Mobile phone screen defect segmentation based on deeplabv3p

Mingwei Xing 36920221153130, Shiran Bian 23020221154068,
Hang Wu 31520221154227, Longyu Cheng 23020221154077

Institute of Artificial Intelligence, Xiamen University, China

Abstract

Recently, with the development of intelligent manufacturing, the demand for surface defect inspection is increasing. However, traditional defect detection methods require manual feature extraction for different defect types, which is not universal, while deep learning has achieved promising results in defect inspection. Based on this, we proposed a network based on the deeplabv3p including the Enhancement Module, the Calibration Module, and deeplabv3p improving with Denser ASPP (Atrous Spatial Pyramid Pooling). We first improve deeplabv3p by using Denser ASPP for problems with small defects, difficult identification and introduce a refinement structure for incomplete segmentation results for mobile screen. Then we proposed the Enhancement Module aiming at the problems of glass panel noise, low contrast, small defects and difficulty to identify. Additionally, we suggest the Calibration Module to address the issue of incorrect labeling. Finally, Our method is implemented on the mobile screen defect dataset (MSDD-3) which is collected from the industrial assembly line. Furthermore, in comprehensive experiments, we demonstrate that our model outperforms other methods in MSDD-3, which achieves 80.88% mIOU.

Introduction

With the progress of technology and globalization, defect detection shows its great significance to ensuring product quality and safety, so it is essential to found defect quickly and correctly. However, defect detection in industrial scenarios also faces additional challenges, such as small differences between defect imaging and background, low contrast, large variations in defect scale and type, and large amounts of noise in defect images. However, product surface defect detection is an important part of machine vision inspection, and the accuracy of its detection directly affects the final quality of the product. Deep learning has achieved good results in this area.

Inspired by the success of deep learning on object recognition, it has been gradually applied to surface defect inspection and become the mainstream method due to its superiority to detect tiny and complicated defects. Deep learning-based defect inspection methods are divided into three types: defect classification, defect detection, and defect segmentation. Defect classification aims to give the image-level label of the defect, while defect detection aims to locate the defect with the box-level label given. Because of the changes in the shapes and scales as well as the irregularity of defects, the first two types of methods are difficult to precisely describe the location and shape of complicated defects. Hence, the segmentation method with pixel-level label attracts more attention. In this paper, we mainly focus on discussing defect segmentation.

In defect segmentation methods, the fruitful defect detection methods include the traditional image-processing methods and the classic machine vision methods. However, these methods highly rely on handcrafted features from domain knowledge and the subjective experience of the designer, which is unable to ensure the model flexibility, owing to the wide variety, random shape, and unfixed location. While deep learning technology, especially Convolutional Neural Network, have got many achievements. Deeplab (Chen et al. 2016) is a semantic image segmentation model, which used atrous convolution to solve the problem that the signal's details are lost when doing down-sampling. On the basis of deeplabv3, deeplabv3p (Chen et al. 2018) using Encoder-Decoder structure to refine the segmentation results. Previous defect segmentation works are likely using deep learning methods like Unet (Ronneberger, Fischer, and Brox 2015) or FCN (Long, Shelhamer, and Darrell 2014). But It is observed that due to the limited defective images of industrial products, defect segmentation for high-resolution images is subject to a typical modes of failures: defects are easy to be misclassified because some are extremely tiny, irregular in shape, and too low-contrast to be easily confused with the background. In our work, we use the improved deeplabv3p model for defect segmentation of the mobile phone screen, and the major contribution of the proposed methods are as follows:

- We improve the deeplabv3p model. We propose a multiple small holes convolution stack structure (Denser ASPP) to solve the problem of small defects and difficulty in recognition. A refinement structure is proposed aiming at the problem of imprecise segmentation results, which uses two branch parallel convolution layer architecture to enhance segmentation fineness.
- We design a Enhancement Module considering the glass panel noise, low contrast, small defects are difficult to identify, which includes Fourier transform, gamma trans-

form and morphological expansion operations. And the Calibration Module is proposed in view of the problem of inaccurate labeling results.

- In comprehensive experiments, we demonstrate that our model outperforms other methods in MSDD-3, which achieves 80.88% mIOU.

Related Work

Traditional Defect Detection.

Traditional feature-based methods are mainly based on the color, shape and other features of the defect, using image processing techniques or combined with traditional machine learning methods to detect the defect. One of the challenges is how to describe defects. Various traditional image processing methods have been proposed to detect defects in images such as thresholding-based methods (Ng 2004), segmentation-based methods (Oliveira and Correia 2009), and edge-detection methods (Dong and Shisheng 2008; Yang, Qi, and Li 2010) using edge detectors such as Sobel (Kanopoulos, Vasanthavada, and Baker 1988). However, these methods are extremely influenced by noise, light, and complicated backgrounds. Hence, the problem of defect inspection is solved in a frequent space. Fourier Transform (liang Wang and Zuo 2016), Gabor Transform, and Wavelet Transform are applied to convert images to frequency domain for better detection. Hou et al. (Hou and Parker 2005) use the Gabor Wavelet Transform operator suitable for texture expression to extract the frequency domain in formation of the image.

Deep Learning based Defect Segmentation.

Semantic segmentation is the task of predicting pixel-level category labels from images. The introduction of fully convolutional neural network (Long, Shelhamer, and Darrell 2014) is a remarkable milestonein semantic segmentation. Most following works build upon it and either take advantage of multi-scale inputs (Dai, He, and Sun 2014; Lin et al. 2015) , or use feature pyramid spatial pooling (Liu, Rabinovich, and Berg 2015; Zhao et al. 2016), or dilated convolutions (Chen et al. 2016, 2017; Wang et al. 2018) to improve the model, and encoder-decoder models (Chen et al. 2018; Li et al. 2018; Badrinarayanan, Kendall, and Cipolla 2017) have also been proved effective. Most following works build upon it. For example, Qiu et al. (Qiu, Wu, and Yu 2019) propose a three-stage supervised segmentation method based on FCN. Tabernik et al. (Tabernik et al. 2020) first use a segmentation network based on FCN to locate surface defects and then use a decision network to predict the probability of defects in the whole image. Huang et al. (Huang, Qiu, and Yuan 2020) integrate saliency detection based on U-Net architecture and input the superposition of the image processed by various saliency methods and the original image. Xie et al. (Xie, Zhu, and Fu 2020) extract the frequency domain features of the image based on discrete Wavelet Transform and fuse them with the multi-scale features of the backbone network, which effectively improves the segmentation ability for small cement cracks.

Methods

Overview. Our network is mostly built on deeplabv3p (Chen et al. 2018), with a number of enhancements. Specifically, it includes the Enhancement Module, the Calibration Module, and the improvement of deeplabv3p using Denser ASPP for the characteristics of generally small defects. As shown in Figure.1, the input image is enhanced by the Enhancement Module, and then features are extracted by a deep neural network. The high-dimensional features are concatenated with the low-dimensional features by the Denser ASPP module, and each pixel is classified by refinement module. Finally use the Calibration Module to get the final prediction result.

Denser ASPP. ASPP(Atrous Spatial Pyramid Pooling) is consists of one 1×1 convolutional layer, three atrous convolutional layers with customizable dilation rates, and three pooling layers. The dilation factor of the atrous convolutional layer can be customized to achieve free multi-scale feature extraction. For each atrous convolutional layer, it is applied as:

$$z[i] = \sum_{k=1}^{K} f[i + r \cdot k] \cdot o[k] \tag{1}$$

where z is the output feature, i is the location and f is the feature map. r represent the atrous rate corresponding to the stride with which we sample the input and $o[k]$ is the k_{th} step convolution operation.

In deeplabv3p, the ASPP module uses 4 atrous convolutions with dilation rates of 1, 6, 12, 18 and a pooling layer. It can be expressed as follows:

$$\begin{aligned} z =& h_{1,1}(f) + h_{3,6}(f) + h_{3,12}(f) \\ &+ h_{3,18}(f) + h_{1,1}(f) + g(f) \end{aligned} \tag{2}$$

where $h_{k,r}(f)$ denote an atrous convolution, $g(f)$ represents global pooling operation.

Atrous convolution can expand the receptive field and obtain multi-scale context information while the parameter amount remains unchanged. This works very well for detecting large objects. However, in industrial defect detection, the defects are usually very small. If the expansion rate used is too large, the output image will become sparse, and too much local information will be lost, resulting in failure to correctly identify small defects. Therefore, we use 8 atrous convolutions with small dilation rates (dilation rates of 1-8, respectively) and a global average pooling to compose the denser ASPP module. This denser structure can obtain more image details, which is very useful for small defects. It can be expressed as follows:

$$z = h_{1,1}(f) + \sum_{i=2}^{8} h_{3,i}(f) + g(f) \tag{3}$$

Refinement Module. In the decoder stage, high dimensional features and low-dimensional features are merged in the channel dimension. It can comprehensively utilize multiple level features to realize the complementary advantages of multiple features and obtain more robust and accurate recognition results. In deeplabv3p, the merged features only get the prediction result of the model through a 3x3 convolution and a 1x1 convolution. We believe that this simple

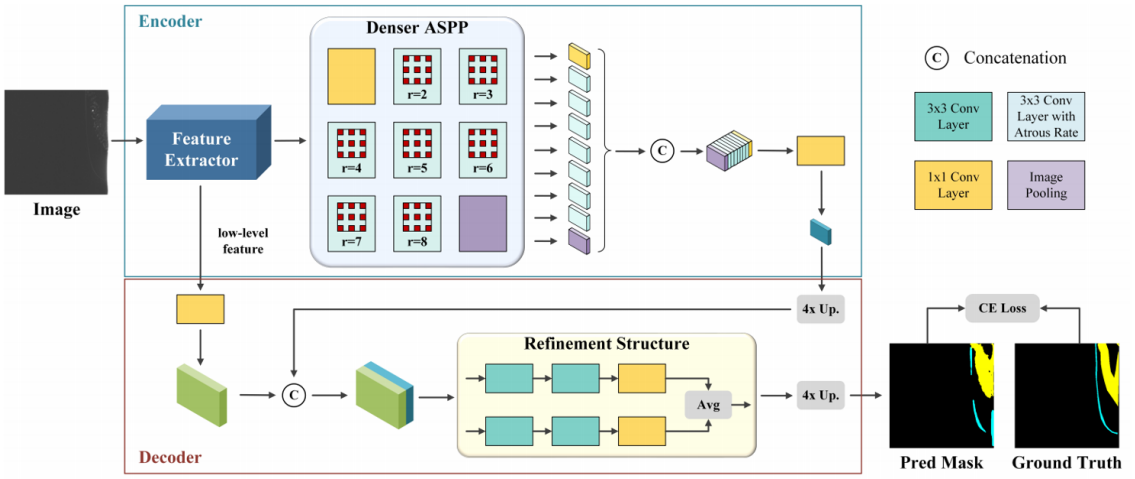

Figure 1: The structure of segmentation network. The encoder module encodes multi-scale contextual information by applying denser atrous convolution at multiple scales, while the simple yet effective decoder module refines the segmentation results along object boundaries.

structure cannot fully utilize the merged features, resulting in partial information loss. Therefore, we designed a more complex structure: two 3x3 convolutions and a 1x1 convolution in series and then an identical structure in parallel. The final output takes the average of the two submolecules. Through this Refinement Module, the loss of information can be effectively reduced. it can be formulated as:

$$p(x) = R(h_{1,1}(f_{low} \oplus \hat{h}_{1,1}(z))) \tag{4}$$

where f_{low} is the low-level feature of feature extractor, $\hat{h}_{1,1}$ denotes a upsampling operation attach to 1×1 conv layer, $\oplus$ denotes the concatenate operation, and R denotes the refinement structure. Therefore, the segmentation network can obtain more fine-grained context information, extremely improving the characteristics of generally tiny defects.

Enhancement Module. The defects in MSDD-3 dataset are small and difficult to identify, resulting in difficulties in labeling and identification. The function of the Enhancement Module is to make small defects more obvious. First, we apply a dilation operation to the mask of the image . This makes the defect boundary expand outwards and alleviates the error of defect boundary labeling during the labeling process. Since the image of the defect part is almost merged with the background, it is very difficult for the model to identify the defect. So we used a gamma transform to improve the contrast between the background and the defect. The gamma transformation can be expressed as:

$$s = cr^{\gamma} \tag{5}$$

$r \in (0, 1)$ is the input value of the grayscale image, and c is the grayscale scaling factor, usually 1. γ is the gamma factor size which controls the scaling of the entire transform.

If the detection is performed in the spatial domain, the shape and size of the defect are not fixed, and the defect is not clearly distinguished from the background, and there is also the influence of noise, which increases the difficulty of detection. Therefore, we transform the image from the spatial domain to the frequency domain through Fourier transform. It can be expressed as:

$$F(\tilde{x})(m.n) = \sum_{w=0}^{W-1} \sum_{h=0}^{H-1} \tilde{x}(h, w) e^{-j2\pi(\frac{h}{H}m + \frac{w}{W}n)} \tag{6}$$

where $j^2 = -1$, x is the gray-scale value after gamma transformation.

Then, we design a frequency domain filter M_β to separate the background information and noise, and retain the defect information, M_β is defined as:

$$M_\beta(h, w) = \mathbb{1}_{(h,w) \in [-\beta H:\beta H, -\beta W:\beta W]} \tag{7}$$

where $\mathbb{1}$ is the indicator function, and $\beta \in (0, 1)$. Then with the help of M_β, we can apply the inverse Fourier transform(F^{-1}) to reconstruct image, turned back to the spatial domain, it can be represented as:

$$\tilde{x}_g = F^{-1}(M_\beta \odot F(\tilde{x})) \tag{8}$$

Calibration Module. The edges of defects are usually blurred and easily mixed with the background, so the detection of defect edges is often more difficult than the main body of defects. So we propose a correction module to correct the results predicted by the model, especially for edge regions. After the model outputs the prediction of the image, by analyzing the predicted category information, the matching convolution kernel size is automatically selected for different categories, and then the corresponding expansion operation is performed to enhance the prediction accuracy of the defect edge.

Experiments

Experimental Setups

Mobile Screen Defect Dataset (MSDD-3). The defect images are collected from the real industrial production line for

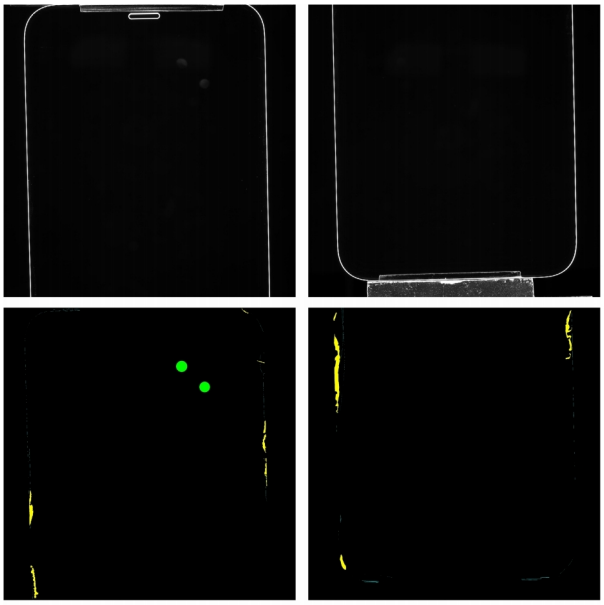

Figure 2: Some examples of defect samples and labels on MSDD-3. The label of "dust" for this dataset is a weakly supervised label, that is, the defective areas are represented by a coarse-grained mask.

mobile screens, the size of an image is 5120×5120 and the mobile screens have three classes of frequent defects (bubble,dust, scratch) and one background class. Note that "dust" is not strictly a surface defect, but it can interfere with the detection of defects. Therefore, we also consider the coarse-grained segmentation of "dust" region. As shown in Figure 2, each column represents a defect sample (top) with a corresponding label (bottom), where the black part is the background (non-defective) region, while the colorful region represents different defect types. To better visualize each defect that exists in the high-resolution images, we show the defects processed by the data enhancement module in Figure 3.

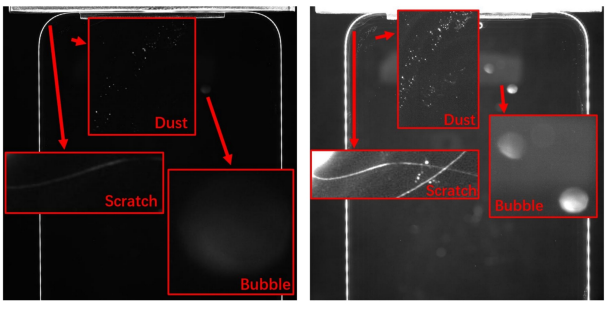

Figure 3: Visualization of highlighted defects. (a) Raw defects. (b) Defects handled by the data Enhancement module.

Table 1: Comparison of different methods.E represents Enhancement Module. C represents Calibration Module.

Methods	Module	mIOU	background	bubble	scratch	dust
D3p	-	72.96	98.34	69.56	55.64	68.29
	+E	78.78	98.44	78.01	65.96	72.73
	+C	74.77	98.35	69.19	64.04	67.49
	+E+C	80.51	98.51	80.4	69.82	73.32
FPN	-	69.51	98.1	70.77	42.75	66.44
	+E	75.45	98.15	5.8	57.79	70.08
	+C	71.89	98.17	73.52	48.01	67.83
	+E+C	75.69	98.18	77.61	58.08	68.9
Unet	-	69.95	98.18	55.5	57.7	68.49
	+E	75.45	98.22	70.7	66.11	66.8
	+C	72.04	98.07	59.7	62.3	68.12
	+E+C	77.04	98.27	73	67.17	69.72
Ours	+E+C	80.88	98.54	81.42	69.89	73.68

Due to hardware performance limitations, the high-resolution image with the size of 5120 × 5120 results in high computational resources in training. Therefore, we divide an image into several patches with the relatively smaller crop size (i.e.,512×512). And then similar to the building process of other datasets, each cropped block is rotated and flipped for data augmentation. Thus, MSDD-3 is built which is divided into the training set with 14400 images and the validation set with 2000 images.

Implementation Details. For fair comparisons, we consider a lightweight network by employing our segmentation network based on ImageNet pre-trained ResNet-18 as our backbone segmentation network for the main experiments to demonstrate the effectiveness of our method. For training the network, we adopt SGD optimizer with initial learning rate 0.001 on MSDD-3 dataset. The momentum and weight decay are set to 0.9 and 0.0001, respectively. The learning rate of the randomly initialized segmentation head is 10× larger than that of backbones. We use the poly scheduling to decay the learning rate during the training process: $lr = \mathrm{lr}_{\mathrm{base}} \times \left(1 - \frac{\mathrm{iter}}{\mathrm{total\ iter}}\right)^{0.9}$. Besides, the number of epochs is set to 60. All experiments are conducted on PyTorch framework, the training batch size is set as 8 with only one GeForce RTX TITAN GPU. The segmentation performance evaluation use mean Intersection-over-Union (mIoU) metrics which is the consistent standard for semantic segmentation tasks.

Experimental Results

We compare our approach with three different semantic segmentation methods, namely FPN, UNet, and Deeplabv3p. For fair comparison, we re-implemented all of the above methods using the same experimental setup, and adopted the same network architecture and test set. The comparison of MSDD-3 is shown in Table 1. Among the methods compared, our method achieved the best 80.88%. And we use Enhancement module and correction module in each comparison method. After using the Enhancement module, the gain on D3p, FPN and Unet is 5.82%, 5.94% and 5.80% respectively, which indicates that the Enhancement module

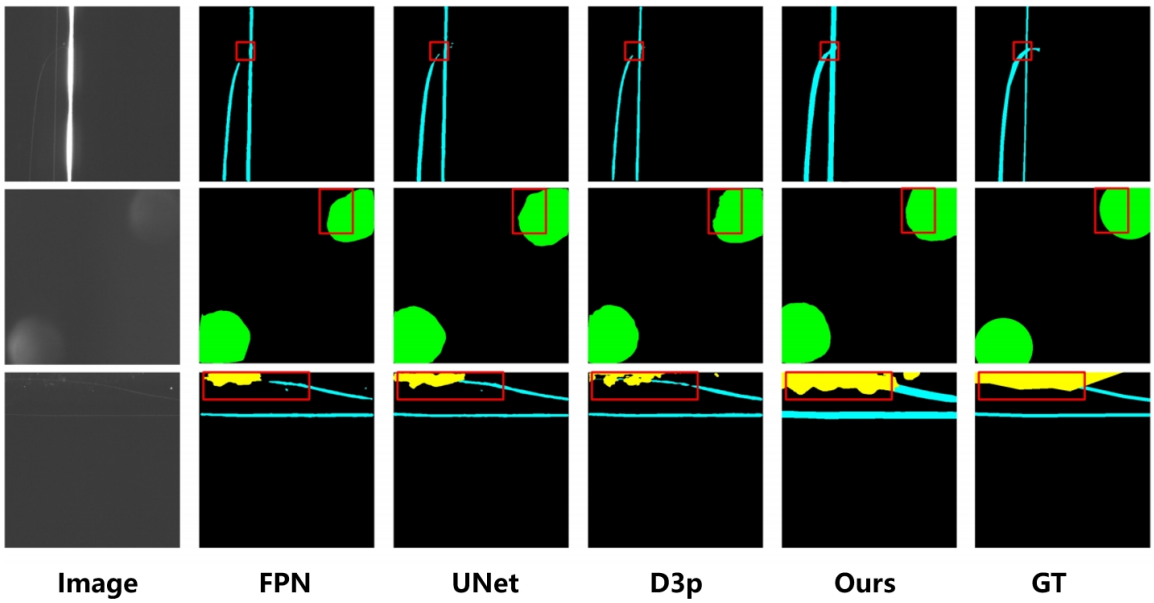

Figure 4: Compared to other methods. Green represents bubbles; Yellow represents the ash layer; Blue is for scratches

can effectively improve the image quality and help the depth model learn features better for specific glass panel data. After using the correction module, the gains of the three methods are 1.81%, 2.38% and 2.39% respectively, which indicates that the label correction module can effectively correct the predicted results of the model. If the two modules are used at the same time, the gains of the three methods are respectively 7.55%, 6.18% and 7.39%, indicating that the two modules can effectively complement each other. Our improved deeplabv3p model is 0.37% better than the original deeplabv3p model after the same use of both modules. Finally, we give the qualitative segmentation results, as shown in Figure 4. Our method is closest to the ground truth, and the segmentation results are smoother.

Ablation Studies

We have verified the effectiveness of our proposed method through sufficient experiments, as shown in Table 2. The first part compares the effectiveness of using a Enhancement module with a correction module, with a gain of 5.82% for the Enhancement module, 1.81% for the correction module, and 7.55% for using both modules simultaneously. The second section compares the effectiveness of improved sections on deeplabv3p, offering 0.39% gain using denserASPP, 0.31% gain using refinement module, and 0.77% gain using both sections. Part 3 compares the effectiveness of using the improved deeplabv3p and Enhancement modules together with the correction module. When using three improved modules at the same time, the gain was 7.92%.

Table 2: We put forward four modules for ablation experiments, respectively is denser ASPP, refinement module Enhancement module (E), calibration module (C).

Denser ASPP	Refinement module	E	C	mIOU
				72.96
		✓		78.78
			✓	74.77
		✓	✓	80.51
✓				73.35
	✓			73.27
✓	✓			73.73
✓	✓	✓		79.97
✓	✓		✓	74.92
✓	✓	✓	✓	80.88

Conclusion

This paper makes some enhancements on the basis of deeplabv3p due to the following two causes: 1) The small defects are difficult to identify, resulting in difficulties in labeling, identification and segmentation. 2) The edges of defects are usually blurry, combined with the influence of noise and contrast, making them easier to mix with the background. We proposed a mobile phone screen defect segmentation based on deeplabv3p consisting of the Enhancement Module, the Calibration Module, and deeplabv3p improving with Denser ASPP and the Refinement Module. Extensive experimental results demonstrate the effectiveness of our method by achieving the highest accuracy compared with state-of-the-art methods on MSDD-3. Based on the inspiring results, we further examine the effectiveness of each component in detail and provide some empirical analysis. In the future, we expect future work to explore more effective methods for mobile phone screen defect segmentation.

References

Badrinarayanan, V.; Kendall, A.; and Cipolla, R. 2017. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Chen, L.-C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; and Yuille, A. L. 2016. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Chen, L.-C.; Papandreou, G.; Schroff, F.; and Adam, H. 2017. Rethinking Atrous Convolution for Semantic Image Segmentation. *arXiv: Computer Vision and Pattern Recognition*.

Chen, L.-C.; Zhu, Y.; Papandreou, G.; Schroff, F.; and Adam, H. 2018. Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. *european conference on computer vision*.

Dai, J.; He, K.; and Sun, J. 2014. Convolutional Feature Masking for Joint Object and Stuff Segmentation. *computer vision and pattern recognition*.

Dong, W.; and Shisheng, Z. 2008. Color Image Recognition Method Based on the Prewitt Operator. *computer science and software engineering*.

Hou, Z.; and Parker, J. 2005. Texture Defect Detection Using Support Vector Machines with Adaptive Gabor Wavelet Features. *workshop on applications of computer vision*.

Huang, Y.; Qiu, C.; and Yuan, K. 2020. Surface defect saliency of magnetic tile. *The Visual Computer*.

Kanopoulos, N.; Vasanthavada, N.; and Baker, R. 1988. Design of an image edge detection filter using the Sobel operator. *IEEE Journal of Solid-state Circuits*.

Li, H.; Xiong, P.; An, J.; and Wang, L. 2018. Pyramid Attention Network for Semantic Segmentation. *british machine vision conference*.

liang Wang, F.; and Zuo, B. 2016. Detection of surface cutting defect on magnet using Fourier image reconstruction. *Journal of Central South University*.

Lin, G.; Shen, C.; van den Hengel, A.; and Reid, I. 2015. Efficient piecewise training of deep structured models for semantic segmentation. *computer vision and pattern recognition*.

Liu, W.; Rabinovich, A.; and Berg, A. C. 2015. ParseNet: Looking Wider to See Better. *arXiv: Computer Vision and Pattern Recognition*.

Long, J.; Shelhamer, E.; and Darrell, T. 2014. Fully Convolutional Networks for Semantic Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Ng, H.-F. 2004. Automatic thresholding for defect detection. *Pattern Recognition Letters*.

Oliveira, H.; and Correia, P. L. 2009. Automatic road crack segmentation using entropy and image dynamic thresholding. *european signal processing conference*.

Qiu, L.; Wu, X.; and Yu, Z. 2019. A High-Efficiency Fully Convolutional Networks for Pixel-Wise Surface Defect Detection. *IEEE Access*.

Ronneberger, O.; Fischer, P.; and Brox, T. 2015. U-Net: Convolutional Networks for Biomedical Image Segmentation. *medical image computing and computer assisted intervention*.

Tabernik, D.; ela, S.; Skvar, J.; and Skoaj, D. 2020. Segmentation-based deep-learning approach for surface-defect detection. *Journal of Intelligent Manufacturing*.

Wang, P.; Chen, P.; Yuan, Y.; Liu, D.; Huang, Z.; Hou, X.; and Cottrell, G. W. 2018. Understanding Convolution for Semantic Segmentation. *workshop on applications of computer vision*.

Xie, Y.; Zhu, F.; and Fu, Y. 2020. Main-Secondary Network for Defect Segmentation of Textured Surface Images. *workshop on applications of computer vision*.

Yang, X.; Qi, D.; and Li, X. 2010. Multi-scale Edge Detection of Wood Defect Images Based on the Dyadic Wavelet Transform. *machine vision and human machine interface*.

Zhao, H.; Shi, J.; Qi, X.; Wang, X.; and Jia, J. 2016. Pyramid Scene Parsing Network. *computer vision and pattern recognition*.