ASTES

Advances in Science, Technology and Engineering Systems Journal Vol. 6, No. 2, 915-920 (2021)

www.astesj.com

Special Issue on Innovation in Computing, Engineering Science & Technology

ASTESJ

ISSN: 2415-6698

Efficient 2D Detection and Positioning of Complex Objects for Robotic Manipulation Using Fully Convolutional Neural Network

Dominik Štursa*, Daniel Honc, Petr Doležel

University of Pardubice, Department of Process Control, Faculty of Electrical Engineering and Informatics, Pardubice, 530 02, Czech Republic

ARTICLE INFO

Article history:
Received: 18 January, 2021
Accepted: 29 March, 2021
Online: 10 April, 2021

Keywords:
Machine Vision
Fully Convolutional Neural Net
U-Net
Machine Learning
Pick and Place

ABSTRACT

Programming industrial robots in a real-life environment is a significant task necessary to be dealt with in modern facilities. The "pick up and place" task is undeniably one of the regular robot programming problems which needs to be solved. At the beginning of the "pick and place" task, the position determination and exact detection of the objects for picking must be performed. In this paper, an advanced approach to the detection and positioning of various objects is introduced. The approach is based on two consecutive steps. Firstly, the captured scene, containing attentive objects, is transformed using a segmentation neural network. The output of the segmentation process is a schematic image in which the types and positions of objects are represented by gradient circles of various colors. Secondly, these particular circle positions are determined by finding the local maxima in the schematic image. The proposed approach is tested on a complex detection and positioning problem by evaluation of total accuracy.

1. Introduction

Automated systems have been developing rapidly for decades and they have increasingly helped to improve the reliability and productivity in all domains of industry. Machine vision, which is the family of methods used to provide imaging-based and image-processing-based inspections and analyses, is an indispensable element of automation. The implementations of machine vision approaches can be found in process control [1], automatic quality control [2], and especially in industrial robot programming and guidance [3].

Considering robot programming and guidance in an industrial environment, more and more intelligent machines are being utilized to deal with various applications. These days, a static and unchanging production environment is often being replaced by dynamically adapting production plans and conditions. Therefore, the assembly lines are managed on a daily basis and consequently, the automated systems and industrial robots need to be capable of dealing with more generalized tasks (general-purpose robotics). Although most robotic applications are still developed analytically or based on expert knowledge of the application approach [4], some industrial robotics producers have begun to implement deep learning methods in their applications like Keyence and their IV2 Vision Sensor. It is beginning to be generally recognized, that deep learning methods can play a significant role especially in the mentioned general-purpose robotics [5].

Deep learning consists of a family of modified machine learning methods aiming to solve the tasks that come naturally to human beings. Deep learning methods are performed directly on the available task-specific data in order to get a heuristic relation between the input data and the expected output. Various deep learning approaches have already been applied successfully to deal with various classification and detection tasks [6, 7] and are also utilized in other domains.

In general-purpose robotics, the "pick and place" task is the key problem to deal with. Generally, a "pick and place" task consists of a robotic manipulator (or group of manipulators) able to pick a particular object of attention and place it in a specific location with defined orientation.

In this contribution, the initial part of the "pick and place" issue is examined. To be more specific, the grasp point or grasping pose, which defines how a robotic arm end-effector should be set in order to efficiently pick up the object, is dealt with. Clearly, there is a broad group of grasp point detection techniques which can be listed either according to the type of perception sensor, or by a

* Corresponding Author: Dominik Štursa, nám. Čs. Legií 565, 530 02 Pardubice, Czech Republic, +420466037124, dominik.stursa@upce.cz

procedure used for object and grasp point detection and positioning. The most current methods are clearly described in survey [8]. Deep learning approaches for the detection of robotic grasping poses are summarized in review [4]. In this contribution, we propose a rapid, efficient, and accurate system for finding multiple object centers for flat and moving surfaces.

The key contribution of this article is:

- Proposal of an efficient grasp point detection method for a robotic arm capable of handling more types of objects. This method uses a classical industrial camera as the input data source.
- As an important part of the method, proposal of a deep learning-based approach to transform an RGB image of the scene of interest into a schematic grayscale frame. In this frame, the types of objects and the feasible positions of the grasp points are coded into gradient shapes of various colors. To the authors' knowledge, this is the first application of this approach to the grasp point detection.
- The proposed grasp point detection method is insensitive to changing light conditions and highly variable surroundings. In addition, it is efficient enough to be used in real time with specific edge computing tools, such as NVIDIA Jetson Nano, Google Coral or Intel Movidius.

The structure of the article is as follows. The aim of the work is formulated, and the goals are defined in the next section. Then, the solution based on deep learning approach is proposed. After that, the case study, which should demonstrate the main features of the proposed solution, is presented. Finally, the results are summarized, and the article is finished with some conclusions. This paper is an extension of work originally presented in the 24th International Conference on System Theory, Control and Computing (ICSTCC) [9].

2. Problem Formulation

As stated already in the preceding section, we deal with the essential task of industrial robotics called the “pick and place” problem. To be more specific, we are interested in the first challenge of this problem, i.e. detection and positioning of the objects. As a very necessary and attractive problem, detection and positioning of objects of interest has been examined and researched from many points of view [10, 11]. These days, due to a greater use of laser scanner technology (e.g. Photoneo Phoxi 3D Laser Scanner, Faro Focus 3D Laser Scanner), clouds of points are often used for object detection and positioning [12, 13]. Apparently, laser scanners in combination with robust 3D object registration algorithms, provide a strong tool to be applied in “pick and place” problems. Nevertheless, solutions based on clouds of points are generally costly and for some materials (shiny metal, glass, etc.), their accuracy decreases. In addition, a large number of laser scanners provide framerates too low to be used with moving objects of interest. Therefore, we propose an alternative solution based on a classical monocular RGB industrial camera as an image acquisition sensor. Besides, we suggest a novel approach based on a fully convolutional neural network in combination with a classical image processing routine, in order to analyze the signal from the industrial camera.

The proposed solution is supposed to meet the requirements of the industrial sector, i.e. stable performance, insensivity to light conditions, quick response and reasonable cost. In order to demonstrate these requirements, we perform a case study, which is summarized at the end of this article. This case study is supposed to fulfill the following conditions:

- Objects – up to four different complex objects should be detected and located.
- Conditions – Detection and positioning accuracy should not be affected by background surface change.
- Framerate – In order to be able to register moving objects, the framerate should exceed 10 frames per second.
- Hardware – The detection and positioning system should be based on hardware suitable for industrial applications. However, the costs should not exceed $500 to be economically viable.

The graphic plan of our task is depicted in Figure 1.

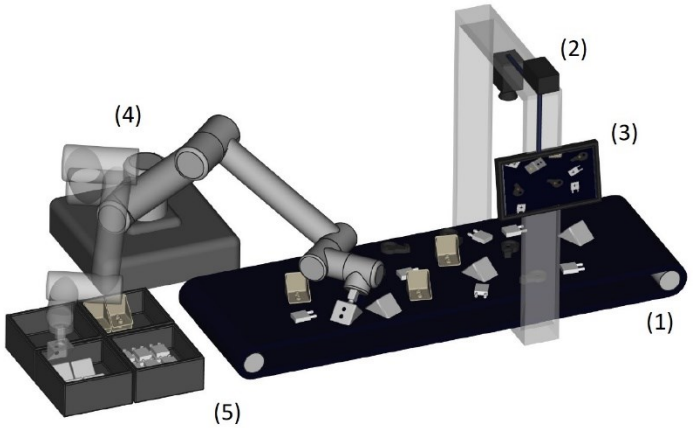

Figure 1: The arrangement of the task - the conveyor belt (1) brings the objects, the industrial camera (2) takes the image of the area, the detection and positioning system (3) determines the grasp points for manipulation and the robotic arm (4) puts the objects to the desired positions (5)

3. Proposed Approach

We propose that the approach for object detection and positioning is composed of two parts. The first part is designed to perform the scanning of the area and it provides the visual data to be processed by the following part. The second part then processes the data and provides particular detected objects and their positions. Using this information, a parent control system should be able to manipulate the objects in order to achieve the desired positions.

3.1. Camera sensor

In order to achieve a sufficient framerate, we propose to implement an ordinary industrial monocular RGB sensor equipped with a corresponding lens as the source of input image. Ostensibly, the camera and lens should be chosen according to the situation in the specific task (the scanned scene size, light, the distance of the camera from the objects, etc.). The tutorials of a camera sensor and lens combination selection are available at the vision technology producer information sources e.g. the Basler Lens Selector provided by Basler AG.

3.2. Detection and positioning system

Various studies published in recent years prove that the family of convolutional neural network topologies (CNN) outperforms classical image processing methods in tasks of object detection and classification benchmarked on various datasets [14, 15]. Following this fact, we propose a detection and positioning system, that uses CNN to transform the original RGB image of the monitored area into a specific schematic image. The main purpose of this particular operation is to create a graphic representation, where the positions of the detected objects are highlighted as gradient circles in defined colors, while the rest of the image is black. Specifically, each pixel in an RGB image representing the scene is labeled with a float number in the range <0; 1>, where 1 means the optimal and 0 means the most unsuited grasp point position. These labels are situated in the R layer, G layer, B layer or all layers of the output of the CNN, according to the type of object. Hence, the positions of the gradient circles represent not only the positions of the detected objects, but the exact points on the object body, which are optimal for manipulation by a robotic arm (grasp points). The proposed approach is described in Figure 2.

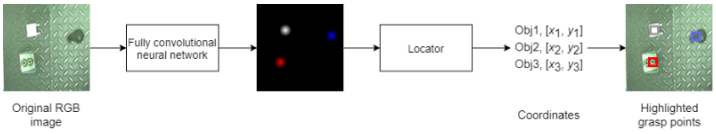

Figure 2: The proposed approach - the original RGB image of the scanned area is transformed with a convolutional neural network into the schematic image, where the optimal grasp point positions for manipulation are highlighted by gradient circles of various colors, each particular color then represents the type of the object

The first step of the proposed procedure, i.e. the transformation of the RGB image into a graphical representation of the object positions, is a key element of our approach. We propose to implement a fully convolutional neural network connected as an encoder-decoder processor. Such a processor provides encoding the original input into a small shape and restoring it using the decoder's capabilities. If the process is successful, the approach provides a correctly transformed image as the output from the decoder. We believe, that the correctly designed fully convolutional neural network is able to code gradient circles of defined colors on the exact positions of grasp points on the object bodies.

Therefore, in the next subsection, we introduce a specific fully convolutional neural network, that transforms an original RGB image into a graphical representation, where object grasp points are represented as radial gradients of defined colors.

3.3. Fully convolutional neural network for image transformation

Apparently, many different fully convolutional neural networks, such as ResNet [16], SegNet [17] or PSPNet [18], have been proposed to deal with various image processing tasks. From a wide family of neural network topologies, we select U-net as an initial point of development.

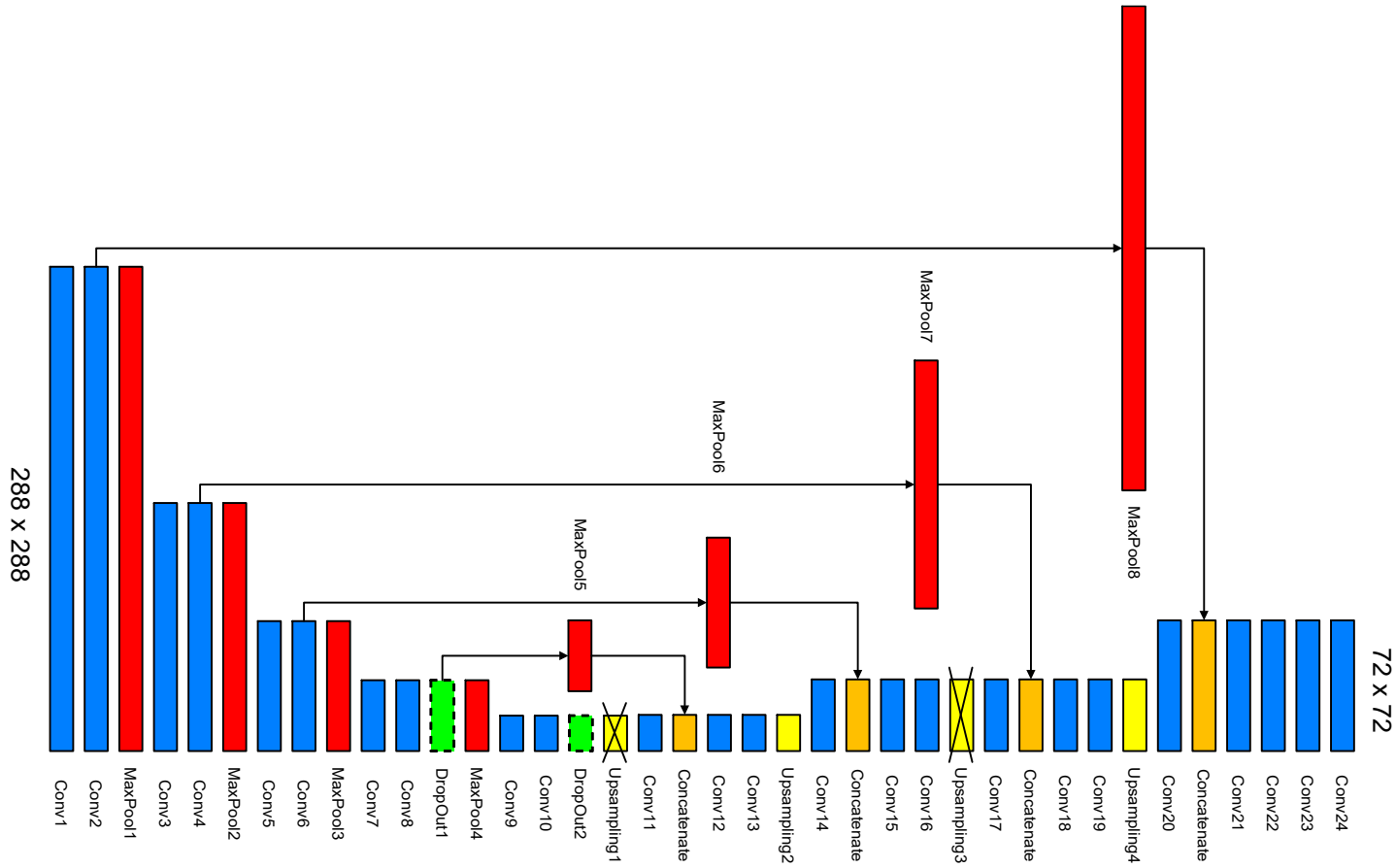

Figure 3: The development of the proposed topology of a fully convolutional neural network - layers removed from the original U-Net are crossed

U-Net is a fully convolutional neural network proposed initially for image segmentation tasks in biological and medical fields. However, it was then adapted to many other applications across different fields. This topology follows a typical encoder-decoder scheme with a bottleneck. In addition, it also contains a direct link between the parts of the encoder and the decoder, which is allowing the network to propagate contextual information to higher resolution layers. U-net topology is adopted from [19].

We streamline U-Net to fit our task in a more efficient way. To be more specific, U-Net originally provides the output data of the same dimensions as the input data. Such a detailed output is not required for a "pick and place" problem in industrial applications. Hence, we reduce the decoder part of U-Net topology. In our case, the output data is 16 times lower, which still provides an accuracy sufficient enough for object detection and positioning, and the topology itself is less computationally demanding. See Figure 3, where the changes applied to U-Net topology are shown in detail.

3.4. Locator for positioning of grasp points

Positioning of the gradient circles in a schematic image (last step in Figure 2) is a generic process of finding local maxima of each implemented color. These positions of the maxima directly represent the grasp points for manipulation using a robotic arm.

Generally, the process of finding local maxima in an array can be performed in several ways. The most obvious solution is to find the indices of the values, which are greater than all their neighbors. However, this approach is very sensitive to noise or small errors in the input data. Hence, it is more appropriate to implement a maximum filter operation, which dilates the original array and merges neighboring local maxima closer than the size of the dilation. Coordinates, where the original array is equal to the dilated array, are returned as local maxima. Clearly, the size of the dilation must be set. In the proposed locator, it is suitable to set it equal to the radius of the gradient circles.

4. Case Study

The aim of this section is to demonstrate the features of our detection and positioning system through the solution of a particular task. The task is properly defined in the next subsection. Subsequently, we propose a hardware implementation of the system and, as the final step of the procedure, a fully convolutional neural network is trained to be able to transform the original RGB images into a graphical representation of object types and positions.

4.1. Object detection and positioning task

For this case study, we need to develop a system for different object detection and positioning. The four object combinations placed on five different types of surfaces were used. The objects of interest are shown in Figure 4 and the surfaces are shown in Figure 5. The objects are of a similar size. Three of them are metallic and one is of black plastic.

4.2. Hardware implementation

The system is composed from a camera sensor and a processing unit, which should process data acquired by the RGB sensor, in order to determine the types and positions of the objects of interest. In this case study, we implement a Basler acA2500-14uc industrial RGB camera as a data acquisition tool. This sensor is able to provide up to 14 5-MPx RGB frames per second. The camera is equipped with a Computar M3514-MP lens in order to monitor the 300 x 420 mm scan area from above at a distance of 500 mm.

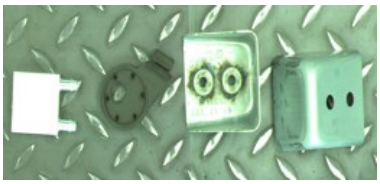

Figure 4: Objects of interest - objects are labelled as Obj1 to Obj4

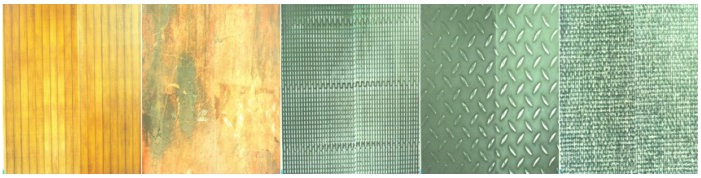

Figure 5: Various types of surfaces used in case study

The processing unit is supposed to be capable of processing images in real-time, as mentioned above. For this case study, the single-board NVIDIA Jetson NANO computer is used. It offers the NVIDIA Tegra X2 (2.0 GHz, 6 cores) CPU together with 8 GiB RAM. Furthermore, it provides wide communication possibilities (USB 2.0, 3.0, SATA, WiFi). The total price of components used for hardware implementation costs around $500.

4.3. Datasets

In this case study, we prepare 1021 original RGB images using various combinations of objects (Figure 4) and surfaces (Figure 5). In order to follow the topology of the CNN (Figure 3), we transform the images to 288 × 288 px. Then, we randomly divide the images into the training set (815 samples) and the testing set (206 samples).

After that, the trickiest part of the development follows. The target images for the training set (the graphical representations of the object types and positions) should be manually prepared. Hence, for any RGB image, we construct a target artificial image, where the optimal grasp point of each individual object in the original image is highlighted by a colored gradient circle. Four colors (red, green, blue and white) are implemented, since we consider four types of objects in this case study. We prepared a custom labeling application to prepare target images. In this application, each input image is displayed, and a human user labels all feasible grasp points using a computer mouse. The application then generates the target images. Several examples of input-expected output pairs are demonstrated in Figure 6.

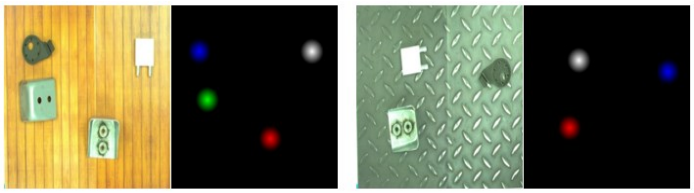

Figure 6: Two input-target pairs in training set - input image resolution is 288x288 px, target image resolution is 72x72 px

4.4. Fully convolutional neural network training

As the last step of the development, we train the network topology depicted in Figure 3.

We select the ADAM algorithm for the neural network weights and biases optimization as it is generally considered as an acceptable performing technique in most of the cases [20]. The random initial weights set with gaussian distribution was used. The experiments are run fifty times in order to reduce the stochastic character of training. The best instance is then evaluated. The training process and its parameters are depicted in Figure 7.

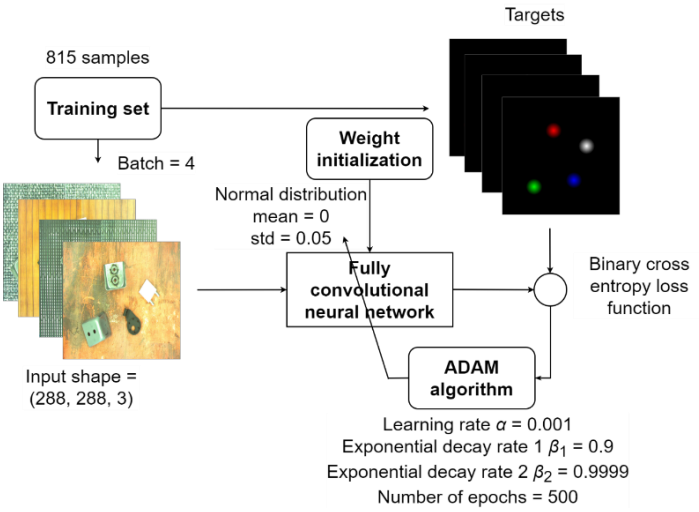

Figure 7: Training process

4.5. Results

We select a confusion matrix as an evaluation metric for detection and positioning system performance. The prediction of the type of the object and position of the grasp point is labeled as correctly predicted, if the local maximum position of the gradient circle of the defined color directly corresponds to the original position of the particular object using the 72 x 72 px map, i.e. the map defined by the target image.

The confusion matrix for the best network trained according to the previous paragraph for the testing set, is summarized in Table 1. Note that the number of correctly predicted free spaces in the image is not present, because it is essentially a black surface that cannot be explicitly evaluated. However, as seen in the table, the detection and positioning system provides 100 % accuracy over the testing set. In addition, implementing Jetson NANO described above, the detection and positioning system is capable of processing 13 frames per second, which is more than required at the beginning of the paper.

Table 1: Confusion matrix (206 images, 611 objects in total)

-	Obj1 pred.	Obj2 pred.	Obj3 pred.	Obj4 pred.	Free space pred.
Obj1 actual	149	0	0	0	0
Obj2 actual	0	153	0	0	0
Obj3 actual	0	0	158	0	0
Obj4 actual	0	0	0	151	0
Free space actual	0	0	0	0	Irrelevant

5. Conclusion

In this contribution, we introduced a novel engineering approach to object grasp point detection and positioning for "pick and place" applications. The approach is based on two consecutive steps. At first, a fully convolutional neural network is implemented in order to transform the original input image of the monitored scene. Output of this process is a graphical representation of the types and positions of the objects present in the monitored scene. Secondly, the locator is used to analyze the graphical representation to get the explicit information of object type and position.

We also performed a case study to demonstrate the proposed approach. In this study, the proposed system provided accurate grasp point positions of four considered objects for manipulation.

In future work, we will try to enhance the system in several ways. Critically, the approach should provide not only the positions of the grasp points, but also the required pose of the robotic arm. This feature will be advantageous especially for clamp grippers. Apart from that, we will try to optimize the processing unit, both from the hardware and software point of view, in order to get the close-to-optimal topology of the fully convolutional neural network and the hardware suitable for it.

Conflict of Interest

The authors declare no conflict of interest.

Acknowledgment

The work was supported from ERDF/ESF " Cooperation in Applied Research between the University of Pardubice and companies, in the Field of Positioning, Detection and Simulation Technology for Transport Systems (PosiTrans)" (No. CZ.02.1.01/0.0/0.0/17_049/0008394).

References

[1] Y. Cheng, M.A. Jafari, "Vision-Based Online Process Control in Manufacturing Applications," IEEE Transactions on Automation Science and Engineering, **5**(1), 140-153, 2008, doi:10.1109/TASE.2007.912058.

[2] M. Bahaghighat, L. Akbari, Q. Xin, "A Machine Learning-Based Approach for Counting Blister Cards Within Drug Packages," IEEE Access, **7**, 83785-83796, 2019, doi:10.1109/ACCESS.2019.2924445.

[3] H. Sheng, S. Wei, X. Yu, L. Tang, "Research on Binocular Visual System of Robotic Arm Based on Improved SURF Algorithm," IEEE Sensors Journal, **20**(20), 11849-11855, 2020, doi:10.1109/JSEN.2019.2951601.

[4] S. Caldera, A. Rassau, D. Chai, "Review of Deep Learning Methods in Robotic Grasp Detection," Multimodal Technologies and Interaction, **2**(3), 2018, doi:10.3390/mti2030057.

[5] R. Miyajima, "Deep Learning Triggers a New Era in Industrial Robotics," IEEE MultiMedia, **24**(4), 91-96, 2017, doi:10.1109/MMUL.2017.4031311.

[6] S. Li, W. Song, L. Fang, Y. Chen, P. Ghamisi, J.A. Benediktsson, "Deep Learning for Hyperspectral Image Classification: An Overview," IEEE Transactions on Geoscience and Remote Sensing, **57**(9), 6690-6709, 2019, doi:10.1109/TGRS.2019.2907932.

[7] F. Xing, Y. Xie, H. Su, F. Liu, L. Yang, "Deep Learning in Microscopy Image Analysis: A Survey," IEEE Transactions on Neural Networks and Learning Systems, **29**(10), 4550-4568, 2018,

doi:10.1109/TNNLS.2017.2766168.
[8] A, Björnsson, M. Jonsson, K. Johansen, "Automated material handling in composite manufacturing using pick-and-place systems – a review," Robotics and Computer-Integrated Manufacturing, **51**, 222-229, 2018, doi:10.1016/j.rcim.2017.12.003.
[9] P. Dolezel, Petr, D. Stursa, D. Honc, "Rapid 2D Positioning of Multiple Complex Objects for Pick and Place Application Using Convolutional Neural Network," in 2020 24th International Conference on System Theory, Control and Computing (ICSTCC), 213-217, 2020, doi:10.1109/ICSTCC50638.2020.9259696.
[10] C. Papaioannidis, V. Mygdalis, I. Pitas, "Domain-Translated 3D Object Pose Estimation," IEEE Transactions on Image Processing, **29**, 9279-9291, 2020, doi:10.1109/TIP.2020.3025447.
[11] J. Pyo, J. Cho, S. Kang, K. Kim, "Precise pose estimation using landmark feature extraction and blob analysis for bin picking," in 2017 14th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI), 494-496, 2017, doi:10.1109/URAI.2017.7992786.
[12] J. Kim, H. Kim, J.-I. Park, "An Analysis of Factors Affecting Point Cloud Registration for Bin Picking," in 2020 International Conference on Electronics, Information, and Communication (ICEIC), 1-4, 2020, doi:10.1109/ICEIC49074.2020.9051361.
[13] P. Dolezel, J. Pidanic, T. Zalabsky, M. Dvorak, "Bin Picking Success Rate Depending on Sensor Sensitivity," in 2019 20th International Carpathian Control Conference (ICCC), 1-6, 2019, doi:10.1109/CarpathianCC.2019.8766009.
[14] S. Krebs, B. Duraisamy, F. Flohr, "A survey on leveraging deep neural networks for object tracking," in 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC), 411-418, 2017, doi:10.1109/ITSC.2017.8317904.
[15] Y. Xu, X. Zhou, S. Chen, F. Li, "Deep learning for multiple object tracking: a survey," IET Computer Vision, **13**(4), 355-368, 2019, doi:10.1049/iet-cvi.2018.5598.
[16] K. He, X. Zhang, S. Ren, J. Sun, "Deep Residual Learning for Image Recognition," in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 770-778, 2016, doi:10.1109/CVPR.2016.90.
[17] V. Badrinarayanan, A. Kendall, R. Cipolla, "SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation," IEEE Transactions on Pattern Analysis and Machine Intelligence, **39**(12), 2481-2495, 2017, doi:10.1109/TPAMI.2016.2644615.
[18] H. Zhao, J. Shi, X. Qi, X. Wang, J. Jia, "Pyramid Scene Parsing Network," in 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 6230-6239, 2017, doi:10.1109/CVPR.2017.660.
[19] O. Ronneberger, P. Fischer, T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," in Medical Image Computing and Computer-Assisted Intervention (MICCAI), 234-241, 2015, doi:10.1007/978-3-319-24574-4_28.
[20] E.M. Dogo, O.J. Afolabi, N.I. Nwulu, B. Twala, C.O. Aigbavboa, "A Comparative Analysis of Gradient Descent-Based Optimization Algorithms on Convolutional Neural Networks," in 2018 International Conference on Computational Techniques, Electronics and Mechanical Systems (CTEMS), 92-99, 2018, doi: 10.1109/CTEMS.2018.8769211.