ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 978-1-6654-0540-9/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICASSP43922.2022.9747311

FDSNET: AN ACCURATE REAL-TIME SURFACE DEFECT SEGMENTATION NETWORK

Jian Zhang, Runwei Ding, Miaoju Ban, Tianyu Guo*

Key Laboratory of Machine Perception, Peking University, Shenzhen Graduate School

{zhangjian, miaoju.ban, levigty}@stu.pku.edu.cn, dingrunwei@pku.edu.cn

ABSTRACT

Surface defect detection is a common task for industrial quality control, which increasingly requires accuracy and real-time ability. However, the current segmentation networks are not effective in dealing with defect boundary details, local similarity of different defects and low contrast between defect and background. To this end, we propose a real-time surface defect segmentation network (FDSNet) based on two-branch architecture, in which two corresponding auxiliary tasks are introduced to encode more boundary details and semantic context. To handle the local similarity problem of different surface defects, we propose a Global Context Upsampling (GCU) module by capturing long-range context from multi-scales. Moreover, we present a representative Mobile phone screen Surface Defect (MSD) segmentation dataset to alleviate the lack of dataset in this field. Experiments on NEU-Seg, Magnetic-tile-defect-datasets and MSD dataset show that the proposed FDSNet achieves promising trade-off between accuracy and inference speed. The dataset and code are available at https://github.com/jianzhang96/fdsnet.

***Index Terms*—** Surface defect detection, semantic segmentation, real-time, auxiliary task

1. INTRODUCTION

Surface defect detection aims to detect the abnormal regions in the workpiece surface, which requires high accuracy and real-time ability for many urgent industrial application. However, manual detection is time-consuming and high-cost. Traditional methods based on texture analysis are not universal for different specific tasks. Recently, with the excellent performance of deep learning, some works apply CNN based methods to different industrial applications, such as steel strip [1], magnetic tile [2], rail [3], power line insulator [4], road crack [5] and PCB [6]. These methods are robust for different surface regions and corresponding modules are proposed to specific issues such as small object. Nevertheless, these methods are mostly based on object detection and can not meet the high precision requirements. Thus, the segmentation network is introduced in this work to achieve more fine-grained results.

* Corresponding author. This work is supported by National key R&D program of China (2018YFB1308600, 2018YFB1308602).

Many general [7, 8, 9, 10] and real-time semantic segmentation networks have been proposed for scene parsing datasets. Since existing surface segmentation networks [1, 3] are not fast enough for real-time industrial detection, we focus on real-time segmentation networks. The real-time semantic segmentation methods can be divided into two categories: lightweight backbone [11, 12, 13] and multi-branch architecture [14, 15], including two-branch architecture [16, 17, 18, 19]. We choose the concise and efficient two-branch network Fast-SCNN [18] as our baseline. However, directly using these methods to deal with the surface defect segmentation task can not achieve good results. There are a lot of differences between scene parsing and surface defect segmentation, such as low contrast between defects and background, local similarity of different defects and noise caused by dust and light change. Therefore, our work pays more attention to boundary details, global context and contrast enhancement.

In this work, a Fast surface Defect Segmentation Network (FDSNet) is proposed. Based on two-branch architecture, we propose boundary detection task for detail branch and multi-label classification task for semantic branch in training phase. These auxiliary tasks can encode more boundary details and high-level information of surface defects. The lightweight Global Context Upsampling (GCU) module is proposed to capture the long-range context and segment the surface defects more completely. To improve the contrast of defect and background, we adopt channel attention in the encoder part of the network. In addition, we collect a new Mobile phone screen Surface Defect (MSD) dataset with pixel-level annotations. Our contributions are summarized as follows:

- We propose a real-time and accurate surface defect segmentation network called FDSNet with two auxiliary tasks. Boundary detection and multi-label classification are introduced to encode more edge details and progressively guide defect classification, respectively.
- A lightweight GCU module is proposed to capture long-range dependency and deal with local similarity of different surface defects.
- Extensive experiments on NEU-Seg, Magnetic-tile-defect-datasets and MSD show the effectiveness of our method. On MSD dataset, our FDSNet achieves **90.2%** mIoU with a speed of **135.0** FPS (**275.9** FPS by TensorRT) for 1440×810 images on one 1080Ti GPU.

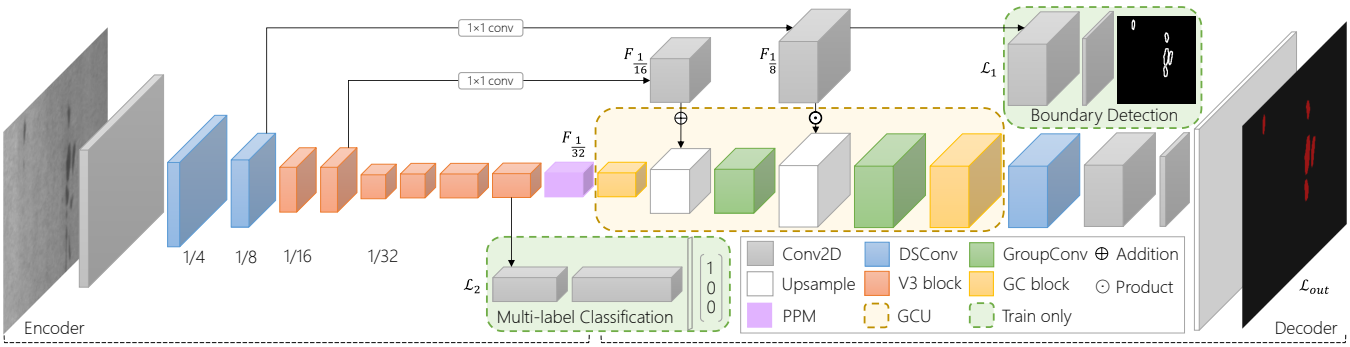


Fig. 1. Overall architecture of the proposed FDSNet. Boundary detection and multi-label classification are introduced in training step as auxiliary tasks. DSConv and PPM denote depthwise separable convolution and pyramid pooling module, respectively.

2. PROPOSED METHOD

As shown in Fig. 1, our method is based on Fast-SCNN [18] and it consists of two parts: downsampling and upsampling, *a.k.a* encoder-decoder architecture. Our FDSNet is single-stream with two auxiliary tasks called boundary detection and multi-label classification. The GCU module upsamples feature maps, captures the long-range context, and utilizes the feature maps from the encoder by skip connection.

2.1. Global Context Upsampling

The feature fusion of different levels with upsampling [20, 17] is a key component of segmentation networks. Bilinear interpolation and transpose convolution are mostly used to upsample feature maps. Our module is shown in Fig. 2 (a). Different from two-branch architecture, we utilize feature maps at 1/8 and 1/16 resolutions by skip connection to maintain more spatial details. The feature maps are gradually upsampled and fused with the low-level information. As shown in Fig. 3 (d), our baseline lacks the ability to establish long-range context. Hence global context (GC) block [21] is inserted at the beginning and the end of the module to capture long-range dependency from multi-scale features. It can handle the inter-class similarity and intra-class difference, and thus the predicted defect region will be more complete. The GC block is a simplified and lightweight self-attention (non-local) block. Due to the high similarity of the attention maps in different query positions, the GC block computes a global attention map for all query positions. It is defined as:

$$z_i = x_i + W_v \sum_{j=1}^{N_p} \frac{\exp(W_k x_j)}{\sum_{m=1}^{N_p} \exp(W_k x_m)} x_j, \qquad (1)$$

where $x = \{x_i\}_{i=1}^{N_p}$ denotes the feature map of the input image, N_p=$H \cdot W$ is the number of positions in the feature map. x and z denote the input and output of the GC block, respectively. W_k and W_v denote linear transformation matrices.

Group convolution is used for faster inference speed. Our module is efficient and lightweight. It can be formulated as:

$$F'_{\frac{1}{16}} = Conv(Up(GC(F_{\frac{1}{32}})) + F_{\frac{1}{16}}),$$
$$F_{out} = GC(Conv(Up(F'_{\frac{1}{16}}) \odot F_{\frac{1}{8}})), \qquad (2)$$

where $Conv$ denotes group convolution followed by BN and ReLU, Up denotes bilinear interpolation and GC denotes GC block. $F_{\frac{1}{8}}, F_{\frac{1}{16}}, F_{\frac{1}{32}}$ are feature maps at different resolution from the encoder stage. $F_{\frac{1}{8}}$ is supervised by boundary detection task and thus contains more spatial details. Element-wise multiplication $\odot$ is adopted to fuse more details.

2.2. Boundary and Semantic Auxiliary Supervision

For two-branch architecture [16, 18], the semantic branch extracts high-level semantic information and the detail branch encodes low-level spatial information. We design boundary detection task for detail branch and multi-label classification task for semantic branch to supervise them to do the specific tasks. The auxiliary tasks are used in the training phase and do not affect the inference speed.

Surface defects mostly have sharp edges and are sensitive to edge features. Inspired by the Detail Guidance module [22], we adopt boundary (edge) detection task for the detail prediction. This binary segmentation task captures useful features of defect boundaries, which can sharpen and refine the prediction. We insert the boundary detection head in the feature map $F_{\frac{1}{8}}$ to generate the feature map contained spatial details. The detail feature map is supervised by the generated ground-truth. Then the learned detail features are fused with context features in decoder by element-wise multiplication.

The local characteristics of different surface defects are similar. Hence we adopt multi-label classification task to progressively guide defect classification, which can provide more semantic information to latter layers. This task is inserted before the pyramid pooling module (PPM) [8] to predict the types of defects contained in the image. When there is only one type of defects in the image, this task can be considered as image classification and carried out well. The task can still provide supporting information in case of many types of defects in the image. The detail head and semantic head of auxiliary tasks are shown in Fig. 2 (b).

As shown in Fig. 2 (c), we generate the auxiliary ground-truth from the semantic segmentation ground-truth. For the binary boundary ground-truth, the Laplacian operator is used to find the edges. Then morphological dilation is adopted to bold the edges because of the fact that the annotations are

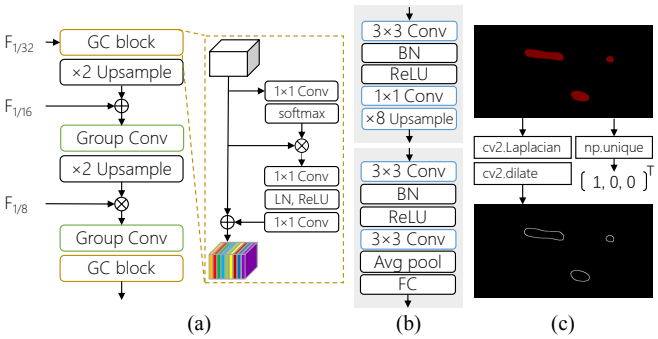


Fig. 2. Details of the proposed components in FDSNet. (a) Global Context Upsampling module. Upsample, LN denote bilinear interpolation and layer normalization. (b) Detail head (upper part) and semantic head (bottom part) of auxiliary tasks. (c) Ground-truth generation of auxiliary tasks.

nowhere near pixel-perfect along the edges [23]. We count the unique values in the ground-truth image and get the class vector ground-truth for the multi-label classification. It can be formulated as:

$$G_B = Dilate(Laplacian(G_{out})),$$
$$G_S = Unique(G_{out}), \tag{3}$$

where $G_{out} \in \mathbb{R}^{H \times W}$ denotes the semantic segmentation ground-truth. G_B, G_S denote the generated boundary and semantic auxiliary task ground-truth.

In the training phase, binary cross-entropy loss is adopted for the auxiliary tasks. Therefore, our method outputs three predictions and the loss function consists of three parts:

$$\mathcal{L} = \mathcal{L}_{out} + w_1 \mathcal{L}_{BCE}(P_B, G_B) + w_2 \mathcal{L}_{BCE}(P_S, G_S),$$
$$\mathcal{L}_{BCE} = -\frac{1}{N} \sum_{i=1}^{N} y_i log(p_i) + (1 - y_i) log(1 - p_i), \tag{4}$$

where $\mathcal{L}_{out}$ is the segmentation loss. P_B, P_S denote the auxiliary prediction. w_1, w_2 denote the weight of the boundary detection and multi-label classification, respectively. FDSNet utilizes auxiliary tasks as a multi-task learning problem.

2.3. Model Acceleration

For surface defect detection, fewer convolution blocks are needed to extract the high-level features, because the defect images are relatively simple. Fast-SCNN adopts nine bottleneck residual blocks for high-level features extraction. we replace them with six blocks introduced by MobileNetV3 [24]. The block uses channel attention (Squeeze-and-Excite), which can improve the contrast between defects and background. TensorRT is also used to accelerate our model for the real-time requirements of industrial applications.

3. EXPERIMENTS

3.1. Datasets

Magnetic-tile-defect-datasets. This dataset [2] contains 392 defect images and 952 images without defects. It includes five types of defects: Uneven, Fray, Crack, Blowhole, and Break. The defect region is cropped from raw images and the resolution of images ranges from 105×283 to 388×516.

NEU-Seg. This dataset [1] contains three kinds of typical surface defects of the hot-rolled steel strip, i.e. Inclusion, Patches and Scratches. The resolution of each image is 200×200 and each class includes 300 images. Due to the original dataset is data-augmented, we convert the SD-saliency-900 [25] into semantic segmentation dataset and use it.

MSD. There are few surface defect datasets for the segmentation task. Mobile phone is widely used and mass-produced, which is a representative industrial product. Thus, we present a mobile phone screen surface defect dataset (Fig. 3). It consists of 1200 images and contains three types of defects: oil, stain and scratch. The defects are made by ourselves and pixel-level labeled by *labelme*. The images are collected by an industrial camera and the resolution is 1920×1080.

3.2. Implementation Details

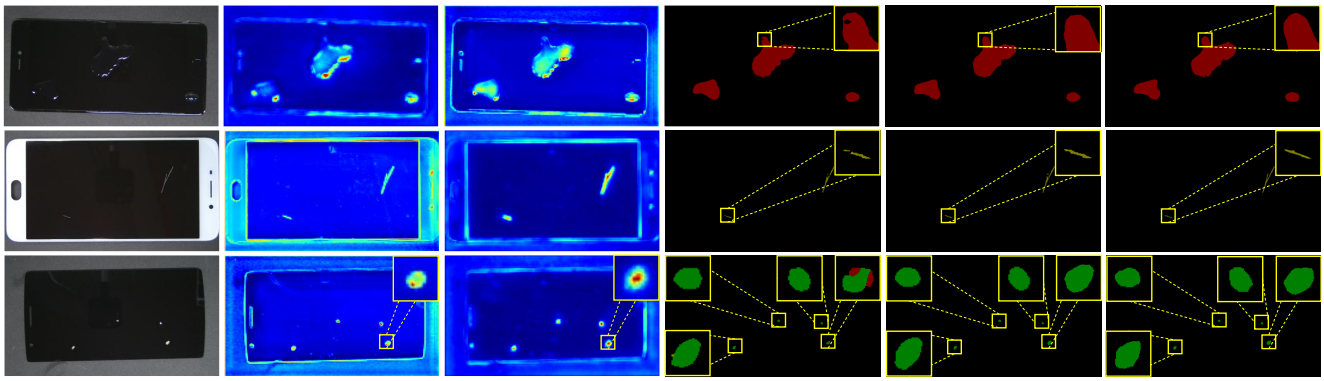
We employ Adam optimizer and a poly learning rate policy where the initial learning rate is multiplied by $(1 - iter/total\_iter)^{0.9}$ after each iteration. The initial learning rate and weight decay coefficients are set to be 10^{-4} and 10^{-5}. For data augmentation, we apply the common scale (0.5 to 2.0), cropping (450×450) and flipping of the image. The batch size is set to 8 for MSD and 16 for other datasets. Three datasets are split to train:val:test=6:2:2 randomly. We train 150 epochs for all experiments. The images of MSD dataset are downsampled to 1440×810 during training and test. The weight of loss function is set to $w_1 = w_2 = 0.5$. Cross-entropy loss is adopted with OHEM [26]. All experiments are conducted by us on one NVIDIA GTX 1080Ti GPU with CUDA 11.1 and PyTorch 1.9.

3.3. Comparisons with State-of-the-arts

We compare our method with general and real-time state-of-the-art semantic segmentation methods on three typical surface defect segmentation datasets. The results of parameters, mIoU (mean Intersection over Union per class) and FPS (frame per second) are reported.

Results on MSD. As shown in Table 1, our method achieves 90.2% in mIoU with a speed of 135.0 FPS, which outperforms all the methods. FDSNet improves Fast-SCNN by **1.1%** in mIoU and **20.0** in FPS. With TensorRT acceleration, FDSNet achieves **275.9** FPS with 88.7% mIoU. Notably, the performance of STDC-Seg is similar to our method, but its inference speed is much lower than ours. FDSNet achieves the best accuracy and inference speed, which demonstrates its effectiveness in surface defect detection.

Results on Magnetic-tile-defect-datasets. we conduct experiments on this challenging dataset. The resolution of the images is different and consequently we report the FPS at average resolution of 320×320. FDSNet achieves **63.9%** mIoU with **181.5** FPS. Our method outperforms all the real-time methods and is also competitive among general methods.



(a) Input (b) Feature Fusion (c) GCU (d) Fast-SCNN (e) FDSNet (f) Ground Truth

Fig. 3. Visualization comparison with Fast-SCNN on MSD test set. Please refer to Section 3.4 for more details.

Results on NEU-Seg. Table 1 shows the comparison results. With the input size 200×200, FDSNet achieves 78.8% mIoU with 186.1 FPS, which improve Fast-SCNN by 1.1% in mIoU and 1.2 in FPS. Our method is the fastest and also achieves a good trade-off between speed and accuracy.

Table 1. Comparisons on test sets of MSD, NEU-Seg and Magnetic-tile-defect-datasets (MT) with the general (upper part) and real-time (bottom part) state-of-the-art methods.

Method	Param (M)	MSD		MT-defect		NEU-Seg	
		mIoU	FPS	mIoU	FPS	mIoU	FPS
FCN-8s [7]	28.63	89.5	15.4	65.9	143.3	79.3	150.5
PSPNet [8]	44.54	90.1	4.2	**74.6**	36.7	82.3	64.4
DeepLabV3+ [9]	56.49	90.0	7.0	29.4	32.4	80.2	37.2
EMANet [10]	51.17	86.5	5.0	66.2	43.5	62.2	75.9
U^2-Net† [27]	1.09	89.2	7.1	45.3	30.8	**84.5**	39.8
DFANet A [13]	1.75	70.1	22.5	18.3	29.8	54.3	30.5
ICNet [14]	25.03	77.4	11.3	63.3	55.0	78.2	57.9
LEDNet [28]	2.21	85.1	23.6	16.3	73.7	81.1	78.4
ENet [11]	0.32	87.0	39.5	33.5	86.0	83.4	86.6
ESPNetV2 [12]	**0.31**	77.7	58.5	44.8	92.3	76.5	95.5
STDC2-Seg [22]	15.3	90.1	47.3	56.8	93.5	70.3	106.0
STDC1-Seg [22]	11.5	90.0	58.9	51.6	132.0	70.0	135.0
BiSeNet [16]	12.20	88.2	50.4	48.6	150.6	81.3	158.9
Fast-SCNN [18]	1.08	89.1	115.0	59.3	179.5	77.7	184.9
FDSNet (Ours)	0.97	**90.2**	**135.0**	63.9	**181.5**	78.8	**186.1**

3.4. Ablation Study

We conduct ablation studies to explore how does each part of FDSNet contribute to the performance gain. Fig. 3 shows the network outputs and visualization of the feature maps after GCU and Feature Fusion module of Fast-SCNN.

Effect of GCU. Table 2 shows that GCU is effective and brings nearly **0.9%** mIoU performance gain. From Fig. 3, we can see the output defects of Fast-SCNN are not continuous. Specifically, there is a hole in the oil and the scratch is broken. Another type of defect mistakenly appears at the edge of the defects. In contrast, the predicted surface defects of FDSNet are complete. Generally, the defect regions are more continuous and complete in our heatmaps. Our method is effective in capturing long-range context and can predict the semantic surface defects into a whole.

Table 2. Ablation studies on MSD test set. M: multi-label classification. B: boundary detection. V3: Section 2.3.

GCU	M	B	V3	mIoU
				89.08
✓				89.95
			✓	89.57
	✓			89.49
	✓	✓		89.96
✓	✓	✓		90.15
✓	✓	✓	✓	90.20

Table 3. Per-class results on the MSD test set. Note that our method improves baseline, especially in stain and scracth.

Class	Fast-SCNN	FDSNet
oil	95.12	95.50
stain	81.90	84.63
scracth	79.39	80.75
mIoU	89.08	90.20

Effect of auxiliary tasks. Without any cost in inference, the auxiliary tasks improve the mIoU by **0.9%**, in which both two tasks are effective. The aforementioned complete outputs and semantic context are also provided by the multi-label classification task. Table 3 shows our method significantly improves baseline in class of stain. Since stain defects on the screen have sharp edges, the boundary detection task can effectively refine the stain boundaries and give better outputs (Fig. 3).

When GCU and auxiliary tasks are adopted simultaneously, the performance gain is not a linear sum of the two parts, because the functions of two parts partly overlap. Notably, reducing the number of convolution blocks does not decrease but improves performance. In addition, the contrast of surface defects and background in the heatmap is higher than baseline, which is gained by the channel attention.

4. CONCLUSION

In this paper, we propose a real-time and accurate surface defect segmentation network called FDSNet and collect a new surface defect segmentation dataset named MSD. To handle local similarity problem and predict surface defects into a whole, a lightweight GCU module is proposed, which can capture long-range context and effectively fuse spatial and semantic information. We further propose boundary detection and multi-label classification auxiliary tasks to guide the network to encode more boundary detail and semantic features of surface defects. Our method achieves competitive speed-accuracy trade-off. Extensive experiments and visualization results show the effectiveness of the proposed method.

5. REFERENCES

[1] Hongwen Dong, Kechen Song, Yu He, Jing Xu, Yunhui Yan, and Qinggang Meng, "PGA-Net: Pyramid feature fusion and global context attention network for automated surface defect detection," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 12, pp. 7448–7458, 2019.

[2] Yibin Huang, Congying Qiu, and Kui Yuan, "Surface defect saliency of magnetic tile," *The Visual Computer*, vol. 36, no. 1, pp. 85–96, 2020.

[3] Defu Zhang, Kechen Song, Jing Xu, Yu He, Menghui Niu, and Yunhui Yan, "MCnet: Multiple context information segmentation network of no-service rail surface defects," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–9, 2020.

[4] Xian Tao, Dapeng Zhang, Zihao Wang, Xilong Liu, Hongyan Zhang, and De Xu, "Detection of power line insulator defects using aerial images analyzed with convolutional neural networks," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 4, pp. 1486–1498, 2018.

[5] Wei Zhai, Jiang Zhu, Yang Cao, and Zengfu Wang, "A generative adversarial network based framework for unsupervised visual surface inspection," in *Proc. ICASSP*, 2018, pp. 1283–1287.

[6] Wei Shi, Zhisheng Lu, Wei Wu, and Hong Liu, "Single-shot detector with enriched semantics for pcb tiny defect detection," *The Journal of Engineering*, vol. 2020, no. 13, pp. 366–372, 2020.

[7] Jonathan Long, Evan Shelhamer, and Trevor Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. CVPR*, 2015, pp. 3431–3440.

[8] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia, "Pyramid scene parsing network," in *Proc. CVPR*, 2017, pp. 2881–2890.

[9] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proc. ECCV*, 2018, pp. 801–818.

[10] Xia Li, Zhisheng Zhong, Jianlong Wu, Yibo Yang, Zhouchen Lin, and Hong Liu, "Expectation-maximization attention networks for semantic segmentation," in *Proc. ICCV*, 2019, pp. 9167–9176.

[11] Adam Paszke, Abhishek Chaurasia, Sangpil Kim, and Eugenio Culurciello, "ENet: A deep neural network architecture for real-time semantic segmentation," *arXiv preprint arXiv:1606.02147*, 2016.

[12] Sachin Mehta, Mohammad Rastegari, Linda Shapiro, and Hannaneh Hajishirzi, "ESPNetV2: A light-weight, power efficient, and general purpose convolutional neural network," in *Proc. CVPR*, 2019, pp. 9190–9200.

[13] Hanchao Li, Pengfei Xiong, Haoqiang Fan, and Jian Sun, "DFANet: Deep feature aggregation for real-time semantic segmentation," in *Proc. CVPR*, 2019, pp. 9522–9531.

[14] Hengshuang Zhao, Xiaojuan Qi, Xiaoyong Shen, Jianping Shi, and Jiaya Jia, "ICNet for real-time semantic segmentation on high-resolution images," in *Proc. ECCV*, 2018, pp. 405–420.

[15] Marin Orsic, Ivan Kreso, Petra Bevandic, and Sinisa Segvic, "In defense of pre-trained imagenet architectures for real-time semantic segmentation of road-driving images," in *Proc. CVPR*, 2019, pp. 12607–12616.

[16] Changqian Yu, Jingbo Wang, Chao Peng, Changxin Gao, Gang Yu, and Nong Sang, "BiSeNet: Bilateral segmentation network for real-time semantic segmentation," in *Proc. ECCV*, 2018, pp. 325–341.

[17] Changqian Yu, Changxin Gao, Jingbo Wang, Gang Yu, Chunhua Shen, and Nong Sang, "BiSeNet V2: Bilateral network with guided aggregation for real-time semantic segmentation," *arXiv preprint arXiv:2004.02147*, 2020.

[18] Rudra P. K. Poudel, Stephan Liwicki, and Roberto Cipolla, "Fast-SCNN: Fast semantic segmentation network," in *Proc. BMVC*, 2019, p. 289.

[19] Wuyang Chen, Xinyu Gong, Xianming Liu, Qian Zhang, Yuan Li, and Zhangyang Wang, "Fasterseg: Searching for faster real-time semantic segmentation," in *Proc. ICLR*, 2020.

[20] Xiangtai Li, Ansheng You, Zhen Zhu, Houlong Zhao, Maoke Yang, Kuiyuan Yang, Shaohua Tan, and Yunhai Tong, "Semantic flow for fast and accurate scene parsing," in *Proc. ECCV*, 2020, pp. 775–793.

[21] Yue Cao, Jiarui Xu, Stephen Lin, Fangyun Wei, and Han Hu, "GCNet: Non-local networks meet squeeze-excitation networks and beyond," in *Proc. ICCVW*, 2019, pp. 1971–1980.

[22] Mingyuan Fan, Shenqi Lai, Junshi Huang, Xiaoming Wei, Zhenhua Chai, Junfeng Luo, and Xiaolin Wei, "Rethinking bisenet for real-time semantic segmentation," in *Proc. CVPR*, 2021, pp. 9716–9725.

[23] Yi Zhu, Karan Sapra, Fitsum A Reda, Kevin J Shih, Shawn Newsam, Andrew Tao, and Bryan Catanzaro, "Improving semantic segmentation via video propagation and label relaxation," in *Proc. CVPR*, 2019, pp. 8856–8865.

[24] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, et al., "Searching for mobilenetv3," in *Proc. ICCV*, 2019, pp. 1314–1324.

[25] Guorong Song, Kechen Song, and Yunhui Yan, "Saliency detection for strip steel surface defects using multiple constraints and improved texture features," *Optics and Lasers in Engineering*, vol. 128, pp. 106000, 2020.

[26] Abhinav Shrivastava, Abhinav Gupta, and Ross Girshick, "Training region-based object detectors with online hard example mining," in *Proc. CVPR*, 2016, pp. 761–769.

[27] Xuebin Qin, Zichen Zhang, Chenyang Huang, Masood Dehghan, Osmar R Zaiane, and Martin Jagersand, "U^2-Net: Going deeper with nested u-structure for salient," *Pattern Recognition*, vol. 106, pp. 107404, 2020.

[28] Yu Wang, Quan Zhou, Jia Liu, Jian Xiong, Guangwei Gao, Xiaofu Wu, and Longin Jan Latecki, "LEDNet: A lightweight encoder-decoder network for real-time semantic segmentation," in *Proc. ICIP*, 2019, pp. 1860–1864.