ISSN 1330-3651 (Print), ISSN 1848-6339 (Online) https://doi.org/10.17559/TV-20240720001866
Original scientific paper

DST-Based 3D Obstacle Detection and Classification for Autonomous Driving

Zhi Xiong JIN*, Ran WEN, Ji Lan HUANG

Abstract: Accurate 3D obstacle detection and classification are crucial for autonomous driving systems. This study proposes a novel algorithm combining Dempster-Shafer Theory (DST) with deep learning techniques to address limitations in traditional computer vision methods for 3D scenes. This method combines the extended-corrosion algorithm based on depth information with multi-feature vector classification using back propagation neural networks. Experimental results on real-world datasets demonstrated an average detection accuracy of 95.81% and an effective obstacle recognition accuracy of 93.26%. The algorithm's average processing time of 191.10 ms per frame met real-time requirements for autonomous driving applications. This approach offers improved accuracy and robustness in complex 3D environments, advancing the field of obstacle detection for intelligent transportation systems.

Keywords: autonomous driving technology; classification and identification; DST; 3D obstacles; target detection

1 INTRODUCTION

In the automatic driving system, obstacle detection is one of the key technologies to ensure the safety of the vehicle. However, due to the complexity of three-dimensional (3D) scenes, traditional 2D image processing and computer vision methods have limitations in processing point cloud data and obstacle pose changes, etc. These challenges limit the effectiveness of these methods in 3D space [1, 2]. Although deep learning methods have made remarkable progress in 3D obstacle object detection and classification recognition in recent years, the existing literature mainly focuses on improving the accuracy and efficiency of the algorithm, while the discussion on multi-source data fusion, real-time and robustness is relatively small. Most methods rely on specific data types or environmental conditions and lack generality. Secondly, algorithms often face computational efficiency challenges when processing large-scale datasets [3-4]. The limitations of the existing researches in 3D obstacle detection and classification are as follows: over-reliance on specific sensor data and lack of comprehensive processing ability of multi-source information; When dealing with large-scale datasets, the computational efficiency is low, which is difficult to meet the real-time requirements. In addition, the detection accuracy and robustness are insufficient when facing complex environments or partially occluded targets [5-7]. Aiming at the shortcomings of existing methods, a 3D obstacle object detection and classification algorithm based on Dempster-Shafer evidence theory (DST) is proposed. The goal of the research is to develop an algorithm that can overcome the limitations of existing technologies to provide an efficient, accurate and robust 3D obstacle detection and classification method to meet the needs of autonomous driving and intelligent transportation systems. The main contribution of the research is to propose a DST-based 3D obstacle detection and classification algorithm, which can effectively integrate data from different sensors and improve the accuracy of detection and classification. A 3D object detection algorithm based on depth information is designed, and the performance of object detection is improved by depth enhancement feature fusion and view aggregation. By combining the improved extended corrosion algorithm with DST, the processing of 3D point cloud data has been optimized, improving the algorithm's ability to identify and classify different obstacles. It is expected to provide a more reliable obstacle detection solution for autonomous vehicles to cope with the complex and changeable traffic environment.

2 RELATED WORKS

With the development of deep learning technology, deep learning methods have made great progress in 3D obstacle detection and classification recognition. The core of the recognition system lies in accurately identifying objects, which usually depends on the appropriate extraction of the object's features to match the specific recognition technique. Rani et al. proposed a feature extraction technique based on deep learning for specific recognition needs. Studies confirmed the effectiveness and practicability of this method [8]. In the complex time-varying surface environment, the recognition rate and computational efficiency of unmanned surface ships are low. Therefore, Peng et al. proposed a network structure based on the Faster R-CNN algorithm to analyze the dataset. The research results show that the method could effectively identify and classify six common targets on the water surface [9]. In ocean missions involving 3D path tracking, the overall goal of the underwater vehicle is to successfully complete the path pre-specified by the operator. Therefore, Rubio et al. proposed an online interval two fuzzy limit learning machine. The research results showed that this method could effectively deal with uncertainty and generate reasonable classification prediction, and the accuracy of test data was close to 90.5% [10]. The orchard environment is complex and full of obstacles. Traditional methods have been unable to effectively enter the work. Therefore, Qian et al. proposed an autonomous obstacle avoidance method for orchard robots based on obstacle classification and recognition. The research results showed that the robot could avoid obstacles approaching the obstacles [11]. Shi et al. proposed a basketball robot recognition system based on multi-sensor in unknown environments. The research results showed that this method could complete the target recognition and location task under unknown environments [12]. Zhou et al.

proposed a hybrid obstacle avoidance approach that addressed the serious challenge of wheeled mobile robots in unstructured environments. This method combined informed fast exploration random tree algorithm with 3D object detection method and model prediction controller [13].

The DST has a wide range of applications, which is particularly suitable for handling incomplete, uncertain, and conflicting information. In response to the uncertainty of storm surges during disaster processes, Sun et al. proposed a disaster loss assessment method based on the DST. The classification accuracy reached 93.1%, which was superior to some existing methods [14]. Yuan et al. proposed a generalized combination rule that combined weight and reliability to address the reliability dependency, non-reliability effectiveness, and inter-generational inconsistency in evidence theory. The research results indicated that the superiority of this rule was demonstrated through numerical comparison and discussion [15]. One of the important reasons for geological risks in the oil and gas exploration is related to the uncertainty of geospatial data and data fusion models. Therefore, Seraj et al. enhanced the geological risk analysis of the Dempster-Shafer data-driven method through fuzzy logic methods. The success rate and area under the prediction curve of this method were 82.2% and 75.9%, respectively [16]. Digital investigators have used databases to investigate cyber-crime. Adane et al. proposed a rule-based outlier detection algorithm for processing financial transaction records. The DST combined various suspicious transaction evidence obtained from audit logs to analyze data. The research results indicated that this method could evaluate the risk of suspicious transactions [17]. Obtaining weights that reflected their importance or decision-maker perceived reliability in group decision-making was a new challenge. Koksalmis et al. proposed a clustering-based method. The research results indicated that when applied to two datasets, the classification rate was significantly improved [18]. The literature review is summarized in Tab. 1.

Table 1 Summary table of literature review

Document number	Method description	Data source	Detection/ classification accuracy	Advantages	Limitations	Application scenario
[8]	Feature extraction technology based on deep learning	Multi-source data	87%	Feature extraction is highly targeted	Depending on a specific data set	Obstacle detection in agricultural environment
[9]	Surface target recognition based on Faster R-CNN	Image data	88%	High recognition rate	Low computational efficiency	Surface unmanned ship
[10]	Online interval 2 fuzzy limit learning machine	Point cloud data	90.5%	Dealing with uncertainty	context-dependent	Underwater vehicle path tracking
[11]	Orchard robot autonomous obstacle avoidance method	Point cloud data	85%	Effective obstacle avoidance	Limited environmental adaptability	Orchard environment
[12]	Basketball robot recognition system based on multi-sensor	Multisensor data	92%	Multi-environment adaptation	Complex computation	Basketball robot sport
[13]	Hybrid obstacle avoidance method	Image data	80%	Meet the needs of unstructured environments	Model-dependent prediction	Wheeled mobile robot
[14]	Disaster loss assessment method based on DST	/	93.1%	Better than existing methods	Limited range of application	Storm surge hazard assessment
[15]	Generalized combination rule	/	/	Solve the parameter infeasible problem	Limited range of application	Evidence theory
[16]	Geological risk analysis based on DST	Geospatial data	82.2% (predicted success rate), 75.9% (AUC)	High prediction success	Strong data dependence	Oil and gas exploration
[17]	Rule-based outlier detection algorithm	Financial transaction record	91%	Assess risk level	Limited range of application	Cyber crime investigation
[18]	Clustering based method	/	/	Classification rate improvement	Limited range of application	Group decision making
Research and propose methods	3D obstacle detection and classification algorithm based on DST	Multi-source data	95.81% (detection), 93.26% (classification)	Handle uncertainty and conflicting information with high accuracy	More computing resources are required	Autonomous vehicles, intelligent transportation systems

In summary, the existing methods have made some progress in the field of 3D obstacle detection and classification recognition, but they still show some limitations in practical application. For example, although feature extraction techniques based on deep learning perform well in specific applications, they often rely on

large amounts of labeled data and have limited adaptability to environmental changes. The Faster R-CNN algorithm has achieved good results in surface target recognition, but the computational efficiency is low, which is not suitable for real-time or resource-constrained application scenarios. The online interval two fuzzy extreme learning machine performs well in dealing with uncertainty, but needs to be adjusted for specific environments. To solve these problems, a 3D obstacle object detection and classification algorithm based on DST is proposed. This algorithm aims to improve the accuracy and robustness of detection by fusing multiple sources of data, while ensuring the real-time performance of the algorithm.

3 RESEARCH METHODS

Firstly, the DST is introduced, but it has some limitations. Therefore, the 3D obstacle target detection and classification recognition algorithm based on DST is improved. Subsequently, a 3D object detection algorithm based on depth information improvement is designed. Finally, a 3D obstacle classification and recognition algorithm based on multiple feature vectors is proposed to improve it.

3.1 Three-Dimensional Obstacle Object Detection Based on Dempster-Shafer Evidence Theory

DST is a mathematical framework designed to handle uncertainty in data analysis and decision-making processes. In DST, different types of evidence are referred to as "credibility". Each credibility represents the support level for a proposition. Credibility can be a numerical value between 0 and 1, indicating the likelihood of a proposition being true. Different credibility levels are merged to obtain a new credibility level that represents more comprehensive evidence [19-21]. The recognition framework refers to the set of things or objects being examined and judged. $\Theta=\{a, b, c, ...\}$ is a non-empty set that contains n pairwise mutually exclusive events. The power function of the recognition framework contains 2^n elements, and $P(\Theta)=\{\varnothing, a, b, c, a, b, a, c, ..., \Theta\}$. The Basic Probability Assignment (BPA) in the hypothetical space is a function m with 2^{Θ} in the [0, 1]. That is the mass function, which satisfies the condition, as shown in Eq. (1).

$$\begin{cases} m(\varnothing)=0 \\ \sum_{A\subseteq\Theta} m(A)=0 \end{cases} \tag{1}$$

Eq. (1) is the definition of the basic probability assignment in DST. m is the basic probability assignment function, which is a core concept in DST for quantifying the degree of trust in a hypothesis or evidence. In DST, $m(\phi)=0$ means assigning a probability value of 0 to cases where none of the hypotheses is trusted, that is, not thinking that all hypotheses hold is a possible option. Θ stands for recognition frame, which is the set of all possible hypotheses, that is, the total set of all possible outcomes when considering the problem. A is a subset of the recognition framework Θ, representing a combination of one or more hypotheses, which can be a single hypothesis or a union of multiple hypotheses. In DST, the sum of the basic probability assignments for all hypotheses is 1, and the probability of the assignment for the empty set is 0, which is consistent with the principle in probability theory that the total probability is 1 and the probability of impossible events is 0. In this way, DST is able to deal with uncertainty and conflicting information and perform effective information fusion among multiple information sources. Trust functions are also known as reliability functions. The trust function based on BPA m in hypothesis space Θ is shown in Eq. (2).

$$Bel(A)=\sum_{B\subseteq A} m(B) \tag{2}$$

The Likelihood Function (LF) based on BPA m in hypothesis space Θ is shown in Eq. (3).

$$Pl(A)=\sum_{B\cap A\neq\varnothing} m(B) \tag{3}$$

In evidence theory, for a hypothesis A in hypothesis space Θ, the trust function and LF are calculated based on BPA to form the trust interval $[Bel(A), Pl(A)]$, which represents the certainty degree for a hypothesis. The trust interval is shown in Fig. 1.

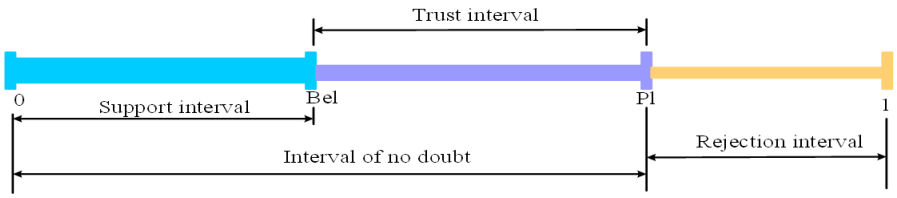


Figure 1 Trust interval

In Fig. 1, a two-dimensional coordinate system is used to show the trust interval, where the horizontal axis represents all possible hypotheses in the hypothesis space and the vertical axis represents the degree of trust. *Bel* is represented as a point or region that directly reflects the degree of direct trust in the hypothesis. *Pl* is represented as another point or region, reflecting the maximum degree to which the hypothesis is likely to be true. The trust interval is defined by *Bel* and *Pl*, forming a range that shows uncertainty from the lowest level of trust to the highest possible level of trust. This representation method helps to intuitively understand the degree of trust in a certain

hypothesis and its uncertainty range under given evidence. For application scenarios, such as decision support systems of autonomous driving systems, this analysis method can help the system better evaluate the credibility of different hypotheses, so as to make more accurate decisions. Dempster synthesis rule is known as evidence synthesis formula. For $\forall A \subseteq \Theta$, the Dempster and composition rules of mass functions m_1 and m_2 on Θ are shown in Eq. (4).

$$m_1 \oplus m_2(A) = \frac{1}{K} \sum_{B \cap C = A} m_1(B) \cdot m_2(C) \tag{4}$$

In Eq. (4), K represents the normalization constant. Eq. (5) displays the specific calculation.

$$K = \sum_{B \cap C \neq \varnothing} m_1(B) \cdot m_2(C) = 1 - \sum_{B \cap C \neq \varnothing} m_1(B) \cdot m_2(C) \tag{5}$$

For $\forall A \subseteq \Theta$, the Dempster composition rule for a finite mass function $m_1, m_2, ..., m_n$ on Θ is shown in Eq. (6).

$$\begin{aligned} &(m_1 \oplus m_2 \oplus, ..., \oplus m_n)(A) = \\ &= \frac{1}{K} \sum_{A_1 \cap A_2 \cap ... \cap A_n = A} m_1(A_1) \cdot m_2(A_2), ..., m_n(A_n) \end{aligned} \tag{6}$$

In Eq. (6), the specific calculation of the normalization constant K is shown in Eq. (7).

$$\begin{aligned} K &= \sum_{A_1 \cap ... \cap A_n \neq \varnothing} m_1(A_1) \cdot m_2(A_2), ..., m_n(A_n) = \\ &= 1 - \sum_{A_1 \cap ... \cap A_n \neq \varnothing} m_1(A_1) \cdot m_2(A_2), ..., m_n(A_n) \end{aligned} \tag{7}$$

Eq. (7) describes the Dempster composition rule in DST, which is used to merge evidence from different information sources. K is the normalization constant in Dempster composition rules, which is used to ensure that the combined evidence still satisfies the fundamental axiom of probability, that is, the sum of all the basic probability assignments is 1. It is calculated by subtracting the sum of the products of all possible disjoint sets of evidence. $A_1, A_2, ..., A_n$ represents a set of hypotheses from different information sources. $m_1, m_2, ..., m_n$ is a basic probability distribution function corresponding to different information sources. $A_1 \cap ... \cap A_n \neq \varnothing$ means that the intersection of all sets. $A_1, A_2, ..., A_n$ is non-empty, that is, there is at least one common element among these sets. DST also has some problems and limitations, such as high computational complexity for large-scale problems. Therefore, in practical applications, appropriate improvements are made based on specific circumstances. The improved 3D object detection algorithm based on depth information utilizes depth images to extract information about the shape, position, and motion of objects. Deep enhancement feature fusion and perspective aggregation are applied to improve the performance. The flowchart of the 3D point cloud data collection function is shown in Fig. 2.

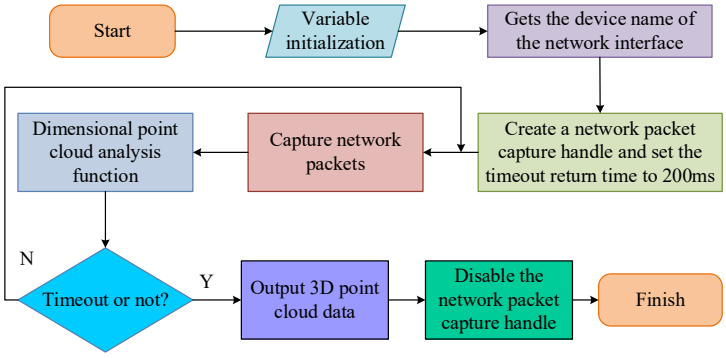


Figure 2 Flow chart of 3D point cloud data acquisition function

The principle of Structural Elements (*SEs*) is as follows. Assuming there are two images *A* and *B*, *A* is the processed images, *B* is used to process *A*, and *B* is the *SE*. *SEs* can be of any shape, usually smaller images. It has an anchor point O that can freely define the anchor position. The mapped and translated *SE* must overlap with some parts of the area in the image *P* where the pixel grayscale value is 1, as shown in Eq. (8).

$$P \oplus SE = \{z \mid (SE)_z \cap P \neq \varnothing\} \tag{8}$$

Corrosion is the sliding of *SE* in image *P*. When the *SEs* are completely included in image *P*, the pixel grayscale of the anchor position in *P* is set to 1. The grayscale value of the remaining pixels is 0, as shown in Eq. (9).

$$P \Theta SE = \{z \mid (SE)_z \cap P^c \neq \varnothing\} \tag{9}$$

An image is first subjected to corrosion calculations using the same *SE*. Then, an expansion operation is performed, which is called an open operation, as shown in Eq. (10).

$$P \circ SE = P\Theta SE \oplus SE \tag{10}$$

After the open operation, some isolated small points and burrs in the graph can be eliminated. However, the overall shape, structure, and position remain unchanged. Closed operation is the process of using the same *SE* to expand an image before performing a corrosion operation, as shown in Eq. (11).

$$P \cdot SE = P \oplus SE\Theta \mathrm{SE} \tag{11}$$

Closed operation can eliminate small holes and sew small cracks, while the overall shape, structure, and position are unchanged. The traditional expansion corrosion algorithm uses a single *SE* to perform expansion corrosion processing on the entire image in the same order, which can effectively remove isolated points and fill small loopholes. However, this method has limitations in processing voxel maps based on 3D point cloud data [22, 23]. Based on the above situation and the characteristic that the distribution density of point cloud data changes with distance, an improved dilation corrosion algorithm based on deep information is proposed. The pseudo-code of the improved expansion corrosion algorithm based on depth information is shown in Fig. 3.

In Fig. 3, based on depth information, the improved expansion corrosion algorithm first selects SEs that match the characteristics of the point cloud data. Through the iterative process, the algorithm firstly expands the point cloud to enlarge the object outline, and then performs corrosion operation to eliminate noise and small objects. In each operation, SEs cover each point in the point cloud, determining whether the point is retained or removed based on whether it is fully included in the original data. After the set number of iterations, the algorithm outputs the processed point cloud data, which makes the obstacle identification clearer and more accurate. The flow chart of 3D obstacle target detection is shown in Fig. 4.

Input: 3D Point Cloud Data P, Structural Element SE, Dilation Iterations E, Erosion Iterations C
Output: Processed 3D Point Cloud Data P'
1. Initialize P' = P
2. Select the Structural Element SE based on the statistical properties of the point cloud data (e.g., average spacing)
3. For each point p in Point Cloud P:
a. Perform Dilation Operation:
i. Copy P to a temporary point cloud data P_temp
ii. For each point q in P_temp:
- Place the center of SE at q
- If SE is fully contained within P, set the intensity value of q to the maximum
- Otherwise, set the intensity value of q to 0
iii. Update P = P_temp
b. Repeat the above steps for E iterations
c. Perform Erosion Operation:
i. Copy P to a temporary point cloud data P_temp
ii. For each point q in P_temp:
- Place the center of SE at q
- Only if SE is fully contained within P, set the intensity value of q to the maximum
- Otherwise, set the intensity value of q to 0
iii. Update P = P_temp
d. Repeat the above steps for C iterations
4. Return the processed 3D point cloud data P'

Figure 3 Pseudo-code diagram of an improved expansion corrosion algorithm based on depth information

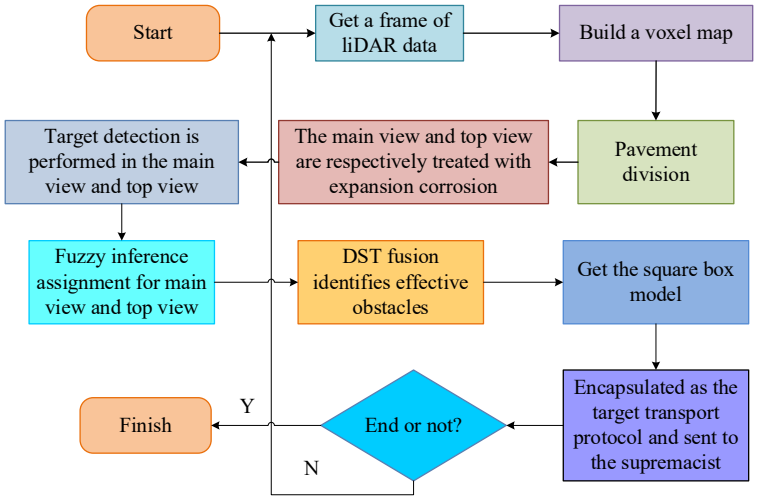


Figure 4 Flow chart of three-dimensional obstacle detection

Firstly, obtain the point cloud data transmitted by the LiDAR, then crop the 3D point cloud data based on the region of interest and establish a voxel map. Then, perform road segmentation on the voxel map to obtain the preprocessed data. The improved expansion corrosion algorithm based on depth information is used for clustering top views. Then, the DST algorithm is combined with basic probability assignment to fuse the information of the main view and the top view, identifying effective obstacles. Finally, a square box model of the effective obstacles is obtained to achieve 3D obstacle target detection.

3.2 3D Obstacle Classification and Recognition Algorithm Based on Multiple Feature Vectors

In obstacle category recognition, due to the uneven distribution of point clouds and other characteristics, there is no good discrimination ability. Different types of obstacles pose varying degrees of threat to autonomous vehicles. Effectively identifying the types of obstacles can

help autonomous vehicle make better path planning and behavior decisions. The stable feature vectors of obstacles in the detection area are extracted as inputs to the Back Propagation Neural Network (BPNN) for training and recognition. There are significant differences in the area size characteristics of different types of obstacle targets in the top view of voxel maps. The area of the voxel map top view is the product of the length and width of the square box model, as shown in Eq. (12).

$$Area = Box_{length} \times Box_{width} \tag{12}$$

The volume of the square box model increases the height information of obstacle targets on the top view area of the voxel map, describing the volume characteristics of different types of obstacle targets in 3D space. The square box model is displayed in Fig. 5.

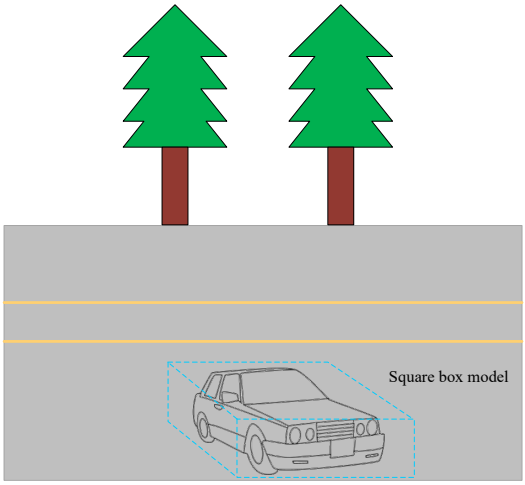


Figure 5 Square box model diagram

In Fig. 5, the square box model is a geometric representation used to describe the volume characteristics of obstacles in 3D space. This model is able to quantify the volume of the obstacle by approximating it as a square box. The blue envelope represents the outer contour of the square box model, which clearly defines the spatial range occupied by obstacles in 3D space. The bottom area of the square box model is obtained by measuring the horizontal projection of the obstacle on the voxel map. The height is obtained by measuring the vertical distance of the obstacle from the ground to the highest point. The blue envelope represents the square box model. The volume $Volume$ of the square box model is the product of the length, width, and height, as shown in Eq. (13).

$$Volume = Box_{length} \times Box_{width} \times Box_{height} \tag{13}$$

The 3D voxel density is the ratio of the effective voxels in the square box model to all voxels in the region. Effective voxels refer to the sum of voxels containing point clouds. All voxels refer to the voxels included in the square box model. The schematic diagram of 3D voxel density is shown in Fig. 6.

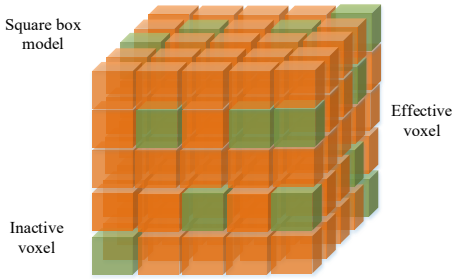


Figure 6 3D voxel density diagram

In Fig. 6, the whole cube represents the square box model of the obstacle, which defines the boundary of the obstacle in 3D space. The interior of the square box model consists of many small cubes, which are called voxels. Voxels are the basic units that make up a three-dimensional model, similar to pixels in a two-dimensional image. Orange voxels represent voxels that contain point cloud data, meaning that these voxels contain actual point cloud information inside. Therefore, they are considered valid. The number of effective voxels is one of the key factors in calculating the 3D voxel density because it is directly related to the actual volume of the obstacle. In contrast, green voxels represent voxels that do not contain point cloud data, meaning that there is no point cloud information inside these voxels. Therefore, they are considered invalid. The number of effective voxels in the square box model is $Voxel_{efficient}$. The number of all voxels is $Voxel_{total}$. The 3D voxel density $Density$ is shown in Eq. (14).

$$Density = Voxel_{efficient} / Voxel_{total} \tag{14}$$

Point cloud reflection intensity $Intensity$ refers to the laser reflection intensity returned by LiDAR when scanning obstacles. For objects of different materials, the reflected intensity returned varies. The variance of point cloud reflection intensity can measure the dispersion degree of reflection intensity at each point. Both car and bicycle obstacles contain metal materials. Pedestrian and bicycle obstacles also contain fabric materials. The variance can better distinguish different obstacles. The variance of point cloud reflection intensity is shown in Eq. (15).

$$E(Intensity) = \frac{\sum_{i=1}^{n}(Intensity_i - \overline{Intensity})}{n-1} \tag{15}$$

Covariance can measure the overall error of two variables. The point cloud covariance matrix is displayed in Eq. (16).

$$H = \begin{bmatrix} \mathrm{cov}(x,x) & \mathrm{cov}(x,y) & \mathrm{cov}(x,z) \\ \mathrm{cov}(y,x) & \mathrm{cov}(y,y) & \mathrm{cov}(y,z) \\ \mathrm{cov}(z,x) & \mathrm{cov}(z,y) & \mathrm{cov}(z,z) \end{bmatrix} \tag{16}$$

In Eq. (16), the H matrix is a symmetric square matrix. $\text{cov}(x,x)$ refers to the variance of x. $\text{cov}(y,y)$ refers to the variance of y. $\text{cov}(z,z)$ is the variance of z. The rest is covariance. The feature decomposition method is used to calculate the eigenvalues and eigenvectors of H. There are constant λ and n - dimensional non-zero vectors, and $Hv = \lambda v$. λ is a scalar. v is the eigenvector corresponding to the eigenvalues λ. If matrix H has n linearly independent eigenvectors, then n eigenvectors are combined to form a matrix, as illustrated in Eq. (16).

$$V = \left[v^{(1)}, v^{(2)}, ..., v^{(n)} \right] \tag{17}$$

The corresponding eigenvalues also form a vector $\lambda = [\lambda_1, \lambda_2, ..., \lambda_n]$. H refers to the matrix of 3×3, and $n = 3$. These three eigenvalues $(\lambda_1, \lambda_2, \lambda_3)$ roughly describe the distribution of point clouds in the x, y, z directions. *BPNN* is a common artificial neural network model used to solve classification, regression, and pattern recognition problems. It is a feed-forward neural network trained through a back-propagation algorithm. *BPNN* has an Input Layer (*IL*), a Hidden Layer (*HL*), and an Output Layer (*OL*). The *IL* receives the raw input data. The *HL* performs feature mapping and transformation on input data through computation. The *OL* performs classification or regression prediction based on the output of the *HL* [24]. The flowchart of the 3D obstacle target recognition method based on *BPNN* is shown in Fig. 7.

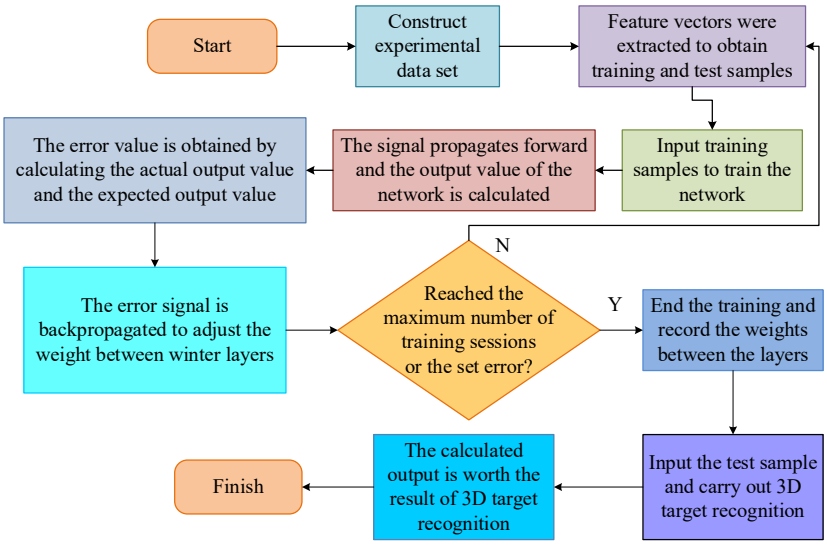


Figure 7 Flow chart of BPNN three-dimensional obstacle object recognition method based on multi-feature vector

The 3D object detection algorithm based on depth information is applied for obstacle target detection. Then, the obstacle point cloud is segmented and a dataset for classification and recognition experiments is established. By extracting the obstacle target feature vectors, *BPNN* training samples and test samples are obtained. The obtained sample data is input into *BPNN*. The output and error values are calculated. The test samples are input into *BPNN*. The output values are calculated to obtain the results of obstacle classification and recognition. Different feature vector combinations are selected. The above steps are repeated to obtain the best experimental results. The feature vector combinations are recorded. The *BPNN* architecture consists of three layers: *IL*, *HL* and *OL*. The *IL* contains 32 neurons and the *HL* consists of two sub-layers. The first *HL* has 50 neurons and the second *HL* is reduced to 30 neurons. This layer-by-layer configuration of neurons helps the network gradually abstract and extract higher-level feature representations while avoiding over-fitting. The *OL* is designed with five neurons, corresponding to the number of obstacle classes to be classified. The overall architecture flow of the 3D target detection system is shown in Fig. 8.

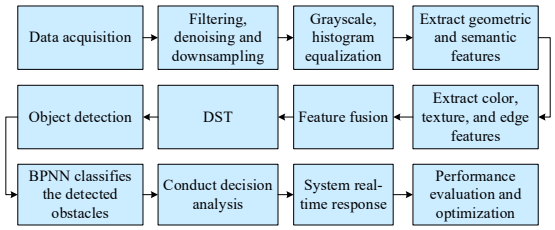


Figure 8 Overall architecture flow chart of 3D target detection system

In Fig. 8, the entire architecture process emphasizes the integrated design from data acquisition to preprocessing, feature extraction, fusion, detection, classification, and decision and response, ensuring the efficiency and accuracy of the system. Based on the application of *DST*, the system can effectively deal with the uncertainty and conflicting information in multi-source data, and improve the robustness in complex environments.

4 RESEARCH RESULTS AND DISCUSSION

To scientifically validate the performance of the proposed 3D obstacle target detection and classification recognition algorithm, experiments are conducted on the expansion corrosion algorithm based on depth information

and the BPNN recognition model based on multiple feature vectors. The output results and accuracy indicators are compared.

4.1 Analysis of Expansion Corrosion Algorithm Based on Deep Information

In order to ensure that the algorithm can effectively process and analyze the 3D point cloud data obtained from the LiDAR, a series of experimental parameters are set. The experimental parameters are shown in Tab. 2.

Table 2 Experimental parameter table

Parameter name	Value/range
Point cloud data resolution	1 cm
Area of interest size	5 m × 5 m × 3 m
Expansion corrosion structural element size	0.5 m × 0.5 m
Clustering width threshold	0.5 m - 2.0 m
Depth information range	0 - 20 m
Reflection intensity variance threshold	0 - 255
Number of layers in BPNN	3 floors
Number of neurons in BPNN	10 - 50
Learning rate	0.1
Training rounds	2000 rounds
Detection accuracy threshold	90%
Running time threshold	200 ms

In Tab. 2, the learning rate is set to 0.1 and the network weight is adjusted. The number of training rounds is 2000 to ensure that the network is fully learned. The detection accuracy threshold is set to 90% to ensure the high accuracy of the algorithm. The running time threshold is 200ms, which meets the requirements of real-time processing. Pedestrians, vehicles, and bicycles on the road of autonomous vehicles are selected as effective obstacles for detection, while other objects are selected as invalid obstacles. To verify the expansion corrosion algorithm based on depth information, the output results of the expansion corrosion algorithm based on depth information are listed, as displayed in Tab. 3.

Table 3 The output result of the improved expansion corrosion algorithm based on depth information

Cluster specification parameters	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5
Clustering width / m	0.64	0.66	1.63	0.56	0.54
Cluster length /m	0.88	1.18	4.07	0.31	0.13
Cluster area / m^2	0.51	0.77	6.44	0.18	0.08
Depth information / m	5.16	5.23	5.22	13.46	12.98

In Tab. 3, there is relatively little clustering data, indicating that each obstacle is correctly clustered and segmented. The expansion corrosion algorithm based on the improved depth information automatically selects the appropriate size of SEs, which solves the excessive and insufficient segmentation in some regions. The detection results of three different road conditions are analyzed statistically. The three sections are three forks, intersections and straight sections. The detection accuracy of 300 consecutive frames per section is shown in Fig. 9.

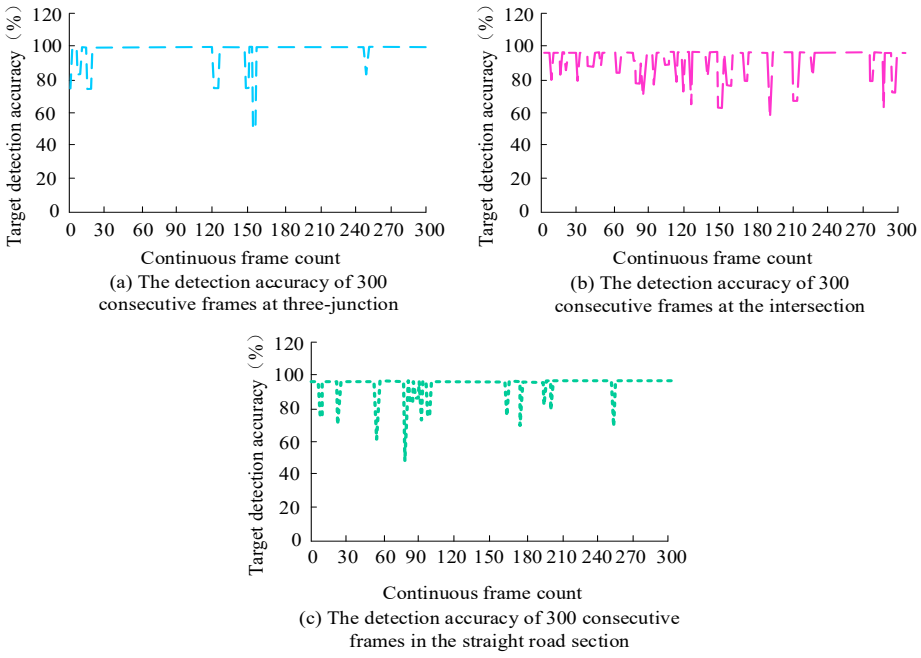


Figure 9 Detection accuracy of 300 frames in each section

In Fig. 9, the average detection accuracy of the proposed algorithm was 95.81%. If the obstacle information scanned by the LiDAR was incomplete, or there were fewer laser lines scanned by the obstacle, there was a missed detection. When there were fewer obstacles in a single frame of information, the detection accuracy of

that frame was lower. The obstacle target detection and recognition in three consecutive frames of road sections and the running time are displayed in Tab. 4.

Table 4 Experimental results of obstacle object detection and recognition in three sections

A section of a road	Target detection and recognition accuracy / %	Effective target detection accuracy / %	Effective target recognition accuracy / %	Total algorithm time / s	Average algorithm time per frame / ms
Section 1	89.87	96.43	93.20	56.24	187.47
Section 2	89.53	95.72	93.53	60.58	201.93
Section 3	88.42	95.59	92.50	55.16	183.87
Average	89.35	95.81	93.26	57.33	191.10

In Tab. 4, the recognition accuracy of effective obstacle targets was 93.26%, which met the requirements of autonomous vehicle obstacle avoidance and path planning systems. The average running time of the single frame algorithm for three road sections was 191.10 ms, which met the obstacle avoidance requirements of autonomous driving. The study conducts statistical tests to verify the importance of improvements relative to existing methods, as shown in Tab. 5.

Table 5 Statistical test result

Indicators	Improved dilation and erosion algorithm based on depth information	Traditional expansion corrosion algorithm	*T*-value	*P*-value
Accuracy rate / %	95.81 ± 0.42	92.15 ± 0.58	7.2	< 0.001
Running time / ms	191.10 ± 5.0	230.45 ± 6.2	6.8	< 0.001
Robustness score	4.5 ± 0.2	3.8 ± 0.3	5.6	< 0.001

In Tab. 5, the average accuracy and standard deviation of the improved expansion corrosion algorithm based on depth information were 95.81% and 0.42%, while the average accuracy and standard deviation of the traditional expansion corrosion algorithm were 92.15% and 0.58%. The *T*-value of *T*-test was 7.2 and *P*-value was less than 0.001, indicating that the improved algorithm was significantly better than the traditional algorithm in accuracy. The average running time of the improved algorithm was 191.10 ms, the standard deviation was 5.0 ms, the average robustness score was 4.5, and the standard deviation was 0.2. The improved algorithm is significantly better than the traditional algorithm in terms of accuracy, running time and robustness, which proves the effectiveness and importance of the improved algorithm. Multiple feature combinations are tested to determine which features are most critical to improving classification accuracy, as shown in Tab. 6.

Table 6 The influence of different feature combinations on classification accuracy

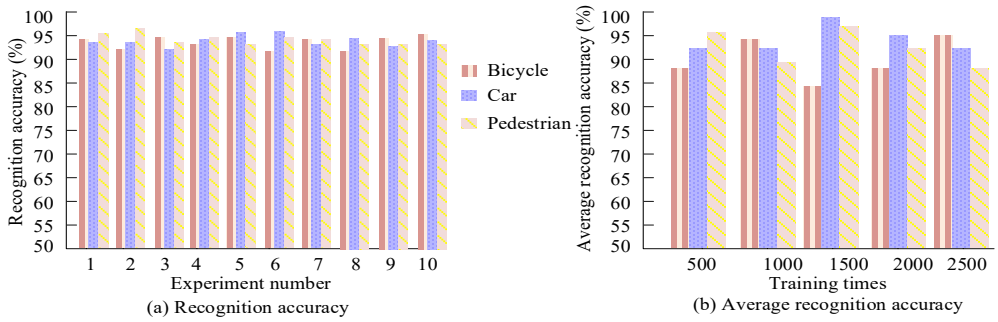
Feature combination numbering	Feature combination content	Classification accuracy / %
A	Area, volume	82.5
B	Area, volume, density	85.0
C	Area, volume, density, reflection intensity variance	88.0
D	Area, volume, density, reflection intensity variance, covariance matrix eigenvalues	91.5

In Tab. 6, the classification accuracy of feature combination A was 82.5%, indicating that basic geometric features were helpful to classification to some extent, but the effect was limited. The classification accuracy of feature combination B was improved to 85.0%, indicating that adding density features had a positive effect on classification performance. The classification accuracy of feature combination C was improved to 88.0%, which indicated that the reflection intensity variance was an effective feature, which improved the classification accuracy. The classification accuracy of feature combination D reached 91.5%, which was the highest among all combinations, showing the importance of multi-feature fusion for improving classification performance. With the increase of feature combination, the classification accuracy is gradually improved, which verifies the effectiveness of multi-feature fusion for improving classification performance.

4.2 Recognition Results Based on Multiple Feature Vectors

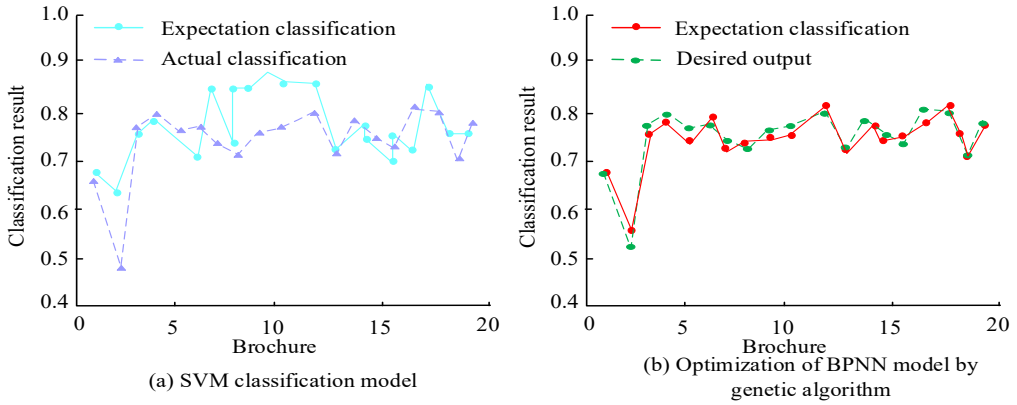
The results of the BPNN 3D target recognition test are shown in Fig. 10.

In Fig. 10, after training, the recognition results of the BPNN test dataset were very stable. The recognition accuracy of the three types of obstacles was also good. As the number of training increased, the accuracy of identifying certain types of obstacles also increased. However, as the training times increased, the training time and computational costs inevitably increased. After training to around 2500 times, the recognition results were stable. To verify the performance of the BPNN based on multiple feature vectors, a comparative experiment is conducted between the BPNN based on multiple feature vectors and the Support Vector Machine (SVM). The recognition results from 20 sets of sample data are classified. The classification results are shown in Fig. 11.



(a) Recognition accuracy (b) Average recognition accuracy

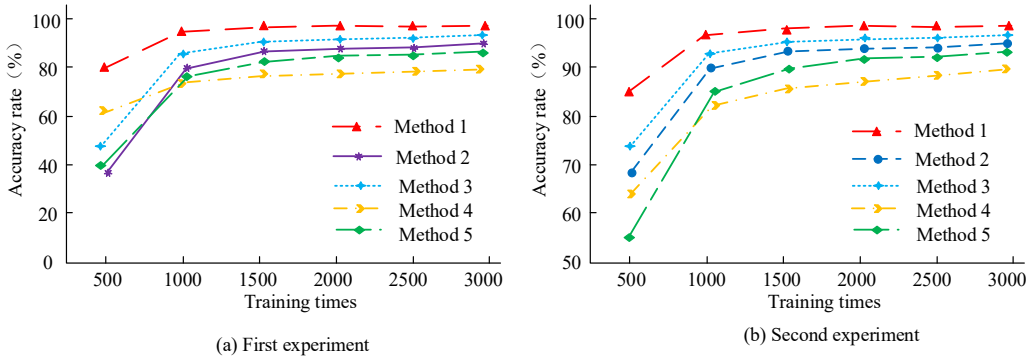
Figure 10 BPNN 3D object recognition test results



(a) SVM classification model (b) Optimization of BPNN model by genetic algorithm

Figure 11 Classification results of two recognition models

Fig. 11a displays the SVM model. The SVM model had significant errors. Fig. 11b displays the BPNN based on multiple feature vectors. The recognition and classification error of the BPNN based on multiple feature vectors was relatively small, with better performance. To compare the accuracy of 3D obstacle object detection and classification recognition algorithm based on the DST (Method 1), other object detection and classification recognition methods are compared with it, including multi-object detection and recognition algorithm based on layer feature matching (Method 2), image denoising and object detection and recognition based on convolutional neural network (Method 3), object detection algorithm based on the deep learning (Method 4) and the target classification and recognition algorithm based on SVM (Method 5). Fig. 12 displayed the results.



(a) First experiment (b) Second experiment

Figure 12 Comparison of accuracy of five algorithms

In Fig. 12, Method 1, the 3D obstacle target detection and classification recognition algorithm based on DST had the highest accuracy, reaching a maximum of 96%. The accuracy of the other four methods was 89%, 90%, 88%, and 81%, respectively, which was not as high as Method 1, indicating that Method 1 had the best performance. In the application scenarios of autonomous vehicle and robot navigation, the study further analyzes and discusses the computational complexity and running time of these methods, as shown in Tab. 7.

Table 7 The influence of different feature combinations on classification accuracy

Methods	Autonomous vehicle		Robot navigation	
	Average running time / ms	Average computational complexity / FLOPs	Average running time /ms	Average computational complexity / FLOPs
Method 1	157	5.0×10^9	140	4.0×10^9
Method 2	314	1.2×10^{10}	288	1.1×10^{10}
Method 3	455	2.0×10^{11}	440	2.2×10^{12}
Method 4	658	3.5×10^{11}	534	3.1×10^{11}
Method 5	92	2.0×10^9	83	1.8×10^9

As shown in Tab. 7, Method 1 performed better than other methods, showing lower average running time and computational complexity in autonomous vehicle and robot navigation scenarios, which makes it to have significant advantages in applications requiring rapid response. The low latency features mean that the algorithm is able to process sensor data in a timely manner, which is crucial for the safe operation of autonomous vehicles and the instant navigation of robots. In terms of computational efficiency, the low complexity of the algorithm facilitates its implementation on platforms with limited computing resources, increasing its feasibility in various environments.

5 DISCUSSION

The proposed 3D obstacle detection and classification algorithm based on DST shows great potential in the practical application of autonomous vehicle by fusing multi-source data and effectively handling uncertainty and conflict information. Compared with other methods in the existing literature, the proposed algorithm shows its superiority in many aspects. For example, compared with the two-stage 3D object detection method based on regional convolutional neural network proposed by Peng et al., this algorithm has higher accuracy and efficiency in object detection and classification recognition [9]. The experimental results showed that the average detection accuracy of this algorithm in road experiments was 95.81%, and the identification accuracy of effective obstacle targets was 93.26%, which was significantly better than the detection accuracy of 85.79% and positioning accuracy of 79.03% reported by Peng et al. In addition, the average running time of the single frame algorithm of this algorithm was 191.10 ms, which met the requirements of automatic driving obstacle avoidance. This indicates the real-time performance of the algorithm, which is crucial for the rapid response of automatic driving vehicles to environmental changes. In addition, the performance of the algorithm may be affected by the quality of the input data, so there is a certain dependence on the accuracy of the sensor and the integrity of the data. In terms of autonomous driving applications, the practical impact of the research algorithm is mainly reflected in the improvement of the vehicle environmental perception ability, which is crucial to achieve safe and efficient autonomous driving. With accurate obstacle detection and classification, vehicles are able to make better path planning and obstacle avoidance decisions, thereby reducing traffic accidents. In terms of potential applications, the algorithm can not only be applied to autonomous vehicles, but can also be extended to other fields, such as robot navigation, intelligent monitoring systems and industrial automation, where the accurate detection and classification of obstacles in 3D space is also of great value.

6 CONCLUSION

This study presents a novel 3D obstacle detection and classification algorithm based on DST, addressing key challenges in autonomous driving technology. By integrating depth information-based expansion-corrosion algorithms with multi-feature vector classification, our approach achieves high accuracy in complex 3D environments. The experimental results, demonstrating an average detection accuracy of 95.81% and an effective obstacle recognition accuracy of 93.26%, highlight the robustness and efficiency of the proposed method. The algorithm's real-time performance, with an average processing time of 191.10 ms per frame, makes it suitable for practical autonomous driving applications. While the current implementation shows promising results, future work should focus on improving performance in scenarios with partially occluded obstacles and integrating dynamic object tracking for enhanced recognition in real-world driving conditions. This research contributes significantly to the advancement of intelligent transportation systems and paves the way for more reliable and efficient autonomous driving technologies.

7 REFERENCE

[1] Sakarkar, G. & Baitule, R. (2021). Vegetable Plucking Machine Using Object Detection: A Case Study. International Journal of Scientific Research in Computer Science. *Engineering and Information Technology*, *7*(2), 501-508. https://doi.org/10.32628/CSEIT217272

[2] Daghouj, D., Abdellaoui, M., Fattah, M., Mazer, S., Balboul, Y., & Bekkali, M. E. (2021). Automatic Target Recognition Based on the Features of UWB Radar Signals. *International Journal on Engineering Applications* (*IREA*), *9*(6), 310-316. https://doi.org/10.15866/irea.v9i6.19590

[3] Liu, W., Zhang, T., Ma, Y., & Wei, L. (2023). 3D Street Object Detection from Monocular Images Using Deep Learning and Depth Information. *Journal of Advanced Computational Intelligence and Intelligent Informatics*. https://doi.org/10.20965/jaciii.2023.p0198

[4] Ovur, S.E., Zhou, X., Qi, W., Zhang, L., & Momi, E. D. (2021). A novel autonomous learning framework to enhance sEMG-based hand gesture recognition using depth information. *Biomedical Signal Processing and Control*, *66*(2), 455-463. https://doi.org/10.1016/j.bspc.2021.102444

[5] Tian, Y., Xiang, X., Peng, X., Yin, Z., & Zhang, W. (2023). Fault diagnosis strategy for few shot industrial process based on data augmentation and depth information extraction. *The Canadian Journal of Chemical Engineering*, *101*(8), 4620-4639. https://doi.org/10.1002/cjce.24818

[6] Nallasivam, M. & Senniappan, V. (2021). Moving human target detection and tracking in video frames. *Studies in Informatics and Control*, *30*(1), 119-129. https://doi.org/10.24846/v30i1y202111

[7] Iancu, D. T. & Florea, A. M. (2023). An improved vehicle trajectory prediction model based on video generation. *Studies in Informatics and Control*, *32*(1), 25-36. https://doi.org/10.24846/v32i1y202303

[8] Rani, S., Lakhwani, K., & Kumar, S. (2022). Three dimensional objects recognition & pattern recognition technique; related challenges: A review. *Multimedia Tools and Applications*, *81*(12), 17303-17346. https://doi.org/10.1007/s11042-022-12412-2

[9] Peng, Y., Yan, Y., Feng, B., & Gao, X. (2021). Recognition and classification of water surface targets based on deep learning. *Journal of Physics: Conference Series*, *1820*(3), 012166-012175. https://doi.org/10.1088/1742-6596/1820/1/012166

[10] Rubio-Solis, A., Martinez-Hernandez, U., Nava-Balanzar, L., Garcia-Valdovinos, L. G, Rodriguez-Olivares, N. A., & Orozco-Muniz, J. P. (2022). Online Interval Type-2 Fuzzy Extreme Learning Machine applied to 3D path following for Remotely Operated Underwater Vehicles. *Applied Soft Computing*, *115*(1), 1-20. https://doi.org/10.1016/j.asoc.2021.108054

[11] Qian, G., Pengjun, M, Pengju, L., Chuanpeng, H., Qian, F., & Jiarui, Z. (2020). Research on obstacle avoidance method of orchard robot based on obstacle classification. *Journal of Chinese Agricultural Mechanization*, *26*(8), 170-177.

[12] Shi, F. & Hu, X. (2022). Fuzzy Dynamic Obstacle Avoidance Algorithm for Basketball Robot Based on Multi-Sensor Data Fusion Technology. *International Journal of Foundations of Computer Science*, *33*(6/7), 649-666. https://doi.org/10.1142/S0129054122420084

[13] Zhou, H., Feng, P., & Chou, W. (2023). A hybrid obstacle avoidance method for mobile robot navigation in unstructured environment. *Industrial Robot*, *50*(1), 94-106. https://doi.org/10.1108/IR-04-2022-0102

[14] Sun, F. (2022). Disaster loss assessment of storm surge based on Dempster-Shafer theory of evidence. *Journal of Tropical Oceanography*, *41*(1), 75-81.

[15] Yuan, B. & Wei, D. A. (2021). Generalized combination rule for evidential reasoning approach and Dempster‐Shafer theory of evidence. *Information Sciences*, *547*(1), 1201-1232.

[16] Seraj, R. R. R. (2021). A hybrid GIS-assisted framework to integrate Dempster-Shafer theory of evidence and fuzzy sets in risk analysis: an application in hydrocarbon exploration. *Geocarto international*, *36*(5), 1-8. https://doi.org/10.1080/10106049.2019.1622602

[17] Adane, D. & Khanuja, H. K. (2020). Monitor and detect suspicious transactions with database forensics and Dempster-Shafer theory of evidence. *International Journal of Electronic Security and Digital Forensics*, *12*(2), 154-173.

[18] Koksalmis, E. & Özgür, K. (2020). Sensor fusion based on Dempster‐Shafer theory of evidence using a large scale group decision making approach. *International Journal of Intelligent Systems*, *35*(7), 1123-1162. https://doi.org/10.1002/int.22237

[19] Zhang, Y., Yan, W., Hong, G. S., Fuh, J. F. H., Wang, D., & Lin, X. (2020). Data fusion analysis in the powder-bed fusion AM process monitoring by Dempster-Shafer evidence theory. *Rapid prototyping journal*, *28*(5), 841-854. https://doi.org/10.1108/RPJ-10-2020-0242

[20] Zhao, K., Li, L., Chen, Z., Sun, R., Yuan, G., & Li, J. A. (2022). Optimization and applications of evidence fusion algorithm based on Dempster-Shafer theory. *Applied Soft Computing*, *124*, 109075-1-109075-16. https://doi.org/10.1016/j.asoc.2022.109075

[21] Wang, Y., Xu, X., Wang, Z., Li, R., Hua, Z., & Song, H. (2023). Shuffle Net-Triplet: A lightweight RE-identification network for dairy cows in natural scenes. *Computers and Electronics in Agriculture*. *Agric*., *205*(1), 107632-107645. https://doi.org/10.1016/j.compag.2023.107632

[22] Yang, L., Huang, Y., Hu, X., Wei, H., & Wang, Q. (2020). Multiclass obstacles detection and classification using stereovision and Bayesian network for intelligent vehicles. *International Journal of Advanced Robotic Systems*, *17*(4),1-8. https://doi.org/10.1177/1729881420947270

[23] Fang, Y., Luo, B., Zhao, T., He, D., Jiang, B., & Liu, Q. (2022). ST-SIGMA: Spatio-temporal semantics and interaction graph aggregation for multi-agent perception and trajectory forecasting. *CAAI Transactions on Intelligence Technology*, *7*(4), 744-757.

[24] Yang, M. (2022). Research on vehicle automatic driving target perception technology based on improved MSRPN algorithm. *Journal of Computational and Cognitive Engineering*, *1*(3), 147-151. https://doi.org/10.47852/bonviewJCCE20514

Contact Information

ZhiXiong JIN
(Corresponding author)
Geely University of China,
641423, Cheng Du China
E-mail: jinzhixiong@hongyicg.com

Ran WEN
Geely University of China,
641423, Cheng Du China
E-mail: windos2007@gmail.com

JiLan HUANG
Geely University of China,
641423, Cheng Du China