Study on a Risk Model and Target Detection for Prediction and Avoidance of Unmanned Environmental Hazard

Chengqun Qiu (✉ ujsqcq@126.com )
Yancheng Teachers University

Yuan Zhong
Yancheng Teachers University

Jie Ji
Chinese Academy of Sciences

Shuai Zhang
Yancheng Teachers University

Hui Zhang
Yancheng Teachers University

Shiqiang Zhao
Yancheng Teachers University

Mingyu Meng
Tokyo Institute of Technology

Research Article

Keywords: environmental hazard prediction, unmanned driving system, target detection, depth learning algorithm, SSD target detection method, method of combining the risk model

DOI: https://doi.org/10.21203/rs.3.rs-738182/v1

License: ☒  This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License
Study on a risk model and target detection for prediction and avoidance of unmanned environmental hazard

Chengqun Qiu1,2,*, Yuan Zhong1, Jie Ji3, Shuai Zhang1, Hui Zhang1, Shiqiang Zhao1, Mingyu Meng4

1School of Clean Energy and Electrical Engineering, 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

Corresponding author at: School of Automotive and Traffic Engineering, Jiangsu University, Zhenjiang 212013, China
E-mail address: ujsqzcq@126.com (C.Qiu)

Abstract

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, the conservative driving strategy based on speed restriction is adopted according to the results of risk prediction. 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 unbalanced sample categories. Software such as MATLAB and Carsim are applied in the system. From the comparison results of the simulations of unmanned vehicles with or without a system, it that the system can provide effective safety guarantee for unmanned driving.

Keywords: environmental hazard prediction; unmanned driving system; target detection; depth learning algorithm; SSD target detection method; method of combining the risk model

Introduction

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 traffic. 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, 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. 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:

1. 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. It is necessary to carry out the information fusion of various sensors and other data sources in order to adapt to the actual driving conditions.

2. Multiple road conditions study is 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.

3. 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.

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. 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 use the data from navigation app and traffic department to realize the joint modeling and statistical analysis.

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. 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. In 2019, It proposed an improved SSD_ARC algorithm for key target detection tasks in driving scenarios. 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.

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. On this basis, this paper proposes an improved SSD method by two steps:

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

2. 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 carrying out 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. 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 thesis’s structure is divided into four aspects:

Section1: 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.

Section2: 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.

Section3: Hardware and software design of environmental hazard prediction and avoidance system.

Section4: 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.
Nomenclature

Subscripts

- $C$ real license plate size
- $d$ euclidean geometric distance
- $P$ frame storage coordinates
- $M_1$ camera internal parameter matrix
- $P_y$ target of $P$ on the $Y$-axis
- $H$ camera head height
- $h$ camera height
- $v_0$ internal parameters provided for calibration
- $f_y$
- $\sigma$ estimated uncertainty
- $L$ the predicted value of the position
- $x_e$ estimated boundary box position and standard deviation
- $L_{reg}$ bounding box regression
- $t_{xi}, t_{yi}$ deviation between the prediction and the anchor boundary frame
- $x_i$ bounding box position
- $y_i$ true box position
- $x_i^*$ true box position
- $y_i^*$
- $C'$ size of the object in the picture
- $fps$ frame rate
- $P'$ frame storage coordinates
- $M_2$ camera position matrix
- $Q$ distance from the camera to the nearest point below
- $V$ pixel height of the image
- $v$ pixel height coordinate of the target in the image
- $c$ category confidence prediction
- $g$ position parameter
- $x_g$ real boundary box position on the ground
- $f$ camera focal length
- $D$ Kullback-Leibler divergence
- $L_{cls}$ classification loss
- $t_{xi}, t_{yi}^*$ deviation between the real and the anchor boundary frame
- $\sigma_t$ adjustable parameter for variable voting

Abbreviations

- $FL$ Focal Loss
- $RNN$ Recurrent Neural Network
- $SSD$ Single Shot MultiBox Detector
- $BDS$ BeiDou Navigation Satellite System
- $CNN$ Convolutional Neural Network
- $LSTM$ Long Short-Term Memory
- $LBS$ Location Based Service
- $KL$ Kullback-Leibler
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 prejudgment’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.

On this basis, 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.

| 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. Driving environment types.**

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 risk is 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.

| 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. Risks based on nearby environment types.

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.

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 query.
Based on the scene

Due to possible errors in location and database, the system confirms and supplements the on-site target detection. First, the 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 the scene. This part is identified through machine vision to add the non-existing results and avoid the omission of risk targets.

Because the risk target is divided into real risk and possible risk, the existing risk target, such as vehicle, pedestrian which can be identified by the completed target detection algorithm, and risk weighting under space-time conditions is evaluated based on historical information.

Figure 1 and Figure 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 vise 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.

| 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  |
### Table 3. Weighted value table based on accident ratio.

|       | 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 |

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, reverse, reverse, 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.

| 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. Risks based on accident types.

According to the current location, weather and time matching accidents, 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**

Deep learning trained models are applied to identify and detect the sequence of captured images. And the algorithm is used to calculate the direction and 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 speed 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, so 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 plate 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:

\[
Scale = \frac{c}{c'}
\]  
(1)

Actual speed size:

\[
Speed = d \times fps \times scale
\]  
(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, vision distortion and other issues should be considered first when CMOS sensors are used. The correction 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:

\[
\begin{bmatrix}
X \\
Y \\
Z \\
1
\end{bmatrix} = \begin{bmatrix}
\frac{s_x}{u_0} & 0 & u_0 \\
0 & \frac{s_y}{v_0} & v_0 \\
0 & 0 & 1
\end{bmatrix} \times \begin{bmatrix}
f & 0 & 0 & 0 \\
0 & f & 0 & 0 \\
0 & 0 & 1 & 0
\end{bmatrix} \begin{bmatrix}
R \\
T \\
0 \\
1
\end{bmatrix} \begin{bmatrix}
\frac{X}{u} \\
\frac{Y}{v} \\
\frac{Z}{w}
\end{bmatrix}
\]  
(3)

where, \((X,Y,Z)\) is the real world coordinate system, \((X_u,Y_u,Z_u)\) 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, the origin of the fixed frame storage coordinate system is \((u_0,v_0)\), set any position as \((u,v)\), \(R\) is the
3x3 rotation matrix, T is the 3x1 translation matrix, and f is the camera focal length. Simplification can be done again:

\[
Z_v \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}
\]

(4)

Finally reduced to:

\[
Z_v \times P = M_1 \cdot M_2 \cdot P'
\]

(5)

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:

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

(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. Using coordinate system conversion:

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

(7)

This time, \( V \) is the pixel height of the image, \( v \) is the pixel height coordinate of the target in the image, and \( v_0 \) and \( f_y \) are internal parameters provided for calibration.

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 firstly 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 depth 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 safety range for estimation. For hidden risks, the risk of ground skid 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 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 $5 \text{m/s}^2$ can be considered dangerous, $2$ to $5 \text{m/s}^2$ is critical danger, and below $2 \text{m/s}^2$ it can be considered safe. However, the road conditions will lead to a decrease in braking performance, which is reflected in the deceleration at the maximum braking effect, referred to as the maximum deceleration. Besides, the braking deceleration for 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.

**Figure 3.** Flow chart of evasion algorithm.

**Algorithm model construction and demonstration based on deep learning**

*Algorithm construction of environmental perception and hazard prediction*

The convolution neural network (CNN) and recurrent 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 not only adapts to complex background, but also can deal with multiple targets. What’s more, a complete description can be obtained by using the end-to-end method on the expression model, proposed by the Northwest University of Technology team in 2018.
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, algorithm based on deep learning has obvious advantages in detection accuracy and efficiency. In this paper, an improved SSD-based target detection algorithm is proposed.

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

The supplementary description of environment perception belongs to the category of image semantic recognition, and the method adopted belongs to the ‘end-to-end’.

The work of feature extraction is completed by CNN classification model. After classification, it is represented by LSTM, 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, etc. The feature of this method lies in dividing attention by color and weighting attention regions appropriately. However, the so-called color attention weighting 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 Figure 4 regularly makes a trade-off of the content, and the trade-off is controlled by four gates. A feature-based weight unit is generated in 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 original SSD paper, the following structure is presented. SSD uses the feature pyramid structure for detection, which uses the characteristic smaps of conv4-3, 6-2, 7, 7-2, 8-2, 9-2. At the same time, position regression and softmax classification are performed. Figure 5 demonstrates that the SSDs can use VGG-16 as the base network. The feature extraction layer in the latter part is also predicted. In addition, the detection is performed not only on additional feature maps, but also on the underlying conv4-3 and 7-feature ditutatations to achieve compatibility with small goals.

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

(a) There are two kinds of feature maps with multi-scale feature map: large feature map corresponds to small target and large target is responsible for small feature map.

(b) The feature map is extracted directly by convolution so that a large feature graph can be obtained with a relatively small convolution kernel.

(c) Setting a priori box each cell generates a priori box with different size, length and width. The
priori box serves as the baseline of the bounding box, and multiple priori frames are generated in different ways during the training process.

Using 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 $2 \times 2$ to $3 \times 3$.

Then the FC8 and dropout layers are replaced by a series of convolution layers, and fine-tuned them using the detection set. The Conv4 layer with a size of $38 \times 38$ in VGG16 will serve as the first feature map for detection. But 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 the original layer of CONV4 was added, forming a total of six. 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 priors, including size, length and width. What’s more, as the size of the feature map increases, the prior box size decreases:

$$w^a_k = s_k \sqrt{a_r}, h^a_k = s_k / \sqrt{a_r}$$

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

Algorithm training and improvement

Training

Prior box matching

Before the work, the prior frame with the target or part of the target is retrieved, and the matching boundary frame will enter the prediction phase. The first step of prior frame matching is to confirm the largest prior frame for 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:
where $N$ represents the number of positive samples, $x_{ij}^p \in \{1, 0\}$ is used as an indication parameter, and when $x_{ij}^p = 1$ is used, the $I$-th prior box matches the $j$-th target with category $p$. $c$ represents the category confidence prediction. And $L$ is the predicted value of the position, which is the position of the boundary of the target selected by 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 L1 loss as follows:

$$L(x, l, g) = \sum_{i \in \text{pos}} \sum_{m \in \{c, x, y, w, h\}} x_{ij}^k \text{smooth}_{L1} \left( \frac{l_i^m - \hat{g}_j^m}{\sigma_i} \right)$$

(10)

$$\hat{g}_j^x = \frac{\left( g_j^x - d_i^x \right)}{\sigma_i}$$

(11)

$$\hat{g}_j^y = \frac{\left( g_j^y - d_i^y \right)}{\sigma_i}$$

(12)

$$\hat{g}_j^w = \log \frac{g_j^w}{\sigma_i}$$

(13)

$$\hat{g}_j^h = \log \frac{g_j^h}{\sigma_i}$$

(14)

$$\text{smooth}_{L1}(x) = \begin{cases} 0.5x^2 & \text{if } |x| < 1 \\ |x| - 0.5 & \text{otherwise} \end{cases}$$

(15)

The parameters are as follows:
\[
\tilde{g}_j^{cx} = \frac{(s_i^{cx} - \hat{s}_i^{cx})}{\text{variance}} \\
\tilde{g}_j^{cy} = \frac{(s_i^{cy} - \hat{s}_i^{cy})}{\text{variance}} \\
\tilde{g}_j^w = \frac{\log \left( \frac{\hat{s}_i^w}{s_i^w} \right)}{\text{variance}} \\
\tilde{g}_j^h = \frac{\log \left( \frac{\hat{s}_i^h}{s_i^h} \right)}{\text{variance}}
\]

For confidence error, it adopts softmax loss:

\[
L_{conf}(x, c) = - \sum_{i \in Pos} x_i^p \log (\tilde{p}_i^c) - \sum_{i \in Neg} \frac{\exp (\hat{c}_i^p)}{\sum \exp (\hat{c}_i^p)} \exp \log
\]

\[
FL(p_t) = - \alpha t (1 - p_t)^Y \log (p_t)
\]
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, the loss function value becomes smaller because the power of $(1-0.95)$ is 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 small prediction probability are added together. Meanwhile more punishments are required for negative samples that are easily divided. The actual formula is as follows:

$$L_{fl} = \begin{cases} -\alpha (1 - y)^{\gamma} \log y^\gamma & y = \text{positive simple} \\ - (1 - \alpha) y^{\gamma \gamma} \log (1 - y^\gamma) & y = \text{negative simple} \end{cases} \quad (22)$$

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 accurate, but it is not always the case. Bounding box prediction is modeled as a Gaussian distribution, and the bounding 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 to the Gaussian distribution, close to the positive sample.
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:

$$D(P\|Q) = \sum_{i \in \mathcal{X}} P(i) \ast \left[ \log \left( \frac{P(i)}{Q(i)} \right) \right]$$  \hspace{1cm} (23)

$$D(P\|Q) = \int_{x} P(x) \ast \left[ \log \left( \frac{P(x)}{Q(x)} \right) \right] \, dx$$  \hspace{1cm} (24)

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 anchor bounding box generated by clustering all real boxes. Then the deviations of the predicted and real bounding boxes from are as follows:

$$t_{x1} = \frac{x_1 - x_{1a}}{w_a}, t_{x2} = \frac{x_2 - x_{2a}}{w_a}$$  \hspace{1cm} (25)

$$t_{y1} = \frac{y_1 - y_{1a}}{h_a}, t_{y2} = \frac{y_2 - y_{2a}}{h_a}$$  \hspace{1cm} (26)

$$t^*_{x1} = \frac{x'_1 - x_{1a}}{w_a}, t^*_{x2} = \frac{x'_2 - x_{2a}}{w_a}$$  \hspace{1cm} (27)

$$t^*_{y1} = \frac{y'_1 - y_{1a}}{h_a}, t^*_{y2} = \frac{y'_2 - y_{2a}}{h_a}$$  \hspace{1cm} (28)

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

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

$$p_\theta(x) = \frac{1}{\sqrt{2\pi}\sigma^2} e^{-\frac{(x-x_\theta)^2}{2\sigma^2}}$$  \hspace{1cm} (29)

Where $x_\theta$ is the estimated boundary box position and standard deviation $\sigma$ is the estimated uncertainty. When $\sigma \to 0$, the boundary box position accuracy is very high.

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

$$P_D(x) = \delta(x - x_g)$$  \hspace{1cm} (30)

Where $x_g$ is the real boundary box position on the ground.

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

$$L_{\text{reg}} = D_{KL} \left( P_d(x) \parallel P_\theta(x) \right)$$

$$= \int P_d(x) \log P_d(x) dx - \int P_\theta(x) \log P_\theta(x) dx$$

$$= \left( x_2 - x_1 \right)^2 \frac{1}{2\sigma^2} + \log \left( \frac{\sigma^2}{2 \pi} \right) - \frac{1}{2} \log(2\pi) - H \left( P_d(x) \right)$$

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 $L_{\text{reg}}$. After the variance of the predicted positions of the bounding boxes 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:

$$p_i = e^{-\frac{(1-\text{IoU}(b_i,b))^2}{\sigma_t}}$$

$$x = \frac{\sum p_i x_i / \sigma^2}{\sum p_i / \sigma^2}$$

subject to $\text{IoU}(b_i,b) > 0$  \hspace{1cm} (33)

Where $\sigma_t$ is an adjustable parameter for variable voting. When $\text{IoU}(b_i,b)$ is larger, $p_i$ is larger, that is, 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 border regression. Besides, the border regression of SSD is improved based on KL loss method. Frames with large variance and adjacent border frames containing the selected frames but too small will get low scores when voting. Moreover, the SSD algorithm can effectively avoid the abnormal situation mentioned above by variance voting instead of $\text{IoU}$ overlap degree.

Model testing and analysis

The environment perception is divided into two parts, the micro part is main perception of the scene by machine vision, which is used to confirm and supplement the macrocosm and microcosm 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 test of region of interest 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
Polytechnical University team. The experimental results on the Flickr8K database are shown in
Table 5, on the Flickr30K database are shown in Table 6, and on the MS COCO database are
shown in Table 7.

| 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.

| Ways            | BLEU-1 | BLEU-2 | BLEU-3 | BLEU-4 |
|-----------------|--------|--------|--------|--------|
| Google NIC      | 66.3   | 42.3   | 27.7   | 18.3   |
| Log Bilinear    | 60     | 38     | 25.4   | 17.1   |
| The design algorithm | 67.1 | 45.1   | 29.9   | 21.1   |

Table 6. The results of the experiments on the database Flickr30K.

| 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.

The focus will be on target detection in the hazard prediction section. First of all, the vehicles test
is carried out, using field test maps and dataset pictures. Secondly, dynamic vehicles need to be
detected, including their speed, distance and running direction. The vehicle target detection is
shown in Figure 7 and Figure 8. The dynamic vehicle direction estimation is shown in Figure 9
and Figure 10. The dynamic vehicle distance estimation is shown in Figure 11. The vehicle speed
detector is used to detect the speed of dynamic vehicles in Figure 12.
Figure 7. Vehicle target detection.  
Figure 8. Vehicle target detection.

Figure 9. Dynamic vehicle direction estimation.  
Figure 10. Dynamic vehicle direction estimation.

Figure 11. Dynamic vehicle distance estimation.  
Figure 12. Vehicle speed detector.

Results and discussion

Simulated route

In order to better reflect function of the system, the paper uses Matlab and Carsim to set up the dangerous situation of vehicle crossing in different conditions and conduct a joint simulation. The simulation system will output the speed constraint throughout the whole simulated driving process. Figure 13(a) simulates the vehicle operation by adjusting the scene, road surface and definition, driving conditions, etc. Figure 13(b) 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 kilometers, 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 Qingnian road, and the results are shown in Table 8 and Table 9 respectively. The system test path map is shown in Figure 14.

**Figure 13.** Simulation interface of CarSim.

**Figure 14.** Simulated route.

| 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                 |                                                 |
Table 8. Latitude and longitude sampling table of Tianshan road.

| 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. Longitude and latitude sampling table of Qingnian road.

Latitude and longitude sampling table of Tianshan road is shown in Table 10. Longitude and latitude sampling table of Qingnian 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 60km/h, simulation speed constraints are provided for 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.

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.

| 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. Latitude and longitude sampling table of Tianshan road.
| 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.** Longitude and latitude sampling table of Qingnian road.

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.

![Figure 15. Speed constraints at 8:00 on Monday and Sunday.](image)

The comparison of speed constraints between Monday and Sunday is shown in Figure 15. On the whole, the speed constraints on Monday are stricter than those on Sunday, which is caused by risk weighting based on experience. 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 required acceleration 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 Figure 16. 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 of vehicles equipped with the system and human-driven vehicles is shown in Figure 17. 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 100km/h strays into the lane while avoiding the normally running vehicle and eventually rolls over. The normally running vehicle with a speed of 100km/h completes emergency braking during the avoidance. The accident site is a freeway.

In the simulation, the speed of 70km/h is set as the normal driving speed, which is consistent with the actual use of expressway. Figure 18(a) demonstrates that the unsystematic vehicles are running normally at a speed of 70km/h. Within a period of visible sight distance, no vehicles were observed to enter the lane in opposite direction. After the danger was found, the collision was avoided by emergency braking. Figure 18(a) shows the speed of each tire of both vehicles. It indicates that the speed of the vehicles decreased sharply and fluctuated greatly during emergency...
braking. The whole braking process has a great influence on passengers and has great uncertainty factors.

(a) Simulation curve of vehicle avoidance without the system

(b) Simulation curve of vehicle avoidance with the system

Figure 18. CarSim Accident Simulation Diagram.

Figure 18(b) shows that the vehicles equipped with the system decelerated before the intersection due to speed constraints, and drove through the intersection at a speed approaching 40km/h. After the other vehicle entered the sight distance, lane change and braking operations were adopted to avoid collision safely and stably. It can be seen that the braking curve of the vehicle with a system is smoother than that of the vehicle without a system. The danger has been successfully avoided with preparation in advance. Therefore, the simulation application proves the effectiveness of the system to some extent.

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. This technology is of great significance to the improvement of the safety and the commercialization of unmanned vehicles. 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. Therefore, the design of this system has an advanced vision and provides a stronger guarantee for the safety of unmanned driving. In this paper, Matlab and CarSim are used to simulate the entire system. The speed constraint and simulation speed diagrams under various condition are output on the selected driving path to verify the effectiveness of the system function. And an accident is simulated, which reflects the value of the system to a certain extent.

The system is innovative in the target detection of core technologies. In the overall design, guided by deep learning and by the industry frontiers such as recognizing 5G and big data, it aims to solve the problems of the unmanned industry based on the existing technological achievements and realize the application transformation providing a reference to the improvement of the unmanned industry.

**References**

1. X.Y. Z. et al. A study on key technologies of unmanned driving. *CAAI Transactions on Intelligence Technology*.1, 4-13 (2016).
2. B. K. 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).
3. Y.J. Li. 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).
4. R. Z. et al. Electric vehicles’ energy consumption estimation with real driving condition data *Transport Res D-Tr E*. **41**, 177-187 (2015).
5. Y, 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).
6. J.J. L. et al. A novel multi-segment feature fusion based fault classification approach for rotating machinery. *Mech Syst Signal Pr*. **122**, 19-41 (2019).
7. F. W. 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).
8. Khalid bin Hasnana. et al. JOMS: System Architecture for Telemetry and Visualization on Unmanned Vehicle. *Procedia Engineering*. **29**, 3899-3903 (2012).
9. Z.Z. Li. et al. Influence of distance from traffic sounds on physiological indicators and subjective evaluation. Transport Res D-Tr E. 87, 102538 (2020).

10. Benjamin Ryder. et al. Spatial prediction of traffic accidents with critical driving events - Insights from a nationwide field study. Transport Res A-Pol. 124, 611-626(2019).

11. Hiroshi Yoshitake, MotokiShino. Risk assessment based on driving behavior for preventing collisions with pedestrians when making across-traffic turns at intersections. IATSS Research. 42, 240-247(2018).

12. Rawia Ahmed Hassan E.L. Rashidya. et al. A big data modeling approach with graph databases for SPAD risk. Safety Sci.110, 75-79 (2018).

13. Brian S. Freeman. et al. Vehicle stacking estimation at signalized intersections with unmanned aerial systems. IJTST. 8, 231-249 (2019).

14. Y.W. C. et al. An effective infrared small target detection method based on the human visual attention. Infrared Phys Techn. 95, 128-135 (2018).

15. Debojit Biswas. et al. An automatic traffic density estimation using Single Shot Detection (SSD) and MobileNet-SSD. Physics and Chemistry of the Earth, Parts A/B/C. 110, 509-520 (2019).

16. A.G. et al. Deep learning for object detection and scene perception in self-driving cars: Survey, challenges, and open issues. Array. 10, 100057 (2021).

17. Farid Feyzi. et al. FPA-FL: Incorporating static fault-proneness analysis into statistical fault localization. J Syst Software. 136, 39-58 (2018).

18. Jan Březina. et al. Fast algorithms for intersection of non-matching grids using Plücker coordinates. Comput Mathe Appl.74, 174-187 (2017).

19. Rondell Burge. et al. An investigation of the effect of texting on hazard perception using fuzzy signal detection theory (fSDT). Transport Res F-Traf. 58, 123-132 (2018).

20. Ellen F. Grumert. et al. Using connected vehicles in a variable speed limit system. Transport Res Procedia.27, 85-92 (2017).

21. 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).

22. 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).

23. 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).

24. 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).

25. Y.T. B. et al. Crash prediction with behavioral and physiological features for advanced vehicle collision avoidance system. Transport Res C-Emer. 74, 22-23(2017).

Acknowledgements

This work is supported by the National High Tech Project “863” [Project No. 2011AA11A286] and Natural Science Foundation of China [Project No. 51405201].

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.