Surround-view Free Space Boundary Detection with Polar Representation

Zidong Cao[1]
caozidong1996@stu.xjtu.edu.cn
Ang Li[1]
bennie.522@stu.xjtu.edu.cn
Zhiliang Xiong[2]
leslie.xiong@forward-innovation.com
Zejian Yuan[1]
yuan.ze.jian@xjtu.edu.cn

[1] Institute of Artificial Intelligence and Robotics
Xi'an Jiaotong University
Xi'an, China

[2] Shenzhen Forward Innovation Digital Technology Co. Ltd
Shenzhen, China

Abstract

Vision-based surround-view free space detection is crucial for automatic parking assist. In this task, precise boundary localization is the most concerned problem. In this paper, we have proposed to reframe the free space as polar representation for the free space boundary, and exploit a transformer framework to regress the representation end-to-end. To restrain the overall shape of the free space, we have introduced a Triangle-IoU loss function, enabling the network to consider the boundary as a whole. Furthermore, we have proposed a challenging newly-built surround-view dataset (SVB) with boundary annotations and supplied a new metric for boundary quality. Experiments on SVB dataset validate the effectiveness of our method, which outperforms existing free space detection methods and runs in real-time with a remarkable reduction in the computational cost. Additionally, our method shows excellent generalization ability to new parking scenes.

1 Introduction

Vision-based surround-view free space detection is one of the fundamental tasks in automatic parking assist (APA). The task is to identify the surround-view free space, i.e., a simply connected road area in a 360-degree view that is drivable for vehicles without collision, from image inputs. Although laser scanners are often used for this task because of its ability to capture accurate depth information, vision-based methods continue to draw interest due to their significant cost advantages. In this paper, we investigate to directly predict free space from surround-view images which are stitched from multiple fish-eye camera inputs (Fig. 1 (a)-(b)). Compared to applying detection on each single view separately and merging the results, detecting from surround-view images is more favorable in APA because of its advantage in inference time by providing a whole free space at one time.

In contrast to street scenes, free space detection in parking scenes raises the demand for boundary precision. That is because obstacles in parking scenes appear denser and crowd more closely around the vehicle. However, precisely locating such a boundary can be very

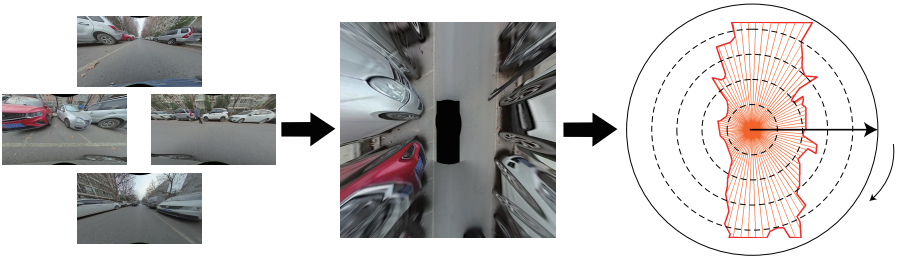

Figure 1: (a) Images captured by fish-eye cameras with four perspectives of front, back, left and right, (b) Surround-view image, (c) Polar representation for the free space boundary.

challenging, especially for surround-view images. Large-scale stretching distortions and harsh shadows are approached from fish-eye camera inputs and corresponding image mosaicing, respectively, making the boundary ambiguous. Besides, in parking scenes, obstacles often refer to vehicles and pedestrians, which vary greatly in size, direction and location.

Due to the high-precision requirement for the boundary localization, existing methods for free space detection are no longer applicable in our task. Most recent methods take advantage of fully convolutional networks (FCNs) [3], which treat free space detection as a binary segmentation problem [15]. Such a pixel-wise representation is over-complicated and regional, which only reflects the overall performance, paying little attention to the precision of the boundary localization. This pixel-wise representation will not only downplay the importance of the boundary localization, but also introduce an extra computational cost, because of subsequent up-sampling convolutional layers after high-dimension feature maps. Although several methods have been proposed to predict the boundary directly [10, 20, 25] with center classification and distance regression, the center heat maps and distance regressions are still in a per-pixel fashion. These methods are also faced with expensive computational costs.

In this paper, we have proposed to reframe free space as polar representation for free space boundary and exploit a transformer framework to regress the polar representation end-to-end. Based on the observation that surround-view free space is a simply connected area and can be easily recovered given its boundary, we argue to convert free space detection to a problem of boundary points prediction. To model these points effectively, we exploit polar representation, which has the inherent advantage in curve description. In Fig. 1, the driving vehicle (a black block) is located at the center of the image and always surrounded by the free space boundary. Therefore, by setting the image center as the origin, each boundary point is decided by a polar angle and a polar radius, where the order of the boundary points is determined naturally. Furthermore, with a specific sampling interval of polar angles, the boundary can be sampled to a group of points, and then compactly represented as a sequence of polar radiuses. Direct sequence regressions not only allow an explicit focus on the accuracy of the boundary localization, but also significantly reduce computational costs.

To capture non-local dependencies in prediction, we use a network built with transformers [4, 19] to integrate obstacle information and model global context. The network takes a surround-view image as input and regresses a sequence of polar radiuses end-to-end. Furthermore, during training, we propose a T-IoU (Triangle-IoU) loss to leverage the relationship of adjacent points and optimize the boundary as a whole.

Finally, to facilitate the development of free space boundary detection in APA, we have built a challenging large-scale dataset with abundant indoor and outdoor parking scenes. Current datasets, such as Cityscapes Dataset [6], KITTI road benchmark [7] and WoodScape Dataset [22], are mainly about street scenes. Tongji Parking-slot Dataset [23] includes parking scenes, but it lacks free space annotations. These facts make the existing datasets unsuitable for the study of surround-view free space detection. Our newly-built dataset complements the above insufficiencies by meticulously collecting parking scenes and manually annotating boundary annotations. In addition, we supply a metric to analyze the boundary precision quantitatively. Our method achieves remarkable performance on our dataset and shows excellent generalization ability to other datasets [23] without additional training.

The main contributions of this paper can be summarized as follows:

• We propose to reframe free space as polar representation for free space boundary, obviously simplifying the representation and enhancing attention on the boundary localization.

• The transformer framework is exploited to address the long sequence prediction. In addition, we propose a T-IoU loss to improve the correlation of adjacent predictions.

• We introduce a large-scale dataset in parking scenes with boundary annotations and an efficient metric to evaluate the boundary quality. Our method has good performance on our dataset and shows strong generalization ability to new parking scenes.

2 Related Work

Traditional algorithms for free space detection range from stixel world algorithm [2] to occupancy grids [1, 16]. In recent years, researchers have applied FCNs [17] in free space detection. The standard FCN model is composed of an encoder-to-decoder architecture, which extracts high-level feature representations in the encoder and up-samples the representations into a full-resolution segmentation in the decoder. Although FCNs achieve excellent accuracy in free space detection [14, 15], they are essentially designed for per-pixel classification. Moreover, some methods predict the vertical coordinate for each image column to represent the free space boundary directly [2, 8, 12, 21]. However, these methods are not applicable in surround-view images, because the surround-view boundary doesn't go along rows, resulting in the number and ordering relations of boundary points on each column ambiguous.

Polar coordinates system has inherent advantages in rotation and direction related problems [5, 25]. [25] utilizes a center point, one polar radius and two polar angles to represent the bounding box in remote sensing images. Similar solutions are proposed in the fields of object detection [18] and instance segmentation [10, 20], which can be summarized with two parallel tasks: center prediction and distance regression. However, they are more like a per-pixel prediction fashion, and need NMS (Non Maximum Suppression) as post-processing. In contrast, our method models the free space boundary in polar coordinates and predicts polar radiuses end-to-end, which abandons per-pixel fashion and has no need for post-processing.

Transformers are widely used in the computer vision field [4, 13, 24], showing extraordinary performance in capturing global context and modeling non-local dependencies. It is natural to exploit transformers to address large-scale distortions and global stabilities.

3 Method

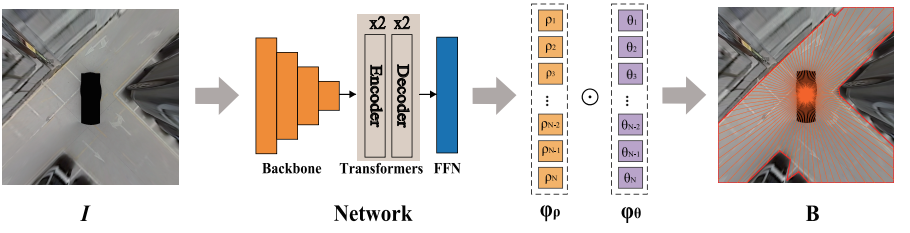

Figure 2: Pipeline. Given a surround-view image I as input, the network outputs a sequence φ_ρ of polar radiuses. $\odot$ represents to pair a polar radius with a polar angle to obtain a sampling point. The free space boundary B is generated by connecting the sampling points.

3.1 Polar Representation for Free Space Boundary

In order to model a surround-view free space boundary in polar coordinates, we first set the image center $c = (x_c, y_c)$ as the origin of the polar coordinate system, the horizontal-right direction as the positive direction of the polar axis, and the clockwise as the positive direction of the polar angle in radians. To form a closed curve, the polar angle is restricted to change in $[0, 2\pi)$. By sampling N boundary points with the same sampling interval of polar angles $\Delta\theta = \frac{2\pi}{N}$, the i-th sampling point can be represented by (ρ_i, θ_i), in which polar radius ρ_i is determined by the distance to c, and polar angle $\theta_i = i \cdot \Delta\theta$, where $i \in \{0, 1, 2, ..., N-1\}$.

So far, in polar coordinate system, a surround-view free space boundary φ can be sequentially represented as: $\varphi = \{(\rho_0, \theta_0), (\rho_1, \theta_1), ..., (\rho_{N-1}, \theta_{N-1})\}$. As θ_i is known, the elements to be predicted are only the polar radiuses, and φ can be further simplified as: $\varphi_\rho = \{\rho_0, \rho_1, ..., \rho_{N-1}\}$.

To evaluate the precision of free space boundary qualitatively, we transform Polar points to Cartesian points. A Cartesian point (x_i, y_i) can be obtained from (ρ_i, θ_i) as follows:

$$x_i = x_c + \rho_i \cdot cos(\theta_i),\ y_i = y_c + \rho_i \cdot sin(\theta_i). \tag{1}$$

Finally, we connect the adjacent points in order with straight lines to form a closed curve, representing the boundary of the free space. The area surrounded by the boundary is regarded as the free space. Our polar representation for the free space boundary is just a sequence with N elements, which is a significant simplification in parameters.

3.2 Boundary Detection Model

Figure. 2 illustrates the overall pipeline of our model. Given a surround-view image I as input, the network outputs a sequence φ_ρ end-to-end, containing a group of polar radiuses. By distributing a predetermined sequence φ_θ of polar angles, a group of sampling boundary points is obtained. After connecting these points with straight lines in turn, we can obtain the predicted boundary B and the corresponding free space.

The network consists of a backbone, transformers [13] and a feed-forward network (FFN) for sequence prediction. In the backbone, ResNet18 [9] is applied to extract low-resolution image feature. The transformer encoder and the transformer decoder are both stacked with two identical layers. Each encoder mainly consists of a multi-head self-attention module [13, 19] to model the image feature relation with paralleled attention operations to generate image embeddings. Each decoder has an extra multi-head cross-attention module [13, 19] after a

self-attention module to compute the interactions with image embeddings and sequence. Finally, FFN projects the output of transformers into φ_ρ by a 3-layer perceptron.

3.3 Triangle IoU loss (T-IoU loss)

In order to restrain the position of boundary points, a naive idea is to utilize l_1 loss to supervise the predicted sequence. However, l_1 loss is designed for the accuracy of a single point. The relationship between adjacent points and the overall shape of the boundary are ignored. This leads to under-smoothness and local ambiguity. Instead, the calculation of IoU treats free space as a whole and expects the boudnary to behave reasonably in both shape and size.

To bring in the advantage of IoU, we first uniformly sample with the sampling interval of polar angles $\frac{2\pi}{N}$ on the ground truth boundary to obtain a discrete sequence of N ground truth polar radiuses. We denote the ground truth sequence as $\hat{\varphi}_\rho$. Polar IoU [20] mentions that the area of free space can be represented with an infinite set of sectorial areas. However, in the limited number of sampling angles, sectorial areas that utilize only one radius can not fit to the complex shape. As the predicted points are connected with straight lines, the sampling free space is gathered by triangles with a shared center point. In this case, we replace sectorial areas in Polar IoU with triangle areas that utilize two adjacent polar radiuses. According to the computational formula of a triangle area $S_\triangle = \frac{1}{2}\sin\Delta\theta \cdot \rho_1\rho_2$, triangle IoU (T-IoU) can be calculated as follows:

$$\text{T-IoU} = \frac{\sum_{i=0}^{N-1} \frac{1}{2}\sin\frac{2\pi}{N} \cdot \rho_i^{min} \cdot \rho_{i+1}^{min}}{\sum_{i=0}^{N-1} \frac{1}{2}\sin\frac{2\pi}{N} \cdot \rho_i^{max} \cdot \rho_{i+1}^{max}}, \tag{2}$$

where $\rho_N = \rho_0$ and $\hat{\rho}_N = \hat{\rho}_0$. ρ_i^{max} indicates $\max(\rho_i, \hat{\rho}_i)$, and ρ_i^{min} indicates $\min(\rho_i, \hat{\rho}_i)$. Our T-IoU not only has more precise representation for free space in limited sampling angles than Polar IoU, but also adapts the rapid change of polar radiuses in slender obstacles better through learning adjacent relationship.

As the range of T-IoU is [0, 1] and the optimal is 1, T-IoU loss can be expressed as the binary cross-entropy of T-IoU. We omit the constant term $\frac{1}{2}\sin\frac{2\pi}{N}$, and T-IoU loss can be simplified as follows:

$$L_{\text{T-IoU}} = -\log(\text{T-IoU}) = \log\frac{\sum_{i=0}^{N-1} \rho_i^{max} \cdot \rho_{i+1}^{max}}{\sum_{i=0}^{N-1} \rho_i^{min} \cdot \rho_{i+1}^{min}}. \tag{3}$$

4 Experiments

4.1 SVB Dataset

To facilitate the development of surround-view free space detection in APA, the first issue we address is to create a large-scale dataset of surround-view images with boundary annotations, named *surround view boundary (SVB)*. Inputs of four fish-eye cameras with four perspectives are stitched into a single bird's-eye view image through image mosaicing technology with the vehicle at the center. The free space boundary is related to vehicles, pedestrians, roadblocks and steps, *etc*. Moreover, labeling the gaps between obstacles is mainly determined by whether the driving vehicle could pass through safely. As shown in Fig. 3, we first manually annotate the turning points where the free space boundary would change its trend, and then connect the turning points in order with straight lines to form the boundary.

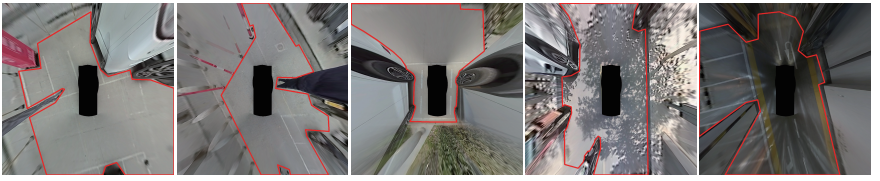

Figure 3: Examples in SVB Dataset. Red lines represent the free space boundary. Slender and large-scale obstacles are all included with various illuminations and parking scenes.

Statistically, SVB dataset is obtained from more than 200 videos. The image size is 1024×1024, corresponding to 18×18 (meters) in real world. In SVB, the percentages of outdoor parking scenes and indoor parking scenes are about 80% and 20%, respectively. To enhance the performance on slender objects such as pedestrians and roadblocks, we raise the percentage of images including slender objects to 21% by manual selection. Moreover, various illumination conditions and weather conditions are both included.

In all, SVB dataset contains 10632 surround-view images with boundary annotations of free space, which are split into a training set of 9569 images, and a test set of 1063 images.

4.2 Evaluation Metrics

The model outputs a sequence φ_ρ of N elements, which is aggregated to a predicted boundary B. Similarly, we sample on ground truth boundary $\hat{\mathrm{B}}$ with the sampling interval of polar angles $\frac{2\pi}{N}$ to get a ground truth sequence $\hat{\varphi}_\rho$ of N polar radiuses.

To attach importance to the boundary, we abandon metrics on the overall performance that are insensitive to the bounary precision, such as IoU and AP. To evaluate the accuracy and analyze the error distribution, mean absolute error (**MAE**) and percentage metric δ_i are reported. $\delta_i = \frac{1}{N}\sum_{k=0}^{N-1} \mathbb{1}(L_1(\varphi_\rho(k) - \hat{\varphi}_\rho(k)) \leq i)$ is the percentage of polar radiuses $\in \varphi_\rho$ whose L_1 error is within the range i (pixel), where $i \in \{1, 2, 5, 10\}$. $\mathbb{1}$ is the indicator function.

Moreover, to analyze the accuracy of the whole boundary, we propose a novel boundary average error (**BAE**). BAE is the mean euclidean distance between B and $\hat{\mathrm{B}}$. We first generate two edge maps (values of boundary points are '1', others are '0'), E and $\hat{\mathrm{E}}$, based on B and $\hat{\mathrm{B}}$, respectively. Next, we convert E with distance transform to D. Note that except BAE which uses $\hat{\mathrm{B}}$ as comparison, other metrics use $\hat{\varphi}_\rho$ as comparison. We define *obs* to be the pixels whose values are '1' in $\hat{\mathrm{E}}$ and compute the mean value of pixels $\in$ *obs* in D. BAE can be calculated as follows:

$$\mathrm{BAE} = \frac{1}{|obs|} \sum_{i \in obs} \mathrm{D(i)}. \tag{4}$$

4.3 Implementation Details

In experiments, ResNet18 [9] is used as our backbone. We resize all images to 512×512. The input images are augmented by rotating, color jittering, horizontal flipping and vertical flipping. We normalize polar radiuses by a factor of the diagonal length of the input image. Except in ablation studies, the number of polar radiuses is set to 360, which is determined by the performance on SVB dataset. We use the Adam algorithm [11] with the initial learning rate set to 10^{-4}, and a decay factor of 0.5 per 10 epochs. Batch size is 16 and training epochs

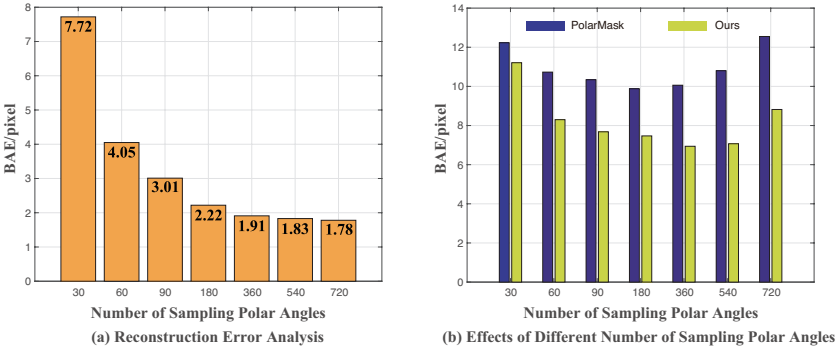

Figure 4: (a) Reconstruction error analysis. (b) The effect of different number of sampling polar angles. Our mehod predicts better especially in more sampling angles.

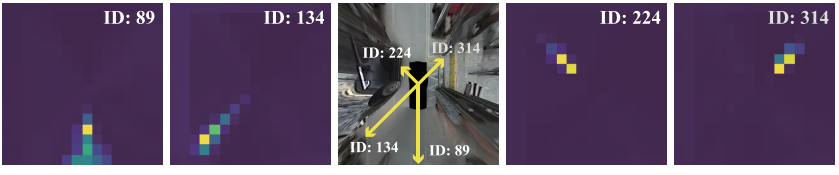

Figure 5: Attention maps in the cross-attention module of the decoder.

are set to 100. To best show the simplicity of our method, we also report FPS and MACs. We conduct all the experiments on a single Tesla M40 GPU.

4.4 Ablation Study

Number of sampling polar angles. Due to the discrete sampling, our polar representation for free space boundary loses part of details. Ground truth polar radiuses $\hat{\varphi}_\rho$ can be aggregated to a sampling ground truth boundary $\hat{\mathrm{B}}_\mathrm{s}$. To analyze the loss quantitatively, we compute the BAE score between the ground truth boundary $\hat{\mathrm{B}}$ and the sampling ground truth boundary $\hat{\mathrm{B}}_\mathrm{s}$, which is the ideal upper bound. From Fig. 4 (a) we observe that sampling more polar angles can achieve higher upper bound and recover more refined structures. When the number of sampling polar angles is more than 360, the improvement in upper bound is trivial.

In Fig. 4 (b), we experiment on the number of sampling polar angles from 30 to 720, and find that outputs with 360 sampling polar angles perform best. With increase in number of sampling polar angles, the model receives more precise structure information, and predicts boundaries with more details. However, when the number of sampling polar angles is too large, the boundary behaves excessively discrete and the capacity of the model is challenged to encode such a tedious sequence, resulting in less smoothness.

Attention in transformers. Fig. 5 shows the attention maps in the cross-attention module of the transformer decoder. We observe that for a specific polar radius in the sequence, image embeddings are effectively related to the possible areas near the free space boundary. Angle information is also implicitly learnt with positional embeddings [19].

Loss	BAE↓	MAE↓	δ_1↑	δ_2↑	δ_5↑	δ_{10}↑
$l1$	7.84	9.80	18.9	30.7	53.5	74.8
Polar IoU loss [20]	7.36	9.15	18.6	32.4	56.0	77.3
T-IoU loss	**6.94**	**8.96**	**23.0**	**34.9**	**56.6**	**77.4**

Table 1: Comparison with different loss functions. Our T-IoU loss has better performance.

Methods	BAE↓	MAE↓	δ_1↑	δ_2↑	δ_5↑	GMACs	FPS
SegNet-Basic [3]	12.20	21.09	20.4	24.9	39.7	29.9	79
PolarMask [20]	9.88	10.43	15.6	28.1	51.5	28.1	67
Ours	**6.94**	**8.96**	**23.0**	**34.9**	**56.6**	**9.7**	78

Table 2: Comparison with different methods on SVB testing set.

T-IoU loss. We study how T-IoU loss affects the learning. According to Tab. 1, T-IoU loss achieves 6.94 pixels in BAE. In contrast, $l1$ loss achieves 7.84 BAE, a margin of 0.90 pixels. The margin shows that training with the overall shape of the boundary is more effective than solely focusing on isolate points. In addition, our T-IoU loss outperforms Polar IoU loss by 0.42 pixels in BAE. We ascribe the improvement to the better representations for the boundary and the consideration of the relationship between adjacent points.

Failure cases. Failure cases of predictions are shown in Fig. 6 due to occlusions. Our polar representation is unable to cover blue areas through rays emitted from the image center.

4.5 Comparisons with State-of-the-Art Methods

Comparison on computatioal costs. In Tab. 2, our method only has 9.7 GMACs, which is smaller than other methods. The differences are mainly from modules that process extracted features. In Fig. 7, the decoder of SegNet-Basic takes 14.8 GMACs, which causes a lot of redundancies due to a series of up-sampling operations. The head of PolarMask also takes 15.1 GMACs with a series of convolutional layers. Instead, transformers only take 0.2 GMACs to predict boundary points, which significantly simplifies the process.

Comparison with free space detection methods. We compare our method with previous free space detection methods, which treat free space as a problem of binary segmentation. SegNet is exploited in fish-eye camera free space detection [15]. For practical applications, the basic version of SegNet [3] is selected. To analyze the boundary precision in segmented methods, we post-process the segmented map: getting the outer boundary of the largest connected area and regarding it as the prediction. In Tab. 2, our method has better performance than segmented methods in all the metrics. For example, BAE score reduces by 5.26 pixels, and MAE score reduces by 12.13 pixels. We attribute the improvement to the effective polar representation for free space boundary and the attention for the boundary. In Fig. 9, segmented methods cause false regions and unclear boundaries, which is inefficient.

Comparison with boundary detection methods. PolarMask [20] is proposed to predict the boundary in instance segmentation with center classification and distance regression. With no need for center classification, we remove the centerness head and the classification head in PolarMask, and just cascade an average pooling layer followed by the regression head.

In Fig. 4, PolarMask (blue bars) achieves its best BAE score in 180 sampling polar angles, while our method (green bars) achieves the best BAE score in 360 sampling polar

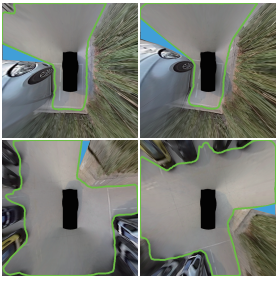

Figure 6: Failure cases.

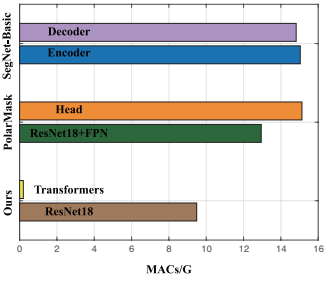

Figure 7: Comparison on computational costs.

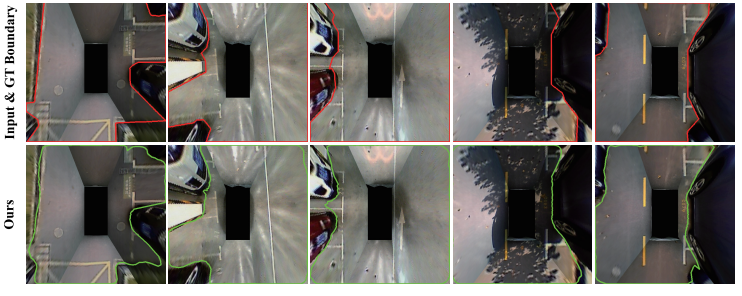

Figure 8: Qualitative results on Tongji Parking-slot Dataset. Red lines are annotations and green lines are predictions. Our model performs well without any additional training.

angles. Moreover, on SVB dataset, our method outperforms PolarMask in BAE score in all numbers of sampling polar angles and the margin becomes significant when the number of sampling polar angles increase. We ascribe it to the capacity of transformers that can capture non-local dependencies to meet the demand of long sequence prediction.

In Tab. 2, our method performs better than PolarMask in all of the mentioned metrics. For example, BAE reduces by 2.94 pixels and δ_1 improves by 7.4%. We think it's because transformers can capture slender structures, and recognize the relationship of obstacles. In Fig. 9 (a), our method recognizes the pedestrian (in yellow box) with clear boundary. However, PolarMask has an ambiguous boundary on the pedestrian. Similar conditions occur in harsh shadows (in yellow box) of Fig. 9 (d) and the pillar (in yellow box) of Fig. 9 (e).

4.6 Generalization on Tongji Parking-slot Dataset

To analyze the generalization ability, we further test on Tongji Parking-slot Dataset [23]. As this dataset lacks annotations of free space, we manually annotate 155 images in the testing set. As it is designed for parking slots, scenarios are relatively simple. We resize the 155 images from 600×600 to 512×512 and test on the selected 155 images with models trained on SVB dataset. Fig. 8 shows the qualitative results. Without any

Methods	BAE↓	MAE↓
SegNet-Basic [3]	18.63	78.89
PolarMask [20]	10.80	13.31
Ours	**5.68**	**8.77**

Table 3: Quantitative results on Tongji Parking-slot Dataset.

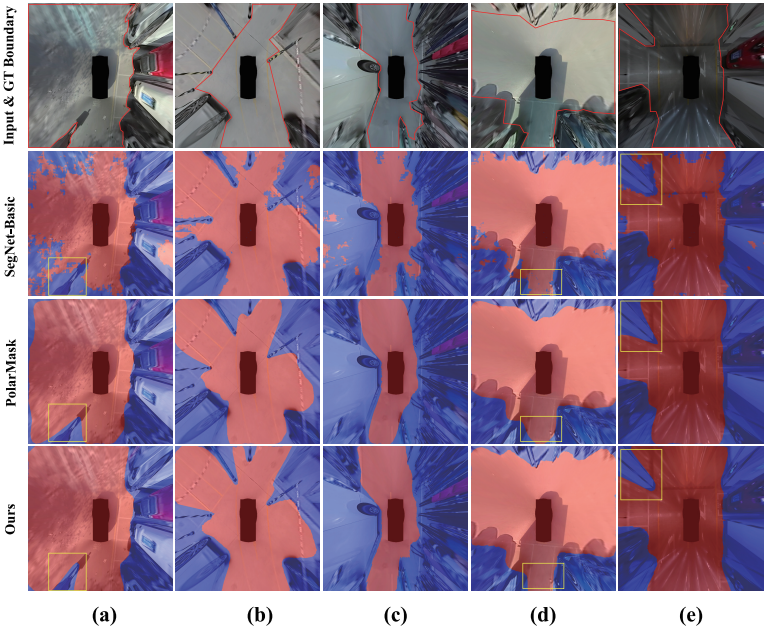

Figure 9: Qualitative results on SVB dataset. We compare with SegNet-Basic [3] and PolarMask [20]. Five challenging scenarios: (a) slender pedestrians, (b) random roadblocks, (c) large-scale obstacles, (d) harsh shadows and (e) indoor parking scenes. Our method has better performance on the above scenarios and behaves strong robustness.

additional training, our model exhibits excellent generalization ability. Moreover, quantitative results in Tab. 3 show that our method even has preciser results, while segmented methods face significant performance degradation in the new dataset. In contrast to per-pixel methods, which are easily influenced by data distribution, directly modeling the free space boundary is beneficial to integrate obstacle information for detection.

5 Conclusion

In this work, we propose to reframe free space detection as polar representation for free space boundary. This representation is explicit to improve attention to the boundary precision and reduce the computational cost. To capture non-local dependencies and restrain the overall shape in the predicted boundary, we exploit a transformer architecture for long sequence prediction and propose a T-IoU loss for better training. In addtion, we have created a large-scale dataset for surround-view free space boundary detection, and supplied a metric to evaluate the boundary precision. Experiments on SVB dataset show that our method can adapt to various and complicated scenarios, and run in real-time with low computational cost. We also demonstrate strong generalization ability in new parking scenes.

Acknowledgement: This work was supported by the National Natural Science Foundation of China (61976170, 91648121).

References

[1] Hernán Badino, Uwe Franke, and Rudolf Mester. Free space computation using stochastic occupancy grids and dynamic programming. In *Workshop on Dynamical Vision, ICCV, Rio de Janeiro, Brazil*, volume 20. Citeseer, 2007.

[2] Hernán Badino, Uwe Franke, and David Pfeiffer. The stixel world - A compact medium level representation of the 3d-world. In *DAGM-Symposium*, volume 5748 of *Lecture Notes in Computer Science*, pages 51–60. Springer, 2009.

[3] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(12):2481–2495, 2017.

[4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV (1)*, volume 12346 of *Lecture Notes in Computer Science*, pages 213–229. Springer, 2020.

[5] Boyo Chen, Buo-Fu Chen, and Chun-Min Hsiao. CNN profiler on polar coordinate images for tropical cyclone structure analysis. In *AAAI*, pages 991–998. AAAI Press, 2021.

[6] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *CVPR*, pages 3213–3223. IEEE Computer Society, 2016.

[7] Jannik Fritsch, Tobias Kuehnl, and Andreas Geiger. A new performance measure and evaluation benchmark for road detection algorithms. In *International Conference on Intelligent Transportation Systems (ITSC)*, 2013.

[8] Noa Garnett, Shai Silberstein, Shaul Oron, Ethan Fetaya, Uri Verner, Ariel Ayash, Vlad Goldner, Rafi Cohen, Kobi Horn, and Dan Levi. Real-time category-based and general obstacle detection for autonomous driving. In *ICCV Workshops*, pages 198–205. IEEE Computer Society, 2017.

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778. IEEE Computer Society, 2016.

[10] Zhenhang Huang, Shihao Sun, and Ruirui Li. Fast single-shot ship instance segmentation based on polar template mask in remote sensing images. In *IGARSS*, pages 1236–1239. IEEE, 2020.

[11] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR (Poster)*, 2015.

[12] Dan Levi, Noa Garnett, and Ethan Fetaya. Stixelnet: A deep convolutional network for obstacle detection and road segmentation. In *BMVC*, pages 109.1–109.12. BMVA Press, 2015.

[13] Ruijin Liu, Zejian Yuan, Tie Liu, and Zhiliang Xiong. End-to-end lane shape prediction with transformers. In *WACV*, pages 3693–3701. IEEE, 2021.

[14] Willem P. Sanberg, Gijs Dubbelman, and Peter H. N. de With. Free-space detection with self-supervised and online trained fully convolutional networks. *CoRR*, abs/1604.02316, 2016.

[15] Tobias Scheck, Adarsh Mallandur, Christian Wiede, and Gangolf Hirtz. Where to drive: free space detection with one fisheye camera. *CoRR*, abs/2011.05822, 2020.

[16] Matthias Schreier and Volker Willert. Robust free space detection in occupancy grid maps by methods of image analysis and dynamic b-spline contour tracking. In *ITSC*, pages 514–521. IEEE, 2012.

[17] Evan Shelhamer, Jonathan Long, and Trevor Darrell. Fully convolutional networks for semantic segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(4):640–651, 2017.

[18] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. FCOS: fully convolutional one-stage object detection. In *ICCV*, pages 9626–9635. IEEE, 2019.

[19] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, pages 5998–6008, 2017.

[20] Enze Xie, Peize Sun, Xiaoge Song, Wenhai Wang, Xuebo Liu, Ding Liang, Chunhua Shen, and Ping Luo. Polarmask: Single shot instance segmentation with polar representation. In *CVPR*, pages 12190–12199. IEEE, 2020.

[21] Jian Yao, Srikumar Ramalingam, Yuichi Taguchi, Yohei Miki, and Raquel Urtasun. Estimating drivable collision-free space from monocular video. In *WACV*, pages 420–427. IEEE Computer Society, 2015.

[22] Senthil Kumar Yogamani, Christian Witt, Hazem Rashed, Sanjaya Nayak, Saquib Mansoor, Padraig Varley, Xavier Perrotton, Derek O'Dea, Patrick Pérez, Ciarán Hughes, Jonathan Horgan, Ganesh Sistu, Sumanth Chennupati, Michal Uricár, Stefan Milz, Martin Simon, and Karl Amende. Woodscape: A multi-task, multi-camera fisheye dataset for autonomous driving. In *ICCV*, pages 9307–9317. IEEE, 2019.

[23] Lin Zhang, Junhao Huang, Xiyuan Li, and Lu Xiong. Vision-based parking-slot detection: A dcnn-based approach and a large-scale benchmark dataset. *IEEE Trans. Image Process.*, 27(11):5350–5364, 2018.

[24] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip H. S. Torr, and Li Zhang. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. *CoRR*, abs/2012.15840, 2020.

[25] Lin Zhou, Haoran Wei, Hao Li, Wenzhe Zhao, Yi Zhang, and Yue Zhang. Arbitrary-oriented object detection in remote sensing images based on polar coordinates. *IEEE Access*, 8:223373–223384, 2020.