

Towards Universal Detection and Localization of Mating Parts in Robotics

Stefan Marx¹, Attique Bashir²^a and Rainer Müller¹^b

¹Chair of Assembly Systems, Saarland University, Saarbrücken, Germany

²Department of Assembly Systems, center for Mechatronics and Automation Technology gGmbH, Saarbrücken, Germany

Keywords: Robotics, Assembly, Joining, Object Detection, Geometric Shapes.

Abstract: A key task in robotics is the precise joining of two components. This approach focuses on detecting basic geometric shapes such as rectangles, triangles, and circles, etc. on the respective mating counterparts. This paper first examines how precisely individual geometric shapes can be localized using stereoscopy with a single camera on the robot arm. After the localization of the individual shapes, the spatial relationships between these shapes are analyzed and then compared with those of a possible joining partner. If several features match, transformation parameters are calculated to define the optimal alignment for an accurate and efficient assembly. This method emphasizes simplicity and effectiveness in identifying complementary geometries for precise positioning during assembly tasks.

1 INTRODUCTION

Across a wide range of industries, from automotive to consumer electronics and medical technology, joining operations such as plugging, clipping, or pressing are essential steps in product assembly. Although these processes are often repetitive and geometrically well-defined, many of them are still carried out manually. This is particularly evident in the insertion of electrical connectors, cable harness assembly, and the mechanical joining of plastic or metal components. The underlying reason lies in the complexity of flexible automation: small variances in part geometry, orientation, or tolerance often require human adaptability, something traditional robotic systems have struggled to replicate.

At the same time, several key trends are accelerating the push toward robotic solutions. The ongoing shortage of skilled workers, driven by demographic change and the rising cost of labor, is putting pressure on manufacturers to automate even low- to mid-complexity tasks. The growing adoption of electric vehicles and the digitalization of machinery lead to an increasing number of connectors and interface points, many of which require precise positioning and certain process forces during assembly. Furthermore, the trend toward mass customization and high product variability requires flexible, adaptable systems capa-

ble of handling different joining geometries without extensive retooling.

Modern robotics, enhanced by vision systems and sensor integration, is increasingly capable of meeting these demands. Recent advances in camera-based localization, force-feedback control, and AI-driven feature detection allow robots to adapt to slight variations in components and reliably perform joining tasks that were previously the domain of human workers. A particularly promising approach lies in the geometric analysis of joining partners, more precisely to identify and match features such as holes, edges, and contours to determine the correct spatial alignment.

The present work builds upon this idea by proposing a method for the visual detection and analysis of basic geometric shapes on joining counterparts, using a single camera mounted on the robot arm. By evaluating spatial relationships between these shapes and comparing them across components, it becomes possible to determine the correct transformation for precise alignment and assembly. This geometric reasoning forms the basis for a lightweight, flexible, and scalable solution for automated joining tasks in a wide variety of industrial contexts.

The paper is structured as follows: First, the state-of-the-art and selected approaches to the generalized detection of joining partners and the automation of difficult joining processes with robots are considered. This is followed by a presentation of the methods considered here. Subsequently, the detection and localization of geometric shapes is discussed in more de-

^a <https://orcid.org/0009-0001-7708-2863>

^b <https://orcid.org/0000-0001-9470-556X>

tail, and an accuracy analysis is presented. This is followed by a summary and an outlook on future work.

2 STATE OF THE ART

In industrial robotics, it is a key challenge for a robot to automatically recognize where components are located and how they are aligned to properly join them. In this chapter, we provide an overview of the current technical approaches to how robots recognize and localize single components and identify joining partners.

In the past, such tasks were mainly solved using CAD models. The robot searches for the model in the camera view and calculates the position of the real object. Algorithms such as Iterative Closest Point (ICP) are used here (Xiang et al., 2017). Such methods can be very accurate, but require good camera data and often only work if the object is not obscured or rotated. In addition, it is cumbersome to model each new part first.

In recent years, methods that work with artificial intelligence (AI) and normal camera images (RGB) have become established. A neural network learns from many sample images to recognize objects and predict their orientation (position and rotation). Well-known approaches such as PoseCNN (Xiang et al., 2017) or YOLO-6D (Tekin et al., 2018) work directly with images and can thus identify specific components. Newer models such as ZebraPose (Su et al., 2022) also try to deal with hidden or difficult to distinguish parts. The problem with this is that these methods usually only work with previously known objects. If a new part is added, such as a different type of connector, the system often has to be retrained. This is a major obstacle for flexible assembly systems.

Another approach focuses less on the concrete “recognition” of an object and more on shapes: Holes, edges, cylinders, surfaces. Systems such as PVNet (Peng et al., 2019) or EPOS (Hodan et al., 2020) analyze such features in the image and derive the position of the object from them. This also works well if, for example, the object is symmetrical or partially obscured. Methods such as SurfEmb (Haugaard and Buch, 2022), which learn a kind of “fingerprint” of the surface - regardless of the specific object - are particularly interesting here. This way of thinking fits well with our goal: instead of recognizing specific parts, we want to use general geometric features to find out which parts fit together in principle.

Instead of relying solely on 2D camera images, many systems today combine color information (RGB) with depth data. Such RGB-D systems use

a depth camera, for example, as in Microsoft Azure Kinect DK. Methods such as DenseFusion (Wang et al., 2019a) and MoreFusion (Wada et al., 2020) combine both types of information to achieve more precise results - even in difficult lighting conditions or when parts overlap.

Completely new methods go one step further: they try not only to recognize a specific part, but also to understand what type of component it belongs to - for example: “This is a cylindrical plug” or “a round opening”. One example of this is the NOCS system (Wang et al., 2019b), which introduces a type of neutral coordinate shape that can be used to recognize any object within a category (e.g. “bottles”). Further developments such as FS-Net (Liu et al., 2019) or SPD (Irshad et al., 2022) show that it is possible to recognize new variants with just a few training examples. This is a major advance for applications in which the components change frequently - in flexible production lines, for example.

Other methods deal directly and very specifically with identifying joining partners. In (Kuo et al., 2019) potential joining positions on printed circuit boards are identified and then matched on the basis of structural relationships to the components. With Form2Fit (Zakka et al., 2020), a framework was developed that learns general guidelines for shape matching between joining partners in order to solve problems, for example, when packaging goods. However, the system is limited to two-dimensional space and requires the joining partners to lie flat on a work surface.

Research is clearly moving in one direction: away from rigid, model-based systems and towards flexible, learning-based processes that can also deal with unknown or slightly modified components.

In the next chapter, we show how our own approach picks up on these developments and combines them with a novel method: We analyze the geometric shapes found on components directly in the sensor data to make universal statements about which parts fit together, regardless of what they are called or which product family they come from.

3 METHOD

Our method for automated recognition and joining of two related components can be divided into the following steps (see Figure 1): It starts with image acquisition. For this purpose, stereo image pairs of the scenes in which the two matching joining parts are assumed to be located are recorded. In the second step, geometric shapes such as circles, rectangles, tri-

angles etc. are then searched for in the images. This is followed by the localization of the shapes in space. The shapes found in a scene are then placed in relation to each other before the features of both scenes are matched. If different features are successfully matched, a set is created, and a position and orientation in space is determined. This is followed by the robot picking up a joining partner and the actual joining process. As the accuracy achieved in the localization of the components is sometimes worse than the joining tolerances, the joining is carried out using force-controlled application. The grasping of a joining partner and the actual joining have not been implemented at the time of publication. The focus of this paper is on the description of the general method and on the proof of sufficient accuracy in the localization of the geometric shapes.

The previously described flow is the intended process in the event that the relevant sides of the joining partners can be captured very well with a camera mounted on the robot. However, the aim of the method is also to be able to automatically join electrical plug connections, such as the high-voltage plug in Figure 2. Here it can be assumed that the plugs are placed on a table, lying in a box or hanging. As a result, it is possible that the end faces relevant for joining cannot be observed very well using stereoscopy and a good viewing angle. Therefore, the method should be adapted later as follows: (see Figure 2). First, a possible connector in the working area is to be identified using object recognition and a simple gripping pose for picking up the connector is to be determined. After the robot has picked up the connector using a simple two-jaw gripper, the connector is held in front of a static camera to identify further features on the front side using the principle of stereoscopy. As described above, the feature sets are first determined as well as the position of these sets in relation to the gripper TCP. This is to compensate for inaccurate gripping or picking up of the connector at the start of the process. The connector socket is then localized using a camera mounted on the robot, again using stereo image pairs.

To detect and pick up a plug, further object recognition is necessary. This would limit the general validity of the method, although there are already promising approaches to recognizing a large number of different plugs (Wang and Johansson, 2023).

3.1 Image Aquisition, Detection and Localization

For this purpose, a 2D camera mounted on a robot arm takes two images of the scene from different an-

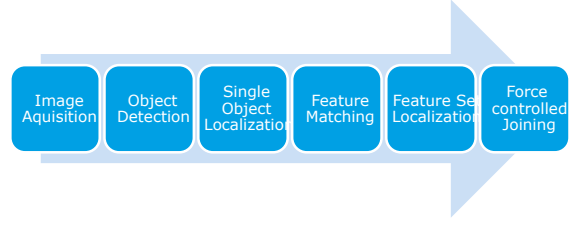


Figure 1: Summary of the method presented in this paper.

gles, in which the two matching joining parts are assumed to be located. Along with the stereo image pairs, the robot's respective positions are also read out and stored. Object recognition is then performed using a YOLOv11 instance segmentation model. This model was trained using two-dimensional printouts of the shapes and three-dimensional objects, where only the front surface was labelled. The object recognition results are masks that are subsequently used to determine the centers of the geometric shapes in image coordinates. Next, the center points are triangulated using the stereo image pairs, and the center point coordinates are transformed into the robot base coordinate system. The accuracy of this localisation process is examined in more detail in the following chapter, as high accuracy is crucial for subsequent matching and joining processes.

3.2 Feature Matching

Once the individual geometric shapes have been identified and located in their respective scenes, they must be related to each other. Based on the established relationships, similarities between the two recorded scenes are analyzed. The following procedure is utilized for this purpose: For each scene, all possible sets consisting of three objects/centers are formed (see Figure 3). These sets are then compared between the scenes and those with the same combinations of geometric shapes are filtered out. This reduces the search space for the subsequent analysis, which initially involves calculating the connection vectors $\underline{r}_{i,i+1}$ between all centre points in each set of three. Then, always starting from one object/center point, the connection vectors to the other two center points are combined with the cross product of the two vectors to form a non-rectangular coordinate system (Figure 3). For a set consisting of a square S, a circle C and a triangle T, for example, the following coordinate system C_{S1} can be formed starting from the square:

$$\underline{n}_\times = \underline{r}_{C,S} \times \underline{r}_{T,S} \quad (1)$$

$$C_{S1} = (\underline{r}_{C,S}, \underline{r}_{T,S}, \underline{n}_\times) \quad (2)$$

In the case of three different object shapes (e.g. square, circle, triangle), it is sufficient to determine

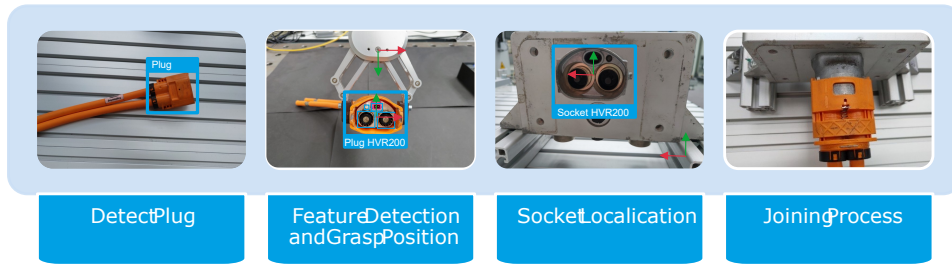


Figure 2: Method sequence for electrical plug connections.

one coordinate system in order to be able to create a clear comparison with a corresponding set from the second scene. It should be considered that these two coordinate systems are created according to the same principle or the same sequence of vectors. For sets with two identical shapes, at least two such coordinate systems must be determined; for three identical shapes, as many as six coordinate systems are required for the comparison. Three combinations are created by selecting the “origin” of the coordinate system and there are two sequences for forming the cross product for each origin.

To make the comparison, the general transformations $\mathbf{M}$ between the two coordinate systems are calculated.

$$\mathbf{M} = \mathbf{C}_j(\mathbf{C}_i)^{-1} \quad (3)$$

If the two coordinate systems match, $\mathbf{M}$ should fulfill the criteria of a rotation matrix: The determinant of $\mathbf{M}$ must therefore be $\det(\mathbf{M}) = 1$. In addition, the following matrix equation must be fulfilled: $\mathbf{M}^T \mathbf{M} = \mathbf{I}$ where $\mathbf{I}$ corresponds to the unit matrix. Due to the way in which the coordinate systems are constructed, the relationships of mirrored center point constellations can also be described with a rotation matrix. This becomes clear, for example, if the coordinate system in scene 1 in Figure 2 is mentally rotated by 180° around the line drawn between the two scenes.

For matching sets consisting of three different shapes, the comparison is unambiguous with only one pair of coordinate systems. Both coordinate systems

can be created from the same vector sequence, and a rotation should be detectable.

For sets with two identical shapes, a total of four pairs of coordinates can be compared, whereby two pairs are sufficient to achieve a clear result. The origin of the coordinate systems is selected in the form that only exists once. For one scene, two coordinate systems are created using the two possible vector sequences to form the cross product, which are then compared with a coordinate system in the second scene. The determinant can take the value 1 for both pairs. On the other hand, the matrix product $\mathbf{M}^T \mathbf{M}$ should only result in the unit matrix in one case. As an additional criterion, a length comparison of the connection vectors according to their order in the matrices $\mathbf{C}_i$ and $\mathbf{C}_j$ can also be used.

For sets with three identical shapes, a total of 12 coordinate systems can be compared with each other, as 6 different vector sequences can be constructed for each scene to calculate the cross product and thus 6 different coordinate systems. Again, it is also sufficient to compare one coordinate system of a scene with the six of the second scene.

Due to the inaccuracies in the determination of the center point coordinates, the tests for a rotation matrix on the real system will not be as unambiguous as described above. It is therefore necessary to define certain tolerance limits around the target values when determining the determinate and the matrix product. However, the values for these tolerance limits still have to be determined in further experiments.

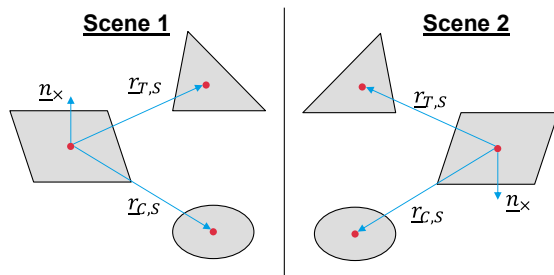


Figure 3: Schematic representation of the coordinate systems for sets of three different shapes.

4 ACCURACY OF THE LOCALIZATION

As described in the previous chapter, the geometric shapes are localized using triangulation based on stereo image pairs. Only a single 2D camera mounted on the robot end effector is used to record the stereo images. Different viewing angles can be set by moving the robot. Due to the time delay between two images, this type of stereoscopy only allows the obser-

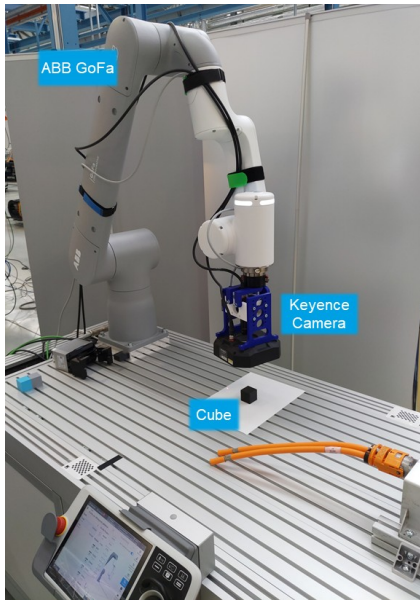


Figure 4: Setup for the accuracy analysis.

vation of static scenes. Further details on the methodology can be found in (Marx et al., 2024). In (Marx et al., 2024) we have shown that using a Kuka Lbr iiwa and an Intel Realsense D435, accuracies of 2.436 mm with a standard deviation of 0.665 mm can theoretically be achieved with this method.

An ABB GoFa in conjunction with an industrial camera from Keyence with a resolution of 2432x2040 pixels (see Figure 4) was used to evaluate the accuracy of the triangulation of the center points and thus the geometric shapes. The increased resolution and a smaller working distance should further improve the accuracy achieved in (Marx et al., 2024).

An important requirement for good triangulation is also good correspondence in the determination of the centers of the geometric shapes for two different viewing angles. If the calculated center points do not aim at the same point on the real object, a triangulation error occurs. The object recognition model used has the greatest influence on the determination of the center points. The used YOLOv11 Nano instance segmentation model was trained with a total of 9783 images and approx. 14000 labeled objects. For this purpose, 1087 images¹ were multiplied with various augmentations, such as rotations, distortions, noise, or brightness adjustments. The images include paper-printed representatives of various shapes, such as rectangles, triangles and circles, as well as 3D-printed solids with the same cross-sections and a height of 35mm. The background was white for the prints on

paper, while different backgrounds with more noise were used for the 3D prints. Figure 5 shows a representative training batch. The images are mostly taken very centrally from above, so that the 2D objects are initially not very distorted and few side surfaces can be seen on the 3D objects. In the case of the 3D objects, only the surfaces or cross-sections were labeled. The idea behind this is that the simple 2D shapes can later be recognized on real three-dimensional objects, such as a bolt, without having to recognize and classify the object itself.

The model was trained with 100 epochs, resulting in the training results shown in Figure 6. Clearly, the loss curves are still steadily declining. The validation losses demonstrate that the model has significantly converged, as reflected by the precision, recall, and mAP values. With values above 0.95 for precision and recall, the model performs very well. The accuracy of the segmentation masks (mAP50-95 (M)) is above average, with values over 0.9. This is likely due to the relatively simple segmentation task. The confusion matrix shows that over 90% of existing objects in each class are correctly identified. However, the values for false positive results for circles and rectangles are significantly too high. These objects are often mistakenly recognized in the background. This phenomenon can be explained by the presence of some rectangular and circular objects in the background of the training data that we did not label, such as an "O" in text. Figure 7 shows an example of a segmentation mask for a square. The mask clearly shows that the segmentation along some edges still causes problems. These errors in the mask affect the subsequent cal-



Figure 5: Training batch for the segmentation model showing different shapes in front of various backgrounds.

¹Link to the dataset (without augmentations): <https://shorturl.at/9kakS>

culuation of the center point, which was implemented here simply by calculating the center of gravity. Since no center points are marked on the real objects and printed shapes, there is no ground truth with which the calculated center points can be compared, and so no accuracy consideration was made here. The determination of the center points is therefore included as an unknown in the subsequent accuracy analysis for the triangulation.

To record data for the evaluation, the robot with camera was moved centrally over an object placed on the work surface. Starting from this position, a program then always determines 51 random points within a certain radius of the starting position. When generating the points, it is ensured that two consecutive

points always have a certain distance between them. The points are then followed one after the other and an image is taken at each position. At each position, the camera's viewing axis is additionally tilted by a random angle between $5-15^\circ$ towards the start position. With our combination of working distance and camera angle of view, this reduces the probability that the object to be detected falls out of the camera's field of view.

A stereo image pair is then always made from two consecutive images and the center points are triangulated. The triangulation always results in two positions, each based on the different viewing angles. If these positions are close to each other, this is a characteristic for a precise determination of the center point coordinates. However, initial tests revealed significant variation in triangulation error. While an absolute error of less than 1 mm was often achieved, triangulations with errors of more than 10 and 20 mm were not uncommon. For this reason, pure triangulation was expanded to include mid point calculations. First, the location of the shortest distance between the two straight lines passing through the triangulated point from the respective camera coordinate system is calculated. Then, the mid point of this shortest distance serves as the new triangulation result. The values were compared with the center point determination based on the camera's intrinsic and extrinsic parameters. This method only works on surfaces calibrated to the camera or robot respectively and when the height of the observed objects is known. Previous work achieved accuracies ranging from 0.8 to 1.5 mm for the x and y coordinates, depending on the working distance.

The accuracies for squares, triangles, circles and hexagons were evaluated. Each of the four shapes was placed in 5 different and random positions on the work surface, which can be seen in Figure 4. This means that 250 triangulations were calculated for each shape, resulting in a total of 1000 measurements. For each random position and shape, the mean values of the total error and its standard deviation regarding the 2D method, mentioned before, were then deter-

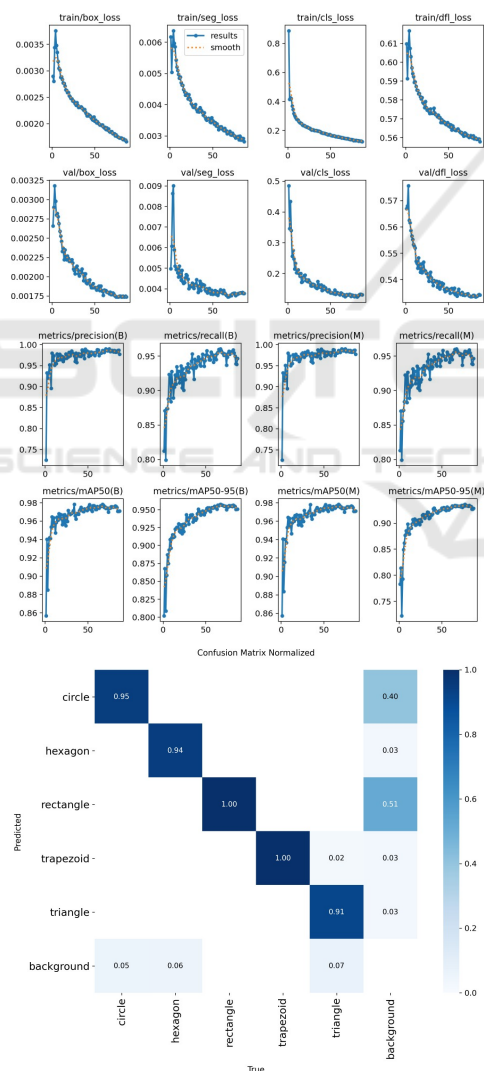


Figure 6: top: Graphs of the training results of the segmentation model. The various loss curves, recall curves, precision curves, and mAP values are displayed., bottom: Confusion Matrix for the 5 different geometric shapes.

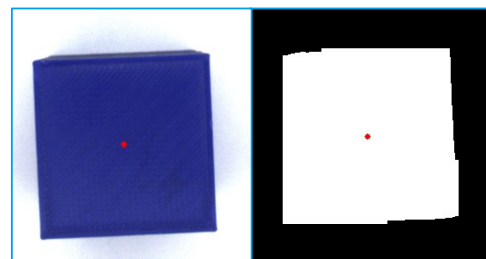


Figure 7: left: image of a cube, right: segmentation mask of the square surface of the cube.

Table 1: Results of the accuracy analysis (all values in mm).

		Square	Circle	Triangle	Hexagon
Position 1	Mean Error	0.661	1.009	2.830	5.396
	Std Dev	0.633	0.773	0.883	1.304
Position 2	Mean Error	1.879	1.367	2.875	6.202
	Std Dev	2.629	1.392	0.917	0.492
Position 3	Mean Error	2.175	2.183	5.177	6.737
	Std Dev	0.915	0.804	0.723	1.139
Position 4	Mean Error	2.837	2.931	6.147	5.984
	Std Dev	0.840	0.687	1.108	0.809
Position 5	Mean Error	4.995	5.118	7.670	7.774
	Std Dev	0.908	0.752	0.920	1.130
Overall	Mean Error	2.509	2.522	4.940	6.419
	Std Dev	1.185	0.882	0.910	0.975

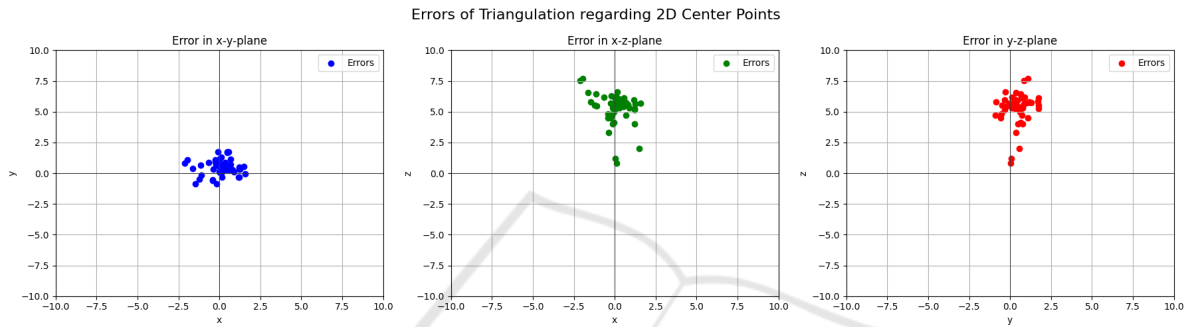


Figure 8: Triangulation error compared to the 2D method for the hexagon at position 1.

mined. The mean values of the mean values and standard deviations were then computed for each shape and the mean values across all shapes were calculated. The results are summarized in Table 1.

An average accuracy of 4.097 mm with a standard deviation of 0.988 mm was achieved across all the shapes considered. The accuracy for triangles and hexagons was particularly poor at 4.94 and 6.419 mm, whereas circles and rectangles were located much more accurately at 2.509 and 2.522 mm. The standard deviation, on the other hand, was fairly constant across all four forms. Examining the composition of the resulting error shows that the large deviations are primarily due to differences in the Z coordinates. Therefore, considering only the x and y coordinates results in an average error of 0.713 mm, with a standard deviation of 0.47 mm. In comparison, the average error in the Z coordinates is 3.873 mm, with a standard deviation of 1.102 mm. Figure 8 illustrates this phenomenon by showing the triangulation error compared to the 2D method for 50 measurements at position 1 for the hexagon. To determine the correct Z value, the table surface was approached again using a TCP measured on the robot. This measurement yielded a table height of -22.5 mm relative to the robot base coordinate system. The triangulation yielded an average table height of -21.94 mm, significantly closer to the -22.5 mm value than the -26.97

mm value determined by the 2D method.

5 CONCLUSION AND FUTURE WORKS

In this paper, a method was presented which, based on the recognition of simple geometric shapes, should enable the universal identification of matching joining parts in the future. For this purpose, AI object segmentation is combined with simple mathematical algorithms to identify matching features. The advantage of this recognition process is that new component geometries do not have to be integrated or trained into the object recognition process repeatedly. The process is able to match as yet unknown objects to each other if the cross-sections to be joined contain simple geometric shapes such as circles, rectangles or triangles and is therefore particularly suitable for multi-variant productions. This method only works if the objects to be joined do not move in space, and if at least three geometric shapes can be recognised on them.

An accuracy analysis was carried out for the included AI object recognition with a YOLO model and the subsequent stereoscopy, as this is essential to enable precise joining afterwards. The individual geo-

metric shapes could be triangulated with an accuracy of less than 1 mm in some cases. The average error was 4.097 mm, but this was due to the poor depth values or z-coordinates of our comparison method. When only considering errors in the x-y plane, the average is just 0.713 mm. In principle, these results allow the assessment that the method described here achieves the accuracies required for a joining process, especially if deviations occurring during joining are also to be compensated for by force control.

In the next steps, the functionality of the method, which was initially demonstrated on constructed examples, will also be checked on real feature sets consisting of individual geometric objects and then also tested on real components, such as connectors. As already described in Section 3, suitable tolerance windows for the similarity check must then be determined.

ACKNOWLEDGEMENTS

The work presented was carried out as part of the VADER² research project supported by the Federal Ministry for Economic Affairs and Climate Action on the basis of a decision of the German Bundestag and funded by the European Union, in cooperation with the RICAIP project³ funded by European Union's Horizon 2020 research and innovation programme under grant agreement No 857306.

REFERENCES

- Haugaard, R. L. and Buch, A. G. (6/18/2022 - 6/24/2022). Surfemb: Dense and continuous correspondence distributions for object pose estimation with learnt surface embeddings. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6739–6748. IEEE.
- Hodan, T., Barath, D., and Matas, J. (2020). Epos: Estimating 6d pose of objects with symmetries. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11700–11709. IEEE.
- Irshad, M. Z., Kollar, T., Laskey, M., Stone, K., and Kira, Z. (5/23/2022 - 5/27/2022). Centersnap: Single-shot multi-object 3d shape reconstruction and categorical 6d pose and size estimation. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 10632–10640. IEEE.

²Vernetzter digitaler Assistent für das datengetriebene Engineering von roboterbasierten Produktionsanlagen

³Research and Innovation Centre on Advanced Industrial Production

- Kuo, C.-W., Ashmore, J. D., Huggins, D., and Kira, Z. (2019). Data-efficient graph embedding learning for pcb component detection. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 551–560. IEEE.
- Liu, C., He, L., Xiong, G., Cao, Z., and Li, Z. (4/29/2019 - 5/2/2019). Fs-net: A flow sequence network for encrypted traffic classification. In *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications*, pages 1171–1179. IEEE.
- Marx, S., Gusenburger, D., Bashir, A., and Müller, R. (2024). Low-cost stereo vision: A single-camera approach for precise robotic perception. In Yi, J., editor, *2024 IEEE 20th International Conference on Automation Science and Engineering (CASE)*, pages 890–896. Piscataway, NJ. IEEE.
- Peng, S., Liu, Y., Huang, Q., Zhou, X., and Bao, H. (2019). Pvnnet: Pixel-wise voting network for 6dof pose estimation. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4556–4565. IEEE.
- Su, Y., Saleh, M., Fetzer, T., Rambach, J., Navab, N., Busam, B., Stricker, D., and Tombari, F. (2022). Zebrapose: Coarse to fine surface encoding for 6dof object pose estimation. arXiv preprint arXiv:2203.09418.
- Tekin, B., Sinha, S. N., and Fua, P. (6/18/2018 - 6/23/2018). Real-time seamless single shot 6d object pose prediction. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 292–301. IEEE.
- Wada, K., Sucar, E., James, S., Lenton, D., and Davison, A. J. (2020). Morefusion: Multi-object reasoning for 6d pose estimation from volumetric fusion. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14528–14537. IEEE.
- Wang, C., Xu, D., Zhu, Y., Martin-Martin, R., Lu, C., Fei-Fei, L., and Savarese, S. (2019a). Densefusion: 6d object pose estimation by iterative dense fusion. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3338–3347. IEEE.
- Wang, H. and Johansson, B. (8/26/2023 - 8/30/2023). Deep learning-based connector detection for robotized assembly of automotive wire harnesses. In *2023 IEEE 19th International Conference on Automation Science and Engineering (CASE)*, pages 1–8. IEEE.
- Wang, H., Sridhar, S., Huang, J., Valentin, J., Song, S., and Guibas, L. J. (2019b). Normalized object coordinate space for category-level 6d object pose and size estimation. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2637–2646. IEEE.
- Xiang, Y., Schmidt, T., Narayanan, V., and Fox, D. (2017). Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes. arXiv preprint arXiv:1711.00199.
- Zakka, K., Zeng, A., Lee, J., and Song, S. (5/31/2020 - 8/31/2020). Form2fit: Learning shape priors for generalizable assembly from disassembly. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9404–9410. IEEE.