Visual Product Inspection Based on Deep Learning Methods

Ivan Kuric[1](✉), Matej Kandera[1], Jaromír Klarák[1],
Vitalii Ivanov[2], and Dariusz Więcek[3]

[1] University of Žilina, Univerzitná 1, Žilina 010 26, Slovak Republic
ivan.kuric@fstroj.uniza.sk
[2] Sumy State University, 2 Rymskogo-Korsakova Street, Sumy 40007, Ukraine
[3] University of Bielsko Biala, Willowa 2, 43-309 Bielsko-Biala, Poland

Abstract. Nowadays, when high industrial productivity is connected with high quality and low product faults, it is common practice to use 100% product quality control. Since the quantities of products are high in mass production and inspection time must be as low as possible, the solution may be to use visual inspection of finished parts via camera systems and subsequent image processing using artificial intelligence. Recently, deep learning has shown itself to be the most appropriate and effective method for this purpose. The present article deals with the above-mentioned method of deep learning, and especially with its application when recognizing certain objects and elements during the visual product inspection.

Keywords: Product quality control · Visual inspection · Deep learning

1 Introduction

In the manufacturing industry, it is important to achieve a high quality of products along with high production productivity. The rapid increase in the product range and the reduction of time spent on manufacturing products, while increasing the complexity of finished products, are first-priority issues in modern manufacturing [1]. The ultimate goal for manufacturers is to achieve 100% quality control, which means that every single part or product on the assembly or production line is inspected and verified to be accepted or rejected. It is not easy to achieve such a state because it means that in the case of a classic approach to the final product quality control, it is necessary for the operator to check every part manually. This process has proven mainly time-ineffective and also represents a very stereotypical activity which can lead to operator inattention and thus to more mistakes. Especially when it comes to fully automated production or assembly lines, the goal of implementing 100% product quality control has proven to be a rather complicated task to solve [2]. The application of automated visual inspection of products based on computer vision principles may be the right solution for such a situation [3, 4]. Furthermore, the main ways of creating a unified information space in the quality management system were presented in researches [5, 6].

V. Tonkonogyi et al. (Eds.): InterPartner 2019, LNME, pp. 148–156, 2020.
https://doi.org/10.1007/978-3-030-40724-7_15

2 Literature Review

The term “computer vision” is nowadays generally associated with systems that work automatically based on the information acquired by the camera (or more cameras). In the field of industrial production, nomenclature machine vision is often used for computer vision. There are some types of such a system for machine vision applications: camera sensors; intelligent cameras, PC systems or custom systems; and hardware performance. Figure 1 describes a simplified scheme of a machine vision based on data processing with the personal computer. This type of system is also used to solve our task, which is to recognize specific objects in the output images of the camera [7].

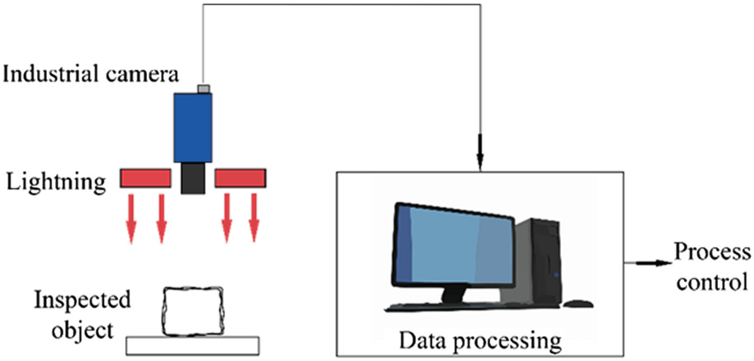

Fig. 1. Visual inspection system with industrial camera and computer data processing.

Systems for automated visual inspection based on machine vision generally consist of the following basic parts:

- An imaging device – usually a camera that consists of an image sensor, lenses, polarizing glass, protective cover, and other special parts;
- Suitable lightning method for use in specific application conditions;
- Frame grabber (which is often no longer necessary when using modern digital cameras);
- Personal computer with adequate hardware performance for further image outputs processing and evaluation.

The main principle of the machine vision system is that the camera captures images that are sent to the computer via one type of the serial communication protocol standards, like CameraLink, GigE, USB or Ethernet. The following procedure is that these images are subsequently evaluated in the computer by the pre-created algorithm [8].

Computer vision and machine learning have received wide implementation and use in different types of production [9–11], as well as in the society [12–14]. A comprehensive overview of the applications which use machine learning is presented in the research [15].

The research [16] is focused on the possibility of using effective recognition algorithms of the OpenCV library in the computer vision area. Furthermore, different algorithms, as well as their comparative analysis of the performance and recognition quality, are discussed in detail.

Deep learning as part of machine learning is also a section of a larger group of artificial intelligence methods [17, 18]. Deep learning, like the “ordinary” machine learning, is mainly based on the use of different types of so-called neural networks. However, the difference between neural networks and networks for deep learning lies in the fact that deep learning networks involve a larger number of hidden layers and are called deep neural networks. Convolutional neural networks seem to be the most successful type of deep neural networks for image processing, thanks to their significant results in the tasks which include image classification and object detection. Convolutional networks, as their name suggests, work on the principle of convolution, which is a type of linear function and is suitable mainly for data processing with a grid topology. Every network that uses a linear function of convolution at last in one of its hidden layers for the purpose of general matrix processing can be called a convolutional network [19].

Every neural network consists of three general parts, which are: an input layer, hidden layer (or layers in the case of deep neural networks) and, last but not least, the output layer. The input layer serves to load input data, which can be of various formats. In our case, in image processing, they will be set as 2D image matrices. Hidden layers serve to process data from the input layer, while the types of chosen hidden layers depend on the main task that the neural network has to fulfill. When it comes to convolutional networks, the most commonly used hidden layer types are Convolutional Layers, Dropout Layers, Batch Normalization Layers, ReLU Layers, Softmax Layers, and Fully Connected Layers. Besides, the values of so-called hyperparameters are also very important for the proper choice of hidden layer types and order. These include filter size, number of filters, number of channels, stride, padding options, number of epochs and so on. By altering the mentioned and many other hyperparameters, it is possible to achieve results with a varying level of success, and, in simple terms, they make it possible to debug the performance of the created network [20].

Depending on the network architecture, convolutional networks can serve to solve various tasks like classification, regression or object detection.

3 Research Methodology

Inspection of specific parts on a printed circuit board (PCB) was chosen as an example of a Deep Learning application for object detection. The main aim of this research is to test the possibility of creating a functional system for recognizing and classifying objects of a certain shape and type in specific images. In industrial applications, special industrial cameras are often used as a recording device. In this project, however, only an ordinary camera and ordinary artificial lighting were used, since the goal was only to create a demonstration network and test its functionality on several test samples.

The mentioned example is programmed using software MATLAB R2018b from the MathWorks company. According to the producer of this software, it is a programming platform designed specifically for engineers and scientists. Its heart is the MATLAB language, a matrix-based language that allows the most natural expression of computational mathematics.

This platform includes various types of specific modules like modules for Deep Learning and Machine Vision. The main task of this experiment was to detect specific types of components and decide whether the type, placement, and orientation of these components are correct or not. During the experiment, three types of components were detected. These components are Valor FL1173, Valor PT0018 transformers and Parallel tasking II 3Com chip, displayed in Fig. 2.

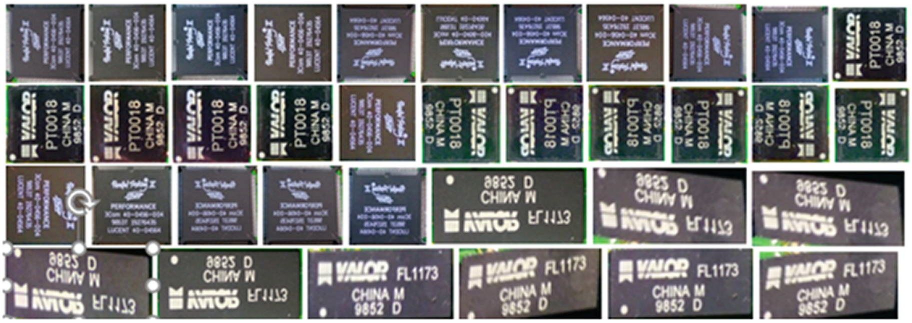

Fig. 2. Sample of training pictures.

As the first step, it was necessary to design an appropriate system. According to the previously defined task, the system was divided into two separate subsystems. The first one is designed for the object of interest detection task. Regions with Convolutional Neural Network (R-CNN) were selected for this purpose. The proposed R-CNN was created using the so-called Transfer Learning method, which is based on the use of the existing network with only a few last layers changed. The core of this “transferred” network is based on AlexNet, as this network is easily available and provides a great foundation for object recognition and detection. For the purpose of detecting specific objects in this experiment, the last three layers of AlexNet were changed. These were replaced by the new Fully Connected Layer, Softmax Layer, and Classification Layer. The new layers were added for the purpose of detecting three specific objects in the picture, and training was carried out using hundreds of pictures for each type of detected components. A sample of pictures that were set as training images is shown above in the article in Fig. 2.

The following step was to verify the trained R-CNN network on testing samples. The result of this process is shown in Fig. 3. The detected objects are in white boundary boxes. The figure also shows annotations beside the rectangular shaped boxes that define the position of components. Annotations consist of labels of each detected component, numbered from 0 to 1, that represent confidence with which the object was correctly detected and evaluation of orientation correctness. The results of this experiment have shown that the best performance of the designed system was in the range from 0,883 to 0,9994 for various components.

One of the tasks consists of orientation detection and classification of components detected in the previous steps. The chosen parts have a square and rectangular shape, so there are more possibilities to assemble them. Based on the analysis of assembling the possibilities of objects, it has been found that Transformers have two possibilities of assembling and Parallel Tasking 3Com chip has as many as four possibilities, but only one position is right for each one.

4 Results

For every component, its own convolutional neural network (CNN) was designed. CNN networks are basically composed of only few types of layers, and their order is often similar. The structure of the network used for component classification is the following: imageInputLayer ([227 227 3]); convolution2dLayer (2,2); reluLayer; maxPooling2dLayer (2,'Stride',1), convolution2dLayer (2,2); fullyConnectedLayer (4); softmaxLayer; classificationLayer ().

The layers listed above are the same layers used for training the CNN network for the orientation classification of component Parallel tasking. The parameters in the first layer were designed with respect to the fact that the detected object is square-shaped and the described situation is the same for VALOR FC. A different situation arises when classifying VALOR FL, whose shape is rectangular. In this case, the imageInputLayer parameters were set to [113 227 3], which represents its rectangular shape. The choice of other layers was based on the network simplicity criterion and on the fact that this network is only one part of a bigger system that consists of four separate neural networks. Such complex networks could be a bit slower, but they are one of the appropriate solutions to the selected task. All in all, the middle layers of the created network consist of three basic types: convolution2dLayer; ReLU Layer; maxPooling2dLayer.

The last three layers represent the classification process. For this task, the most important value is in fullyConnectedLayer, which represents the number of possibilities for assembling the detected components. The values were selected according to Table 1 and then, the neural network was trained with the following options: sgdm, MaxEpochs: 20; InitialLearnRate: 0,0005; ExecutionEnvironment: gpu (the network was trained and tested with the NVIDIA graphics card GTX 1080 Ti).

Table 1. The possibilities of component orientation during the assembly process.

Component label	Orientation possibilities (represents rotation around center)			
VALOR_FL	0° (Correct)	–	180° (Wrong)	–
VALOR_FC	0° (Correct)	–	180° (Wrong)	–
PARALLEL_TASKING	0° (Correct)	90° (Wrong)	180° (Wrong)	270° (Wrong)

Afterward, a separated program was created, where four created neural networks and testing pictures were loaded. As the first step, the program recognized the desired object (representing a specific component) in the picture and labeled this object with the previously defined boundary box. As mentioned above, this boundary box consists of the object label, confidence number, and orientation value. The final output from the main program is a picture with all the required information.

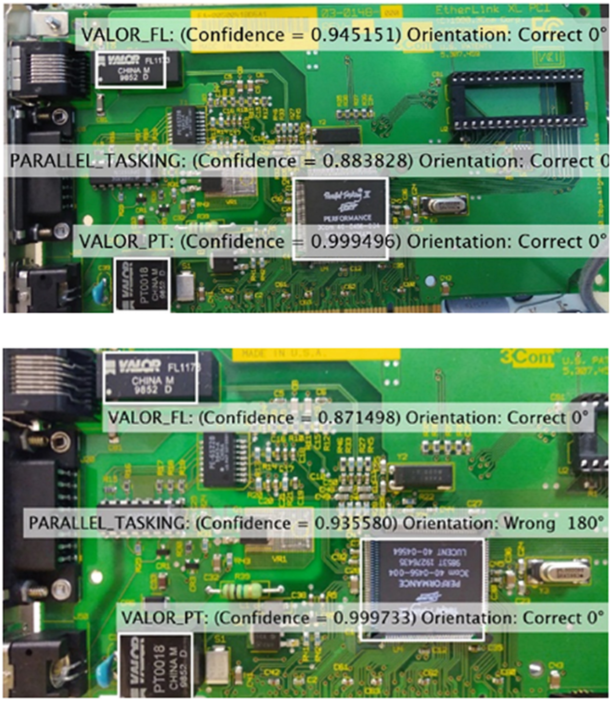
VALOR_FL: (Confidence = 0.945151) Orientation: Correct 0°
PARALLEL_TASKING: (Confidence = 0.883828) Orientation: Correct 0
VALOR_PT: (Confidence = 0.999496) Orientation: Correct 0°
VALOR_FL: (Confidence = 0.871498) Orientation: Correct 0°
PARALLEL_TASKING: (Confidence = 0.935580) Orientation: Wrong 180°
VALOR_PT: (Confidence = 0.999733) Orientation: Correct 0°

Fig. 3. Results of created network verification on testing images.

Figure 3 shows the results of the created network verification. In the picture on the left, the created network detected all desired components correctly, with confidence in the range between 0,8838 and 0,999, which is a satisfying result.

All parts were assembly corrected, so the network also correctly detected their orientation, which was in all cases 0° and Correct. In the picture on the right, the network correctly classified all components with approximately similar confidence. However, the orientation of all components was not right. The parallel tasking chip was rotated upside-down, and the created network had no problem with the detection of its orientation, and marked it as wrong, with the component rotation of 180° from the desired position.

5 Conclusions

In order to achieve high-quality products, it is necessary for the manufacturer to be able to ensure that the products he is shipping to the market meet the declared quality and performance. Different methods are used during the product control process, and one of the basic methods is visual quality control. In the industrial area, and in serial production, in particular, it is necessary to control huge quantities of products in a short time. Therefore, it is appropriate to employ automated visual inspection systems in this area. Such systems also have many advantages over manual control by workers, in particular, the fact that they can work 24 h a day and 7 days a week, they provide a simple way to collect data from the control process and, last but not least, they replace the stereotypical type of work. With these systems, the principle is that the camera captures the image which is subsequently processed by a particular algorithm.

Deep convolutional neural networks are one of the most suitable methods for processing such types of input. The method presented in this article is based on the principle of transfer learning, when using an existing functional network, changing some of its layers and transforming itself into a particular application using a brand new set of training data. The program consists of four networks with the common aim of detecting, locating and classifying specific Printed Circuit Board components. The present research is considered to be the initial attempt in the field of neural network formation for the application of the given method in the field of visual quality control of specific types of products in industrial practice.

Acknowledgment. The research was supported by the Research Grant Agency under contract No. VEGA 1/0504/17 “Research and development of methods for multicriteria accuracy diagnostics of CNC machines”.

References

1. Ivanov, V., Dehtiarov, I., Pavlenko, I., Kosov, I., Kosov, M.: Technology for complex parts machining in multiproduct manufacturing. Manage. Prod. Eng. Rev. **10**(2), 25–36 (2019). https://doi.org/10.24425/mper.2019.129566
2. Saniuk, S., Saniuk, A., Caganova, D.: Cyber industry networks as an environment of the Industry 4.0 implementation. Wireless Netw. (2019). https://doi.org/10.1007/s11276-019-02079-3. in press
3. Dodok, T., Čuboňová, N., Więcek, D.: Optimization of machining processes preparation with usage of strategy manager. In: MATEC Web of Conferences, vol. 244, p. 02004 (2018)
4. Pivarciova, E., et al.: Analysis of control and correction options of mobile robot trajectory by an inertial navigation system. Int. J. Adv. Robot. Syst. **15**(1), 172 (2018)
5. Denysenko, Y., Dynnyk, O., Yashyna, T., Malovana, N., Zaloga, V.: Implementation of CALS-technologies in quality management of product life cycle processes. In: Ivanov, V., et al. (eds.) Advances in Design, Simulation and Manufacturing. DSMIE 2018. Lecture Notes in Mechanical Engineering, pp. 3–12. Springer, Cham (2019). https://doi.org/10.1007/978-3-319-93587-4_1

6. Dynnyk, O., Denysenko, Y., Zaloga, V., Ivchenko, O., Yashyna, T.: Information support for the quality management system assessment of engineering enterprises. In: Ivanov, V., et al. (eds.) Advances in Design, Simulation and Manufacturing II. DSMIE 2019. Lecture Notes in Mechanical Engineering, pp. 65–74. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-22365-6_7
7. Kuric, I., Bulej, V., Sága, M., Pokorný, P.: Development of simulation software for mobile robot path planning within multilayer map system based on metric and topological maps. Int. J. Adv. Robot. Syst. **14**(6), 14 (2017). ISSN 1729-8814
8. Kumičáková, D., Tlach, V., Císar, M.: Testing the performance characteristics of manipulating industrial robots. Trans. VŠB – Tech. Univ. Ostrava, Mech. Ser. **62**, 39–50 (2016)
9. Wang, S.Y., Zhang, P.Z., Zhou, S.Y., Wei, D.B., Ding, F., Li, F.K.: A computer vision based machine learning approach for fatigue crack initiation sites recognition. Comput. Mater. Sci. **171**, 109259 (2020). https://doi.org/10.1016/j.commatsci.2019.109259
10. Okarma, K., Fastowicz, J.: Computer vision methods for non-destructive quality assessment in additive manufacturing. In: Burduk, R., Kurzynski, M., Wozniak, M. (eds.) Progress in Computer Recognition Systems. CORES 2019. Advances in Intelligent Systems and Computing, vol. 977, pp. 11–20. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-19738-4_2
11. Saniuk, A., Caganova, D., Cambal, M.: Performance management in metalworking processes as a source of sustainable development. In: Conference proceedings of METAl-2013, pp. 2017–2022 (2013)
12. Paul, A., Ghosh, S., Das, A.K., Goswami, S., Das Choudhury, S., Sen, S.: A Review on agricultural advancement based on computer vision and machine learning. In: Mandal, J., Bhattacharya, D. (eds.) Emerging Technology in Modelling and Graphics. Advances in Intelligent Systems and Computing, vol. 937, pp. 567–581. Springer, Singapore (2020). https://doi.org/10.1007/978-981-13-7403-6_50
13. Melinte, D.O., Dumitriu, D., Mărgăritescu, M., Ancuţa P.N.: Deep learning computer vision for sorting and size determination of municipal waste. In: Gheorghe, G. (eds.) Proceedings of the International Conference of Mechatronics and Cyber-MixMechatronics – 2019. ICOMECYME 2019. Lecture Notes in Networks and Systems, vol. 85, pp. 142–152. Springer, Cham (2020) https://doi.org/10.1007/978-3-030-26991-3_14
14. Straka, M., Khouri, S., Rosova, A., Caganova, D., Culkova, K.: Utilization of computer simulation for waste separation design as a logistics system. Int. J. Simul. Model. **17**(4), 583–596 (2018). https://doi.org/10.2507/IJSIMM17(4)444
15. Shekhar, H., Seal, S., Kedia, S., Guha, A.: Survey on applications of machine learning in the field of computer vision. In: Mandal, J., Bhattacharya, D. (eds.) Emerging Technology in Modelling and Graphics. Advances in Intelligent Systems and Computing, vol. 937, pp. 667–678. Springer, Singapore (2020) https://doi.org/10.1007/978-981-13-7403-6_58
16. Mukhanov, S.B., Uskenbayeva, R.: Pattern recognition with using effective algorithms and methods of computer vision library. In: Le Thi, H., Le, H., Pham Dinh, T. (eds.) Optimization of Complex Systems: Theory, Models, Algorithms and Applications. WCGO 2019. Advances in Intelligent Systems and Computing, vol. 991, pp. 810–819. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-21803-4_81
17. Pavlenko, I., Ivanov, V., Kuric, I., Gusak, O., Liaposhchenko, O.: Ensuring vibration reliability of turbopump units using artificial neural networks. In: Trojanowska, J., Ciszak, O., Machado, J., Pavlenko, I. (eds.) Advances in Manufacturing II. Manufacturing 2019. Lecture Notes in Mechanical Engineering, pp. 165–175. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-18715-6_14

18. Pavlenko, I., Trojanowska, J., Ivanov, V., Liaposhchenko, O.: Parameter identification of hydro-mechanical processes using artificial intelligence systems. Int. J. **2019**(5), 19–26 (2019)
19. Goodfellow, I., Bengio, Y., Courville, A.: Deep Learning. The MIT Press, Cambridge (2017). 800 p. ISBN 978-0-262-03561-3
20. Zoph, B., Le, Q., V.: Neural Architecture search with reinforcement learning. In: International Conference on Learning Representations (2016). 16 p. https://arxiv.org/pdf/161101578.pdf