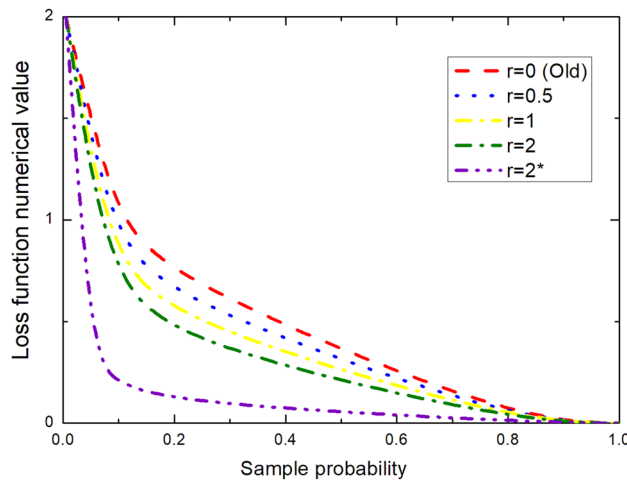


Figure 6. This figure fits the linear relationship between loss function and sample probability when $r=0$, $r=0.5$, $r=1$, etc. Through it, we can clearly see the relationship between lost function numerical value and sample probability.

$$\hat{g}_j^{cx} = \frac{(g_j^{cx} - d_i^{cx})}{\text{variance}} \tag{15}$$

$$\hat{g}_j^{cy} = \frac{(g_j^{cy} - d_i^{cy})}{\text{variance}} \tag{16}$$

$$\hat{g}_j^w = \frac{\log\left(\frac{g_j^w}{d_i^w}\right)}{\text{variance}} \tag{17}$$

$$\hat{g}_j^h = \frac{\log\left(\frac{g_j^h}{d_i^h}\right)}{\text{variance}} \tag{18}$$

For confidence error, it adopts softmax loss:

$$L_{conf}(x, c) = -\sum_{i \in Pos} x_{ij}^p \log\left(\frac{\exp(c_i^p)}{\sum_p \exp(c_i^p)}\right) - \sum_{i \in Neg} \log(\hat{c}_i^0) \tag{19}$$

Improvement based on focal loss. The main reason why single-level detection is not as accurate as two-level detection is the imbalance of sample categories. Category imbalance will bring too many negative samples, which account for most of the loss function. Therefore, the focal loss is proposed as a new loss function. The loss function is modified According to the standard cross entropy loss in Fig. 6. This function can reduce the samples that are easy to classify by changing the evaluation method, so as to apply more weights to the samples that are difficult to classify in the training process. The formula is as follows:

$$FL(p_t) = -\alpha t(1 - p_t)^\gamma \log(p_t) \tag{20}$$

Firstly, a factor is added to the original standard cross entropy loss, thereby reducing the loss of easily classified samples. This makes us pay more attention to difficult and misclassified samples. For example, $\gamma=2$, for a positive sample with a prediction result of 0.95, value of the loss function becomes smaller because the power of $(1-0.95)$ is very small. However, for negative samples with a prediction probability of 0.3, the loss becomes relatively large, which is achieved by suppressing the loss of positive samples.

Therefore, the new method pays more attention to this indistinguishable sample. In this way, the influence of simple samples is reduced, and the effect will be more effective only when a large number of samples with low prediction probability are added together. Meanwhile more penalties are required for easily distinguishable negative samples. The actual formula is as follows:

$$L_{fl} = \begin{cases} -\alpha (1 - y')^\gamma \log y' & y = \text{postive simple} \\ -(1 - \alpha)y'^\gamma \log(1 - y') & y = \text{negtive simple} \end{cases} \tag{21}$$

In the experiment, $\gamma = 2$ and $\alpha = 0.25$ have the best effect.

Improvement based on KL loss. The traditional boundary box regression loss (i.e., Smooth L1 loss) does not take the deviation of the actual boundary on the ground into consideration. When the classification score is very high, the regression function is considered to be accurate, but it is not always the case.

Bounding box prediction is modeled as Gaussian distribution, and the boundary box of positive samples is modeled as a Dirac delta function. The asymmetry of these two distributions is measured by KL divergence. When KL divergence approaches 0, these two distributions are very similar. KL loss is the KL divergence of minimizing the Gaussian distribution predicted by the bounding box and the Dirac delta distribution of positive samples. In other words, KL loss makes the bounding box prediction approximate gaussian distribution and close to positive samples. And it converts the confidence into the standard deviation of the bounding box prediction.

The two probability distributions P and Q of a discrete or continuous random variable whose KL divergence is defined as:

$$D(P\|Q) = \sum_{i \in X} P(i) * \left[\log \left(\frac{P(i)}{Q(i)} \right) \right] \tag{22}$$

$$D(P\|Q) = \int_x P(x) * \left[\log \left(\frac{P(x)}{Q(x)} \right) \right] dx \tag{23}$$

Before calculating KL divergence, the bounding box needs to be parameterized. (x_1, y_1, x_2, y_2) is the upper left and lower right coordinates of the prediction bounding box. $(x_1^*, y_1^*, x_2^*, y_2^*)$ is the coordinates of the upper left and lower right corners of the real box. $(x_{1a}, y_{1a}, x_{2a}, y_{2a}, h_a, w_a)$ is an anchored bounding box generated by aggregating all real boxes. Then the deviations of the predicted and real bounding boxes are as follows:

$$t_{x1} = \frac{x_1 - x_{1a}}{w_a}, t_{x2} = \frac{x_2 - x_{2a}}{w_a} \tag{24}$$

$$t_{y1} = \frac{y_1 - y_{1a}}{h_a}, t_{y2} = \frac{y_2 - y_{2a}}{h_a} \tag{25}$$

$$t_{x1}^* = \frac{x_1^* - x_{1a}}{w_a}, t_{x2}^* = \frac{x_2^* - x_{2a}}{w_a} \tag{26}$$

$$t_{y1}^* = \frac{y_1^* - y_{1a}}{h_a}, t_{y2}^* = \frac{y_2^* - y_{2a}}{h_a} \tag{27}$$

Similarly, the parameter without * indicates the deviation between the prediction and the anchored boundary frame, and the parameter with * indicates the deviation between the real and the anchored boundary frame.

Assuming that the coordinates are independent, a univariate Gaussian function is used for simplicity:

$$P_\Theta(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-x_e)^2}{2\sigma^2}} \tag{28}$$

where x_e is the estimated boundary box position and the standard deviation σ is the estimated uncertainty. When $\sigma \rightarrow 0$, the position accuracy of boundary box is very high.

The real boundary box on the ground can also be expressed by Gaussian distribution, and becomes Dirac delta function when $\sigma \rightarrow 0$:

$$P_D(x) = \delta(x - x_g) \tag{29}$$

where x_g is the real boundary box position on the ground. At this point, we can construct a bounding box regression function with KL loss, and establish a formula to minimize the KL error of $P_\Theta(x)$ and $P_D(x)$ on N samples:

$$\hat{\Theta} = \underset{\Theta}{\operatorname{argmin}} \frac{1}{N} \sum D_{KL}(P_D(x)\|P_\Theta(x)) \tag{30}$$

KL divergence is used as the loss function L_{reg} for bounding box regression, and the classification loss L_{cls} remains unchanged. For a single sample:

$$\begin{aligned} L_{reg} &= D_{KL}(P_D(x)\|P_\Theta(x)) \\ &= \int P_D(x) \log P_D(x) dx - \int P_D(x) \log P_\Theta(x) dx \\ &= \frac{(x_g - x_e)^2}{2\sigma^2} + \frac{\log(\sigma^2)}{2} + \frac{\log(2\pi)}{2} - H(P_D(x)) \end{aligned} \tag{31}$$

Ways	BLEU-1	BLEU-2	BLEU-3	BLEU-4
Google NIC	63	41	27	–
Log bilinear	65.6	42.4	27.7	17.7
The design algorithm	67.2	45.8	32.1	18.9

Table 5. The results of the experiments on the database Flickr8K. Methods are divided into Google NIC, Log Bilinear and The design algorithm.

Ways	BLEU-2	BLEU-3	BLEU-4
Google NIC	42.3	27.7	18.3
Log bilinear	38	25.4	17.1
The design algorithm	45.1	29.9	21.1

Table 6. The results of the experiments on the database Flickr30K. Methods are divided into Google NIC, Log Bilinear and The design algorithm.

Ways	BLEU-1	BLEU-2	BLEU-3	BLEU-4
Google NIC	66.6	46.1	32.9	24.6
Log Bilinear	70.8	48.9	34.4	24.3
The design algorithm	72.3	51.8	37.1	25.1

Table 7. The results of the experiments on the database MS COCO. Methods are divided into Google NIC, Log Bilinear and The design algorithm.

When the prediction of the bounding box is inaccurate, because the prediction closer to the real bounding box is certainly stable and its variance is small, the smallest possible variance can reduce L_{reg} . After the variance of the predicted position of the bounding box is obtained, the candidate positions are voted according to the known variance of adjacent bounding boxes. Besides, the candidate coordinate values with the largest score are selected to be weighted to update the coordinates of the bounding box, so as to make the positioning more accurate. What's more, border boxes with lower positions and lower colors have higher weights. The new coordinates are calculated as follows:

$$p_i = e^{-(1-IoU(b_i,b))^2/\sigma_t} \quad \text{subject to } IoU(b_i, b) > 0$$

$$x = \frac{\sum_i p_i x_i / \sigma_{x,i}^2}{\sum_i p_i / \sigma_{x,i}^2} \quad (32)$$

where σ_t is an adjustable parameter for variable voting. When $IoU(b_i, b)$ is larger, p_i is larger, the two bounding boxes overlap each other more and do the same for the remaining coordinate values. SSD detects the generated preselected box computing loss through FL loss function classification and boundary regression. Besides, the boundary regression of SSD is improved based on KL loss method. Frames with large variance and adjacent boundary frames containing the selected frames but too small will get low scores when voting. Moreover, the SSD algorithm can effectively avoid the above anomalies by variance voting instead of IoU overlap degree.

Model testing and analysis. The environment perception is divided into two parts, the micro part is the main perception of the scene by machine vision, which is used to confirm and supplement the macro and micro perception.

First of all, we tested the *Roi* weighting using live campus photos taken on May 7, 2020. The advantage of this algorithm is that the region of interest can be identified first, and then the further perception can be completed. Therefore, the region of interest test was performed first, and the effect of attention weighting was significant.

Second, the environment perception test was carried out because the region of interest was weighted and the weighted region was described firstly. After testing, the algorithm can complete the perception of the simple traffic scene and recognize the red light of the intersection, the bus and the right-turn sign on the road, and can supplement and confirm the environment perception part.

At the same time, different databases, Google Nic, Log BILINEAR and other different algorithms are used to compare with experiments, because the Algorithm has good performance on Flickr8K, Flickr30K and MS COCO databases, and validated the experimental results of the Northwestern Polytechnic University team. The experimental results on the Flickr8K database are shown in Table 5, Flickr30K database are shown in Table 6, and MS COCO database are shown in Table 7.

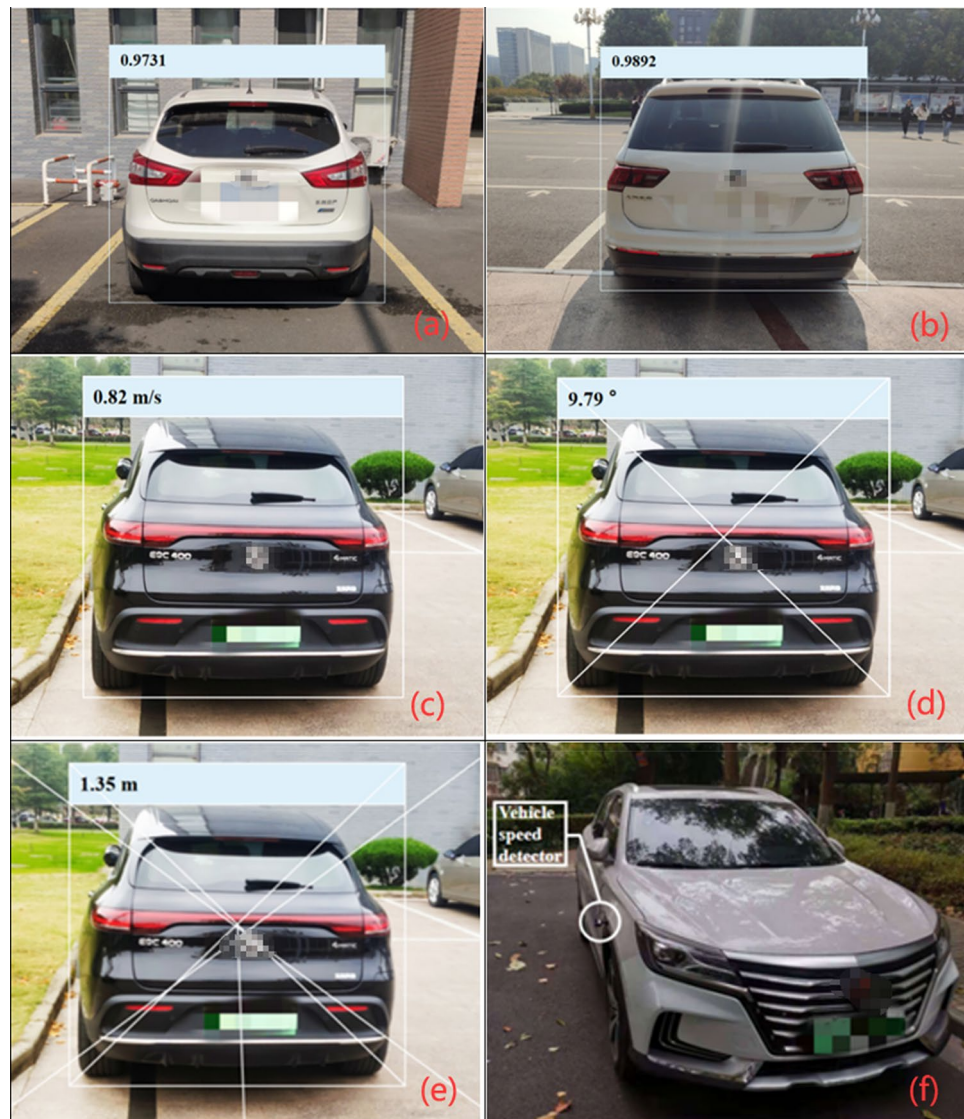


Figure 7. (a, b) Detect the dynamic vehicle, including its speed and driving direction. As shown in the figure, the distance between the vehicle and the parking space line is 0.9731 m and 0.9892 m respectively. (c, d) Dynamic vehicle detection mainly detects its driving speed and direction. As shown in the figure, the vehicle speed is 0.82 m/s and the driving direction is 9.79° to the left. (e, f) The figure shows that the vehicle speed detector is used to detect the actual vehicle and can accurately detect its driving speed and direction.

The focus will be on target detection in the hazard prediction section. First of all, the vehicle test is carried out by using field test maps and data set pictures. Secondly, dynamic vehicles need to be detected, including their speed, distance and running direction. The vehicle target detection is shown in Fig. 7a and b. The dynamic vehicle direction estimation is shown in Fig. 7c and d. The dynamic vehicle distance estimation is shown in Fig. 7e. The vehicle speed detector is used to detect the speed of the dynamic vehicle in Fig. 7f.

Results and discussion

Simulated route. In order to better reflect the function of the system, the paper uses Matlab and CarSim to set up the dangerous situation of vehicle crossing under different conditions and conduct a joint simulation. The simulation system will output the speed constraint throughout the whole simulated driving process. Figure 8a simulates the vehicle operation by adjusting the scene, road surface and definition, driving conditions, etc. Figure 8b shows the speed constraint of the simulation system output in the whole simulation driving process.

Choose the route from school to the bus station. The path passes through two campuses, two residential areas, a commercial center and four intersections. The total length of this path is 5.6 km, which can meet the needs of system test and simulation. In order to facilitate the simulation, the path of latitude and longitude are sampled. In addition, the path can be divided into two paths, Tianshan Road and Youth road, and the results are shown in Tables 8 and 9 respectively.

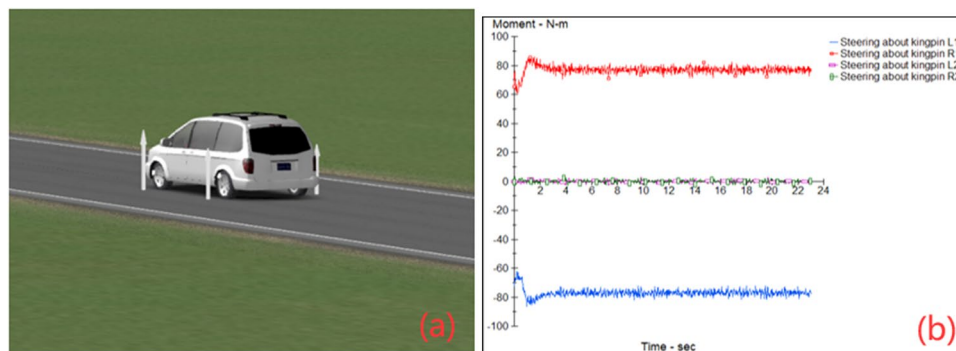


Figure 8. Simulation interface of CarSim. (a) Simulate vehicle operation by adjusting scene, road surface and definition, driving conditions, etc. (b) It shows the speed constraints of the simulation system output in the whole simulation driving process.

Positions	Latitude and longitude coordinates	Remarks
Starting point	120.205976, 33.37899	West Gate of Yancheng Normal University's New Long Campus
Tianshan road	120.20524, 33.37991	-
Crossroads	120.204431, 33.380875	Century avenue and Tianshan Road Intersect
Tianshan road	120.202715, 33.382887	-
Yancheng institute of technology	120.201997, 33.38361	-
Tianshan road	120.200865, 33.384786	-
Tianshan road	120.198403, 33.387544	Unknown path junction
Xinfeng community	120.198403, 33.387544	-
Crossroads	120.195753, 33.390318	Tianshan Road and Youth Road Intersect

Table 8. The table mainly includes location, longitude and latitude coordinates and remarks. The location is divided into starting point and Tianshan Road. The longitude and latitude coordinates are consistent with the location.

Positions	Latitude and longitude coordinates	Remarks
Youth road	120.193714, 33.389948	-
Tongyu River Bridge	120.190229, 33.388803	-
Tongyu River Bridge	120.186968, 33.38768	There is a side road junction
Youth road	120.181596, 33.385909	There is a side road junction
Complex intersection	120.180096, 33.385344	Youth Road and Fangong Road Intersect
Howard square	120.174985, 33.382133	-
Crossroads	120.173907, 33.381576	Youth Road and Wengang Road Intersect
Youth road	120.176206, 33.382812	-
End	120.178425, 33.384033	Tianshan Road and Youth Road Intersect

Table 9. The table mainly includes location, longitude and latitude coordinates and remarks. The location is divided into Youth road and Tongyu River Bridge. The longitude and latitude coordinates are consistent with the location.

Latitude and longitude sampling table of Tianshan Road is shown in Table 10. Longitude and latitude sampling table of Youth Road is shown in Table 11. Based on the collected coordinates of latitude and longitude, the whole system can be simulated and tested. In Matlab, the REGEXP function can be used to get a Web page, so as to get the location name directly through the map API and the output environment data through Json. Then the starting latitude and longitude for the test are selected to successfully obtain the remote data.

System simulation. In order to facilitate the simulation of the system function, the speed constraint on the simulation path is visualized. Considering the unity of safety and efficiency, the time has a great influence on speed constraint. Assuming that the vehicle is traveling at 60 km/h, simulation speed constraints are provided on Monday, Sunday at 8:00 and Monday at 8:00 and 23:00. By the way, the system will adjust appropriately according to the risk weighting.

Positions	Latitude and longitude coordinates	Remarks
Starting point	120.205976, 33.37899	West Gate of Yancheng Normal University's New Long Campus
Tianshan road	120.20524, 33.37991	-
Crossroads	120.204431, 33.380875	Century avenue and Tianshan Road Intersect
Tianshan road	120.202715, 33.382887	-
Yancheng institute of technology	120.201997, 33.38361	-
Tianshan road	120.200865, 33.384786	-
Tianshan road	120.198403, 33.387544	Unknown path junction
Xinfeng community	120.198403, 33.387544	-
Crossroads	120.195753, 33.390318	Tianshan Road and Youth Road Intersect

Table 10. The table mainly includes location, longitude and latitude coordinates and remarks. The location is divided into Starting point and Tianshan road. The longitude and latitude coordinates are consistent with the location.

Positions	Latitude and longitude coordinates	Remarks
Youth road	120.193714, 33.389948	-
Tongyu River Bridge	120.190229, 33.388803	-
Tongyu River Bridge	120.186968, 33.38768	There is a side road junction
Youth road	120.181596, 33.385909	There is a side road junction
Complex intersection	120.180096, 33.385344	Youth Road and Fangong Road Intersect
Howard square	120.174985, 33.382133	-
Crossroads	120.173907, 33.381576	Youth Road and Wengang Road Intersect
Youth road	120.176206, 33.382812	-
End	120.178425, 33.384033	Tianshan Road and Youth Road Intersect

Table 11. The table mainly includes location, longitude and latitude coordinates and remarks. The location is divided into Youth road and Tongyu River Bridge. The longitude and latitude coordinates are consistent with the location.

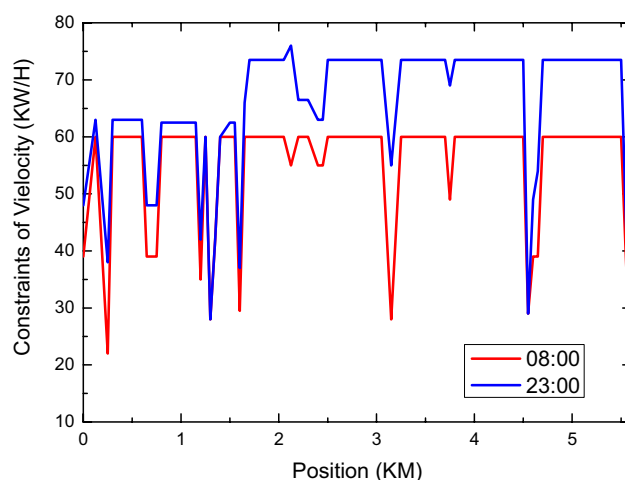


Figure 9. This figure reflects the constraints of velocity between 8 a.m. and 11 p.m. From the figure, we can see that the value of 11 p.m. is greater than that of 8 a.m.

The speed constraint was loaded into CarSim for dynamic simulation, and the data at 23:00 on Monday was selected to check the difference between simulations with and without the system.

What's more, the system will adjust appropriately according to the risk weighting. The speed constraint was loaded into CarSim for dynamic simulation, and the data at 23:00 on Monday was selected to check the difference between simulations with and without the system.

The comparison of speed constraints between Monday and Sunday is shown in Fig. 9. On the whole, the speed constraints on Monday are stricter than those on Sunday, which is caused by risk weighting based on experience.

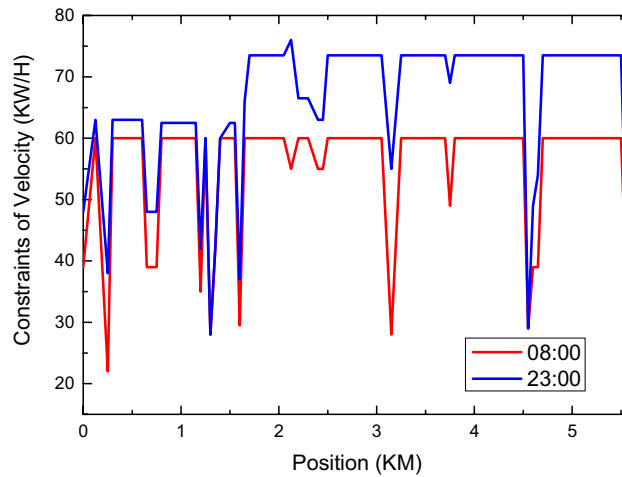


Figure 10. This figure reflects the change law of speed between 8 a.m. and 11 p.m. it can be seen from this figure that the speed limit at 11 p.m. is significantly greater than that at 8 a.m.

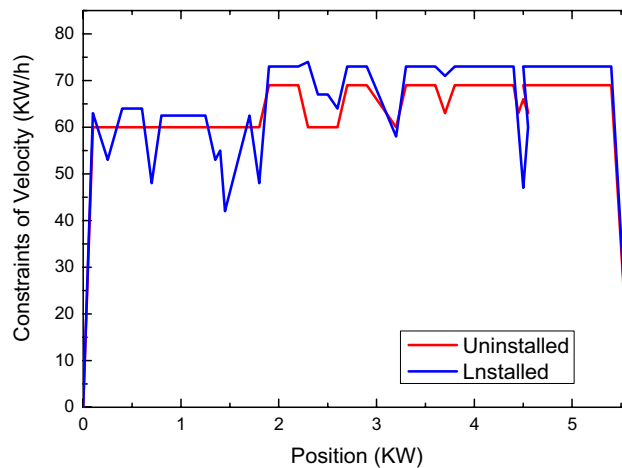


Figure 11. This figure reflects the change law of speed limit between unloading and installation points. It can be seen from this figure that the change of unloading speed limit is stable and the speed limit fluctuates greatly during installation.

And on the basis of time weighting, roads and intersections of different levels are weighted simultaneously by the system, and corresponding speed constraints are finally formed. At this time, the speed constraint does not consider the vehicle dynamics or the comfort of driving. In practical application, the speed constraint needs to consider the acceleration required of the current speed of the vehicle to implement the speed constraint. The acceleration needs to be comprehensively considered according to the center of gravity, braking performance, acceleration performance, ground friction coefficient, etc., which are ignored during the simulation.

The comparison results of speed constraints at different times on the same day are shown in Fig. 10. The speed constraint at 23:00 has been relaxed and vehicles are allowed to travel beyond the standard speed. In practical application, the unmanned driving system needs to combine the road supervision situation with the traffic situation on site to execute the speed. This speed only outputs speed constraints from the perspective of environmental hazards and does not represent the final execution speed.

The comparison of simulated driving speeds between vehicles equipped with the system and human-driven vehicles is shown in Fig. 11. Since most of the front part of the simulated route passes schools and intersections while other parts are expressways with few intersections, different driving speeds are simulated on the basis of actual driving. Under the ideal condition of smooth traffic, human driving vehicles will be affected by road grade, traffic control and subjective judgment. Besides, the driving speed of vehicles equipped with this system is similar to that of human beings in trend, and the speed constraint is strictly implemented according to the risk grade. It can be seen that the vehicles can realize the defensive driving of human beings more intelligently and flexibly, relying on accurate scientific and objective data analysis conclusions instead of subjective experience.

In order to reflect the efficiency of the system more particularly, the application test of the vehicle with or without the system is carried out through the car accident simulation built in CarSim.

The content of the traffic accident simulation is that an oncoming vehicle with a speed exceeding 100 km/h strays into the lane while avoiding the normally running vehicle and eventually rolls over. The normally running vehicle with a speed of 100 km/h completes emergency braking during the avoidance. The accident site is a freeway.

In the CarSim simulation, the speed of 70 km/h is set as the normal driving speed, which is consistent with the actual use of expressway. Within a period of visible sight distance, vehicles were not observed to enter the lane in opposite direction. The collision was avoided by emergency braking. The whole braking process has a great influence on passengers and uncertainty factors. The danger has been successfully avoided. The simulation results indicate the effectiveness of the system.

Conclusion

Environmental hazard prediction and avoidance technology is the key in the research field of unmanned vehicles, which provides an important guarantee for the driving of unmanned vehicles in the real environment. Nowadays, most unmanned driving systems are equipped with hazard prediction and avoidance systems. However, environment-oriented data-based environmental hazard prediction and avoidance technology has not been developed enough. In this paper, Matlab and CarSim are used to simulate the entire system. The speed constraints and simulation speed diagrams under various condition are output on the selected drive path to verify the effectiveness of the system function. The system is innovative to solve the problems of unmanned environmental hazard in the target detection. The next work is to experiment with hyper-parameter tuning and model training by real-world observations in the further research.

Received: 21 July 2021; Accepted: 31 May 2022

Published online: 17 June 2022

References

- Zhang, X. Y. *et al.* A study on key technologies of unmanned driving. *CAAI Trans. Intell. Technol.* **1**, 4–13 (2016).
- Kim, B. *et al.* Automated complex urban driving based on enhanced environment representation with GPS/map, radar, lidar and vision. *8th IFAC-AAC-11*. vol. **49** (IFAC, 2016).
- Li, Y. J. *et al.* Research on static decoupling algorithm for piezoelectric six axis force/torque sensor based on LSSVR fusion algorithm. *Mech. Syst. Signal. Pr.* **110**, 509–520 (2018).
- Zhang, R. *et al.* Electric vehicles' energy consumption estimation with real driving condition data. *Transp. Res. D* **41**, 177–187 (2015).
- Ye, C. *et al.* Landslide detection of hyperspectral remote sensing data based on deep learning with constrains. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **12**, 5047–5060 (2019).
- Liang, J. J. *et al.* A novel multi-segment feature fusion based fault classification approach for rotating machinery. *Mech. Syst. Signal Pr.* **122**, 19–41 (2019).
- Wang, F. *et al.* A novel integrated approach for path following and directional stability control of road vehicles after a tire blow-out. *Mech. Syst. Signal Pr.* **93**, 431–444 (2017).
- Hasnana, K. *et al.* JOMS: System architecture for telemetry and visualization on unmanned vehicle. *Procedia Eng.* **29**, 3899–3903 (2012).
- Li, Z. Z. *et al.* Influence of distance from traffic sounds on physiological indicators and subjective evaluation. *Transp. Res. D* **87**, 102538 (2020).
- Ryder, B. *et al.* Spatial prediction of traffic accidents with critical driving events- Insights from a nationwide field study. *Transp. Res. A*. **124**, 611–626 (2019).
- Yoshitake, H. & Shino, M. Risk assessment based on driving behavior for preventing collisions with pedestrians when making across-traffic turns at intersections. *IATSS Res.* **42**, 240–247 (2018).
- Ahmed Hassan, R. *et al.* A big data modeling approach with graph databases for SPAD risk. *Saf. Sci.* **110**, 75–79 (2018).
- Freeman, B. S. *et al.* Vehicle stacking estimation at signalized intersections with unmanned aerial systems. *IJTST*. **8**, 231–249 (2019).
- Chen, Y. W. *et al.* An effective infrared small target detection method based on the human visual attention. *Infrared Phys. Technol.* **95**, 128–135 (2018).
- Biswas, D. *et al.* An automatic traffic density estimation using single shot detection (SSD) and MobileNet-SSD. *Phys. Chem. Earth A/B/C*. **110**, 509–520 (2019).
- Gupta, A. *et al.* Deep learning for object detection and scene perception in self-driving cars: Survey, challenges, and open issues. *Array*. **10**, 100057 (2021).
- Feyzi, F. *et al.* FPA-FL: Incorporating static fault-proneness analysis into statistical fault localization. *J. Syst. Softw.* **136**, 39–58 (2018).
- Březina, J. *et al.* Fast algorithms for intersection of non-matching grids using Plücker coordinates. *Comput. Math. Appl.* **74**, 174–187 (2017).
- Burge, R. *et al.* An investigation of the effect of texting on hazard perception using fuzzy signal detection theory (fSDT). *Transp. Res. F* **58**, 123–132 (2018).
- Grumert, E. F. *et al.* Using connected vehicles in a variable speed limit system. *Transp. Res. Procedia*. **27**, 85–92 (2017).
- El-Gamal, A. & Saleh, I. Radiological and mineralogical investigation of accretion and erosion coastal sediments in Nile Delta region. *Egypt. J. Oceanogr. Mar. Sci.* **3**, 41–55 (2012).
- Zhang, C. *et al.* Identifying and mapping individual plants in a highly diverse high-elevation ecosystem using UAV imagery and deep learning. *ISPRS J. Photogramm. Remote. Sens.* **169**, 280–291 (2020).
- Zhu, X. X. *et al.* Deep learning in remote sensing: a comprehensive review and list of resources. *IEEE Geosci. Remote Sens. Mag.* **5**, 8–36 (2017).
- Cui, B., Fei, D., Shao, G., Lu, Y. & Chu, J. Extracting Raft aquaculture areas from remote sensing images via an improved U-Net with a PSE structure. *Remote Sens. Basel*. **11**, 2053 (2019).
- Ba, Y. T. *et al.* Crash prediction with behavioral and physiological features for advanced vehicle collision avoidance system. *Transp. Res. C*. **74**, 22–23 (2017).
- Moshayedi, A. J. *et al.* Portable image based moon date detection and declaration: System and Algorithm Code Sign. *2019 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications*. 45640, (CIVEMSA, 2019).
- Moshayedi, A. J. *et al.* Kinect based virtual referee for table tennis game: TTV (Table Tennis Var System). *6th International Conference on Information Science and Control Engineering*. 48695, (ICISCE, 2019).

28. Moshayedi, A. J. *et al.* WiFi based massager device with node MCU through arduino interpreter. *J. Simul. Anal. Novel Technol. Mech. Eng.* **11**, 73–79 (2018).
29. Moshayedi, A. J. *et al.* Mission and obstacles in design and performance. *J. Simul. Anal. Novel Technol. Mech. Eng.* **12**, 5–18 (2019).
30. Moshayedi, A. J. *et al.* Simulation study and PID Tune of automated guided vehicles (AGV). *2021 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications*. 52099, (CIVEMSA, 2021).
31. Moshayedi, A. J. *et al.* PID tuning method on AGV (automated guided vehicle). *J. Simul. Anal. Novel Technol. Mech. Eng.* **12**, 53–66 (2019).
32. Penman, T. D. *et al.* Improved accuracy of wildfire simulations using fuel hazard estimates based on environmental data. *J. Environ. Manage.* **301**, 113798 (2022).
33. Zheng, H. *et al.* Reliability analysis of products based on proportional hazard model with degradation trend and environmental factor. *Reliab. Eng. Syst. Saf.* **216**, 107964 (2021).
34. Saha, A. *et al.* Modelling multi-hazard threats to cultural heritage sites and environmental sustainability: The present and future scenarios. *J. Clean. Prod.* **320**, 128713 (2021).
35. Van Fan, Y. *et al.* Forecasting plastic waste generation and interventions for environmental hazard mitigation. *J. Hazard. Mater.* **424**, 127330 (2022).
36. Dey, P. *et al.* Hybrid CNN-LSTM and IoT-based coal mine hazards monitoring and prediction system. *Process Saf. Environ. Prot.* **152**, 249–263 (2021).
37. Crundall, D. *et al.* A novel driving assessment combining hazard perception, hazard prediction and theory questions. *Accid. Anal. Prev.* **149**, 105847 (2021).
38. Yang, J. *et al.* Driving assistance system based on data fusion of multisource sensors for autonomous unmanned ground vehicles. *Comput. Netw.* **192**, 108053 (2021).
39. Liu, H. Introduction of the train unmanned driving system. *Unmanned Driv. Syst. Smart Trains* **1**, 1–45 (2021).
40. Brown, D. *et al.* Detecting firmware modification on solid state drives via current draw analysis. *Comput. Secur.* **102**, 102149 (2021).
41. Oduncu, E. *et al.* An in-depth analysis of hyperspectral target detection with shadow compensation via LiDAR. *Signal Process. Image Commun.* **99**, 116427 (2021).
42. Ding, L. *et al.* Detection and tracking of infrared small target by jointly using SSD and pipeline filter. *Dig. Signal Process.* **110**, 102949 (2021).
43. Huang, Z. *et al.* Mobile phone component object detection algorithm based on improved SSD. *Procedia Comput. Sci.* **183**, 107–114 (2021).
44. Bai, G. *et al.* An intelligent water level monitoring method based on SSD algorithm. *Measurement* **185**, 110047 (2021).
45. Li, Y. *et al.* Multi-block SSD based on small object detection for UAV railway scene surveillance. *Chin. J. Aeronaut.* **33**, 1747–1755 (2021).
46. Tripathy, S. *et al.* SSD internal cache management policies: A survey. *J. Syst. Architect.* **19**, 102334 (2021).
47. Qin, J. *et al.* Reaserch and implementation of social distancing monitoring technology based on SSD. *Procedia Comput. Sci.* **183**, 768–775 (2021).
48. Ahmed, I. *et al.* IoT-based crowd monitoring system: Using SSD with transfer learning. *Comput. Electr. Eng.* **93**, 107226 (2021).
49. Lin, W. *et al.* Fast, robust and accurate posture detection algorithm based on Kalman filter and SSD for AGV. *Neurocomputing* **316**, 306–312 (2021).
50. Luo, Q. *et al.* 3D-SSD: Learning hierarchical features from RGB-D images for amodal 3D detection. *Neurocomputing* **378**, 364–374 (2020).
51. Sun, X. *et al.* A modified SSD method for electronic components fast recognition. *Optik* **205**, 163767 (2020).
52. Jang, Y. *et al.* Registration-free Face-SSD: Single shot analysis of smiles, facial attributes, and affect in the wild. *Comput. Vis. Image Understand.* **182**, 17–29 (2019).
53. Tseng, K. *et al.* Semi-supervised image depth prediction with deep learning and binocular algorithms. *Appl. Soft Comput.* **92**, 106272 (2020).
54. Chen, L. *et al.* Employing deep learning for automatic river bridge detection from SAR images based on Adaptively effective feature fusion. *Int. J. Appl. Earth Observ. Geoinfor.* **102**, 102425 (2021).
55. Lee, J. *et al.* Preserving copyright in renovating large-scale image smudges based on advanced SSD and edge confidence. *Optik* **156**, 606–618 (2018).
56. He, W. *et al.* Combining species sensitivity distribution (SSD) model and thermodynamic index (exergy) for system-level ecological risk assessment of contaminants in aquatic ecosystems. *Environ. Int.* **133**, 105275 (2019).
57. Lee, J. *et al.* Preserving copyright in renovating large-scale image smudges based on advanced SSD and edge confidegnce. *Optik* **140**, 887–899 (2017).
58. Lu, L. *et al.* A comprehensive risk evaluation method for natural gas pipelines by combining a risk matrix with a bow-tie model. *J. Nat. Gas Sci. Eng.* **25**, 124–133 (2015).
59. Zhao, H. Risk assessment method combining complex networks with MCDA for multi-facility risk chain and coupling in UUS. *Tunn. Undergr. Space Technol.* **119**, 104242 (2022).
60. Li, D. *et al.* Fast detection and location of longan fruits using UAV images. *Comput. Electron. Agric.* **190**, 106465 (2021).
61. Li, X. *et al.* Fast and accurate green pepper detection in complex backgrounds via an improved Yolov4-tiny model. *Comput. Electron. Agric.* **191**, 106503 (2021).
62. Wang, Q. *et al.* Pest24: A large-scale very small object data set of agricultural pests for multi-target detection. *Comput. Electron. Agric.* **175**, 105585 (2020).
63. Guo, Y. *et al.* Dense construction vehicle detection based on orientation-aware feature fusion convolutional neural network. *Autom. Constr.* **112**, 103124 (2020).
64. Zhankaziev, S. *et al.* Principles of creating range for testing technologies and technical solutions related to intelligent transportation systems and unmanned driving. *Transp. Res. Procedia.* **50**, 757–765 (2020).
65. Mafi-Gholam, D. *et al.* Spatial modeling of exposure of mangrove ecosystems to multiple environmental hazards. *Sci. Total Environ.* **740**, 140167 (2020).
66. NyunKim, H. *et al.* Derivation and validation of a combined in-hospital mortality and bleeding risk model in acute myocardial infarction. *IJC Heart Vasc.* **33**, 100732 (2021).
67. Li, Y. *et al.* Combined risk assessment method based on spatial interaction: A case for polycyclic aromatic hydrocarbons and heavy metals in Taihu Lake sediments. *J. Clean. Prod.* **328**, 129590 (2021).
68. Wang, C. *et al.* Dynamic risk analysis of offshore natural gas hydrates depressurization production test based on fuzzy CREAM and DBN-GO combined method. *J. Nat. Gas Sci. Eng.* **91**, 103961 (2021).
69. Choubin, B. *et al.* Earth fissure hazard prediction using machine learning models. *Environ. Res.* **179**, 108770 (2019).

Acknowledgements

This work is supported by the Jiangsu Natural Science Foundation of China [Project No. BK20211364], Jiangsu Province Intelligent Optoelectronic Devices and Measurement-Control Engineering Research Center Foundation

of China [Project No. 2022IODMCERC004] and Teaching Research Foundation of YanCheng Teachers University [Project No. 2021YCTCJGY017].

Author contributions

C.Q.Q. conducted experimental work; drafted manuscript; analysed data. Y.Z. assisted with measurements. J.J., S.Z., H.Z., S.Q.Z. assisted with target detection and analysis of data. M.Y.M. supervised and conceived study. All authors commented on and reviewed paper.

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to C.Q.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2022