2020 25th International Conference on Pattern Recognition (ICPR)
Milan, Italy, Jan 10-15, 2021

EDD-Net: An Efficient Defect Detection Network

Tianyu Guo, Linlin Zhang, Runwei Ding, Ge Yang
Key Laboratory of Machine Perception
Shenzhen Graduate School, Peking University
Shenzhen, China
Email: levigty@stu.pku.edu.cn, catherinezll@pku.edu.cn, dingrunwei@pku.edu.cn, yangge@pkusz.edu.cn

***Abstract*—As the most commonly used communication tool, the mobile phone has become an indispensable part of our daily life. The surface of the mobile phone as the main window of human-phone interaction directly affects the user experience. It is necessary to detect surface defects on the production line in order to ensure the high quality of the mobile phone. However, the existing mobile phone surface defect detection is mainly done manually. Currently, there are few automatic defect detection methods to replace human eyes. How to quickly and accurately detect the surface defects of the mobile phone is an urgent problem to be solved. Hence, an efficient defect detection network (EDD-Net) is proposed. Firstly, EfficientNet is used as the backbone network. Then, according to the small-scale of mobile phone surface defects, a feature pyramid module named GCSA-BiFPN is proposed to obtain more discriminative features. Finally, the box/class prediction network is used to achieve effective defect detection. We also build a mobile phone surface oil stain defect (MPSOSD) dataset to alleviate the lack of dataset in this field. The performance on the relevant datasets shows that the proposed network is effective and has practical significance for industrial production.**

***Index Terms*—surface defect detection, deep learning, GCSA-BiFPN, EDD-Net**

I. Introduction

Nowadays, mobile phones affect every aspect of our lives. People's dependence on mobile phones is increasing day by day, which also puts forward higher requirements on the quality of mobile phones. However, various types of mobile phone surface defects inevitably exist on the production line, mainly including scratches, pits, oil stains, etc. Thus, effective detection of mobile phone surface defects on the production line is of great significance for quality control. Early detection of surface defects mainly relied on manual visual inspection, which had many limitations such as poor accuracy, strong subjectivity, slow speed, high cost, and so on. To overcome the limitations of manual visual inspection, machine vision detection methods appear recently to replace the human eye for measurement and judgment.

The existing machine vision detection methods are mainly divided into traditional machine vision detection methods and deep learning based detection methods. Early traditional machine vision detection methods were based on traditional image processing. These methods used manual features and simple machine learning methods, focusing on how to highlight defect features through preprocessing [1]–[5]. Generally, a lot of preprocessing operations were required before defect detection, such as edge detection to determine the location of the mobile phone screen area, geometric correction to keep the target area horizontal, color space conversion, binarization, etc. However, the size, the shape, and the type of surface defects of mobile phones vary differently on account of the different image acquisition environment. It is difficult to adopt a unified preprocessing method. With the excellent performance of the two-stage object detectors [6]–[9] and the one-stage object detectors [10]–[14] in the field of object detection in recent years, researchers have begun to use deep learning to deal with defect detection. However, the current deep learning methods for mobile phone surface defect detection rarely pay attention to the difference between defect detection and object detection such as background difference, scale difference, low contrast, and so on. Directly using the advanced object detection method to deal with the defect detection task may not be able to achieve good results. More attention should be paid to the small-scale challenge and the context information around the defects.

Thus, to address these issues, a defect dataset and an efficient mobile phone surface defect detection network named EDD-Net are proposed. As shown in Fig. 1, the input image first passes through a lightweight backbone network - EfficientNet [15] to extract features. Then, the GCSA-BiFPN which consists of the bidirectional feature pyramid module, global context module, and spatial attention module is proposed to obtain more discriminative features. Finally, a box/class prediction network is used to achieve defect detection. To summarize, the main contributions are three-fold:

- For the small-scale and low contrast of the defects, we propose an improved feature pyramid module GCSA-BiFPN, which pays attention to the discriminative context and spatial information, greatly improving the performance.
- Based on the advanced object detector EfficientDet, our EDD-Net D0-D2 are proposed. The three networks achieve both high accuracy and better efficiency, which are of great significance to practical application scenarios across a wide spectrum of resource constraints.
- We develop a dataset named MPSOSD dataset for mobile phone surface defect detection, which provides research-worthy images for this field.

II. Related Work

In this section, we provide a brief overview of existing traditional machine vision detection methods and the deep

978-1-7281-8808-9/20/$31.00 ©2020 IEEE

2020 25th International Conference on Pattern Recognition (ICPR) | 978-1-7281-8808-9/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICPR48806.2021.9412300

Authorized licensed use limited to: University Town Library of Shenzhen. Downloaded on May 30,2021 at 16:35:28 UTC from IEEE Xplore. Restrictions apply.

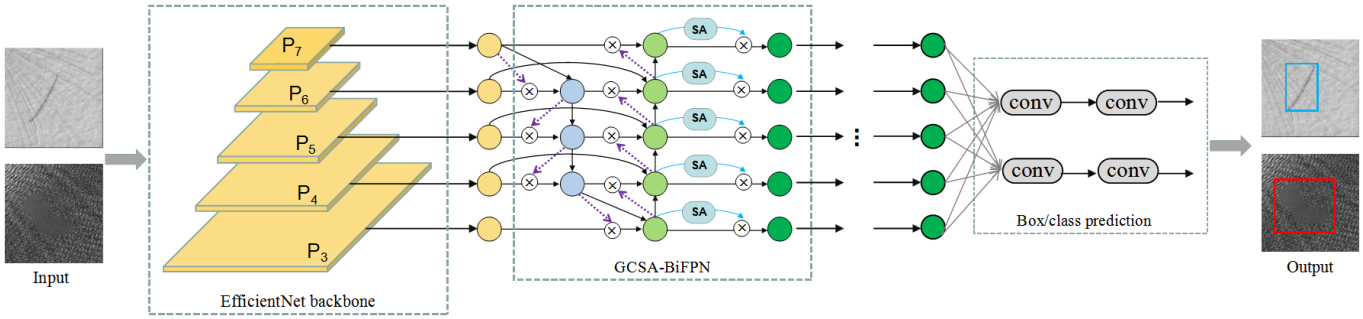

Fig. 1. The network architecture of EDD-Net. It employs EfficientNet [15] as the backbone network, GCSA-BiFPN as the feature network, and uses box/class prediction network to obtain the detection results.

learning based detection methods.

A. Traditional Machine Vision Detection Methods

The traditional machine vision detection methods focus on the image preprocessing to achieve defect detection, mainly through a series of operations such as morphological processing, edge detection, feature extraction, and so on.

Liu et al. [1] proposed an inline defect detection system. The system divides pixels of the input image in the image preprocessing stage, extracts the gray-scale statistical features of the pixels, and then uses the proposed Support Vector Data Description (SVDD) classifier to classify the defects. Subsequently, QK-SVDD [2] was proposed to significantly improve the generalization performance of traditional SVDD. Li et al. [3] constructed an eigen-defect matrix to characterize the variation between defect images. Liang et al. [4] transferred the defect detection problem to that if an image can be sparsely represented under the redundancy dictionary or not and proposed sparsity ratio of the sparse representation coefficients to determinate whether the image is defective or not. Jian et al. [5] proposed the contour-based registration method to align the images. Then the combination of subtraction and projection was used to identify defects.

For traditional methods, it is necessary to preprocess the collected images before defect detection. The preprocessing operations are too cumbersome and inappropriate preprocessing operations in cross-scenarios seriously affect the final detection result. For the actual application scene with noisy images, the traditional machine vision detection methods have poor robustness and are far from reaching the industrial demand.

B. Deep Learning based Detection Methods

Compared with traditional machine vision detection methods that tend to only work well under specified conditions, deep learning based methods have strong robustness to a certain extent and do not require excessive preprocessing. In addition, the ability of deep learning for object detection has proven to exceed traditional machine learning methods in major international evaluations [16]. Hence, deep learning based methods have received growing attention now.

Wang et al. [17] used the sliding window to sample on the original image and designed deep convolutional neural networks to automatically extract powerful features with less prior knowledge about the images for defect detection. Yang et al. [18] used the gray level co-occurrence matrix (GLCM) to calculate defect features. Then fed them into the neural network to train the defect classifier. Ma et al. [19] designed a CNN on the basis of GoogLeNet [20], which can be combined with the sliding window to detect the defect areas. Lei et al. [21] proposed an end-to-end mobile phone screen defect detection framework, which was composed of a scale insensitive MSDDN and a self-comparison driven SCN. Zhao et al. [22] proposed to establish a GAN to repair defect areas in the samples, and then make a comparison between the input sample and the restored one to indicate the accurate defect areas. Some researchers directly used object detectors (i.e., R-CNN methods [6]–[8], Yolo [10], SSD [13], RetinaNet [14], etc) to detect defects [23], [24] and achieved good performance.

The deep learning based methods omit the tedious preprocessing, but can also detect defects well and have strong robustness. Obviously, the defect detection task and the object detection task are similar. However, existing deep learning based methods rarely pay attention to their differences, such as scale difference, context difference, and contrast difference.

III. Our Method

In this section, we first introduce the overall architecture of our proposed Efficient Defect Detection Network (EDD-Net). Then the novel modules designed for defect detection task are described in detail.

A. Network Architecture

As is known to us all, the defect detection task is based on actual application scenarios and has high requirements for scene dependence and real-time performance. In recent years, many detectors have achieved higher accuracy on the MS COCO dataset [25], and at the same time have become more costly. There is also a lot of work aimed at developing more effective detectors, such as one-stage detectors and anchor-free detectors. Although these detectors tend to achieve higher efficiency, they usually sacrifice accuracy. Tan et al. [12]

Authorized licensed use limited to: University Town Library of Shenzhen. Downloaded on May 30,2021 at 16:35:28 UTC from IEEE Xplore. Restrictions apply.

TABLE I
CONFIGS FOR EDD-NET D0-D2

	Input size	Backbone	GCSA-BiFPN		Box/class layers
			channels	layers	
EDD-Net D0	512 × 512	EfficientNet-B0	64	3	3
EDD-Net D1	640 × 640	EfficientNet-B1	88	4	3
EDD-Net D2	768 × 768	EfficientNet-B2	112	5	3

built a scalable detection architecture EfficientDet D0-D7 with high accuracy and efficiency within a wide range of resource constraints, making the practical application of the detector in different scenarios possible.

Actually, EDD-Net D0-D2 are under the framework of EfficientDet, which can easily achieve real-time performance while the accuracy is considerable. The overall network architecture of EDD-Net is shown in Fig. 1. The EfficientNet [15] is used as the backbone network and the proposed GCSA-BiFPN is applied as the feature network. Finally, we employ the box/class prediction network to obtain the detection results. Both GCSA-BiFPN and box/class prediction layers are repeated multiple times and the detailed network architecture parameters are shown in Table I. More details about the core modules in EDD-Net are introduced in the following.

B. GCSA-BiFPN

Since its introduction, the top-down structure of FPN [26] has been widely used for multi-scale feature fusion. In recent years, some other work is also exploring more effective cross-scale feature fusion. PANet [27] added an additional bottom-up pathway aggregation network on the basis of FPN. M2Det [28] proposed a U-shaped module to fuse multi-scale features. NAS-FPN [29] used neural architecture search to automatically design a feature network. Although it has better performance, it requires thousands of GPUs during the search process. And the resulting feature network is irregular which is difficult to explain. EfficientDet [12] proposed BiFPN to optimize multi-scale feature fusion in a more intuitive way. However, no matter what kind of pyramid fusion method, it rarely used the guided context information during fusion and the spatial information of the feature map is not fully excavated. Hence, Global Context and Spatial Attention Bidirectional Feature Pyramid Network (GCSA-BiFPN) which consists of Bidirectional Feature Pyramid (BiFPN) module, Global Context (GC) module, and Spatial Attention (SA) module is proposed to obtain more discriminative features.

Global Context (GC) module: Studies [30] have shown that high-level features with rich semantic information can be used to weight low-level features to select precise resolution details in segmentation task. The same idea is applied to the BiFPN structure. And in BiFPN, the global context information can be obtained by the following:

$$GC(F) = \sigma(AvgPool(F)) \tag{1}$$

where σ represents the sigmoid function and $AvgPool()$ represents the global average pooling. As shown in Fig. 2,

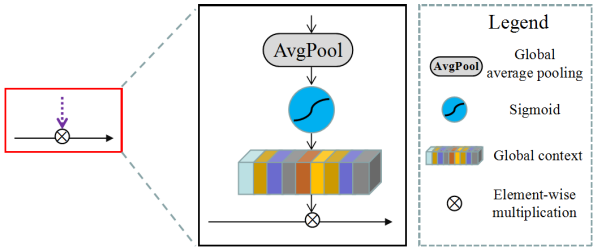

Fig. 2. Details of the Global Context (GC) module. The GC module use global average pooling to generate global context information, and generate weighted vectors after the sigmoid function to guide the feature generation of the next level.

in the top-down pathway, high-level features perform global average pooling to generate global context information, which can guide the generation of low-level feature maps through the sigmoid function. Correspondingly, in the bottom-up pathway, the low-level features perform global average pooling to generate global context information to guide the generation of high-level feature maps through the sigmoid function. The two processes complement each other and feature maps can be generated more efficiently. Notably, only a small amount of calculation is introduced, but it makes feature pyramid generation more instructive.

Spatial Attention (SA) module: Most of the traditional attention modules are applied after the convolutional layer. However, more attention should be paid to the spatial information of the generated feature pyramid in the defect detection task. Thus, the spatial attention module is connected after each pyramid layer. The spatial attention can be obtained by the following equation:

$$SA(F) = \sigma(f^{7\times7}([AvgPool(F), MaxPool(F)])) \tag{2}$$

where σ denotes the sigmoid function, $AvgPool()$ and $MaxPool()$ represent global average pooling and global max pooling respectively. As shown in Fig. 3, the SA module first combines the spatial information generated by channel-based global average pooling and channel-based global max pooling. Then, after convolution with kernel size 7 and sigmoid function, the weighted map is generated to optimize the original feature representation. In this way, more spatially discriminative information of each pyramid layer is obtained to seek better performance.

Based on the BiFPN module, GC module, and SA module, the GCSA-BiFPN is built as shown in Fig. 4. A list of multi-scale features $\vec{P}^{in} = (P_3^{in}, ..., P_7^{in})$ is obtained after backbone network, where P_i^{in} represents a feature level with resolution of $1/2^i$ of the input image. As a specific example, here we describe the fused features at level 6 for GCSA-BiFPN:

$$P_6^{fi} = Conv(\frac{GC(P_7^{in}) \otimes P_6^{in} + Resize(P_7^{in})}{2}) \tag{3}$$

$$P_6^{si} = Conv(\frac{P_6^{in} + GC(P_5^{si}) \otimes P_6^{fi} + Resize(P_5^{si})}{3}) \tag{4}$$

Authorized licensed use limited to: University Town Library of Shenzhen. Downloaded on May 30,2021 at 16:35:28 UTC from IEEE Xplore. Restrictions apply.

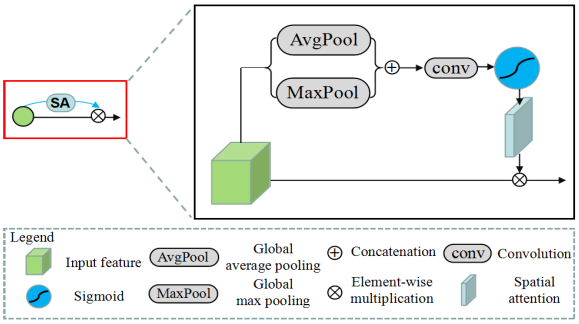

Fig. 3. Details of the Spatial Attention (SA) module. The SA module first combines the spatial information generated by global average pooling and global max pooling. Then the spatial attention is generated after convolution and sigmoid function to optimize the original feature representation.

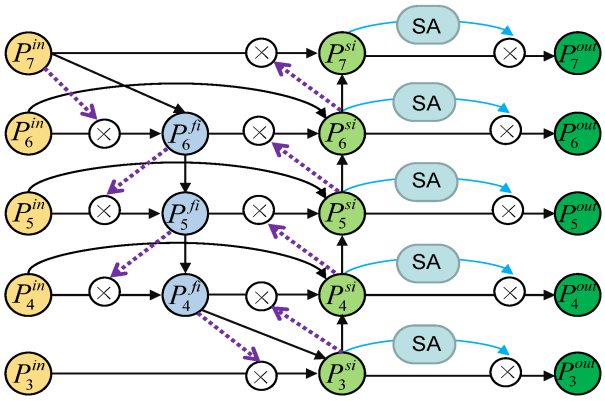

Fig. 4. Network structure of our GCSA-BiFPN. Based on the BiFPN, GC, and SA module, the GCSA-BiFPN is designed with better accuracy and efficiency trade-offs.

$$P_6^{out} = SA(P_6^{si}) \otimes P_6^{si} \tag{5}$$

where $\otimes$ denotes element-wise multiplication, the attention values are broadcasted during multiplication. P_6^{in} is the input feature map, P_6^{fi} is the first intermediate feature map of level 6 in the top-down pathway, and P_6^{si} is the second intermediate feature map of level 6 in the bottom-up pathway. P_6^{out} represents the final output feature map of level 6 for a GCSA-BiFPN module. All the feature maps of other levels are constructed in a similar manner. In order to further improve efficiency, we use deep separable convolutions [31] and add batch normalization and activation functions after each convolution layer.

IV. Experiments

In this section, we first introduce the datasets and experiment setup, then we compare our method with state-of-the-art methods. Finally, the ablation studies are conducted to verify the effectiveness of each component in our method.

A. Datasets

MPSOSD Dataset (Mobile Phone Surface Oil Stain Defect Dataset) is our collected the real mobile phone surface defect dataset. It contains a total of original 750 defect images with 958 defect targets collected from two different mobile phones. We flip the original images horizontally and vertically for data augmentation and finally obtained 3000 defect images. The size of each image is about 180×570. In order to increase the difficulty of the defect detection task, the evaluation protocol is a cross mobile phone evaluation protocol. That is, there are two kinds of mobile phones in the dataset, A and B. The images collected from mobile phone A are all used for training, and the images collected from mobile phone B are all used for testing. Finally, we divided 1700 images as the training set and 1300 images as the testing set.

DAGM2007 Dataset [32] is a competition dataset provided by the German Chapter of the European Neural Network Society. The dataset consists of 10 classes of the defect in total, including 2100 defect images. Although the defect data is artificially generated, it is similar to the problem in the real world. Many types of defects are very similar to the common defect types of mobile phones. The annotations provided in the original dataset is an ellipse around the defect, and we convert the ellipse annotations to the bounding box annotations. For defect images, we divided the training set and the testing set following the original evaluation protocol. Finally, we used 1046 defect images for training and 1054 defect images for testing.

B. Implementation Details

The implementation of our proposed method is based on the deep learning framework PyTorch with two NVIDIA GeForce GTX 1080 GPUs and 8G RAM. In order to ensure the experimental results, our experiment uses EfficientDet's pre-trained weights on the COCO dataset for fine-tuning. Each model is trained using Adam optimizer with learning rate 1e-4. We use swish activation [33], [34] and also employ commonly-used focal loss [14] with α=0.25 and γ=2.0. The aspect ratio of the anchors is $[1/2, 1, 2]$. Notably, we do not use auto-augmentation for any of our models.

C. Comparison with State-of-the-Arts

In this section, we compare the accuracy and efficiency of our method with current advanced object detection methods on the two datasets.

Fig. 5 shows the results of our method for detecting oil stains on the MPSOSD dataset. It can be intuitively seen that oil stain defects can be well detected by our EDD-Net. Excitingly, the low-contrast oil stain (i.e., the upper right one) which is even difficult for human eye to distinguish can also be well detected by the EDD-Net.

As shown in Table II, we group models together if they have similar accuracy on the MPSOSD dataset, and compare the performance between our EDD-Net and other detectors in each group. #Params and #FLOPs denote the number of parameters and the number of multiply-adds. Notably, our EDD-Net achieves better accuracy and efficiency than previous detectors on the MPSOSD dataset. The great thing is that our EDD-Net D0 achieves similar accuracy to RetinaNet-ResNet50 [14] with

Authorized licensed use limited to: University Town Library of Shenzhen. Downloaded on May 30,2021 at 16:35:28 UTC from IEEE Xplore. Restrictions apply.

TABLE II
PERFORMANCE ON MPSOSD DATASET

Methods	Backbone	AP_{50}	AP_{75}	#Params	#FLOPs
Yolo v3 [11]	Darknet-53	85.14	42.80	61.52M	32.76B
Cascade R-CNN [9] w/FPN [26]	ResNet-50	84.93	83.92	69.70M	707.66B
Cascade R-CNN [9] w/FPN [26]	ResNet-101	85.83	78.29	88.64M	897.06B
EfficientDet-D0 [12]	EfficientNet-B0	90.20	**87.70**	**3.83M**	**2.29B**
RetinaNet [14]	ResNet-50	**93.30**	82.90	36.33M	74.17B
EDD-Net D0(ours)	**EfficientNet-B0**	**94.10**	**85.20**	**3.83M**	**2.30B**
RetinaNet [14]	ResNet-101	94.70	92.30	55.32M	101.75B
EfficientDet-D1 [12]	EfficientNet-B1	95.30	**94.10**	**6.56M**	**5.58B**
Faster R-CNN [8]	ResNet-50	95.63	55.02	27.99M	155.39B
FPN [26]	ResNet-101	95.69	63.56	60.24M	432.37B
FPN [26]	ResNet-50	96.24	60.74	41.30M	299.79B
Faster R-CNN [8]	ResNet-101	**97.36**	61.64	46.94M	250.04B
EDD-Net D1(ours)	**EfficientNet-B1**	**98.40**	**94.70**	**6.56M**	**5.59B**
EfficientDet-D2 [12]	EfficientNet-B2	**98.40**	**97.60**	**8.01M**	**10.02B**
EDD-Net D2(ours)	**EfficientNet-B2**	**99.50**	**91.90**	**8.01M**	**10.03B**

TABLE III
PERFORMANCE ON DAGM2007 DATASET

Methods	Backbone	AP_{50}	AP_{75}
Two-stage:			
Faster R-CNN [8]	ResNet-50	92.71	26.22
Faster R-CNN [8]	ResNet-101	92.89	43.08
FPN [26]	ResNet-50	96.99	62.85
FPN [26]	ResNet-101	96.85	62.03
Cascade R-CNN [9] w/FPN [26]	ResNet-50	96.96	71.91
Cascade R-CNN [9] w/FPN [26]	**ResNet-101**	**97.44**	**74.45**
One-stage:			
Yolo v3 [11]	Darknet-53	87.81	25.97
RetinaNet [14]	ResNet-50	97.10	61.40
RetinaNet [14]	ResNet-101	95.40	64.50
EDD-Net D0(ours)	EfficientNet-B0	95.40	56.40
EDD-Net D1(ours)	**EfficientNet-B1**	**97.10**	**71.20**
EDD-Net D2(ours)	EfficientNet-B2	96.00	66.50

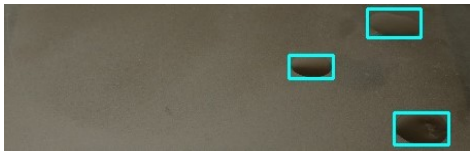

Fig. 5. Detection results for oil stain on MPSOSD dataset.

$32\times$ fewer FLOPs. Interestingly, the complex detectors such as Cascade R-CNN [9] with FPN [26] did not achieve good performance on the MPSOSD dataset. The reason may be that our evaluation protocol on the MPSOSD dataset is challenging, and complex detectors may not be able to pay attention to the difference between defect detection task and object detection task. Compared with Faster R-CNN [8] and FPN [26], our EDD-Net D1 achieves similar accuracy with fewer parameters and fewer FLOPs. On high-accuracy regime, our EDD-Net D2 also consistently outperforms EfficientDet-D2 [12] with almost the same amount of parameters and FLOPs, as it combines context information and spatial information, which are helpful for detecting defects.

Table III compares the performance of our EDD-Net with existing one-stage and two-stage detectors on the DAGM2007 dataset. For the two-stage detectors, FPN [26] introduces feature pyramid to improve accuracy compared to Faster RCNN [8]. Cascade R-CNN [9] with FPN [26] cascades FPN to further improve the performance and both AP_{50} and AP_{75} are the best in the two-stage detectors. For the one-stage detectors, our EDD-Net performs much better than Yolo v3 [11]. EDD-Net D0 achieves the same AP_{50} with RetinaNet-ResNet101 [14]. And our EDD-Net D1 achieves the best performance in AP_{50} and AP_{75} compared with other one-stage detectors. Interestingly, EDD-Net D2 does not perform as well as EDD-Net D1, which probably because the structure of the EDD-Net D1 is more suitable for the DAGM2007 dataset. In general, our detectors EDD-Net D0-D2 always perform well in one-stage detectors.

TABLE IV
THE EFFECTIVENESS OF OUR PROPOSED MODULES

Module	Backbone								
	EfficientNet-B0			EfficientNet-B1			EfficientNet-B2		
WFF*	✓	✗	✗	✓	✗	✗	✓	✗	✗
GC		✓	✓		✓	✓		✓	✓
SA			✓			✓			✓
AP_{50}	90.2	92.6	**94.1**	95.3	97.3	**98.4**	98.4	98.8	**99.5**

*Weighted Feature Fusion.

D. Ablation Studies

In Section III-B, we propose the GCSA-BiFPN which consists of BiFPN module, GC module, and SA module to improve the accuracy of defect detection. Hence, ablation studies are provided to comprehensively evaluate the modules in this section.

As can be seen from Table IV, EfficientDet with EfficientNet B0-B2 uses weighted feature fusion while our final model cancels weighted feature fusion and introduces GC and SA modules. It can be seen intuitively that our GC module and SA module have brought AP_{50} improvements for three different backbones. It is worth noting that with the introduction of a small amount of additional computation, our final complete network is improved by 3.9%, 3.1%, and 1.1% compared to

Authorized licensed use limited to: University Town Library of Shenzhen. Downloaded on May 30,2021 at 16:35:28 UTC from IEEE Xplore. Restrictions apply.

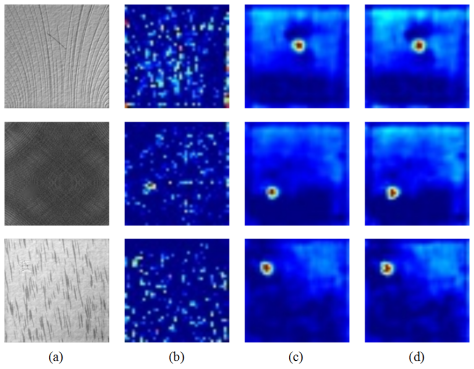

Fig. 6. Visual explanation of key information learned after different modules: (a)original images; (b)Feature map of P_4^{in} after EfficientNet; (c)Feature map of P_4^{si} after GC module; (d)Feature map of P_4^{out} after SA module.

the baseline network using weighted feature fusion. It proves that our EDD-Net is more suitable for the defect detection task.

What's more, in order to have a more intuitive explanation about the contribution of each module in EDD-Net, Fig. 6 shows the feature maps after different modules on the DAGM2007 dataset. As shown in Fig. 6, it is clear that our modules can better focus on the information of the defects. This also validates that our GCSA-BiFPN not only incorporate multi-scale features but also pay attention to the important contextual and spatial information.

V. Conclusion

In this paper, an efficient defect detection network EDD-Net is proposed and a defect dataset is built. Focusing on the difference between defect detection task and the object detection task, EDD-Net has three novel aspects to adapt to defect detection compared with ordinary detectors. Firstly, EDD-Net is under the framework of advanced object detector EfficientDet which guarantees the detection results. Secondly, a novel feature pyramid module GCSA-BiFPN is proposed to fully use the context information and spatial information. Thirdly, our EDD-Net D0-D2 can easily achieve real-time performance while the accuracy is considerable, which has practical significance in different scenarios on the production line. In the future, we will focus on proposing a more lightweight network to further improve efficiency while maintaining accuracy for the defect detection task.

Acknowledgment

This work is supported by National Key R&D Program of China (2018YFB1308600, 2018YFB1308602).

References

[1] Y. Liu, Y. Huang, and M. Lee, "Automatic inline defect detection for a thin film transistor–liquid crystal display array process using locally linear embedding and support vector data description," *Measurement Science and Technology*, vol. 19, no. 9, 2008.

[2] Y. Liu and Y. Chen, "Automatic defect detection for TFT-LCD array process using quasiconformal kernel support vector data description," *International Journal of Molecular Sciences*, vol. 12, no. 9, pp. 5762–5781, 2011.

[3] D. Li, L. Liang, and W. Zhang, "Defect inspection and extraction of the mobile phone cover glass based on the principal components analysis," *The International Journal of Advanced Manufacturing Technology*, vol. 73, no. 9, pp. 1605–1614, 2014.

[4] L. Liang, D. Li, X. Fu, and W. Zhang, "Touch screen defect inspection based on sparse representation in low resolution images," *Multimedia Tools and Applications*, vol. 75, no. 5, pp. 2655–2666, 2016.

[5] C. Jian, J. Gao, and Y. Ao, "Automatic surface defect detection for mobile phone screen glass based on machine vision," *Applied Soft Computing*, vol. 52, 2016.

[6] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recogintion(CVPR)*, 2014, pp. 580–587.

[7] R. Girshick, "Fast R-CNN," in *IEEE Conference on Computer Vision and Pattern Recogintion(CVPR)*, 2015, pp. 1440–1448.

[8] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Advances in Neural Information Processing Systems(NIPS)*, 2015, pp. 91–99.

[9] Z. Cai and N. Vasconcelos, "Cascade R-CNN: Delving into high quality object detection," in *IEEE Conference on Computer Vision and Pattern Recogintion(CVPR)*, 2018, pp. 6154–6162.

[10] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *IEEE Conference on Computer Vision and Pattern Recogintion(CVPR)*, 2016, pp. 21–37.

[11] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.

[12] M. Tan, R. Pang, and Q. V. Le, "EfficientDet: Scalable and efficient object detection," in *IEEE Conference on Computer Vision and Pattern Recogintion(CVPR)*, 2020.

[13] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot multibox detector," in *European Conference on Computer Vision(ECCV)*, 2016, pp. 21–37.

[14] T. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," in *IEEE International Conference on Computer Vision(ICCV)*, 2017, pp. 2999–3007.

[15] M. Tan and Q. V. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *International Conference on Machine Learning (ICML)*, 2019, pp. 6105–6114.

[16] N. Tajbakhsh and K. Suzuki, "Comparing two classes of end-to-end machine-learning models in lung nodule detection and classification: MTANNs vs. CNNs," *Pattern Recogintion*, vol. 63, 2016.

[17] T. Wang, Y. Chen, M. Qiao, and H. Snoussi, "A fast and robust convolutional neural network-based defect detection model in product quality control," *The International Journal of Advanced Manufacturing Technology*, vol. 94, pp. 3465–3471, 2018.

[18] S. Yang, C. Lin, S. Lin, and H. Chiang, "Automatic defect recognition of TFT array process using gray level co-occurrence matrix," *Optik*, vol. 125, no. 11, pp. 2671–2676, 2014.

[19] L. Ma, Y. Lu, X. Nan, Y. Liu, and H. Jiang, "Defect detection of mobile phone surface based on convolution neural network," *DEStech Transactions on Computer Science and Engineering*, 2017.

[20] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *IEEE Conference on Computer Vision and Pattern Recogintion(CVPR)*, 2015, pp. 1–5.

[21] J. Lei, X. Gao, Z. Feng, H. Qiu, and M. Song, "Scale insensitive and focus driven mobile screen defect detection in industry," *Neurocomputing*, vol. 294, pp. 72–81, 2018.

[22] Z. Zhao, B. Li, R. Dong, and P. Zhao, "A surface defect detection method based on positive samples," in *Pacific Rim International Conference on Artificial Intelligence*, 2018, pp. 473–481.

[23] R. Ding, L. Dai, G. Li, and H. Liu, "TDD-Net: A tiny defect detection network for printed circuit boards," *CAAI Transactions on Intelligence Technology*, vol. 4, no. 2, pp. 110–116, 2019.

[24] W. Shi, Z. Lu, W. Wu, and H. Liu, "A single-shot detector with enriched semantics for PCB tiny defect detection," in *The 3rd Asian Conference on Artificial Intelligence Technology (ACAIT)*, 2019.

[25] T. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollar, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *European Conference on Computer Vision(ECCV)*, 2014, pp. 740–755.

[26] T. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *IEEE Conference on Computer Vision and Pattern Recogintion(CVPR)*, 2017, pp. 936–944.

Authorized licensed use limited to: University Town Library of Shenzhen. Downloaded on May 30,2021 at 16:35:28 UTC from IEEE Xplore. Restrictions apply.

[27] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in *IEEE Conference on Computer Vision and Pattern Recogintion(CVPR)*, 2018, pp. 8759–8768.
[28] Q. Zhao, T. Sheng, Y. Wang, Z. Tang, L. Cai, Y. Chen, and H. Ling, "M2Det: A single-shot object detector based on multi-level feature pyramid network," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2019, pp. 9259–9266.
[29] G. Ghiasi, T. Lin, and Q. V. Le, "NAS-FPN: Learning scalable feature pyramid architecture for object detection," in *IEEE Conference on Computer Vision and Pattern Recogintion(CVPR)*, 2019, pp. 7036–7045.
[30] H. Li, J. A. P. Xiong, and L. Wang, "Pyramid attention network for semantic segmentation," in *The British Machine Vision Conference(BMVC)*, 2018.
[31] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *IEEE Conference on Computer Vision and Pattern Recogintion(CVPR)*, 2017, pp. 1800–1807.
[32] German Chapter of the European Neural Network Society, "DAGM 2007," https://conferences.mpi-inf.mpg.de/dagm/2007/prizes.html.
[33] P. Ramachandran, B. Zoph, and Q. V. Le, "Searching for activation functions," in *International Conference on Learning Representations(ICLR)*, 2018.
[34] S. Elfwing, E. Uchibe, and K. Doya, "Sigmoid-weighted linear units for neural network function approximation in reinforcement learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 107, pp. 3–11, 2018.

Authorized licensed use limited to: University Town Library of Shenzhen. Downloaded on May 30,2021 at 16:35:28 UTC from IEEE Xplore. Restrictions apply.