

Article

Twisting Pathway Understanding Inspired by Global Precedence Effect in a Field Environment

Luping Wang^{1,*}, Lijuan Chen¹, Yuan Feng², Shanshan Wang³ and Hui Wei⁴

¹ Laboratory of 3D Scene Understanding and Visual Navigation, School of Mechanical Engineering, University of Shanghai for Science and Technology, Jungong Road 516, Shanghai 200093, China

² State Grid Shanghai Electric Power Company Construction Consulting Branch, Shanghai 200120, China

³ Intel Asia-Pacific Research & Development Ltd., Zixing Road 880, Shanghai 201100, China

⁴ Laboratory of Algorithms for Cognitive Models, School of Computer Science, Fudan University, Zhangheng Road 825, Shanghai 201203, China

* Correspondence: 15110240007@fudan.edu.cn

How To Cite: Wang, L.; Chen, L.; Feng, Y.; et al. Twisting Pathway Understanding Inspired by Global Precedence Effect in a Field Environment. *Control and Robotics Express Communications* 2026, 1(1), 5.

Received: 11 March 2026

Revised: 3 July 2026

Accepted: 6 July 2026

Published: 21 July 2026

Abstract: Scene understanding is a core prerequisite for robots' autonomous decision-making. However, real-world application faces challenges like understanding unknown, unstructured twisting paths in field environments. Traditional methods using 3D point clouds or multi-sensor fusion are energy-intensive and computationally expensive, making them laborious for resource-limited field robots. Inspired by the global precedence effect (GPE)—the phenomenon that humans perceive global stimuli preferentially, aiding understanding of multiple zigzag paths—we present a method using only a monocular camera to understand unstructured twisting paths. We extracted improved edge-based lines. Multiple curves were formed by grouping lines based on proximity and continuity, inspired by GPE. By analyzing curve orientations, twisted zigzag paths can be reshaped and twist direction estimated, providing decision basis for robot autonomous navigation. Unlike deep learning algorithms, our method is highly interpretable without prior training or camera internal parameters. It has low energy consumption and low investment, requiring no external equipment. Pure geometric reasoning contributes to robustness under lighting variations and various colors. Pixel-wise classification accuracy was compared with ground truth. Experimental results show the method successfully elucidates unstructured twisting paths, providing navigation-relevant scene interpretation in field environments.

Keywords: twisting pathway; global precedence effect; field environment

1. Introduction

Autonomous decision-making in robots relies on robust 3D scene understanding. However, interpreting unknown, unstructured twisting paths in field environments remains a challenging problem. Computational, memory and energy resources on field robots are very limited, so low-power, low-complexity twisting path understanding algorithms are more suitable for applications on resource-constrained field robots.

Traditional methods based on 3D point clouds or multiple sensors have high energy consumption, high cost, and low efficiency, which greatly hinder their application in the acquisition and processing of information on the mobile side of field scenes with variable environments. In contrast, the use of low-cost monocular cameras, combined with computational methods inspired by human cognitive mechanisms, is more advantageous in understanding the scene in terms of consumption and efficiency, and is also more applicable to field environments.



Copyright: © 2026 by the authors. This is an open access article under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Publisher's Note: Scilight stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Humans are more inclined to perceive global information when stimuli from the external environment are present, which is known as the global precedence effect (GPE). It is indicated that eccentricity and size are the main determinants of the GPE [1,2]. When the stimulus was displayed at a peripheral location, relative size was probably the prime determinant of global advantage and interference. Although the whole of a stimulus is composed of parts, global stimulus information has a higher visual priority. Trackable visual cues from natural stimuli are more useful for understanding a 3D scene and have advantages in terms of energy consumption and efficiency compared to 3D point clouds or fused sensor approaches [3]. The human visual system can fully and efficiently utilize fragmented stimulus cues to analyze the relative geometric relationships between different stimuli and thus infer their spatial organization. This cognitive mechanism contributes to our understanding of twisting paths in wild environments. It should be clarified that our use of the GPE is conceptual rather than computational. We do not implement a neural or psychophysical model of GPE; rather, we take inspiration from the phenomenon to justify why grouping fragmented lines into global curves—based on proximity and continuity—is a reasonable strategy for understanding twisting paths in field environments.

As shown in the Figure 1, unlike structured indoor and urban scenes, there are a wide variety of road surfaces in wilderness environments, such as grass, sand, ice, snow, and muddy pavement, resulting in unstructured, intermittent, and blurred road boundaries [4]. In addition, the ever-changing lighting and colors in natural environments make most edges fragmented and disorganized, which poses a significant obstacle to understanding unstructured twisting paths in wilderness environments.



Figure 1. A twisting pathway in field environment.

In this study, we present a methodology to understand unstructured twisting pathways through a low-cost monocular camera without prior training. First improved edge-based lines are detected. Inspired by perceptual global precedence effect, multiple curved lines, made up of a number of lines that follow local grouping principles such as proximity and continuity, are formed. Based on geometric inferences of curved lines, twisting surfaces of unstructured pathways are reshaped and corresponding orientations are estimated, which helps to understand an unstructured twisting pathway.

For machine learning methods, it requires a large amount of labeled data and pre-training, and thus requires the support of powerful computational resources, such as high-performance graphics processors (GPUs) and high-capacity storage devices, which greatly hinders the fusion of the physical space of the scene with the information space. Unlike deep learning-based methods, no prior training is needed. Furthermore, the low-cost algorithm can process the captures without knowledge of the camera's intrinsic parameters nor of precise depth. Pure geometric inferences endow the proposed method reliable robustness against varying illumination and colors, enabling more stable understanding of field environment.

The main contributions of this work are threefold:

- (1) We establish a systematic mapping from the two core properties of the global precedence effect—global advantage and global interference—to a two-stage curve extraction pipeline (candidate generation and refinement), providing a cognitively grounded alternative to purely heuristic line grouping.
- (2) We propose a training-free, calibration-free geometric framework that not only segments path surfaces but also estimates their spatial orientation, providing richer navigation-relevant information than binary segmentation alone.

- (3) Unlike parametric curve-fitting or learning-based methods, our approach jointly recovers boundaries and orientation from monocular input without camera intrinsics or pre-training, offering a lightweight alternative for field robotics.

The percentage of pixels that were evaluated through comparing estimated twisting pathways to the ground truth. Experimental results demonstrated that the proposed methodology can successfully understand field twisting pathways and has promising application prospects in energy consumption and computational efficiency.

2. Related Work

Many approaches for reconstructing scenes were introduced like SFM [5] and visual SLAM [6]. A tracking model was introduced to address the problem of occlusions within large groups of featureless objects through 3D point clouds [7]. However, 3D point cloud-based methods are very energy intensive and appear to be helpless when considering geometric cues such as edges, textures and surfaces.

The detection of roads has been the top priority of automatic driving research. Segmenting the curves by lane line detection is the more widely researched approach. Among the methods in recent years, there are [8] based on real-time lane detection, where the region of interest (ROI) is input and the inverse IPM is applied to get the output stream. There are also methods [9] and point cloud extraction methods [10] that use RANSAC to recognise lane models and estimate selected lane patterns. Also, a deep neural network called LaneNet is used for lane detection [11]. However, these methods, which lack cognitive mechanisms, have difficulty understanding unstructured multiple zigzag pavements due to the disruptive clutter of the field environment.

Semantic segmentation is a more widespread road segmentation method in recent years. Geometric segmentation were introduced for diverse structured environments such as indoor and urban scenes [12]. Viswanath proposed an off-road semantic segmentation (OFFSEG) framework [13], which employs widely adopted deep learning backbones for merge-class semantic segmentation and incorporates a colour segmentation method to enhance scene understanding. Ding et al. proposes a multi-class probability pyramid as a representation built on top of pixel-level semantic scene labels [14]. This representation unravels the variations in scene semantics across resolutions by controlling the granularity of the local semantic information to form a region-level abstraction. Metzger's performance of modern semantic segmentation models has also been evaluated, especially their segmentation efficiency [15], and their experiments demonstrated the advantages of finegrained semantic classes in improving the overall prediction accuracy, and used Boundary Jaccard (BJ) score with IoU as the evaluation criteria. The success of neural networks in image classification has fuelled the development of neural networks dedicated to semantic image segmentation, such as FCN [16], SegNet [17], PSPNet [18] and DeepLabv3 [19]. As a result, the researchers have also shared test datasets dedicated to semantic segmentation [20–22], which include urbanised roads as well as unstructured roads. However, these data-driven methods are prone to failure due to a lack of estimating unstructured surfaces of twisted pathways.

There is also a widely used segmentation method for curves in roads is the curve fitting method. There are curve recognition algorithms [23] based on straight curve models through machine learning [24], which can obtain the mathematical equations of the straight line model by using the improved Hough variation, and build the curve model based on the continuity of the lane lines as well as the tangent relationship between the straight line model and the curve model. The curve fitting method is used to achieve the detection and recognition of straight and curved lanes. Similarly Kasyap has based on the Hough transform [25], from which the bicurve fitting technique required for interpolating paths is adopted. Pure geometric models were established to detect the walkable floor but they rely on accurate vanishing points [26]. Galimova has worked out an MPC based path tracking control algorithm for road trains [27], which proposes a prediction model and a formula for calculating the reference state and control formulas for the inputs. However, these methods cannot account for the geometric orientation of unstructured surfaces with multiple curved paths.

In addition to most of the algorithms studied, road detection has also been performed by other methods. Based on other hardware aspects, Dai has processed the turning manoeuvres of vehicles in curved road sections by combining the CTR (Constant Turn Rate) model with the UKF (Unscented Kalman Filter) algorithm through smartphone sensors [28] for high-precision estimation of vehicle motions. There is also curve trajectory prediction based on vehicle infrastructure co-operation [29], road marking and road edge mapping using an inverse visual detector model [30], and an inverse sensor model that takes into account the spatial error of the line feature detector is proposed. There is also lane change detection using visual odometer trajectory curves [31] with cumulative and curve-fitting methods for visual odometer curves. Mei has also developed an off-road detection method using a monocular camera [32], but it cannot parse the orientation information of unstructured twisting paths.

Since most methods are designed for scene understanding of structured paths and require prior training, it is necessary to propose a method capable of understanding unstructured twisting paths with higher efficiency and lower energy consumption through pure geometric modeling and without prior training and external devices, which can meet the requirements for efficient autonomous navigation of robots in the field.

3. Materials and Methods

Twisting paths in wild environments invariably encounter unexpected materials, as well as varying variations in light and color, resulting in fragmented and scattered edges. However, there are still traceable geometric clues. These fragmented configurations follow the grouping principle in terms of proximity and continuity, and can form global curves with high visual priority that help us understand these twisting paths in the wild.

3.1. Overview

Figure 2 shows an overview of the method. For monocular input, the graph is first augmented using a logarithmic transformation. The original graph is then segmented and binarized using the maximum interclass variance method. Based on the extracted lines, a connected domain algorithm is designed to present connected lines and extract the corresponding information of long curves. Finally, the surface composed of eligible long lines is reshaped and the twisting paths are understood.

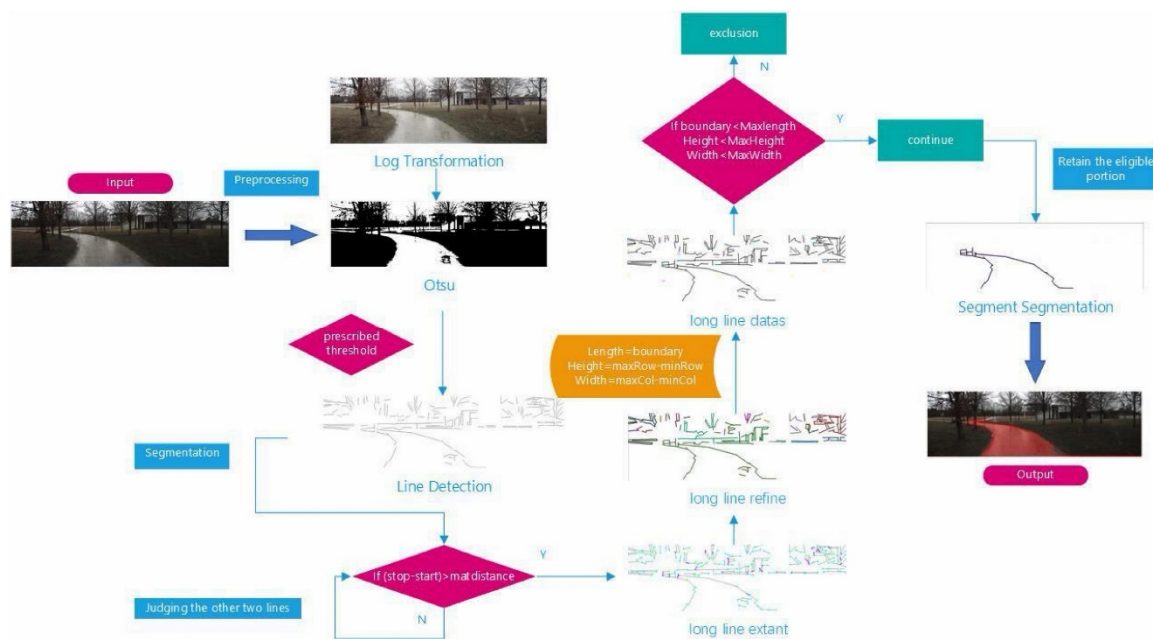


Figure 2. Our system architecture. The input is a monocular capture. Improved edge-based lines are extracted. Through extracting global precedence-based curves, it is possible to understand a twisting pathway including approximating its orientation in field environment.

3.2. Improved Edge-Based Lines

Unstructured curved roads have complex and diverse interferences. Edge is a more stable cognitive clue than color and texture in understanding a scene. Applying an improved edge map to realize high-level vision tasks is of vital importance. Using the road part and the interference part beyond the edge has a large gray difference, we can divide the image into target and background according to the gray value, find the inter-class variance each time, assign the largest to the variable, and then extract to get the improved scene lines. For an unexpected field environment, as shown in Figure 1, the plants on the roadside and the shadow light on the ground appear to be burdensome on extracting edge and lines; hence, it is necessary to design a component to improve line extraction.

We refer to our approach as Logarithmic Otsu to distinguish it from standard Otsu thresholding. The Otsu algorithm itself is not modified; the improvement lies in the preprocessing stage: a logarithmic transformation is applied to the input image, $I' = c \cdot \log(1 + I)$, where c is a scaling constant (set to $255/\log(256)$ for 8-bit images). This non-linear mapping expands low-intensity variations and compresses high-intensity ranges. This

preprocessing improves the inter-class variance separability under uneven illumination conditions, enabling the standard Otsu method to produce more stable and reliable binary segmentation in field environments.

Based on the principle of maximum interclass variance method, the improved Otsu was used to determine the threshold of binarised segmentation of the capture. Accordingly, the foreground and background images have the maximum interclass variance. It divides the image into two parts, background and foreground, according to the grey scale characteristics. Because the variance is a measure of the uniformity of the grey scale distribution, the larger the inter-class variance between the background and the foreground, the larger the difference between the two parts of the image. When part of the foreground is wrongly divided into the background or part of the background is wrongly divided into the foreground, the difference between the two parts becoming smaller. Therefore, the segmentation with the largest interclass variance implies the smallest probability of misclassification.

The Otsu threshold is determined by maximizing the inter-class variance, which minimizes misclassification probability. To reduce the sensitivity of threshold selection to varying illumination, we apply a logarithmic transformation prior to Otsu, expanding low-intensity details while compressing high-intensity ranges. The original image and the processed image are shown in Figure 3.



Figure 3. Top row: Input image. Bottom row: logarithmic transformed image.

Assuming that there exists a threshold TH to divide all the pixels into two classes C_1 (less than TH) and C_2 (greater than TH), the following is defined:

$$m_1 = \frac{\sum_{i=0}^k i \cdot p_i}{\sum_{i=0}^k p_i} \quad (1)$$

$$m_2 = \frac{\sum_{i=k+1}^{L-1} i \cdot p_i}{1 - \sum_{i=0}^k p_i} \quad (2)$$

where m_1 and m_2 are the mean values of the two classes of pixels C_1 and C_2 , respectively. Then the probability of a pixel being divided into the different classes can be defined as follows:

$$p_1 = \sum_{i=0}^k p_i \quad (3)$$

$$p_2 = 1 - p_1 \quad (4)$$

where p_1 and p_2 are the probabilities of a pixel being divided into class C_1 and C_2 , respectively. For the global mean of the image, it is defined as follows:

$$m_G = \sum_{i=0}^{L-1} i \cdot p_i \quad (5)$$

$$m = \sum_{i=0}^k i \cdot p_i \quad (6)$$

where m_G is the global mean of the image. m is the cumulative mean up to grey level k .

Accordingly, it is possible to find that the maximised grey level k is the threshold of the Otsu method, and the following can be defined:

$$\sigma^2 = \frac{(m_G \cdot p_1 - m)^2}{p_1(1 - p_1)} \quad (7)$$

where σ^2 represents the expression of inter-class variance.

After the logarithmic transformation, the range of grey level parameters that need to be adjusted by the inter-class variance algorithm will then become smaller. The final effect obtained through the inter-class variance algorithm is shown in Figure 4. Thus, those edges from some of the shadows and background plants can be eliminated, and more stable edges are extracted. Based on the refined edges, corresponding lines can be extracted [33] as follows:

$$\text{line}_i = [x_i^1, y_i^1, x_i^2, y_i^2] = [p_i^1, p_i^2], i \in [1, N] \quad (8)$$

where N is the number of lines, and lines are shown in Figure 4.

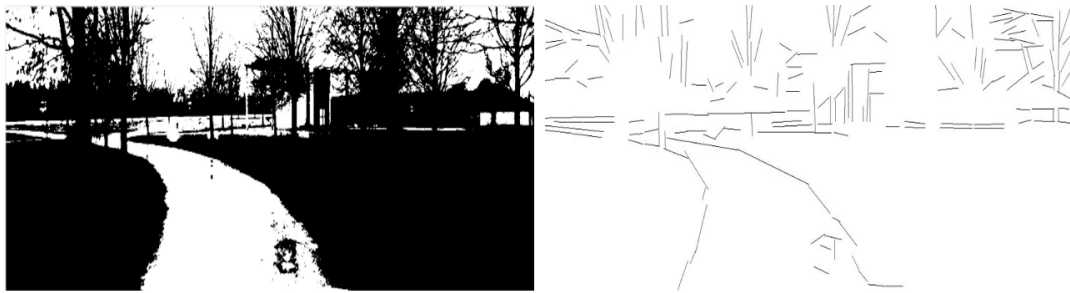


Figure 4. (Left): interclass variance algorithm for preprocessing segmentation of bends and background maps. (Right): extracted.

We acknowledge that global thresholding methods, including Otsu, can be challenged by non-uniform illumination and strong localized shadows common in field environments. However, the logarithmic transformation applied prior to thresholding compresses the dynamic range of high-intensity variations while expanding low-intensity details, which effectively reduces the intra-class variance caused by uneven illumination. This preprocessing step improves the separability of the path region from its surroundings under a variety of lighting conditions. In scenes where the inter-class variance fails to produce a clear bimodal distribution—such as extremely low-contrast or uniformly shadowed surfaces—the method may underperform; these failure cases are discussed in Section 5.3.

3.3. Curve Extraction Inspired by Global Precedence Effect

We translate the two core properties of GPE into algorithmic stages: global advantage guides the generation of curve candidates (Section 3.3.1), while global interference suppression motivates the refinement process (Section 3.3.2).

3.3.1. Curve Candidates

Inspired by the brain's cognitive mechanism global precedence effect, lines that follow grouping principles such as proximity and continuity tend to form a global curve, which has a higher visual priority in the human visual system. Therefore, we design a connected component algorithm to gain the ability to organize lines meaningfully and extract global curves quickly like humans.

A whole curve can be seen as a combination of a group of lines, so it is important to obtain the first two lines of a curve, which can be defined as follows:

$$\mu_{i,j} = \text{MIN}(d(p_i^a, p_j^b)), i, j \in [1, N], a, b \in 1, 2 \quad (9)$$

where $\mu_{i,j}$ is the proximity of each two lines line_i and line_j . d is the function to compute Euclidean distance between points. A smaller $\mu_{i,j}$ represents better proximity. Here, we rank the first two-lines combinations according to the $\mu_{i,j}$ value. It is obvious that the top-ranked combinations of better proximity are more likely to be a start of a curve.

Furthermore, a group of lines in a whole curve also have a good continuity, which can be defined as follows:

$$\zeta_{i,j} = \left| \arctan\left(\frac{y_i^2 - y_i^1}{x_i^2 - x_i^1}\right) - \arctan\left(\frac{y_j^2 - y_j^1}{x_j^2 - x_j^1}\right) \right| \quad (10)$$

where $i, j \in [1, N]$, and a smaller $\zeta_{i,j}$ represents a smoother continuity of line_i and line_j .

In this way, the first two lines of a curve can be extracted as follows:

$$c_{i,j} = [\text{line}_i, \text{line}_j], i, j \in [1, N] \quad (11)$$

$$\text{s.t. Min } \mu_{i,j}, \text{Min } \zeta_{i,j} \quad (12)$$

where $c_{i,j}$ represents a start which has the potential to be a whole curve. Then its two points that are further from the intersection can be defined as follows:

$$p_i^f = p_i^m + \frac{l_i}{2} \left(\frac{p_i^m - p_{i,j}^0}{d(p_i^m, p_{i,j}^0)} \right) \quad (13)$$

$$p_j^f = p_j^m + \frac{l_j}{2} \left(\frac{p_j^m - p_{i,j}^0}{d(p_j^m, p_{i,j}^0)} \right) \quad (14)$$

where p_i^f and p_j^f are two points further from $p_{i,j}^0$ (that is the intersection of two lines), respectively. p_i^m and p_j^m are midpoints; l_i and l_j are length of line_i and line_j , respectively.

In order to determine the next line (line_k) that can be combined into the $c_{i,j}$, the sequent grouping can be similarly defined as follows:

$$\mu_{i,j,k} = \text{MIN}(d(p_i^f, p_k^w), d(p_j^f, p_k^w)), i, j, k \in [1, N], w \in 1, 2 \quad (15)$$

$$\zeta_{i,j,k} = \left| \arctan\left(\frac{y_j^f - y_i^f}{x_j^f - x_i^f}\right) - \arctan\left(\frac{y_k^2 - y_k^1}{x_k^2 - x_k^1}\right) \right| \quad (16)$$

where $\mu_{i,j,k}$ and $\zeta_{i,j,k}$ represent the value of proximity and continuity for line_k and $c_{i,j}$, respectively. Then, a sequent group can be described as follows:

$$\text{curve} = [c_{i,j}, \text{line}_k], i, j, k \in [1, N] \quad (17)$$

$$\text{s.t. Min } \mu_{i,j}, \text{Min } \zeta_{i,j}, \text{Min } \mu_{i,j,k}, \text{Min } \zeta_{i,j,k} \quad (18)$$

where curve represents a combination with increased number of lines. Meanwhile, if line_k is selected to be a component, it is listed in a tabu table, which means it will not be selected again in later loop. Thus, it is possible to extract a curve. Finally, a number of curves are extracted by repeating in this way, which can be described as follows:

$$C = \{\text{curves}_s\}, s \in S \quad (19)$$

where curves_s is an extracted curve, and S represents the number of extracted curves.

The refinement score for a curve candidate C_k is defined as:

$$S(C_k) = w_L \cdot \tilde{L}_s(C_k) + w_W \cdot \tilde{W}_s(C_k) + w_H \cdot \tilde{H}_s(C_k) \quad (20)$$

where $\tilde{L}_s$, $\tilde{W}_s$, and $\tilde{H}_s$ are normalized length, width, and height of the curve's bounding box. The weights are empirically set to $w_L = 0.5$, $w_W = 0.3$, $w_H = 0.2$. Curves with $S(C_k) > \tau_s$ ($\tau_s = 0.6$ in all experiments) are retained. The top $S^* = 20$ curves are retained as refined global curves.

The thresholds used in our experiments are as follows. For proximity grouping ($\mu_{i,j}$), we set an upper bound of 50 pixels—lines with endpoint distances exceeding this value are not considered for grouping. For continuity ($\zeta_{i,j}$), we set a threshold of 30 degrees—lines with angular differences exceeding this value are rejected. These thresholds were determined empirically on a validation subset of the TAS500 dataset and were kept fixed across all datasets and experiments (TAS500, CurveLanes, FDWW, and Cityscapes) without per-dataset tuning. The tabu

table ensures that each line is assigned to at most one curve, preventing duplicate usage. The grouping process terminates when no remaining line satisfies both the proximity and continuity thresholds.

Accordingly, all the extracted curves are shown in left image in Figure 5.



Figure 5. (Left): Curves extraction inspired by global precedence effect. Through grouping lines that follow proximity and continuity, a number of global whole curves are shaped, which has a higher visual priority to be perceived. **(Right):** Refined global curves. A number of global whole curves with higher parameters can be refined.

3.3.2. Refinement

An extracted curve curves_s , which consists of a number of lines, can also be described as follows:

$$\text{curves}_s = \{\text{lines}_{s,t}, t \in T\} \quad (21)$$

where T represents the number of lines in the curves_s . $\text{lines}_{s,t}$ is a line in curves_s . Then, it is possible to acquire the corresponding parameters, e.g., length, width and height, which can be defined as follows:

$$L_s = \sum_{t=1}^T \Xi(\text{lines}_{s,t}), t \in T \quad (22)$$

$$W_s = |\text{MAX}(X_{\text{curves}_s}) - \text{MIN}(X_{\text{curves}_s})| \quad (23)$$

$$H_s = |\text{MAX}(Y_{\text{curves}_s}) - \text{MIN}(Y_{\text{curves}_s})| \quad (24)$$

where Ξ is function to calculate the length of a line. X_{curves_s} and Y_{curves_s} are x and y coordinates in a curve. L_s , W_s and H_s are the length, width and height of the curves_s , respectively. Accordingly, parameters for all curves can be shown in Figure 6.

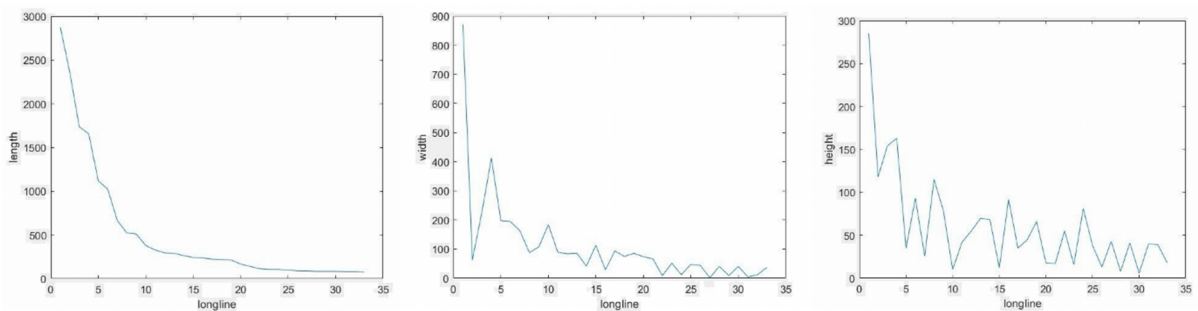


Figure 6. Parameters of global curves. **(Left):** length. **(Middle):** height. **(Right):** width. The horizontal axis shows the number of curve candidates, and the vertical axis shows their length, height and width. These parameters help us refine the global curves.

It is obvious that a global whole curve of higher parameters has higher visual priority to be perceived. Therefore, curve candidates can be refined by follows:

$$C^* = \{\text{curves}_{s^*}\}, s^* \in S^* \quad (25)$$

$$\text{s.t. MAX}(L_s); \text{MAX}(W_s); \text{MAX}(H_s); \quad (26)$$

where C^* are refined global curves, and S^* represents the number of refined global curves, with the result shown in right image in Figure 5.

3.4. Twisting Surfaces

Since a twisting pathway may conclude more than one refined curve, it is imperative to determine the boundaries of surfaces.

Regarding refined global curves, it is necessary to approximate their surfaces, which helps to understand such twisting pathways. Assuming that the refined curves are described as follows:

$$C^* = \{\text{lines}_{s^*}\} = [X_{C^*}, Y_{C^*}], s^* \in S^* \quad (27)$$

where X_{C^*} and Y_{C^*} are x and y coordinates in C^* , respectively.

Then the following can be found:

$$B_{\eta^*} = \{\Gamma(\text{line}_{\eta^*}, \text{lines}_{s^*})\} = \{X_{B_{\eta^*}}, Y_{B_{\eta^*}}\} \quad (28)$$

where $\eta^* \in [\text{MIN}(Y_{C^*}), \text{MAX}(Y_{C^*})]$. $\text{line}_{\eta^*} = [0, \eta^*, W, \eta^*]$ is an assumed virtual line in which W is the width of the image. lines_{s^*} is line in the refined curves C^* . For each horizontal scan line at $y = \eta^*$, we find the intersection points with the refined curves and take the leftmost and rightmost intersections as the estimated boundary points for that row, thereby reconstructing the complete path surface. Γ is a function to calculate the intersection of two lines, while B_{η^*} represents a set of the intersections corresponds to η^* . Then it is possible to determine the boundary points by formulating the follows:

$$B_{\eta^*}^1 = \{\text{MIN}(X_{B_{\eta^*}}), Y_{B_{\eta^*}}\} \quad (29)$$

$$B_{\eta^*}^2 = \{\text{MAX}(X_{B_{\eta^*}}), Y_{B_{\eta^*}}\} \quad (30)$$

where $B_{\eta^*}^1$ and $B_{\eta^*}^2$ are two approximated boundaries, such as blue and yellow curves shown in left image in Figure 7. Finally, twisting surfaces can be segmented with results shown in right image in Figure 7.

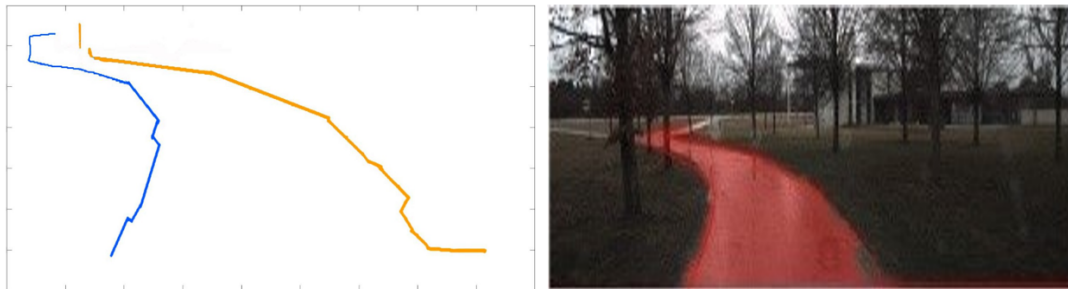


Figure 7. (Left): Approximated boundaries. Blue and yellow curves are estimated boundaries for $B_{\eta^*}^1$ and $B_{\eta^*}^2$, respectively. **(Right):** Twisting surfaces. Through approximating boundaries in the refined curves, twisting surfaces can be estimated.

3.5. Orientation

Modeling surface orientations and adjusting directions to accomplish navigation based on knowledge of curved path surfaces is critical for autonomous robots. Since it is difficult to obtain the exact depth of a path from a single image, traceable visual cues from refined global curves can help us understand the direction.

To compute the derivative f for orientation inference, we apply a Savitzky–Golay filter (window size = 11, polynomial order = 3) to the $X_{B_{\eta^*}}$ and $Y_{B_{\eta^*}}$ sequences before differentiation. This smoothing step reduces noise caused by irregular boundaries and outlier points. The derivative is then computed using central differences on the smoothed sequence. If the smoothed sequences of both $\text{MIN}(X_{B_{\eta^*}})$ and $\text{MAX}(X_{B_{\eta^*}})$ exhibit the same sign of derivative (both negative or both positive) over a contiguous region of at least 10 rows, we classify that region as a left or right turn, respectively. This prevents false detections from local fluctuations. We ranked the points in $B_{\eta^*}^1$ and $B_{\eta^*}^2$ according to the value $Y_{B_{\eta^*}}$, respectively, with corresponding value of $\text{MIN}(X_{B_{\eta^*}})$ (color in blue) and $\text{MAX}(X_{B_{\eta^*}})$ (color in yellow) shown in Figure 8.

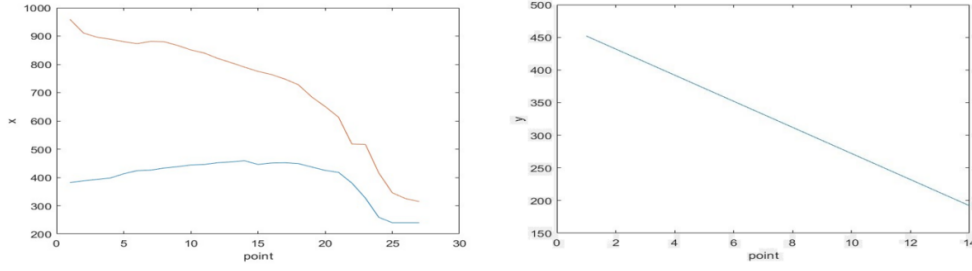


Figure 8. Orientation inference. (Left): $\text{MIN}(X_{B_{\eta^*}})$ (color in blue) and $\text{MAX}(X_{B_{\eta^*}})$ (color in yellow). (Right): $Y_{B_{\eta^*}}$. With decreasing $Y_{B_{\eta^*}}$, it is possible to infer the orientation through a similar variation of $\text{MIN}(X_{B_{\eta^*}})$ and $\text{MAX}(X_{B_{\eta^*}})$. It demonstrated that this pathway will turn left, and then turn right later.

Then, it is possible to infer the orientation through infer the corresponding X value as shown in Table 1.

Table 1. Orientation decision.

Condition 1	Condition 2	$\Lambda_{B_{\eta^*}}$
$f(\text{MIN}(X_{B_{\eta^*}})) < 0$	$f(\text{MAX}(X_{B_{\eta^*}})) < 0$	left
$f(\text{MIN}(X_{B_{\eta^*}})) > 0$	$f(\text{MAX}(X_{B_{\eta^*}})) > 0$	right

Here f represents the derivative, and $\Lambda_{B_{\eta^*}}$ represents left and right turn, respectively. When $\text{MIN}(X_{B_{\eta^*}})$ (color in blue) and $\text{MAX}(X_{B_{\eta^*}})$ (color in yellow) have a similar variation, which can be considered as:

$$f(\text{MIN}(X_{B_{\eta^*}})) < 0, f(\text{MAX}(X_{B_{\eta^*}})) < 0 \quad (31)$$

or

$$f(\text{MIN}(X_{B_{\eta^*}})) > 0, f(\text{MAX}(X_{B_{\eta^*}})) > 0, \quad (32)$$

the orientation of pathway can be approximated. As shown in Figure 8, since there are both $f(\text{MIN}(X_{B_{\eta^*}})) < 0$ and $f(\text{MAX}(X_{B_{\eta^*}})) < 0$, this pathway is going to turn left. Then, because there are both $f(\text{MIN}(X_{B_{\eta^*}})) > 0$ and $f(\text{MAX}(X_{B_{\eta^*}})) > 0$ later, the pathway will turn right later.

Currently, the orientation estimation is demonstrated through qualitative examples (Figure 8) that show the method's ability to infer turning directions (left/right) consistently across diverse scenes. A more systematic quantitative evaluation with ground-truth orientation annotations is a direction we plan to pursue in future work, which would provide numerical metrics such as mean angular error and classification accuracy.

4. Experimental Results

4.1. Experimental Architecture

In this study, we present a geometric model that utilizes a low-cost monocular camera to understand various twisting paths. Since the model does not require accurate depth data or any prior training, the experiments were performed on a computer with an Intel Core i5-11400 2.70 GHz CPU and 16 GB of RAM, with no additional high-performance GPU.

4.2. Datasets

Diverse datasets have been established to understand scenes, like widespread KITTI [34] and Cityscapes [35] datasets, both of which are structured city driving datasets; the outside unstructured environment (e.g., TAS500 [15], FDWW [36], RUGD [37] and CurveLanes [38]). They contain a variety of unstructured path and road scenes with various lighting conditions such as daytime and nighttime, as well as additional disturbances such as tunnels, rain and unexpected decorations. The presented approach aims to understand unstructured twisting paths in field environments; therefore, the TAS500 [15], FDWW [36] and Curve Lanes [38] datasets concluding a mass of various field ways were adopted to assess the effectiveness of the proposed methodology.

For evaluation, we selected images from each dataset based on the following criteria: (1) the presence of a clear, twisting or curved path; (2) diverse surface materials (gravel, dirt, grass, cobblestone); and (3) varied lighting conditions (day/night, shadow/sun). From TAS500 [15], we used the full test split (640 images). From CurveLanes

[38], we selected 500 images containing unstructured road segments. From FDWW [36], we selected 350 images from the field pathway category. All reported examples are representative of the full test set—the results in Figures 9–12 show typical outputs across the dataset.

4.3. Evaluation

We note that direct comparison between a binary path extraction method and multi-class semantic segmentation networks involves inherent task differences. The proposed method outputs a single-class (path vs. non-path) binary map, whereas deep learning baselines perform multi-class pixel-wise classification. To provide a more equitable basis for comparison, we report only the binary path segmentation performance for deep learning methods (collapsing all non-path classes into background). The comparison is intended to illustrate the relative performance of a lightweight, training-free geometric approach against resource-intensive learning-based methods. The reported results for competing methods were reproduced under the same test protocol using publicly available pre-trained weights.

4.3.1. Metrics

We used the mean Boundary Jaccard (mBJ) and mean Intersection over Union (mIoU) for evaluation metric. For mIoU, it calculates the correctly classified pixels, i.e., $\text{IoU} = \frac{TP}{TP+FP+FN}$, where TP , FP , and FN are the numbers of true positive, false positive, and false negative pixels, respectively. In the definition of BJ_c , the complete binary image of class c is denoted as S_{ps}^c in the prediction and S_{gt}^c in the ground truth. The pixels on the binary image boundary are denoted as B_{ps}^c in the prediction and B_{gt}^c in the ground truth. We calculate the distance from the ground truth binary image boundary B_{gt}^c to the predicted class image S_{ps}^c . The border threshold θ specifies the distance (in pixels) from the respective border (here $\theta = 8$ and $\theta = 12$). The total number of true positives is calculated as $TP_c = TP_{B_{gt}^c}^c + TP_{B_{ps}^c}^c$. Finally, the false negatives are calculated as $FN_c = |B_{gt}^c| - TP_{B_{ps}^c}^c$, and the false positives are calculated as $FP_c = |B_{ps}^c| - TP_{B_{gt}^c}^c$. The BJ measure for class c is based on the Jaccard index equation:

$$BJ_c = \frac{TP_c}{TP_c + FN_c + FP_c} \quad (33)$$

where TP , FP , and FN are the numbers of true positive, false positive, and false negative pixels, respectively. Similar to the IoU metric, the BJ metric is greater than zero if there is any overlap between the predicted segmentation map of the class and the ground truth map.

4.3.2. Results

We conducted experiments on the TAS500 dataset [15]. Regarding an unstructured way, as shown in Figure 9, the estimated twisting surfaces were compared to the ground truth, and the percentage of pixels that were correctly classified were measured, as shown in Table 2. For all datasets, the evaluation was performed using the same boundary threshold ($\theta = 12$ for our method and $\theta = 8$ for deep learning methods, consistent with the original TAS500 evaluation protocol). Input images were processed at their native resolutions without resizing for our method; deep learning results were obtained using the resolutions reported in their respective papers.

Table 2. Evaluation on twisting pathways on TAS500 [15]. Our method was compared against many algorithms including SQNet [39], ERFNet [40], ENet [41], LinkNet [39] and Fast-SCNN [42]. Note: Deep learning results are obtained using publicly available pre-trained weights and evaluated under the same protocol. GPU memory values are shown for reference only and do not represent a direct hardware comparison due to different memory allocation mechanisms (VRAM vs. RAM).

Method	Resolution	θ	Video Memory	mBJ	mIoU
SQNet	1024×512	8	24(GB)	63.46%	57.64%
ERFNet	1024×512	8	12(GB)	61.49%	59.42%
ENet	1024×512	8	11(GB)	64.49%	51.35%
LinkNet	1024×512	8	12(GB)	66.34%	57.04%
Fast-SCNN	2026×620	12	12(GB)	62.98%	53.00%
Our method	1500×459	12	4(GB)	76.61%	81.27%

The unstructured surfaces is mostly composed of gravel and is uneven with cluttered edges caused by unexpected vegetation, rainwater, and other factors. The deep learning-based methods require GPU devices of higher

performance. However, our method demonstrates competitive performance in understanding various field curved pathways while relying solely on CPU-based computation and requiring no pre-training, as shown in Figure 10.

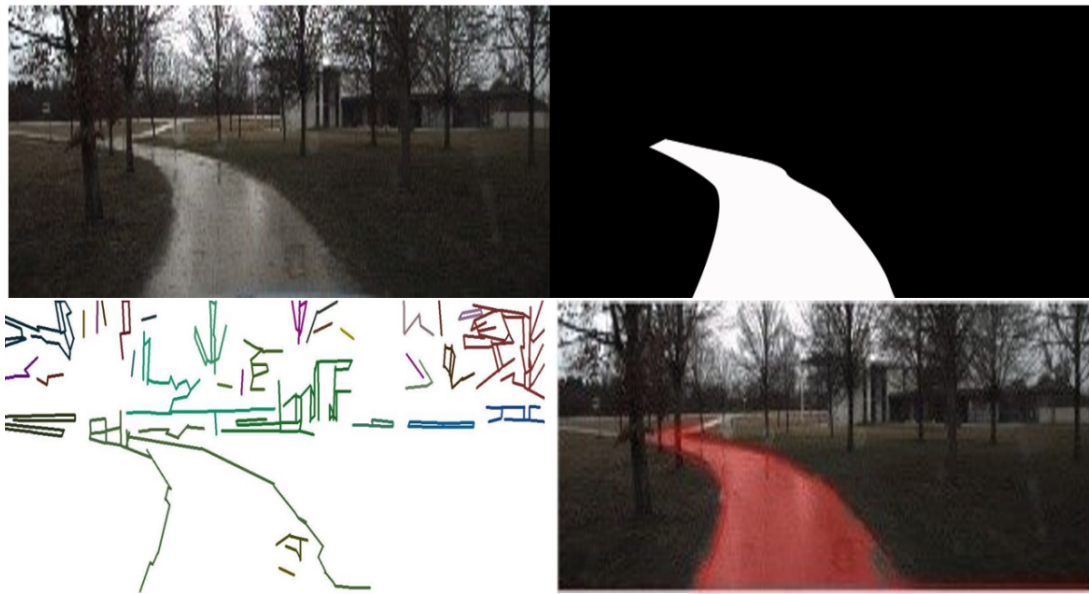


Figure 9. Understanding a field twisting pathway. (Top left): input pathway [15]. (Top right): ground truth. (Bottom left): Curves inspired by global precedence effect. (Bottom right): our understanding. Compared to the ground truth, unstructured twisting surfaces can be understood.



Figure 10. Results on the TAS500 [15]. First/third row: diverse pathways. Second/fourth row: our understanding. Our method can successfully understand the curved pathways in field environment.

Unstructured roads from the CurveLanes dataset [38] were also measured. Comparing the estimated pathway to the ground truth, the evaluated percentage is shown in Table 3. For various unstructured roads with changing illumination (e.g., day and night) and unexpected disturbance (e.g., fences and trees), as shown in Figure 11, the proposed method is able to understand both the straight roads and curved ways in such unstructured environments.

Table 3. Evaluation on unstructured ways on CurveLanes dataset [38].

Method	Resolution	θ	mBJ	mIoU
Our method	1244 × 700	12	72.69%	78.03%

In addition, the proposed algorithm was performed on the FDWW dataset (field pathway category) [36], with experimental results shown in Table 4.

Table 4. Evaluation on field twisting pathways on FDWW dataset [36].

Method	Resolution	θ	mBJ	mIoU
Our method	1244 × 700	12	73.39%	81.89%

For diverse off-road curved pathways, especially twisting pathways paved by unstructured materials such as cobblestones and rocks, our method can also successfully understand them without prior training, as shown in Figure 12.

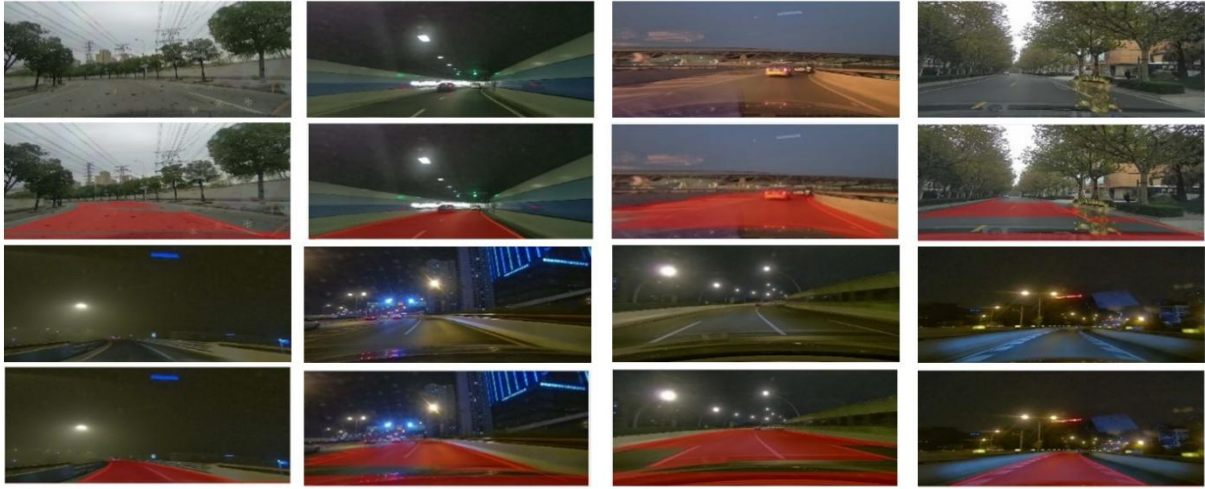


Figure 11. Experimental results on CurveLanes dataset [38]. First/third row: unstructured roads. Second/fourth row: our understanding. Obviously, both the straight roads and curved ways are understood.



Figure 12. Experimental results on FDWW dataset [36]. First/third row: twisting pathways. Second/fourth row: our understanding. Diverse twisting pathways, especially S-shaped pathways paved by unstructured cobblestones and rocks, can also be understood effectively.

4.4. Complexity

The method is based on geometric reasoning. For all detected lines (N), a single connection can be viewed as the meeting of two endpoints, i.e., the synthesis of two lines, and thus its corresponding time complexity is $O(N^2)$. For curve candidates, the pathway surface segmentation depends on the number of refined curves N_R , and the time complexity is $O(N^2 \log N) + O(N_R)$. Therefore, for the different number of extracted lines N , the complexity is $O(N^2 \log N)$. The method is based on geometric reasoning. For all detected lines (N), the curve grouping step follows a two-stage process. The curve candidate generation stage identifies initial line pairs with a complexity of $O(N^2)$. The subsequent curve connection stage iteratively extends each candidate curve by searching for the best-matching line among the remaining set, leading to an overall complexity of $O(N^2 \log N)$. In practice, for the datasets tested in this work, the number of detected lines N typically ranges from 50 to 400, depending on scene clutter and texture density. Within this range, the curve grouping step completes in a reasonable time on a standard CPU. For highly cluttered scenes where N exceeds 1000, the quadratic complexity may become a bottleneck; this is acknowledged as a limitation in Section 5.3.

4.5. Ablation Study

To evaluate the contribution of each key component in our proposed pipeline, we conducted a systematic ablation study on the TAS500 test set (640 images). Starting from the full method, we sequentially removed each component while keeping all other settings unchanged. Performance was measured using mean Intersection over Union (mIoU) and mean Boundary Jaccard (mBJ). The results are summarized in Table 5.

Table 5. Ablation study results on the TAS500 [15] dataset. Note: “w/o” stands for “without”.

Configuration	mIoU	mBJ
Full method	81.27%	76.61%
w/o logarithmic transformation	73.45%	68.32%
w/o proximity grouping	65.18%	61.07%
w/o continuity grouping	68.93%	64.15%
w/o curve refinement	76.42%	71.88%
w/o surface modeling	62.11%	58.44%

The ablation results confirm that each component contributes meaningfully to the overall performance. Surface modeling provides the largest contribution: removing it causes a drop of 19.16 percentage points in mIoU, indicating that simple boundary filling is insufficient for recovering complete path surfaces. Proximity grouping ranks second with a 16.09 percentage point drop, confirming that initial curve candidate quality is critical. Continuity grouping contributes 12.34 percentage points, demonstrating the importance of smooth curve extension for path integrity. The logarithmic transformation contributes 7.82 percentage points, with its effect being more pronounced under uneven illumination. Curve refinement provides the smallest but still positive contribution of 4.85 percentage points, helping to filter out spurious non-path curves. Overall, the complete pipeline integrates these components in a complementary manner, achieving the best performance on the TAS500 benchmark.

5. Discussion

5.1. Comparison

We distinguish three levels of capability for field robot perception: (1) Path segmentation: binary classification of image pixels into path vs. non-path regions. (2) Path understanding: beyond segmentation, this level recovers geometric properties of the path, including surface orientation, width variation, and curvature profile. (3) Navigation readiness: the integration of path understanding with motion planning, obstacle avoidance, and robot control to enable autonomous traversal. Our method operates at levels (1) and (2), providing both binary path maps and spatially varying orientation estimates. Level (3) integration is left for future work.

Comparative experiments are conducted on TAS500 dataset [15], and the results are shown in Figure 13.

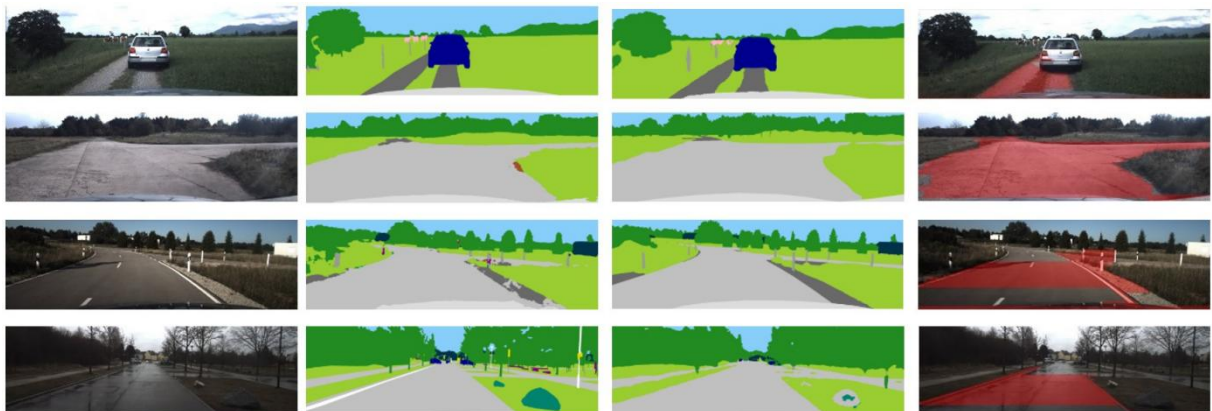


Figure 13. Experimental comparisons on TAS500 [15]. First column: off-road scenes. Second column: ground truth. Third column: Metzger’s method. Fourth column: our understanding. Our algorithm can understand field ways with various curved shape (e.g., second row).

To evaluate the generalization ability of our method on structured roads, we perform more comparisons on the Cityscape dataset [35], as shown in Figure 14. Romera implements semantic segmentation of scenes using a two-branch framework based on deep learning models, but only for structured urban road scenes [40].

Experimental results show that our approach can be successfully understood not only for unstructured twisting paths in wild environments, but also for structured urban road scenes.

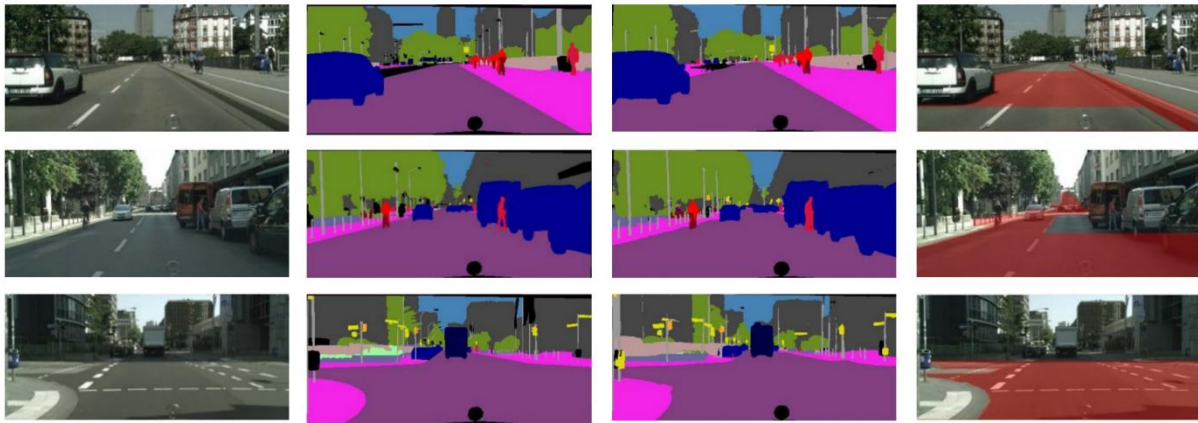


Figure 14. Experimental comparison on structured Cityscapes datasets [35]. First column: structured roads [35]. Second column: ground truth. Third column: Romera's method [35]. Fourth column: our understanding. Our method can also understand structured roads.

5.2. Real-World Applicability

In order to fully address the applicability of the method in the real world, we also discuss in detail how the algorithm performs under different weather conditions, different lighting, and different colors due to different times of the day, as shown in Figure 15 and Table 6. The experimental results show that the algorithm in this paper can successfully understand the twisting paths with high robustness even under changing light, color, and noise situations.

The robustness tests shown in Figure 15 and Table 6 were generated using the following image transformations on the original TAS500 test images: (1) Noise: Gaussian noise with $\sigma = 0.05$ (normalized intensity range) was added to simulate sensor noise; (2) Saturation: Saturation was increased by 50% in HSV color space; (3) Illumination: Global brightness was adjusted by gamma correction with $\gamma = 0.6$ (darkening) and $\gamma = 1.8$ (brightening), with the results averaged; (4) Color: Random color channel permutation and hue shifts of $\pm 30^\circ$ were applied to simulate white-balance variations. All transformations were performed using OpenCV functions. The evaluated images maintain the same resolution as the original dataset images, and the ground truth masks remain unchanged as the path geometry is unaffected by these photometric variations.

Table 6. Experimental results under different illumination, color, saturation and noise in TAS500 [15].

Difference	Original	Noise	Saturation	Illumination	Color
mIoU	81.27%	82.19%	81.97%	81.29%	80.68%

In order to verify that the algorithm can be applied to real driving scenarios, we transmitted videos captured by a monocular camera to the experimental computer. The videos (with a resolution of 1158×620) were taken at night with insufficient and varying illumination, including street lights, lights from surrounding houses, vegetation, and other vehicles. As shown in Figure 16, our method performs well. Especially at sharp curves (e.g., the third row), our algorithm is still effective in understanding the road surface, providing more time to assess braking distance and take appropriate measures.



Figure 15. Comparison under different weather conditions. First column: original scene. From second column to right column: changes in image noise, saturation, illumination and color respectively. Our algorithm successfully

understands twisting paths even under changing light, color, and noise in different weather and at different times of the day.

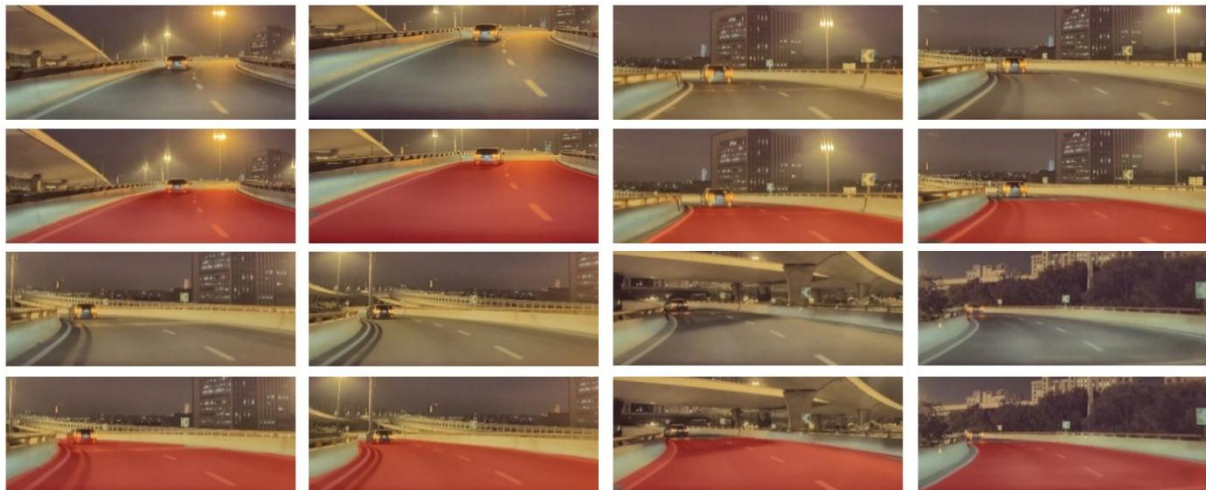


Figure 16. Following a curved road. First/third row: input frames. Second/fourth row: curved road understanding. Our method also works well on structured roads, especially for the sharp bend (e.g., third and fourth row).

The robustness of the proposed method was evaluated under various photometric variations as shown in Figure 15 and Table 6. The experimental results demonstrate that the algorithm maintains high mIoU (all above 80%) under different illumination, colour, saturation, and noise conditions. This robustness can be attributed to the edge-based geometric reasoning: edges are computed from local gradient information, which is inherently less sensitive to global intensity shifts and colour variations compared to pixel-intensity-based methods. However, as discussed in Section 5.3, extreme conditions such as near-complete shadow coverage or extremely low contrast may still cause performance degradation.

5.3. Limitations and Future Work

While the proposed method shows promising results, it has inherent limitations. (1) The method performs poorly in scenes with extremely low contrast or where the path is almost entirely featureless (e.g., uniform sand or snow with no tire tracks). In such cases, the initial line extraction step may yield insufficient features for reliable curve formation; (2) when strong shadows completely eliminate the luminance gradient at the path boundary, the binarization step may fail in those localized regions; (3) when dense vegetation or occlusions interrupt the path edges for extended regions, the continuity-based grouping may be interrupted; (4) when the scene contains multiple intersecting paths, the grouping may produce ambiguous curve assignments; and (5) when the path surface is visually indistinguishable from the background (e.g., dirt path on dirt ground), the assumption of edge-based separation breaks down. (6) Orientation estimation is currently presented qualitatively due to the lack of publicly available datasets with ground-truth directional annotations for unstructured field paths. Quantitative evaluation of orientation accuracy, including mean angular error and turning classification metrics, is planned for future work.

Future work will address these limitations. We plan to integrate temporal information from video sequences to improve curve extraction in low-feature areas, leveraging motion cues to disambiguate the path from static background clutter. Additionally, we aim to incorporate a lightweight, high-level semantic prior to guide geometric grouping, helping to distinguish between path boundaries and other dominant lines in the scene without resorting to full-blown deep learning. This would maintain the low-power advantage of our system while enhancing its robustness in highly complex environments.

6. Conclusions

This work describes a method for understanding unstructured twisting paths in a field environment using a monocular camera based on a cognitive mechanism of the brain—global precedence effect. Images are preprocessed using gray scale variation and maximum interclass variance method. Then, global precedence curves are extracted by analyzing the proximity and continuity of stimuli, which in turn shape the multiple unstructured surfaces of the twisting path. Based on the orientation of the global curves, the spatial orientation of the twisting paths can also be estimated, thus enabling the understanding of twisting paths in the field. The proposed GPE-inspired, geometry-only method enables training-free understanding of unstructured twisting paths using a

monocular camera. Experimental results demonstrate robust performance under varying illumination and color conditions, offering a lightweight and interpretable alternative for field robot perception. The method achieves a promising balance between algorithmic complexity and perceptual performance, with potential for low-power implementation on resource-constrained field robots.

Author Contributions

L.W.: Conceptualization, Methodology, Software, Writing—original draft preparation; L.C.: Data curation, Investigation, Validation, Writing—original draft preparation; Y.F.: Methodology, Software, Visualization, Writing—reviewing and editing; S.W.: Supervision, Project administration, Writing—reviewing and editing; H.W.: Conceptualization, Methodology, Supervision, Writing—reviewing and editing. All authors have read and agreed to the published version of the manuscript.

Funding

This work was supported by the NSFC Project (Project Nos. 62003212).

Institutional Review Board Statement

Not applicable.

Informed Consent Statement

Not applicable.

Data Availability Statement

Data will be made available on reasonable request.

Conflicts of Interest

The authors declare no conflict of interest.

Use of AI and AI-Assisted Technologies

No AI tools were utilized for this paper.

References

1. Amirchiabani, G.; Lovegrove, W.J. Role of eccentricity and size in the global precedence effect. *J. Exp. Psychol. Hum. Percept. Perform.* **1996**, *22*, 1434–1447.
2. Rezvani, Z.; Katanforoush, A.; Pouretmad, H. Global precedence changes by environment: A systematic review and meta-analysis on effect of perceptual field variables on global–local visual processing. *Atten. Percept. Psychophys.* **2020**, *82*, 2348–2359.
3. Wang, L.; Wei, H. Avoiding non-Manhattan obstacles based on projection of spatial corners in indoor environment. *IEEE/CAA J. Autom. Sin.* **2020**, *7*, 1190–1200.
4. Baheti, B.; Innani, S.; Gajre, S.S.; et al. Eff-UNet: A novel architecture for semantic segmentation in unstructured environment. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, Seattle, WA, USA, 14–19 June 2020; pp. 1473–1481.
5. Magerand, L.; Del Bue, A. Revisiting projective structure from motion: A robust and efficient incremental solution. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *42*, 430–443.
6. Bescós, B.; Cadena, C.; Neira, J. Empty cities: A dynamic-object-invariant space for visual SLAM. *IEEE Trans. Robot.* **2021**, *37*, 433–451.
7. Cavagna, A.; Melillo, S.; Parisi, L.; Ricci-Tersenghi, F. SpaRTA: Tracking across occlusions via partitioning of 3D clouds of points. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *43*, 1394–1403.
8. Mahmoud, A.; Ehab, L.; Reda, M.; et al. Real-time lane detection based on line segment detection. In Proceedings of the 2018 New Generation of CAS (NGCAS), Valletta, Malta, 20–23 November 2018; pp. 57–61.
9. Agrawal, S.; Deo, I.K.; Haldar, S.; et al. Off-road lane detection using superpixel clustering and RANSAC curve fitting. In Proceedings of the 2018 15th International Conference on Control, Automation, Robotics and Vision (ICARCV), Singapore, 18–21 November 2018; pp. 1942–1946.

10. Ye, C.; Zhao, H.; Ma, L.; et al. Robust lane extraction from MLS point clouds towards HD maps especially in curve road. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 1505–1518.
11. Merugu, K.; Adarsh, S. Multi lane detection, curve fitting and lane type classification. In Proceedings of the 2022 IEEE 19th India Council International Conference (INDICON), Kochi, India, 24–26 November 2022; pp. 1–6.
12. Wang, L.; Wei, H. Recognizing slanted deck scenes by non-Manhattan spatial right angle projection. *IEEE Intell. Syst.* **2022**, *37*, 75–85.
13. Viswanath, K.; Singh, K.; Jiang, P.; et al. OFFSEG: A semantic segmentation framework for off-road driving. In Proceedings of the 2021 IEEE 17th International Conference on Automation Science and Engineering (CASE), Lyon, France, 23–27 August 2021; pp. 354–359.
14. Ding, L.; Sherony, R.; Mehler, B.; Reimer, B. Perceptual evaluation of driving scene segmentation. In Proceedings of the 2021 IEEE Intelligent Vehicles Symposium (IV), Nagoya, Japan, 11–17 July 2021; pp. 1444–1450.
15. TAS500 Dataset url. Available online: <https://mucar3.de/icpr2020-tas500/> (accessed on 17 July 2024).
16. Shelhamer, E.; Long, J.; Darrell, T. Fully convolutional networks for semantic segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 640–651.
17. Badrinarayanan, V.; Kendall, A.; Cipolla, R. SegNet: A deep convolutional encoder–decoder architecture for image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 2481–2495.
18. Zhao, H.; Shi, J.; Qi, X.; et al. Pyramid scene parsing network. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 6230–6239.
19. Chen, L.; Papandreou, G.; Schroff, F.; et al. Rethinking atrous convolution for semantic image segmentation. *arXiv* **2017**, arXiv:1706.05587.
20. Blum, H.; Sarlin, P.E.; Nieto, J.; et al. Fishyscapes: A benchmark for safe semantic segmentation in autonomous driving. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW), Seoul, Republic of Korea, 27–28 October 2019; pp. 2403–2412.
21. Lambert, J.; Liu, Z.; Sener, O.; et al. MSeg: A composite dataset for multi-domain semantic segmentation. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 14–19 June 2020; pp. 2876–2885.
22. Jiang, P.; Osteen, P.; Wigness, M.; et al. RELIS-3D dataset: Data, benchmarks and analysis. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 30 May–5 June 2021; pp. 1110–1116.
23. Upadhyay, P.; Srivastava, S.; Pandey, S.S.; et al. Lane line detection using machine learning. In Proceedings of the 2022 3rd International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT), Ghaziabad, India, 11–12 November 2022; pp. 1–6.
24. Wang, H.; Wang, Y.; Zhao, X.; et al. Lane detection of curving road for structural highway with straight–curve model on vision. *IEEE Trans. Veh. Technol.* **2019**, *68*, 5321–5330.
25. Kasyap, V.L.V.S.K.B.; Macharla, A.; Sri, T.K.; et al. RoadSDNet: A robust algorithm for road boundary detection and segmentation using mixed networks and Hough transform. In Proceedings of the 2022 Algorithms, Computing and Mathematics Conference (ACM), Chennai, India, 29–30 August 2022; pp. 30–34.
26. Wang, L.; Wei, H. Bending path understanding based on angle projections in field environments. *J. Artif. Intell. Soft Comput. Res.* **2024**, *14*, 25–43.
27. Galimova, G.; Volkov, V.; Demyanov, D. An MPC-based path-following control algorithm of the road train. In Proceedings of the 2021 International Russian Automation Conference (RusAutoCon), Sochi, Russia, 5–11 September 2021; pp. 964–969.
28. Dai, M.; Feng, T.; Guo, L.; et al. Vehicle tracking and motion estimation on curve road segment by using smartphone sensors. In Proceedings of the 2019 5th International Conference on Transportation Information and Safety (ICTIS), Liverpool, UK, 14–17 July 2019; pp. 1117–1121.
29. Li, Z.; Liang, T.; Ren, Q. Curve trajectory prediction through vehicle infrastructure cooperation. In Proceedings of the 2022 IEEE 17th Conference on Industrial Electronics and Applications (ICIEA), Chengdu, China, 16–19 December 2022; pp. 1092–1097.
30. Shipitko, O.; Kabakov, A.; Grigoryev, A.; et al. Road markings and road edges mapping with inverse visual detector model. In Proceedings of the 2022 IEEE 95th Vehicular Technology Conference (VTC2022-Spring), Helsinki, Finland, 19–22 June 2022; pp. 1–6.
31. Salleh, D.N.S.D.A.; Seignez, E.; Kipli, K. Visual odometry based vehicle lane-changing detection. In Proceedings of the 2022 IEEE International Conference on Artificial Intelligence in Engineering and Technology (IICAET), Kota Kinabalu, Malaysia, 13–15 September 2022; pp. 1–6.
32. Mei, J. Dataset url. Available online: <http://www.poss.pku.edu.cn/index.html> (accessed on 21 July 2024).
33. Li, Y.; Tong, G.; Sun, A.; et al. Road extraction algorithm based on intrinsic image and vanishing point for unstructured road image. *Robot. Auton. Syst.* **2018**, *109*, 86–96.

34. Geiger, A.; Lenz, P.; Stiller, C.; et al. Vision meets robotics: The KITTI dataset. *Int. J. Robot. Res.* **2013**, 32, 1231–1237.
35. The Cityscapes Dataset url. Available online: <https://www.cityscapes-dataset.com/> (accessed on 2 August 2024).
36. Public FDWW Dataset url. Available online: <https://download.csdn.net/download/u011280710/74922587> (accessed on 4 August 2024).
37. Wigness, M.B.; Eum, S.; Rogers, J.G.; et al. A RUGD dataset for autonomous navigation and visual perception in unstructured outdoor environments. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Macau, China, 3–8 November 2019; pp. 5000–5007.
38. CurveLane-Dataset url. Available online: <https://github.com/SoulmateB/CurveLanes> (accessed on 5 August 2024).
39. Chaurasia, A.; Culurciello, E. LinkNet: Exploiting encoder representations for efficient semantic segmentation. In Proceedings of the 2017 IEEE Visual Communications and Image Processing (VCIP), St. Petersburg, FL, USA 10–13 December 2017; pp. 1–4.
40. Romera, E.; Álvarez, J.M.; Bergasa, L.M.; et al. ERFNet: Efficient residual factorized ConvNet for real-time semantic segmentation. *IEEE Trans. Intell. Transp. Syst.* **2018**, 19, 263–272.
41. Yu, C.; Wang, J.; Peng, C.; et al. BiSeNet: Bilateral segmentation network for real-time semantic segmentation. *arXiv* **2018**, arXiv:1808.00897.
42. Poudel, R.P.K.; Liwicki, S.; Cipolla, R. Fast-SCNN: Fast semantic segmentation network. In Proceedings of the 30th British Machine Vision Conference (BMVC), Cardiff, UK, September 9–12, 2019; BMVA Press: Durham, UK, 2019; p. 289.