Pavel Kostenetskiy
Associate Professor
School of Electrical Engineering and Computer Science
South Ural State University
Russia

Rustem Alkapov
Student
School of Electrical Engineering and Computer Science
South Ural State University
Russia

Nikita Vetoshkin
Programmer
Supercomputer Simulation Laboratory
South Ural State University
Russia

Roman Chulkevich
Lecturer
School of Electrical Engineering and Computer Science
South Ural State University
Russia

Ilya Napolskikh
Student
School of Electrical Engineering and Computer Science
South Ural State University
Russia

Ostap Poponin
Programmer
Supercomputer Simulation Laboratory
South Ural State University
Russia

Real-Time System for Automatic Cold Strip Surface Defect Detection

Detection and classification of surface defects of the rolled metal are one of the main tasks for correctly assessing product quality. Historically, these tasks were performed by a human. However, due to a multitude of production factors, such as high rolling rate and temperature of the metal, the results of such human work are rather low. Replacing a human controller with an artificial intelligence system has been relevant for a long time; it is merely necessary within the concept of Industry 4.0.
This paper is devoted to the development of the prototype system automatic detection and classification of defects for one of the Iron-and-Steel Works of the Chelyabinsk region in the Russian Federation. The prototype consists of the Preprocessor, Classifier, Server, Database, and User interface. The main focus is on achieving high classification accuracy, which is planned to be obtained through the use of convolutional neural networks.

Keywords: *automatic defect detection, automatic defect classification, artificial neural networks, computer vision.*

1. INTRODUCTION

Product quality control is an essential step in metallurgical plants because quality assurance means the compliance of mechanical and other properties with specific values. However, when it comes to metal-rolling, the condition of surface or coating of rolling strips creates a separate important quality factor, which also needs to be controlled.

Surface control is usually carried out visually (by a human) at the final stages of production. However, even here, the rolling rate is very high, and there is minimal time to carry out control. At the end of the process, the metal strip is rolled up into a roll. After that, it becomes even harder to detect the defect. Often this leads to shipment of low-quality metal to the consumer, which further entails significant financial losses for the manufacturer.

Thus, the need for metallurgical enterprises for automation systems for detecting and classifying surface defects is very high. However, despite this high need, there are very few developers of such systems (Codnex, EES, Matra, Parsytec, Siemens-VAI, and Sipar). At the same time, existing systems perform defect detection well, but still, have difficulty with their classification. The challenge is since steel surfaces rolled on different rolling mills even of the same plant often differ in appearance. It leads to the complication of the algorithms for detecting and classifying surface defects [1]. Also, the complexity is due to a large number of varieties of surface defects. Characteristics and classification of defects vary for each specific production and rolling mill and are not regulated by the generally accepted standard. Even small changes in the production process can cause the emergence of new types of defects [4].

The architecture and detailed specifications of commercial vision-based steel surface inspection systems are not publicly available. However, there are many research projects on this topic. In [5], an example was described using the algorithm for feature extraction, created without using machine learning in combination with a neural network for classification. In [6], a similar approach is used to extract features, but the neural network is used to detect defects. The paper [7] presents the architecture of the entire system, including the hardware part. In [8], a convolutional neural network was used, showing a high percentage of efficiency, similar to the last work. The process of detecting and classifying defects of the surface of metal strips usually occurs in three stages [9]:

- finding regions of interest (ROIs) by segmentation;
- extracting features from the regions of interest;

Received: February 2019, Accepted: June 2019
Correspondence to: Dr. Pavel Kostenetskiy
South Ural State University, 76, Lenin prospekt,
Chelyabinsk, Russia, 454080
E-mail: kostenetskiy@susu.ru
doi:10.5937/fmet1904765K

- defects classification based on the detected features.

In Industry 4.0, the quality assurance of the products produced at the metallurgical plants will be paramount, because otherwise, it will be necessary to introduce systems for detecting defects in input materials at each plant using metal products. According to the concept of Industry 4.0, metallurgical production will be represented by a cyber-physical system [10]. Also, for this, it is necessary that the cyber-physical system can independently control the quality of the metal and recognise founded defect. It will help to assess the need to replace the failing mechanisms involved in the production or to decide whether to slow down or completely stop the production line. In a modern smart enterprise, the introduction of basic automation will not be enough, because new defects may appear in production, for which new decisions will have to be made. This work is devoted to the development of a prototype system for automatic detection and classification of defects, which can be applied as part of the cyber-physical system within the framework of The Fourth Industrial Revolution [11]. The main goal of the research is to achieve high accuracy of defect classification and the possibility of automatic further training of the artificial neural network using cloud technologies.

In Figure 1, there is a diagram of the Automatic Defect Recognition System (ADRS). The rectangles denote modules. The arrows indicate the interaction of the modules.

The preprocessor is responsible for obtaining images from video cameras, binding images to physical coordinates, splitting images into parts, finding ROIs and sending them to the classification module. *The classifier* is responsible for image classification, preparation of training data, and neural network training. *The user interface* is responsible for managing the system. *The database* stores system data. *The server* is responsible for dataflow managing, API providing and business logic.

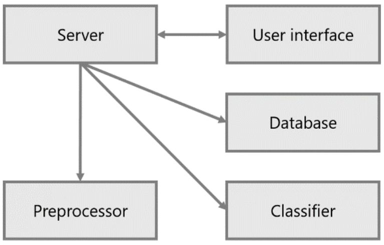


Figure 1. Scheme of System of Automatic Defect Detection (ADRS)

2. HARDWARE

ADRS will automatically detect and classify visible defects on a coated metal tape in real time before rolling. ADRS will promptly notify the operator about the appearance of the defect, its class and location, as well as record the defects in the roll passport. The system will reduce the number of unnoticed defects to almost zero.

The installation of the ADRS will include the following main components:

- Hardware system for obtaining images: a system of cameras and related equipment that captures an image of moving metal tape and transfers it to the next ADRS subsystems;
- Hardware-software and hardware system of image preprocessing: a system that performs process and reduction images obtained from cameras to the necessary form (determination of tape edges, normalisation);
- Hardware-software system for recognition and classification of metal-roll defects: a system that searches for defects on images received from the preprocessing system and, classifies them;
- Dispatcher's automated workplace: operator’s workplace with developed software for managing ADRS and monitoring the current state of the tape and roll as a whole.

2.1 Camera connection interface selection

Gigabit Ethernet, USB 3.0 Vision, and Camera Link interfaces are usually used for connection of computer vision cameras. The PCI bus is used to connect controllers of corresponding interfaces. The main problem of camera connection is the necessity to have dozens of high-speed ports on a server. Extension cards are used for this purpose. One PCI Express 3.0 4x extension card can have either up to 4 USB 3.0 ports, that work in parallel and does not share its bandwidth or only one Camera Link port. In case of using Gigabit Ethernet cameras, one dual port 10 Gigabit Ethernet card with a standard network switch in enough. Camera connection interfaces comparison is presented in Table 1.

Table 1. Camera connection interfaces

Interface	Maximum bandwidth (Gb/s)	Maximum cable length (m)	Interface card price per 1 camera ($)	Cost of an active repeater for cable extending ($)
Gigabit Ethernet	1	100	38	–
USB 3.0 Vision	4.8	3	20	200
Camera Link Extended Full	6.8	3	330	650

2.2 Camera Requirements Calculation

The developed prototype is focused on the use of area scan cameras, as, according to the authors, commercial steel surface inspection systems of high resolution, working with small defects (0.5 mm square) and high speed of the strip (7 m/s), are built on their basis. Line scan cameras are usually used only as an auxiliary part of such systems to search for specific types of defects.

Input parameters for calculating the parameters of cameras and the maximum speed of the tape is presented in Table 2:

- the minimum size of a visible defect by width and height (mm) – min_defect_size;

- the minimum number of pixels by width and height required for confident detection of a minimum defect – min_defect_pixels;
- parameters of a metal strip – width (mm);
- speed (m/s);
- percentage of an image intersection from neighbours' cameras - overlap;
- video camera parameters - width x and height y of a matrix in pixels.

The maximum physical size of a rectangle (x×y mm) that can be captured by a single camera on a metal strip with sufficient resolution to recognise minimal defect are calculated by formulas ($X = \frac{x \times min\_defect\_size}{min\_defect\_pixels}$; (1) and ($Y = \frac{y \times min\_defect\_size}{min\_defect\_pixels}$. (2):

$$X = \frac{x \times min\_defect\_size}{min\_defect\_pixels}; \tag{1}$$

$$Y = \frac{y \times min\_defect\_size}{min\_defect\_pixels}. \tag{2}$$

Minimum required frame rate of the camera at a known speed of the metal line is calculated by the formula (3):

$$fps = \frac{speed \times 1000}{Y} \tag{3}$$

Minimum required frame rate of the camera at a known speed of the metal line is calculated by the formula (4):

$$n = 2\left\lceil \frac{width}{X \times \left(1 - \frac{overlap}{100}\right)} \right\rceil \tag{43}$$

Table 2. Given values

Name	Value
min_defect_size	0.5 mm
min_defect_pixel	3
width	1650 mm
speed	7 m/s
overlap	5%

Table 3. Cameras specification

Model	Interface	Resolution (px)	FPS	Maximal conveyor speed (m/s)	Camera cost ($)	Price with accessories ($)	Quantity	Sum (k$)
acA1300 60gm NIR	GE	1280x 1024	60	10.2	754	1392	18	25,0
acA2000 50gm NIR	GE	2048x 1088	50	9.07	1699	2337	12	28,0
acA2000 165um NIR	USB 3	2048x 1088	165	29.9	1526	2326	12	27,9
acA2040 90um NIR	USB 3	2048x 2048	90	30.7	1806	2606	12	31,2
acA2040 180km NIR	Camera Link	2048x 2048	180	61.4	2506	3756	12	45,1

Basler® monochrome infrared cameras listed in

Table 3 were considered as an example. Monochrome cameras have been used to reduce the amount of data transmitted from cameras and to speed up preprocessing algorithms, as these algorithms typically use monochrome images. The average cost of a lens that completes a camera set is $600.

2.3 ADRS installation diagram

Main hardware components of ADRS for one stand will be:

- framework for placing cameras at the top and bottom of the metal tape - 1 pc;
- sealed protective covers with an integrated cooling system to protect the camera complex - 2 pcs;
- industrial video cameras - 16 pcs.;
- IR flash illumination system - 1 set;
- switching system - 1 set;
- video camera synchronisation system - 1 set;
- speed sensor of the metal tape - 1 pc.;
- roll end sensor - 1 pc.;
- server with GPU for preprocessing images - 1 pc.;
- server with GPU for the recognition and classification of the defects on metal - 1 pc.;
- data storage system - 1 pc.;
- uninterruptible power supply - 1 pc.;
- server rack - 1 pc.;
- cooling system;
- operator's AWP based on PC with monitor - 1 set.

The diagram of ADRS in presented in Figure 2. The diagram of ADRS. The base of the communication network will be network switches with 12 Gigabit Ethernet (GE) ports with Power over Ethernet (PoE) support and 10 Gigabit Ethernet port (10 GE) to connect to the image pre-processing server. The cameras will be connected via the GE interface directly to the network switch. The mechanical features of the installation cause the use of two switches instead of one with 24 ports. The image preprocessing server will be connected to the switches by two 10 GE interfaces. Switching between servers and storage will be performed using a conventional 16-port Gigabit Ethernet switch with port aggregation because the image preprocessing server will reduce the number of images coming from cameras by removing frames that do not contain defects and normalising frames sent to the server for defect classification to determine a class of the defect.

The look of the rolls may differ from one another. Therefore, ADRS has a subsystem for adapting to such deviations, and can also control lighting automatically. The backlight modules use up their resource with time, and their brightness decreases. The system calculates the optimal operation mode of the backlight modules based on the constant analysis of images received from cameras. The ADRS system uses the LED backlight that works in the infrared range. Due to the installation of several separate LED modules, the system has high maintainability.

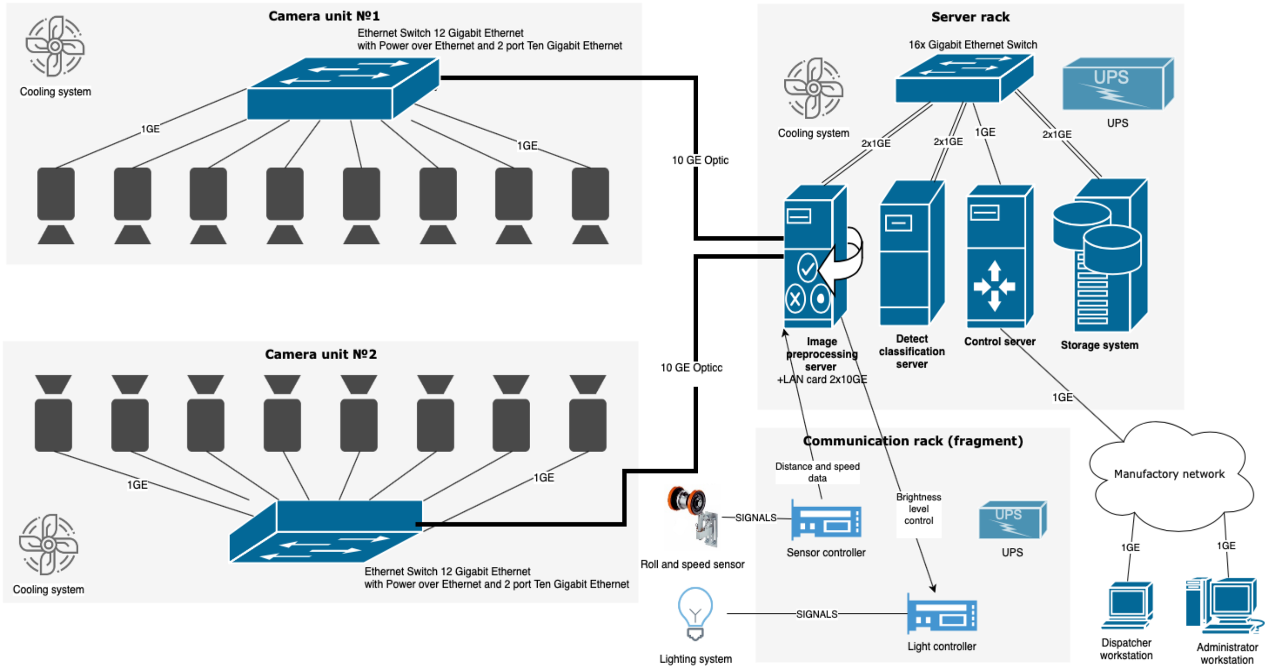


Figure 2. The diagram of ADRS

3. IMAGE PREPROCESSING UNIT

Preprocessor unit processes image dataflow from cameras and split it into pieces. Then every piece is checked with computer vision algorithms. Pieces that have a defect are then sent to the defect classification module.

2.4 Image stitching and blending

The most-known computer vision library nowadays is OpenCV**Error! Reference source not found.**.

Because of the width of a metal sheet, one camera cannot get a picture of a surface with sufficient resolution for spotting defects. Because of this, several parallel cameras are used in stand prototype. Images from all cameras need to be united in one panoramic to ease monitor faults detection process for the operator and to get one complete image of a defect, captured from two and more cameras.

An algorithm for creating a panorama usually includes a stage of finding special points and matching pictures with it. This stage can be skipped because critical points that are not defects are challenging to find on a metal sheet. Besides, the cameras' offset is fixed and known.

Next stage for creating a panoramic image is image stitching and brightness equalisation. A gradient-mixing [12] method was chosen for stitching because of its simplicity in implementation and more hidden stitch than in alpha-channel approach.

2.5 Defect Detection

It is necessary to locate regions on an image that stands out from the background since it might be defects.

The input is a grayscale image. Sobel filter is then implemented [13] for finding gradients along both axes. After that, a binary image is created [14]. At this stage, regions that alter along both axes are highlighted with white colour. Next, morphological opening [15][15] is used to enhance the results, and both of the images overlap. Then morphological closure takes place, and contours of white zones are located and filtered by area, relative to the minimum possible area of the defect.

2.6 Design

After the analysis, the next functional requirements were distinguished. The system must:

- locate visible defects on the surface of a metal sheet;
- provide API for getting images of located defects;
- stitch several images to one;
- get images from several cameras with a particular frequency;
- be able to output image to debugging interface;
- provide UI for setting parameters of image stitching;
- split input images into equal-sized parts;
- be capable of highlighting visible defects on images visually.

2.7 The computer vision stand prototype

The computer vision stands prototype was built in «Supercomputer modelling» laboratory of South Ural State University. Stand includes two cameras, fixed on a static frame, pointed down and connected to a computer. All elements are immobile. The test-bed prototype diagram is presented in Figure 3.

The operator is a user who interacts with the system. Defect classification system represents a program module that gets data from the image preprocessor module. The operator can watch a selected type of video stream – combined or from one camera. In the first case, images from all the cameras are united in one piece. In the second, image if shown form a single camera.

The operator can also calibrate cameras, i.e., change the parameters of image stitching. The defect classification system can also get a region of interest, i.e., get an image with a defect that was obtained from the image preprocessing system.

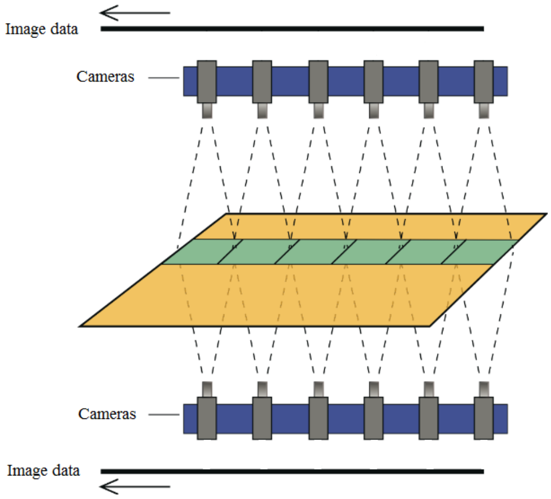


Figure 3. Test-bed prototype diagram

2.8 Module implementation

The preprocessing module is a part of the system of automatic visible defects recognition on technological lines. This module gets the video stream from cameras with a specific frequency and separates it into equal-sized parts. Then, every part is checked on defects with computer vision algorithm. Parts of images where the defect is located are then sent to the defect classification module. Also, for debugging purposes, the module supports the stitching of images and highlighting defects on the image.

2.9 Description of system components

Next classes were created to store data:

- ImagePart – an object that keeps metadata and part of the input image;
- Config – an object that holds values from the configuration file.

Classes that were created to control data:

- Preprocessor – the main class that initialises the rest of the module's classes;
- Splitter – splits an image into equal-sized parts;
- Detector – detects defects on an image;
- Blender – class that combines several images into one.

For system interaction with external systems, next interfaces were implemented:

- Streamer – a class that connects the system to cameras and gets images;
- Sender – a class that sends image parts with metadata into defect classification module.

2.10 System architecture implementation

Diagram of a main preprocessing cycle activity is presented in Figure 4.

Input image splits into equal-sized parts. The size is determined by the input data format of a neural network in the defect classification module.

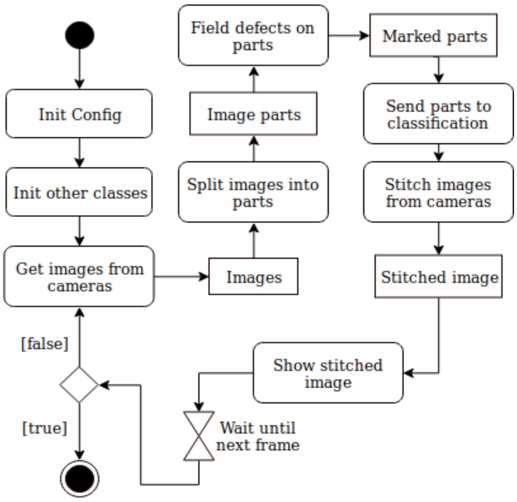


Figure 4. Main preprocessing cycle activity diagram

Activity diagram of the defect detection algorithm is presented in Figure 5.

The meaning of the defect detection algorithm is visible defects has bounds, on which pixel brightness of a grayscale image changes dramatically. A defect then might be distinguished completely after finding its borders by Sobel filer along both axes.

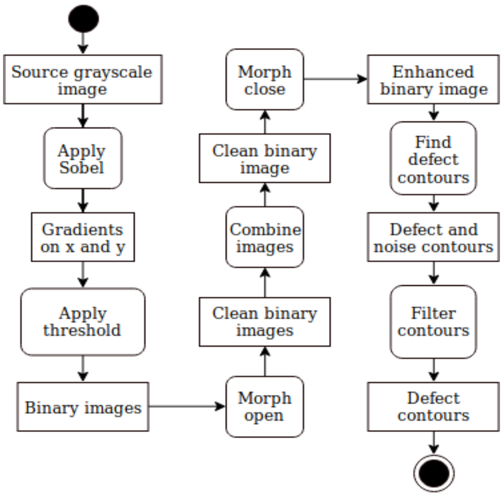


Figure 5. Activity diagram of the defect detection algorithm

If the array of received bounds is not empty, a part of an image, on which defects were located, marked as «has_defects» and sends to defect classification module. This array might also be used for defect detection in debugging purposes (see Figure 6).

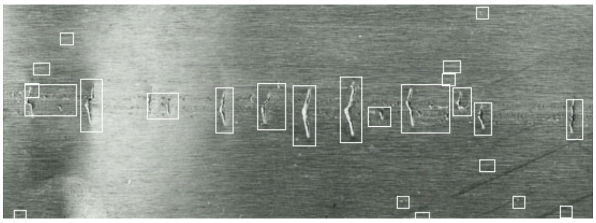


Figure 6. Example of distinguishing and highlighting of defects

2.11 Computational experiments

Computational experiments were conducted for defect detection algorithm on images of defects on hot rolled metal surfaces from Northeastern University surface defect database, USA.

After conduction of experiments, there were found coefficients of the algorithm that allows locating the highest number of defects.

4. DEFECTS CLASSIFICATION BASED ON A NEURAL NETWORK

The main tasks of the classifier are defect classification, preparation of training dataset for the neural network, and its training.

4.1 Classification approaches

Support vector machine and neural networks are among the most known classification approaches. Also, k-nearest-neighbours and fuzzy-logic are used in a number of tasks [9], [16].

Support Vector Machine is a binary classification approach. It is used to distinguish two similar classes [17],[18]. The approach was improved to identify more classes, also known as multiclass classification [19],[20]. In this method, a few binary classifications used to identify every class. More generally it requires N classification (where N is the number of classes) to distinguish one class from the [21]The method is described in [21], but there are some cons, the most significant of which is resource consumption in terms of hardware requirements.

Neural Networks is a universal approach in machine learning. It is widely used in image processing tasks, like classification, detection, segmentation (both semantic and instance). Its properties are flexible and suitable for the solution of a large class of problems [22], which includes defects classification [5],[23]. It is enough to use one-hidden-layer networks to classify defects, with input-layer size, hidden-layer size, and output-layer size parameters. Application of neural network in terms of defect on a metal surface is popular because of its capability to distinguish a lot of features [5],[23]. This approach is used for its performance and efficiency.

Northeastern University provides a dataset for defects classification and detection problems [24]. There are six different classes of most typical defects in metal manufacturing: rolled-in scale, patches, crazing, pitted surface, inclusion, and scratches. Each image is represented in black and white, has the size of 200x200, and every class has 300 images.

4.2 System requirements

The system should have the following requirements:

- accuracy of classification at least 95%;
- variable size of existing classes;
- retraining on the updated dataset.

4.3 Neural network topology

Yann LeCun's LeNet-5 topology has been taken for a basis [25]. Improvements were applied.

Activation function. ReLU for its properties replaces sigmoid: ReLU significantly increases the rate of convergence of stochastic gradient descent in comparison with the sigmoid [26].

Neural Networks topology is presented in Figure 7**Error! Reference source not found.**; there are convolutional and max-pooling layers responsible for feature extraction. The origin 200x200x3 tensor image transforms into 6x6x128 (128 maps of 6x6 size). Then classification part follows of 295366 learning parameters. It consists of a flatten layer, which allows representing maps as a vector of features, fully-connected layers, dropout layer, and one more fully-connected layer.

Topology and accuracy were achieved after some iterative process. Many topologies with different parameters were compared (according to the accuracy), and the best are selected. This process stopped when one topology out of two had been chosen.

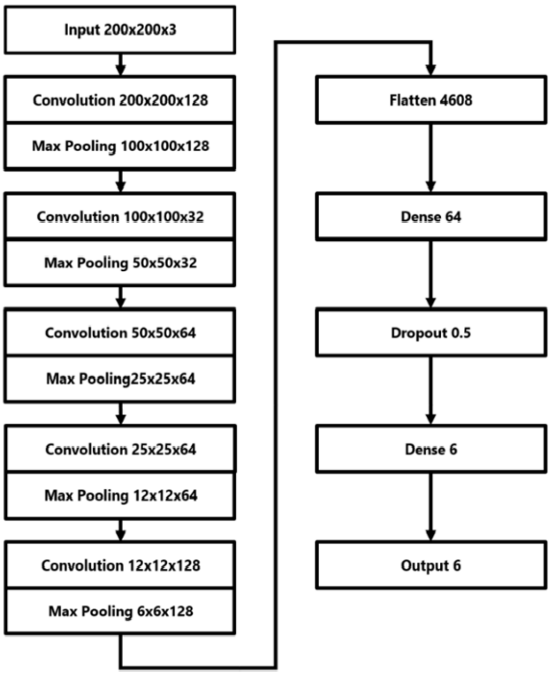


Figure 7. Neural network topology

4.4 Dataset processing

There are 1800 marked images in dataset divided into several classes of defects. This quantity is insufficient to obtain high accuracy of predictions. It means that data should be augmented to prevent overfitting. Image Data Generator module based on Keras framework was provided to achieve this goal. Abstract class DataPreparer is the base for data processing before learning. We provide NeuDataPreparer based on DataPreparer to prepare dataset from Northeastern University. Normalisation (see Figure 8) brings brightness and brightness spread in the input images into one range by Z-Score function [27].

4.5 Neural network programming representation

The Sequential model represented in Keras is a linear stack of layers. Sequential was used as a container for the neural network implementation. Details of implementation are described below.

Dropout layer was used with a 0.5 coefficient to prevent overfitting. Usually, it also provides better learning efficiency and final score [28]. Result layer is represented as a vector of probabilities of some class and computed by Softmax function [29].

During computational experiments, Adam optimiser [30] showed the best score.

Categorical Cross-Entropy was used as loss function because it applies to multiclass classification.

Keras provides different tools to manipulate the parameters of a neural network. The most convenient way to make a sequential model more accurate is callback functions. It gives an ability to manage the state of the neural network on each epoch of learning. The following functions were provided:

- Check Point is a function which allows saving the best parameters during the learning process (according to the specific metric). The metric used in neural networks is accuracy on the validation set (val acc). Callback function would save the parameters of a neural network if it scored better accuracy than a score on the previous step.
- Early Stopping is a function which allows stopping learning if it did not achieve better score during the specified amount of epoch.
- Reduce Learning Rate is a function which allows as to decay learning rate during the learning process.

The global parameters of a neural network are:

- steps per epoch are the number of steps before the end of one iteration during learning (epoch). Same for validation steps;
- input shape describes the number of dimensions of the input layer/size of images;
- Base Learning Rate, Learning Rate Factor, Min Learning Rate and Learning Rate Patience are used in Reduce Learning Rate Callback;
- upper bound for epoch number is epochs parameter, but in fact, the learning process stops due to Early Stopping callback function.

4.6 Programming implementation

TensorFlow as backend and Keras as the main framework were used for the implementation of the neural network.

We aggregated all requirements in one module named SteelDefectClassifier.

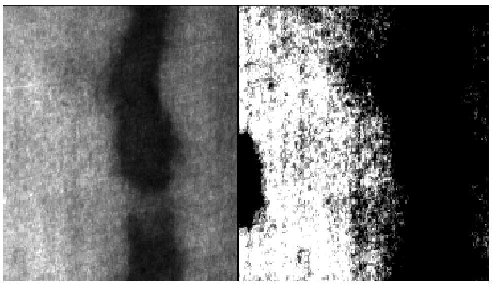


Figure 8. Original and normalized defect

The module provides the following functions:

- __get_callbacks returns a list of callback functions;
- __get_model returns the Sequential model of the neural network described above;
- classify returns vector of probabilities of provided images;
- load_model loads pre-trained network which provided as an argument;
- fit_model allows training neural network on provided DataPreparer.

Comparison of models is presented in Table 4. NVIDIA GeForce GTX 1080 Ti was used to train and test the neural network.

Table 4. Comparison of results

Models	Input Dimensions	Execution time	Accuracy	Classes of defects
Current model	200x200x3	2.1 ms	98,1%	6
Yi.L., Li.G., Jiang M. [31]	114x114	1.3 ms	99%	7
Guifang Wu. [6]	-	40 ms	88%	4
Masci J. [7]	150x150	6.2 ms	90%	7

5. GRAPHICAL USER INTERFACE OF OPERATOR'S AUTOMATED WORKPLACE

One of the parts of ADRS is the graphical interface. It is designed to ensure the interaction of the operator with other parts of the system. The graphical interface consists of various windows, each of which serves for a specific purpose.

5.1 Description of the main windows of the graphical interface

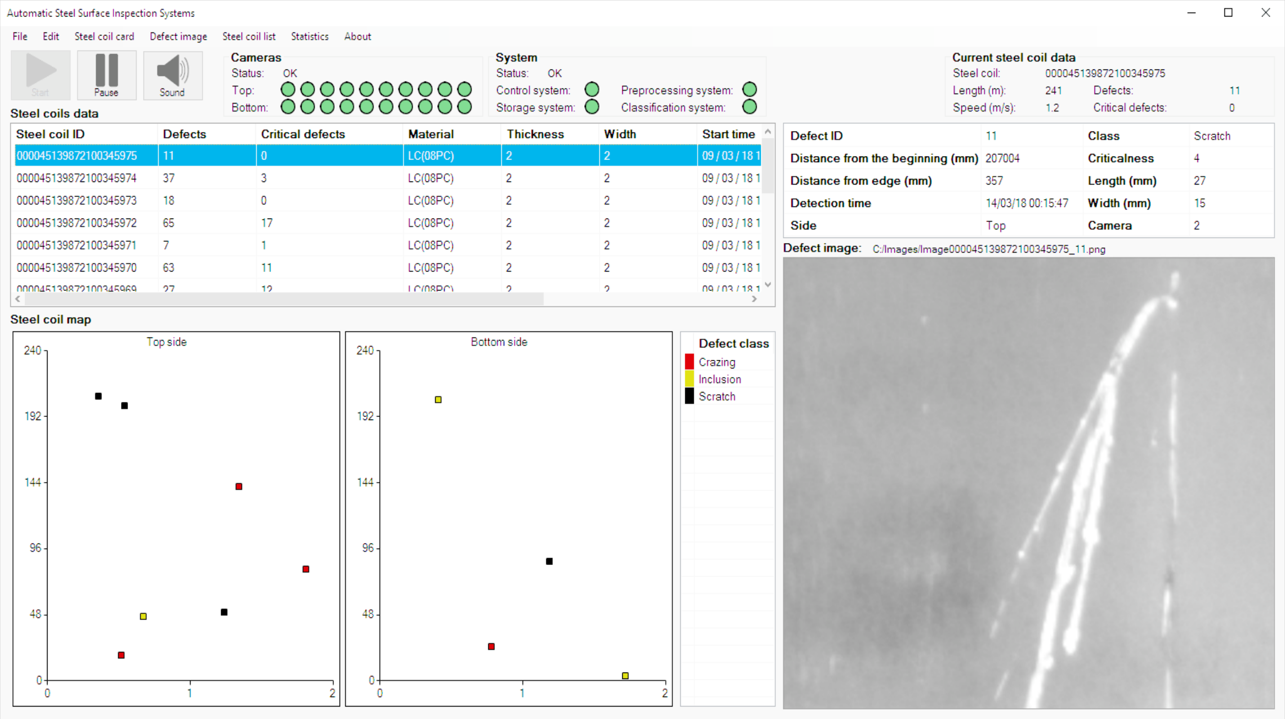
The main window of the graphical interface of the operator's automated workplace is shown in Figure 9. The window contains an image of the last detected defect, its description, a coil map, a list of produced coils and production information about the current coil. The coil map demonstrates where the defects are located on the top and bottom sides of the metal sheet. Each defect class has a unique colour.

Defects on a coil map are indicated by coloured dots, depending on their class. Information on the main window is updated automatically, but the operator can disable update on the image with the last detected defect and its description.

He can also disable an audible warning signal when a new defect appears. In the top part of the main window, there are 20 indicators of high-speed cameras and four indicators of other components of ADRS. Each indicator displays the status of its component using a different colour. It allows the operator to quickly respond to any malfunctions that may occur in the system during its operation.

The coil list window displays a list of produced coils and brief information about it. In this window, the operator can search a coil by various criteria and go to the window of the control protocol for this coil.

The coil control protocol window displays detailed production information about a coil, a list of defects, a coil map of a top and bottom sides. Production information contains an identifier, physical parameters, coil production time, number of defects, and their classes. The defect list contains all defects that were found on this coil and allows the operator to go to the defect report window. This window also contains buttons for printing and editing the coil control protocol.



Cameras
Status: OK
Top: ● ● ● ● ● ● ● ● ● ●
Bottom: ● ● ● ● ● ● ● ● ● ●

System
Status: OK
Control system: ●
Storage system: ●
Preprocessing system: ●
Classification system: ●

Current steel coil data
Steel coil: 000045139872100345975
Length (m): 241
Speed (m/s): 1.2
Defects: 11
Critical defects: 0

Steel coils data

Steel coil ID	Defects	Critical defects	Material	Thickness	Width	Start time
000045139872100345975	11	0	LC(08PC)	2	2	09 / 03 / 18 1
000045139872100345974	37	3	LC(08PC)	2	2	09 / 03 / 18 1
000045139872100345973	18	0	LC(08PC)	2	2	09 / 03 / 18 1
000045139872100345972	65	17	LC(08PC)	2	2	09 / 03 / 18 1
000045139872100345971	7	1	LC(08PC)	2	2	09 / 03 / 18 1
000045139872100345970	63	11	LC(08PC)	2	2	09 / 03 / 18 1
000045139872100345969	27	12	LC(08PC)	2	2	09 / 03 / 18 1

Defect ID	11	Class	Scratch
Distance from the beginning (mm)	207004	Criticalness	4
Distance from edge (mm)	357	Length (mm)	27
Detection time	14/03/18 00:15:47	Width (mm)	15
Side	Top	Camera	2

Defect image: C:/Images/Image000045139872100345975_11.png

Steel coil map

Defect class
🟥 Crazing
🟨 Inclusion
⬛ Scratch

Figure 9. The main window GUI of the operator's workstation

The defect report window contains a detailed description of the defect and its image. The description of the defect contains the physical parameters of the defect, class, and its coordinates on the coil. This window contains buttons for printing and editing a defect report.

The editor window allows the operator to edit information about a coil or detected defects and add a new defect. This is necessary if ADRS incorrectly classified or missed a defect.

The graphical interface allows the operator to get visual information about detected defects, statistics of their occurrence, and generate coil control protocols. The operator can decide on the expediency of stopping rolling mill and carrying out maintenance work based on this information.

5.2 The Functional Requirements for the GUI

The operator's automated workplace interface must provide the following main features:

- promptly notify the operator about a new defect on the rolled steel coil;
- display the image and information of detected defect;
- display and print out the defect's statistics;
- display the list of produced steel coils;
- form and print out steel coil control protocols and defects reports;
- display the current state of the steel coil;
- display diagnostic information about the health of the hardware imaging system;
- display diagnostic information about the health of other components of ADRS;
- provide users management in the system;
- edit information about steel coils and defects.

5.3 Implementation of GUI

The graphical interface was implemented in C# using Microsoft.NET Framework software platform.

To interact with other components of ADRS, the graphical interface of an operator's automated workplace has an API. The interaction is based on REST architecture. The data is transmitted in JSON format. Web sockets are used to display the status of unit and ADRS in real time.

6. CONCLUSIONS AND FUTURE WORK

ADRS system prototype was developed. It consists of the Preprocessor, Classifier, Server, Database, and User interface. Preliminary experiments that were carried out on the prototype showed the stability of the detection of defects and rather high accuracy of their classification – 98.1% on test data with six classes of defects. High classification accuracy was achieved through the use of convolutional neural networks. Despite this, work is currently underway to collect and augment a training set for an artificial neural network to further improve the classification results.

Using the results of the paper, we will be able to automate the process of faulty components detection of the production line by having high accuracy of defect classification and information about the causes of certain types of defects. The automatic detection of faulty production line components is one more step towards Industry 4.0 and full automation of production. Therefore, further research will be related to this task.

ACKNOWLEDGEMENT

This work was supported in part by Act 211 Government of the Russian Federation, contract № 02.A03.21.0011, by the Ministry of education and science of the Russian Federation, government order № 2.7905.2017 and № 8.9694.2017.

The reported study utilised the supercomputer resources of South Ural State University [32].

REFERENCES

[1] Baindorf, I., Anstots, T., Eberle, A., Ernenpuch, L. Holzhauser J.: Optimizatsiya protsessa i kachestva korrozionnostoykoy polosy na osnove sistemy kontrolya poverhnostey - Chernye Met., Vol. 3, pp. 45–56, 2005.

[2] Lemon, S.: Avtomaticheskiy osmotr poverhnosty polosy dlya obespecheniya sploshnogo kontrolya kachestva v linii, Chernye Met., Vol. 4, pp. 49–54, 2003.

[3] Stolzenberg, M., Gruber, K., Henkenmayer, H. and Joncourt, K.: Sistemy osmotra poverhnosty listovoy staly: Rezultaty issledovaniy, Chernye Met., Vol. 12, pp. 40–48, 2002.

[4] Vetoshkin, N., Valkevich, N. and Kostenetskiy P.: Razrabotka polzovatelskogo interfeysa dlya sistemy avtomaticheskogo kontrolya kachestva poverhnosty listovoy staly s pokrytiem, in: *Parallel Computational Technologies (PCT'2018)*, 2018, p. 400.

[5] Caleb, P. and Steuer, M.: Classification of surface defects on hot rolled steel using adaptive learning methods, in: *KES'2000. Fourth International Conference on Knowledge-Based Intelligent Engineering Systems and Allied Technologies. Proceedings*, (Cat. No.00TH8516), 2000, Vol. 1, pp. 103–108.

[6] Guifang Wu, Hoonsung Kwak, Seyoung Jang, Ke Xu, and Jinwu Xu: Design of online surface inspection system of hot rolled strips, in: *2008 IEEE International Conference on Automation and Logistics*, 2008, pp. 2291–2295.

[7] Masci, J., Meier, U., Ciresan, D., Schmidhuber, J. and Fricout, G.: Steel defect classification with Max-Pooling Convolutional Neural Networks, in: *The 2012 International Joint Conference on Neural Networks (IJCNN)*, 2012, pp. 1–6.

[8] Popat, M. and Barai, S. V.: Defect detection and classification using machine learning classifier.

[9] Neogi, N., Mohanta, D. K. and Dutta, P. K.: Review of vision-based steel surface inspection systems, EURASIP - J. Image Video Process., vol. 2014, no. 1, p. 50, Dec. 2014.

[10] Navickas V., Kuznetsova S. V. Gruzauskas: Cyber-Physical Systems Expression in Industry 4.0 Context - Financial and credit activity: problems of theory and practice, vol. 5, no. 23, pp. 188-197, 2017.

[11] Sanfelice R. G.: Analysis and Design of Cyber-Physical Systems: A Hybrid Control Systems Approach - Cyber Physical Systems: From Theory to Practice, p.3–31, 2015.

[12] Rankov, V., Locke, R. J., Edens, R. J., Barber, P. R. and Vojnovic, B.: An algorithm for image stitching and blending - SPIE Newsroom, Vol. 5701, No. March, pp. 190–199, 2005.

[13] Gupta S. and Mazumdar, S. G.: Sobel Edge Detection Algorithm - Int. J. Comput. Sci. Manag. Res., Vol. 2, No. 2, pp. 1578–1583, 2013.

[14] Senthilkumaran N., Vaithegi S.: Image Segmentation By Using Thresholding Techniques For Medical Images - Comput. Sci. Eng. An Int. J., Vol. 6, No. 1, pp. 1–13, 2016.

[15] Chudasama, D.: Image Segmentation using Morphological Operations - Int. J. Comput. Appl., Vol. 117, No. 18, pp. 16–19, 2015.

[16] Jovanović, R. Ž., Sretenović, A. A.: Ensemble of radial basis neural networks with K-means clustering for heating energy consumption prediction - FME Trans., Vol. 45, pp. 51–57, 2017.

[17] Park, C., Won, S.: An automated web surface inspection for hot wire rod using undecimated wavelet transform and support vector machine, in: *2009 35th Annual Conference of IEEE Industrial Electronics*, 2009, pp. 2411–2415.

[18] Hongbin Jia, Murphey, Y. L., Jinajun Shi, Tzyy-Shuh Chang: An intelligent real-time vision system for surface defect detection, in: *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004*, 2004, p. 239–242 Vol. 3.

[19] Ghorai, S., Mukherjee, A., Gangadaran, M. and Dutta, P. K.: Automatic Defect Detection on Hot-Rolled Flat Steel Products, IEEE Trans. - Instrum. Meas., Vol. 62, No. 3, pp. 612–621, Mar. 2013.

[20] Jie Zhao, Yongmin Yang and Ge Li: The cold rolling strip surface defect on-line inspection system based on machine vision, in: *2010 Second Pacific-Asia Conference on Circuits, Communications and System*, 2010, pp. 402–405.

[21] Batsuuri, S., Ahn, J. and Ko, J.: Steel surface defects detection and classification using SIFT and voting strategy, - Int. J. Softw. Eng. Its Appl., Vol. 6, 2012.

[22] Bettine, F., Ameddah, H. and Manaa, R.: A neural network approach for predicting kinematic errors solutions for trochoidal machining in the matsuura MX-330 five-axis Machine, - FME Trans., Vol. 46, pp. 453–462, 2018.

[23] Yazdchi, M. R., Mahyari, A. G. and Nazeri, A.: Detection and Classification of Surface Defects of Cold Rolling Mill Steel Using Morphology and Neural Network, in: *2008 International Conference on Computational Intelligence for Modelling Control & Automation*, 2008, pp. 1071–1076.

[24] Song, K. and Yunhui, Y.: NEU surface defect database. [Online]. Available: http://faculty.neu.edu.cn/yunhyan/NEU_surface_defect_database.html. [Accessed: 01-Feb-2018].

[25] LeCun, Y., Bengio, Y. and Hinton, G.: Deep learning, - Nature, Vol. 521, No. 7553, pp. 436–444, May 2015.

[26] Krizhevsky, A., Sutskever, I. and Hinton, G. E.: ImageNet classification with deep convolutional neural networks, - Commun. ACM, Vol. 60, No. 6, pp. 84–90, May 2017.

[27] Kreyszig, E.: Advanced Engineering Mathematics, John Wiley & Sons, Inc., 2011.

[28] Hinton, G., Deng, L., Yu, D., Dahl, G., Mohamed, A., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T. and Kingsbury, B.: Deep Neural Networks for Acoustic Modeling in Speech Recognition: The Shared Views of Four Research Groups, - IEEE Signal Process. Mag., Vol. 29, No. 6, pp. 82–97, Nov. 2012.

[29] Bridle, J.S.: Probabilistic Interpretation of Feed-forward Classification Network Outputs, with Relationships to Statistical Pattern Recognition, in: *Neurocomputing, Berlin, Heidelberg: Springer Berlin Heidelberg*, 1990, pp. 227–236.

[30] Ruder, S.: An overview of gradient descent optimization algorithms, - Biometrics, Vol. 55, No. 2, pp. 591–596, Sep. 2016.

[31] Yi, L., Li, G. and Jiang, M.: An End-to-End Steel Strip Surface Defects Recognition System Based on Convolutional Neural Networks, - Steel Res. Int., Vol. 88, No. 2, pp. 176–187, Feb. 2017.

[32] Kostenetskiy P., Semenikhina P. SUSU Supercomputer Resources for Industry and fundamental Science // 2018 Global Smart Industry Conference (GloSIC). IEEE, 2018. P. 1–7.

СИСТЕМ ЗА АУТОМАТСКУ ДЕТЕКЦИЈУ ДЕФЕКАТА НА ПОВРШИНИ ХЛАДНЕ ТРАКЕ У РЕАЛНОМ ВРЕМЕНУ

П. Костенецкиј, Р. Алкапов, Н. Ветошкин, Р. Чулкевич, И. Наполских, О. Попонин

Детекција и класификација површинских дефеката ваљаног метала је један од главних задатака за правилно оцењивање квалитета производа. Историјски гледано, ове задатке је обављао човек. Међутим, због мноштва фактора производње, као што су висока брзина ваљања и температура метала, резултати таквог људског рада су прилично ниски. Замена контролора човека системом вештачке интелигенције је дуго времена била релевантни задатак, док је то неопходно у оквиру концепта Индустрије 4.0.

Овај рад је посвећен развоју прототипног система аутоматске детекције и класификације дефеката за једну од челичана Чељабинске регије у Руској Федерацији. Прототип се састоји од претпроцесора, класификатора, сервера, базе података и корисничког интерфејса. Главни фокус је на постизању високе тачности класификације, што се планира постићи кориштењем конволутних неуронских мрежа.