Research on Intelligent Extraction and Visualization Methods for Lane-Level Road Defects

Haoyu Li [1], Jianqin Zhang [1], Jianxi Ou [1],Zheng Wen [1]

1 School of Geomatics and Urban Information, Beijing University of Civil Engineering and Architecture, Beijing, 102612, China
1244722495@qq.com, zhangjianqin@bucea.edu.cn, 1108130421002@stu.bucea.edu.cn,wenzheng bucea@163.com

Keywords: Roadway inspection, Road defect detection, Rane line detection, Deep learning, Lane-level visualization

Abstract

To enhance the efficiency and visualization level of road maintenance work, this paper proposes a lane-level road defect visualization method based on multi-source data fusion. Traditional visualization methods usually only display single data or overall road conditions, which are difficult to meet the needs of intuitive presentation of complex road operation situations. To address this, this paper combines multi-source data such as Beidou GPS data, road inspection images, defect detection results, lane line information, and camera calibration to construct a complete multi-source data fusion visualization framework. Firstly, by introducing the Polar R-CNN network model to efficiently extract lane line information, and using the improved YOLOv8 model for object detection of road defects; secondly, in order to obtain the morphological features of road defects, this paper proposes an image segmentation method based on anchor box cropping and improved Otsu threshold algorithm, which effectively enhances the extraction effect of crack texture details; then, inverse perspective mapping (IPM) is used to transform the inclined images into orthographic images to achieve accurate mapping of the spatial positions of objects. The experimental results show that this method performs well in lane line detection, defect shape extraction, and spatial positioning, and can accurately visualize the display of different types of defects in multiple lanes, providing an intuitive and efficient decision support tool for road maintenance departments. The visualization scheme proposed in this paper not only enhances the interpretability and interactivity of data but also provides an important reference for the development of future intelligent road inspection systems.

1. Introduction

With the rapid development of China's transportation infrastructure and the continuous expansion of the road network, the importance of road maintenance work has become increasingly prominent. As a key link in ensuring road traffic safety and service life, the timely detection and visualization of road defects are of great significance for maintenance decision-making. Traditional road defect identification largely depends on manual inspection, which is not only inefficient and costly but also fails to meet the needs of large-scale road monitoring. In recent years, with the continuous progress of computer vision and artificial intelligence technology, image-based automatic road defect recognition methods have gradually become a research hotspot.

In practical applications, the visualization of road defects serves as an important bridge connecting identification results with maintenance decisions. (Fan et al., 2019) proposed a crack detection method that combines deep learning and threshold segmentation, which first determines whether the image contains cracks through convolutional neural network, and then carries out filtering and adaptive threshold segmentation on the crack-containing image in order to extract the cracked area. (Geng et al., 2024) improved the YOLOv8L model for embedded devices, introduced the SE attention mechanism after the C2f structure, and designed the Faster Block, which effectively improved the detection efficiency. (Zhou et al., 2025) developed a road disease detection and localization system in forest area using binocular camera and target detection network, realizing real-time identification and localization. (Han et al., 2023) used VGG and SSD models to jointly detect multiple types of diseases, and constructed a road health map based on ArcMap, which combined with a disease scatter plot to visualize the distribution of diseases.

However, the current mainstream visualization methods are mostly based on single data sources or only display the overall road conditions, lacking a detailed presentation of lane-level defect information. This coarse-grained visualization approach fails to accurately reflect the distribution characteristics of defects in different lanes, limiting the ability to manage maintenance work in a refined manner.

To address this, this paper proposes a lane-level road defect visualization method based on multi-source data fusion. By integrating Beidou GPS positioning data, road inspection images, lane detection results, and defect recognition and segmentation information, a complete visualization framework is constructed to accurately locate and visually express different types of defects in multiple lanes.

The main research content of this paper includes: (1) using the advanced Polar R-CNN network model to achieve efficient lane detection; (2) combining the improved Otsu threshold algorithm with image preprocessing strategies to extract the morphological features of road defects; (3) using inverse perspective mapping (IPM) to transform inclined images into orthographic images, enhancing spatial mapping accuracy; (4) designing a multi-source data storage structure based on the JSON format to achieve unified data management and visualization rendering.

The innovations of this study are: (1) proposing a lane-level road defect visualization approach, making up for the shortcomings of traditional methods in spatial granularity; (2) integrating various sensors and image processing technologies to enhance the accuracy and practicality of the visualization results; (3) constructing a complete technical process and data structure system, providing an extensible basic framework for the development of subsequent intelligent road inspection systems.

Through experimental verification, the method proposed in this paper has shown good performance in lane line recognition, defect shape extraction, and spatial positioning, and can effectively assist road maintenance departments in achieving more efficient and intuitive decision support.

2. Intelligent Extraction and Visualization Methods for Lane-Level Road Defects

2.1 Introduction to Multi-Source Data and Design of Data Collection System

Traditional data visualization methods mainly achieve visualization effects through a single data source. The visualization effects of a single data source are simple and cannot meet the growing complex visualization business needs. Therefore, by introducing multi-source data, it is possible to achieve visualization effects of data from multiple dimensions, effectively enhancing user experience and the intuitiveness of display. In this paper, data from multiple sources are selected for the demonstration of visualization effects, mainly including textual data, Beidou GPS receiver location data, road defect detection data, lane line data, and other intermediate data such as camera calibration. Since road sign data is three-dimensional and difficult to draw in a two-dimensional image, it is not considered in this experiment. Among them, textual data is used to explain and supplement the basic situation of the road; Beidou GPS receiver location data is used to locate the shooting point of the photo and to determine the real-world coordinates of other points in the photo; road defect detection data is used to obtain feature information of road defects for rendering on the digital base map; lane line data is used to record the position of lane lines to restore the lane lines on the map; camera calibration data is used to project the inclined images into orthographic images. The main equipment of the acquisition system includes industrial cameras, industrial lenses, industrial cables, industrial camera polarizers, and Beidou GPS. The equipment situation is shown in the Figure 1 below.

(a) Original image (b) Two types of Otsu operator segmentation

Figure 1. Self-developed road inspection equipment

2.2 Lane Line Data Extraction Based on Polar R-CNN

The Polar R-CNN network model (Wang et al., 2024), proposed in 2024, is a relatively novel anchor-based lane detection method inspired by object detection methods such as YOLO and Faster R-CNN. Currently, the academic community has introduced several anchor-based methods for lane detection, with representative works including LaneATT (Tabelini et al., 2021) and CLRNet (Zheng et al., 2022). Although anchor-based methods perform well, they mainly have two issues.

The first issue is anchor redundancy, which requires manually designing dense anchor boxes to cover various scenarios, resulting in low efficiency; the second issue is a heavy reliance on NMS (Non-Maximum Suppression) post-processing, which performs poorly in dense lane scenarios and has complex deployment.

Therefore, the Polar R-CNN network model mainly proposes two improvements to address the aforementioned issues: introducing a Local Polar System and a Global Polar System based on the polar coordinate system to create more accurate anchor points, thereby reducing the number of anchors proposed in sparse scenarios; and proposing an NMS-Free detection framework to address the complexity of NMS post-processing, introducing a triple head structure with a GNN block to improve deployment efficiency and performance in dense lane scenarios. The main structure is shown in the Figure 2 below.

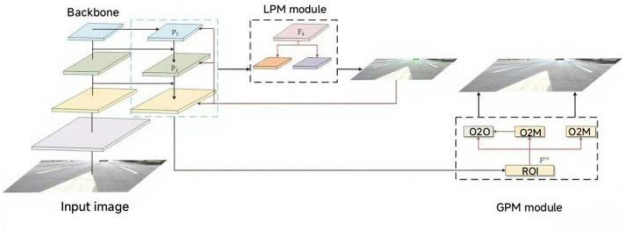


Figure 2. Polar R-CNN network structure diagram

Polar R-CNN mainly includes four modules: the Backbone main network, the FPN module, the LPM anchor generation module, and the GPM detection module. The main network is responsible for extracting high-level semantic features of the image, while the FPN further performs multi-scale feature fusion to enhance the model's adaptability to lanes of different shapes. The LPM module, based on the idea of polar coordinates, predicts the lane anchor points and their confidence at each position on the feature map. Through training, the model can generate more accurate and fewer candidate anchor points, thus reducing redundant calculations. The GPM module extracts features from these candidate anchor points and completes the final lane detection through a triple head structure that includes O2M classification, O2M regression, and O2O classification, achieving an efficient inference process without NMS. This paper optimizes the main network for adaptability while keeping the main structure of Polar R-CNN unchanged. The original Polar R-CNN uses a general Backbone structure (such as the ResNet series), while this paper selects ResNet18 and DLA34 as the main networks for comparative experiments based on the characteristics of the road dataset. Through comparative experiments, it was found that DLA34 outperforms ResNet18 in terms of precision (Precision), recall rate (Recall), and F1@50 metrics (see Table 2), so DLA34 was ultimately chosen as the main network. This adjustment improves the model's performance in lane detection tasks and represents a structural adaptation and performance optimization of Polar R-CNN for specific application scenarios, achieving improved detection performance.

2.3 Disease Morphology Extraction Based on the Improved Otsu Algorithm

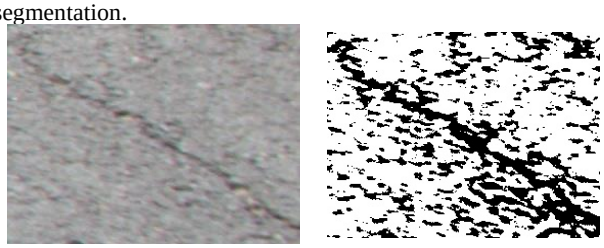
Since the anchor boxes in object detection only indicate the relative position of road defects and lack information such as texture, shape, length, and area of the defects, it is necessary to use relevant image segmentation algorithms to extract the texture and morphological information of road defects to enhance the visualization effect and facilitate the statistical analysis of road defect areas.

To this end, this paper introduces an anchor-box-based image segmentation method that can better balance the advantages and disadvantages of traditional methods and deep learning

algorithms. Firstly, the object detection network has low hardware requirements and fast model inference speed, which can meet the real-time detection needs of road inspection using edge computing devices; secondly, visualization does not require high segmentation accuracy, only the recording of relevant data on road defect morphology, without the need for statistical analysis of defect areas; finally, the implementation cost of this method is low, and there is no need for pixel-level annotation of road defect datasets, reducing manual annotation costs.

The Otsu threshold algorithm (Chen et al., 2021), also known as the maximum inter-class variance method, is a method that can automatically calculate the threshold value for image segmentation. Its basic idea is to divide the pixels in the image into two categories based on the grayscale information characteristics of the image, and the threshold value is optimal when the variance between these two categories is maximized.

However, the traditional Otsu operator is easily affected by the grayscale distribution of the image itself and environmental noise, leading to unsatisfactory segmentation results in most cases, and due to the binary classification of the traditional Otsu operator, there is a situation of over-segmentation when facing complex environments. Especially during road inspection, road cracks and asphalt colors do not have a large deviation, often leading to over-segmentation.



(a) Original image (b) Two types of Otsu operator segmentation

Figure.3 Two types of Otsu operator segmentation effect diagram

As can be seen from the content of Figure 3, due to the reflection effect of ground asphalt, there will be a situation similar to the gray scale of road cracks, so it is difficult for the traditional Otsu operator to extract road crack information finely.

Therefore, this paper proposes an improvement strategy for the Otsu operator. Firstly, to reduce the impact of environmental noise on road cracks, logarithmic transformation and bilateral filtering are used as preprocessing techniques. Secondly, to address the poor segmentation performance of the two-class Otsu algorithm, research indicates that there are typically four different gray levels of objects in road surface scenes, leading to the introduction of the four-class Otsu algorithm. Thirdly, due to the high computational load of the four-class Otsu algorithm, optimization strategies are proposed to effectively reduce the computational load. Finally, relevant post-processing techniques are introduced to enhance the segmentation effect and accuracy of the algorithm.

2.3.1 Image Enhancement Preprocessing

In the image preprocessing enhancement part, the logarithmic transformation method is first used to enhance the difference degree between road cracks and background. The formula of logarithmic transformation is as follows:

$$s = c \times \log(1 + r) \quad (1)$$

$$c = \frac{255}{\log(1 + \max(r))} \quad (2)$$

Where is rsc the pixel value of the original image, is the pixel value of the transformed image, and is the scaling constant.

After the logarithmic transformation, the noise in the image is also amplified, necessitating denoising. Since bilateral filtering can effectively denoise images while preserving edges, it was chosen as the method$p = (x, y)\Omega I_{filtered}(p)$. Given a pixel and its neighborhood, the filtered value is calculated using the following formula.

$$I_{filtered}(p) = \frac{1}{W_p}\sum_{q\in\Omega} I(q) \times f_r(|I(p) - I(q)|) \times f_s(||p - q||) \quad (3)$$

$$f_r(|I(p) - I(q)|) = exp(-\frac{|I(p) - I(q)|^2}{2\sigma_r^2}) \quad (4)$$

$$f_s(||p - q||) = exp\,(-\frac{||p - q||^2}{2\sigma_s^2}) \quad (5)$$

$$W_p = \sum_{q\in\Omega} f_r(|I(p) - I(q)|) \times f_s(||p - q||) \quad (6)$$

Using the above method, the image achieves a good denoising effect while preserving the details of the cracks.

2.3.2 Four-class Otsu Algorithm and Optimization Method

The traditional Otsu algorithm performs binary segmentation by using a single threshold to separate the image into two classes. The four-class Otsu algorithm, on the other hand, introduces multiple thresholds. Its calculation process is similar to that of the traditional Otsu algorithm. First, the grayscale image is computed, and the probability density of each gray level is determined. Then, the overall mean gray value of the image is calculated using the corresponding formula.

$$\mu_T = \sum_{0}^{255} x \times p(x) \quad (7)$$

For each pair$(t_1, t_2, t_3) t_1 < t_2 < t_3$, the inter-class variance is calculated by the following formula.

First, use the following formula to calculate $\omega_0\omega_1\omega_2\omega_3$ the weight of each class:

$$\omega_0 = \sum_{x=0}^{t_1} p(x) \quad (8)$$

$$\omega_1 = \sum_{x=t_1+1}^{t_2} p(x) \quad (9)$$

$$\omega_2 = \sum_{x=t_2+1}^{t_3} p(x) \quad (10)$$

$$\omega_3 = \sum_{x=t_3+1}^{255} p(x) \quad (11)$$

Then use the following formula to calculate $\mu_0\mu_1\mu_2\mu_3$the mean of each class:

$$\mu_0 = \frac{\sum_{x=0}^{t_1} x \times p(x)}{\omega_0} \quad (12)$$

$$\mu_1 = \frac{\sum_{x=t_1+1}^{t_2} x \times p(x)}{\omega_1} \quad (13)$$

$$\mu_2 = \frac{\sum_{x=t_2+1}^{t_3} x \times p(x)}{\omega_2} \quad (14)$$

$$\mu_3 = \frac{\sum_{x=t_3+1}^{255} x \times p(x)}{\omega_3} \quad (15)$$

$$a_b^2(t) = \omega_0 \times (\mu_0 - \mu_T)^2 + \omega_1 \times (\mu_1 - \mu_T)^2 + \omega_2 \times (\mu_2 - \mu_T)^2 + \omega_3 \times (\mu_3 - \mu_T)^2 \quad (16)$$

Since the four-category Otsu algorithm $(t_1, t_2, t_3)S_1$enumerates each pair, the number of enumeration can be obtained by the following formula.

$$S_1 = C(256,3) \approx 2.7 \times 10^8 \quad (17)$$

From the above equation, it can be observed that the computational complexity of the four-class Otsu algorithm is relatively high. To address this, an approximate search strategy is introduced. By adjusting the gray level range [0, 255] with a step size of 4 and 2, the number of parameters can be reduced while maintaining a similar segmentation performance. When the step size is 4, the number of enumeration combinations is S_2; when the step size is 2, the number of enumeration combinations is S_3. Therefore, according to the following formula, we can obtain:

$$S_2 = C(64,3) = 39711 \quad (18)$$

$$S_3 = C(128,3) = 326{,}976 \quad (19)$$

Therefore, it can be concluded that when the step size is set to 2, the computational load is reduced by approximately 828 times; when the step size is 4, the computational load is reduced by approximately 6,850 times.

2.3.3 Image Post-Processing

Since the Otsu algorithm performs threshold segmentation based on grayscale images, larger noise regions in the original image with gray values similar to those of cracks may still be classified as part of the cracks. Therefore, after applying the Otsu segmentation, this paper introduces an image post-processing procedure to address this issue. The post-processing method used includes morphological filtering and area filtering.

Morphological filtering mainly consists of two operations: erosion and dilation. Dilation expands the foreground regions of the image using a structuring element, while erosion shrinks the foreground pixels and expands the background region. Opening operation is defined as first performing erosion followed by dilation, and is primarily used to remove small noise points from the image. Closing operation involves first performing dilation followed by erosion, and is used to fill gaps between objects and connect broken parts.

$$A \circ B = (A \ominus B) \oplus B \quad (20)$$

$$A \cdot B = (A \oplus B) \ominus B \quad (21)$$

Area filtering is a morphological operation that processes targets or noise in an image based on the size of their areas. Typically, an area threshold is set to either remove objects smaller than the threshold or retain objects larger than the threshold.

$$A(C_i) = \sum_{p \in C_i} 1 \quad (22)$$

2.4 Monocular Distance Measurement Based on Inverse Perspective Transformation

In this paper, since the ground within the camera's field of view can be approximately abstracted as a plane, the method of inverse perspective mapping based on the ground plane is selected to achieve monocular distance measurement.

When the camera captures a scene, the resulting image is a projection of the 3D world coordinates onto the 2D image coordinate system. This process is known as perspective mapping, which is similar to the principle of pinhole imaging. In contrast, inverse perspective mapping (IPM) is an image processing technique that transforms a perspective-distorted image into an orthographic (top-down) view. The coordinate transformation process is illustrated in Figure 4.

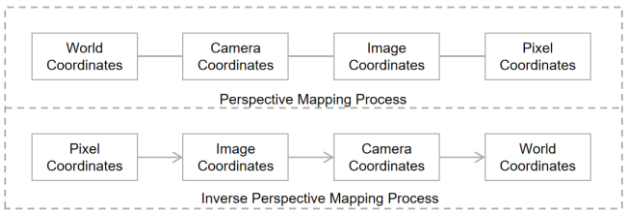


Figure.4 Coordinate transformation process diagram

Since the vehicle is fixed on the front side for collection, it only needs to calculate the single H correspondence matrix once at the beginning of collection. The specific process of calculating the single correspondence matrix is as follows.

2.4.1 Equipment Installation

First, install the equipment. Since the collection distance is set to 20 meters, use a steel tape measure to draw a dividing line 20 meters away, parallel to the ground projection of the optical axis. Next, adjust the camera and the equipment angle to ensure the equipment is level, the line of sight is centered, and the upper boundary of the image aligns with the dividing line. This completes the installation of the equipment.

2.4.2 Control Point Drawing and Measurement

Since the coordinates of the control points in a rectangle are easy to calculate, the four corner points of $abYXX - X$the rectangle are selected as control points. Then, a rectangle is drawn on the ground with a length of meters and a width of meters, perpendicular to the direction of the ground projection of the optical axis. Next, the distance from the rectangle to the camera is measured and denoted as, and the offset distance between the rectangle's centerline and the direction of the ground projection of the optical axis is denoted as. When the rectangle is left-shifted, it is denoted as, and when it is right-shifted, it is denoted as.

2.4.3 Coordinate Extraction and Position Calculation

Import the captured images and use the PyCharm script to obtain the coordinates of the four control points in the image, starting $(u_1, v_1)(u_2, v_2)(u_3, v_3)(u_4, v_4)(W, H)$from the top-left corner of the rectangle and marking them clockwise as,,, and. To ensure the relative position of the drawn rectangle is correct after transformation, it is also necessary to calculate the relative position of the real-world rectangle in the pixel coordinate system. Assuming the projected image size is and the height is.

First, define the scale in the image coordinate system and the pixel coordinate system. The scale $c_x$is denoted as scale, and the value of the central pixel of the image is. The formula is as follows:

$$scale = \frac{H}{20} \quad (23)$$

$$c_x = \frac{W}{2} \quad (24)$$

Therefore, the coordinates of the real-world rectangle in the pixel coordinate system can be derived.

$$(x_1, y_1) = (c_x - ((\frac{a}{2} + X) \times scale), H - ((Y + b) \times scale)) \quad (25)$$

$$(x_2, y_2) = (c_x + ((\frac{a}{2} - X) \times scale), H - ((Y + b) \times scale)) \quad (26)$$

$$(x_3, y_3) = (c_x + ((\frac{a}{2} - X) \times scale), H - (Y \times scale)) \quad (27)$$

$$(x_4, y_4) = (c_x - ((\frac{a}{2} + X) \times scale), H - (Y \times scale)) \quad (28)$$

2.4.4 Calculation and Transformation of the Homography Matrix

After obtaining the control points and their corresponding image points, the homography matrix is computed using OpenCV's cv2.getPerspectiveTransform(); the inverse perspective transformation is then applied to the image via cv2.perspectiveTransform.

3. Experimental Results and Analysis

3.1 Lane Line Extraction

Since the road images captured by this collection system do not match the perspective of commonly used datasets, to ensure the detection accuracy of the model for the inspection data collected by this system, it is necessary to annotate the captured images. After annotation, a lane line dataset containing 2500 samples was obtained. To ensure the training effect of the model, the data was randomly divided into training and validation sets at a ratio of 80% and 20%.

After the data division, the model training process was carried out. To compare the performance of different backbone networks in the actual lane line extraction task, ResNet18 and DLA34 were selected as the backbone networks for comparative experiments. According to the requirements of the task and the characteristics of the data, it is necessary to configure the model's parameters, such as input image size, learning rate, and other parameters. During the training process, parameters and optimizations are adjusted according to the changes in the model's loss function and evaluation metrics. The main configuration parameters are shown in Table 1.

Parameter Name	Meaning	Configuration	Parameter Name	Meaning	Configuration
backbone	Backbone Network	DLA34/ ResNet18	epoch_num	Number of Training Epochs	300
pretrained	Pre-training	True	lr	Learning Rate	0.0006
batchsize	Batch	16	max_lanes	Maximum Number of Lanes	4
imgsz	Input Size	320×800	conf_thres	Confidence Threshold	0.48

Table 1 Training parameters

To ensure the reliability of the experimental results, all comparative experiments were conducted under the same software environment and hardware configuration. After model training, the test set was used to validate the model. The experimental results are shown in Table 2.

Model	Backbone Network	P	R	F1@50
Polar R-CNN	ResNet18	71.65	70.39	71.01
Polar R-CNN	DLA34	71.83	70.54	71.18

Table 2 Comparative experimental results

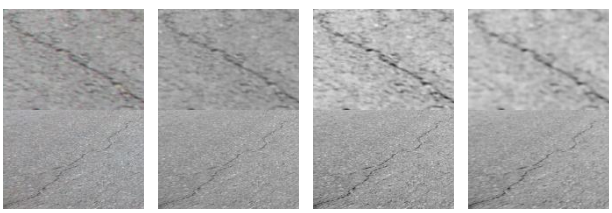
According to the experimental results, the DLA34 backbone network has better accuracy and recall rate compared to the ResNet18 backbone network. Therefore, in this paper's lane line detection task, the DLA34 backbone network is selected as the main model. The specific recognition results can be seen in Figure 5.

(a) Original image (b) Lane extraction image

Figure.5 Lane line network extraction result image

3.2 Disease Morphology Extraction

First, the crack image is subjected to image grayscale conversion, logarithmic transformation preprocessing, and bilateral filtering preprocessing. The preprocessing results are shown in Figure 6. Among them, 6(a) is the original image; 6(b) is the grayscale image after grayscale conversion; 6(c) is the image after logarithmic transformation of the grayscale image; and 6(d) is the final result after bilateral filtering.



(a) Original image (b) Grayscale (c) Logarithmic transformation (d) Bilateral filtering

Figure.6 Image preprocessing results

To verify the effectiveness of the Otsu optimization method, the runtime and segmentation performance of the two-class Otsu, three-class Otsu, four-class Otsu, and the four-class Otsu algorithm with an approximate search strategy were compared. The comparison results are shown in Table 3 below.

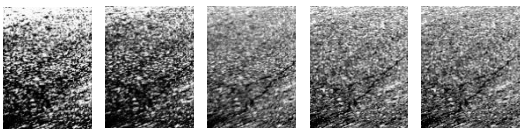
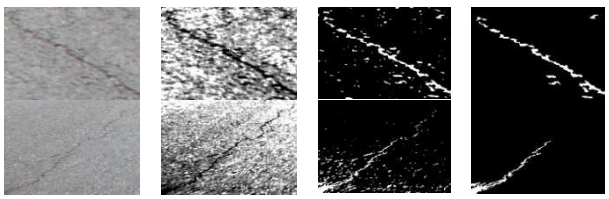
	Two-Class Otsu	Three-Class Otsu	Four-Class Otsu	This algorithm(Step=2)	This algorithm(Step=4)
Runtime/s	0.003002	0.006387	43.797021	5.623630	0.680031
Segmentation Performance					

Table 3 Comparison table of methods

According to the above table, the four-class Otsu method has a relatively long runtime. The optimized algorithm proposed in this experiment can effectively reduce the computation time. Furthermore, based on the segmentation results, it can be seen that the optimized algorithm has minimal impact on the image quality, which verifies the effectiveness of the optimization algorithm.

Finally, by integrating the post-processing workflow, the complete extraction of road cracks using a four-class Otsu optimized algorithm with a step size of 2 was achieved. The extraction results are shown in Figure 7. Figure 7(a) represents the input original image; Figure 7(b) shows the result after four-class Otsu segmentation; Figure 7(c) displays the result after morphological closing operation; and Figure7(d) illustrates the final result after area filtering.



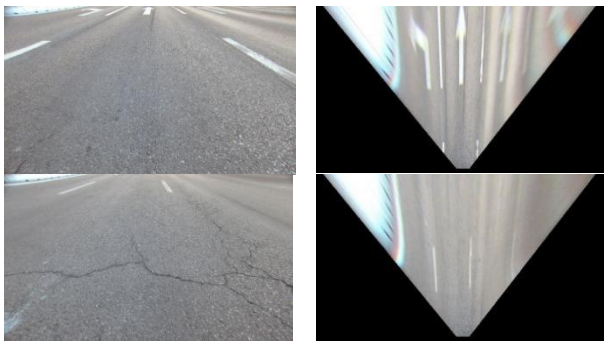
(a) Original image (b) Four categories Otsu (c) Closing operation (d) Area filter

Figure.7 Road morphology data extraction through layer map

The above figure 7 shows that the four-class Otsu segmentation algorithm can effectively extract the morphological features of road defects. Additionally, the morphological closing operation helps to expand the background region.

3.3 Perspective Transformation

In this paper, a control point rectangle is first constructed, and the homography matrix is calculated. The homography matrix is then used to perform an inverse perspective transformation on the original image. The specific results are shown in Figure 8. Among them, Figure 8(a) is the original image, and Figure 8(b) is the transformed image.



(a) Original image (b) Transformed image

Figure.8 Inverse perspective transformation effect diagram

According to the above Figure 8, the inverse perspective transformation algorithm achieves good results, accurately restoring the relative positions of objects in the real world. At the same time, it can be seen that the position calculation method proposed in this paper effectively reconstructs the relative position of the control point rectangle in the real world.

3.4 Visualization Results

After constructing the data storage structure, the input image is visualized by extracting information from the structural data. First, the input image is fed into the Polar R-CNN network and the object detection algorithm to extract lane line and defect bounding box data. The visualization results of the lane line and defect bounding box extraction are shown in Figure 9. Figure 9(a) represents the original input image; Figure 9(b) shows the result after lane line extraction using the network; and Figure 9(c) displays the bounding boxes of detected defects obtained through the object detection model.

(a) Original image (b) Lane line extraction image (c) Disease anchor diagram

Figure.9 Lane line and defect anchor frame extraction

Then, the lane line information is recorded and stored in the structural data file. The images within the defect bounding boxes are cropped, and the cropping information is recorded. The cropped images are saved, and the cropping schematic is shown in Figure 10. Subsequently, the cropped defect detail images are processed using the method described in Section 2.3 of this paper to extract the morphological information of the defects. The specific results are shown in Figure 11.

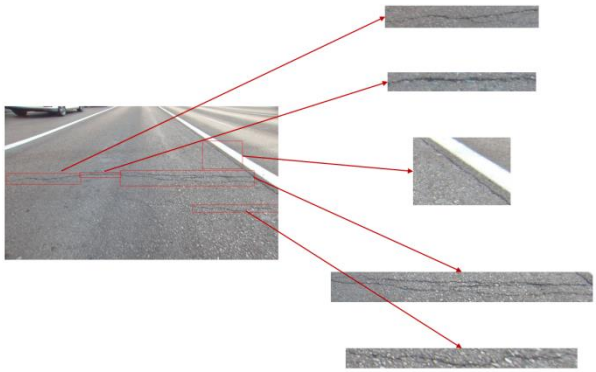
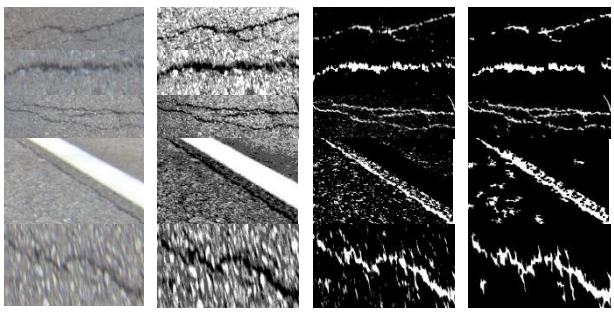


Figure.10 Image cropping effect diagram



(a) Original image (b) Four categories Otsu (c) Closing operation (d) Area filter

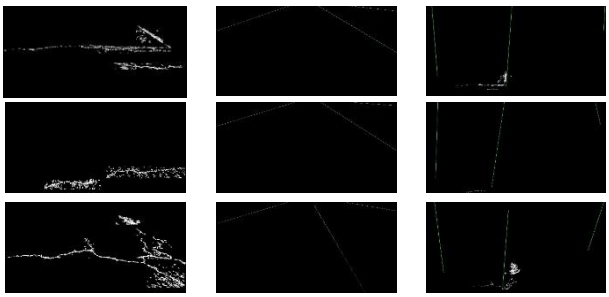
Figure.11 Post-segmentation extraction flow chart

After extracting the cropped images, the threshold segmentation results are recorded and saved into the corresponding structural data files. The threshold segmentation results are then mapped back to the original image using coordinate mapping. At the same time, the lane line information is read and also mapped back onto the original image. The reconstructed results are shown in Figure 12.

(a) Original image (b) Threshold restore (c) Overall restoration diagram

Figure.12 Data restoration effect diagram

Through observation of the above results, it can be seen that the proposed algorithm achieves good extraction performance for deep cracks. To eliminate the influence of the background on the visualization and to enhance the clarity of the visual results, the lane line information and road crack feature information mentioned above are extracted using a mask. After applying the inverse perspective transformation to the extracted masks, an orthographic visualization result of the lane lines and defects is obtained. The specific results are shown in Figure 13.



(a) Disease feature mask (b) Lane mask (c) Fusion visualization results

Figure.13 Final rendering

4. Conclusion

This paper presents a lane-level road defect visualization method based on multi-source data fusion to improve the efficiency and visualization level of road maintenance work. Traditional visualization methods often lack the spatial granularity to display detailed lane-level defect information, making it difficult to support refined road maintenance decision-making. To address this issue, the study integrates multi-source data, including road inspection images, lane line detection results, and road defect recognition and segmentation information, to construct a comprehensive visualization framework.

The proposed method employs advanced deep learning models, such as the Polar R-CNN network for efficient lane line detection and an improved YOLOv8 model for road defect detection. Additionally, an image segmentation approach combining anchor box cropping and an improved Otsu threshold algorithm is introduced to extract detailed morphological features of road defects. Inverse perspective mapping (IPM) is applied to transform inclined images into orthographic views, ensuring accurate spatial positioning of defects.

Experimental results demonstrate that the proposed method performs well in lane line detection, defect shape extraction, and spatial mapping. The visualization framework provides an intuitive and interactive representation of lane-level road defects, offering road maintenance departments an efficient decision-support tool. Furthermore, the integration of multi-source data and image processing technologies enhances the practicality and accuracy of the visualization results, making this method a valuable reference for the development of future intelligent road inspection systems.

In conclusion, the proposed lane-level road-defect visualization method not only enhances the interpretability and intuitiveness of the data, but also advances the further refinement of intelligent road inspection technologies.

References

Chen C, Seo H, Jun C H, et al., 2022: A potential crack region method to detect crack using image processing of multiple thresholding[J]. *Signal, Image and Video Processing* ,16(6): 1673–1681.

Fan R, Bocus M J, Zhu Y, et al., 2019: Road crack detection using deep convolutional neural network and adaptive thresholding.

Proceedings of the IEEE Intelligent Vehicles Symposium (IV) , IEEE, pp. 474 – 479.

Geng Huantong, Liu Zhenyu, Jiang Jun et al., 2024: An embedded road crack detection algorithm based on improved YOLOv8. *Computer Applications* , 44(5), 1613 – 1618.

HAN Yu, ZHANG Meng, LI Yuhong et al., 2023: Intelligent and comprehensive pavement disease detection method based on deep learning and ArcMap. *Journal of Jiangsu University (Natural Science Edition)* , 44(4), 490 – 496.

Tabelini L, Berriel R, Paixao T M et al., 2021: Keep your eyes on the lane: Real-time attention-guided lane detection. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* , 294 – 299.

Wang S, Liu J, Cao X et al., 2024: Polar R-CNN: End-to-end lane detection with fewer anchors. *arXiv preprint arXiv:2411.01499* .

ZHOU Jiasun, LI Shuang, LI Junhui et al., 2025: Road disease detection and localization system in forest areas. *Journal of Forest Engineering* , 10(1), 152–159.

Zheng T, Huang Y, Liu Y et al., 2022: Clrnet: Cross layer refinement network for lane detection. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* , 898–907.