International Journal of Latest Trends in Engineering and Technology
Vol.(19)Issue(3), pp.001-008
DOI: http://dx.doi.org/10.21172/1.193.01
e-ISSN:2278-621X

STEEL SURFACE DEFECT DETECTION USING EFFICIENTNET NEURAL NETWORK

Raupov Jonibek[1], Chang Gyoon Lim[1]

Abstract- **Nowadays, it has become important to improve product quality, and increase the number of products in the steel processing industry. As production speeds up, it is important too quickly and accurately identify the defect to have competitive power. In this paper, we propose an EfficientNet Neural Network to detect and identify various defects that might be present in produced steel. Recently, EfficientNet is one of the most powerful deep neural networks, which has shown excellent performance. The Single SSD (The Single Shot MultiBox Detector) network is then integrated with the main neural network (CNN) EfficientNet. SSD (The Single Shot MultiBox Detector) use for bounding boxes. By using these models, we aim to decrease the amount of faulty steel material supplied in the market and help prevent accidents. We propose to increase the high efficiency and accuracy of the defect detection process using EfficientNet Neural Network. The research results showed that shed new light on steel defect detection in real manufacturing scripts.**

Keywords – Steel Surface Defect Detection, Convolutional Neural Network, EfficientNet Neural Network, SSD

I. INTRODUCTION

Steel is one of the foremost necessary building materials in times. Steel buildings area extremely resistant against natural and synthetic disasters, making this material popular all over the world. Surface quality inspection and detection of various defects is a very important process. The detection of surface defects improves the quality of the product and makes it competitive in the market.

Recently, many digital systems develop that allow the shortcomings of the metal surface to address at a sufficiently high level. However, several shortcomings of a similar form are known, and their identification and recognition require further research. The development of algorithms for the detection and identification of surface imperfections of different roughness remains relevant. Also, systems designed for the attachment are sagacious to the light of the rolled metal strip. It is necessary to ensure the uniformity of the light flux during the process [2].

The key features of the existing various defects on the steel surface, such as cracks, porosity, burrs, and so on, depict in proper standards[1]. We can account for many defects using neural networks created based on images of detected defects or suitable surface samples [2]. In recent years, neural networks are the most commonly used classifiers in the detection of steel surface defects. Neural networks are a method that can achieve excellent results for face detection, defect detection, and natural vision classification. Neural networks are used to identify trends in which very complex data can make sense, many computer techniques can not detect, and unique patterns cannot be distinguished[3]. Different neural networks have been developed and they use different principles to determine their own rules. Each of the artificial neural networks has its strengths. Recently, many authors performed several studies based on convolution neural networks (CNN) and widely used in the detection of defects [4]. The fact is that convolutional neural networks comprise input images and that they use architecture to limit them more intelligently [5]. Most CNN's are typically all expanded by using more layers or deeper. e.g. ResNet18, ResNet34, ResNet152, so on. These algorithms achieved effective results under advanced surface preconditions to detect defects. However, just going deeper saturates the income quickly. For example, when ResNet 1000 is compared to ResNet152, Res1000 is no more accurate than Res152 because income decreases quickly after 100-150 layers.

[1] *Department of Computer Engineering, Chonnam National University, 50Daehak-ro, Yeosu-si, Jeollanam-do, South Korea*

Scaling CNN's only in one-sided (e.g., depth only) will result in quickly worsening incomes relative to the computing rising needed [6].

Localization of defects can lead the observer to intuitively find and understand surface defects. In fact, the localization of defects was included in the category of object detection. To do this, some researchers considered the detection of surface defects as a defect detection problem. Some object detection algorithms achieved high levels of accuracy and memory speed like Single Shot MultiBox Detector (SSD)[7], and Faster-Region Convolutional Neural Network (Faster-RCNN) [8]. There is some advantage of using an object detection algorithm to detect defects on the surface is that it explores immediately from prosperous and excellent algorithms in object detection tasks.

Previous studies have mainly used Bilinear Models, MobileNet-SSD (The Single Shot MultiBox Detector),Support Vector Machine(SVM), Decision Trees, and End-to-end defect detection networks (EDDN).

This paper introduces a detection of defects on steel surface based on Efficient Neural Network based on SSD [9]. For every position on the quality map, the projected model divides the output area of the boxes, delimiting the defects on a set of standard boxes of different ratios and dimensions. It creates trust points for each standard category through the network recommended for forecasting. Because of the important disparity among the confident and adverse sample, we present a difficult adverse operational method to ease this issue of detail disparity. When scaling down the model, it implements scaling in-depth, width, and resolution. Therefore, more efficient and accurate results achieve by paying more attention to these three dimensions. The aim of this research is to develop a more accurate and efficient method of detecting steel surface defects by measuring the architecture of the EfficientNet model in three steps based on SSD.

II. Related Work

Scientists reached to better the accuracy of salience prediction models for significant research in past years. The primary goal of them was the efficient allocation of resources. The major goal of scientists is the efficient allocation of resources, the accuracy of salience prediction models in past years. The principal goal of scientists is the efficient allocation of resources. This pre-processing assignment demands not only correction but also quick and efficient models. There have been several significant progress in calculating efficient neural models in recent years. The following reviews several of the more significant progress related to this work.

A. *Defect Detection –*

In the past, several neural network architectures were also used to detect defects on metal surfaces, and they have achieved excellent results.Fei Zhou proposed the Bilinear Model to detect defects on the surface. They could achieve an excellent result by using the Bilinear Model. D-VGG16 (Double-Visual Geometry Group16), primarily intended as the quality operations of the bilinear model, are output to the soft-max function to achieve the automatic classification of defects. Therefore, a warm map of the original representation received over request by Grad-CAM to one of the output characteristics in D-VGG16. Last, after processing, defects in the input representation are placed automatically [10].

Yiting Li considered the detection model of defects on surface Defect Based on MobileNet-SSD and SSD (The Single Shot MultiBox Detector) network use for the meta-structure and the main CNN integrated MobileNet into the MobileNet-SSD. The results showed that this method automatically detected surface defects more precisely and quickly than conventional machine learning methods and lightweight models [11].

Ehab A. Kholief applied Machine Learning which integrated with Deep Auto-encoder Network. This model gives better performance than others. This paper referred to the detection and classification of imperfections on the surface in hot-rolled steel lines using digital intensity images taken from different samples of this technology [12].

Sina Rezaei Aghdam investigated Decision Trees applied to LBP-based features for a fast method of detection defect on a steel surface. The classification model presented decision trees and then implemented Bootstrap Aggregating (Bagging) and Principal Component Analysis (PCA) on characteristics being taken by a domestic binary standard-based operator[13].

Xiaoming Lv proposed the defect of the metal. They instructed a recent EDDN (end-to-end defect detection network) that supported the SSD. The broad tests on two datasets showed that the presented model was powerful and could meet precision conditions for metallic defect detection [14].

B. Lightweight Architectures –

Convolutional neural networks (CNN) have been made universal in computer vision together with various other areas, but the pure size of contemporary CNN's means that for the most sensible applications, an important speed up and compaction is frequently needed.

MobileNetv2: This has made Advances in creating neural networks that allow devices with limited computing power, such as mobile phones, to run fast enough. The key idea of Mobile Net v1 was to replace convolution layers with depth-separated convolutions. It was a great idea. The major change in v2 architecture implements a reversed bottleneck part as a major building block in that point-wise convolutions with batch normalization apply to map among channel images with various dimensions, and depth wise or separable convolution is used to decrease FLOPs[15]. The number of input and output channels must be the same in the bottleneck structure, and the middle layer contains lower channels than input and output. In the back window there are more channels in the middle layer [16].

ShuffleNet: ShuffleNet suggests a foundation bottleneck unit that uses point-wise group convolution as a replacement for normal convolution to decrease FLOPS, together with channel shuffle to allow cross-group information flow [17]. To decrease input channels, Depth wise convolution is used, if in bottleneck units [16].

ShuffleNetv2: ShuffleNet V2 reorders the bottleneck structure. A channel split is used to divide the input of the unit into 2 branches. It uses concatenation in place of supplement to combine data flow of the 2 branches. It applies channel shuffle after the concatenation of the flow [17]. Group convolution returns to standard convolution with batch normalization. The network plan suggests that the convolution of the group and its particular casedepth wise convolution serve as a major component in decreasing the theoretical number for float multiply add activities [16].

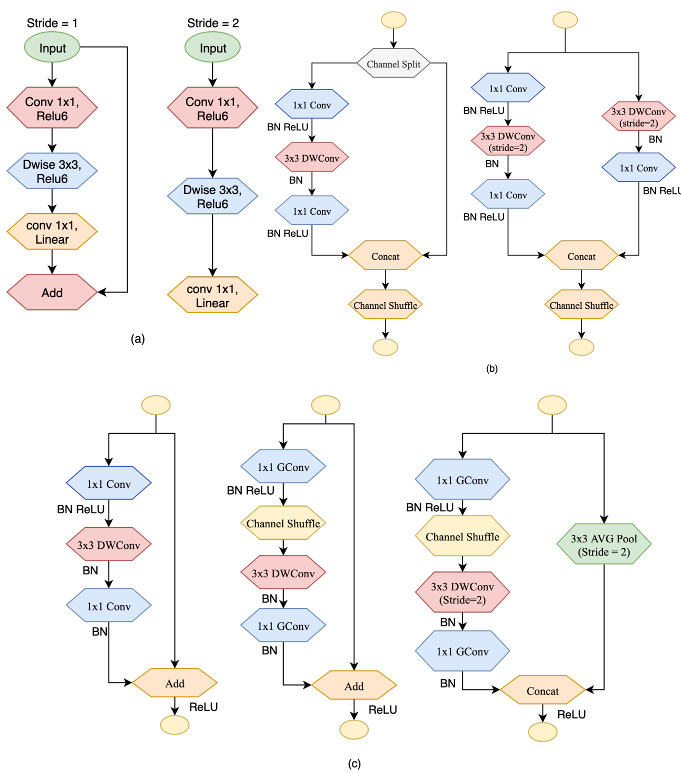

Figure 1. Lightweight units: (a) MobileNetv2 units [19], (b) ShuffleNet v2 units [20], and (c) ShuffleNetunits[21].

*EfficientNet:*It composes The EfficientNet method band of 8 models from B0 to B7, with every next method number applying to alternatives with more parameters and higher precision. In EfficientNet, many authors try an integrated technique to network design. While network depth, width, and image resolution are the 3 elements that alter the computing intricacy of a model, they proposed greater precision experiments using a compound scaling method to discover ideal formations of depth, width, and resolution - focusing on all three in combination has made for more effective results. Because of this, it scales the EfficientNet model architecture in three stages [18].

III. METHODOLOGY

In this section, we first present the EfficientNet Neural Network how to work. Then introduce The Single Shot MultiBox Detector (SSD) and our model.

A. EfficientNet –

EfficientNet, as the name suggests, is very much efficient computationally and achieves a state-of-the-art result on the ImageNet dataset. We understand the effectiveness of this in three crucial stages.

- *Depth wise Convolution and Pointwise Convolution:* To considerably decrease computational costs, it divides the initial convolution into two stages, and it loses minimal accuracy.
- *Inverse Res*: the first ResNet blocks comprise a layer that compresses and expands the channels. It connects skip links to wealthy channel layers. In MBConv, though, blocks compose of a layer that primarily expands channels and next tightens them, so it misses layers with lower channels.
- *Linear bottleneck:* apply linear activation in the last layer in every block to deter loss of data from ReLU.

A key component of EfficientNet is MBConv, to which it includes compression and excitation optimization. MBConv is the same as the inverse residual blocks used in MobileNet v2. Based on the labels between the bottlenecks, by adding a tiny number of channels, it couples this with a depth separable convolution, which reduces the count to almost n^2 in comparison with the other layers. Here, if k denotes the size of the kernel, it determines the height and width of the 2-dimensional convulsive window [18].

B. SSD structure –

The SSD network serves as a regression model. It uses boundary box regression and the qualities of various convolution layers to classify regression. It reaches good accuracy and speed of detection. The SSD uses a variety of feature maps to execute classification and localization regression - some of them from the base network. It allocates a group of standard boxes to every cage of the feature maps. Thus, each feature map produces the expression (A + 4) * k * d * c, where k is the number of standard constraints and *c* is the number of classes, and d * c is the size of the feature map [22]. Then, SSD applies several feature maps of the various sizes to use a large level together with low-level data. Although the proportion of default bounding boxes is defined, the size of the bounding box versus on each feature map. Given the use of the □ feature map in the SSD configuration, the standard box is the | k □ [1, m] can be expressed as follows:

$$s_k = s_{min} + \frac{s_{max} - s_{min}}{m-1}(k-1) \quad (1)$$

where s_{min} is parameters that can be set. The size of the standard boxes indicates the scale function. The proportions for the default boxes are indicated as $a_r \in \{1, 2, 3, 0.5, 0.33\}$ and the height h and width w of each box is $w_k^a = s_k\sqrt{a_r}$ and $h_k^a = s_k/\sqrt{a_r}$, respectively. Next, for a proportion of one, a default box with scale $s_k' = \sqrt{s_k s_{k+1}}$ is added. The center of each default box is $\left(\frac{m+0.5}{|f_k|}, \frac{n+0.5}{|f_k|}\right)$ where $|f_k|$ is the feature map size and m, n ∈ [0, $|f_k|$] [22].

The loss function of the SSD detector is delineated as a classification, L_{conf}, and summation of localization loss, L_{loc}

$$Total\ Loss = \frac{1}{N}\left(L_{conf} + L_{loc}\right) \quad (2)$$

where N is the number corresponding to the standard bounding boxes [11].

C.EfficientDefectNet –

SSD applies the EfficientNet model pre-prepared on ImageNet as its foundation model for withdrawing helpful picture elements. Above this module, SSD supplements various Conv quality layers of reducing sizes. They can be considered as a pyramid representation of pictures at separate scales. Instinctively great powdery characteristic maps at previous levels are fine, and rude-grained quality maps can detect prominent objects well. In SSD, detection occurs in each pyramidal layer, targeting objects of several sizes.

The EfficientNet based on the SSD model significantly decreases the number of parameters and achieves good results in infinite hardware situations. This model includes four areas: an input layer for the importation of the destination image, an EfficientNet foundation network for separating image properties, an SSD for bounding box and classification regression and an output layer for the exportation detection result.

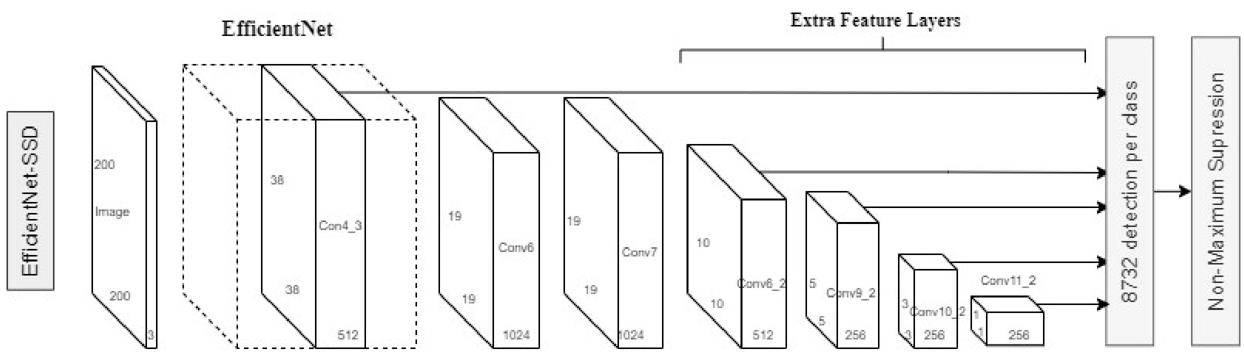

Figure 2. EfficientNet SSD network architecture

IV. EXPERIMENTAL RESULTS

A.Results and Discussion –

*Data Set:*The NEU steel defect detection dataset comprises 300 pictures of each of the 6 classes (1800 images). Images are grayscale with a similar size of 200 × 200. The defects on steel surface comprise patches, scratches, rolled-in scale, inclusion, pitted surface, and crazing.

Implementation: The project starts with loading a dataset and extracting texture features such as contrast, dissimilarity, homogeneity, and asymmetry. We localize and identify the defected part in the image by drawing a bounding box on the defect. All methods of networks were performed using the TensorFlow framework and trained on Intel(R) Core (TM) i5-6600K CPU @ 3.50GHz. During the training, we apply standard data augmentation techniques that participants are the random zooming, random horizontal flipping and random rotation. We have a tendency to use the TensorFlow Object Detection API, an evidenced library for this purpose. It splits the data 87.5 for training and 12.5 percent for testing.

Results: The metrics used for this experiment are the average precision (AP) of the class of steel (defect) and mean average precision (mAP).

Table 1. The Mean Average Precision (mAP) Defect detection performance.

Part	Proposed Method
mAP@1IOU	98.2
mAP@3IOU	95.1
mAP@.3 IOU (small)	80
mAP@3 IOU (medium)	82.5
mAP@3IOU (large)	79.3
mAP@[0.1:0.6] IOU	81

When the proposed defect detection system is trained by EfficientNet based on SSD, the system reaches a mAP (Mean Average Precision) of 81**.**

Table 2 indicates the results of the recall comparison. The proposed method can achieve good results on In, Cr, Ps, Pa, and Rs defects while the SSD achieves slightly higher result in comparison with the proposed method on defect Sc. Table 3 details the collation results of AP and mAP. The proposed method can achieve good results for Cr, Pa, Ps, Rs and Sc defects, while the SSD reaches slightly higher results than the proposed method for Rs and in defects.

Table 2. Comparison of Recall. The proposed method achieves the best Recall values on the 5 defects class. The bold shows the best data.

Types	**Recall**					
	Cr	In	Pa	Ps	Rs	Sc
YOLO-V2	0.564	0.821	0.915	0.805	0.880	0.887
YOLO-V3	0.615	0.768	0.915	0.573	0.611	0.837
Faster-RCNN	0.881	0.931	0.85	0.951	0.892	0.969
SSD	0.959	0.968	0.928	0.976	0.941	**0.983**
Proposed Method	**0.959**	**0.968**	**0.977**	**0.991**	**0.961**	0.979

Table 3. Comparison of Average Precision (AP). The proposed method achieves the best AP values on the 4 defect class. Thus, the proposed method reaches the best mAP value. The bold shows the best data.

AP	**Recall**						
	Cr	In	Pa	Ps	Rs	Sc	**mAP**
YOLO-V2	0.221	0.601	0.792	0.465	0.254	0.744	0.517
YOLO-V3	0.234	0.594	0.784	0.241	0.348	0.591	0.471
Faster-RCNN	0.379	0.796	0.864	0.827	0.549	0.859	0.758
SSD	0.421	**0.805**	0.849	0.846	**0.631**	0.841	0.791
Proposed Method	**0.423**	0.771	**0.876**	**0.861**	0.591	**0.866**	**0.811**

As detailed in Tables 2 and 3, it is difficult for YOLO methods to achieve good results in a class of six defects. The cause of these surface defects is generally small in scale, it is difficult to determine with the YOLO-V2 and YOLO-V3 methods. However, the proposed method can better divide multi-scale defects, and the map can reach 0.811. Although Faster-RCNN uses anchor boxes to solve this problem, it is still lesser than the proposed method.

Relative experiments were established to further confirm the presented algorithm. The proposed algorithm during the experiment was compared to five lightweight networks like Decision Tree, SVM, EDDN, MobileNet-SSD in terms of detection accuracy, training time. The characteristic extraction accuracy of every algorithm proposed is given in Table 4.

Table 4. The performance of detection results with different methods. SVM: Support Vector Machine. EDDN: end-to-end defect detection network

Number	Model	Accuracy	Training Time(Day)
1	Decision Tree	88.3	2
2	SVM	90	1
3	EDDN SSD	91	2
4	MobileNet-SSD	95	1
5	EfficientNet SSD	**97**	**<1**

As shown in both tables, the EfficientNet neural network is stable and quick due to the improved SSD. Overall, the proposed algorithm outperformed other lightweight algorithms in terms of accuracy of detection, training time. The training time of our proposed algorithm was less than a day, which is a very good result for detecting defects on the steel surface.

During the progress of the proposed defect detection system, several experiments were performed to reach a better result. The theory of EfficientNet, a lightweight network, improves the detection accuracy, speed, reduces the training time of this algorithm and decreases the computing load. This section compares the results got with the above-presented results and discusses the properties of the proposed systems.

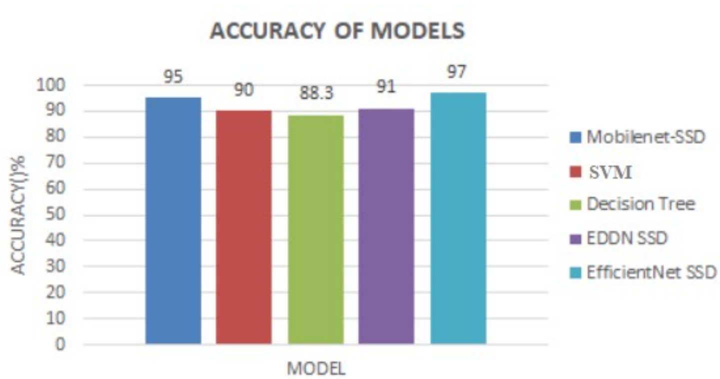

Figure 3. Model performance evaluation

The results show that the proposed method can recognize most defects in the manufacturing environment at high speed with accuracy compared to other lightweight models. EfficientNet neural network performs fast and stable, thanks to the improved SSD.

V. CONCLUSION

We propose a surface defect detection method based on the EfficientNet neural network with Single Shot MultiBox Detector and apply it to locations of defects on the steel surface. We have also performed some experiments on the major aspects that influence the speed and accuracy of the detection of defects on a steel surface. The best model of the multilevel detector based on EfficientNet showed an average accuracy of detection of 0.97 for much damage.

For future works, the work will also be extended to cover other models that can detect defects. Our long-term perspective includes plans to identify various deficiencies more quickly and accurately. We hope this will assist practitioners to select a suitable method when using object detection in the actual world.

REFERENCES

[1] Abbas Bahrami and Mahdi KianiKhouzani, "Root Cause Analysis of Surface Cracks in Heavy Steel Plates during the Hot Rolling Process," The Isfahan University of Technology, 20 July 2019.

[2] Ihor Konovalenko, Pavlo Maruschak, Janette Brezinová, Ján Viˇnáš and Jakub Brezina, "Steel Surface Defect Classification Using Deep Residual Neural Network," Ternopil, Ukraine, 6 June 2020.

[3] Khosrow Kaikhah and Sandesh Doddamet, "Discovering Trends in Large Datasets Using Neural Networks," Department of Computer Science Texas State University-San Marcos, Texas 78666.

[4] Xian Tao and Dapeng Zhang, "Automatic Metallic Surface Defect Detection and Recognition with Convolutional Neural Networks," China, 6 September 2018.

[5] Convolutional Neural Networks (CNNs / ConvNets)

[6] Less Wright, "EfficientNet from Google — Optimally Scaling CNN model architectures with "compound scaling," 31 May 2019.

[7] Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu C.Y, Berg A.C, "*SSD: Single shot multibox detector," In Proceedings of the European Conference on Computer Vision (ECCV)*, Amsterdam, The Netherlands, 16 October 2016.

[8] Ren S, He K, Girshick R, Sun J, "Faster r-cnn: Towards real-time object detection with region proposal networks," *In Proceedings of the Advances in Neural Information Processing Systems,* Montreal, QC, Canada, 12 December 2015.

[9] Liu W, Anguelov, D, Erhan, D, Szegedy C, Reed S, Fu, C.Y and Berg A.C, "SSD: Single Shot Multibox Detector," *In Proceedings of the European conference on computer vision*, Amsterdam, The Netherlands, 8–16 October 2016.

[10] Fei Zhou, Guihua Liu, Feng Xu and Hao Deng, "A Generic Automated Surface Defect Detection Based on a Bilinear Model," Southwest University of Science and Technology, Mianyang 621010, China, 3 August 2019.

[11] Yiting Li, Haisong Huang and Qingsheng Xie, "Research on a Surface Defect Detection Algorithm Based on MobileNet-SSD," Key Laboratory of Advanced Manufacturing Technology, Ministry of Education, Guizhou University, 17 September 2018.

[12] Ehab A Kholief, Samy H Darwish, M Nashat Fors, "Detection of Steel Surface Defect Based on Machine Learning Using Deep Auto-encoder Network," Conference: 7th International Conference on Industrial Engineering and Operations Management(IEOM). April 2017.

[13] Sina Rezaei Aghdam and Ehsan Amid, "A fast method of steel surface defect detection using decision trees applied to LBP based features," *Industrial Electronics and Applications (ICIEA)*, Jule 7, 2012

[14] Xiaoming Lv, Fajie Duan, Jia-Jia Jiang, Xiao Fu and Lin Ga, "Deep Metallic Surface Defect Detection: The New Benchmark and Detection Network," 11 March 2020.

[15] M. Sandler, A. Howard, M. Zhu and A. Zhmoginov, "MobileNetV2: Inverted residuals and linear bottlenecks," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2018.

[16] Feiyan Hu and Kevin McGuinness, "A Computationally Efficient Network for Visual Saliency Prediction. Insight Centre for Data Analytics," Dublin City University, August 2020.

[17] N. Ma, X. Zhang, H. T. Zheng and J. Sun, "Shufflenet v2: Practical guidelines for efficient CNN architecture design," in Proceedings of the European conference on computer vision (ECCV), 2018.

[18] Ayyüce Kızrak, "Reviewing EfficientNet: Increasing the Accuracy and Robustness of CNNs," 17 January 2020.

[19] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, Liang-Chieh Chen," MobileNetV2: Inverted Residuals and Linear Bottlenecks," Computer Science, 21 Mart 2019.

[20] Ningning Ma, Xiangyu Zhang, Hai-Tao Zheng, Jian Sun, "ShuffleNet V2: Practical Guidelines for Efficient CNN Architecture Design," 30 Jule 2018.

[21] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, Jian Sun: ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices," Computer Science, December 2017.

[22] Saeed Arabi, Arya Haghighat, Anuj Sharma, "A deep learning-based solution for construction equipment detection: from development to deployment," 18 April 2019.