



Budapest University of Technology and Economics
Department of Automation and Applied Informatics

PROCEEDINGS OF THE
AUTOMATION AND APPLIED COMPUTER
SCIENCE WORKSHOP 2022 (AACS'22)

Editors:
István Vajk
Dmitriy Dunaev

István Vajk, Dmitriy Dunaev(Eds.)

Automation and Applied Computer Science Workshop 2022 (AACS'22)

Budapest, Hungary. June 24, 2022

Automation and Applied Computer Science Workshop 2022 (AACS'22)

Budapest University of Technology and Economics
Faculty of Electrical Engineering and Informatics
Department of Automation and Applied Informatics

1117 Budapest, Magyar tudósok körútja 2. Q.

url <http://www.aut.bme.hu/aacs>
e-mail: aacs@aut.bme.hu
tel: (+36-1) 463-28-70
fax: (+36-1) 463-28-71

The cover of the proceeding is designed by Tibor Kovács

Editors:
István Vajk
Dmitriy Dunaev

ISBN 978-963-421-875-3

Organization

Automation and Applied Computer Science Workshop 2022 (AACCS'22) is organized by the Department of Automation and Applied Informatics at the Budapest University of Technology and Economics on June 24, 2022

Program Committee

Attila Balogh
Ruth Bars
László Blázovics
Hassan Charaf
Kristóf Csorba
Ákos Dudás
Dmitriy Dunaev
Péter Ekler
Bertalan Forstner
András Futó
Zoltán Gájász
Tibor Gál
János Hamar
Jenő Hetthéssy
Attila Kárpáti
László Keviczky

Tamás Kökényesi
Tibor Kovács
Bence Kővári
László Lengyel
Gyula Max
Gergely Mezei
Ákos Nagy
Balázs Rakos
Márton Sipos
Péter Stumpf
Zoltán Sütő
Luca Szegletes
Gábor Tevesz
István Vajk
István Varjasi

Sponsored by

ARDINSYS Zrt.
AutSoft Kft.
evosoft Hungary Kft.
Microsoft Magyarország Kft.
Nokia Hungary Kft.
Quattrosoft Kft.
Sagemcom Magyarország Kft.

Table of Contents

1	Szabolcs Veréb Adaptive single phase shift control of dual active bridge converters	9
2	Yara Aboubakr and Balázs Rakos Photon-coupled, photoswitchable protein-based AND logic gate	20
3	Kinga Lauter and Luca Szegletes A comparative study of subgraph isomorphism algorithms	32
4	Cintia Lia Szűcs The effectiveness of derived local features in online signature verification	44
5	Gábor Knyihár and Kristóf Csorba Evaluation of non-destructive testing of pills	56
6	Laszlo Sari and Luca Szegletes Subgraph isomorphism in dynamic graphs	66
7	László Stranyóczy Midpoint voltage balancing and ripple reduction of three-level inverters	78
8	Kristóf Gábor Bándy Quadratic regression model based model predictive control of grid-side converters	89
9	Husam Kareem and Dmitriy Dunaev Impact of using different LUTs' inputs on XOR-based ring oscillator PUF behavior	103
10	László Kálmán Trautsch and Szilvia Erdős State space modelling of a final exam scheduling problem	115
11	Bálint Palásti Efficiency improvement and asymmetry compensation of inverters connected to a three-phase network by transient-free disconnection of one phase	127
12	Saif Alsarayreh Analysis of PMSM emulation using a three-phase inverter with LCL output filter	135
13	Gergő Bendegúz Békési, et al. A use case of artificial intelligence in measurement automation	147

14 Norbert Somogyi	
Formal verification of multi-level modeling requirements	157
15 Adorjan Kovacs	
Model predictive lateral stability envelope control for unmanned vehicles using minimal tuning parameters	169
16 Szabolcs Torma and Luca Szegletes	
EEGWave: a denoising diffusion probabilistic approach for EEG signal generation	181
17 Dániel Pásztor, et al.	
Static routing algorithm in wireless sensor networks	195
18 MD Moshir Rahman Chowdhury	
The real-time simulation and analysis of a three-phase active- front-end (AFE) rectifier	203
19 Mustafa Shubbar and Balázs Rakos	
Performance estimation of a self-adapting pixel antenna - MOM diode infrared energy harvesting device	213
20 Vince Abosa Szakal and Bertalan Forstner	
A novel method for creating datasets used in computer vision- based quality assurance	222
21 David Sik and Janos Levendovszky	
Predictive maintenance framework with smart data lake environment	234
22 Mohamed Azouz Mrad, et al.	
Comparing measurements in the prediction of in vitro dissolution profile using artificial neural networks	241
23 Bálint Béres, et al.	
Face anonymization for single person pose estimation: A comparative study of the effect of face anonymization	252

Editor's Introduction

As the Chair, I am honoured and pleased to welcome you to the Automation and Applied Computer Science (AACS) Workshop 2022 to take place on June 24, 2022 at the Department of Automation and Applied Informatics of BME. The series of these workshops started in 2006 and since then it has been organized to close each Spring semester. The workshop provides an opportunity for the young generation connected to the Department to present their latest scientific achievements, and to get feedback on their results. This conference will facilitate networking, information exchange and collaboration within and among the scientific groups and the participants of the young scientific generation.

This volume contains the papers of AACS'22. The workshop topics cover all major activities of the Department. These include control theory, control engineering, energetics, industry 4.0, robotics, power electronics, hardware design, data mining and intelligent systems, machine learning and artificial intelligence, graph and model transformation, computer graphics, metamodeling and domain-specific languages, mobile and distributed systems.

For the 17th AACS, the Program Committee has selected 23 papers to include in the final program after at least two anonymous and independent reviews per paper. I would like to thank all the reviewers for their time and effort. I also acknowledge the authors themselves, without their valuable input there would have been no conference. Finally, I would like to thank the proceedings team for the decorative outcome.

I hope the conference will be of professional and personal benefit to all of us in our work to improve the quality and presentation of your scientific achievements. I hope to meet you at AACS'22 Workshop.

István Vajk
Chair of AACS'22

A Use Case of Artificial Intelligence in Measurement Automation

Gergő Bendegúz Békési^{*1,3}, Péter Ekler¹, Ágnes Urbin², Szilárd Szőke³

Department of Automation and Applied Informatics¹

Faculty of Electrical Engineering and Informatics

Budapest University of Technology and Economics

and

Department of Mechatronics, Optics and Mechanical Engineering Informatics²

Faculty of Mechanical Engineering

Budapest University of Technology and Economics

and

ThID Team³

Robert Bosch Kft.

*Correspondence: bekesigergobendeguz@edu.bme.hu

Abstract. Due to global trends such as electrification, proliferation of smart devices, increasing electrical energy consumption, the expansion of electronics and automotive electronics, and the growing popularity of electric cars, special emphasis is being placed on the electronics industry, including the power transistor industry. T3Ster provides an opportunity for thermal examination and identification in field of automotive electronics. T3Ster measurements can be performed more economically and competitively with robotization, which requires a vision system. This paper presents the hardware selection and the artificial intelligence (AI) based software development of the vision system using fully convolutional networks (FCN) and machine learning (ML) algorithms. The developed system can help the robot to detect electronic components. Using automated runs, different neural networks were evaluated, modifying depth, number of channels, resolution, and batch size. The applied technology was the Python 3 programming language primarily using the OpenCV, Pytorch and Torchvision libraries.

Keywords: Machine Vision; Fully Convolutional Neural Network; Computer Vision; Machine Learning; Artificial Intelligence; Electronic Components; Optics

1 Introduction

Due to global trends of the 21st century special emphasis is being placed on the power transistor industry. These electronic devices must deal with more and more heat power, because of their increasing electrical energy consumption, although their surfaces are getting smaller and smaller by the time. This trend is a serious technical challenge, since too high temperature is the leading cause of failure. Thus, thermal examination and identification is extremely important,

which can be performed by the T3Ster equipment [1]. We with the ThID team at Robert Bosch Kft. participated in a project that aimed to automate and accelerate T3Ster measurements.

We developed a vision system as a complex electronic device for a robot, controlled in a LabVIEW environment. As a result, our system can help the robot to detect the MOSFETs and other electronic components. After the detection, the robot can automatically obtain the required measurements. We developed both the hardware and software for the system. We selected a camera with appropriate sensor and optics, and implemented a server platform capable of serving several computers that would control or monitor the measurement at the same time with the help of a Raspberry Pi 4 Model B. In Autodesk Inventor Professional 2022, we designed the tool that supports mechanical integration and then created a prototype with a 3D printer. In the most emphatic part of our work, we performed semantic segmentation and keypoint detection for robot navigation using artificial intelligence, fully convolutional neural networks, and computer vision algorithms. We implemented the fully convolutional neural networks and created the dataset necessary for the teaching. That dataset was made with the camera and optics that are used in real measurements and annotation happened manually. We have studied neural networks comprehensively. Using automated runs, we taught and evaluated networks, modifying depth, number of channels, resolution, and batch size. We did these in the open-source Python 3 programming language primarily using the OpenCV [2], Pytorch [3] and Torchvision [4] libraries.

The following sections of this paper focus on the image processing part of the described project which consists of two parts. The first part is the selection of the optics and sensor that is detailed in Section 2. The second part is the software development which is presented in Section 3. Finally, Section 4 summarizes the results and highlights the conclusions.

2 Selection of the camera system

In this section, we describe the selection of the optics and sensors for the system to achieve the 100 μm accuracy required to navigate the robot properly.

2.1 Calculation of the range of the focal length to select the lens

The accuracy requirement mentioned in Section 2 can be fulfilled in two steps. In the first step, the adequate range of the focal length has to be estimated. It is an estimation because in the first step the difference between the object distance and the object distance measured from the lens is unknown. Thus, these are taken to be the same and clarified in the second step.

Furthermore, the object is assumed to be located along the optical axis for each measurement. So, based on the laws of industrial optics, the required range of focal length can be calculated as follows.

$$f = \frac{1}{\frac{x}{a \cdot px \cdot |s|} - \frac{1}{s}} \quad (1)$$

The variables of Eq. (1) are described in Table 1.

Table 1: The variables of the focal length calculation

Variable	Description
a	The number of pixels of the smallest part to be observed.
px	The pixel size.
x	The size of the smallest part to be observed.
s	The object distance.
f	The focal length.

Knowing that

- a is recommended to be chosen from the range [5; 15],
- 2 μm is a good initial estimation of px ,
- x is 100 μm because of the requirements,
- and s is required to be in the range of [-100 mm; -20 mm] because of the setup,

a Waveshare Electronics 8-50mm Zoom Lens is a good choice for the vision system because it covers the range of the result of the calculation which is $f = 8\text{-}27\text{ mm}$.

2.2 Selection of sensor

After having the lens a sensor has to be selected. Based on the technical parameters of the lens, a can be defined as a function of px , and the object distance (s) can be distinguished from the object distance measured from the lens (s_0).

$$a = \frac{x}{px} \cdot \frac{1}{|s_0 - d + f|} \cdot \frac{1}{\frac{1}{f} + \frac{1}{|s_0 - d + f|}} \quad (2)$$

The variables of Eq. (2) are described in Table 2.

Table 2: The variables of the focal length calculation

Variable	Description
s_0	The object distance measured from the lens.
d	The total length of the lens.

Therefore, it is advisable to choose px , so that a remains within the recommended range for every possible f . The Sony IMX477 sensor meets this requirement with a pixel size of 2.2 μm . The value of a is important to be in the recommended range for two reasons. If a is too large, some parts of the MOSFETs might not be detectable. And if a is too small, the smallest part is not well detectable.

3 The software development of the vision system

After performing the hardware tasks of image processing, the software implementation can take place. It basically consists of configuration, semantic segmentation, and keypoint-based displacement calculation.

3.1 Configuration

Unequal light bending can cause radial distortion. When rays bend more near the edges of a lens than they do at the optical center, a non-linear variation in resolution can occur. Radial distortion must be eliminated during configuration. In doing so, calibration can be performed based on the conventional distortion approach according to [5]. The equation of distortion is as follows.

$$\begin{bmatrix} x_d \\ y_d \end{bmatrix} - \begin{bmatrix} x_u \\ y_u \end{bmatrix} = \begin{bmatrix} x_u \\ y_u \end{bmatrix} \cdot \sum_i k_i \cdot r^{2i} \quad (3)$$

The variables of Eq. (3) are described in Table 3.

Table 3: The variables of the equation of radial distortion

Variable	Description
x_d	The x coordinate of the distorted pixel position.
y_d	The y coordinate of the distorted pixel position.
x_u	The x coordinate of the undistorted pixel position.
y_u	The y coordinate of the undistorted pixel position.
r	The radial distance.
k_i	The i^{th} constant coefficient describing the bias.

The left side of the equation is the distortion in pixels. This is a nonlinear optimization that can be solved by the Levenberg-Marquardt algorithm. The camera matrix is calculated by singular value decomposition.

Another calibration is the calibration of the position and orientation of the optical system. In case of any change in these, the same object appears in the picture differently. In the picture, each shape appears as a result of perspective projection, that is not a distance-preserving transformation and does not preserve the ratios of lengths. It is not angle-preserving and parallelism-preserving

transformation but it preserves collinearity. Actually, this is the descriptor of the Pinhole camera model, as every projection line passes through the same point. The projection can be defined as follows.

$$\begin{bmatrix} s \cdot x' \\ s \cdot y' \\ s \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & b_1 \\ a_{21} & a_{22} & b_2 \\ c_1 & c_2 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (4)$$

The variables of Eq. (4) are described in Table 4.

Table 4: The variables of the perspective projection

Variable	Description
s	The scale factor.
x'	The horizontal coordinate in the projected image.
y'	The vertical coordinate in the projected image.
a_{ij}	Variables that describe the orientation matrix of the transformation.
b_i	Variables that describe the translation vector of the transformation.
c_j	The variables of the projection vector.
x	The original horizontal coordinate.
y	The original vertical coordinate.

Since the matrix of the projection is defined by 8 variables, it can be calculated from 4 original-projected point pairs. It can be performed with the built-in functions of OpenCV, applying a perspective transformation based on the known points in the calibration image.

3.2 Semantic segmentation

After having an undistorted image, semantic segmentation has to be performed to specify the region of interest (RoI), where the keypoints have to be found for navigation. As a result of a detailed literature search, we found FCNs to be the most suitable for this task because conventional ML algorithms, like k-means clustering did not give an accurate solution [6]. The effectiveness of FCNs is confirmed by both [7], [8] and [9].

The photo-dataset for learning was created with the selected lens and sensor, and data augmentation was done with an own class to increase the amount of data by adding slightly modified copies. This actually meant random vertical and horizontal mirrorings and 90° and -90° rotations. The data-loading was done with the modification of the *DataLoader* class of Pytorch from the *torch.utils.data* module.

In addition to the operation of convolution, every convolutional layer in the network also includes nonlinearity and normalization. These can be derived as a class from the *pytorch.nn* module with an extra *convolution* attribute that is a *pytorch.nn.Conv2d* object. It is important to enter *stride*, *padding* and

channel numbers as input here. The applied forward function is the *ReLU*. The structure of the transposed convolution layer is very similar. The main difference is that the extra attribute is a *pytorch.nn.Transpose2d* object.

We create 5 types of network architectures from the layers discussed above and further vary their parameters. The architecture class is also derived from the *pytorch.nn module*. As a first step, we create a convolution layer as an attribute that receives the 3 color channels as input and returns the input channels of the architecture. Then n downscaling and convolution follow each other, which are also realized by convolutional layers. The parameters of the i -th downscaling are 2^{i-1} times the number of input channels in the architecture, the size of the kernel, and the stride. Furthermore, in the case of convolution, the input parameter is 2^i times the input channel number of the architecture. Next, we use the transposed convolutions according to a similar logic and a classifier is placed at the very end of the network. Downscaling and upscaling is done with feedforward.

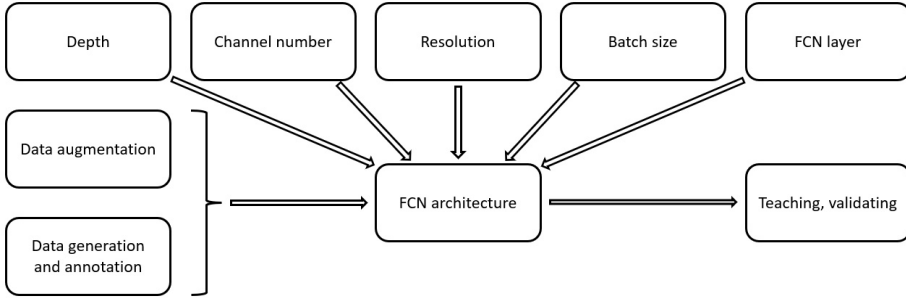


Figure 1: Semantic segmentation framework

In teaching, we used the internationally accepted cross-entropy loss function and the Adam optimization algorithm. During the runs, we changed the depth, channel numbers, resolution, and batch size as shown in Figure 1. Depth ranged from 2 to 6. Possible values for the channel numbers were 8, 16, 32, and 64. The height-width resolutions used are 64-128, 128-192, 192-320, 256-384, 320-512, 384-640, 448-704, 512-832, 576-896, 640-1024, 704-1152, 768-1216 and 832-1344. And the possible batch sizes are 128, 64, 32, 16, 8 and 4. In each case, 50 teachings were run and after every 10th teaching, the step-length was decreased to a thousandth.

3.3 Evaluation of semantic segmentation

The essence of semantic segmentation is to determine the class of each pixel in an image. Thus, the resolution of its output is the same as the resolution of the input. There are as many types of output for each pixel as classes. For this work, it must be decided for each pixel whether it is an electronic component or not. Thus, two values can be obtained, 1 or 0. The widely used and accepted

quality measure of this decision is the IoU (Intersection over Union) value, which quantifies the overlap ratio between the segmentation output and the actual segmentation. This is described as follows.

$$IoU = \frac{\text{output of segmentation} \cap \text{solution of segmentation}}{\text{output of segmentation} \cup \text{solution of segmentation}} \quad (5)$$

Figure 2 shows the run results per epoch of the network with the best validation IoU .

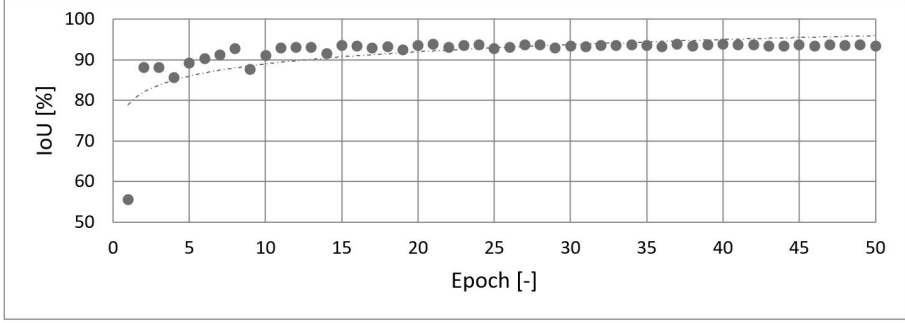


Figure 2: The results of the best architecture

Thus, the network with 8 input channels, 5 depths and 576-896 resolution performed the best with batch size 4. The best validation result was 93.78% and the best test result was 94.01%. Thus, it can be seen that not the most complex architecture calls forth the best solution. This is also highlighted in the following four figures. These summarize the average results of all runs in terms of image resolution, batch size, depth, and input channel number. At image resolution, the vertical numbers of the pixels are displayed on the right.

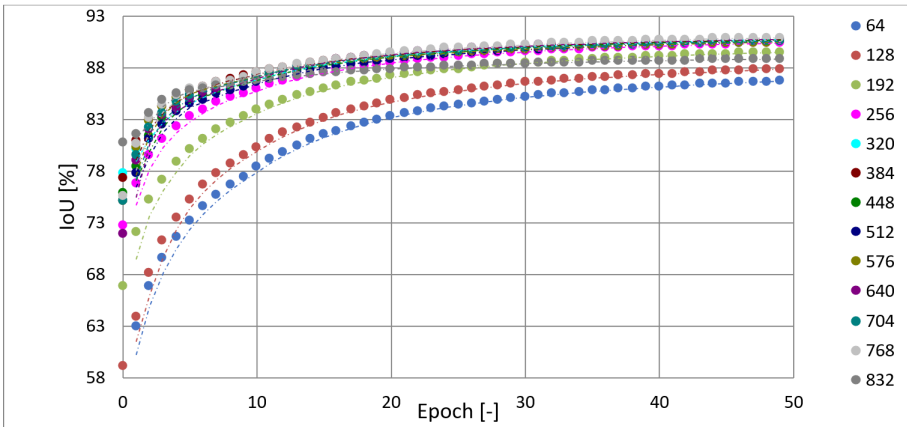


Figure 3: Results of architectures using images with different resolutions

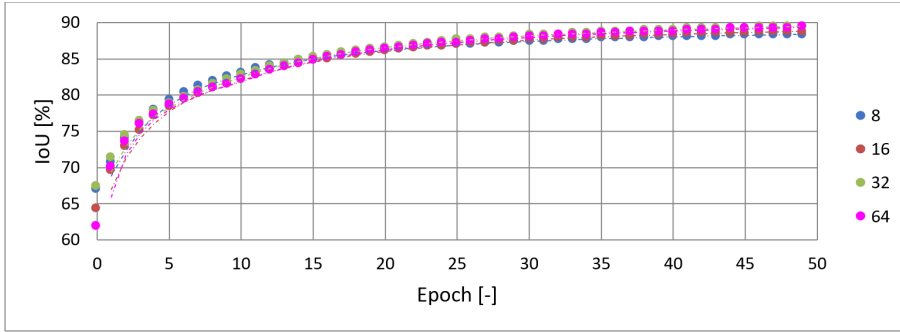


Figure 4: Results of architectures with different input channel numbers

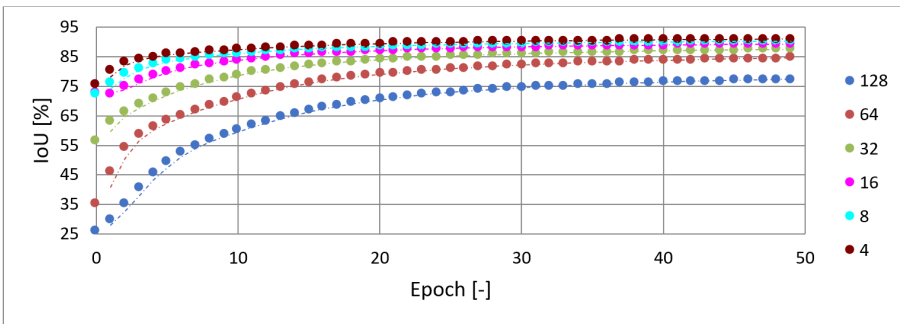


Figure 5: Results of architectures using different batch sizes

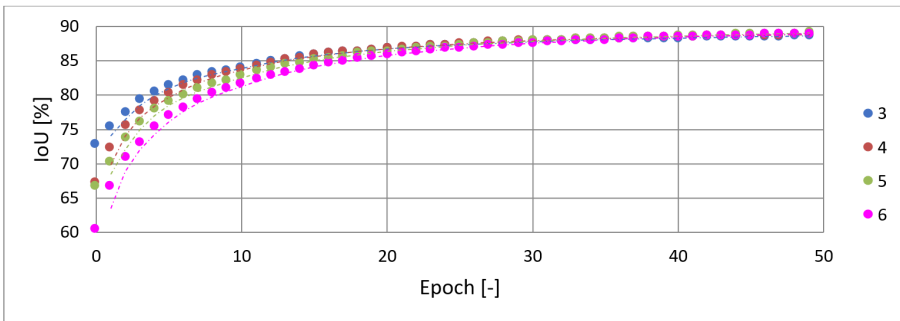


Figure 6: Results of architectures of different depths

In case of the averages, a characteristically monotonic IoU increase can be observed as a function of epochs. It is noticeable that the images with higher resolution provide better results. In addition, although it did not provide the best results, the 32 input channel numbers generally give slightly better results

than the others. The suitability of batch size 4 is evidenced by Figure 5, and Figure 6 proves that 5 is the best architecture depth.

3.4 Keypoint detection for displacement calculation

Knowing the RoI, keypoints can be searched and the displacement can be determined from their coordinates. We found these keypoints with the help of the Scale Invariant Feature Transform (SIFT) detector, which is also available in OpenCV and whose patent expired in March 2020, so it can be used in the industry free of charge. This algorithm basically searches for corner points and creates an invariant descriptor for all transformations except perspective distortion based on their local environment. The perspective distortion is eliminated at the calibration so it is not a problem.

The displacement calculation was tested in 50 cases with a calliper and every time the error of the calculation was in the range of [50 μm ; 100 μm]. The error of the rotation calculation was always under 1° in each case. So, the most important requirements were fulfilled.

4 Conclusion

The semantic segmentation was rated with the internationally accepted *IoU* metric applied in OpenCV. With a train-validate-test ratio of 233-111-111, we achieved a validation result of 93.78% and a test result of 94.01%. The best was the 5-depth, 8-channel, 4-batch neural network with the resolution of 576-896. It is also important to note that the requirements for the displacement calculation with 100 μm accuracy have also been met.

All in all, we created an electronic system with a robust implementation of navigation to the components, that are measured. We achieved 100 μm and 1° accuracy, which was associated with an IoU value of 94% accuracy for semantic segmentation. We tested our system in the real, original measurement environment. So, our work is absolutely applicable for additional scientific and industrial projects. Furthermore, using the FCNs and SIFT algorithm together, that helped to reach presented results, is a new approach. In further research, the described method is planned to be extended to the measurement of direct bonded copper metallized ceramics.

Acknowledgments

We wish to thank to Robert Bosch Kft. for supporting this project under the Bosch Engineering Students Team Scholarship.

The part of this work that is related to Department of Automation and Applied Informatics has been also carried out in the frame of project no. 2019-1.1.1-PIACI-KFI-2019-00263, which has been implemented with the support provided from the National Research Development and Innovation Fund of Hungary, financed under the 2019-1.1. funding scheme.

References

- [1] “Simcenter T3STER: Characterize the thermal properties of components.” <https://www.plm.automation.siemens.com/global/en/products/simcenter/t3ster.html>. [Last visited: 25. 04. 2022.].
- [2] “OpenCV: Open source computer vision library.” <https://opencv.org>. [Last visited: 25. 04. 2022.].
- [3] “Pytorch: Documentation.” <https://pytorch.org/docs/stable/index.html>. [Last visited: 25. 04. 2022.].
- [4] “Torchvision: Documentation.” <https://pytorch.org/vision/stable/index.html>. [Last visited: 25. 04. 2022.].
- [5] P. Drap and J. Lefèvre, “An exact formula for calculating inverse radial lens distortions,” *Sensors*, vol. 16, 2016.
- [6] G. B. Békési, “Measurement automation of MOSFETs and other electronic components using artificial intelligence, neural networks and computer vision,” *Scientific Students’ Association Report, BME*, 2021.
- [7] Y. Guo, Y. Liu, and T. Georgiou, “A review of semantic segmentation using deep neural networks,” *International Journal of Multimedia Information Retrieval*, vol. 7, pp. 87–93, 2018.
- [8] F. Lateef and Y. Ruichek, “Survey on semantic segmentation using deep learning techniques,” *Neurocomputing*, vol. 338, pp. 321–348, 2019.
- [9] B. Li, Y. Shi, Z. Qi, and Z. Chen, “A survey on semantic segmentation,” in *2018 IEEE International Conference on Data Mining Workshops (ICDMW)*, pp. 1233–1240, 2018.