

Article

A Fully Convolutional Network-Based Tube Contour Detection Method Using Multi-Exposure Images

Xiaoqi Cheng, Junhua Sun * and Fuqiang Zhou

School of Instrumentation Science and Opto-electronics Engineering, Beihang University, Beijing 100191, China; chengxiaoqi@buaa.edu.cn (X.C.); zfq@buaa.edu.cn (F.Z.)

* Correspondence: sjh@buaa.edu.cn

Abstract: The tube contours in two-dimensional images are important cues for optical three-dimensional reconstruction. Aiming at the practical problems encountered in the application of tube contour detection under complex background, a fully convolutional network (FCN)-based tube contour detection method is proposed. Multi-exposure (ME) images are captured as the input of FCN in order to get information of tube contours in different dynamic ranges, and the U-Net type architecture is adopted by the FCN to achieve pixel-level dense classification. In addition, we propose a new loss function that can help eliminate the adverse effects caused by the positional deviation and jagged morphology of tube contour labels. Finally, we introduce a new dataset called multi-exposure tube contour dataset (METCD) and a new evaluation metric called dilate inaccuracy at optimal dataset scale (DIA-ODS) to reach an overall evaluation of our proposed method. The experimental results show that the proposed method can effectively improve the integrity and accuracy of tube contour detection in complex scenes.

Keywords: fully convolutional network; tube contour detection; multi-exposure images; U-Net; dilation operation

Citation: Cheng, X.; Sun, J.; Zhou, F. A Fully Convolutional Network-Based Tube Contour Detection Method Using Multi-Exposure Images. *Sensors* **2021**, *21*, 4095. <https://doi.org/10.3390/s21124095>

Academic Editor: Stefanos Kollias

Received: 24 April 2021

Accepted: 11 June 2021

Published: 14 June 2021

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



Copyright: © 2021 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 (<http://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Tubes are widely used in the fields of aerospace, automobiles, ships, and other fields for transporting liquids or gases such as fuel, coolant, and lubricating fluid. These tubes are generally metallic. The contours of tubes in two-dimensional images usually appear as edges containing certain shallow features (such as gradient, intensity) and deep features (such as texture, shape, and spatial relation), as shown in Figure 1a. Accurate detection of these contours is very important for achieving three-dimensional reconstruction and measurement of tubes [1–3]. In theoretical research and practical applications, many scholars have proposed various methods that can be used for the detection of tube contours.

In the field of image processing, many edge detection algorithms have been proposed [4,5], and some of them have been used to perform tube contour detection in some single tube measurement applications [3,6]. These kinds of methods only require the gradient information in the image to complete the edge detection work. Therefore, these methods have the advantage of simple design, easy operation and high efficiency. However, due to the lack of high-level features, these methods are easily disturbed by messy backgrounds, uneven lighting and ambient noise.

There are some other researchers [7,8] who adopted the combination of multiple feature descriptors, such as texture, shape and spatial relation, to realize the tubular object recognition under general background. To a certain extent, these kinds of methods improve the robustness and stability of the result. However, these methods need a

considerable amount of expertise for the precise modeling of deep features, which is not only difficult to design but also limits the scope of applications.

Nowadays, deep learning technology has achieved excellent performance in various fields. Especially, fully convolutional networks (FCN) can take arbitrarily sized images as input and achieve pixel-level dense prediction, so they are widely used for image segmentation [9,10], edge detection [11] and so on. Existing FCN models, such as U-Net [12] and HED [13], can also be used for tube contour detection with a little modification. Since the training process of neural networks enables the automatic learning of feature hierarchies, these kinds of methods show great potential in the tube contour detection task. However, the existing FCN models are not designed especially for the task of tube contour detection, and the detection results commonly have problems such as low positioning accuracy and low integrity.

In the present study, aiming at the practical problems encountered in the application of tube contour detection under complex background, a FCN-based tube contour detection method is proposed. Meanwhile, we introduce a new dataset and a new evaluation metric to achieve an overall evaluation of our proposed method.

1.1. Related Work

In this paper, the previous works on tube contour detection are divided into three categories: gradient-based methods, multiple feature-based methods and deep learning-based methods.

1.1.1. Gradient-Based Methods

In the pipeline multi-camera reconstruction system established by Zhang et al. [6], the classical Canny algorithm [4] was adopted to realize tube contour detection, while in the tube axis reconstruction method described by Sun et al. [3], an accurate sub-pixel edge location algorithm based on partial area effect [5] was used to detect tube contours. These methods require the tube contours to appear as step edges, and the edge detection processes depend heavily on the gradient features. Therefore, these methods are easily affected by messy backgrounds, and backlights are generally required in practical applications to obtain high-contrast images with clear tube contour edges. Figure 1b shows the edge detection result of Figure 1a by the Canny algorithm [4].

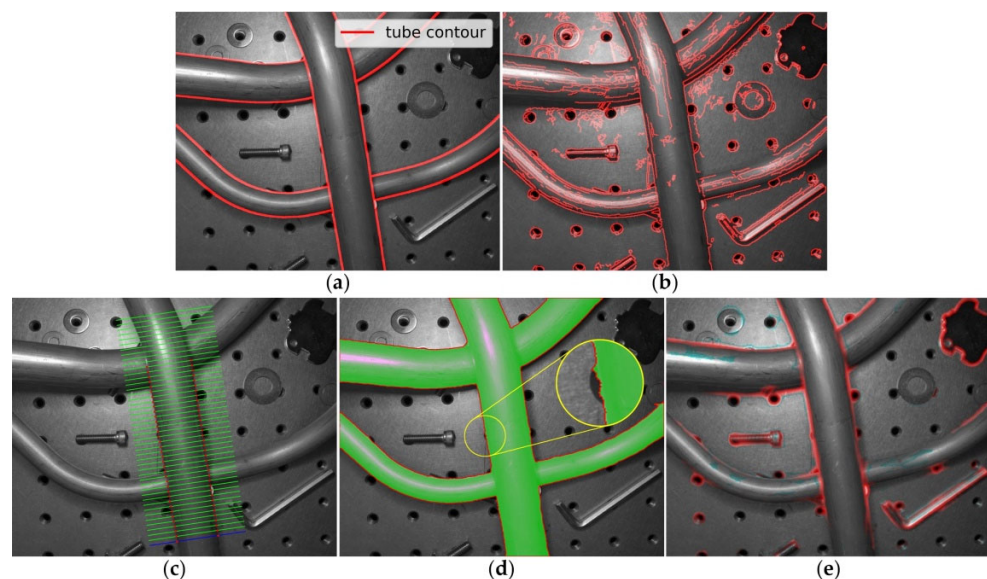


Figure 1. Tube contours detected by different methods. (a) Ground truth; (b) Canny [4]; (c) Aubry [8]; (d) U-Net [12]; and (e) HED [13].

1.1.2. Multiple Feature-Based Methods

Thirion et al. [7] proposed to implement tube segmentation in industrial images by fusing methods from physics-based vision, edge and texture analysis, probabilistic learning and the use of the graph-cut formalism. Figure 1c shows the tube contour detection result of Figure 1a by the method proposed by Aubry et al. [8]. In this method, an intensity profile is first obtained by manually defining a parallel strip segment primitive (the blue line in Figure 1c), then the tubular object recognition process will be accomplished by matching of this intensity profile (the green line in Figure 1c). The tube contour detection task can be further accomplished based on these methods. Due to the combination of multiple features, these methods have shown better performance under general background than methods that use gradient features only.

1.1.3. Deep Learning-Based Methods

Here, we focus on the potential of FCN models in tube contour detection task. The idea of FCN was first introduced by Long et al. [14] for image segmentation. The biggest characteristic of this network is that it converts fully connected layers of traditional convolutional neural network to convolutional layers. Later on, Ronneberger et al. [12] proposed U-Net, which consists of a contracting path to capture context and a symmetric expanding path that enables precise localization. This network has achieved excellent performance in segmentation tasks with small dataset sizes. Such as Bardis et al. [15] proposed to use U-Net to realize organ segmentation with 8–320 samples, and Wang et al. [16] modified U-Net for ore image semantic segmentation with 76 samples. Subsequent networks such as SegNet [17], DeepLab series [18–20] and PSPNet [21] are all FCN-based, and they have shown good performance on segmentation tasks with large-scale datasets. The networks mentioned above can also be used for tube region segmentation directly, but further post-processing is needed to obtain tube contours. Figure 1d shows the tube segmentation result of Figure 1a by the U-Net [12]. Although the tube regions are well segmented (the green area in Figure 1d), the positioning accuracy of tube contours is poor (the red contours in Figure 1d). This is mainly because the training process of segmentation network pays more attention to the classification results of tube's inner regions, which occupy a large proportion, rather than the classification results of contour points, which occupy a small proportion.

Other type of FCN-based networks, such as HED [13] and its variants [22], are designed for general edge detection task. The HED network inserts a side-output layer after the last convolutional layer of each stage in a VGGNet [23], and deep supervision is imposed at each side-output layer, so that the result is toward the edge detection. Figure 1e shows the edge detection result of Figure 1a by the HED [13]. Due to the lack of semantic description of the detected edges, these networks cannot be directly used for tube contour detection under complex background.

In order to detect the edges of specific objects, Yu et al. [24] defined a category-aware semantic edge detection task and proposed CASENet to complete the task. The CASENet adopts a modified ResNet-101 [25] architecture with dilated convolution as the backbone and uses a multi-label loss function to supervise the training process. In recent years, some other networks [26,27] have been proposed for the same purpose. Both of these networks have achieved excellent performance on large standard benchmark datasets such as SBD [28] and Cityscapes [29]. Nevertheless, tube contour detection under complex background is still a challenging task in practical applications, mainly due to the following two aspects:

- Poor tube image quality. For metallic tubes, some areas in the scene may exceed the dynamic range of camera sensor due to problems such as reflections, shadows and uneven lighting, which also means over-saturated or under-saturated regions may appear in tube images. One single low dynamic range (LDR) image cannot provide complete and absolute tube contour information in the process of detection;

- Low contour label accuracy. It is difficult to accurately label the tube contours by manual, and the binary labels naturally have jagged morphology instead of ideal smooth contours. These factors will directly affect the performance of networks.

1.2. Contributions

In order to improve the accuracy and integrity of tube contour detection under complex background, we present a novel FCN-based tube contour detection method. The main contributions of this paper are as follows:

- We propose to use high-resolution multi-exposure (ME) images as the input of an FCN model. These ME images of different dynamic ranges can guarantee the integrity of tube contour information;
- A new loss function, which is calculated based on dilated contours, is introduced and used in the training of FCN. Minimizing this loss function will help to eliminate the adverse effects caused by positional deviation and jagged morphology of tube contour labels;
- We present a new dataset and a new evaluation metric to verify the effectiveness of our proposed method in the tube contour detection task.

2. Method

Currently, most of the neural networks used for image segmentation and edge detection have the following three characteristics: (1) FCN architecture is adopted to ensure that they can be trained end-to-end and provide efficient dense prediction; (2) deconvolutional layers are used to restore the condensed feature maps to full-size at the top of these networks; (3) skip connections are used for combining semantic information from deep layers with appearance information from shallow layers to produce accurate pixel-level classification. As one typical implementation of FCN, the U-Net [12] adopts an encoder-decoder architecture, which is not only simple in structure but also has been proven to be an effective model for image segmentation tasks with small datasets [15,16]. The FCN used for tube contour detection in this study is modified based on the U-Net. What follows is a detailed description of the network architecture and the loss function adopted in the training process.

2.1. Network Architecture

The architecture of the FCN is shown in Figure 2. In this figure, the rectangular blocks represent the feature maps. The height of each rectangular block reflects the size of the feature map, and the width of each rectangular block represents the number of channels in the feature map. In addition, the arrows between the feature maps (except for the orange arrows) represent the layer operations. Different colored arrows indicate different kinds of layers. The detailed correspondence is illustrated at the bottom of the figure. The orange arrows in the figure indicate copying of feature maps.

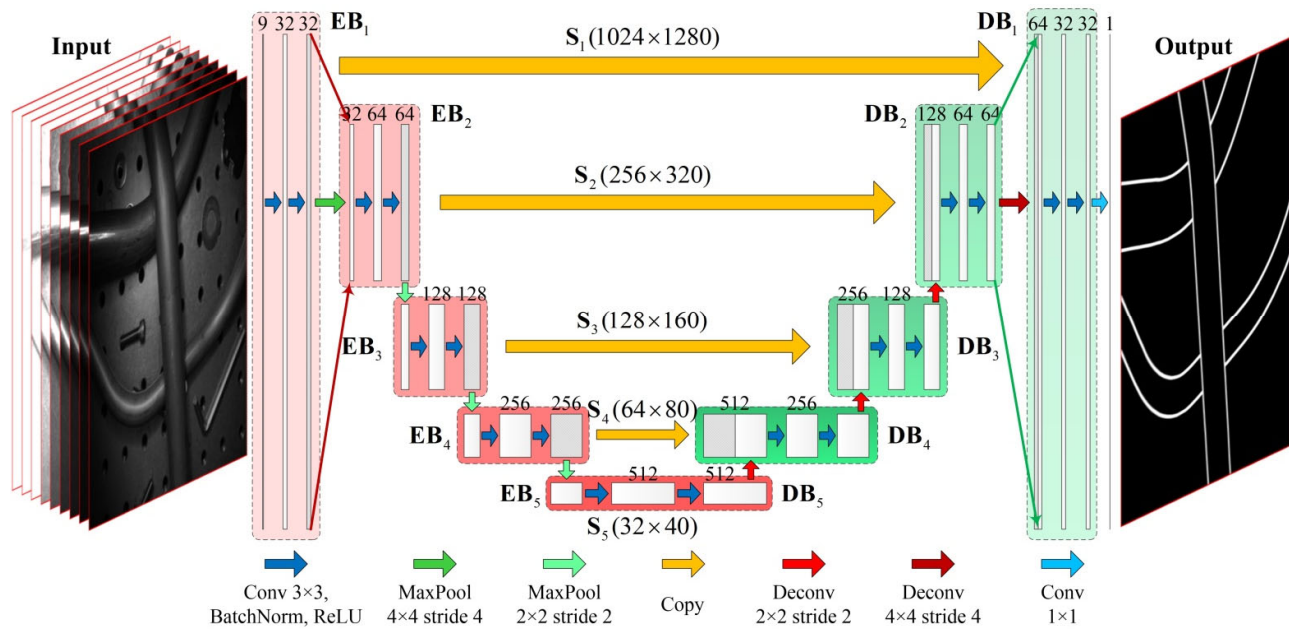


Figure 2. The architecture of the designed FCN.

The FCN takes ME images of a static scene instead of one single image as input. Each group includes under-exposure, normal-exposure, and over-exposure images, to ensure that the network can obtain the information of tube contours in different dynamic ranges. Here, we take 9 images with resolution of 1024×1248 pixels as an example to illustrate the architecture of the customized FCN model based on the U-Net.

The input ME images are first passed through a contracting path containing 5 encode blocks ($EB_1 \sim EB_5$) to extract multi-dimensional feature maps under 5 different spatial scales ($S_1 \sim S_5$). Each encode block contains two 3×3 convolutional layers with 1 pixel zero padding, and each convolutional layer followed by a batch normalization layer and a rectified linear unit (ReLU). The adjacent encode blocks are connected by max pooling layers for down-sampling. The number of feature channels in EB_1 is 32, and it is doubled at each down-sampling step.

Subsequently, the FCN also contains an expansive path, which is composed of 5 decode blocks ($DB_5 \sim DB_1$, where DB_5 is EB_5). This path is used for up-sampling the feature maps of different spatial scales, and eventually outputting a feature map, which has the same size as the input images. The internal structure of decode blocks is the same as encode blocks, but the adjacent decode blocks are connected by deconvolutional layers for up-sampling the feature maps. The number of feature channels is halved at each up-sampling step.

In addition, there are three points to note: (1) All the input feature maps of DB_i ($i = 1 \sim 4$) are composed of two parts: half of the channels are from the output of EB_i on the same spatial scale S_i , and the other half are from the output of DB_{i+1} followed by a deconvolutional operation. This strategy integrates the semantic information from deep layers with appearance information from shallow layers to produce accurate and detailed tube contour detection results; (2) The max pooling layer between EB_1 and EB_2 , and the deconvolutional layer between DB_2 and DB_1 are with kernel size 4×4 and stride 4, while the rest max pooling and deconvolutional layers are with kernel size 2×2 and stride 2. This design aims to increase the contraction and expansion rate of the FCN, so as to reduce the memory space occupied by high-resolution feature maps and increase the receptive field size. Compared to the original U-Net [12], this modification can reduce the memory usage of training by approximately 36%; (3) The output of DB_1 is followed by

a convolutional layer with kernel size 1×1 for converting the 32-channels feature map to a single-channel output.

Based on the above descriptions, the detailed layer configuration of the designed FCN is shown in Figure 3. In this figure, rectangles with different colors indicate different kinds of layers. In summary, the proposed network contains a total of 19 convolutional layers, 4 max pooling layers and 4 deconvolutional layers, among which the first 18 convolutional layers adopt kernel size 3×3 , with each followed by a batch normalization layer and a ReLU layer, while the kernel size of the last convolutional layer is 1×1 . Besides, the first max pooling layer in the contracting path and the last deconvolutional layer in the expansive path are both with kernel size 4×4 and stride 4, while the rest max pooling layers and deconvolutional layers are both with kernel size 2×2 and stride 2. In addition, the dimensions of the input and output feature maps of each block in the designed FCN are shown in Table 1.

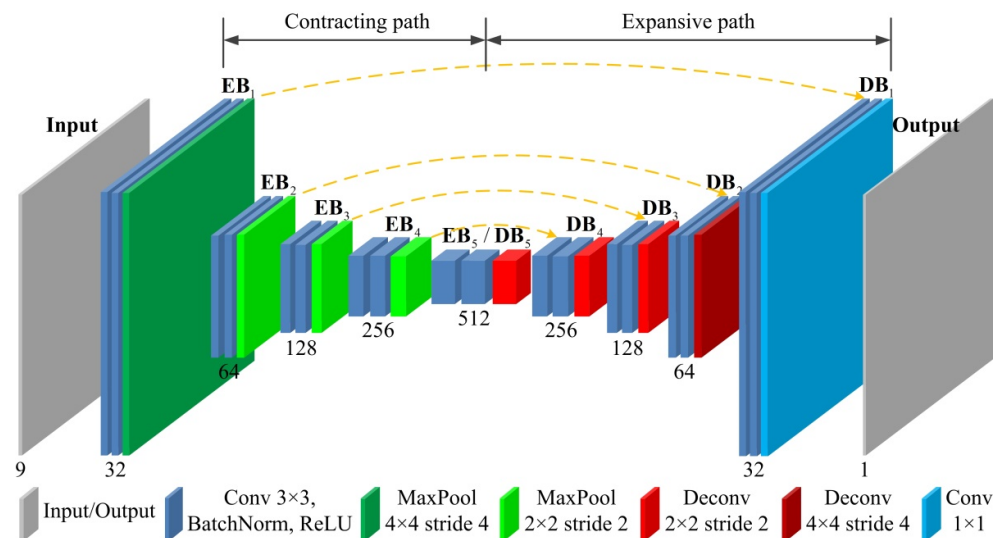


Figure 3. The detailed layer configuration of the designed FCN.

Table 1. The dimensions of feature maps in different spatial scales.

Spatial Scale	Contracting Path			Expansive Path		
	Block	Input	Output	Block	Input	Output
S_1	\mathbf{EB}_1	$1028 \times 1280 \times 9$	$1028 \times 1280 \times 32$	\mathbf{DB}_1	$1028 \times 1280 \times 64$	$1028 \times 1280 \times 32$
S_2	\mathbf{EB}_2	$256 \times 320 \times 32$	$256 \times 320 \times 64$	\mathbf{DB}_2	$256 \times 320 \times 128$	$256 \times 320 \times 64$
S_3	\mathbf{EB}_3	$128 \times 160 \times 64$	$128 \times 160 \times 128$	\mathbf{DB}_3	$128 \times 160 \times 256$	$128 \times 160 \times 128$
S_4	\mathbf{EB}_4	$64 \times 80 \times 128$	$64 \times 80 \times 256$	\mathbf{DB}_4	$64 \times 80 \times 512$	$64 \times 80 \times 256$
S_5	Block	Input		Output		
	$\mathbf{EB}_5(\mathbf{DB}_5)$	$32 \times 40 \times 256$		$32 \times 40 \times 512$		

2.2. Loss Function

The FCN converts the tube contour detection task to a single-label binary classification problem. Hence, the well-known binary cross-entropy (BCE) can be used as the loss function here. Let the input ME images be $\{I_1, \dots, I_H\}$ with the same size of $M \times N$. \mathbf{O} denotes the single-channel output of the FCN, and \mathbf{T} denotes the ground truth contour label corresponding to the input data, where the essence of \mathbf{T} is a binary image of the real contour (0 indicates non-contour, and 1 indicates contour). The output \mathbf{O} first needs to be converted to the tube contour's probability map \mathbf{P} by sigmoid function, and then the BCE loss can be expressed as

$$\mathcal{L}_{BCE} = - \sum_{i=0}^M \sum_{j=0}^N w_{T_{ij}} \left[T_{ij} \log(P_{ij}) + (1 - T_{ij}) \log(1 - P_{ij}) \right] \quad (1)$$

where $P_{ij} = 1 / (1 + \exp^{-O_{ij}})$, and i and j represent the row number and column number respectively. Besides, considering that there are fewer pixels on the tube contours, we also introduce the weight coefficient $w_{T_{ij}}$ into the loss function to balance the positive and negative samples. The implicit premise of using BCE loss is that the contour labels are absolutely accurate. Otherwise, the FCN will be forced to learn inaccurate contours, which is directly manifested as overfitting. However, there are inevitably positional deviations between the manually labeled tube contours and the real tube contours. Meanwhile, the binary labels naturally have jagged morphology. These two factors are not conducive to the training of FCN. Therefore, we propose a new loss function, called dilated contour (DC) loss, for the FCN training. The specific calculation process of DC loss is shown in Figure 4.

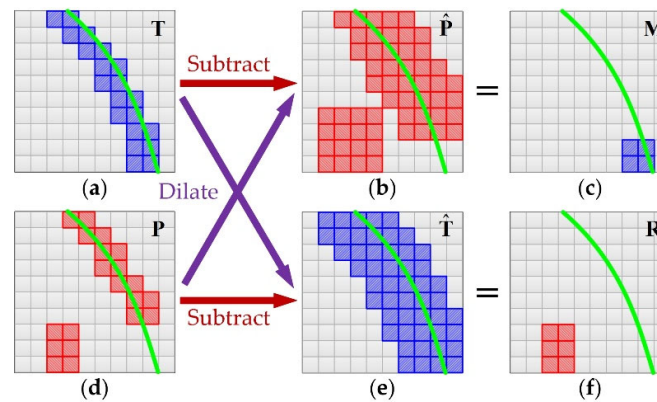


Figure 4. The calculation process of DC loss. (a) Ground truth; (b) dilated probability map; (c) missing contour; (d) probability map; (e) dilated ground truth; (f) redundant contour.

Similar to BCE loss, the calculation process of DC loss also depends on the tube contour's label T and probability map P , as shown in Figure 4a,d. The blue boxes in Figure 4a indicate pixels labeled as tube contour, the red boxes in Figure 4d indicate pixels predicted as tube contour and the green curve indicates the real position of tube contour. First, we perform a morphological dilation operation on T and P , respectively, so as to expand the contour width. This process is the same as max pooling, which slides a window of fixed size (3×3) over the input and takes the max value in the window. Then, we get the dilated contours as shown in Figure 4b,e. The process of dilation can be expressed as follows:

$$\begin{cases} \hat{P}_{ij} = \max_{i',j' \in \{-1,0,1\}} P(i+i', j+j') \\ \hat{T}_{ij} = \max_{i',j' \in \{-1,0,1\}} T(i+i', j+j') \end{cases} \quad (2)$$

Then, we subtract \hat{P} from T and \hat{T} from P . This process can be expressed as follows:

$$\begin{cases} M_{ij} = \text{sat}_{(0,1)}(T_{ij} - \hat{P}_{ij}) \\ R_{ij} = \text{sat}_{(0,1)}(P_{ij} - \hat{T}_{ij}) \end{cases} \quad (3)$$

where sat denotes saturation arithmetic that limits the result to a fixed range, which is 0~1. The subtraction results \mathbf{M} and \mathbf{R} represent the missing and redundant contours of \mathbf{P} , respectively, as shown in Figure 4c,f. Finally, the DC loss can be calculated by \mathbf{M} , \mathbf{R} and \mathbf{T} , as follows:

$$\mathcal{L}_{DC} = \frac{\sum_{i=0}^M \sum_{j=0}^N (\mathbf{M}_{ij} + \mathbf{R}_{ij})}{\sum_{i=0}^M \sum_{j=0}^N \mathbf{T}_{ij}} \quad (4)$$

The DC loss mainly reflects the missing contours and redundant contours due to the introduction of dilation operation. Therefore, the use of DC loss can effectively eliminate the adverse effects caused by positional deviation and jagged morphology of tube contour labels. However, the basic premise of using DC loss is that the FCN can output approximate binary probability maps, which is difficult to meet in the early stage of training. Therefore, it's necessary to comprehensively use BCE loss and DC loss in the whole training process. The effectiveness of this training strategy will be verified in Section 4.2.2.

3. Multi-Exposure Tube Contour Dataset

Considering that there is no appropriate public dataset available for the overall evaluation of our proposed method, we introduce and share a new dataset called multi-exposure tube contour dataset (METCD). Each sample of this dataset contains ME images of a static scene constructed with tubes, the corresponding high dynamic range (HDR) image and tube contour labels with different widths.

3.1. Multi-Exposure Image Acquisition

We constructed a ME images acquisition system, which consisted of an industrial camera (MER-504-10GM-P) from Daheng, an industrial lens (M1214-MP2) from Computar and a LED ring light source. The lens aperture was set to f/5.6. The acquisition objects were 7 metallic tubes; 4 of them were used to acquire train set, while the other 3 were used to acquire validation set. Besides, all these tubes were used to acquire test set. For each scene, we randomly placed 1~3 tubes and some distraction objects (such as bolts, nuts, and washers) in the camera's field of view, and then photographed with 9 different exposure times (2048, 4096, 8192, 16384, 32768, 65536, 131072, 262144 and 524288 microseconds). The acquired ME images were resized to 1024×1280 pixels at last. A sequence of captured tube ME images is shown in Figure 5.

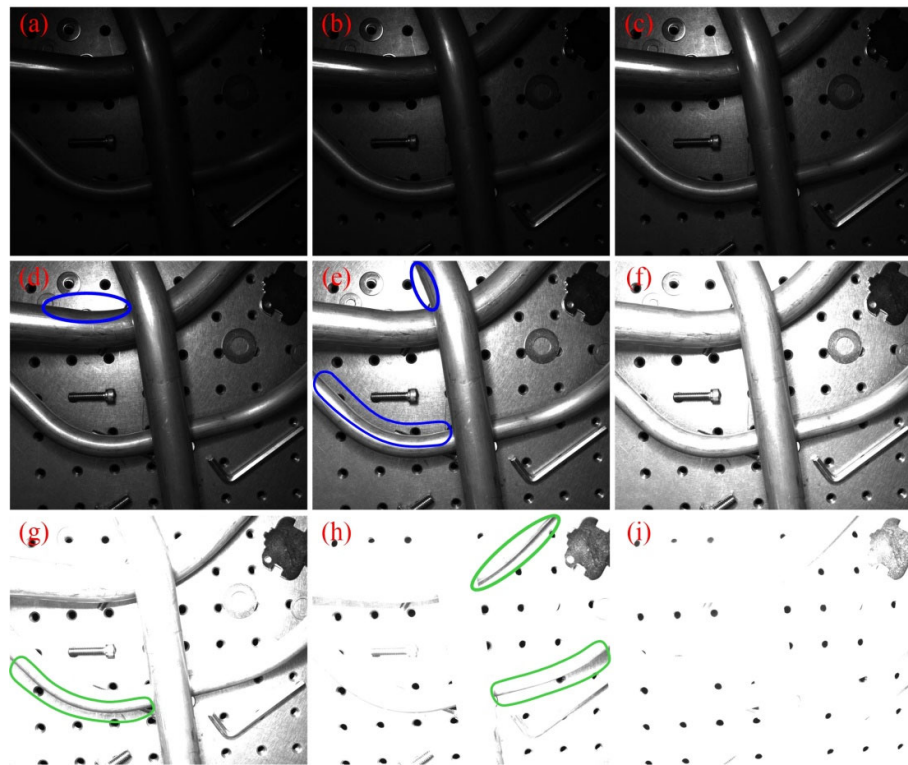


Figure 5. A sequence of tube ME images with different exposure times. (a) 2048 us; (b) 4096 us; (c) 8192 us; (d) 16384 us; (e) 32768 us; (f) 65536 us; (g) 131072 us; (h) 262144 us; and (i) 524288 us.

As we can see from Figure 5, the parameter settings described above can ensure that each group of ME images includes under-exposure, normal-exposure and over-exposure images. By comparison, it can be found that the blue areas in Figure 5d,e tend to reflect light, so the tube contours can be better represented in the images with shorter exposure times. While the green areas in Figure 5g,h are on the dark side, the tube contours can be better represented in the images with longer exposure times. This proves that the use of ME images can effectively guarantee the information integrity of the tube contours. The METCD contains ME images of 72 different scenes, 30 of them are used for FCN training (train set), 10 of them are used for evaluation (validation set) and the rest are used for additional testing (test set).

3.2. Labeling Process

As shown in Figure 5, due to problems such as reflections, shadows and uneven lighting, one single LDR image cannot guarantee that the tube contours all appear as obvious step edges, while an HDR image can represent a greater range of luminance levels [30], which will help to realize the precise labeling of tube contours. Therefore, first we fused the ME images into HDR images using the method proposed by Debevec [31], then we used tonemapper with bilateral filtering and set 2.2 as the value for gamma correction to make the HDR images displayable and finally the tube contours were labeled based on these displayable HDR images. However, it is inevitable that there will be positional deviations (about 1~2 pixels) between the manually labeled tube contours and the real tube contours. We provided four labels with different tube contour widths (1, 2, 4 and 8 pixels) for each sample of METCD. The influence of contour width on the performance of the proposed FCN will be discussed in Section 4.2.3. The tone mapped HDR image of Figure 5 and the corresponding tube contour labels with different widths are shown in Figure 6.

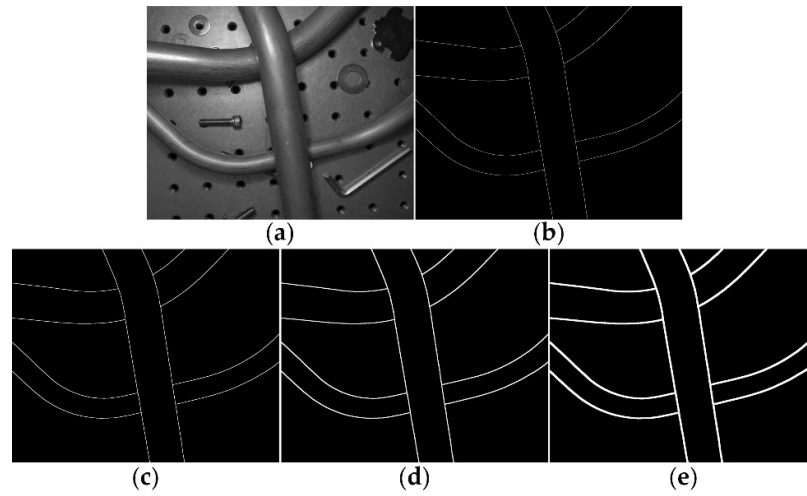


Figure 6. The tone mapped HDR image and the corresponding tube contour labels with different widths. (a) Tone mapped HDR image; (b) 1-pixel contour width; (c) 2-pixel contour width; (d) 4-pixel contour width; and (e) 8-pixel contour width.

3.3. Evaluation Metrics

Tube contour detection is essentially binary classification for each pixel. Therefore, the classical metrics average precision (AP) and maximum F-measure at optimal dataset scale (MF-ODS) are employed to evaluate the performance of different methods on METCD. Here, the optimal dataset scale (ODS) refers to the best result at the optimal threshold across the entire validation set. We also notice that the number of contour pixels is significantly smaller than the number of non-contour pixels. In consequence, the maximum Matthews correlation coefficient at optimal dataset scale (MCC-ODS), which is less influenced by imbalanced data [32], is adopted as another evaluation metric.

In addition, for the same reason as using DC loss, we propose a new evaluation metric called dilate inaccuracy at optimal dataset scale (DIA-ODS) to evaluate the performance of FCN in the tube contour detection task. Figure 7 shows the calculation process of dilate inaccuracy (DIA) for a single sample. In this figure, each small square represents a pixel (black indicates non-contour pixels and white indicates contour pixels). As can be seen from this figure, the calculation process of DIA is quite similar to that of the DC loss shown in Figure 4. The main difference between these two processes is that the calculation of DIA requires a threshold $t \in [0, 1]$ to perform binarization processing on the probability map \mathbf{P} . This process can be expressed as follows:

$$\tilde{\mathbf{P}}_{ij} = \begin{cases} 1 & \mathbf{P}_{ij} > t \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

Then, we substituted the binarized probability map $\tilde{\mathbf{P}}$ and the ground truth label \mathbf{T} into Equation (2) (dilate operation) and Equation (3) (subtract operation), to obtain the missing contour \mathbf{M} and redundant contour \mathbf{R} . Finally, the ratio of the sum number of white pixels in \mathbf{M} and \mathbf{R} to the total number of the white pixels in \mathbf{T} is the DIA of the sample under the current threshold t . Let us take the situation shown in Figure 7 as an example, there are 20 white pixels in \mathbf{T} , and there are 4 and 6 white pixels in \mathbf{M} and \mathbf{R} respectively. Therefore, the DIA of this sample calculated under the current threshold is $(4+6)/20 = 0.5$.

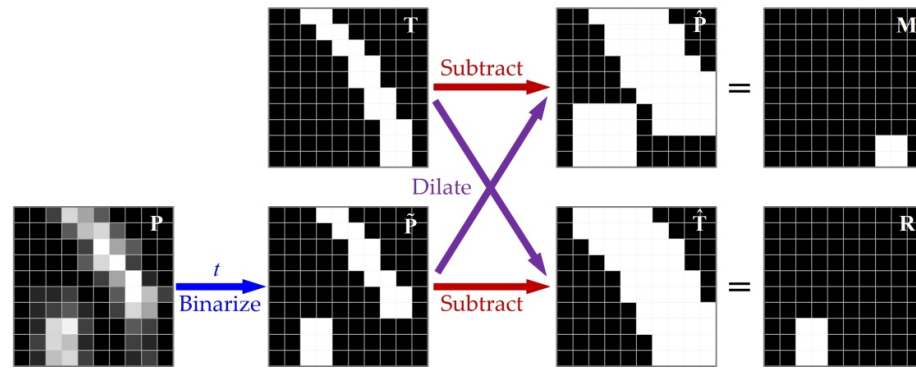


Figure 7. The calculation process of DIA for a single sample.

The above DIA calculation process is completed for a single sample. Suppose there are K samples in the validation set, then there will be probability maps $\{P_1, \dots, P_K\}$ and ground truth labels $\{T_1, \dots, T_K\}$. We choose a fixed threshold t to perform binarization processing on all the $\{P_1, \dots, P_K\}$ and get the binarized probability maps $\{\hat{P}_1, \dots, \hat{P}_K\}$. Then, the calculation of the missing contours $\{M_1, \dots, M_K\}$ and redundant contours $\{R_1, \dots, R_K\}$ for all samples are completed based on the method described in Figure 7. Finally, the DIA calculated on the validation set can be expressed as:

$$\text{DIA} = \frac{\sum_{k=1}^K \sum_{i=0}^M \sum_{j=0}^N (M_{ijk} + R_{ijk})}{\sum_{k=1}^K \sum_{i=0}^M \sum_{j=0}^N T_{ijk}} \quad (6)$$

It is not difficult to find from the above description that different threshold values will yield different DIA results. We select the minimum DIA across the entire validation set by varying threshold t , then we get DIA-ODS. Obviously, the lower DIA-ODS indicates the better performance.

4. Experiments

In order to verify the effectiveness of the proposed method, we conducted several experiments on the METCD. First the hyper-parameters and training strategies adopted in this paper will be detailed in Section 4.1, and then comparative experiments from four aspects are given in Section 4.2. The source code and METCD are available in the GitHub repository: https://github.com/chexqi/Tube_Contour_Detection.

4.1. Experimental Framework

- **Data Augmentation:** Due to the small number of training samples in the METCD, data augmentation was performed to improve the generalization ability of the network and avoid overfitting. We augmented the training data with the following ways: (1) random horizontal and vertical flipping with probability 0.5; (2) random rotation between $(-45^\circ, +45^\circ)$; (3) random horizontal and vertical translation in the scale of $(-0.2, +0.2)$; and (4) random scaling in the range of $(1, 1.2)$.
- **Optimization:** The Adam optimizer was used as the optimization algorithm. The size of mini-batches was set to 3. The learning rate was initialized to 0.01 and decayed by a factor of 0.9 every 70 epochs. We ran all experiments for 700 epochs. The joint loss function adopted in the training process was combined by BCE loss and DC loss, as follows:

$$\mathcal{L} = w_{BCE} \mathcal{L}_{BCE} + w_{DC} \mathcal{L}_{DC} \quad (7)$$

where $w_{BCE}, w_{DC} \in \{0, 1\}$, and they indicate whether \mathcal{L}_{BCE} and \mathcal{L}_{DC} were used in the calculation process of \mathcal{L} , respectively. In addition, in order to solve the problem of imbalance between positive and negative samples, the weight coefficient in Equation (1) was assigned as

$$w_{T_{ij}} = \begin{cases} 1 & \text{if } T_{ij} = 0 \\ 2 & \text{if } T_{ij} = 1 \end{cases} \quad (8)$$

- Implementation: all experiments were conducted on a machine equipped with an Intel Core i9-9900X CPU, 32GB RAM and a NVIDIA GTX 2080Ti GPU.

4.2. Experimental Results

First, we conducted experiments from two aspects, data input and loss function, to validate the effectiveness of using ME images and DC loss. Then, we evaluated the influence of contour width on the performance of the FCN model. Finally, our proposed method was compared with other neural network-based methods. We evaluated each model using the metrics illustrated in Section 3.3 (AP, MF-ODS, MCC-ODS and DIA-ODS), and all these metrics are shown by % in this paper.

4.2.1. Data Input

In order to prove the validity of using ME images as input, we compared the performance of our proposed FCN with three different inputs: one with a single normally exposed LDR image (LDR-FCN), one with a single fused HDR image (HDR-FCN) and one with a sequence of ME images (ME-FCN). In this experiment, the contour labels with 2 pixels width were adopted for training, and the network was optimized only by minimizing the traditional BCE loss ($w_{BCE} = 1$, $w_{DC} = 0$). The other parameters followed the settings described in Section 4.1. In addition, considering the uncertainty caused by random initialization and data augmentation, we ran each test 10 times and averaged the results. The performances of FCN models with different inputs on the validation set are presented in Figure 8 using the box diagram method, while the mean values of evaluation metrics are reported in Table 2.

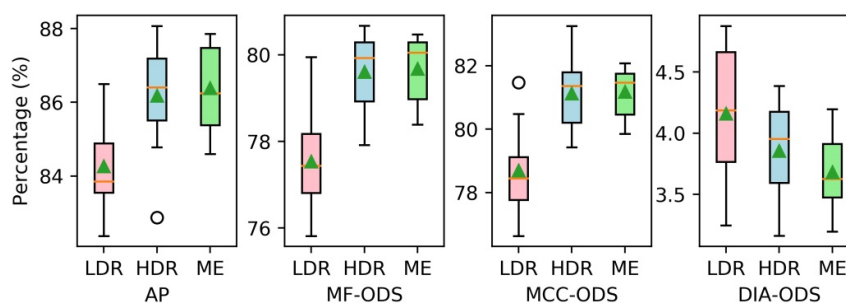


Figure 8. The performances of FCN models with different inputs.

Table 2. The mean values of evaluation metrics computed from FCN models with different inputs.

Models	AP (%)	MF-ODS (%)	MCC-ODS (%)	DIA-ODS (%)
LDR-FCN	84.3	77.5	78.7	4.2
HDR-FCN	86.2	79.6	81.1	3.9
ME-FCN	86.4	79.7	81.2	3.7

It can be clearly observed that the HDR-FCN models and the ME-FCN models achieve higher AP, MF-ODS and MCC-ODS and lower DIA-ODS than the LDR-FCN

models. Furthermore, the ME-FCN models give the best performance in term of AP (86.4%), MF-ODS (79.7%), MCC-ODS (81.2%) and DIA-ODS (3.7%), and avoid using the HDR image fusion operation. Figure 9 shows some comparative tube contour detection results between an LDR-FCN model and an ME-FCN model.

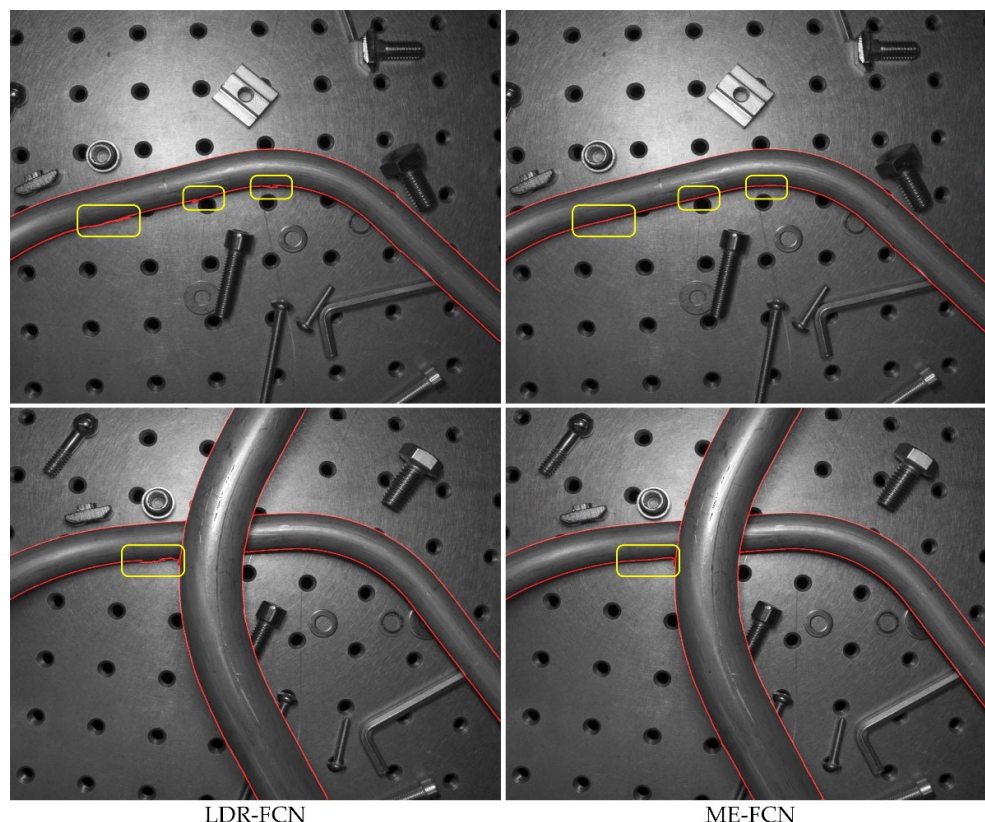


Figure 9. Tube contour detection results on several samples. The images in the left column were produced by an FCN model that took the LDR image as input, and in the right column were produced by an FCN model that took ME images as input.

4.2.2. Loss Function

On the basis of verifying the effectiveness of taking ME images as input, experiment about loss function was conducted. The FCN models in this experiment were trained with two different losses: one with BCE loss only (BCE-FCN, $w_{BCE} = 1$, $w_{DC} = 0$), and the other with both BCE loss and DC loss (BCE-DC-FCN, $w_{BCE} = 1$, $w_{DC} = 0$ in the first 420 epochs and $w_{BCE} = 1$, $w_{DC} = 1$ in the following 280 epochs). The contour labels with 2 pixels width were also adopted here for training. Figure 10 shows the BCE loss and DC loss of the training processes for different FCN models.

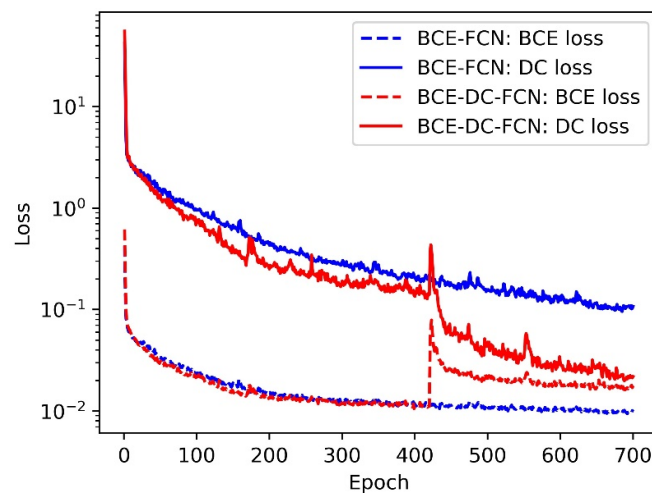


Figure 10. The BCE loss and DC loss of the training processes for FCN models trained with different losses.

The blue curves and the red curves indicate the losses of BCE-FCN model and BCE-DC-FCN model, respectively, while the dashed curves and solid curves refer to the values of BCE loss and DC loss, respectively. As can be seen from Figure 10, compared with FCN trained with only BCE loss, although the training process of BCE-DC-FCN achieves higher BCE loss, it can also effectively reduce DC loss. Then, we ran each test 10 times. The performances of FCN models trained with different loss functions are presented in Figure 11, while the mean values of evaluation metrics are reported in Table 3.

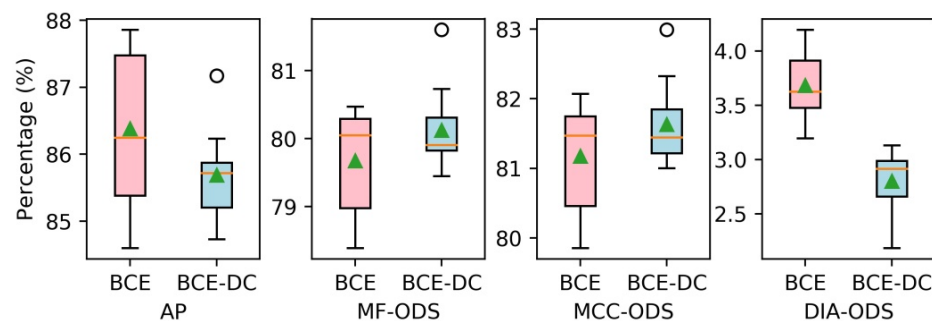


Figure 11. The performances of FCN models trained with different losses.

Table 3. The mean values of evaluation metrics computed from FCN models trained with different losses.

Models	AP (%)	MF-ODS (%)	MCC-ODS (%)	DIA-ODS (%)
BCE-FCN	86.4	79.7	81.2	3.7
BCE-DC-FCN	85.7	80.1	81.6	2.8

As can be seen from Figure 11 and Table 3, the BCE-DC-FCN models almost achieve the same performance in terms of AP (86.4% to 85.7%), MF-ODS (79.7% to 80.1%) and MCC-ODS (81.2% to 81.6%) as the BCE-FCN models, but they significantly reduce the DIA-ODS (3.7% to 2.8%). This indicates that the tube contours detected by BCE-DC-FCN models are completer and more correct (fewer missing contours and redundant contours). Figure 12 shows some comparative tube contour detection results between a BCE-FCN model and a BCE-DC-FCN model.

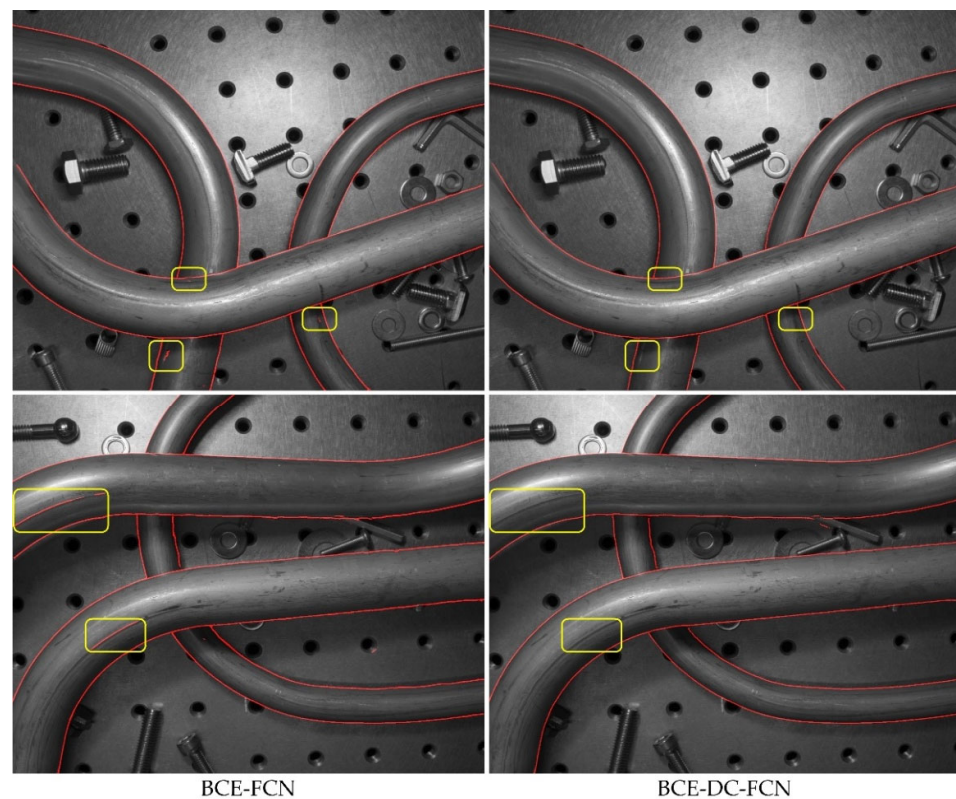


Figure 12. Tube contour detection results on several samples. The images in the left column were produced by an FCN model that was trained with BCE loss, and in the right column were produced by an FCN model that was trained with BCE-DC loss.

4.2.3. Contour Width

Due to the positional deviation and jagged morphology of contour labels, its impossible to achieve absolutely accurate labeling of tube contours. When labels with different contour widths are used for the FCN training and evaluation, the proportion and effect of mislabeled pixels will be different. Therefore, on the basis of confirming the effectiveness of using ME images and BCE-DC loss, we tested the influence of labels with different contour widths on the performance of FCN. The FCN models in this experiment were divided into four groups. They were trained on labels with 1-pixel (Width1-FCN), 2-pixel (Width2-FCN), 4-pixel (Width4-FCN) and 8-pixel (Width8-FCN) contour width, respectively. The other parameters followed the settings described in Section 4.1. The box diagrams and mean values of obtained AP, MF-ODS, MCC-ODS and DIA-ODS are shown in Figure 13 and Table 4, respectively, which depict results after 10 evaluations.

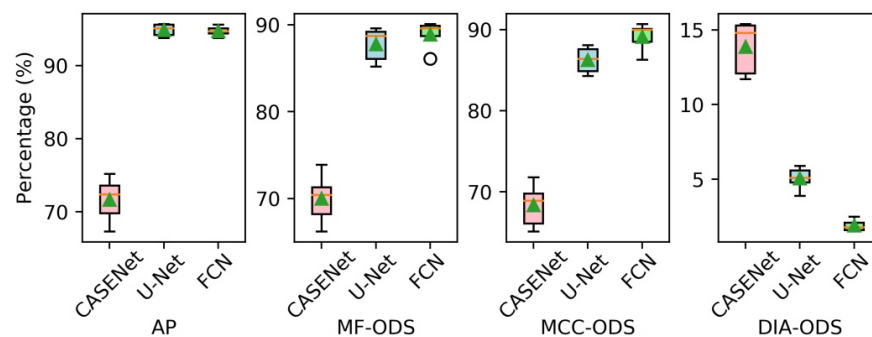


Figure 13. The performances of FCN models trained on labels with different contour widths.

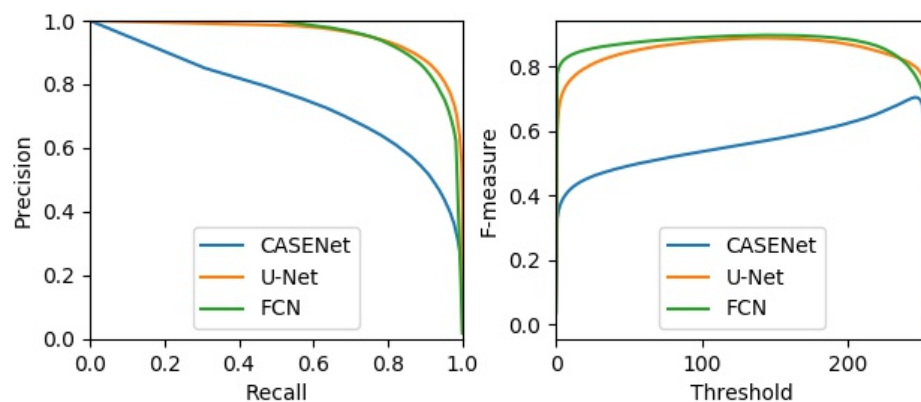
Table 4. The mean values of evaluation metrics computed from FCN models trained on labels with different contour widths.

Models	AP (%)	MF-ODS (%)	MCC-ODS (%)	DIA-ODS (%)
Width1-FCN	58.6	62.7	66.3	4.9
Width2-FCN	85.7	80.1	81.6	2.8
Width4-FCN	94.8	87.7	87.8	2.5
Width8-FCN	98.0	92.2	91.5	2.6

The experimental results show that the AP, MF-ODS and MCC-ODS increase significantly with the increase of contour width. This is mainly due to the decreasing proportion of mislabeled pixels in tube contour labels. However, it should be noted that the DIA-ODS is not effectively reduced when contour width is larger than 2 pixels, which means the missing and redundant contours are not effectively reduced, while the width of the predicted tube contours will be wider as the contour width of ground truth labels increases. This phenomenon is unfavorable for three-dimensional reconstruction and measurement of tubes.

4.2.4. Comparison with Other Methods

In the following, we compared our method with other neural network-based methods that can be used for tube contour detection. The experiment was divided into three groups. In the first group, the CASENet [24] was employed for tube contour detection. This network has been proved to be effective for large-scale semantic edge detection tasks. In the second group, the original U-Net [12] was used for tube contour detection. This network has achieved good results in various image segmentation tasks with small datasets. Our FCN was adopted for tube contour detection in the third group. All the three groups in the experiment were trained and tested on the METCD. Considering that if the high-resolution images (1024×1280 pixels) were directly used as the input of CASENet and original U-Net, the memory used in training with a mini-batch of 3 would reach up to 24.3 GB and 33.9 GB respectively. This was beyond the hardware limit of the experimental platform. So, we first resized the tube images and the 4 pixel-width labels to 512×640 pixels and then carried out experiments with these low-resolution samples. The data enhancement method described in Section 4.1 was adopted by all networks. In addition, the training process of CASENet adopted the multi-label loss described in [24], while the training process of the original U-Net adopted the dice loss introduced in [33]. Figure 14 shows the precision-recall curves and the F-measure curves of different methods. The box diagrams and mean values of obtained AP, MF-ODS, MCC-ODS and DIA-ODS are shown in Figure 15 and Table 5, respectively, which depict results after 10 evaluations.

**Figure 14.** Comparison of precision-recall curves and F-measure curves of different methods.

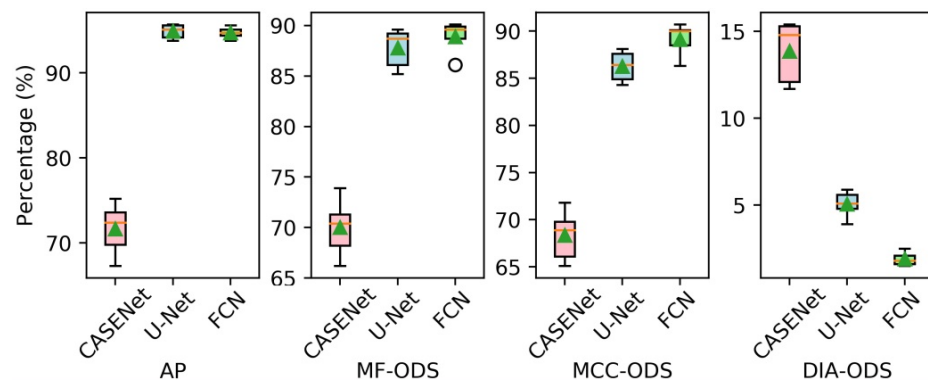


Figure 15. The performances of different methods.

Table 5. The mean values of evaluation metrics computed from different methods.

Models	AP (%)	MF-ODS (%)	MCC-ODS (%)	DIA-ODS (%)
CASENet	71.7	70.0	68.3	13.9
U-Net	94.9	87.8	86.3	5.1
FCN	94.7	88.9	89.1	1.9

As can be seen from Figure 14, Figure 15 and Table 5, the CASENet models achieve the worst performance in terms of AP (71.7%), MF-ODS (70.0%), MCC-ODS (68.3%) and DIA-ODS (13.9%). This indicates that the CASENet is not applicable to the tube contour detection task with small dataset. The original U-Net models and our FCN models almost achieve the same performance in terms of AP (94.9% and 94.7%), MF-ODS (87.8% and 88.9%) and MCC-ODS (86.3% and 89.1%), but our FCN models achieve lower DIA-ODS (1.9%). Figure 16 shows some comparative tube contour detection results between the three kinds of methods.

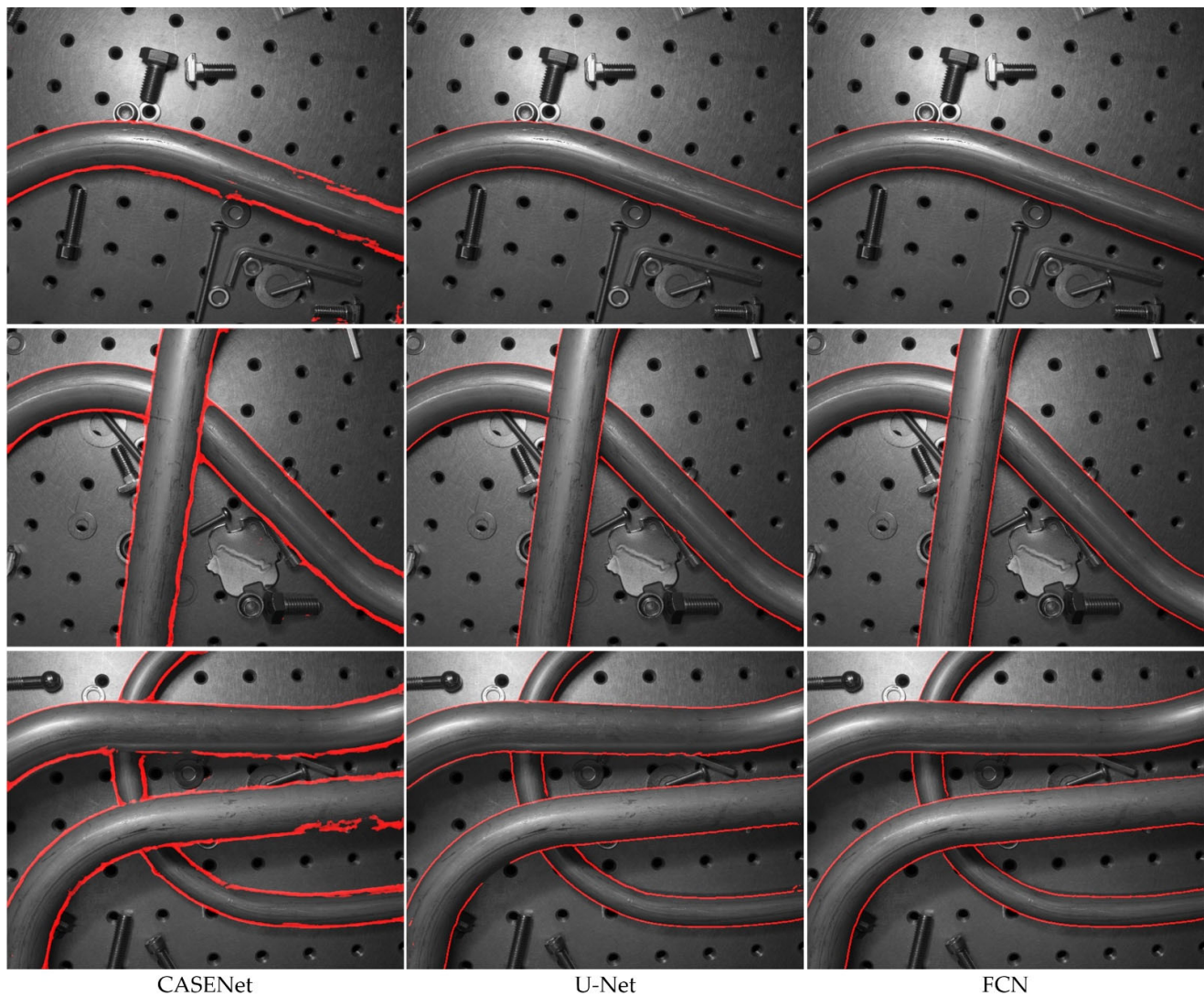


Figure 16. Tube contour detection results of comparing methods on the METCD. The images in the left column were produced by the CASENet model, in the middle column were produced by the original U-Net model and in the right column were produced by our FCN model.

As can be seen from the above experimental results, although our FCN architecture accelerates the contraction and expansion rate of feature maps and enables high-resolution images to be used as input, this does not reduce the performance of tube contour detection. In addition, benefits from the using of ME images and BCE-DC loss are that the tube contours detected by our FCN model are completer and more correct.

Its important to note that the METCD presented in this paper is a small dataset with only 30 samples for training. Therefore, the FCN model trained on the METCD is not applicable to other new scenarios. Figure 17 shows the detection effect of tube contours under a new scenario. Among them, Figure 17a presents the result obtained by an FCN model trained on the METCD and Figure 17b presents the result obtained by an FCN model retrained on the new labeled dataset. It can be seen that under the new scenario, it is necessary to label a new dataset and retrain the model to get the best performance.

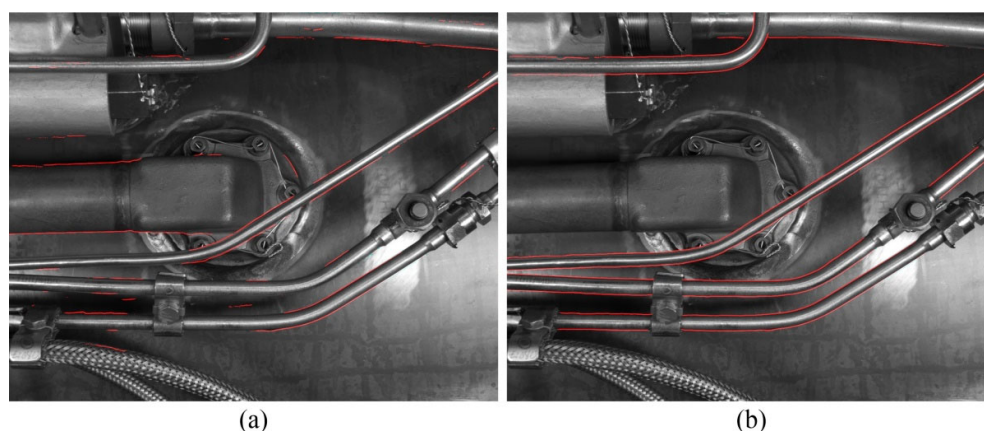


Figure 17. The detection effect of tube contours under a new scenario. (a) Using an FCN model trained on the METCD; (b) using an FCN model retrained on the new labeled dataset.

To the best of our knowledge, this is the first paper in which deep learning method is employed in the tube contour detection task under complex background. The experiments in this section also proved that the proposed method is more suitable for the task of tube contour detection than the other existing neural networks that can be used for the same task. Moreover, the METCD introduced and shared in this paper also provides a useful benchmark for the community of tube contour detection methods. We believe that such a finely labeled dataset is very important for the training and evaluation of new developed tube contour detection methods.

5. Conclusions

Aiming at the practical problems encountered in the application of tube contour detection under complex background, we propose a new FCN-based tube contour detection method. Our FCN architecture is modified based on the U-Net. We accelerate the contraction and expansion rate of feature maps, so as to enable the FCN model to take high-resolution images as input and produce a thin tube contour detection result. In order to solve the problem of poor tube image quality, ME images of a static scene are used as the input of the FCN model. In addition, we propose a new loss function, namely, DC loss, to solve the problem of low contour label accuracy. We also present a new dataset called METCD and a new evaluation metric called DIA-ODS to evaluate the performance of the proposed method. The experimental results show that the proposed method can effectively improve the integrity and accuracy of tube contour detection in complex scenes.

Author Contributions: Conceptualization, X.C. and J.S.; methodology, X.C. and J.S.; software, X.C.; validation, X.C., J.S. and F.Z.; investigation, X.C., J.S. and F.Z.; resources, J.S.; data curation, X.C.; writing—original draft preparation, X.C.; writing—review and editing, J.S.; visualization, F.Z.; supervision, F.Z.; project administration, F.Z.; funding acquisition, F.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by National Key Research and Development Program of China, grant number 2018YFB2003802.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The source code and METCD are available in the GitHub repository (https://github.com/chexqi/Tube_Contour_Detection), accessed on 24 April 2021.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Caglioti, V.; Giusti, A. Reconstruction of canal surfaces from single images under exact perspective. In Proceedings of the European Conference on Computer Vision, Graz, Austria, 7–13 May 2006; pp. 289–300; doi:10.1007/11744023_23
2. Liu, S.; Liu, J.; Jin, P.; Wang, X. Tube measurement based on stereo-vision: a review. *Int. J. Adv. Manuf. Technol.* **2017**, *92*, 2017–2032; doi:10.1007/s00170-017-0254-9
3. Sun, J.; Zhang, Y.; Cheng, X. A high precision 3D reconstruction method for bend tube axis based on binocular stereo vision. *Opt. Express* **2019**, *27*, 2292–2304; doi:10.1364/OE.27.002292
4. Canny, J. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **1986**, *8*, 679–698; doi:10.1109/TPAMI.1986.4767851
5. Trujillo-Pino, A.; Krissian, K.; Alemán-Flores, M.; Santana-Cedr s, D. Accurate subpixel edge location based on partial area effect. *Image Vis. Comput.* **2013**, *31*, 72–90; doi:10.1016/j.imavis.2012.10.005
6. Zhang, T.; Liu, J.; Liu, S.; Tang, C.; Jin, P. A 3D reconstruction method for pipeline inspection based on multi-vision. *Measurement* **2017**, *98*, 35–48; doi:10.1016/j.measurement.2016.11.004
7. Thirion, B.; Bascle, B.; Ramesh, V.; Navab, N. Fusion of color, shading and boundary information for factory pipe segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Hilton Head Island, SC, USA, 13–15 June 2000; pp. 349–356; doi:10.1109/CVPR.2000.854845
8. Aubry, N.; Kerautret, B.; Debled-Rennesson, I.; Even, P. Parallel strip segment recognition and application to metallic tubular object measure. In Proceedings of the International Workshop on Combinatorial Image Analysis, Kolkata, India, 24–27 November 2015; pp. 311–322; doi:10.1007/978-3-319-26145-4_23
9. Ghosh, S.; Pal, A.; Jaiswal, S.; Santosh, K.C. SegFast-V2: Semantic image segmentation with less parameters in deep learning for autonomous driving. *Int. J. Mach. Learn. Cybern.* **2019**, *10*, 3145–3154; doi:10.1007/s13042-019-01005-5
10. Liu, L.; Chen, S.; Zhang, F.; Wu, F.; Pan, Y.; Wang, J. Deep convolutional neural network for automatically segmenting acute ischemic stroke lesion in multi-modality MRI. *Neural Comput. Appl.* **2020**, *32*, 6545–6558; doi:10.1007/s00521-019-04096-x
11. Hu, X.; Liu, Y.; Wang, K.; Ren, B. Learning hybrid convolutional features for edge detection. *Neurocomputing* **2018**, *313*, 337–385; doi:10.1016/j.neucom.2018.05.088
12. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Munich, Germany, 5–9 October 2015; pp. 234–241; doi:10.1007/978-3-319-24574-4_28
13. Xie, S.; Tu, Z. Holistically-nested edge detection. *Int. J. Comput. Vis.* **2017**, *125*, 3–18; doi:10.1007/s11263-017-1004-z
14. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3431–3440; doi:10.1109/CVPR.2015.7298965
15. Bardis, M.; Houshyar, R.; Chantaduly, C.; Ushinsky, A.; Glavis-Bloom, J.; Shaver, M.; Chow, D.; Uchio, E.; Chang, P. Deep learning with limited data: Organ segmentation performance by U-Net. *Electronics* **2020**, *9*, 1199; doi:10.3390/electronics9081199
16. Wang, W.; Li, Q.; Xiao, C.; Zhang, D.; Miao, L.; Wang, L. An improved boundary-aware U-Net for ore image semantic segmentation. *Sensors* **2021**, *21*, 2615; doi:10.3390/s21082615
17. Badrinarayanan, V.; Kendall, A.; Cipolla, R. SegNet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 2481–2495; doi:10.1109/TPAMI.2016.2644615
18. Chen, L.C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. Semantic image segmentation with deep convolutional nets and fully connected CRFs. *arXiv* **2014**, arXiv:1412.7062.
19. Chen, L.C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *40*, 834–848; doi:10.1109/TPAMI.2017.2699184
20. Chen, L.C.; Zhu, Y.; Papandreou, G.; Schroff, F.; Adam, H. Encoder-decoder with atrous separable convolution for semantic image segmentation. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; pp. 833–851; doi:10.1007/978-3-030-01234-2_49
21. Zhao, H.; Shi, J.; Qi, X.; Wang, X.; Jia, J. Pyramid scene parsing network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2881–2890; doi:10.1109/CVPR.2017.660
22. Lou, L.; Zang, S. Research on edge detection method based on improved HED network. *J. Phys. Conf. Ser.* **2020**, *1607*, 012068; doi:10.1088/1742-6596/1607/1/012068
23. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv* **2014**, arXiv:1409.1556.
24. Yu, Z.; Feng, C.; Liu, M.Y.; Ramalingam, S. CASENet: Deep category-aware semantic edge detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1761–1770; doi:10.1109/CVPR.2017.191
25. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778; doi:10.1109/CVPR.2016.90
26. Acuna, D.; Kar, A.; Fidler, S. Devil is in the edges: Learning semantic boundaries from noisy annotations. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 11075–11083. doi:10.1109/CVPR.2019.011133

27. Hu, Y.; Chen, Y.; Li, X.; Feng, J. Dynamic feature fusion for semantic edge detection. In Proceedings of the International Joint Conference on Artificial Intelligence, Macao, China, 10–16 August 2019; pp. 782–788.
28. Hariharan, B.; Arbeláez, P.; Bourdev, L.; Maji, S.; Malik, J. Semantic contours from inverse detectors. In Proceedings of the IEEE International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011; pp. 991–998; doi:10.1109/ICCV.2011.6126343
29. Cordts, M.; Omran, M.; Ramos, S.; Rehfeld, T.; Enzweiler, M.; Benenson, R.; Franke, U.; Roth, S.; Schiele, B. The cityscapes dataset for semantic urban scene understanding. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 3213–3223; doi:10.1109/CVPR.2016.350
30. Song, Y.; Jiang, G.; Yu, M.; Zhang, Y.; Shao, F.; Peng, Z. Naturalness index for a tone-mapped high dynamic range image. *Appl. Opt.* **2016**, *55*, 10084–10091; doi:10.1364/AO.55.010084
31. Debevec, P.E.; Malik, J. Recovering high dynamic range radiance maps from photographs. In Proceedings of the Annual Conference on Computer Graphics and Interactive Techniques, Los Angeles, CA, USA, 3–8 August 1997; pp. 369–378; doi:10.1145/1401132.1401174.
32. Luque, A.; Carrasco, A.; Martín, A.; de las Heras, A. The impact of class imbalance in classification performance metrics based on the binary confusion matrix. *Pattern Recognit.* **2019**, *91*, 216–231; doi:10.1016/j.patcog.2019.02.023
33. Milletari, F.; Navab, N.; Ahmadi, S.A. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In Proceedings of the International Conference on 3D Vision, Stanford, CA, USA, 25–28 October 2016; pp. 565–571; doi:10.1109/3DV.2016.79