Development of an Algorithm for the Program to Recognize Defects on the Surface of Hot-Rolled Metal

Alexander Vozmilov
South Ural State University (Nation Research University)
Chelyabinsk, Russia
vozmiag@rambler.ru

Leonid Andreev
Northern Trans-Ural State Agricultural University
Tyumen, Russia
andreevln@gausz.ru

Andrey Lisov
South Ural State University (Nation Research University
Chelyabinsk, Russia
lisov.andrey2013@yandex.ru

2022 International Conference on Industrial Engineering, Applications and Manufacturing (ICIEAM) | 978-1-6654-8369-8/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICIEAM54945.2022.9787116

***Abstract*—The problem of recognizing defects on the surface of hot-rolled metal is quite old, but technologies have only recently reached a sufficient level for the automation of this process [1–3]. One of the most suitable methods is the application of convolutional neural networks [4–7]. As the qualitative dataset for network training we selected the NEU (Northeastern University) surface defect database, which is a dataset of the most identified cases of hot rolled defects. Also, these data were supplemented with images of a clean hot-rolled surface. An algorithm for recognizing defects was developed. Camera parameters for machine vision were calculated.**

Keywords—machine learning, convolutional neural networks, hot rolled defect recognition

I. INTRODUCTION

Artificial intelligence (AI) has made great breakthroughs in recent years [8,9]. Computer vision is an important technique in the AI field which enables machines to see and interpret the world. Computer vision techniques require fewer expensive professional instruments and sensors and provide independence of the humans subjective experience [10, 11].

Quality inspection and control in the steel-manufacturing industry are critical issues for assuring product quality and increasing productivity. Steel defects have been deemed one of the main causes of production cost increase, so monitoring the quality of steel products during the manufacturing process is essential [12]. Defects can be attributed to various factors, e.g., operational conditions and facilities [13,14]. Steel defects must be detected to analyze the cause and enable rapid response and control. A sophisticated diagnostic model is required to detect defects properly and to enhance the capability of quality control.

The computer vision market is undergoing a steady transformation, constantly creating new solutions and technological advances. Industry 4.0 benefited even more from this technology during the pandemic, as it is widely used for inventory purposes. In 2018, the global computer vision market exceeded $9.2 billion and is expected to surpass $13.0 billion by 2025. North America and Europe are the main leaders in adopting computer vision in various fields.

Along with the growing interest in technology and the growth of the market, companies are paying more attention to the technological advances that artificial intelligence has to offer. According to research, computer vision is one of the most widely used technologies. It is used by at least 20 percent of companies worldwide.

II. CONVOLUTIONAL NEURAL NETWORK

Traditional machine learning inspections based on machine vision generally include three steps: image preprocessing, feature extraction, and classification. Support vector machine (SVM) and artificial neural networks (ANN) [6] have been used as classification methods. Common problems of these methods include their inability to handle complex background images or detect multiple targets in an image. [16]. An algorithm based on convolutional neural networks (CNN) is excellent for solving this problem [4-7].

CNN [17] were first proposed for image classification. In recent years, the quality of image classification has rapidly developed due to increased computing speed and the promotion of various competitions such as the ImageNet Large-Scale Visual Recognition Challenge (ILSVRC) [17].

Many classification models such as VGG [18], GoogLeNet [19], and ResNet [20] have been proposed and validated. With the development of artificial intelligence, CNN have been used to complete more complex tasks such as image segmentation and object detection. Recent deep learning methods, such as the YOLO series [21–23], RCNN series [24–26], SSD series [27,28], and RFCN [6], have performed well in object detection tasks, including the Pascal VOC challenge [29] and MS COCO dataset [30]. Such methods have been used to detect surface defects of workpiece and some datasets have been established; for example, Northeastern University (NEU) has established public strip steel surface defect datasets NEU-CLS [31] and NEU-DET [32]. The former is used for defect classification and the latter is used for defect identification.

The architecture of developed neural network is shown in Fig. 1. The existing NEU surface defect database dataset [32] (Fig. 2) was supplemented with 300 images of a clean hot-rolled surface. This enables the neural network to detect 6 types of defects, as well as the normal state of the rolled surface [33].

978-1-6654-8369-8/22/$31.00 ©2022 IEEE

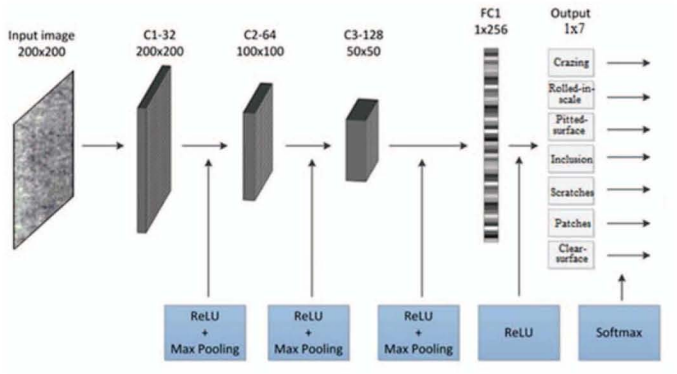

Fig. 1. CNN architecture for classification defects

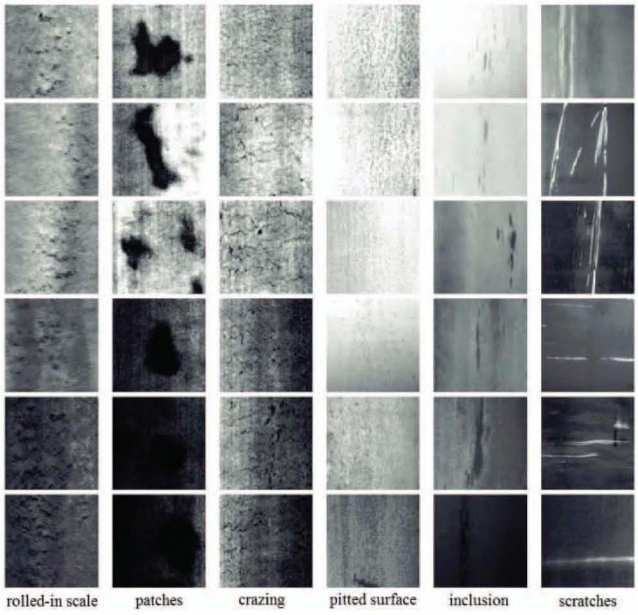

Fig. 2. Examples of metal defects

III. ALGORITHM OF THE PROGRAM

The algorithm for training a convolutional neural network is shown in Fig. 3.a. We can use Google.Collab for development. To begin with, we connect Google.Disk, to which the models will be saved and from which the dataset and test images will be loaded to verify network health. We also connect additional libraries for machine learning, working with files, visualizing results, and to facilitate the performance of mathematical operations with matrices and vectors. The architecture and hyperparameters of the network are mostly determined empirically, since increasing the parameters does not always lead to an improvement in the final result. Training will be stopped if recognition accuracy exceeds 93% (A) or if the network finishes training for a given number of training epochs. Training of the model takes roughly 40 minutes and the final result is not determined due to the stochastic effect of determining the weights. The weights and architecture of successful models are saved to the cloud after completion so that the model can later be loaded without additional time expenditures.

After saving the weights and architecture of model, the model is uploaded. Next, a series of tests is carried out on the uploaded model to determine its quality and speed of operation. The algorithm of this stage is shown in Fig. 3.b.

After training and testing the convolutional neural network, work can begin on the main function of the program—determining defects on the rolled sheet using a specialized camera for machine learning. These cameras are more durable and protected for industrial conditions and can transmit raw, unprocessed images with high definition and contrast. The height of the camera above the rolled surface is calculated based on the initial camera parameters such as resolution and viewing angle. The operational algorithm of the camera is shown in Fig. 4. The frequency of shooting the camera, determining the speed of movement of sheet metal, as well as determining the area of capturing the camera.

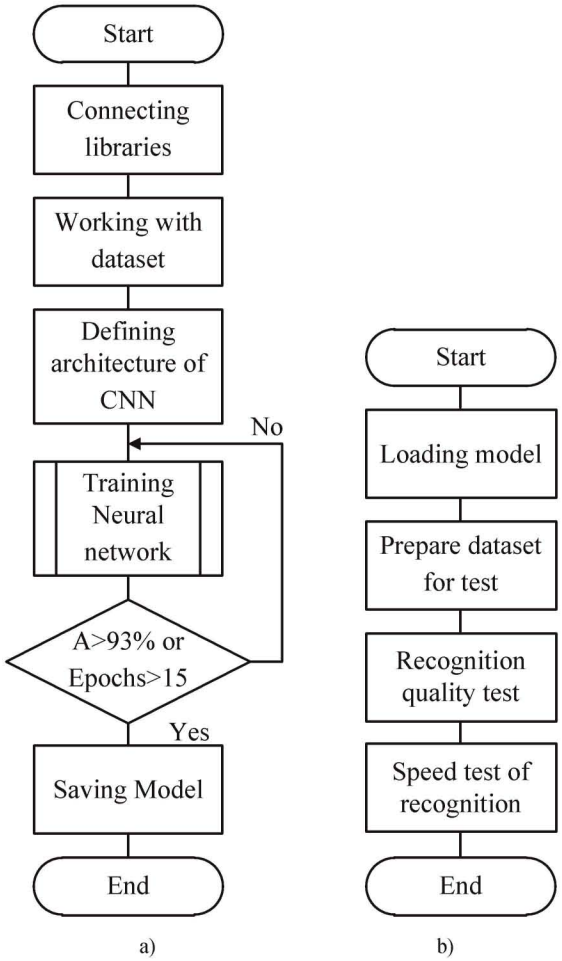

Fig. 3. a) convolutional neural network learning algorithm, b) trained neural network testing algorithm

The algorithm for the operation of a convolutional neural network for recognizing defects is shown in Fig. 5. First, the snapshots from the camera obtained at the previous stage are sent for preprocessing. At this stage, the original images are split into fragments of 200x200 pixels (See Fig. 6). The results are saved in a special array to build graphs and diagrams of the prevalence of defects and recommendations regarding the rolling equipment are made. The surface quality of the rolled

sheet is also classified at this stage according to GOST 5246-2016, Table 10 [34].

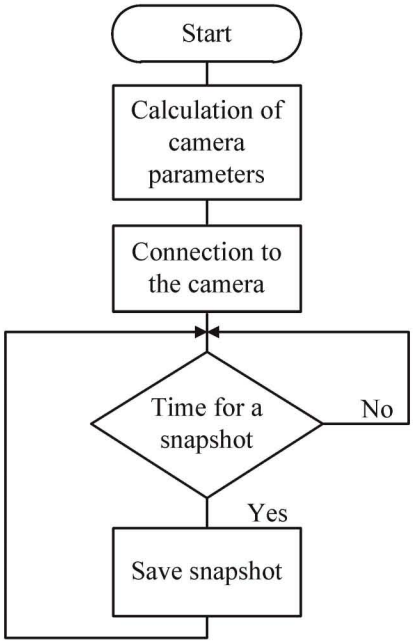

Fig. 4. Algorithm of the camera

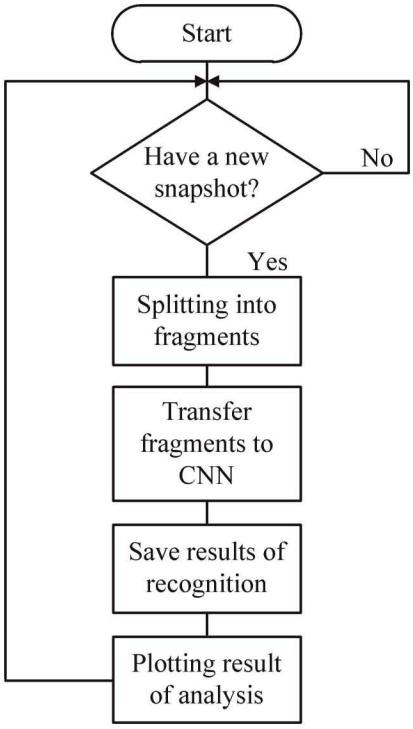

Fig. 5. Algorithm for the operation of a convolutional neural network for recognizing defects on a metal surface

IV. Calculating the Height of the Camera Above the Surface

Cameras for machine vision should be protected from humidity, dust, and high temperatures and transmit raw, unprocessed images for more accurate recognition. Below is the calculation of the camera height above the sheet metal surface. An example of calculation is shown for the TDS-VLXT-123C-FO camera; the parameters for the camera are provided in Table 1. The resulting image requires preprocessing; the processing scheme is shown in Fig. 6.

The camera is used in high temperature environments. To prevent damage to equipment, the camera is equipped with an emergency shutdown function. The Device Temperature Status Transition Selector allows the user to select different threshold values for temperatures: Normal to High: freely programmable value; High to Exceeded: fixed value (if exceeded, image recording stops); and Exceeded to Normal: freely programmable temperature value for error-free re-enabling of the camera.

The width of the camera frame is 4096 pixels, and the height of the frame is 3000 pixels. The camera angle is 72°. Based on Fig. 7, we calculate the height of the camera above the surface. To do this, we build a pyramid with square base ABCE (Fig. 8), where AB is the frame width, AE is the height, and AH is the camera height above the surface. We then select the MON triangle (Fig. 9), from which we find OH—the height of the camera above the surface.

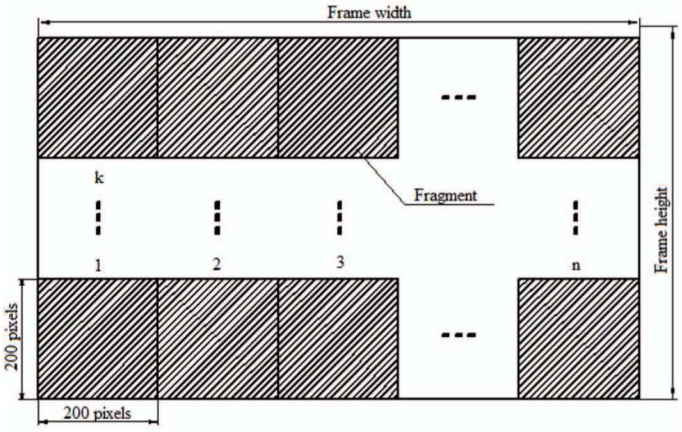

Fig. 6. Splitting the frame into fragments

TABLE 1. Camera parameters of TDS-VLXT-123C-FO

Parameter	Value
Model Name	Sony IMX253
Resolution	4096 × 3000 pixels
Viewing angle	72^0
Matrix type	Digital RGB
FPS	46 frames / sec, at a color depth of 12 bits
Image Buffer	1024 MB
Interface	10 Gigabit Ethernet
Storage Temperature	-10…+70 ^{0}C
Degree of protection	IP68

Let us calculate OH:

$$OH = \sqrt{OM^2 - MH^2} \quad (1)$$

$$MH = \frac{1}{2}MN \quad (2)$$

$$MN = AE \quad (3)$$

$$OH = \sqrt{OM^2 - 0.25 \cdot AE^2} \quad (4)$$

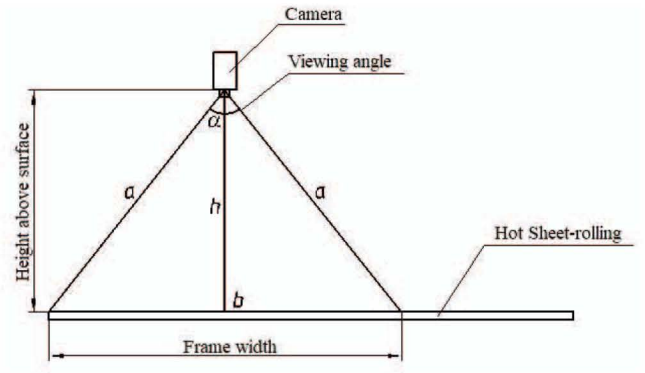

Fig. 7. Camera. Side view

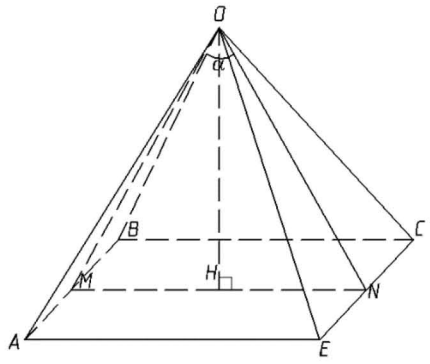

Fig. 8. The pyramid OABCE from Camera Frame

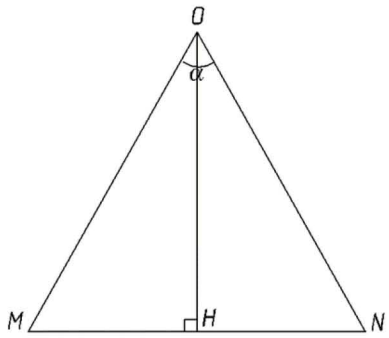

Fig. 9. Triangle MON, isolated from the pyramid OABC

Let us calculate AE, based on the ratio of frame width to height (in pixels) and move to real values by multiplying by the width of the sheet (1.9 m):

$$AE = b = \frac{\text{Frame width}}{\text{Frame height}} \cdot 1.9 \quad (5)$$

Then AE:

$$AE = b = \frac{4096}{3000} \cdot 1.9 = 2.594 \text{ m}$$

We calculate the side OM of the triangle, applying to the cosine:

$$MN^2 = OM^2 + ON^2 - 2 \cdot OM \cdot ON \cdot \cos(a) \quad (6)$$

Since MON is an isosceles triangle, OM = ON, MN = AE. Therefore:

$$MN^2 = 2 \cdot OM^2 - 2 \cdot OM^2 \cdot \cos(a) \quad (7)$$

$$OM^2(2 - 2 \cdot \cos(a)) - AE^2 = 0 \quad (8)$$

Then we find the root of the quadratic equation from the expression below:

$$OM = \sqrt{\frac{AE^2}{2 - 2 \cdot \cos(a)}} \quad (9)$$

Then OM will be equal to:

$$OM = \frac{2.594 \cdot \sqrt{2 - 2 \cdot \cos(72)}}{2 - 2 \cdot \cos(72)} = 2.206 \text{ m}$$

Thus, OH can be calculated:

$$OH = \sqrt{OM^2 - 0.25 \cdot AE^2} = \sqrt{2.206^2 - 0.25 \cdot 2.594^2} = 2.594 \text{ m}$$

CONCLUSION

The developed algorithm is universal and allows for the solution of any search and classification problem for fragments of the size of the training sample on larger images.

The proposed option for using the existing framework and cloud storage can save on the costs of developing, debugging, and testing the finished network, as it becomes possible to work on one project in groups without purchasing expensive equipment.

The method for calculating camera height can be used to determine most effective height to capture the entire area of the sheet. These calculations can be easily performed with software to recalculate the height when replacing the camera or changing the size of the rolled sheet.

The parameters are calculated once during network and camera setup. The recognition system is organized based on these calculations.

REFERENCES

[1] D. You, X. Gao, and S. Katayama, "WPD-PCA-based laser welding process monitoring and defects diagnosis by using FNN and SVM," IEEE Trans. Ind. Electron, vol. 62, no. 1, pp. 628–636, 2014.

[2] K. Fukushima and K. Neocognitron, "A self-organizing neural network for a mechanism of pattern recognition unaffected by shift in position," Biological Cybernetics, vol. 36, no. 4, pp. 193–202, 1980.

[3] R. Vilar, J. Zapata, and R. Ruiz, "An automatic system of classification of weld defects in radiographic images," NDT E Int., vol. 42, no. 5, pp. 467–476, 2009.

[4] F. Jingwen, "Recognition of surface defects on steel sheet using transfer learning," Xian Jiaotong University, vol. 5, no. 2, pp. 25–34, 2020.

[5] G. Yiping, G. Liang, L. Xinyu, and X. Yan, "A semisupervised convolutional neural network-based method for steel surface defect recognition," Robotics and Computer Integrated Manufacturing, vol. 6, pp. 825–837, 2020.

[6] X. Junjie and J. Minping, "A convolutional neural network-based method for workpiece surface defect detection," School of Mechanical Engineering, Southeast University, Nanjing, People's Republic of China – Measurement, vol. 176, pp. 379–387, 2021.

[7] Q. Luo, X. Fang, and J. Su, "Automated visual defect classification for flat steel surface: a survey," IEEE Trans. Instrum. Meas., vol. 69, no. 12, pp. 9329–9349, 2020.

[8] R. Chellappa, A. Sankaranarayanan, and A. Veeraraghavan, "Statistical methods and models for video-based tracking, modeling, and recognition," Foundations and Trends® in Signal Processing, vol. 3, no. 1-2, pp. 1–151, 2010.

[9] H. Yoon, H. Elanwar, and H. Choi, "Target-free approach for vision-based structural system identification using consumer-grade cameras," Struct Control Hlth, vol. 23, pp. 1405–1416, 2016.

[10] X. Yang, B. Yuequan, C. Jiahui, Z. Wangmeng, and L. Hui, "Surface fatigue crack identification in steel box girder of bridges by a deep fusion convolutional neural network based on consumer-grade camera images," Structural Health Monitoring, vol. 18, no. 3, pp. 653–674, 2019.

[11] Y. LeCun, Y. Bengio, "Convolutional networks for images, speech, and time series," Handbook Brain Theory Neural Netw, vol. 3361, no. 10, pp. 147–156, 1995.

[12] G. W. Song, B. A. Tama, J. Park, J. Y. Hwang, and T. Bang, "Temperature control optimization in a steel-making continuous casting process using multimodal deep learning approach," Steel Res. Int., vol. 90, pp. 125–141, 2019.

[13] S. Ghorai, A. Mukherjee, M. Gangadaran, and P. K. Dutta, "Automatic defect detection on hot-rolled flat steel products," IEEE Trans. Instrum. Meas, vol. 62, pp. 612–621, 2012.

[14] Q. Luo and Y. He, "A cost-effective and automatic surface defect inspection system for hot-rolled flat steel," Robot. Comput.-Integr. Manuf, vol. 38, pp. 16–30, 2016.

[15] H. Jaeger, "Artificial intelligence: Deep neural reasoning," Nature, vol. 538, no. 7626, pp. 467–468, 2016.

[16] O. Russakovsky, J. Deng, and H. Su, "Imagenet large scale visual recognition challenge," Int. J. Comput. Vision, vol. 115, no. 3, pp. 211–252, 2015.

[17] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," ArXiv Preprint ArXiv, vol. 15, pp. 1011–1045, 2014.

[18] C. Szegedy, W. Liu, and Y. Jia, "Going deeper with convolutions," 015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1–9, 2015.

[19] K. He, X. Zhang, and S. Ren, "Deep residual learning for image recognition," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778, 2016.

[20] J. Redmon, S. Divvala, and R. Girshick, "You only look once: Unified, real-time object detection," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 779–788, 2016.

[21] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger", 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 7263–7271, 2017.

[22] J. Redmon and A. Farhadi, "An incremental improvement," ArXiv Preprint, pp. 180–201, 2018.

[23] R. Girshick, J. Donahue, and T. Darrell, "Region-based convolutional networks for accurate object detection and segmentation," IEEE Trans. Pattern Anal. Mach. Intell., vol. 38, no. 1, pp. 142–158, 2015.

[24] S. Ren, K. He, and R. Girshick, "Faster r-cnn: Towards real-time object detection with region proposal networks," Advances in Neural Information Processing Systems, pp. 91–99, 2015.

[25] W. Liu, D. Anguelov, and D. Erhan, "SSD: single shot multibox detector," in B. Leibe, J. Matas, N. Sebe, M. Welling, Eds. Computer Vision – ECCV 2016, Lecture Notes in Computer Science, Springer, Cham, vol. 9905, 2016, pp. 21–37.

[26] C. Y. Fu, W. Liu, and A. Ranga, "Dssd: Deconvolutional single shot detector," ArXiv Preprint ArXiv, pp. 1011–1023, 2017.

[27] S. Behnke, "Hierarchical neural networks for image interpretation," Lecture Notes in Computer Science. Springer, vol. 27, 2003, pp. 67–77.

[28] J. Dai, Y. Li, and K. He, "R-fcn: object detection via region-based fully convolutional networks," Neural Inform. Process. Syst., pp. 379–387, 2016.

[29] M. Everingham, L. Van Gool, and C. K. I. Williams, "The pascal visual object classes (voc) challenge," Int. J. Comput. Vision, vol. 88, no. 2, pp. 303–338, 2010.

[30] X. Chen, H. Fang, and T. Y. Lin, "Microsoft coco captions: Data collection and evaluation server," Arxiv Preprint ArXiv, pp. 741–752, 2015.

[31] Q. Luo, X. Fang, L. Liu, C. Yang and Y. Sun, "Automated visual defect detection for flat steel surface: A survey", IEEE Transactions on Instrumentation and Measurement, vol. 69, no. 3, pp. 626–644, March 2020.

[32] GOST 21014-88. Rolled Ferrous Metals. Terms and Definitions of Surface Defects. Moscow: Publishing House of Standards, 1990.

[33] GOST 52246-2016. Hot-Dip Galvanized Sheet Metal. Moscow: Standards Publishing House, 2017.

[34] L. Andreev, V. Yurkin, and Ye. Basumatorova, "The purification of the environment from hydrogen sulphide by using wet electro-filter," E3S Web of Conferences, vol. 135, pp. 147–155, 2019.