Using Computer Vision to Recognize Defects on the Surface of Hot-rolled Steel

2022 International Ural Conference on Electrical Power Engineering (UralCon) | 978-1-6654-6652-3/22/$31.00 ©2022 IEEE | DOI: 10.1109/UralCon54942.2022.9906737

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

Vil Urmanov
Bashkir State Agrarian University
Ufa, Russia
uvg55@mail.ru

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

***Abstract*—The problem of recognizing defects on the surface of hot-rolled steel is quite old, but technology has only recently reached a sufficient level to enable automation of this process. One of the most suitable methods is applying convolutional neural networks (CNN). We selected the Northeastern University surface defect database, which is a dataset of the most identified cases of hot rolled defects, as the qualitative dataset for network training. This article presents CNN models to recognize 6 defects with an accuracy of 93.59% and 6 defects and images of a clean surface with an accuracy of 92.31%. The recognition time was 0.001384±5% seconds for all samples. As well, program give recommendations based on the most common defects of a particular type.**

***Keywords*—machine learning, computer vision, CNN, hot rolled defect recognition.**

I. INTRODUCTION

The task of recognizing defects on the surface of hot-rolled metal lay for a long time on the technologist, but at present this task can also be solved by computers [1-3]. The most common method is artificial neural networks [4-7]. Artificial intelligence (AI) has made great breakthroughs in recent years [8,9]. One of the most important uses of computer vision in manufacturing is for automating quality inspection during the production process. Maintaining quality standards is of utmost importance in the field of manufacturing. While one can do this manually through engaging quality control experts, the chances of human error are quite high and naturally limited. [10, 11]. 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 offers. According to the study, computer vision is one of the most widely used technologies.

Steel defects have been deemed one of the main causes of production cost increase, so it is essential to monitor the quality of steel products during manufacturing [12]. Defects can be attributed to various factors, e.g., operational conditions and facilities [13,14].

Much attention is paid to visual diagnostic systems for detecting defects in the steel surface. Traditional human inspection is less automated and more time consuming, among other disadvantages [15,16]. Image-based systems have been developed to provide more sophisticated, faster, and more automated verification than existing methods [17]. In addition, surface defects account for more than 90% of defects in steel products [18]. Defects on the steel surface, such as scratches, patches, and inclusions damage the material properties and corrosion resistance, as well as appearance [19]. In this article, a visual inspection system has been developed to detect steel surface defects in accordance with GOST 52246-2016.

II. DEVELOPING A CNN

For the first time, convolutional neural networks (CNN) were proposed for image classification in [20] study. In recent years, the quality of image classification has improved rapidly due to the increase in computational speed and the promotion of various competitions such as the ImageNet Large-Scale Visual Recognition Challenge (ILSVRC). The CNN architecture usually consists of two main steps: feature extraction and classification. The extracted feature representations are passed to the architecture classification component, after which the model infers the probability that a defect belongs to a particular class. The weights and biases in the model are optimized by training the neural network with the backpropagation algorithm. A CNN can successfully capture spatial and temporal dependencies in an image by applying appropriate filters. The architecture better fits the image dataset by reducing the number of parameters involved and allowing weights to be reused. In other words, the network can be trained to better understand the complexity of an image.

Such methods have been used to detect surface defects of workpieces and some datasets have been established; for example, Northeastern University (NEU) established the public strip steel surface defect datasets NEU-CLS [21] and NEU-DET [22]. The former is used for defect classification and the latter is used for defect identification.

The main problem with these datasets is the confusing spatial characteristics of the images. In some cases, it is difficult to identify spatial information as it appears sporadically in different aspects of the images. For example, patterns in scratches vary from horizontal to vertical stripes. In addition, grayscale in images often changes due to the effect of lighting [19]. Therefore, it is quite difficult to achieve high recognition accuracy without the use of additional filtering and image processing methods. This article discusses the accuracy that can potentially be obtained without the use of these additional methods.

978-1-6654-6652-3/22/$31.00 ©2022 IEEE

In this article was used the NEU dataset provided by Song and Yan [22], to establish a steel surface defect diagnostics model. The data set contains six classes of defect: rolled-in scale, patches, crazing, pitted surface, inclusion, and scratches. Each class contains 300 samples (200x200 px images). In our experiment, the data set was randomly split into training, validation, and testing data sets, where 91.67% (275) images are used as a training set and 8.33% (25) are used as testing and validation set. Google Colaboratory was used as the framework for our experiment.

A neural network summary model for recognizing 6 defects is presented below in Fig. 1; the main working library is Tensorflow. Network training results showed the following results: loss: 0.2637; accuracy: 0.9333; val_loss: 0.4073; val_accuracy: 0.9359. These results were used as a reference when developing a network that located all six defects and clean surfaces.

```
Model: "sequential_1"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d_3 (Conv2D)           (None, 199, 199, 32)      416

 max_pooling2d_3 (MaxPooling  (None, 99, 99, 32)       0
 2D)

 conv2d_4 (Conv2D)           (None, 98, 98, 64)        8256

 max_pooling2d_4 (MaxPooling  (None, 49, 49, 64)       0
 2D)

 conv2d_5 (Conv2D)           (None, 48, 48, 128)       32896

 max_pooling2d_5 (MaxPooling  (None, 24, 24, 128)      0
 2D)

 flatten_1 (Flatten)         (None, 73728)             0

 dense_2 (Dense)             (None, 256)               18874624

 dropout_1 (Dropout)         (None, 256)               0

 dense_3 (Dense)             (None, 6)                 1542

=================================================================
Total params: 18,917,734
Trainable params: 18,917,734
Non-trainable params: 0
```

Fig. 1. Compiled CNN Summary (6 defects).

A neural network model in Tensorflow library for recognizing 6 defects and clean surfaces is presented below; a summary of this model is shown in Fig. 2. The network training results showed the following results: loss: 0.1998; accuracy: 0.9548; val_loss: 0.8002; val_accuracy: 0.9231.

```
model = tf.keras.models.Sequential([
    tf.keras.layers.Conv2D(8, (2,2), activation='relu', input_shape=(200, 200, 3)),
    tf.keras.layers.MaxPooling2D(2, 2),
    tf.keras.layers.Conv2D(32, (2,2), activation='relu'),
    tf.keras.layers.MaxPooling2D(2,2),
    tf.keras.layers.Conv2D(64, (2,2), activation='relu'),
    tf.keras.layers.MaxPooling2D(2,2),
    tf.keras.layers.Flatten(),
    tf.keras.layers.Dense(256, activation='relu'),
    tf.keras.layers.Dropout(0.2),
    tf.keras.layers.Dense(7, activation='softmax')
])
```

The developed neural network showed a similar recognition accuracy for 7 classes: 92.31% versus 93.59% with 6 classes. This CNN contains a significantly fewer parameters: 9.5 million versus almost 19, which greatly affects performance. Fig. 3 and Fig. 4 show the CNN learning process and network test on 16 random images. Green text signifies a correct prediction. The class predicted by the neural network is written outside the brackets, and the real value is inside the brackets.

```
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_3 (Conv2D)            (None, 199, 199, 8)       104
_________________________________________________________________
max_pooling2d_3 (MaxPooling2 (None, 99, 99, 8)         0
_________________________________________________________________
conv2d_4 (Conv2D)            (None, 98, 98, 32)        1056
_________________________________________________________________
max_pooling2d_4 (MaxPooling2 (None, 49, 49, 32)        0
_________________________________________________________________
conv2d_5 (Conv2D)            (None, 48, 48, 64)        8256
_________________________________________________________________
max_pooling2d_5 (MaxPooling2 (None, 24, 24, 64)        0
_________________________________________________________________
flatten_1 (Flatten)          (None, 36864)             0
_________________________________________________________________
dense_2 (Dense)              (None, 256)               9437440
_________________________________________________________________
dropout_1 (Dropout)          (None, 256)               0
_________________________________________________________________
dense_3 (Dense)              (None, 7)                 1799
=================================================================
Total params: 9,448,655
Trainable params: 9,448,655
Non-trainable params: 0
```

Fig. 2. Compiled CNN Summary (6 defects + clear surface).

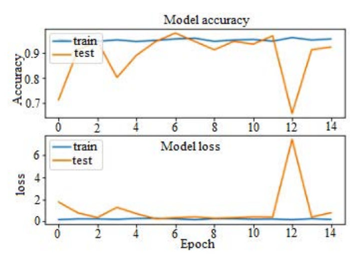

Fig. 3. Learning process of CNN.

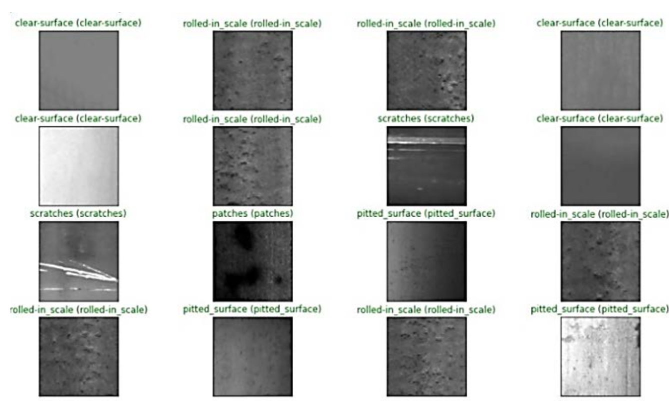

Fig. 4. Visualization of the results of neural network predictions.

A single image is calculated and classified in 0.02268 seconds; for other images, the execution time is within ±5%. It is not necessary to display images for the correct operation of the algorithm, so we will remove the visualizing component. In this case, the recognition time was 0.001384±5% seconds for all samples. The speed of the neural network is sufficient to process the original image received from the camera into fragments of 200x200 pixels in 1.7 seconds in Python.

For the convenience of the operator, defect statistics were also added to the code: a bar chart (Fig. 5), graphs (Fig. 6), and a circular chart (Fig. 7) of the defect distribution.

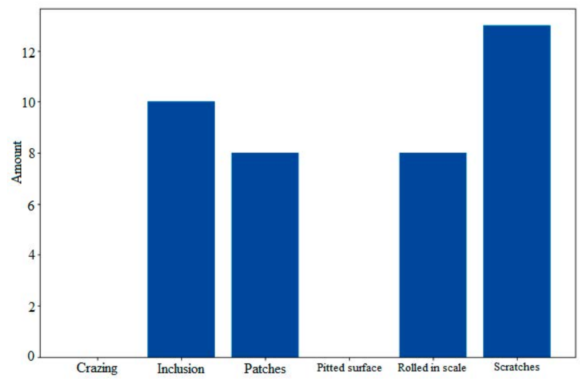

Fig. 5. Defect distribution.

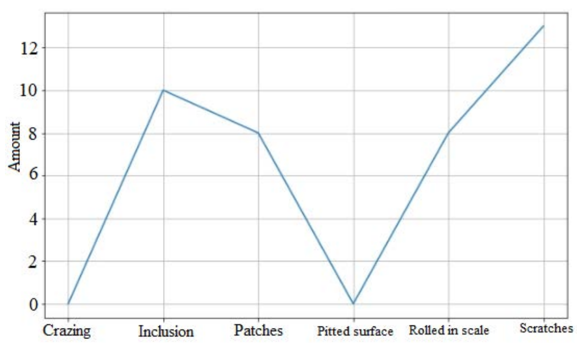

Fig. 6. Defect distribution.

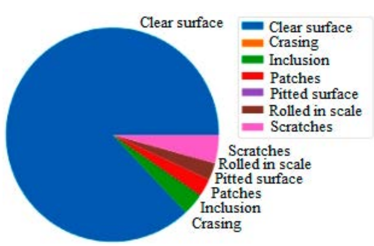

Fig. 7. Defect distribution.

III. Camera for Computer Vision

The algorithms of operation of the program and the camera in particular is presented in this paper [23]. Fig. 8 shows a schematic of the hot rolled mill, including the computer vision camera. The camera is located almost at the end of the intermediate roller table in front of the roll pusher and registers the quality of the finished product.

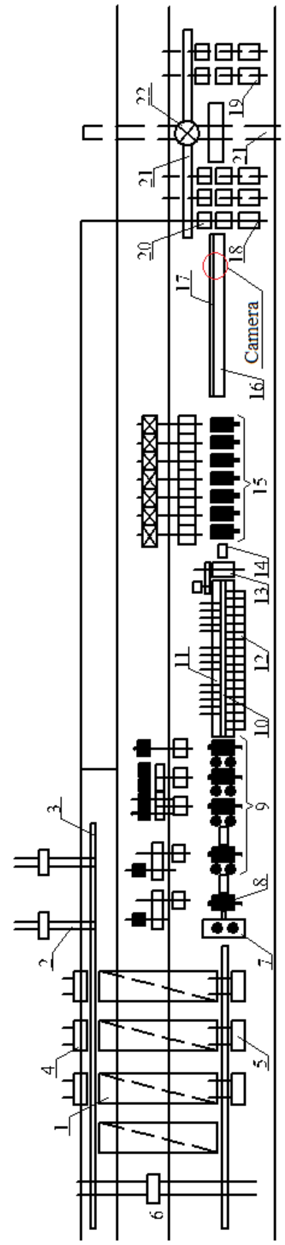

Fig. 8. Schematic of the hot rolled mill. **1** heating furnaces, **2** slab feeder, **3** loading roller table, **4** furnace pusher, **5** slab receivers, **6** transfer car for slabs, **7** vertical scale crushing stand, **8** duos roughing stand, **9** universal roughing stands , **10** intermediate roller table, **11** ejector, **12** pocket for thick strips, **13** flying shears, **14** finishing roller scale remover, **15** finishing group of stands, **16** discharge roller table, **17** water cooling system for strips, **18** winders for winding of thin strips, **19** winder drive, **20** roll pusher, **21** roll conveyor containers, **22** lifting and turning.

The surface quality of the rolled sheet is also classified at this stage according to GOST 5246-2016, Table 10 [24]. The category assignment accuracy was achieved 90.5%. Based on [25] and [26], recommendations are given for adjusting equipment and materials to improve the quality of the final product. Recommendations are made based on the most common defects of a particular type. The recommendations are provided as pieces of text, for example:

- "Attention! The stripe shows crazing and pitting. To improve the quality of rolled products to the standard, the mill rolls of the finishing group need grinding".
- "The predominant defects are inclusions and rolled-in scales. To improve the surface quality, improve the cleaning slabs and adjust the operation of the horizontal and vertical scale breakers in front of the universal roughing stands".
- "The predominant defect is patches. To improve the quality of the surface, check the operation of the scale breaker and descaling".
- "The main defect is scratches on the surface of the metal. To improve the surface quality, remove the sharp parts of the guide reinforcement or clean the surface of the finishing rolls".

CONCLUSION

As a result of the study, the most optimal option for defect recognition was chosen—a convolutional neural network, TensorFlow was chosen as the main machine learning library, and Google Colaboratory was used as the framework.

The Northeastern University surface defect database includes images of the most common surface defects in hot rolled products. This dataset was manually supplemented with 300 images of clean surfaces of hot rolled products; the resulting expanded data set was used to train the network.

The developed program determines defects and clean surfaces with an accuracy of 92.31%. The program assigns the rolled product quality group with an accuracy of 90.5%, in accordance with the data received.

The speed of the neural network is sufficient to process the original image received from the camera into fragments of 200x200 pixels in 1.7 seconds, send them to the neural network, process the results, and display them on the screen.

At the end of the analysis, the frequency of recognized defects and clean surfaces are presented in graphs, pie charts, and bar charts.

The created system also has a cheap possibility of retraining for the possibility of expanding the classification of defects, without a significant loss in recognition accuracy.

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(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(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(5), pp. 467–476, 2009.

[4] F. Jingwen, "Recognition of surface defects on steel sheet using transfer learning", Xian Jiaotong University, vol. 5(2), pp. 25–34, 2020.

[5] G Yiping, G. Liang, L. Xinyu and Xuguo 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 (12), pp. 9329–9349, 2020.

[8] R. Chellappa, A. Sankaranarayanan and A. Veeraraghavan, "Statistical methods and models for video-based tracking, modeling, and recognition" Found Trends Signal Pr vol. 3(12), 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(3), pp. 653–674, 2019.

[11] Y. Le Cun, Y. Bengio, "Convolutional networks for images, speech, and time series," Handbook Brain Theory Neural Netw, vol. 3361(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] Y. He, K. Song, Q. Meng, Y. Yan, "An end-to-end steel surface defect detection approach via fusing multiple hierarchical features," IEEE Trans. Instrum. Meas., 2019.

[16] W. Choi, H. Huh, B. A. Tama, G. Park, S. Lee, "A neural network model for material degradation detection and diagnosis using microscopic images," IEEE Access, vol. 7, pp. 92151–92160, 2019.

[17] K. Liu, H. Wang, H. Chen, E. Qu, Y. Tian, H. Sun, "Steel surface defect detection using a new Haar–Weibull-variance model in unsupervised manner," IEEE Trans. Instrum. Meas., vol. 66, pp. 2585–2596, 2017.

[18] W. Chen, Y. Gao, L. Gao, X. Li, “A new ensemble approach based on deep convolutional neural networks for steel surface defect classification,” Procedia CIRP, vol. 72, pp. 1069–1072, 2018.

[19] K. Song, Y. Yan, “A noise robust method based on completed local binary patterns for hot-rolled steel strip surface defects,” Appl. Surf. Sci., vol. 285, pp. 858–864, 2013.

[20] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” arXiv preprint arXiv vol. 15, pp. 1011–1045, 2014.

[21] Q. Luo, X. Fang, J. Su et al., “Automated visual defect classification for flat steel surface: a survey,” IEEE Trans. Instrum. Meas., vol. 69(12), pp. 9329–9349, 2020.

[22] Qiwu Luo et al., “Automated visual defect detection for flat steel surface: a survey,” IEEE Trans. Instrum. Meas., vol. 69(3), pp. 626–644, 2020.

[23] A. Vozmilov, L. Andreev, A. Lisov, “Development of an algorithm for the program to recognize defects on the surface of hot-rolled metal,” International Conference on Industrial Engineering, Applications and Manufacturing (ICIEAM), 2022.

[24] GOST 52246-2016, Hot-dip galvanized sheet metal. Moscow: Standards Publishing House, 2017, 26 p.

[25] V. I. Anuryev, Handbook of the designer - machine builder in 3 volumes. Moscow: Mashinostroenie Publishing House, 1999, 585 p.

[26] GOST 7564-97, General rules for sampling, blanks and samples for mechanical and technological testing. Moscow: Standards Publishing House, 2017, 16 p.