Grasping Point Detection Using Monocular Camera Image Processing and Knowledge of center of Gravity

Dominik Stursa[1], Petr Dolezel[2], and Daniel Honc[3]

[1] University of Pardubice, Faculty of Electrical Engineering and Informatics, 532 10 Pardubice, Czech Republic, https://fei.upce.cz/en
dominik.stursa@upce.cz
[2] petr.dolezel@upce.cz
[3] daniel.honc@upce.cz

Abstract. The ability to grasp objects is one of the basic functions of modern industrial robots. In this article, the focus is placed on a system for processing the image provided by a robot visual perception system leading to the detection of objects grasping points. The proposed processing system is based on a multi-step method using convolutional neural networks (CNN). The first step is to use the first CNN to transform the input image into a schematic image with labeled objects centers of gravity, which then serves as a supporting input to the second CNN. In this second CNN, original input and supporting input images are used to obtain a schematic image containing the grasping points of the objects. This solution is further compared with a network providing grasping points directly from the input image. As a result, the proposed method provided a 0.7% improvement in the average intersection over union for all of the models.

Keywords: Image Processing, Grasping Point Detection, Convolutional Neural Network, Center of Gravity

1 Introduction

In recent decades, many manufacturers have resorted to the use of automated robotic lines, which bring a number of advantages. The first and probably most obvious reason is the reduction in production costs. Although the initial cost may seem daunting, manufacturing companies will recoup their investment in the long run due to several key benefits such as the ability to work 24/7 without interruption or the high accuracy and repeat-ability of operations [1].

In addition, one of the biggest advantages is their predictable behaviour and precision of movements. This leads to a greater ability to consistently produce high quality products with little variation, all with minimal need for human control. The safety of workers, speeding up production and increasing profits are significantly influencing the number of robotic manipulators in industry [2].

Not only this pressure from the industry, but also advances in digital image processing are opening the door for all sorts of robotic applications. Machine vision is typically used in robotic industry to detect objects [3] or their grasping points [4], to locate obstacles [5] or to find product defects [6], and for quality control.

Based on this information about the surrounding world, the robotic manipulator can execute a controlled movement that, for example, ensures the removal of objects from the conveyor belt to defined locations in the new position (so-called pick and place), which is one of the typical robotic applications [7].

Over the past decades, approaches to the pick and place applications have been introduced. Thanks to the possibilities of individual parts of the pick and place system, it is possible to categorize them. For example, it is possible to categorize these approaches according to the robotic system used, the camera system used, the object detection method developed using an analytical procedure or a data-driven procedure.

Analytical approaches consider the geometric shape of the target object and try to find exactly possible positions to grasp the object. Thanks to deep learning methods, similar or even better results can be achieved by applying empirical approaches without the need for analytical processing [8]. Detailed descriptions of the benefits and differences of these approaches can be found in surveys [9,10].

Machine vision methods can be categorized according to many criteria that narrow down the problem. An example is the dimension of the solution needed, which delineates whether the problem needs to be solved in 2D or 3D space [11]. In our case, the application limitation is defined by a robotic manipulator with 3 degrees of freedom (DoF) and a suction cup end-effector. As the robotic manipulator is not able to grasp object randomly rotated in 3D space, but only object vertically accessible, this leads to limitation of computer vision solution in 2D space.

In this contribution, we consider the localization of the object's center of gravity as an accompanying information to the system to find a suitable grasping point, which is the initial part of the pick and place application.

1.1 Related work

Finding object grasping points has recently become a widespread issue that helps find faster or more accurate solutions to make pick and place more efficient. Convolutional neural networks are increasingly being used to find grasping points [12]. One method that provides an improvement in the accuracy of grasping point estimation is to use neural networks to generate bounding boxes of important grasping points using point-wise convolution [13].

Other methods used in this area are deep learning methods that finds salient point coordinates by shape context information [14]. The use of object shape context information is also used in a more modern processing approach using image segmentation, where the grasping points are highlighted and used for further not so demanding processing [15].

It is also more common to combine different convolutional neural networks to add different information about object positions and grasping points locations with each other [16] or to obtain differently defined outputs that can play a role in the decision-making process to select the correct grasp point [17].

2 Problem Formulation

This section serves to correctly define the aim of the article. As mentioned, we are concerned with the first necessary step in the object pick and place task, which is machine vision. Specifically, determining the center of gravity of an irregular object with a homogeneous mass distribution using image data processing. The need to obtain the center of gravity of the object is related to the effort to find the most suitable grasping point, which will load the robotic manipulator in uncontrolled axes only minimally during the pick and place application.

Objects of interest are differently sized normalized triangular construction supports illustrated together with expected robotic manipulation implementation in Fig. 1.

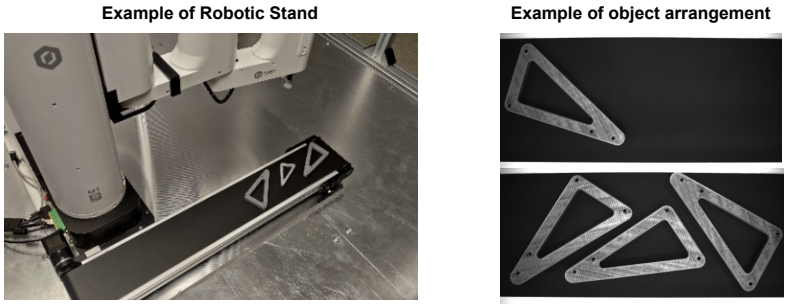

Fig. 1. Demonstration of the problem and arrangement of monitored objects on a conveyor belt.

According to the complexity of the shape and size of the objects along with the monotony of the surface on which the objects are placed, a monochromatic monocular camera was chosen to capture the image for processing.

2.1 Proposed Solution

Since we want to use image processing to find the center of gravity of the object, which we then use as auxiliary information for the evaluation of the grasping point, the following two procedures were chosen to determine the center of gravity. The first procedure is to construct a parameterized analytical solution. The

second procedure is to obtain the center of gravity of the object using basic image processing methods. Both of these procedures will be used to label the data for convolutional neural network training.

Once the object center of gravity has been marked, this information is fed together with the input image into a second step in which a search for suitable grasping points is performed. Both steps are implemented by converting the input image into a schematic image using a convolutional neural network. In the output schematic image, the coordinates of possible grasping points are then calculated from the individual segmented entities. The whole process is illustrated in Fig. 2

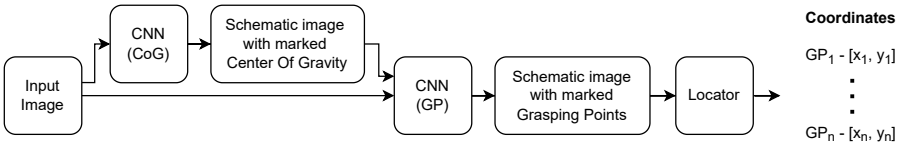

Fig. 2. Diagram of proposed processing approach.

Analytical Solution Using the analytical solution, the center of gravity was calculated exactly according to the geometry of the triangular construction support, which is described in Fig. 3

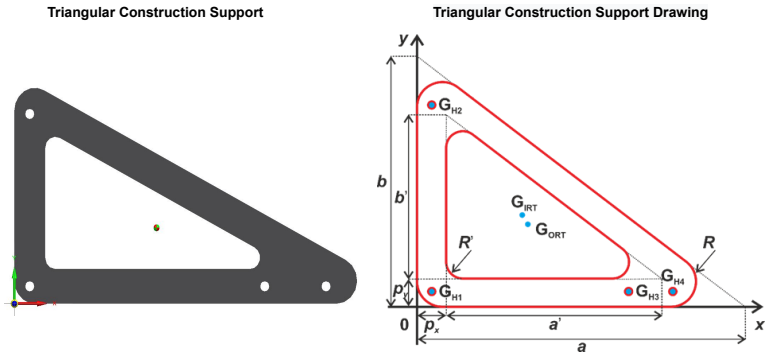

Fig. 3. Drawing of the triangular structure with all significant points, parameters and centers of gravity marked.

Used triangular beams contain rounded edges and cut-outs that cause the center of gravity to shift relative to a normal solid triangle. As a result, the

analytical calculation of the center of gravity must be treated as a composition of the centers of gravity of the individual parts, leading to the following equations for the center of gravity coordinates of the object.

$$\mathrm{x}_G = \frac{x_{OTR} \cdot A_{OTR} - x_{ITR} \cdot A_{ITR} - \sum_{n=1}^{m} x_{H_n} \cdot A_{H_n}}{A_{OTR} - A_{ITR} - \sum_{n=1}^{m} A_{H_n}}, \tag{1}$$

$$\mathrm{y}_G = \frac{y_{OTR} \cdot A_{OTR} - y_{ITR} \cdot A_{ITR} - \sum_{n=1}^{m} y_{H_n} \cdot A_{H_n}}{A_{OTR} - A_{ITR} - \sum_{n=1}^{m} A_{H_n}}, \tag{2}$$

where x_{OTR} and y_{OTR} are outer round shaped triangle coordinates of center of gravity; x_{ITR} and y_{ITR} are inner round shaped triangle coordinates of center of gravity; x_{H_n} and y_{H_n} are center coordinates of holes; and all A correspond to the contents of the area of the individual elements named identically as in the case of coordinates. Parameter m corresponds to the number of mounting holes in the support. Depending on these calculations, a center of gravity label is then created in the schematic image according to the size of the parameterized triangle.

Image Processing Solution The application of basic image processing methods to calculate the center of gravity can be used when there is only the one object in the image, and only on objects that are fully filled [18].

Since the objects themselves contain cutouts, the problem in this case is also split to find multiple individual important centers of gravity, which will later be folded to obtain the main center of gravity. For the purpose of this method, the holes for the screws are neglected and only a large cut-out in the middle of the object is considered. The result is the determination of the center of gravity of the cut-out and the center of gravity of the filled object separately and the subsequent subtraction of the both individual coordinates in both axes.

The separate processing consists in converting the grayscale image into a binary image, in which the coordinates of the center of gravity are calculated. To obtain a binary image, the input image is processed using thresholding. The next step is to perform a pattern fill, which is used both to calculate center of gravity for filled object, but also after subtracting it from the original binary image to create the cut-out object and then calculating its center of gravity. The following is the composition of the centers of gravity of the components of the object and the marking of the center of gravity itself in the schematic figure. The size of the center of gravity marking is based on the size of the object area. The individual steps of image processing procedure for labeling the center of gravity of an object is shown in Fig. 4.

3 Experiments Procedure

In this section, the design of the system for estimating the center of gravity and for estimating the grasping points of triangular structural supports will be

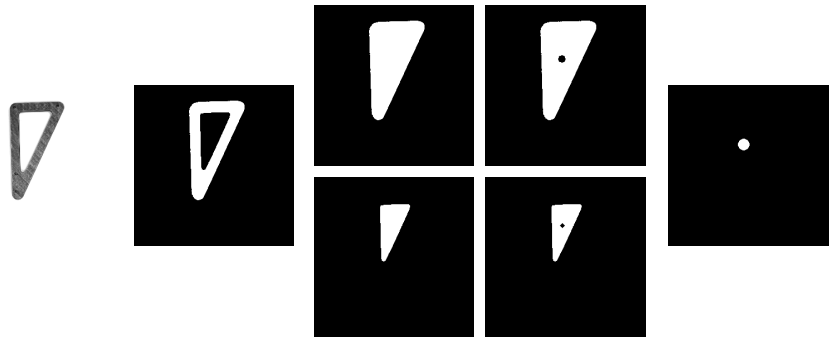

Fig. 4. Image processing procedure for marking the center of gravity of an object.

described. Both systems are based on convolutional neural networks performing image segmentation and producing a schematic image.

Two approaches have been chosen for comparison, the first is the aforementioned approach illustrated in Fig. 2, the second is the conventional approach that performs the processing of the whole image in one step providing directly the resulting grasping points. Moreover, due to the two methods for labeling the data, these approaches are also compared and evaluated with each other.

Both variants used the encoder-decoder principle, specifically a scalable version of the U-Net architecture that can be scaled according to provide the necessary segmentation. A scheme of this proposed scalable architecture based on the U-Net topology is shown in Fig. 5.

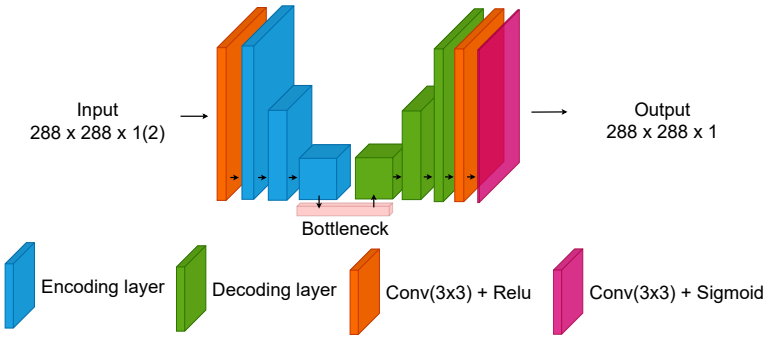

Fig. 5. The encoding layer halves the width and height of its layer while doubling its depth. The opposite function is performed by the decoding layer, which doubles the width and height of its inputs and reduces the depth to half of the previous decoding layer or bottleneck. Everything then depends on the parameters of the first convolution and the depth of the network (how many encoding or decoding modules are used).

3.1 Dataset Creation

In order to acquire training and testing data, the special stand was prepared. A Basler acA2500-60um industrial monochrome camera [19] was used as a proper sensor. This sensor was set to provide 2048 × 2048 px images. The camera was equipped with a Computar M3514-MP lens [20] in order to monitor the scan area of 300×300 mm from a distance of 500 mm, which was calculated using the basler configuration tool [21].

Data Acquisition For the purpose of verifying the ability to adapt to the parameters (sizes) of triangular construction supports and due to the possibilities of the labeling techniques, the dataset was divided into 2 types of images. Specifically, the images where triangular construction support in different positions and rotation is present once in time, for 3 different supports sizes, where 200 images were taken for each size. The second group, used only for testing purposes, are images with multiple objects on a single image, of which 300 were created for these purposes. Thus, a total of 900 images were taken for training and testing neural network topologies. Examples of these images are shown in the following Fig. 6.

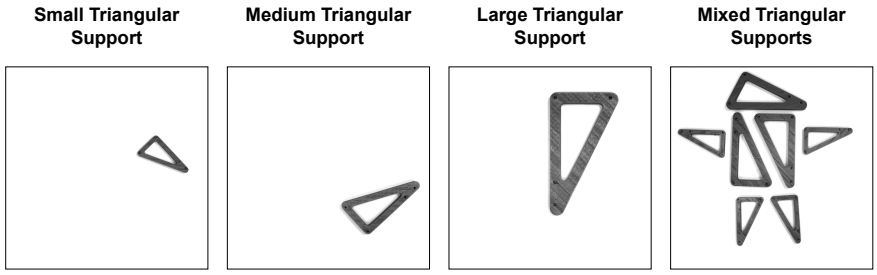

Fig. 6. Examples of different sizes of triangular construction support and of their mix.

Data Labeling Each image from the training and test set was labeled for neural network use. Specifically, the positions of the center of gravity and the positions of the grasping points, for which corresponding schematic images were created according to the two techniques mentioned in section 2.1. A custom labeling application was created, which for the first method was able to manually determine the important points (selected mounting holes of the triangular support), thanks to which the image is automatically labeled also with respect to the parameterization of the size of the marking determined by the distance of the selected points. The position of the center of gravity and grasping points is then calculated using calculations based on knowledge of the design drawing.

The second part was the automatic marking purely using the described digital image processing to gain the center of gravity. An example of the given marking is shown in Fig. 7.

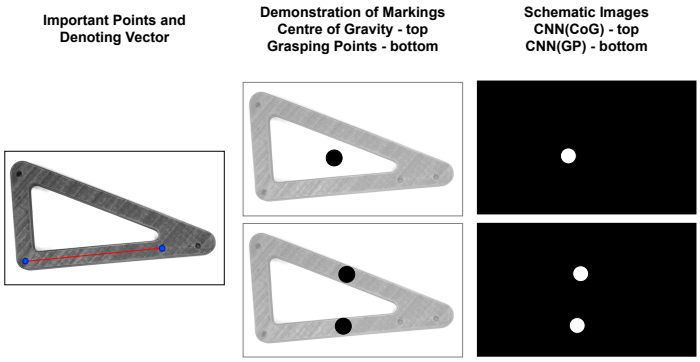

Fig. 7. Example marking of the center of gravity and grasping points of an object.

Data Augmentation Furthermore, the data was augmented to expand the dataset and thus provide better training of the networks. The original dataset was augmented as follows. Thanks to the possibility of applying basic image processing, the background for the original images was removed and replaced with three additional shades in the black and white spectrum, resulting in a light grey, dark grey and black background. This step increased the dataset size to 2400 frames for training and 1200 frames for testing.

3.2 Neural Network Training

Overall, given the problem, the experiments with topologies can be divided into four individual parts. The first and second parts are training the neural network to gain the center of gravity, where for the first method the computed centers of gravity from manual labeling are used and for the second method the labels from image processing are used. The third part is a CNN training to determine grasping points, where the input information is not only the input image but also a supporting schematic image capturing the center of gravity of the objects. Fourth is to train a neural network to directly determine the grasp points from the input image only.

To solve the defined problem, separate training was performed within each problem. For each problem, 4 configurations of depth parameters and initial filter count of scalable topology were chosen. The ADAM optimizer was used as the optimization algorithm, and weights were set for each topology randomly with

Gaussian distribution. Due to the stochastic nature of neural networks, each topology was trained five times to select the best performing model.

4 Results

For all the individual neural network solutions and topologies, the Intersection over Union (IoU) metric, which is commonly used to evaluate the performance of detection techniques, was selected.

A comparison of the methods for determining the center of gravity was carried out first. In this case, certain configurations of the scalable network were chosen for both labeling methods. Specifically, this was a combination of depths 2 and 4 along with the number of filters corresponding to values of 64 or 32. This resulted in 4 combinations of both that provided significantly different values for the testing set. The values of these results are shown in the following Tab. 1.

Table 1. Results of intersection over union for centers of gravity.

Topology	**Model 1**	**Model 2**	**Model 3**	**Model 4**	**Model 5**	**Mean value**
Computed - 2, 32	0.5941	0.5568	0.5685	0.5956	0.5355	0.5701
Computed - 2, 64	0.5935	0.6206	0.6267	0.6708	0.6014	0.6226
Computed - 4, 32	0.7460	0.8141	0.8391	0.7694	0.8347	0.8007
Computed - 4, 64	0.7632	0.7995	0.8361	**0.8554**	0.8326	0.8173
Processed - 2, 32	0.6041	0.5018	0.5510	0.5615	0.5532	0.5543
Processed - 2, 64	0.6015	0.5996	0.6671	0.6109	0.6244	0.6207
Processed - 4, 32	0.7545	0.8341	0.8415	0.7583	0.8422	0.8061
Processed - 4, 64	0.7752	0.7856	**0.8641**	0.8041	0.8221	0.8102

As second, the topology was tested to determine the grasping points from both types of inputs. These topologies used the best performing model of CoG determination providing input information. Results of IoU for tested topologies are summarized in Tab. 2

Table 2. Results of grasping points estimation with supporting information.

Topology	**Model 1**	**Model 2**	**Model 3**	**Model 4**	**Model 5**	**Mean val.**
Comp. - 4, 64; 4,64	0.7613	0.7351	0.7815	0.7351	**0.8213**	0.7669
Comp. - 4, 64; 4,32	0.7515	0.7995	0.7319	0.7442	0.7726	0.7599
Proc. - 4, 64; 4,64	0.7503	0.7982	0.8111	0.8211	**0.8301**	0.8022
Proc. - 4, 64; 4,32	0.7611	0.7112	0.8078	0.8042	0.7896	0.7748

Lastly, the data were evaluated using the direct method to determine the grasping points, for which the 4 combinations of sizes and depths are shown in the Tab. 3

Table 3. Results of grasping points estimation.

Topology	Model 1	Model 2	Model 3	Model 4	Model 5	Mean val.
T1 - 6, 32	0.6621	0.6011	0.6056	0.6202	0.6434	0.6265
T1 - 6, 64	0.6888	0.6055	0.5995	0.6332	0.6611	0.6376
T1 - 8, 32	0.7511	0.7684	0.7881	0.7687	0.7222	0.7597
T1 - 8, 64	0.8044	0.7800	0.7944	0.7992	**0.8155**	0.7987

5 Conclusion

The task of the first investigation was to determine whether generating data labels based on empirical computations provides higher accuracy than automatic data labeling. However, it can be determined from the data that the effect of the center of gravity method alone does not have a strong influence in the average values.

One of the other objectives of this work was to test whether preprocessing some key information about the object (the center of gravity in this case) can lead to more accurate estimation of the grasping points. For these cases, it is necessary to compare Tab. 2 and Tab. 3, which show the accuracy of the determination of the grasping points. These tables show that, on average, more accurate results are obtained with connected networks (with supporting information) containing a comparable number of parameters than with a separate network. Specifically, for parametrically comparable topologies, there is an average improvement of 0.7%.

In the future, the influence of preprocessing itself could be subjected to a more rigorous analysis that would precisely normalize the influence of the number of parameters against the resulting accuracy.

Acknowledgments. We would like to thank Dr. Lubos Rejfek for creating the images and their subsequent labeling for the dataset. We would also like to thank Associate Professor Jiri Tucek for providing the mathematical relations for accurate calculation of the center of gravity of triangular construction supports.

Fundings. 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. G. Profozich. California manufacturing technology consulting: Ready or not, robotics in manufacturing is on the rise. https://www.cmtc.com/blog/overview-of-robotics-in-manufacturing, 2021. 2022-2-7.

2. International Federation of Robotics. Robot race: The world's top 10 automated countries. https://ifr.org/ifr-press-releases/news/robot-race-the-worlds-top-10-automated-countries, 2021. 2022-02-08.
3. M. Schwarz, A. Milan, A.S. Periyasamy, and S. Behnke. Rgb-d object detection and semantic segmentation for autonomous manipulation in clutter. *International Journal of Robotics Research*, 37(4-5):437–451, 2018.
4. A. ten Pas, M. Gualtieri, K. Saenko, and R. Platt. Grasp pose detection in point clouds. *International Journal of Robotics Research*, 36(13-14):1455–1473, 2017.
5. M. Mancini, G. Costante, P. Valigi, and T.A. Ciarfuglia. J-mod 2 : Joint monocular obstacle detection and depth estimation. *IEEE Robotics and Automation Letters*, 3(3):1490–1497, 2018.
6. X. Gong, Y. Bai, Y. Liu, and H. Mu. Application of deep learning in defect detection. volume 1684, 2020.
7. G. Du, K. Wang, S. Lian, and K. Zhao. Vision-based robotic grasping from object localization, object pose estimation to grasp estimation for parallel grippers: a review. *Artificial Intelligence Review*, 54(3):1677–1734, 2021.
8. An overview of 3d object grasp synthesis algorithms. *Robotics and Autonomous Systems*, 60(3):326–336, 2012. Autonomous Grasping.
9. S. J. Dharbaneshwer, Sankara J. Subramanian, and Kai Kohlhoff. Robotic grasp analysis using deformable solid mechanics. *MECCANICA*, 54(11-12):1767–1784, SEP 2019.
10. K. Kleeberger, R. Bormann, W. Kraus, and M. Huber. A Survey on Learning-Based Robotic Grasping. *Current Robotics Reports*, 1(4):239–249, DEC 2020.
11. A. Bjornsson, M. Jonsson, and K. Johansen. Automated material handling in composite manufacturing using pick-and-place systems – a review. *Robotics and Computer-Integrated Manufacturing*, 51:222–229, 2018.
12. W. Miao, G. Li, G. Jiang, Y. Fang, Z. Ju, and H. Liu. Optimal grasp planning of multi-fingered robotic hands: A review. *Applied and Computational Mathematics*, 14(3):228–247, 2015.
13. Y. Teng and P. Gao. Generative robotic grasping using depthwise separable convolution. *Computers and Electrical Engineering*, 94, 2021.
14. J. Bohg and D. Kragic. Learning grasping points with shape context. *Robotics and Autonomous Systems*, 58(4):362–377, 2010.
15. T. Luddecke, T. Kulvicius, and F. Worgotter. Context-based affordance segmentation from 2d images for robot actions. *Robotics and Autonomous Systems*, 119:92–107, 2019.
16. D.-Y. Kim, K.-H. Sim, and G.-H. Lee. Object detection by combining two different cnn algorithms and robotic grasping control. *Journal of Institute of Control, Robotics and Systems*, 25(9):811–817, 2019.
17. E. Corona, G. Alenyà, A. Gabas, and C. Torras. Active garment recognition and target grasping point detection using deep learning. *Pattern Recognition*, 74:629–641, 2018.
18. H.C. Van Assen, M. Egmont-Petersen, and J.H.C. Reiber. Accurate object localization in gray level images using the center of gravity measure: Accuracy versus precision. *IEEE Transactions on Image Processing*, 2002.
19. Basler. Basler ace. https://www.baslerweb.com/en/products/cameras/area-scan-cameras/ace/aca2500-60um/, 2022. 2022-02-18.
20. Computar. Computar lenses. https://computar.com/product/705/M3514-MP, 2022. 2022-02-18.
21. *Lens Selector by Basler*, 2022 (accessed January 15, 2022). https://www.baslerweb.com/en/products/tools/lens-selector/.