"Instrumentation Engineering, Electronics and Telecommunications – 2019"
Proceedings of the V International Forum (Izhevsk, Russia, November 20–22, 2019)

DOI: 10.22213/2658-3658-2019-66-75

An Affordable Deep Learning Based Solution to Support Pick and Place Robotic Tasks

M. Mahmoodpour [1], A. Lobov[1,2], S. Hayati [3], A. Pastukhov [4]

[1] Automation Technology and Mechanical Engineering, Tampere University, Tampere, Finland
E-mail: {mehdi.mahmoodpour, andrei.lobov}@tuni.fi
[2] Dept. of Mechanical and Industrial Engineering, Norwegian University of Science and Technology, Trondheim, Norway
E-mail: andrei.lobov@ntnu.no
[3] Software Development Department, Elektrobit Automotive GmbH, Oulu, Finland
E-mail: saboktakin.hayati@elektrobit.com
[4] International Research Centre "Biotechnologies of the Third Millennium", ITMO University, Saint-Petersburg, Russia
E-mail: artem.pastukhov1984@gmail.com

Received: July 15, 2019

Abstract. In the current competitive market, the lack of financial resources is a major challenge for Small and Medium Enterprises (SMEs). As a result, SMEs seek low-cost technologies to be employed in their enterprises. Moreover, robotics together with vision systems have become an indispensable part of the current production systems regarding their capabilities to improve the productivity and performance of manufacturing processes. The aim of this paper is to describe an affordable solution for the pick and place robot operation using computer vision technology. The application is designed to identify, pick and place the objects with different arrangements on the palette. For the fulfillment of the vision system, the latest deep learning techniques and image processing software is used to detect the parts. Moreover, the proposed application is dynamic and scalable, created for different use cases so that the operation for various parts with diverse shapes can be handled. The application is tested and validated at the Tampere University Robotic Laboratory (Robolab).

Keywords: computer vision, Convolutional neural networks, deep learning, robot vision

INTRODUCTION

In the current manufacturing enterprises, rapid reconfiguration and quick response to the frequent changes is a critical demand in order to meet the customers' expectations [1]. The inclusion of advanced technologies can help firms to respond to this demand and support the benefits of the enterprise more effectively. On the other hand, since the SMEs have budget constraints to spend in technological infrastructure, they look for solutions that need less financial investment and can fit with their current infrastructure seamlessly. Nevertheless, the advancement of technology has brought low-cost modern technologies in existence over recent years, which allows SMEs to stay in the market and grow more quickly. The other ad-

vantage associated with affordable technologies is that more people are able to launch their startups easily and contribute to the economic growth of the global market.

From the first day the robots were employed in the manufacturing industries, they have made major changes to the production management in industrial sectors with high accuracy and reliability demand. Especially in today's fast-paced production environments, robots play a pivotal role in the industrial automation domain due to their remarkable performance in boosting the efficiency and quality of manufacturing processes as well as lowering the operation costs. As a result, SMEs have become a target for robotic technologies vendors helping them to utilize robots in their manufacturing operations. Thus, robot vendors have introduced the new generation of robots that can fit into small spaces of small manufacturers premises. Collaborative robots or "cobots" are good examples that have enabled SMEs to take advantage of robotic to improve their productivity. Moreover, the adoption of robotic vision systems has been acknowledged by industry sector in recent years. Over the past years, there has been a considerable improvement in the performance of vision technologies while the cost of using such technologies has dropped dramatically. Furthermore, the integration of vision technology with robotic operations can significantly increase the productivity of robots. According to above-mentioned issues, robotic applications including vision technology should be enough affordable for SMEs so that they can leverage the benefits come with this technology without being concerned to allocate a significant financial budget. Also, the smooth integration of machine vision technology with the current robotic operations of SMEs is another crucial aspect that should be taken into consideration. In this context, any machine vision solution should be developed in a way that the commissioning and installing can be carried out simply and quickly by the field operators of SMEs without special skills. Furthermore, the machine vision module should be standalone and easy to maintain. In addition, particularly for pick and place operation, the machine vision module should be flexible enough so that can be reconfigured quickly to support pick and place operation of the wide range of parts with different configurations for multiple use cases.

In this paper, we have focused on low-cost vision systems for SMEs to implement the pick and place robotic operations. The proposed modular and dynamic application allows the SMEs to integrate the solution to their robotic manipulators smoothly without the need to make major changes in current systems. The hardware we used, as computation resource to accomplish our solution, is Raspberry Pi 3, a single-board computer that represents excellent value for a small cost.

The rest of the article is organized as follows. First, the literature review of computer vision systems and techniques is carried out. Next, the state-of-the-art computer vision for robotics is provided. Then, the methodology of research is discussed, followed by the implementation and results with the verification of the proposed solution. Finally, the last section concludes the paper and presents the future direction of research.

COMPUTER VISION SYSTEMS / TECHNIQUES

The computer vision is defined as "extracting descriptions of the world from pictures or sequences of pictures" [12]. In general, any computer vision process consists of three main steps: (1) Image acquisition; (2) Image processing; (3) Image analysis and decision-making. In the following, a brief review of each step is provided.

Image acquisition, which is defined as retrieving an image from hardware-based sources such as different kinds of cameras to represent the real-world scene as digital data to be used for further processing [9]. The hardware system for image acquisition uses image sensors to convert ambient light into digital signals, which eventually can be stored and represented as

digital images. Technically, in image processing, digital images are defined as an array of pixels in a two-dimensional matrix, where each pixel represent the intensity value of brightness [13].

Image processing, the acquired image in the first step could be not of sufficiently high quality because of imaging conditions or problems related to the storage of images [9]. As a result, a pre-processing mechanism is required to enhance the quality of captured images; compress the size of the image; perform image restoration, and carry out feature extraction by employing complex algorithms for better human perception and machine interpretation [23]. The image processing process can be defined as an input-output system in which the image processor applies sophisticated operations on the acquired image in order to generate a high-quality image as an output [16]. Typically, this type of image processing which involves primitive operations to enhance the quality of the image is known as low-level processing.

Image analysis and decision-making means the analysis of images and extract information by which the actual decision can be made. Mid-level image processing techniques such as edge detection and segmentation are used to derive features and attributes from the input image. The extracted features allow computer vision system to derive meaningful information from images, and by feeding achieved information to Machine Learning (ML) algorithms and Artificial Intelligence (AI) technology, cognitive insight can be drawn from the input image and proper action can be taken accordingly i.e. high-level image processing. For instance, with the help of mid-level image processing algorithms such as segmentation, the attributes of a scene can be extracted to enable the high-level process system to understand the surrounding environment using AI algorithms for autonomous navigation. Fig. 1 [3] illustrates the steps of a computer vision system discussed.

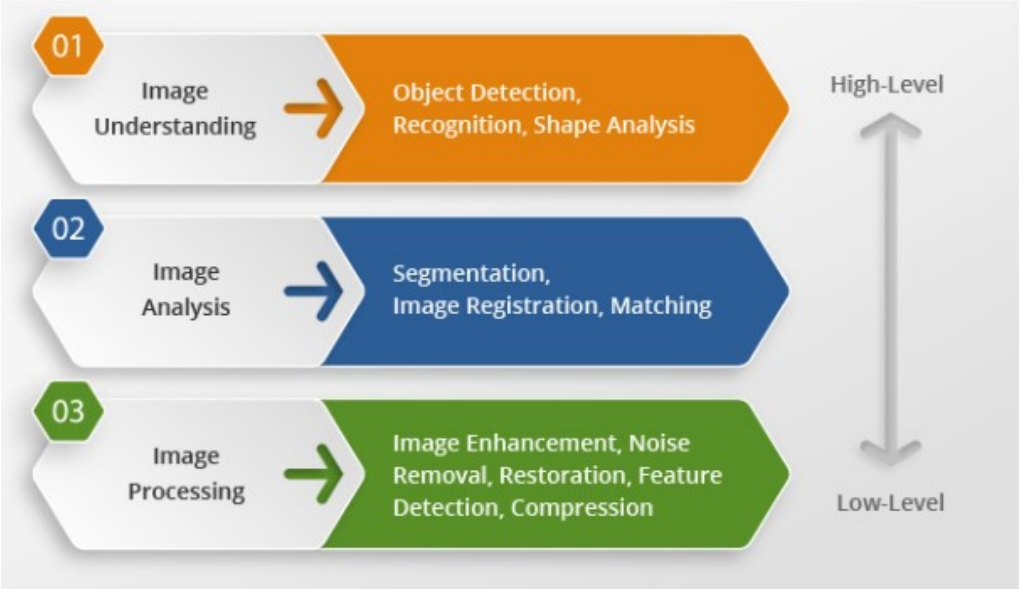


Figure 1. The architecture of a typical computer vision system

ML as a sub-field of AI enables the system to automatically learn from previous data through detecting pattern on data and make a prediction about future data [4]. In this context, computer vision has been influenced by machine learning dramatically over the past 20 years [17]. Machine learning provides powerful techniques for computer vision for adjusting the parameters and using the learned experience to generate, validate and tune the hypothesis [26]. These techniques can be categorized into two major categories: deep learning-based and

traditional machine learning algorithms. Examples of the latter include, random forest (RF) [5], support vector machine (SVM) [18], Hidden Markov model (HMM) [28] and so on. Further, the former category, deep learning, can facilitate feature extraction in the different representation of data by sampling multi-level abstractions of input data, examples include Convolutional Neural Network (CNN) [20] and Recurrent Neural Network (RNN) [22], etc. While there are other deep learning algorithms with a generative approach, such as Generative Adversarial Network (GAN) [14] and Variational Autoencoders (VAE) [24]; they build a model based on simulated observations that extracted from a probability density function [8]. Additionally, deep learning handles large scale datasets such as ImageNet with satisfactory performance [11].

COMPUTER VISION SYSTEMS FOR ROBOTIC APPLICATIONS

The adoption of computer vision for industrial application and particularly robotics began from the early 1980s [27], thus enabling computers to gain a visual understanding of the surrounding environment through extracting information from digital images of the real world [15]. The development of computer vision technology is growing fast according to emerging low-cost cameras, affordable processing power and evolving vision algorithms [7]. The applications of vision technology for robots in manufacturing processes include identification and locating parts, inspection, assembly, quality control, human-robot collaboration, track the objects, robot navigation and etc. The vision system for robotics can be either scene-related or object-related [2]. The scene-oriented applications involve pathfinding, obstacle avoidance, localization, mapping for mobile robots. On the other hand, object-related vision systems are used to detect objects for different applications such as pick and place, material assembly, quality inspection, machine tending, etc.

In recent years, the usage of deep learning for robot vision applications has become a subject undergoing intense study in robotics realm. The comprehensive survey conducted by authors in [25] reviews the latest achievements and advances in deep learning based robot vision system.

METHODOLOGY

As already discussed in the introduction section, any system which aims to enable SMEs to leverage from advanced technologies should be enough affordable. In this study, this essential prerequisite was taken into consideration in order to allow SMEs to employ the designed system in their manufacturing processes. In this context, the hardware and software of the system are chosen in a manner that the entire system is affordable.

Hardware selection: The hardware used to perform system computations, should be capable of carrying out all the tasks defined within the application. Currently, there are a couple of budget computers available in the market. One of the most widely used single-board computers is Raspberry Pi 3 (RP), which can be used for a variety of functions such as automation projects, Internet of Things, educational purposes, industrial applications, etc. Moreover, RP benefits from strong community support, meaning there are many tutorials, resources, and guides provided by contributors all around the world. Also, RP provides a set of general-purpose input/output (GPIO) pins and different ports in order to support the connectivity of a wide range of accessories and peripherals, making it a versatile standalone computer to interact with real-world applications. In particular, for a computer vision application, RP can be equipped with USB webcam or Raspberry Pi camera module making RP capable of handling image acquisition and computer vision algorithms. Furthermore, RP supports all necessary

machine learning and deep learning libraries which are needed to implement a computer vision application.

Software Design: For the software part of the project, as the aim of the research is to enable the integration of vision-system with different robots in a smooth manner, the system is designed in such a way that each module in the system works independently from other modules. This loosely coupled architecture improves the flexibility to the system, thus enabling the development of new modules (e.g. controller for different brands of robots) in an extensible and reusable way. Figure 2 illustrates the designed architecture for the application.

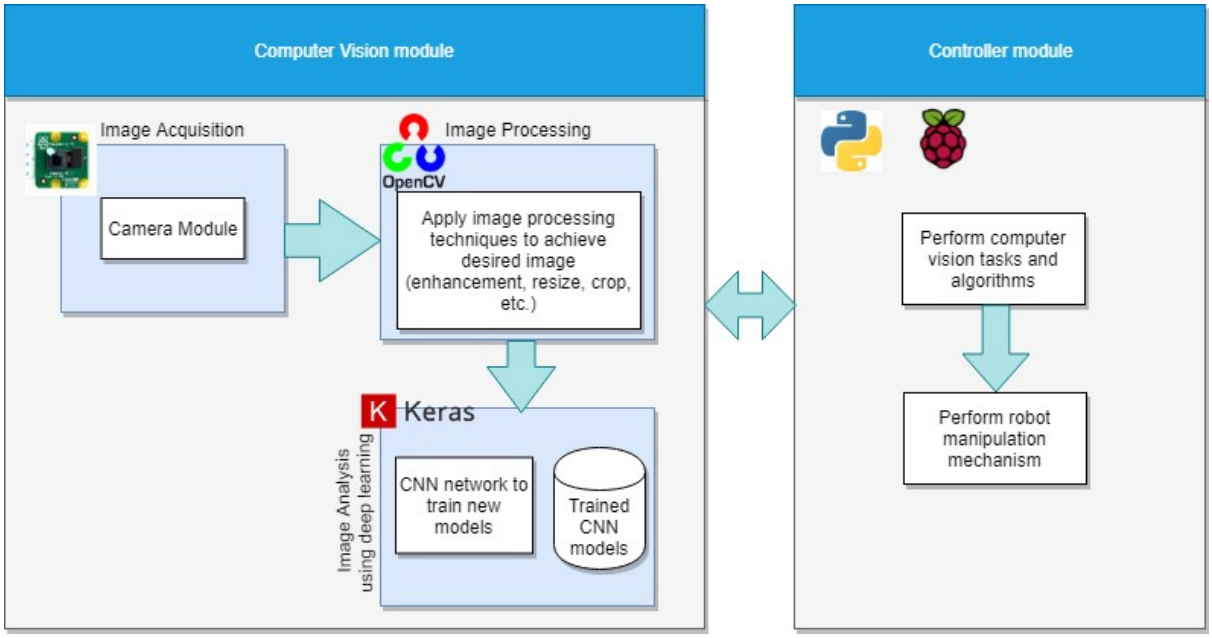


Figure 2. Architectural view of the designed system

As can be seen in the system architecture, the computer vision module uses camera hardware to acquire an image from the external world. Then, the image processing techniques are applied to the acquired image according to the specification of CNN classifier. Next, for the image analysis purpose, the processed image is fed to the CNN network to extract meaningful information from the input image. The controller module interacts with the computer vision system to retrieve the required information in order to carry out the robot manipulation mechanism.

In this study, different tools and libraries are used to perform image processing and deep learning algorithms for the vision part of the project which is discussed briefly in the following.

OpenCV: OpenCV framework [6] is a powerful open source computer vision library which facilitates conducting image processing techniques and algorithms by providing built-in functions. OpenCV is a well-documented library which provides C++, Python, Java and MATLAB interfaces, and supports multiple operating systems such as Windows, Linux, Mac OS, and Android.

Keras: Deep learning techniques have empowered computer vision algorithms significantly. Keras [10] is an open-source high-level neural networks library that allows developing deep learning models for computer vision applications in an easy-to-use manner. It was designed with the aim of enabling fast experimentation with neural networks. Keras has recently integrated into TensorFlow [21], thus allowing using TensorFlow functionalities within Keras

if needed. It is written in Python and runs on CPU and GPU. Moreover, it supports both convolutional networks and recurrent networks, as well as the combinations of the two.

IMPLEMENTATION

In this section, the technical implementation of the proposed methodology for the validation purpose is described.

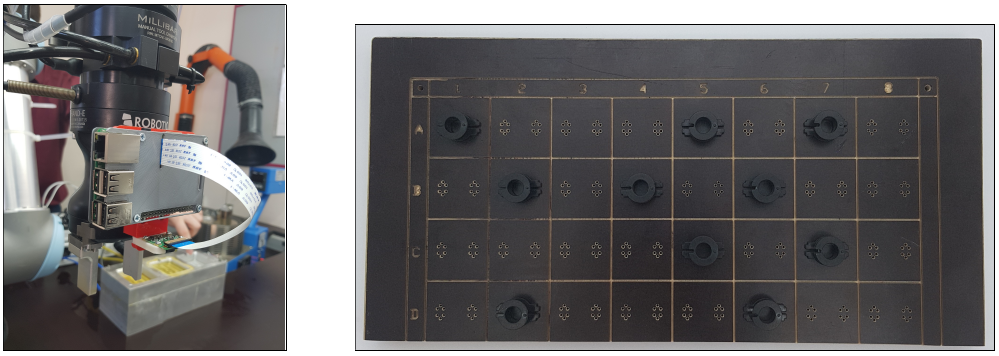
Vision system hardware: As was stated in the methodology section, Raspberry Pi provides reasonable computing resources with an affordable price. In this study, the Raspberry Pi 3 Model B+ was selected as hardware to perform experiments. Raspberry Pi 3 Model B+ is the latest product in RP 3 series with a 1.4 GHz 64-bit quad-core processor, 1 GB RAM, and support for network communication via Ethernet and wireless LAN. Moreover, the 8-megapixel Raspberry Pi Camera Module v2 was used for image acquisition. All necessary tools and software related to computer vision and deep learning were installed on RP.

Robot setup: The Tampere RoboLab [19] was the place, where all the experiments of study were conducted there. Tampere RoboLab is the learning environment established at Tampere University which aims to provide real-life equipment for pedagogical purposes and particularly robotics education. In this study, for robotic pick and place task, the UR5 robot from Universal Robot in RoboLab was chosen to conduct the experiments in order to validate the solution. The UR5 is a medium size and light-weight collaborative robot (cobot) which widely is used by manufacturers, and particularly SMEs for repetitive tasks such as picking, placing and testing according to its affordable cost compared to other similar products. It can be integrated seamlessly with the working environment through connecting external sensors and actuators as well as other external resources such as, for instance, machine vision system. It supports multiple communication protocols to interact with the external world and remote control such as Modbus and TCP/IP. The communication method we used to control UR5 remotely was TCP/IP socket connection via Ethernet. To do this, the client program needs to run on an external device which in our case is RP and the URScript commands are sent to a server hosted on the robot. We used TCP port 30002 on the robot to send commands over a TCP/IP socket from RP and control the robot remotely. The IP address was configured for both server and client considering that they must be in the same subnet. In this case, RP mounted on the UR5 robot can handle the entire robot control process. Also, to integrate the vision system hardware with the robot, the RP and camera module was mounted on the robot and near to end effector. For this purpose, a plastic holder manufactured by the 3D printer was used to fit the RP and camera on the robot (Figure 3.a).

Using pallets for the pick and place operations results in shorter cycle times according to the repeatability and precision of work. In this regard, a sample indexing table with 32 workpiece positions was used for the experiments. The indexing table (Figure 3.b) is designed specifically for Switch Mode Power Supply (SMPS) transformer coil to handle the pick and place task for soldering operations.

The approach we took for the fulfillment of the vision system to identify objects with different arrangements on the indexing table was based on the implementation of image classification using deep learning. The CNN image classifier allows us to detect the objects on the indexing table for further robotic operations. The image classification model we built using Python and Keras, takes an input (i.e. image) and outputs a class, which in our case the labels are "empty" and "filled". This way, we can classify the given image of every single cell of indexing table using our trained model and predict whether there is a part on cell or not. In order to train our image classifier model, we built a dataset divided into two subsets of images:

- **Empty:** containing the images of empty cells (1000 samples, 28×28 images).
- **Filled:** containing the images of filled cells (1000 samples, 28×28 images).



(a) (b)

Figure 3. Mounting RP on the robot (a); indexing table for SMPS transformer coils (b)

The images were captured in various illumination conditions to build a rich dataset in order to improve the generalization of the trained model. We designed feedforward neural network architecture with backpropagation to train our model. The designed CNN network consists of multiple layers including two convolutional layers, one dense hidden layer and the output layer with two labels. The training was carried out by RP and the trained model was stored on RP for later use in the application.

Figure 4. The plot of model accuracy/loss on training

In order to see the training performance of deep learning classifier model, the results are plotted as shown in Figure 3. As can be seen, the accuracy and loss parameters for training and validation converge during the training process, which implies that the model is well generalized and overfitting is handled properly. The network trained for 20 epochs and we achieved 98.91 % testing accuracy. For the training/testing split on data, we dedicated 80 % of the images for training and the rest for testing.

The trained classification model can be loaded in order to enable the application to detect parts. In the following, we explain the details of how the computer vision module works in the system. Figure. 5 illustrates the sequence diagram of the interaction between the different components of the system to accomplish pick and place task using computer vision.

Once the trained CNN model is loaded, the image from the work cell is captured by the camera module. Next, the indexing table is extracted from the captured image using OpenCV edge detection functions such as Canny, Dilate and Erode. Then, we use OpenCV functions to loop over the indexing table image and crop the image of each cell to be examined by image classifier and predict the emptiness or fullness of cell. The results are stored in memory for further use by the robot.

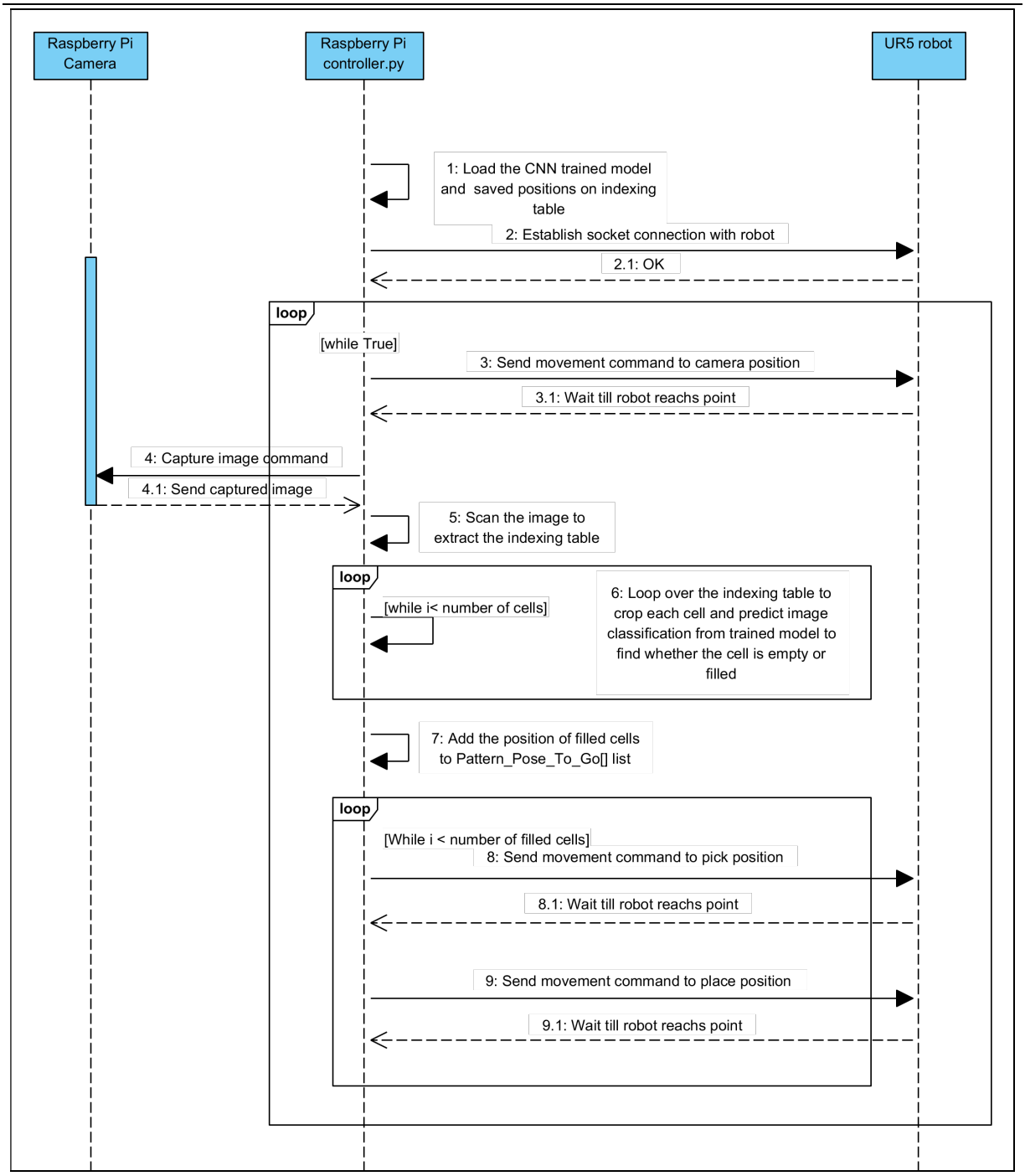


Figure 5. Sequence diagram of system components interactions

For each indexing table, the position of cells is stored in a CSV file using the UR5 palletizing wizard. This file is loaded by application to be used for robot manipulation tasks. Next, according to the results achieved from the computer vision module, a list, containing the position of filled cells is created that the robot should pick the part. Then, RP sends movement commands to robot regarding the created list. The operation continues until the robot reaches all the cells and performs the pick and place task.

RESULTS

The vision system integrated with a robot was examined at Tampere University RobLab. In order to ensure the proper functioning of the designed system, several experiments were carried out with different arrangements of parts on the indexing table. Also, the tests were fulfilled in different illumination conditions to test the performance of the system. The test results were satisfactory and the vision system detected the parts with 100 % accuracy. The performance of the trained classifier was flawless because of having a rich dataset of images for the specific part (i.e. 2000 images in total). However, it should be noted that since there are many parts with different shapes and sizes, building the image dataset for all the parts would be a tedious and time-consuming process. Accordingly, an automated process to create the image dataset is in need to handle this issue.

CONCLUSION

In this paper, we discussed the necessity of allowing the SMEs to leverage the latest technologies in order to be able to compete with larger corporations. In this context, we explained that affordable robotic vision systems could contribute to SMEs' productivity improvement significantly and help them to stay in the market. Moreover, we studied the state-of-the-art technologies and tools in computer vision domain and their usage in robotics. We proposed an affordable solution using Raspberry Pi as a cheap and flexible computer, capable of handling the computational requirements of moderate deep learning applications, which is sufficient for SMEs manufacturing processes. The application was tested and results proved that RP is able to perform adequately for deep learning-based vision system. The proposed application is scalable, created for different use cases so that the pick and place task of various parts with different shapes and sizes can be handled. For the future phase of the research, we plan to implement the solution on other available affordable computers in the market and study the performance benchmarks of each device. Moreover, the implementation of the automated process to create image dataset for training the image classifier model will be taken into consideration.

REFERENCES

1. Abele, E., Meyer, T., Näher, U., Strube, G., & Sykes, R. (Eds.). (2008). *Global production: A handbook for strategy and implementation.* Springer Science & Business Media.
2. Alenyà, G., Foix, S., & Torras, C. (2014). ToF cameras for active vision in robotics. *Sensors and Actuators A: Physical, 218*, 10–22. doi: 10.1016/j.sna.2014.07.014
3. *Application for image processing.* (2019). Retrieved May 14, 2019, from: http://www.smartcoreinc.com/?page_id=2121&ckattempt=1 .
4. Bishop, C. M. (2006). *Pattern recognition and machine learning.* Springer.
5. Bosch, A., Zisserman, A., & Munoz, X. (2007). Image classification using random forests and ferns. In: *2007 IEEE 11th International Conference on Computer Vision* (pp. 1-8). Rio de Janeiro, Brazil: IEEE. doi: 10.1109/ICCV.2007.4409066
6. Bradski, G. (2000). The OpenCV Library. *Dr. Dobb's Journal of Software Tools, 25*(11), 120–126.
7. Bradski, G., & Kaehler, A. (2008). *Learning OpenCV: Computer vision with the OpenCV library.* O'Reilly Media, Inc.
8. Cao, Y, Jia, L., Chen, Y., Lin, N., Yang, C., Zhang, B., Liu, Z., Li, X., & Dai, H. (2019). Recent advances of generative adversarial networks in computer vision. *IEEE Access, 7*, 14985–15006. doi: 10.1109/ACCESS.2018.2886814

9. Chan, T. F., & Shen, J. 2005. *Image processing and analysis: Variational, PDE, wavelet, and stochastic methods.* Siam.
10. Chollet, F., & others. 2015. *Keras*.
11. Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., & Li Fei-Fei. (2009). ImageNet: a large-scale hierarchical image database. In: *2009 IEEE Conference on Computer Vision and Pattern Recognition* (pp. 248–255). Miami, USA: IEEE. doi: 10.1109/CVPR.2009.5206848
12. Forsyth, D. A., & Ponce, J.. (2003). *Computer vision: A modern approach.* Prentice-Hall.
13. Gonzalez, R. C., & Woods, R. E. (2002). *Digital image processing.* Prentice Hall.
14. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair S., Courville, A., & Bengio, Y. (2014). Generative adversarial nets. In: *Advances in Neural Information Processing Systems, 27,* 2672–2680.
15. Hartley, R., & Zisserman, A. (2003). *Multiple view geometry in computer vision.* Cambridge university press.
16. Jain, A. K. (1989). *Fundamentals of digital image processing.* Englewood Cliffs, USA: Prentice Hall.
17. Jordan, M. I., & Mitchell, T. M. (2015). Machine learning: Trends, perspectives, and prospects. *Science, 349*(6245), 255–260. doi: 10.1126/science.aaa8415
18. Ko, B. C. Cheong, K.-H., & Nam, J.-Y. (2009). Fire detection based on vision sensor and support vector machines. *Fire Safety Journal, 44*(3), 322–329. doi: 10.1016/j.firesaf.2008.07.006
19. Lanz, M., Pieters, R., & Ghabcheloo, R. (2019). Learning environment for robotics education and industry-academia collaboration. *Procedia Manufacturing 31,* 79–84. doi: 10.1016/j.promfg.2019.03.013
20. LeCun, Y., Kavukcuoglu, K., & Farabet, C. (2010). Convolutional networks and applications in vision. In: *Proceedings of 2010 IEEE International Symposium on Circuits and Systems* (pp. 253–256). Paris, France: IEEE. doi: 10.1109/ISCAS.2010.5537907
21. Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Zh., Citro, C., Corrado, G. S., et al. (2015). *TensorFlow: Large-scale machine learning on heterogeneous systems.* Retrieved May 14, 2019, from: https://www.tensorflow.org/
22. Mnih, V., Heess, N., Graves, A., & Kavukcuoglu, K. (2014). Recurrent models of visual attention. In: *NIPS'14 Proceedings of the 27th International Conference on Neural Information Processing Systems, 2* (pp. 2204–2212). Montreal, Canada. Retrieved May 14, 2019, from: http://papers.nips.cc/paper/5542-recurrent-models-of-visual-attention.pdf
23. Petrou, M., & Petrou, C. (2010). *Image processing: The fundamentals.* 2nd ed. John Wiley & Sons. doi: 10.1002/9781119994398
24. Pu, Yu., Gan, Z., Henao, R., Yuan, X., Li, Ch., Stevens, A., & Carin, L. (2016). Variational autoencoder for deep learning of images, labels and captions. In: Lee, D. D., Sugiyama, M., Luxburg, U. V. (Eds.). *Advances in Neural Information Processing Systems 29 (NIPS 2016),* 2352–2360.
25. Ruiz-del-Solar, J., Loncomilla, P., & Soto, N. (2018). A survey on deep learning methods for robot vision. *ArXiv, abs/1803.10862*.
26. Sebe, N., Cohen, I., Garg, A., & Huang, T. S. (2005). *Computational imaging and vision: Vol. 29. Machine learning in computer vision.* Springer Science & Business Media. doi: 10.1007/1-4020-3275-7
27. Vernon, D. (1991). *Machine vision: Automated visual inspection and robot vision.* Prentice Hall.
28. Yamato, J., Ohya, J., & Ishii, K. (1992). Recognizing human action in time-sequential images using hidden Markov model. In: *Proceedings 1992 IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (pp. 379–385). Champaign, USA: IEEE. 10.1109/CVPR.1992.223161