Journal of *Marine Science and Engineering*

MDPI

Article

Research on the Vanishing Point Detection Method Based on an Improved Lightweight AlexNet Network for Narrow Waterway Scenarios

Guobing Xie [1,2], Binghua Shi [3,4,*], Yixin Su [1], Xinran Wu [1], Guoao Zhou [1] and Jiefeng Shi [1]

[1] School of Automation, Wuhan University of Technology, Wuhan 430070, China
[2] China Railway 11th Bureau Group Real Estate Development Co., Ltd., Wuhan 430050, China
[3] Hubei Key Laboratory of Digital Finance Innovation, Hubei University of Economics, Wuhan 430205, China
[4] School of Information Engineering, Hubei University of Economics, Wuhan 430205, China
* Correspondence: shibinghua1988@163.com

Abstract: When an unmanned surface vehicle (USV) navigates in narrow waterway scenarios, its ability to detect vanishing points accurately and quickly is highly important for safeguarding its navigation safety and realizing automated navigation. We propose a novel approach for detecting vanishing points based on an improved lightweight AlexNet. First, a similarity evaluation calculation method based on image texture features is proposed, by which some scenarios are selected from the filtered Google Street Road Dataset (GSRD). These filtered scenarios, together with the USV Inland Dataset (USVID), compose the training dataset, which is manually labeled according to a non-uniformly distributed grid level. Next, the classical AlexNet was adjusted and optimized by constructing sequential connections of four convolutional layers and four pooling layers and incorporating the Inception A and Inception C structures in the first two convolutional layers. During model training, we formulate vanishing point detection as a classification problem using an output layer with 225 discrete possible vanishing point locations. Finally, we compare and analyze the labeled vanishing point with the detected vanishing point. The experimental results show that the accuracy of our method and the state-of-the-art algorithmic vanishing point detector improves, indicating that our improved lightweight AlexNet can be applied in narrow waterway navigation scenarios and can provide a technical reference for autonomous navigation of USVs.

Keywords: unmanned surface vehicle; vanishing point detection; texture feature similarity; narrow waterway scenarios

check for updates

Citation: Xie, G.; Shi, B.; Su, Y.; Wu, X.; Zhou, G.; Shi, J. Research on the Vanishing Point Detection Method Based on an Improved Lightweight AlexNet Network for Narrow Waterway Scenarios. *J. Mar. Sci. Eng.* **2024**, *12*, 765. https://doi.org/10.3390/jmse12050765

Academic Editor: Weicheng Cui

Received: 1 April 2024
Revised: 21 April 2024
Accepted: 29 April 2024
Published: 30 April 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

In autonomous driving, using deep learning methods to detect vanishing points in road lane lines has become relatively mature, but relatively few studies have evaluated vanishing point detection in water environments. For broad areas such as the ocean, there is no single vanishing point; however, for narrow areas, vanishing points play an important role in the location detection and path design of unmanned surface vehicles (USVs) [1]. A USV is a kind of surface robot with integrated technology composed of multiple disciplines, including intelligent remote control, wireless communication, autonomous navigation and obstacle avoidance algorithms [2,3]. Comprehensive, accurate and effective perceptions of the navigation environment and positioning decisions are of high importance for the autonomous navigation of unmanned ships. Unlike overland and open water environments, narrow water environments are relatively complex, which brings new challenges to the autonomous positioning and navigation of USVs [4].

The ability of an autonomous USV to detect the vanishing point is crucial for determining the direction of heading and staying on the route. The vanishing point refers to the intersection point produced by the extension line of each parallel edge of the three-dimensional

graph [5]. Vanishing points can commonly be found in fields, railroads, streets, tunnels, forests, buildings and objects such as ladders (when viewed from the bottom up) [6]. There is a unique advantage of using the vanishing point as a descriptive feature for visual localization. The vanishing point is different from other characteristic points on the plane. It contains the direction information of the straight line. The analysis of vanishing points can reveal a large number of three-dimensional structures and direction information of the scene, which greatly simplifies the description of the scene [7]. Vanishing point detection and estimation have attracted great attention in various research fields, such as camera calibration, 3D reconstruction, pose estimation, depth estimation, target reconstruction and robot navigation [8].

By identifying the location of the vanishing point in the image, the system can adjust the posture of the USV automatically to ensure stable navigation in narrow waters. Traditional vanishing point detection methods include Gabor filters, Gaussian filters or finding the intersection of Hough transform lines based on a voting scheme with up to an $O(n2)$ time complexity [9]. To reduce or compensate for this defect, the area to be searched can be reduced by methods such as sea and sky division, which divides the navigation scene into navigable areas and non-navigable areas. The vanishing point must exist on navigable areas or the boundaries between the two areas; in contrast, some studies [10,11] use four Gabor orientation channels and selective voting to speed up the voting process. Deep learning approaches have dramatically improved the state-of-the-art results in many machine learning domains, such as computer vision, object recognition and image segmentation. The traditional vanishing point detection method needs to extract a large number of line segments and contours from the image, which is particularly time-consuming and has a high labor cost. In addition, the presence of unrelated lines and profiles in the scene can also affect detection accuracy.

Although these achievements can estimate the vanishing point with certain accuracy, the technical difficulties of this paper are reflected in the following aspects. First, unlike those of unmanned ground vehicle highways, the available data on narrow waters are limited. Second, unlike traditional lane detection methods, in traditional lane detection methods, the route is a winding curve, and the Hough transform or line segment detector (LSD) cannot be adopted for detection. Finally, there is diverse interference in narrow water environments, such as riparian vegetation and reflections on the surface. Generally, large deep networks such as convolutional neural networks (CNNs) require large datasets to train models [12]. However, at this stage, the fully annotated visual image datasets of narrow waterway navigation scenes are small, and their number is far below the standard for training deep learning models. This is inspired by the Google Street Road Dataset (GSRD), which includes 1 million images with labeled vanishing point locations. With the assistance of vanishing point detector transfer learning trained on this dataset, our annotated narrow-water USV Inland Dataset (USVID) can be fine-trained. Given the limitations of traditional approaches and the good performance of deep neural networks, we use deep learning models to learn a vanishing point detector automatically.

Traditional vanishing point detection methods use line detection to locate vanishing points. These techniques focus on human-made or indoor environments. However, when applied to more complex natural scenarios, traditional methods may suffer from ambiguous information generated by irregular line directions. The main contributions of this paper are as follows:

(1) A texture similarity based dataset expansion method is proposed, which can provide a solution for a small target dataset size.
(2) A new dataset of vanishing point detection data in narrow waters was established, including some samples of bad weather images, such as rainy and low-illumination conditions.
(3) A unified end-to-end trainable lightweight AlexNet network is proposed to solve the detection problem in complex narrow water environments.

We consider using data-driven deep learning to detect vanishing points in a narrow water environment. The remainder of this paper is organized as follows. A review of the related literature on vanishing point detection is presented in Section 2. Section 3 provides design considerations and preliminary details, including the criteria for complex maritime scenarios. The details of our proposed method are presented in Section 4, which contains a similarity calculation method and constructing the detection network model. In Section 5, the experimental results obtained with our proposed method are presented and compared with those of other relevant state-of-the-art approaches. The subsequent conclusions and future work prospects are given in Section 6.

2. Related Work

In this section, we investigate the relevant advances and research findings in vanishing point detection from two perspectives: traditional methods and deep learning-based methods.

2.1. Traditional Vanishing Point Detection Methods

Generally, traditional vanishing point detection algorithms can be divided into four categories. The first involves the use of spatial transformation technology, such as Gaussian sphere transformation and Hough transformation, to transform the information contained in an image to a limited space. Yang [13] proposed a fast vanishing point detection method based on row space features, which clusters similar vanishing points in the row space by analyzing the row space features and subsequently screens the vanishing points in the candidate lines. The general idea is to calculate all possible linear intersections and then solve them by least squares. Due to the calculation of all possible line intersections, the algorithm becomes complex. To overcome this shortcoming, Ebrahimpour [14] introduced a new procedure for finding the vanishing point based on visual information and K-means clustering. Unlike other solutions, the authors do not need to find the intersection of lines to extract the vanishing point. This approach has greatly reduced the complexity and processing time of the algorithm.

In the second category, the statistical estimation method is used to estimate the line parameter according to the edge feature point in the image, the vanishing point is calculated from these parameters or the edge feature point is used to construct functions and estimate the line and the vanishing point at the same time. Chen [15] designed a series of voting point selection strategies based on the background area to eliminate interference in the background area and improve the accuracy of the algorithm. To reduce the algorithm efficiency, he designed an angle priority voting function to treat the candidate point that receives the most votes in the voting space as the vanishing point.

The third category is based on the image texture. Rasmussen [16] first proposed using the texture directions to estimate the vanishing point. He used 72 directional sets of Gabor filters to accurately estimate the orientation of each pixel and voted for vanishing points by a global hard voting scheme. Ref. [17] proposed a contour texture detector to speed up the detection of pixels; this method retrieves pixels with reliable advantage vectors. In contrast to previous texture-based approaches that do not use response amplitude, this method considered the texture responses in road pixels.

More recently, the fourth category of optimal techniques has been introduced into vanishing point detection. Alan [12] used a recently proposed population-based method, a teaching–learning-based optimization algorithm (TLBO), to improve the efficiency of metaheuristic methods for identifying vanishing points. Ref. [18] proposed a vanishing point detection algorithm based on line-set optimization. The LSD algorithm is used to detect the lines, and the extracted line set is subsequently optimized to remove the invalid interference lines in the image, which improves the vanishing point detection accuracy. Ref. [19] used random forest and patchwise weighted soft voting to improve the efficiency of vanishing point detection. This approach is approximately 6 times faster in terms of detection speed than the generalized Laplacian of Gaussian filter-based method. Ref. [20] proposed an efficient and optimized voter selection strategy to identify vanishing points in

general road images. The main objective of this algorithm is to reduce the computational complexity and improve the efficiency of vanishing point detection algorithms for various types of road images.

2.2. CNN-Based Object Detection Methods

The target detection task is applied mainly to find and classify the target locations in the image. The traditional manual feature selection algorithms can be divided into two-stage target detection and one-stage target detection algorithms. The two-stage target detection algorithm first generates regions, which is called the region proposal, and then classifies samples through the convolutional neural network. Common two-stage target detection algorithms include R-CNN [21], Mask R-CNN [22], SPPNet [23], Fast R-CNN [24] and Faster R-CNN [25]. One-stage target detection architecture, directly positioning and classifying through DCNNs [26], and single-stage target detection can directly generate the coordinates of targets in one stage without the need to generate a candidate region process. Common one-stage target detection algorithms include YOLO [27–29] series, SSD [30], DSSD [31] and FSSD [32].

With the continuous development of deep learning, deep CNNs are becoming more widely used in the field of target detection and have been applied in many fields, such as agriculture, transportation and medicine. Compared with traditional manual feature-based methods, deep learning-based object detection methods can learn low-level and high-level image features and have better detection accuracy and generalizability. Borji [6] proposed a classification-based vanishing point detection algorithm and a dataset for vanishing point detection. The algorithm grids out the image and represents the image according to the CNN feature dimension. If the vanishing point falls within a grid, the classification category of the grid is positive. The obvious problem is that the vanishing point is rough and not refined enough. Ref. [33] proposed a unified end-to-end trainable multi-task network for co-processing lane and road mark detection and identification guided by vanishing points in severe weather conditions. Then, several versions of the proposed multi-task network are trained and evaluated, and the importance of each task is validated. The resulting approach, VPGNet, can detect and classify lanes and road markings and predict a vanishing point with a single forward pass.

Although these traditional methods for vanishing point detection are widely used in engineering practice, deep learning methods have issues with data adaptability. To reduce the impact of data changes on vanishing point detection, integrating image information into the network is desirable. Sheshkus [34] suggested a new neural network architecture for vanishing point detection in images that introduces fast Hough transforms into the network to enhance the network feature expression, making the vanishing point detection more robust. Liu [8] proposed a structure accurately predicted by the heatmap and vanishing point. The heatmap features are used to strengthen the subsequent feature map expression. For accurate prediction of the vanishing point, YOLO is used, which first predicts the feature grid location and subsequently predicts the offset based on the grid.

3. Problem Modeling and Data Preparation

In unmanned driving, unmanned ground vehicles and unmanned aerial vehicles are being developed increasingly. USVs are widely applied in water quality monitoring, river modeling, underwater detection, river rescue, garbage cleaning and other areas, but they still face many technical challenges. Unlike ground vehicles, there is still relatively limited public standard data in the field of USVs. The currently available image datasets for assisting in detecting vanishing points are mostly focused on urban and outdoor roads, and fewer images are specialized for narrow waterways. The USVID, which is the first inland dataset of real narrow waterway scenarios under multi-sensor and multi-weather conditions, was chosen. With respect to the different inland narrow river scene datasets, a total of 19,600 images were collected corresponding to five weather condition scenarios, i.e., cloudy, misty, overcast, rainy and sunny.

However, deep learning approaches typically require a much larger number of training examples. The similarity between road vanishing point detection and vanishing point detection in narrow water navigation scenarios is considered. For this purpose, we first use a similarity calculation method based on image texture features to calculate the similarity of the GSRD, which contains 1,053,425 images with resolution of 300 × 300 pixels from 24 routes across 21 countries. We filtered out the image scenes similar to the navigation scenes of complex narrow waterways and mixed them with the USV Inland dataset to solve the vanishing point detection problem. A flowchart for preparing the vanishing point detection dataset for narrow waters is shown in Figure 1.

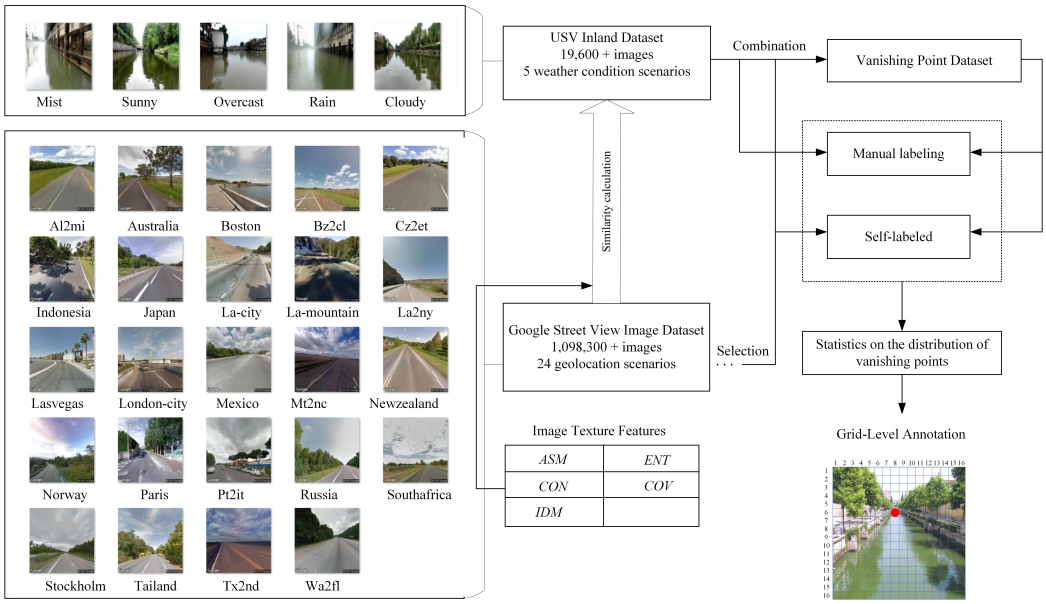


Figure 1. Flowchart of the dataset preparation for vanishing point detection.

In Figure 1, we first utilize the texture similarity between the two datasets to filter some scenes from the GSRD and mix them with the original USVID dataset for training. This approach can make up for the problem that the dataset size is not large enough for narrow waterway vanishing point detection. Then, all training images are resized to 300 × 300 pixels. To the best of our knowledge, the vanishing point locations in the GSRD are self-labeled. Moreover, the USVID requires manual labeling, and thus, we refer to the discretized vanishing point labels in a total of 15 × 15 = 225 labels in the reference [9]. For narrow waterway scenarios, we propose projecting pixel-level annotations to the grid-level mask, which means that pixel-level markers are not used. Since the pixel-level annotation is too small, we propose projecting pixel-level annotations to the grid-level mask to obtain grid-level labels. Considering that narrow waters are mostly complex and meandering structures, the possible locations of vanishing points are randomized, instead of being center-biased.

Since we equate the problem of detecting shadow vanishing points to a classification problem with 225 categories. Therefore, by counting the distribution of vanishing point locations in the GSRD and the USVID, it can be found that non-uniform grid-level labeling is more suitable for addressing the narrow waterway vanishing point detection problem.

4. Our Method

4.1. Similarity Calculation Method Based on Image Texture Features

Before performing accurate vanishing point detection, we must distinguish between the actual navigational environment captured by the CCTV system and whether the USV's navigational scenario is in open water or complex water. Open-water scenarios are usually defined as external waters with a wide field of view, making sea-level detection relatively

simple. However, complex water scenarios usually involve heavy traffic from inland rivers and harbors, and sea-level detection in complex scenarios is susceptible to a number of factors. Since the spatial relationship is considered to be a function of the distance between two pixels, texture features can be used to calculate the similarity of two different datasets.

The texture features were extracted from camera images using a gray-level co-occurrence matrix (GLCM). Haralick [35] proposed a variety of statistical feature measures to describe the texture features of different images, specifically including energy, entropy, contrast, inverse difference moment, correlation, variance, sum mean, sum variance, difference variance, difference mean, difference entropy, correlation information measure and the maximum correlation coefficient; however, there is a problem of duplication and redundancy among these feature measures. To solve this problem, five texture feature quantities, namely, the energy, entropy, contrast, inverse difference moment and correlation, that have a low correlation and are easy to compute are screened out. Before constructing the GLCM, we need to convert the original narrow waterway scenarios into grayscale images.

Assume that a narrow waterway scenario is converted into a grayscale image, which is described as follows:

$$I_{\text{gray}} = 0.299 \times R(x,y) + 0.587 \times G(x,y) + 0.114 \times B(x,y) \tag{1}$$

where $R(x,y)$, $G(x,y)$, $B(x,y)$ denotes the three channels of the original narrow waterway scenarios. (x_1, y_1) and (x_2, y_2) are two pixel points in image I with spacing d in direction θ; then, the GLCM for that scenario is computed as follows:

$$P(i,j,d,\theta) = \{(x_1,y_1),(x_2,y_2) \in M \times N \mid I(x_1,y_1) = i, I(x_2,y_2) = j\} \tag{2}$$

The energy parameter is also known as the angular second moment (ASM), which is one of the features of GLCM. The ASM is usually used to describe the uniformity of the gray-level distribution in an image. The ASM is often used to describe the uniformity of the grayscale distribution of narrow waterway scenarios. When the distribution of elements in the GLCM is more concentrated near the main diagonal, a smaller value indicates that the pixel grayscale distribution is more homogeneous and finely textured; conversely, it indicates that the pixel grayscale distribution is non-homogeneous or coarsely textured.

$$ASM = \sum_{i=0}^{N_g-1} \sum_{j=0}^{N_g-1} P(i,j,d,\theta)^2 \tag{3}$$

Entropy (ENT) is used to describe the amount of information contained in narrow waterway scenarios. If a scene image does not contain texture features, its GLCM is a zero matrix corresponding to an ENT value of zero; in contrast, the greater the amount of texture information contained in the scenario is, the greater the corresponding ENT value.

$$ENT = \sum_{i=0}^{N_g-1} \sum_{j=0}^{N_g-1} P(i,j,d,\theta)^2 \log_2 P(i,j,d,\theta) \tag{4}$$

Contrast (CON) is used to reflect the degree of image texture furrow depth and image clarity. In particular, in a narrow waterway scenario, the clearer the image texture is, the greater the variability in neighboring grayscale pairs and the greater the CON value; conversely, the CON value is smaller.

$$CON = \sum_{i=0}^{N_g-1} \sum_{j=0}^{N_g-1} (i-j)^2 P(i,j,d,\theta)^2 \tag{5}$$

The inverse difference moment (IDM) is a statistical feature quantity that reflects the degree of localized variation in the image texture. A large IDM indicates that there

is less variation between the textures of different regions in narrow waterway scenarios; conversely, a large IDM indicates that there is more variation between the textures of different regions.

$$IDM = \sum_{i=0}^{N_g-1} \sum_{j=0}^{N_g-1} P(i,j,d,\theta)/(1+(i-j)^2) \tag{6}$$

Correlation (COR) is used to measure the degree of similarity of the elements of the GLCM in the row or column direction. When the row or column similarity is high, the COR value is larger, indicating a lower scene image complexity; conversely, the complexity is greater.

$$COR = \sum_{i=0}^{N_g-1} \sum_{j=0}^{N_g-1} (i-\mu_1)(j-\mu_2)/\sigma_1\sigma_2 \tag{7}$$

where μ_1 and μ_2 denote the mean values of the elements along the row and column directions of the normalized GLCM, respectively, and σ_1 and σ_2 represent their mean square values.

By combining the above feature parameters, five different feature parameters can be extracted for narrow waterway scenario images and combined into a texture feature vector:

$$\boldsymbol{E} = [ASM, ENT, CON, IDM, COR] \tag{8}$$

After texture feature extraction, the images in the GSRD are subjected to similarity matching calculations with the images in the USVID. We use the Mahalanobis distance to measure the degree of similarity between two datasets; this metric is an effective measure of the similarity between two datasets proposed by Mahalanobis. Its calculation formula is as follows:

$$\boldsymbol{d} = \sqrt{(\boldsymbol{E}_0 - \boldsymbol{E}_1)^T S^{-1} (\boldsymbol{E}_0 - \boldsymbol{E}_1)} \tag{9}$$

where $\boldsymbol{E}_0$ and $\boldsymbol{E}_1$ denote the texture feature vectors of the USV Inland dataset and the Google Street Road dataset, respectively. S is the covariance matrix of $\boldsymbol{E}_0$ and $\boldsymbol{E}_1$.

4.2. Designing the Vanishing Point Detection Method

In practice, to reduce the problem of high complexity in classical AlexNets and reduce nonessential training costs, we propose an improved lightweight AlexNet model. Our proposed network model reduces the 5-layer convolution to 4-layer convolution (adding a pooling layer), which is designed as an alternating sequential connection of 4 convolutional layers and 4 pooling layers. Considering that the kernel of the first convolutional layer is 11 × 11, its computational parameters are large. For this reason, we incorporate Inception A and Inception C structures in the first two convolutional layers to decompose the convolution instead of the traditional convolution for feature extraction to reduce the network computational cost further. The details of the changes are as follows:

(1) The first convolution layer utilizes the Inception A structure instead, and its network structure parameters are shown in Table 1. As shown in Table 1, at 5 × 5, 3 × 3, and 1 × 1, three different convolution kernel scales were selected instead of using 11 × 11 convolution and multichannel feature extraction, while the pooling results of the input values were calculated, and then, the channels were fused in turn. In this way, the number of parameters and computations of the original 11 × 11 convolutional kernels can be reduced from 121 computational units to 46 computational units.

(2) The second layer of convolution utilizes the Inception C structure instead; its network structure is shown in Table 2. On the basis of keeping the three 1 × 1 pixel convolution kernels of the bottom layer unchanged, three groups of 1 × 7 and 7 × 1 convolution kernels are used for decomposition convolution, and the number of channels within the structure is increased by a value of 96 according to the number of feature maps of the new input layer. Table 3 shows that the number of parametric kernel calculations can be reduced from the original 25 computational units when using three groups of 1 × 7 kernel

and 7×1 convolution kernels instead of 5×5 convolution kernels for decomposition convolution, which are reduced to 21 computational units.

Table 1. The first layer of convolution network structure parameters.

Network Layer	Input Channels	Kernel	Padding	Output Channels
Dec. Layer 1	30	1×1	-	48
Dec. Layer 2	30	1×1	-	48
	48	5×5	2	64
Dec. Layer 3	30	1×1	-	48
	48	3×3	1	64
	64	3×3	1	80
Pooling Layer	30	1×1	-	16
Output Layer	48 + 64 + 80 + 16	-	-	208

Table 2. The second layer of convolution network structure parameters.

Network Layer	Input Channels	Kernel	Padding	Output Channels
Dec. Layer 1	224	1×1	-	96
Dec. Layer 2	224	1×1	-	64
	64	1×7	3	64
	64	7×1	3	96
Dec. Layer 3	224	1×1	-	64
	64	7×1	3	64
	64	1×7	3	64
	64	7×1	3	64
	64	1×7	3	96
Dec. Layer 4	224	1×1	-	96
Pooling Layer	384	1×1	-	384
Output Layer	96 + 96 + 96 + 96	-	-	384

The vanishing point detection network model for narrow waterway scenarios obtained through the above operations is shown in Figure 2. First, the network model takes AlexNet as the basic network structure and constructs 4 layers of convolutional layers and pooling layers connected in alternating order to realize lightweight processing. Then, the Inception A and Inception C structures in the Inception V3 module is fused in the first two convolutional layers to replace traditional convolution with decomposition convolution, which reduces the amount of model parameter computation and improves the accuracy of the model simultaneously. Finally, after feature extraction, the feature image is fed into the fully connected layer, and the Softmax classifier is used to calculate the probability that the input image belongs to category 225 to confirm the location of the narrow waterway vanishing point in the image scene.

Table 3. Statistical characterization of texture parameters.

Datasets	Scenarios	Frames	ASM		ENT		CON		IDM		COR	
			Mean.	Sd.	Mean.	Sd.	Mean.	Sd.	Mean.	Sd.	Mean.	Sd.
USVID	Cloudy	11,059	0.4562	0.0001	0.9054	0.0003	0.2943	0.0070	0.9640	0.0002	0.1249	0.0004
	Mist	4176	0.5549	0.0000	0.8242	0.0003	0.7734	0.0591	0.9304	0.0004	0.0965	0.0002
	Overcast	1160	0.3311	0.0000	0.9112	0.0001	0.3778	0.0033	0.9632	0.0000	0.1130	0.0001
	Rain	1643	0.4461	0.0000	0.9047	0.0000	0.2598	0.0007	0.9693	0.0000	0.1094	0.0001
	Sunny	1158	0.5096	0.0001	0.8571	0.0003	0.5394	0.0090	0.9473	0.0001	0.0925	0.0001
GSRD	Al2mi	52,627	0.3017	0.0001	0.9241	0.0004	0.3116	0.0054	0.9147	0.0030	0.2124	0.0078
	Australia	66,145	0.3329	0.0002	0.9083	0.0008	0.3733	0.0113	0.9012	0.0017	0.2076	0.0040
	Boston	38,215	0.3627	0.0001	0.8916	0.0006	0.4585	0.0107	0.8779	0.0015	0.1839	0.0032
	Bz2cl	37,426	0.3036	0.0001	0.9202	0.0005	0.3223	0.0063	0.8742	0.0021	0.2455	0.0045
	Cz2et	44,945	0.3283	0.0001	0.9215	0.0004	0.3304	0.0052	0.8920	0.0022	0.2326	0.0063
	Indonesia	48,135	0.3394	0.0004	0.8834	0.0019	0.5169	0.0463	0.8868	0.0015	0.1753	0.0042
	Japan	49,608	0.3080	0.0002	0.9172	0.0007	0.3536	0.0083	0.8804	0.0058	0.2555	0.0237
	La_city	33,752	0.3215	0.0001	0.9060	0.0004	0.4028	0.0065	0.8777	0.0022	0.2137	0.0046
	La_mountain	63,658	0.3317	0.0001	0.8765	0.0004	0.5190	0.0086	0.9055	0.0009	0.1495	0.0030
	La2ny	40,982	0.3718	0.0001	0.9221	0.0004	0.3193	0.0050	0.8938	0.0023	0.2306	0.0054
	Lasvegas	31,452	0.3943	0.0003	0.8685	0.0013	0.6532	0.0413	0.8615	0.0025	0.1775	0.0085
	London_city	46,281	0.3701	0.0002	0.8883	0.0009	0.4905	0.0216	0.8753	0.0016	0.1740	0.0030
	Mexico	46,536	0.2982	0.0001	0.9248	0.0005	0.3151	0.0062	0.8956	0.0042	0.2307	0.0106
	Mt2nc	59,199	0.3145	0.0087	0.9275	0.0004	0.3030	0.0059	0.9101	0.0017	0.2210	0.0039
	Newzealand	36,862	0.3147	0.0001	0.9175	0.0005	0.3348	0.0054	0.8996	0.0016	0.2187	0.0034
	Norway	84,451	0.3272	0.0002	0.8973	0.0011	0.4148	0.0125	0.8884	0.0087	0.2031	0.0297
	Paris	44,901	0.3998	0.0002	0.8779	0.0009	0.5377	0.0220	0.8905	0.0020	0.1606	0.0035
	Pt2it	55,884	0.3104	0.0001	0.9164	0.0005	0.3791	0.0095	0.8800	0.0034	0.2286	0.0102
	Russia	36,374	0.2903	0.0001	0.9168	0.0004	0.3295	0.0049	0.9161	0.0009	0.1994	0.0030
	Southafrica	45,728	0.3290	0.0001	0.9214	0.0004	0.3307	0.0051	0.8922	0.0022	0.2325	0.0064
	Stockholm	28,211	0.3685	0.0001	0.9092	0.0003	0.3661	0.0053	0.8915	0.0019	0.2097	0.0038
	Tailand	32,972	0.3278	0.0002	0.9075	0.0009	0.3805	0.0135	0.8840	0.0018	0.2213	0.0050
	Tx2nd	34,993	0.2978	0.0001	0.9369	0.0003	0.2694	0.0032	0.8775	0.0024	0.2643	0.0049
	W2fla	39,007	0.3261	0.0001	0.9229	0.0004	0.3144	0.0049	0.9014	0.0019	0.2219	0.0045

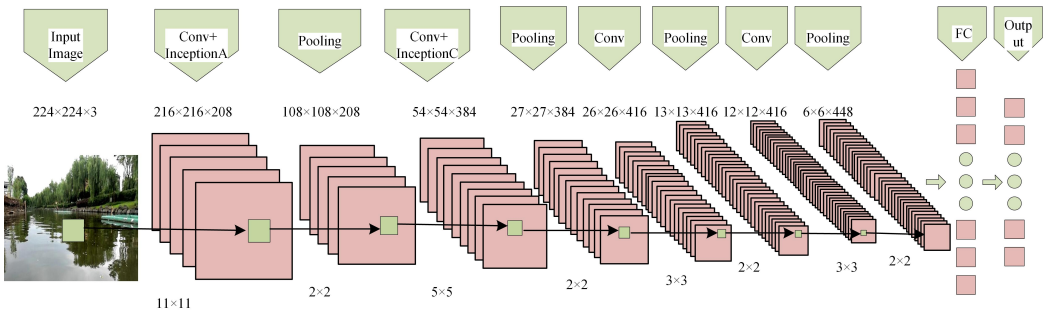


Figure 2. The improved lightweight AlexNet network for vanishing point detection.

5. Experiments and Results

The above image scenes are unified to 300 × 300 × 3 resolution for each frame of the image before classification, rating to ensure training efficiency and save computational resources. The compilation language used in this experiment was Python 3.7, the network model was built based on the PyTorch deep learning framework, the network model was loaded onto the GPU for the process, and the server graphics card model was RTX4080. The processor is a 13th Generation Intel(R) Core(TM) i7-13700K 3.40 GHz processor. The experiment consists of two parts: one is to validate the effectiveness of the scene similarity matching method based on image texture, and the other is to test the effectiveness of the network model for narrow waterway vanishing point detection based on the improved lightweight AlexNet.

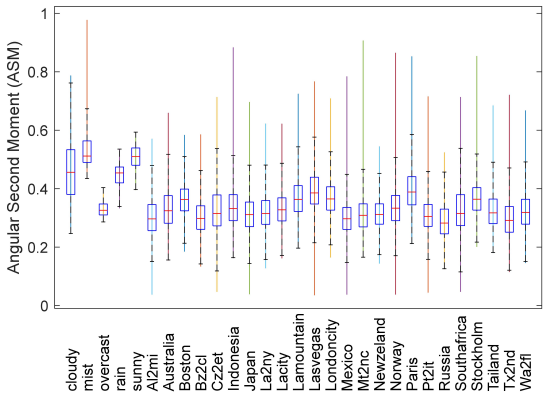
5.1. Dataset for the Narrow Waterway Scenarios

Since the USVID and the GSRD contain a total of 29 specific scenes, their backgrounds have strong similarities for detecting vanishing points in scene images. Therefore, we first calculated the mean and standard deviation of the texture features of these scenarios, as shown in Table 3.

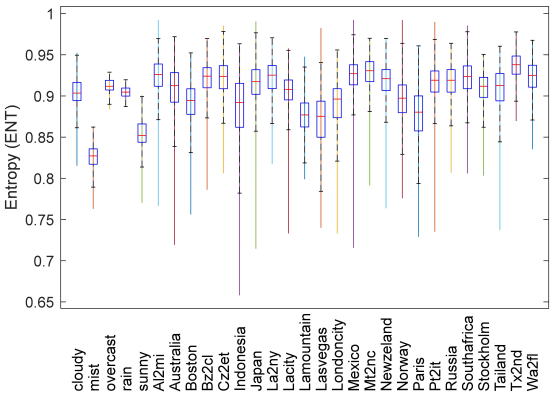
Figure 3 shows the boxplots of the textural features of the 29 scenes. Even though the distribution of each texture feature of the scenes within the USV Inland dataset has a large degree of discrete variability, the weights of different features need to be fully considered when performing the texture feature similarity matching operation. Based on these texture parameter values, the similarity between the USV Inland dataset and the Google Street Road dataset is calculated using the Mahalanobis distance, and the heatmap representation of the degree of similarity between the two datasets is shown in Figure 4.

During the experiment, the USVID is randomly divided into training and testing sets at a 7:3 ratio, while the GSRD is sequentially assigned to different training dataset groups according to the similarity calculation results in Figure 4, such as Group A to Group H, for a total of eight groups of comparison experiments. Group A consists of all USVID and represents the standard narrow waterway scenarios, while Group B adds the Stockholm scene from the GSRD to Group A. Group B is the most similar to the standard narrow waterway scenarios. Group C adds a new scenario, Paris, to Group B, which is the second most similar scenario to the standard narrow water scenario. Group D consists of the standard narrow waterway scenarios and the first three most similar scenes from the GSRD. Group E consists of the standard narrow waterway scenarios and three random scenes from the GSRD. Group F consists of the standard narrow waterway scenarios and the first four most similar scenes from the GSRD, while Group G consists of the standard narrow water scene and four random scenes from the GSRD. Group G consists of the standard narrow water scene and four random scenes from the GSRD. Group G is composed of the standard narrow waterway scenarios and the first four random scenes from the GSRD. Group E and Group G are the reference groups of Groups D and F to exclude the impact of dataset size on the detection results. Group H is composed of all 29 scenarios.

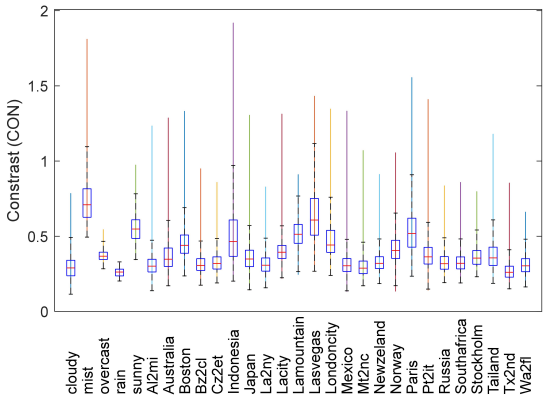
For different dataset groups, the classical AlexNet model was used for training, and the activation function was ReLU. The initial learning rate was set to 0.001, the learning rate was adjusted to 0.5 every 10 rounds of training, the "batch size" was set to 100, the "epoch" was set to 50, and the dropout layer was used in the fully connected layer to prevent overfitting. The corresponding parameter was set to 0.5 to prevent overfitting. Each set of experiments was repeated 10 times, and the mean training time and training accuracy are shown in Table 4.



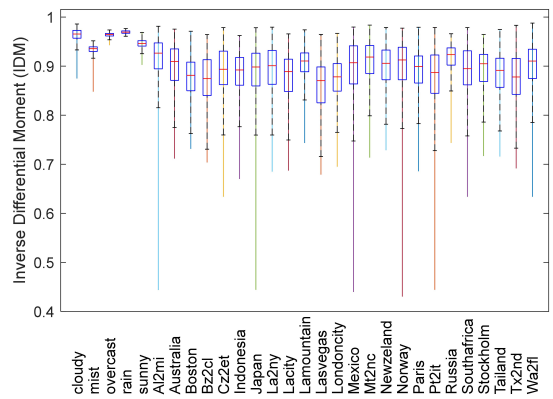
(**a**) ASM



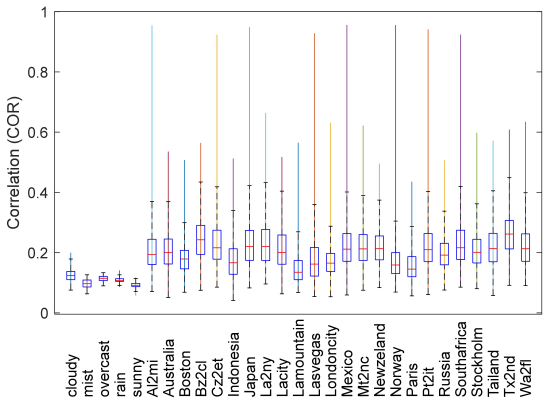
(**b**) ENT



(**c**) CON



(**d**) IDM



(**e**) COR

Figure 3. Boxplots of texture parameter distributions.

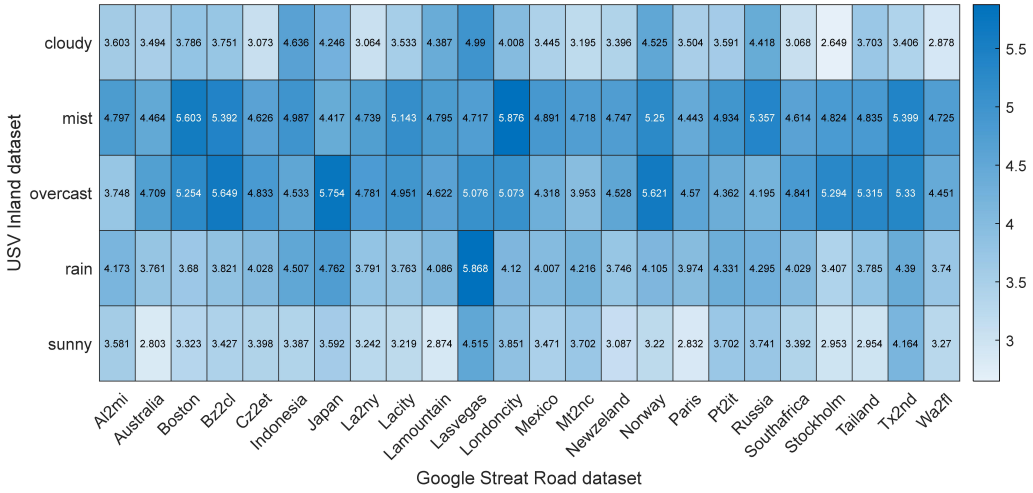


Figure 4. Heatmap representation of similarity.

Table 4. Comparison of vanishing point detection results for different groups.

Groups	Composition	Train		Test Accuracy
		Time	Accuracy	
Group A	Cloudy Mist + Overcast + Rain + Sunny	55min	86.42%	58.33%
Group B	Cloudy + Mist + Overcast + Rain + Sunny + Stockholm	60 min	87.31%	71.47%
Group C	Cloudy + Mist + Overcast + Rain + Sunny + Stockholm + Paris	77 min	89.05%	83.56%
Group D	Cloudy + Mist + Overcast + Rain + Sunny + Stockholm + Paris + Australia	80 min	89.17%	90.52%
Group E	Cloudy + Mist + Overcast + Rain + Sunny + Lasvegas + Londoncity + Tx2nd	81 min	88.73%	79.36%
Group F	Cloudy + Mist + Overcast + Rain + Sunny + Stockholm + Paris + Australia + Boston	90 min	89.34%	88.43%
Group G	Cloudy + Mist + Overcast + Rain + Sunny + Tailand + Mexico + Lacity + Norway	93 min	87.65%	86.73%
Group H	All 29 scenarios	4 h 25 min	90.41%	85.35%

As seen from the results, Group A represents a typical narrow waterway scenario, and the accuracy of the model can reach 86.42% in the training set; however, the accuracy is lower in the testing with only 58.33%, which can be attributed to two reasons; one is that the model itself is overfitted, and the other is that the labels in the training dataset and the testing set are not uniformly distributed. Two measures are taken to address these two points, respectively. One is to further optimize the model structure, such as the improved lightweight AlexNet model in the following context. The second measure is to enrich the sample labels of the dataset by the similarity of texture features, and the experimental results are shown in Group B–Group H. As the size of the narrow water shadow cancellation point detection dataset increases, the accuracy rate in the training stage increases gradually and slowly, and finally at approximately 90%. The accuracy at the testing time is affected not only by the size of the training dataset but also by the similarity of the texture features of the samples in the training dataset for narrow waters. This effect is manifested by the fact that the higher the similarity of the datasets used for training is, the greater the accuracy of the performance on the test dataset.

A comparison of the results for Group D vs. Group E and Groups F vs. Group G shows that the sizes of these two sets of data used in the comparison experiments are basically the same, but the accuracy of Group D and Group F is better than that of Group E and Group G. In addition, for the effect of the size of the dataset on the model, we compare the results of several groups of experiments and see that as the size of the dataset increases, the length of the training time increases, but the accuracy of the test set does not increase after it reaches 90% and falls back as the size of the dataset increases. This is because most of the scenes in the GSRD have large differences in texture features from those in the USVID, and the training of this dataset does not significantly improve the performance of the model. Therefore, the above results indicate that the fabrication of narrow water vanishing point detection dataset based on image texture similarity is effective, but we need to further improve the network model to shorten the training time and improve the detection accuracy due to the excessive accuracy and training time of the model.

5.2. Improved Lightweight AlexNet Network Performance Analysis

In this paper, to verify the effect of our network improvement, we designed a network specifically for the first two convolutional layers in the Inception V3 module to perform ablation experiments. The other conditions used were similar to those used for the design of five different networks: network R1, the classic AlexNet literature, network R2, and the original network based on the basis of the lightweight improvement; that is, the original five-layer convolution was simplified to a four-layer convolution and one pooling layer was added. Network R3, based on R2, incorporated the first convolutional layer into the Inception A structure. Network R4, based on R2, incorporated the second convolutional layer into the Inception C structure. Network R5, based on R2, incorporated both Inception A and Inception C structures. The recognition accuracies and loss values of these five networks are recorded in Figure 5 and Figure 6, respectively.

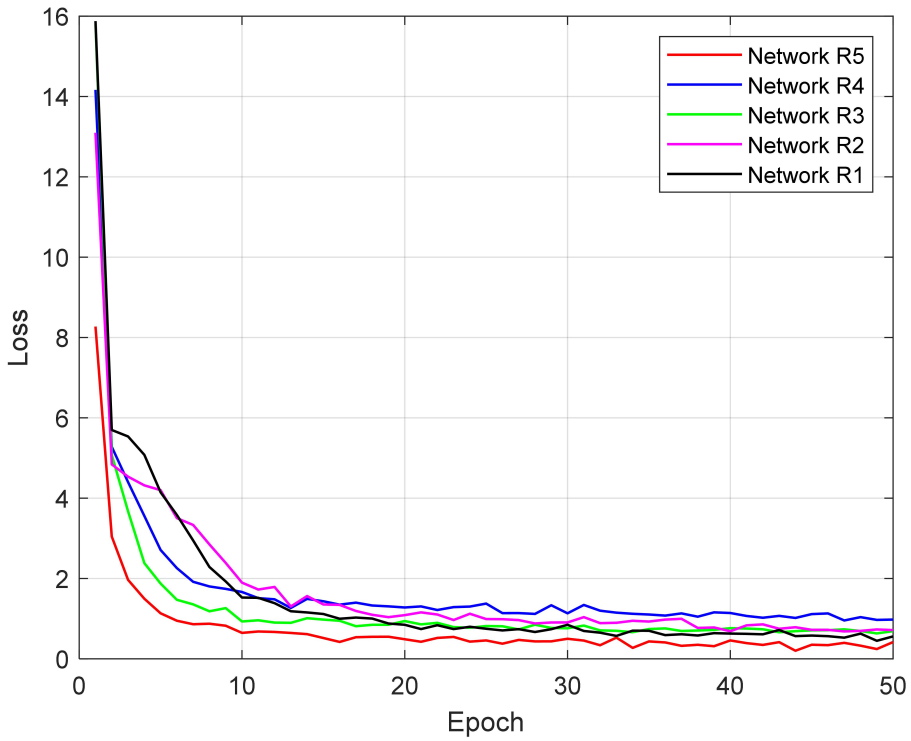


Figure 5. Comparison of loss values under ablation conditions.

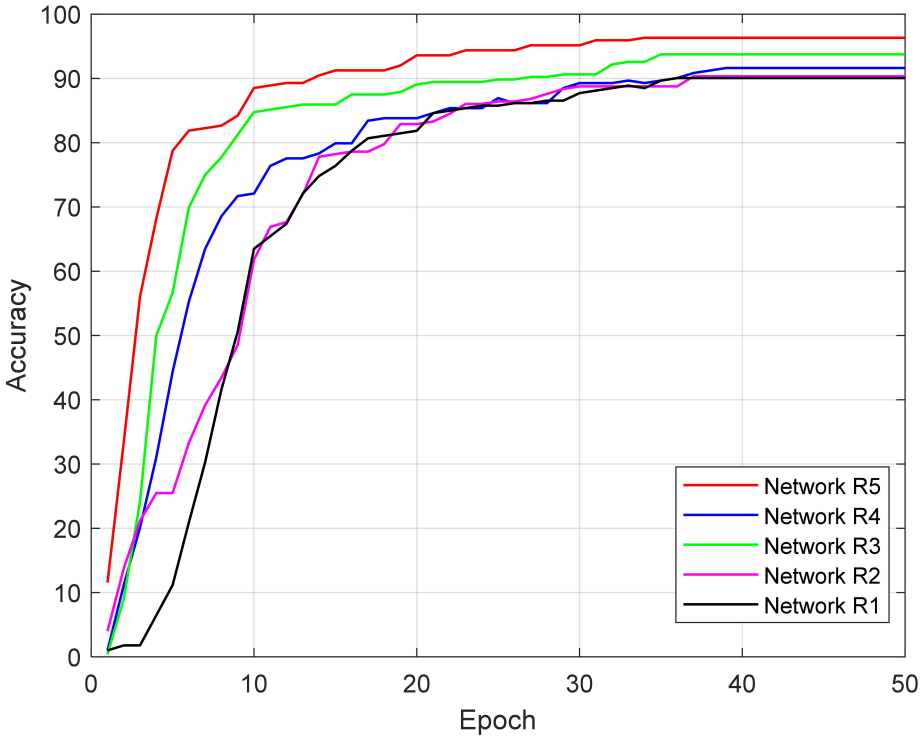


Figure 6. Comparison of detection accuracy under ablation conditions.

As seen from the comparison of the results, in the classical network R1, the loss function converges poorly in multiple iterations of training and needs to be stabilized at 90.62% accuracy after 30 epochs. For network R2, the effect after lightweight improvement is basically consistent with the convergence effect and recognition accuracy compared to network R2, but the number of model parameters is reduced from the original 60.7 MB to 27.5 MB, indicating that the lightweight operation of the classical model does not affect the performance of the model. Network R3 and Network R4 are incorporated into the Inception A and Inception C structures, respectively, and from the results, the accuracy of the accuracy network R3 is significantly better than that of network R4 and network R2, with the highest accuracy reaching 93.75%, an improvement of 3.45%. This phenomenon may occur because compared to Network R4, Network R3 focuses on optimizing the extraction of shallow features, which are relatively important for shadow cancellation point detection. Network R5, on the other hand, incorporates both Inception A and Inception C structures, and after incorporation, the number of model parameters increases compared to network R2, reaching 44.7 MB but still less than Network R1; additionally, this network achieves the best results, with the loss value being the first to start converging, and an accuracy of 96.33%. This accuracy also exceeds the 92% of that of the DeepVP network [9].

Moreover, we randomly selected 16 narrow waterway scenario images from the test set for testing, and the vanishing point detection results are shown in Figure 7. The green symbol in the figure indicates the location of the manually labeled vanishing points, and the red box is the detection result of our proposed network Model R5. The results, show that the detection results for all 16 pairs of images are consistent with the manually labeled results. In terms of the lightweight effect of the model, the average recognition time of our proposed R5 model is approximately 664 ms, which is affected by the fusion module, making it 11 ms more than the R2 model, but compared with the longest network Model R1, which takes 691 ms, the time consumed is reduced by 27 ms. Although our proposed R5 model increases the complexity compared with the lightweight model, its recognition effect is indeed improved.

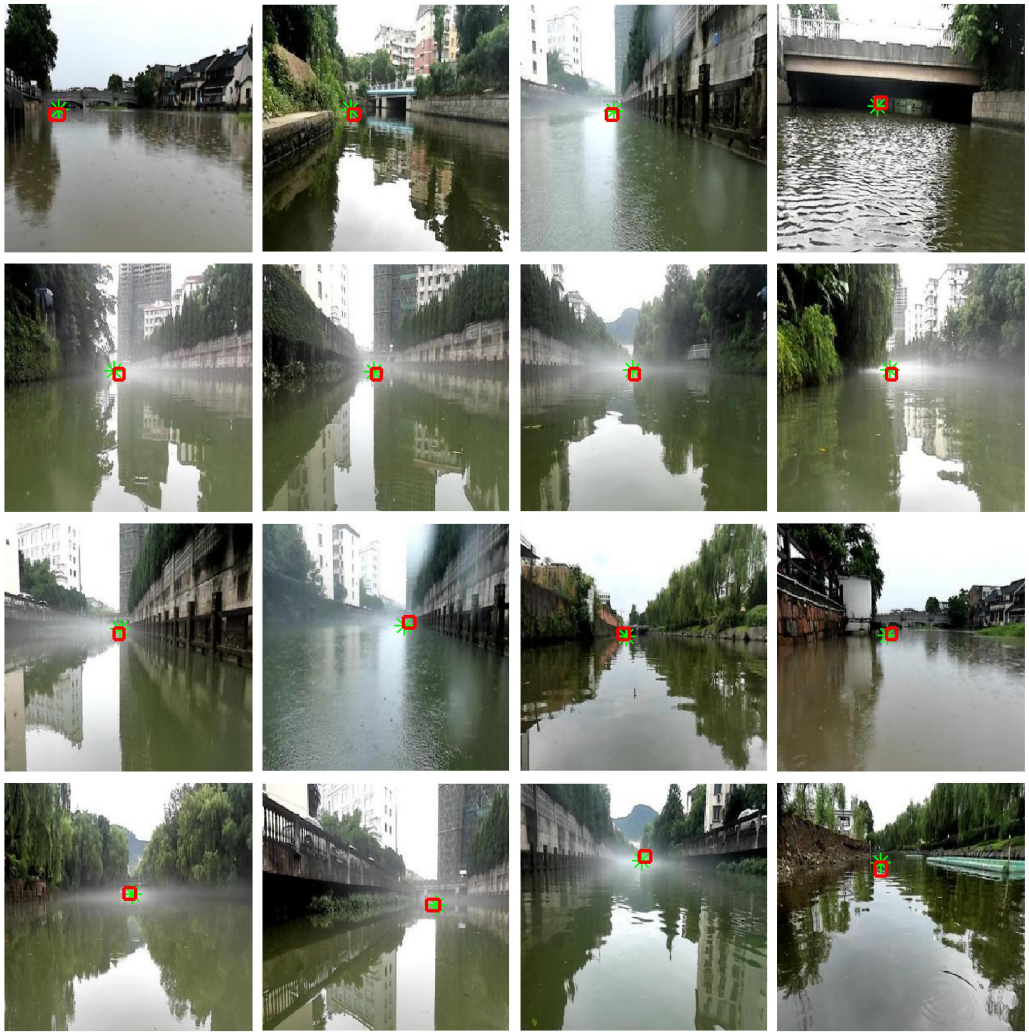


Figure 7. Detection of vanishing points in narrow waterway scenarios based on Network R5.

6. Conclusions

To solve the problem of vanishing point detection for narrow waterways in complex environments, we train a deep model end-to-end, take a narrow waterway scenario image as input, and output its vanishing point location. Unlike traditional methods that require a series of steps to predict the vanishing point location, our approach is a fast, feed-forward neural network evaluation that directly returns the vanishing point. We built a new vanishing point detection dataset to address the special characteristics of narrow waterway scenarios and the lack of publicly available datasets for the vanishing point detection problem. The vanishing point detection dataset contains five weather scenes, namely, cloudy, misty, overcast, rainy and sunny scenes, from the USVID. The narrow waterway scene images in this dataset are labeled with grid-level labels, and three scenes are filtered in the GSRD according to the similarity of the image texture features and the task scenarios. We also select three kinds of scenes in the GSRD according to the similarity between the image texture features and the task scenes and jointly train them together as expansion samples. The detection accuracy of the model used for joint training increases as the similarity of the scenario texture features increases, but the accuracy tends to stabilize as the size of the joint training dataset increases. The experiments showed that the data used for joint training can be controlled within a specific range to achieve better detection results. Moreover, we propose a unified end-to-end trainable vanishing point detection network, that incorporates the Inception A and Inception C structures in the first two convolutional layers after lightweighting the classical network. A comparison of our results shows that

our proposed improved lightweight Model R5 can improve the detection accuracy of vanishing points in narrow waterway scenarios. However, the detected fading points in this paper still belong to the grid-level, and there is still an error distance from the actual fading point locations. Next, we will integrate the multi-source information provided by ship kinematics equations, GPS direction angles and other ship-borne sensor positioning devices. We will also work on exploring ways to make the network more generalized, such as integrating dynamic obstacles and channel conditions to reduce dependence on specific scenarios, as well as investigating new target solving methods to further improve the localization accuracy of vanishing points.

Author Contributions: G.X.: Conceptualization, data curation, formal analysis, investigation, methodology, funding acquisition, validation, visualization, writing-original draft, writing—review and editing. B.S.: Conceptualization, resources, software, funding acquisition, writing—review and editing. Y.S.: Conceptualization, resources, software, funding acquisition, writing—review and editing. X.W.: Measurement, investigation, methodology, software. G.Z.: Investigation, methodology, software. J.S.: Conceptualization, formal analysis, funding acquisition, methodology, project administration, resources, software, supervision, writing—review and editing. All authors have read and agreed to the published version of the manuscript.

Funding: The study is supported by the Natural Science Foundation of China (No. 52201363), Natural Science Foundation of Hubei Province (No. 2023AFB562), Hubei Provincial Education Department Scientific Research Program Project (No. Q20222202, Q20212204), Ideological and Political Department Project of Hubei Province (No. 21Q210) and Hubei University of Economics Research and Cultivation Key Project (No. XJZD202106).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The datasets generated and analyzed during this study are available from the corresponding author upon request.

Acknowledgments: We thank the Wuhan University of Technology and Hubei University of Economics for their support.

Conflicts of Interest: Author Guobing Xie was employed in China Railway 11th Bureau Group Real Estate Development Co., Ltd. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Singh, Y.; Sharma, S.; Hatton, D.; Sutton, R. Optimal path planning of unmanned surface vehicles. *India J. -Geo-Mar. Sci.* **2018**, *47*, 1325–1334. Available online: http://hdl.handle.net/10026.1/10743 (accessed on 31 March 2024).
2. Xu, T. An intelligent navigation system for an unmanned surface vehicle. *Fault Detect. Superv. Saf. Tech. Process.* **2007**, *6*, 1503–1508. Available online: https://pearl.plymouth.ac.uk/handle/10026.1/604 (accessed on 31 March 2024).
3. Shi, B.; Wang, C.; Di, Y.; Guo, J.; Zhang, Z.; Long, Y. Research on a Horizon Line Detection Method for Unmanned Surface Vehicles in Complex Environments. *J. Mar. Sci. Eng.* **2023**, *11*, 1130. [CrossRef]
4. Shi, B.; Su, Y.; Wang, C.; Wan, L.; Luo, Y. Study on intelligent collision avoidance and recovery path planning system for the waterjet-propelled unmanned surface vehicle. *Ocean Eng.* **2019**, *182*, 489–498. [CrossRef]
5. Guo, K.; Cao, R.; Tian, Y.; Ji, B.; Dong, X.; Li, X. Pose and Focal Length Estimation Using Two Vanishing Points with Known Camera Position. *Sensors* **2023**, *23*, 3694. [CrossRef] [PubMed]
6. Borji, A. Vanishing point detection with convolutional neural networks. *arXiv* **2016**, arXiv:1609.00967.
7. Frank, M.O.; Ovchinnikov, K.D.; Ryzhov, V.A. Review of Russian and foreign experience of marine unmanned surface vehicles development. *Mar. Intellect. Technol.* **2022**, *57*, 2022. [CrossRef]
8. Liu, Y.B.; Zeng, M.; Meng, Q.H. Unstructured Road Vanishing Point Detection Using the Convolutional Neural Network and Heatmap Regression. *arXiv* **2020**, arXiv:2006.04691. [CrossRef]
9. Chang, C.K.; Zhao, J.; Itti, L. DeepVP: Deep Learning for Vanishing Point Detection on 1 Million Street View Images. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, QLD, Australia, 21–25 May 2018; pp. 4496–4503. [CrossRef]

10. Chang, C.K.; Siagian, C.; Itti, L. Mobile robot monocular vision navigation based on road region and boundary estimation. In Proceedings of the 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, Vilamoura-Algarve, Portugal, 7–12 October 2012; pp. 1043–1050. [CrossRef]
11. Moghadam, P.; Starzyk, J.A.; Wijesoma, W.S. Fast Vanishing-Point Detection in Unstructured Environments. *IEEE Trans. Image Process.* **2012**, *21*, 425–430. [CrossRef]
12. Lopez-Martinez, A.; Cuevas, F.J. Vanishing point detection using the teaching learning-based optimisation algorithm. *IET Image Process.* **2020**, *14*, 2487–2494. [CrossRef]
13. Yang, Q.; Ma, Y.; Li, L.; Gao, Y.; Tao, J.; Huang, Z.; Jiang, R. A fast vanishing point detection method based on row space features suitable for real driving scenarios. *Sci. Rep.* **2023**, *13*, 3088. [CrossRef]
14. Ebrahimpour, R.; Rasoolinezhad, R.; Hajiabolhasani, Z.; Ebrahimi, M. Vanishing point detection in corridors: Using Hough transform and K-means clustering. *IET Comput. Vis.* **2012**, *6*, 40–51. [CrossRef]
15. Chen, X.; Zhang, W.; Yang, L.; Zheng, X. Research on vanishing point detection of unstructured road scene combined with stereo vision. *J. Northwestern Polytech. Univ.* **2023**, *40*, 1431–1439. [CrossRef]
16. Rasmussen, C. RoadCompass: Following rural roads with vision + ladar using vanishing point tracking. *Auton. Robot.* **2008**, *25*, 205–229. [CrossRef]
17. Yang, G.; Wang, Y.; Yang, J.; Lu, Z. Fast and Robust Vanishing Point Detection Using Contourlet Texture Detector for Unstructured Road. *IEEE Access* **2019**, *7*, 139358–139367. [CrossRef]
18. An, X.; Zhao, T.; Jin, S.; Yang, C. Vanishing Point Detection based on Line Set Optimization. *J. Phys. Conf. Ser.* **2021**, *1748*, 032052. [CrossRef]
19. Riaz, I.; Fan, X.; Shin, H.; Rehman, Y. Vanishing point detection using random forest and patch-wise weighted soft voting. *IET Image Process.* **2016**, *10*, 900–907. [CrossRef]
20. Mistry, V.H.; Makwana, R. Optimized Voting Scheme for Efficient Vanishing Point Detection in General Road Images. *Int. J. Adv. Comput. Sci. Appl.* **2016**, *7*, 123–130. [CrossRef]
21. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013. [CrossRef]
22. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask R-CNN. *arXiv* **2018**, arXiv:1703.06870.
23. He, K.; Zhang, X.; Ren, S.; Sun, J. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 1904–1916. [CrossRef]
24. Girshick, R. Fast R-CNN. *arXiv* **2015**, arXiv:1504.08083.
25. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *arXiv* **2016**, arXiv:1506.01497.
26. Xu, L.; Ren, J.; Liu, C.; Jia, J. Deep convolutional neural network for image deconvolution. *Adv. Neural Inf. Process. Syst.* **2014**, *27*. Available online: https://dl.acm.org/doi/10.5555/2968826.2969026 (accessed on 31 March 2024).
27. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. *arXiv* **2016**, arXiv:1506.02640.
28. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv* **2018**, arXiv:1804.02767.
29. Zhu, X.; Lyu, S.; Wang, X.; Zhao, Q. TPH-YOLOv5: Improved YOLOv5 Based on Transformer Prediction Head for Object Detection on Drone-captured Scenarios. *arXiv* **2021**, arXiv:2108.11539.
30. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S. SSD: Single Shot MultiBox Detector. In Proceedings of the Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016. Available online: https://arxiv.org/abs/1512.02325 (accessed on 31 March 2024).
31. Fu, C.Y.; Liu, W.; Ranga, A.; Tyagi, A.; Berg, A.C. DSSD: Deconvolutional Single Shot Detector. *arXiv* **2017**, arXiv:1701.06659.
32. Li, Z.; Zhou, F. FSSD: Feature Fusion Single Shot Multibox Detector. *arXiv* **2018**, arXiv:1712.00960.
33. Lee, S.; Kim, J.; Yoon, J.S.; Shin, S.; Bailo, O.; Kim, N.; Lee, T.H.; Hong, H.S.; Han, S.H.; Kweon, I.S. VPGNet: Vanishing Point Guided Network for Lane and Road Marking Detection and Recognition. *arXiv* **2017**, arXiv:1710.06288.
34. Sheshkus, A.; Chirvonaya, A.; Matveev, D.; Nikolaev, D.; Arlazarov, V. Vanishing point detection with direct and transposed fast Hough transform inside the neural network. *arXiv* **2020**, arXiv:2002.01176.
35. Haralick, R.M.; Shanmugam, K.; Dinstein, I. Textural Features for Image Classification. *IEEE Trans. Syst. Man Cybern.* **1973**, *6*, 610–621. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.