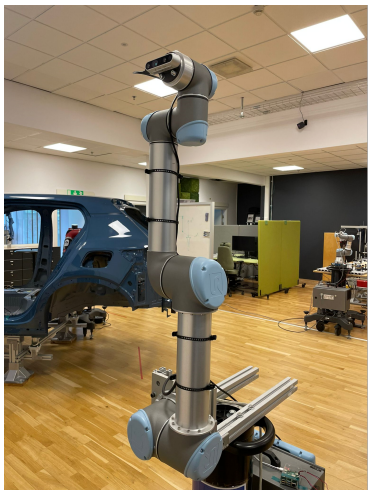
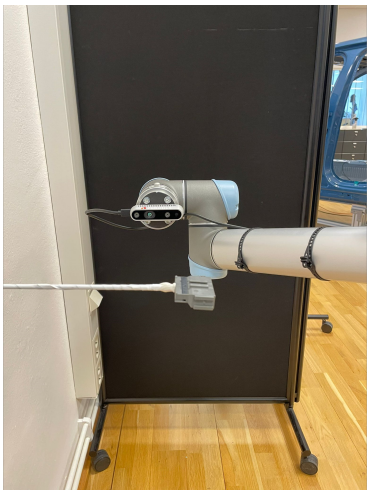
CHALMERS
UNIVERSITY OF TECHNOLOGY


Using Robots to Collect Data for Machine Vision Tasks

An Application in Robotized Assembly of Automotive Wire Harnesses

Master's thesis in Industrial and Materials Science

KARIM EL-NAHASS, and
GONZALO URBANOS URIEL

Department of Industrial and Materials Science
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2024
www.chalmers.se

Master's thesis 2024

Using Robots to Collect Data for Machine Vision Tasks

An Application in Robotized Assembly of Automotive Wire Harnesses

KARIM EL-NAHASS, and
GONZALO URBANOS URIEL

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Industrial and Materials Science
Division of Production Systems
Chalmers University of Technology
Gothenburg, Sweden 2024

Using Robots to Collect Data for Machine Vision Tasks
An Application in Robotized Assembly of Automotive Wire Harnesses
KARIM EL-NAHASS
GONZALO URBANOS URIEL

Supervisor: Hao Wang, Department of Industrial and Materials Science
Examiner: Björn Johansson, Department of Industrial and Materials Science

Master's Thesis 2024
Department of Industrial and Materials Science
Division of Production Systems
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Cover: Setup of the automated dataset collection model, see Section 3.1.1

Typeset in LaTeX
Printed by Chalmers Reproservice
Gothenburg, Sweden 2024

Using Robots to Collect Data for Machine Vision Tasks
An Application in Robotized Assembly of Automotive Wire Harnesses
KARIM EL-NAHASS
GONZALO URBANOS URIEL
Department of Industrial and Materials Science
Chalmers University of Technology

Abstract

The current final assembly process of automotive wire harnesses into vehicles predominantly relies on manual labor and skill. This reliance leads to safety and ergonomic issues when lifting heavy wire harnesses and applying high-pressure manipulations to components for 8 hours a day. This thesis combines collaborative robotics and artificial intelligence to collect connector data for machine vision tasks in the automotive industry, addressing the problem of insufficient data. The research investigates an approach using a robotic setup for automated data collection. The framework includes data acquisition utilizing a UR5 robot and an Intel RealSense D435 camera; robot-camera communication using a Raspberry Pi 4b as a bridge; and an automatic labeling tool. The collected dataset comprises 8 different connectors commonly used in automotive wire harnesses. The resultant datasets (first the manually annotated dataset and second the automatic annotated dataset) are evaluated using YOLOv8, a deep-learning based object detection model. The evaluation results present a higher accuracy ($mAP_{50} = 93.5\%$) for the manually annotated dataset compared to the automatic labeling approach ($mAP_{50} = 74.4\%$) which suggests that there is still room for improvement on the automatic labeling tool used. This accuracy difference is concluded to be due to the inability to control lighting conditions in the workspace in the lab.

Keywords: accuracy, automatic annotation, connector, dataset, ergonomics, object detection, robotic system, wire harness.

Acknowledgements

This master's thesis marks the end of our academic career. During these years of studying both the master's program and the bachelor's degree, we have acquired great knowledge about engineering and so many skills that we are going to bring with us to a professional environment to further develop our career. We would like to use this space that is provided to us to thank all those people who have accompanied us along this path, who have given us great advice and have taught us many lessons.

First of all, we would like to greatly thank our supervisor Hao Wang for the opportunity he has given us to work on this master thesis. We are particularly grateful for the trust he placed in us from the very beginning, his willingness and readiness to help us at any time, and his valuable advice. Finally, we would like to highlight the kindness with which he has always treated us and how easy it has been to work with him during this project.

We would also like to express our sincere gratitude to everyone who contributed to or assisted us during this master's thesis. Starting with our examiner, Björn Johansson, who always had a joke in mind and continuing with our industrial supervisors from Volvo Cars, Dan Lämkull and Alf Andersson and Wiretronic AB, Patrick Andersson.

Furthermore, we are grateful to the entire SII-Lab team. It has been a privilege to meet and work with all of you. Special mention goes to Omkar and Sven, who were always ready to help and made our work much easier. We apologize for gradually monopolizing all the resources in the lab.

Karim El-Nahass & Gonzalo Urbanos Uriel, Gothenburg, June 2024

List of Acronyms

Below is the list of acronyms that have been used throughout this thesis listed in alphabetical order:

2D	Two-Dimensional
3D	Three-Dimensional
6D	Six-Degree-of-Freedom
AI	Artificial Intelligence
AMT	Amazon Mechanical Turk
AP	Average Precision
ARP	Augmented Reality Pen
ARS	Augmented Reality Semiautomatic-labeling
CAD	Computer Aided Design
DoF	Degrees of Freedom
EV	Electric Vehicle
EWASS	Empowering human Workers for Assembly of Wire Harnesses
HRC	Human-Robot Collaboration
HSV	Hue Saturation Value
IP	Internet Protocol
ISO	International Organization for Standardization
mAP	mean Average Precision
PnP	Perspective-n-Point
P-R	Precision-Recall
QR	Quick Response
RGB	Red-Green-Blue
RGB-D	Red-Green-Blue-Depth
SSD	Single Shot Detector
SDF	Single Distance Fusion
TCP	Tool Center Point
UR5	Universal Robot 5
uAP	unidirectional Pose Accuracy
uRP	unidirectional Pose Repeatability
YOLO	You Only Look Once

Contents

List of Figures

List of Tables

List of Algorithms

1
Introduction

1.1 Background

Nowadays society is experiencing a drastic switch from combustion engine vehicles to electric vehicles (EVs) given ongoing technologic trends in vehicle electrification and autonomous driving [2]. Even though wire harnesses are extensively used in combustion engine vehicles, EVs require a larger number of wire harnesses to be assembled. These wire harnesses are part of the essential infrastructure of an EV as they support power and signal transmission within the electronic system. Therefore, it is of great importance to ensure the quality of the assembly of these wire harnesses in automobiles. However, the current final assembly process of automotive wire harnesses into vehicles predominantly relies on manual labor and skill, as it involves manual operations such as lifting heavy wire harnesses and applying high-pressure manipulations to various components, resulting in significant safety and ergonomic issues for human operators. Consequently, this presents significant challenges in effectively controlling and enhancing the quality and productivity of the assembly.

Figure 1.1: Wire harnesses in an EV

Current industry is looking for methods to promote a more flexible robotic assembly on objects that are difficult for pre-configuration and pre-programming. Combining both, collaborative robots and artificial intelligence can help solve many ergonomic problems that arise in automotive wire harness assemblies [3]. It is possible in the short term to train deep-learning based object detection models to detect and identify wire harness connectors [1]. However, manipulating them and carrying out assembly processes using collaborative robots is still a challenging task [3]. After all, industry looks for a way to automate assembly of wire harnesses to ensure high

quality and increase productivity by reducing assembly time.

1.2 Problem Description

This thesis aims to help solve a recurring problem in the final assembly of electrical connectors in the automotive industry. The problem itself consists of the non-ergonomic postures adopted by operators who work 8 hours a day with wire harnesses, as this workload negatively affects their long-term health. To address this problem, the idea is to use machine vision systems along with robotics and artificial intelligence. However, to use AI and train deep learning models, large amounts of data are required. This is where the problem that gives purpose to this thesis arises: insufficient data. It is essential to develop a semi-automated technique for data collection and annotation since manual methods are tedious, labor intensive, time-consuming, and often less accurate. Additionally, electrical connectors come in various shapes, colors, and levels of detail, requiring a scalable benchmark that can take in new connectors as they are added to the dataset. An annotated dataset of task-specific objects is crucial for the consistent and rigorous development and evaluation of learning-based vision systems.

1.3 Purpose and Aim

The purpose of this thesis is to aid humans in the dataset collection procedure for a 2D object detection task. Automating the dataset collection will facilitate collect large amounts of data continuously with minimal human intervention, ensuring that datasets are sufficiently large and varied while speeding up the data collection process. Furthermore, the purpose of this thesis is to reduce human error and achieve standardization when labeling data by proposing an annotation tool. Altogether, visual machine perception enabled, industrial robots can be improved to achieve higher levels of autonomy to manipulate objects required in flexible automation applications such as wire harnesses assembly.

This thesis aims to suggest technical solutions for enhancing robotic visual perception in assembly tasks. Furthermore, the aim of this thesis is to design an automated dataset generation system, which includes data collection and annotation, and evaluation of the resultant dataset.

1.4 Research Questions

The collected dataset will be used to train object detection models to detect electrical connectors inside the car's skeleton while performing assembly operations. Understanding the challenges associated with enabling robotic visual perception is crucial before proposing and testing solutions. The following research questions will be addressed during the development of this master thesis:

RQ1. What are the current state-of-the-art methodologies for benchmark dataset collection?
This research question aims to gather essential information about previous works to understand the methodologies used in benchmark dataset collection. By exploring existing literature and studies, we seek to identify the techniques, tools, and best practices that define the current state of the art. This understanding will provide a foundation for developing improved methodologies and highlight any gaps or limitations in the current approaches.

RQ2. How can we automate the benchmark dataset collection that includes both 2D and 3D data, and evaluate the 2D dataset?
The aim of this research question is to explore effective strategies for gathering a comprehensive benchmark connector dataset containing both 2D and 3D data. It also focuses on the methods for evaluating the collected 2D data. This will help ensure the accuracy, reliability, and usefulness of the dataset for various applications.

RQ3. What are the advantages of employing a semi-automated data collection methodology compared to traditional manual data collection processes?
This research question aims to investigate the main benefits of using a semi-automated data collection methodology over traditional manual approaches. By comparing efficiency, accuracy, and scalability, we seek to highlight the improvements that semi-automation can bring to the data collection process. This will provide insights into how semi-automated methods can enhance data quality and reduce human effort in various applications.

1.5 Scope and Delimitations

This thesis topic combines the fields of robotics and automation together with computer vision and AI. The scope of the thesis is to investigate and design a semi-automated robotic system which is able to carry out data collection following the standards of synchronized Human Robot Collaboration (HRC) [4].

This thesis does not involve designing any type of deep learning model. Instead, a publicly available object detection model is used to evaluate the collected dataset. Although a 3D dataset is not collected per se, depth images indicating the distance of an object from the camera are gathered. Properly utilizing depth images to train an AI model requires cleaning and preprocessing; however, this aspect is not the focus of this thesis. Additionally, although in this thesis a method is proposed for 6D pose annotation, the evaluation of 6D pose in context is not conducted in this thesis. This would require a 3D dataset with CAD models, which is beyond the scope of this work.

2

Theory

2.1 Deep Learning-Based Connector Detection for Robotized Mating of Connectors

Despite that numerous studies have been conducted in the domain of wiring harness assembly systems, such as employing vision and force control to achieve the mating of electric connectors [5], [6], [7], [8], high-speed manipulation of connectors using robots [9], manipulation of deformable linear objects such as cables [10], and recognition systems for electric connectors utilizing image processing [11], only one previous study, to our knowledge, has endeavored to detect automotive wire harness connectors using deep learning [1].

Hao Wang et al. [1] investigated deep learning-based connector detection techniques for robotized assembly of automotive wire harnesses. In this work, a dataset for connector detection was collected with the aim of creating an intelligent detection system that subsequently enables the manipulation of electrical connectors in the assembly of electric vehicles to address the current ergonomic issues faced by operators. To collect this dataset, an iPhone 11 camera was used to acquire RGB pictures with a resolution of 4032×3024 pixels, where the distance between the camera and connectors varied between $20cm$ and $40cm$. Following, 15 images (6 main views + 9 random views) for 20 connectors were taken and after some data augmentation techniques based on modifying the HSV color space slightly, applying horizontal flipping and applying the mosaic technique, a 360 image dataset was collected.

2.2 Datasets for Object Detection

2.2.1 2D Datasets

There has been significant research conducted on 2D object detection over the past decade, since this became a research field with potential outcomes in industry. Early benchmarks for 2D object detection rely on datasets available on the internet, such as ImageNet [12], Objects365 [13] and PASCAL [14]. These datasets aim to be large-scale and diverse to address a wide range of applications. Additionally, COCO dataset [15] aims to collect images depicting complex everyday scenes containing common objects in their natural context. Despite their differences, all these datasets

share a common characteristic: manual annotation of bounding boxes and classes by human annotators. This annotation process is typically conducted either by the research group responsible for the dataset or through online annotation platforms like Amazon Mechanical Turk (AMT), where one can put up tasks for users to complete and get paid. While such approaches are suitable for large-scale labeling, they still rely on human annotators. Although annotation tools like LabelMe or Labelimg exist to assist in annotating object instances, to the best of our knowledge, there is no publicly available automatic annotation tool that meets the requirements of this thesis. Therefore, in this these, a robotic system for data collection and an automated procedure for annotating bounding boxes and object classes is proposed.

As research in the field progresses, there is a growing demand for datasets that are increasingly tailored to specific applications and their contexts. This is the case, for example, with works that are dedicated to the detection of pedestrians [16], cars [17] and faces [18]. In line with our field of research, related to the detection of electrical connectors to facilitate their assembly for operators in the automotive industry, our dataset will focus only on electrical connectors.

Valery Ilin et al. [19] designed a robotic system for automated 2D object dataset collection with annotations. The motivation behind this design is to aim for better accuracy and robustness during training. In their paper they mention an article pusblished by WIRED magazine [20] which describes errors in one of ImageNet fundamental datasets. It was estimated that the dataset contains roughly 6% errors out of 14 million labeled images. The reason behind is clearly manual annotation and inexperienced users. In their work, an automated labeling system was developed using a 6 DoF robot equipped with a camera in its end-effector to identify and label QR codes, bar codes, price tags, along with auxiliary sensors to track the end-effector's position.

2.2.2 3D Datasets - Model free

The current state of the art in 3D dataset collection is quite broad since using 3D data is applicable for many fields such as in 3D reconstruction [21], [22], autonomous driving [23], detection of object instances [24], [25], [26], [27], [28], pose estimation [25], [29], [30] and object manipulation [28], [31]. In addition, many available 3D benchmarks utilize 3D CAD models of the object by either scanning them or using synthetic data to capture its texture and details. 3D models are used to perform 2D-3D keypoint matching to be able to carry out pose estimation [30], [21], however, in this thesis the focus will only be on model-free datasets since that is the scope of the project.

In contrast to 2D datasets, 3D datasets for object recognition include depth images or point clouds. In computer graphics, a depth map operates similarly to a grayscale image, conveying information about the distance of scene objects' surfaces from the viewpoint [32]. Typically, in RGB-D images, the depth map, a 2D grayscale representation of the scene, is merged with the RGB image. Each pixel in the depth map

corresponds to the actual distance between the sensor and the object, enabling a one-to-one correspondence between pixels in the registered RGB and depth images [32]. On the other hand, point clouds represent the 3D outer shape of an object, by containing a set of vectors in a three-dimensional coordinate system [33]. These vectors are expressed in cartesian coordinates (X, Y, Z). Not only geometric information, but also, each point cloud contains RGB color pixel, gray value, depth and normals [34]. It is possible to collect a 3D dataset that includes depth maps or point clouds by using depth (Kinect or Intel RealSense depth cameras) or with LiDAR technology.

Dennis Stumpf et al. proposed SALT [35], which is a tool to semi-automatically annotate RGB-D video sequences to generate 3D bounding boxes for full 6 DoF object poses as well as pixel level instance segmentation masks for both RGB and depth. It offers built-in pre-processing functionalities to facilitate the dataset creation process. However it is still a semi-automatic approach, where, the user accesses the scene with an interactive 3D pointcloud viewer. For a sequence of N RGB-D frames the user identifies objects and people by drawing 3D bounding boxes on the first frame, while for the subsequent frames, copy and interpolation features are used. Bounding boxes are copied onto the next frames while the user applies small adjustments like translation or rotation.

The current state of the art investigates the use of augemented reality to collect a 3D dataset. In [36] Daniele De Gregorio et al. a semi-automatic labeling (ARS) method is proposed, where moving a 2D camera employing a robot and an augmented reality pen (ARP) are used to define the initial object's bounding box. In this work, two different datasets are collected, one on electromechanical components (industrial scenario) and the other on fruits (daily-living scenario). Experiments are carried out on both datasets to evaluate the accuracy of the annotation tool. This is done by comparing the manual annotated dataset with the autogenerated dataset and later compare the performance of both datasets on object detection models like YOLO and Single Shot Detector (SSD). This work resembles the goal that is meant be achieved in this master thesis, which is, collecting a dataset for electric connectors by automating as much as possible the raw data collection, pre-processing and data augmentation and finally image annotation.

2.3 Dataset collection

2.3.1 6D Pose Annotation

6D pose annotation involves labeling an object's position and orientation in three-dimensional space. The six degrees of freedom (6 DoF) defining an object's pose include the X, Y, and Z Cartesian coordinates for translation, as well as the roll, pitch, and yaw angles for orientation in space. Annotating the 6D pose of objects is crucial as it enables deep learning models to perform pose estimation in object detection applications. This capability is particularly valuable for subsequent manipulation of objects using robots.

Previous research shows different methodologies to label the pose of objects. In short, these, are based on either 3D models or 2D images. Existing datasets such as Rutgers APC [25] and Linemod [37] annotate the pose of an object either with RGB information or in RGB-D with the help of a 3D model, whereas, works like PoseCNN [38] and Gen6D [39] are deep learning models which have been trained with their own custom dataset plus external ones. Their goal is to estimate an object's pose from an input image.

In PoseCNN, the YCB-Video [38] dataset was collected to evaluate the model. It contains 6D poses of 21 objects from the YCB object set [40] in 92 videos with a total of 133,827 frames. Regarding 6D pose annotation, instead of labeling al the video frames manually, the authors focused on specifying the object poses only in the initial frame of each video. Signed Distance Fusion (SDF) representations of each objects were used to improve the accuracy of their poses in the first depth frame. Then, the camera's trajectory was initialized by setting the object poses relative to each other and tracking their positions throughout the depth video. In the end, a global optimization step refines both the camera trajectory and the relative poses of the objects. On the other hand, in Gen6D, the authors created GenMOP dataset [39], similarly to PoseCNN, to validate the effectiveness of their proposed method. In GenMOP, COLMAP was used in every video sequence to reconstruct the camera poses individually and keypoints were annotated manually on the object to facilitate alignment across sequences.

2.3.2 Dataset Evaluation

From the 2D datasets discussed in Section 2.2.1 it is key to investigate how these datasets were evaluated, which evaluation metrics were used and what results were obtained.

Mainly, in all datasets previously described, the evaluation metrics are generally the following:

- **Precision**: is the ability of a model to identify only relevant objects.
- **Recall**: is the ability of a model to find all ground-truth bounding boxes. It is the percentage of correct positive predictions among all given ground truths.

In many works, the precision-recall curve can be found, which serves as an illustrative way of viewing the trade-off between precision and recall for different confidence values. Furthermore, the main metrics used to evaluate the performance of object detection models are the unidirectional Average Precision (AP) and the Mean Average Precision (mAP):

- **AP**: is calculated for each class and it is the area under the precision-recall curve. It provides a single number summarizing the precision-recall tradeoff for tht class.
- **mAP**: is a metric which provides an overall performance measure. It is the

average AP overall classes.
The Objects365 dataset [13] was evaluated on FPN [41] and RetinaNet [42] which are two different object detection convolutional networks. The mAP results obtained for FPN and RetinaNet were 22.5% and 18.7% respectively. There are also other publicly available object detection models nowadays. For example, the electrical connector dataset collected by Hao Wang et al. [1] compared the perfomance of YOLOv5 and Faster R-CNN. The mAP's for YOLOv5 and Faster R-CNN obtained were 88.5% and 65.7% respectively. In light of these results, it is promising to carry out more experiments using newer versions of the YOLO family, for instance, YOLOv8 [43] as it has an improved architecture and uses newer deep-learning techniques.

2.4 Pose Accuracy and Repeatability UR5

When working with industrial or collaborative robots, understanding their level of accuracy and repeatability is crucial. Certain applications in production demand high levels of precision and consistency, such as welding, assembly, or object manipulation. Therefore, it's essential for us to be aware of the pose accuracy and pose repeatability achievable by the UR5 in this work.k. Additionally, the performance of the robot used is also critical to guarantee the quality of collected data.

- **Accuracy**: is a measure of the error between the value of the point the robot is programmed to go to and value of the point the robot actually goes to.
- **Repeatability**: is the ability of the robot's arm to return to the same position from the same direction.

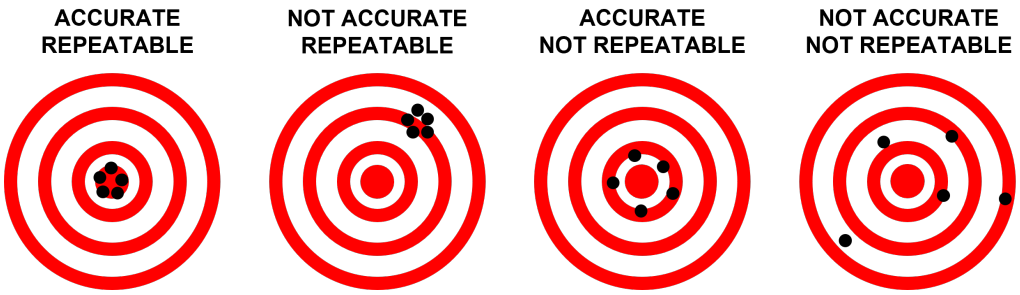


Figure 2.1: A representation of the different levels of accuracy and repeatability

Martin Pollák et al. [44] present an evaluation of pose accuracy and pose repeatability for the UR5 robot, focusing on measurements conducted in accordance with ISO 9283 standards. Employing six linear incremental sensors (Heidenhain MT 25) with associated evaluation units (Heidenhain VRZ 401) alongside a camera, the investigation centered on a standardized workspace configuration utilizing an imaginary ISO cube. Prior to measurement execution, a comprehensive test plan was devised including a list of purchased measurement components meeting standard requirements.

Conducted under varied loading conditions (50% and 100% of the robot's rated payload) and movement speeds (10%, 50%, and 100% of maximum speed), a total of 60 measurement sets were processed, each comprising 30 individual measurements.

For the unidirectional pose accuracy (uAP), the robot's manufacturer does not state a value of reference to evaluate the results, only for the unidirectional pose repeatability (uRP) (±0.1 mm). Therefore, the uRP value was used as a reference to evaluate both uAP and uRP tests. The findings reveal that the UR5 robot's pose accuracy and repeatability within the defined workspace aligns closely with manufacturer specifications (uAP $<$ 0.1 mm and uRP $<$ 0.1 mm), indicating its capability to consistently reproduce poses under differing operational conditions.

The unidirectional pose accuracy was calculated according to Formula 2.1. Where x_c, y_c, and z_c are programmed values, and x_j, y_j, and z_j are measured ones.

$$\mathbf{uAP} = \sqrt{\left(\mathbf{uAP}_x\right)^2 + \left(\mathbf{uAP}_y\right)^2 + \left(\mathbf{uAP}_z\right)^2} \tag{2.1}$$

$$\mathbf{uAP}_x = (\overline{x} - x_c) \qquad \mathbf{uAP}_y = (\overline{y} - y_c) \qquad \mathbf{uAP}_z = (\overline{z} - z_c) \tag{2.2}$$

$$\overline{x} = -\frac{1}{n}\sum_{j=1}^{n} x_j \qquad \overline{y} = -\frac{1}{n}\sum_{j=1}^{n} y_j \qquad \overline{z} = -\frac{1}{n}\sum_{j=1}^{n} z_j \tag{2.3}$$

Whereas, the unidirectional pose repeatability is calculated using Formula 2.4, where x_j, y_j, and z_j are measured values.

$$\mathbf{uRP}_l = \pm 3 S_l \tag{2.4}$$

$$S_l = \sqrt{\frac{1}{n-1}\sum_{j=1}^{n}(l_j - \bar{l})^2} \tag{2.5}$$

$$\bar{l} = \frac{1}{n}\sum_{j=1}^{n} l_j \tag{2.6}$$

$$l_j = \sqrt{(x_j - \overline{x})^2 + (y_j - \overline{y})^2 + (z_j - \overline{z})^2} \tag{2.7}$$

2.5 Human-Robot Collaboration

Within the Human-Robot Collaboration (HRC) field, different levels of collaboration can be found, regarding the different aspects that needs to be taken into consideration. Wang et al. [45] discarded the idea of summarising the human-robot cell in terms of working time, workspace, aim and contact but expanded it into the following aspects: workspace, direct contact, work task, simultaneous process, and sequential process.

1. Workspace: It is important to assess whether humans and robots need to share the same workspace to carry out the designated task or if there is a need of separating the working area with physical or virtual fences or limits otherwise.
2. Contact: Inside this shared workspace, whether there is physical contact or not between human and robot needs to be considered.
3. Shared working task: indicates whether human and robot work alongside each other towards the same working objective.
4. Simultaneous process: indicates whether human and robot work at the same time either sharing workspace or not on the same or different tasks.
5. Sequential process: refers to the execution of tasks in a specific order (one action after the other) without overlapping operations between human and robot.

Having discussed the most important aspects that need to be considered when classifying the type of human-robot collaboration, one can now discuss the different human-robot collaboration levels and based on that establish which one fits best to this specific use-case.

- Cell: the robot is encaged ensuring there is no physical interaction with the human operator.
- Coexistence: humans and robots work alongside each other without the presence of any cage, although the workspace is not shared.
- Synchronised: humans and robots share the workspace. Only one interaction partner (either human or robot) is actively working in the workspace.
- Cooperation: shared workspace, in which both humans and robots have tasks to perform. There is no direct contact between the robot and human. They share the same resource but complete respective working tasks in a sequential order.
- Collaboration: human and robot share the same workspace and work simultaneously on the same product or component.

3
Methods

3.1 Data Acquisition

The data is obtained using the Intel RealSense Depth Camera D435, which is able to capture images in RGB, depth and infrared. At each position, which is represented in Fig. 3.5, the RGB, depth image and depth values of that position are stored. Each image size, RGB and depth (needs to be aligned) has a resolution of 640x480 pixels.

3.1.1 System's Overview

To classify the type of human-robot collaboration it is employed to take pictures from different viewpoints of the object of interest, there is a need to outline the key features of our system first and then classify the collaboration type accordingly. In our specific task, human and robot share workspace, however, there is only one interaction partner in the workspace at a time. This is because when the program is being executed in the robot, no one should enter the workspace so that no collisions are produced. The actual sequence or workflow of the system for the picture taking of one of each the connectors is the following:

1. Place the connector and fix it in place.
2. Run the program in the robot.
3. Wait until the program has finished.
4. Step inside the workspace to rotate the connector 90°.
5. Run the program in the robot for 2^{nd} time.
6. Replace the connector and repeat procedure.

Given that the task is to gather data from 360° angles, it is crucial to have the connector placed in a way that it is possible to take pictures from all angles without there being anything in the way of it. This is made possible using a rod, where the connector is placed on one end of the rod and the other end of it is attached to the wall, which allows flexibility in height and positioning. See Figure 4.2, for a visualisation of this.

Having good quality images is a crucial part of this thesis. Considering that the connectors that is being collected are somewhat small, it is important to get every single detail of it, to enable the model to distinguish between them. Another important aspect for the data collection, is the depth values of an image. Using a Realsense Depth Camera D435, this was possible to achieve, since it allows capturing RGB, depth and infrared images. The ideal range for the best quality for the RGB images

and the depth values are between 300mm and 3000mm. Given that, the distance between the camera and the connector was set to 310mm, for best performance. The collected images have a resolution of 640x480 pixels for both the RGB and the depth image. This is the optimal solution, ensuring that every pixel in the RGB image has a corresponding depth value.

To take into account for better reachability, and that the pictures has to be taken from a distance that is greater than 300mm, the robot that was chosen to be used was a Universal Robot 5 (UR5). The UR5 has a max reach of 850mm making it the optimal choice for capturing the images, since the diameter of the spherical environment that is used in is roughly 620mm. Furthermore, the robot was tilted 90°, which allows the robot to move in a circular motion around the connector, thus expanding on the reachibility. The camera was placed at the TCP, making the positioning of the camera relative to the TCP more accurate.

Taking care of the communication between the robot, computer and the camera is done using a Raspberry Pi 4 Model B, which will be described in Section 3.1.3.

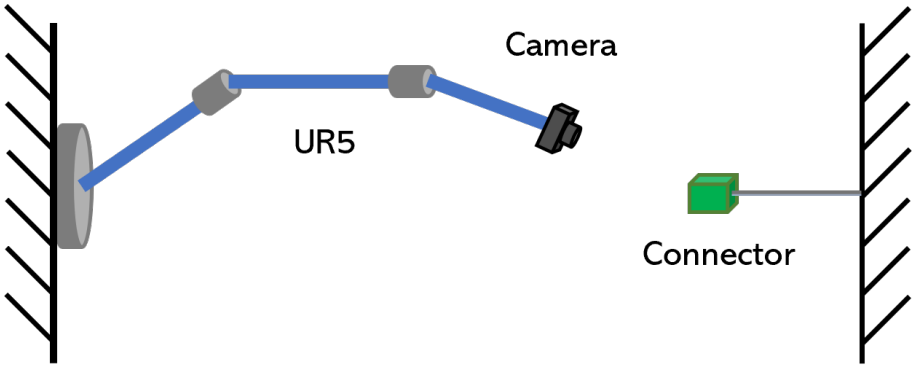
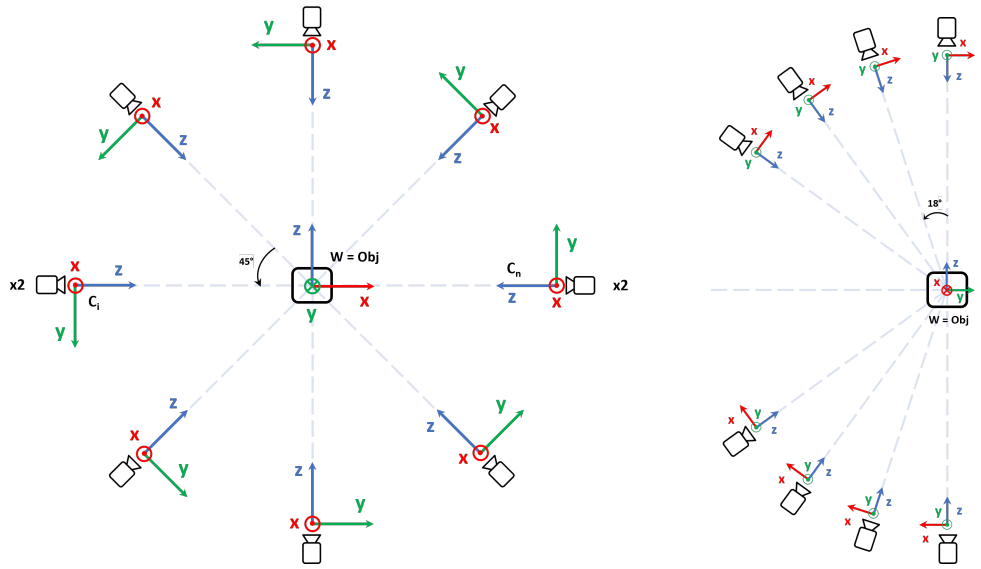


Figure 3.1: System's setup including all its components: UR5 robot, RealSense D435 camera, fixture and connector.

3.1.2 Robot Path-Planning

To collect data, a path was predefined for the UR5 to follow. The goal was to capture as many poses as possible, therefore, the path that was defined covers 40 waypoints. The path was defined in a way that it follows the outline of a hemisphere using MoveJ instructions which gives more movement flexibility to the robot and less joint limitations. The path is fixed to ensure consistency each time the pictures are collected for a new connector. In short, the robot's path consists of 4 arcs from above the object and 4 arcs from below it, with 5 waypoints in each arc. This ensures that the connector's main views (front, rear, top, bottom and sides) plus views that show the object rotated by 45°, are captured. The front and left side views of the setup can be seen in Fig. 3.2, where each waypoint can be observed, thus, the tracing the path.



(a) Front view **(b)** Left view

Figure 3.2: Pathplaning waypoints of the angles the pictures are taken from.

The radius of the semi-sphere is $0.31m$ as the depth camera has a depth capturing range of $0.3m < d < 3m$ where the depth is more accurate in the shorter the distance to the object is. The decision to work on a distance of $0.31m$ from the connector is to ensure the best accuracy from the depth values.

3.1.3 Robot-Camera Communication

A communication between the UR5 robot and the computer is crucial, since it is used to get a seamless dataset collection. The communication between the two goes through a Raspberry Pi. The robot is connected to the Raspberry Pi 4 Model B using two wires, one for input and one for output. The computer is then able to wirelessly communicate with the Raspberry Pi using internet protocol, where it is possible to read the input given to the Raspberry Pi through a specific pin and send out an output through another pin, thus enabling a two-way communication. The wireless connection between the computer and the Raspberry Pi is made possible using a python library called `gpiozero`, that allows a creation of a so called `PiGPIOFactory`, where one can specify the IP address of the Raspberry Pi.

The two-way communication is used to send a signal from the robot to the computer when it has reached a position, this notifies the computer to take a picture at the current position of the robot using the Intel RealSense Depth Camera D435. Once the picture has been taken and stored, a signal is sent from the computer to the robot which signals the robot to move to the next position. This process is repeated for all the positions. In short, the robot moves to a position, sends a signal `take_picture` to the computer and awaits a `move_robot` signal before moving to the next position. These signals are analog signals, but are interpreted as digital signals by the Raspberry Pi meaning the signals are either high (true) or low (false). See Algorithm 1,

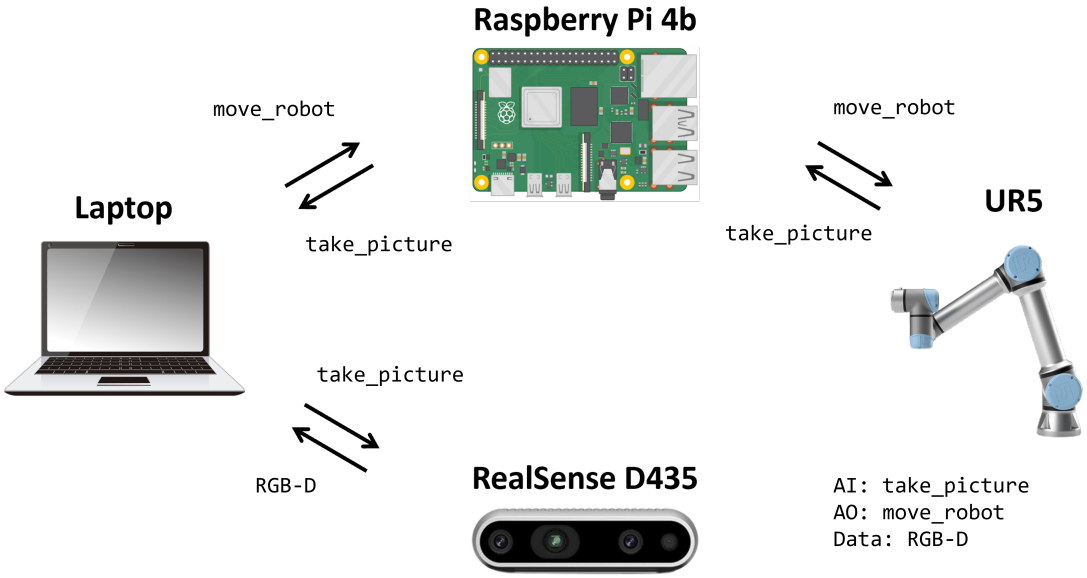


Figure 3.3: Robot-Camera Communication.

Algorithm 1 Communication algorithm

```
N ← 1
while N ≤ 80 do
    MoveJ waypoint N
    take_picture ← 3.3V
    while move_robot < 3.0 do          ▷ While loop represents Wait command
        pass
    end while
    take_picture ← 0.0V
    N ← N + 1
    if N == 40 then
        Flip connector                 ▷ The connector should be flipped 90°
    end if
end while
```

3.2 Data Annotation

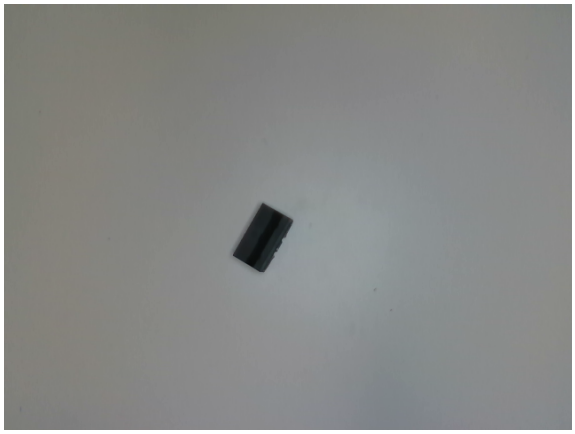
When working with object detection, annotation is a crucial part. Annotation tells the model where the object that it is supposed to train on is located. Without it, there is no way for the model to be able to train on a specific object if it does not know its whereabouts in an image. There are multiple tools for manual annotation, which is a tedious task when working with hundreds and thousand of images. In this thesis, the 2D annotation was done automatically.

Apart from the whereabouts of the object in 2D, it is also important to know its 6D coordinates, namely the x, y, z, roll, pitch, and yaw. This enables the model to

recognise its positioning in 3D and its rotation in relation to the camera.

3.2.1 2D Bounding Box

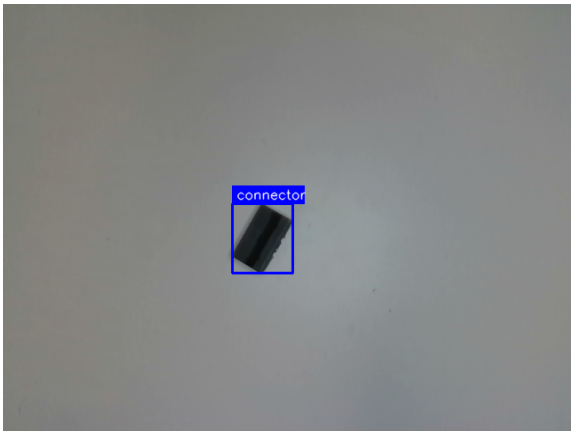
2D labeling is usually done by drawing a 2D bounding box around the object in the image, which tells the model that the object is within this bounding box. The annotation process comprises two stages. Initially, the background is filtered out using the depth camera. Depth values outside the range of $[10, 350]$ are excluded. The upper limit is set to 350 to account for the distance of $\sim$ 310 between the camera and the connector, ensuring that objects beyond this point are classified as background. This leaves us with the connector and the white rod. In the next step, we need to remove the white rod from the image. We do this by filtering out any white color present in the image. As a result, we are left with only the connector and a transparent background. This in turn, makes it possible to replace the transparent background with any other background depending on the use case of the model's object detection. Allowing it to be trained to recognise different environment thus enhancing the performance of the model. Using the transparent images, automating the annotation is made possible, by obtaining the contour of the connector and excluding the transparent background. Using the $x_{\min}$, $y_{\min}$, $x_{\max}$, and $y_{\max}$, of the contour, two coordinates for the bounding box is achieved, namely ($x_{\min}$, $y_{\min}$) and ($x_{\max}$, $y_{\max}$). These two coordinates, represent two opposite corners of the bounding box, and using them the bounding box can be drawn, as can be seen in Figure 3.4



(a) Original RGB image

(b) Filter out white pixels and depth that exceeds a given threshold

(c) Extracting the contour from the transparent image



(d) Drawing bounding box from the contour on the original image

Figure 3.4: The four steps of the automatic annotation

3.2.2 Pose Annotation

In this benchmark, during data collection, the electrical connector is fixed at all times, whereas, the camera moves around taking pictures by means of the UR5 robot. Visual Components was used to carry out the offline programming for the path planning, the simulation tool allows to relate object coordinate frames by using homogeneous transformation matrices which include the rotation matrix and the translation vector. In order to annotate all the camera poses in a text file the path statements are iterated over. These contain all the positions where the camera will stop to take a picture. Then, the homogeneous transformation matrices ${}^{O}T_{C_i}$ are read for N positions, by first storing the translation vector and afterwards the rotation vector in a text file. In Figure 3.5 the object and camera coordinate frames are depicted and those transformations are used to retrieve the pose of the camera with respect to the object.

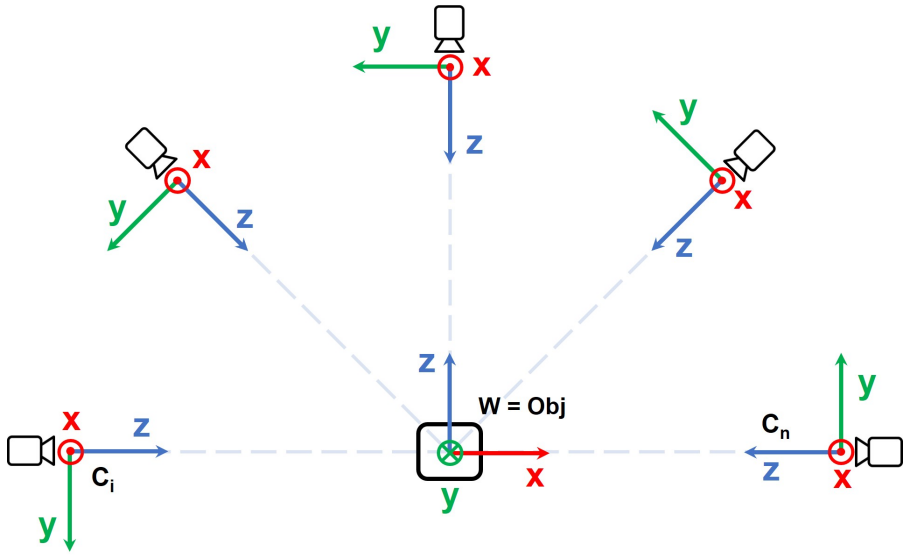


Figure 3.5: Object and camera coordinate frames for pose annotation

After the camera poses have been stored, it is essential to calculate the pose of the object with respect to the camera ${}^{C}T_{O}$ for all camera positions. This is done by calculating the inverse of the homogeneous transformation matrix ${}^{O}T_{C}$:

$$ {}^{O}T_{C} = {}^{C}\,T_{O}^{-1} \tag{3.1} $$

To recover the homogeneous transformation matrices from the text file, the rotation vector roll (θ), pitch (φ) and yaw (ψ) needs to be transformed into a rotation matrix R. This is done by computing the rotation matrices Rx, Ry and Rz and multiply them the correct order:

$$ R_x = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta) & -\sin(\theta) \\ 0 & \sin(\theta) & \cos(\theta) \end{pmatrix} \tag{3.2} $$

$$ R_y = \begin{pmatrix} \cos(\varphi) & 0 & \sin(\varphi) \\ 0 & 1 & 0 \\ -\sin(\varphi) & 0 & \cos(\varphi) \end{pmatrix} \tag{3.3} $$

$$ R_z = \begin{pmatrix} \cos(\psi) & -\sin(\psi) & 0 \\ \sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{3.4} $$

$$ R = R_z \cdot R_y \cdot R_x \tag{3.5} $$

Once all homogeneous transformation matrices for the object's poses have been calculated they are transformed again into a $(T_x, T_y, T_z, \theta, \varphi, \psi)$ vector and the vectores are stored in a separate text file.

3.3 Data Preprocessing

To achieve a good performance while carrying out object detection, it is important to pre-process the data accordingly to ensure data properties like image size and feature scales are consistent. In our benchmark for dataset collection of automotive electric connectors all images are resized to a common standard of 640x480 and normalization is performed in a pre-training stage to scale and center the data to a common scale of [0, 255] in RGB. The goal of normalization is to stabilize the training process and accelerates convergence, which results in shorter training times.

3.4 Data Augmentation

In the presented benchmark for dataset collection of automotive electric connectors data augmentation techniques were applied in order to artificially increase the size and diversity of the training dataset [46]. By exposing a model to a wider variety of data during training, the model's generalization ability is improved by making it more robust to different variations in the data, however, overusing data augmentation can lead to decreased performance on the test set [46]. Furthermore, data augmentation is an effective approach to reduce overfitting, which occurs when a model learns the training data too well and is unable to generalize the new data and therefore end misspredicting unseen objects. As a result of robustness and reduced overfitting, data augmentation leads to improved accuracy on both the training and test sets [46].

The following data augmentation techniques were applied to achieve increased performance when training on object detection pipelines.

3.4.1 Apply blur

When deciding which blur to use, it's important to consider the different types of blur available, what they signify, and their intended applications. First, Gaussian Blur smooths the image by using a Gaussian distribution function and emphasizing pixels closer to the filter's center [47]. Next, motion blur simulates the blur effect resulting from either camera or object movement during picture-taking [48]. Furthermore, bilateral blur preserves the sharpness of image edges while smoothing the interior regions, avoiding the loss of important features in the image [49]. Median blur replaces each pixel's value with the median value of the neighboring pixels, helping to reduce random noise without affecting the sharpness of edges [50].

Among the various types of blur discussed, incorporating motion blur into the dataset augmentation process yields the most effective results for enhancing model robustness during training. This is particularly relevant in the context of object detection, where cameras are utilized to identify and classify objects, and slight blurring of frames may occur due to camera movement.

3.4.2 Vary brightness and add noise

In line with our efforts to simulate real-world conditions in car environments, where lighting fluctuations and shadow occurrences are common, the dataset is enhanced by adjusting image brightness within a specified range [-120, 120]. This augmentation strategy aims to mirror the variability encountered in practical scenarios. Concurrently, the findings of José A Rodríguez-Rodríguez et al. [51], underscore the importance of accounting for such variations in illumination. Their study focuses in the impact of noise and brightness on object detection models, YOLOv5, v8 and Faster-RCNN. They conclude that while excessive noise or drastic reductions in brightness typically hinder detector performance, moderate levels of noise or slight dimming of illumination may, in fact, aid in detecting objects that would otherwise go unnoticed in unaltered images. This insight is invaluable for designing robust object detection systems, especially those operating in environments characterized by noise and low-light conditions.

To augment the data noise-wise, Gaussian and salt and pepper noise was added to the images. This forces the model to learn features that are robust to small variations in the input. As a result,the model would be more likely to detect and classify the object even if the image image deviates slightly from the training data.

3.5 Dataset Evaluation

As discussed in section 2.3.2, the previous work carried out by Hao Wang et al. [1] investigated using two different deep-learning-based detectors to evaluate the dataset. These were YOLOv5 and Faster R-CNN. In this thesis work, Hao Wang et al.'s dataset was trained using an improved version of YOLOv5, YOLOv8. The obtained dataset was evaluated using YOLOv8 and was later compared with both benchmark dataset evaluation results.

4

Results

The results chapter focuses on presenting the dataset obtained for the developed methodology as well as an evaluation of the collected dataset using a deep-learning based object detector (YOLOv8). The presentation follows the same structure as the one described in the Methodology chapter which is: data acquisition, data annotation, data pre-processing and augmentation, and finally the dataset evaluation. In this results section a comparison is included between the previous work carried out in connector dataset collection [1] but trained on a YOLOv8 and the dataset collected with the proposed benchmark to draw fruitful conclusions.

4.1 Data Acquisition Results

The collected datasets from the presented framework contain 8 different classes of connectors used in the wire harness final assembly in the automotive industry. Figure 4.1 shows the connectors included in the datasets. With the current data collection method, 80 images per connector are collected in an average of 3 minutes.

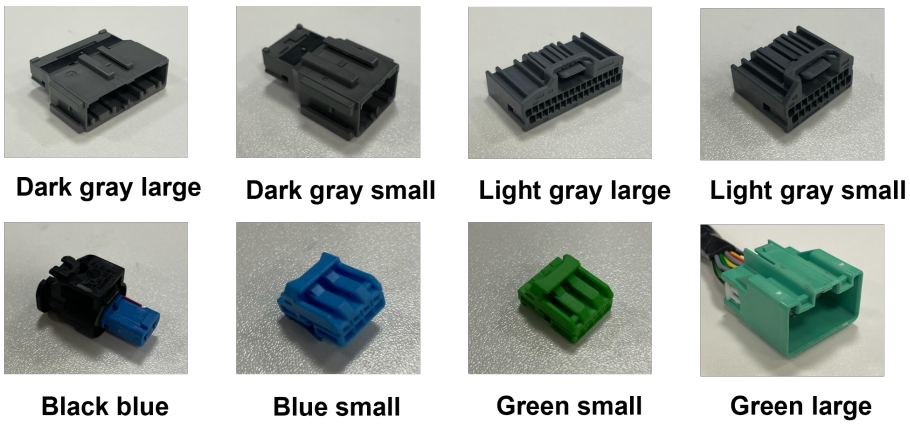


Figure 4.1: The eight connectors that were collected and evaluated

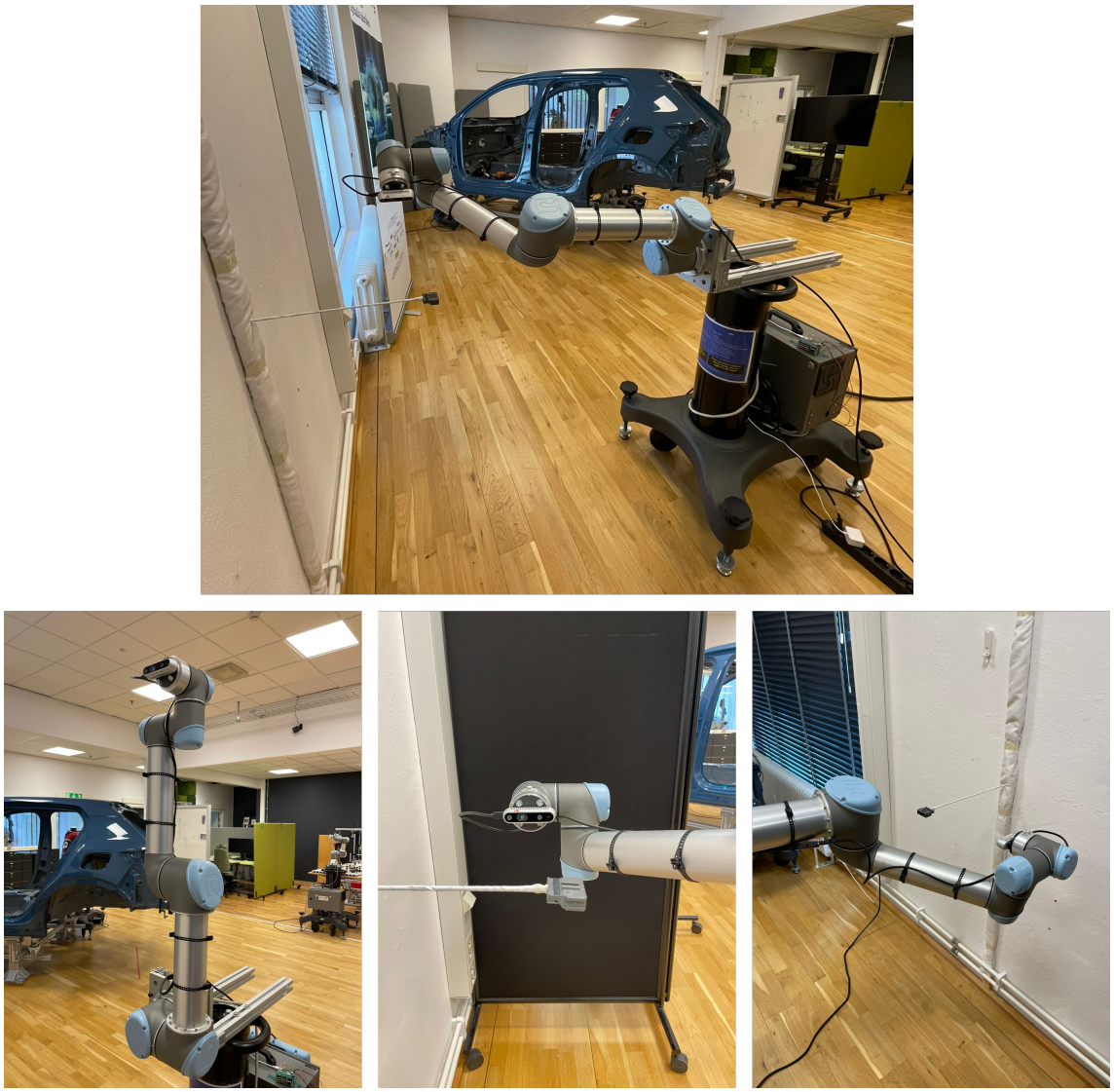
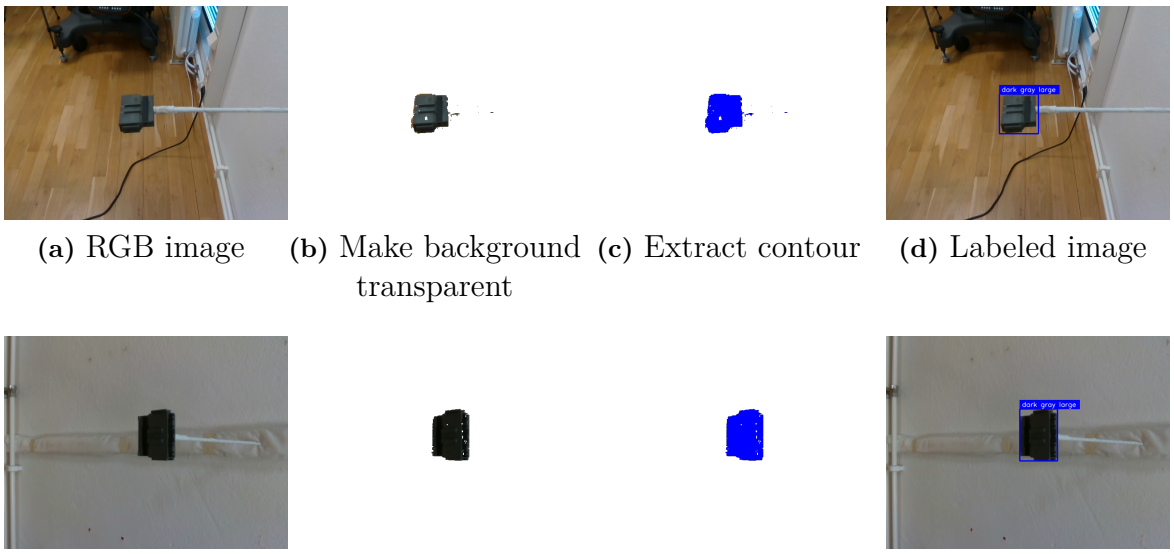


Figure 4.2: The dataset collection setup.

4.2 Data Annotation Results

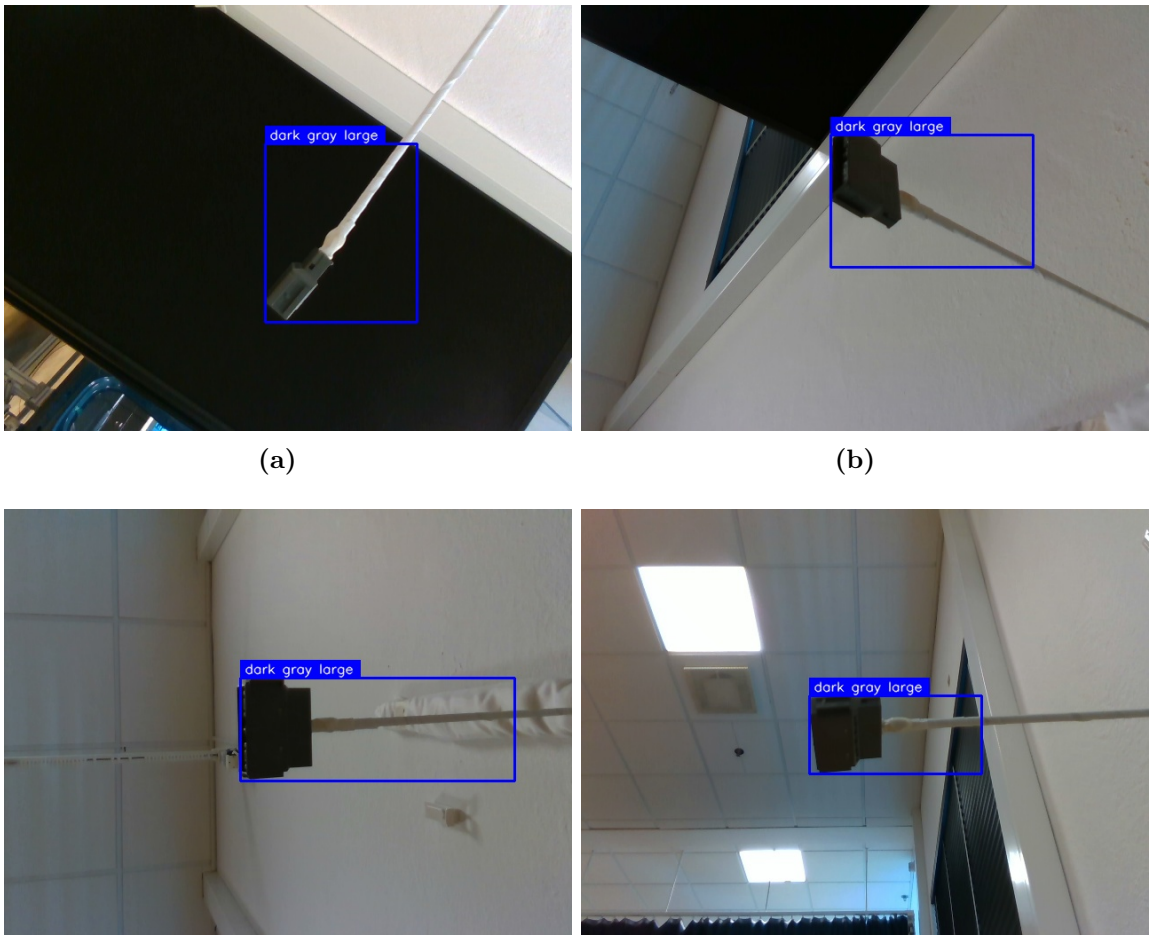


(a) RGB image **(b)** Make background transparent **(c)** Extract contour **(d)** Labeled image

(e) RGB image **(f)** Make background transparent **(g)** Extract contour **(h)** Labeled image

Figure 4.3: Annotation steps for two different frames of the same connector

Even though many images were annotated correctly (Figure 4.3) with the bounding box delimiting the position of the connector in the image, some of them were incorrectly annotated too as a result of the uncontrollable lighting conditions in the setup. An example of incorrectly annotated images included in the dataset for the automatic annotation approach can be seen in Fig. 4.4



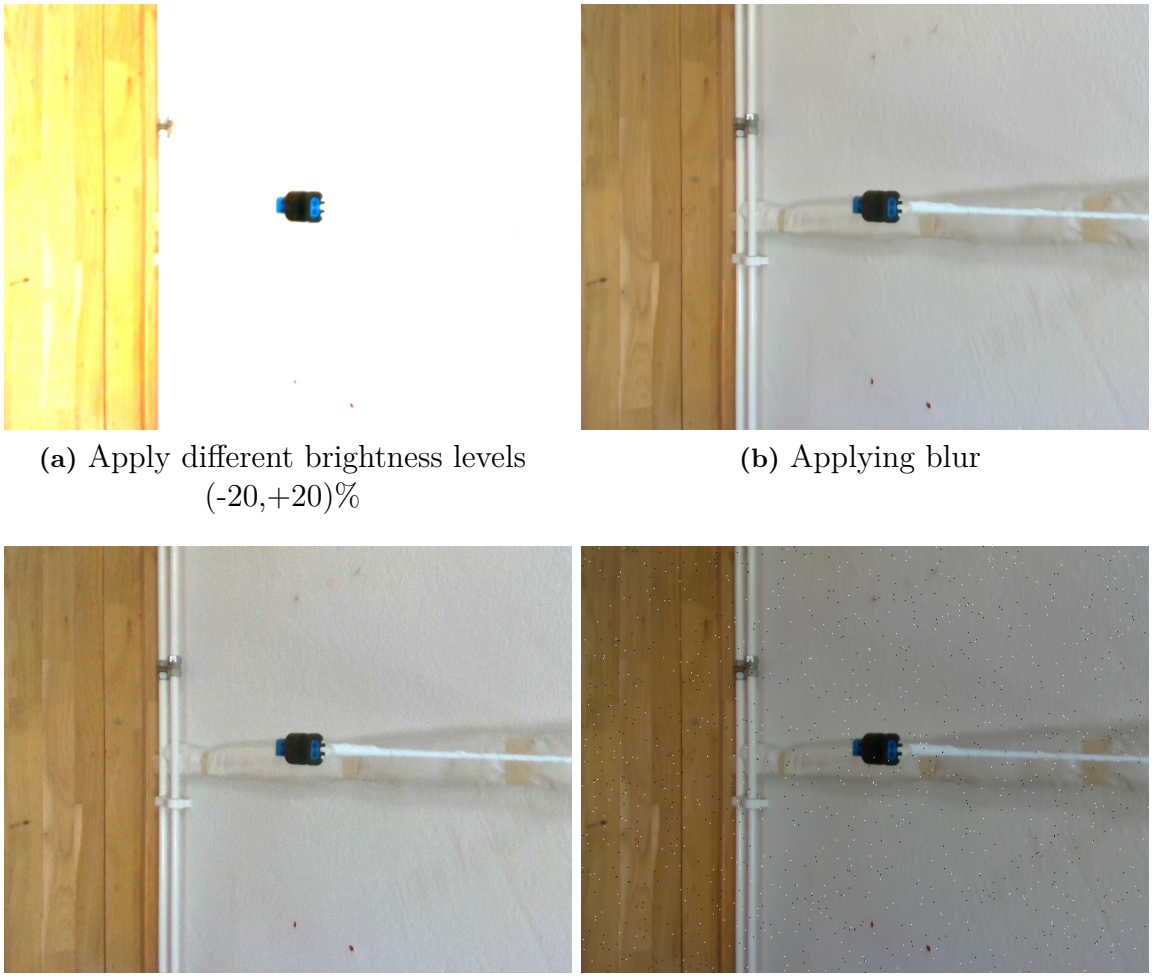
(a) (b)

(c) (d)

Figure 4.4: Four examples of where the lighting caused inaccuracies in the annotations

4.3 Data Preprocessing and Augmentation Results

The data augmentation techniques used to inflate the dataset and bring variety are shown in Figure 4.5



(a) Apply different brightness levels (-20,+20)%

(b) Applying blur

(c) Applying Gaussian noise

(d) Adding salt and pepper noise

Figure 4.5: The four data preprocessing and augmentation steps

4.4 Dataset Statistics

In this section, statistics for both, the manual and automatic annotated datasets are presented. The total number of annotated images is 1541 for the manual dataset against 3200 annotated images for the automatic dataset. The reason behind the number difference of object annotated instances is that the manual dataset was generated and augmented using Roboflow's online tool, whereas the automatic dataset was augmented using our own data augmentations. The data is primarily divided into three main subsets: training data (Train), validation data (Val), and test data (Test), with a ratio of 87%/8%/4% for the manual dataset and a split of 70%/20%/10%. The number of object instances is shown in tables 4.1 and 4.2:

Table 4.1: The numbers of annotated object instances in the manual dataset.

Class	Train	Val	Test	Total
Black Blue	171	16	8	195
Light Gray Small	168	16	8	192
Light Gray Large	168	16	8	192
Dark Gray Large	168	16	8	192
Dark Gray Small	168	16	8	192
Green Large	168	16	9	193
Green Small	168	16	9	193
Light Blue	168	16	8	192

Table 4.2: The numbers of annotated object instances in the automatic dataset.

Class	Train	Val	Test	Total
Black Blue	280	80	40	400
Light Gray Small	280	80	40	400
Light Gray Large	280	80	40	400
Dark Gray Large	280	80	40	400
Dark Gray Small	280	80	40	400
Green Large	280	80	40	400
Green Small	280	80	40	400
Light Blue	280	80	40	400

4.5 Data Evaluation Results

4.5.1 Precision

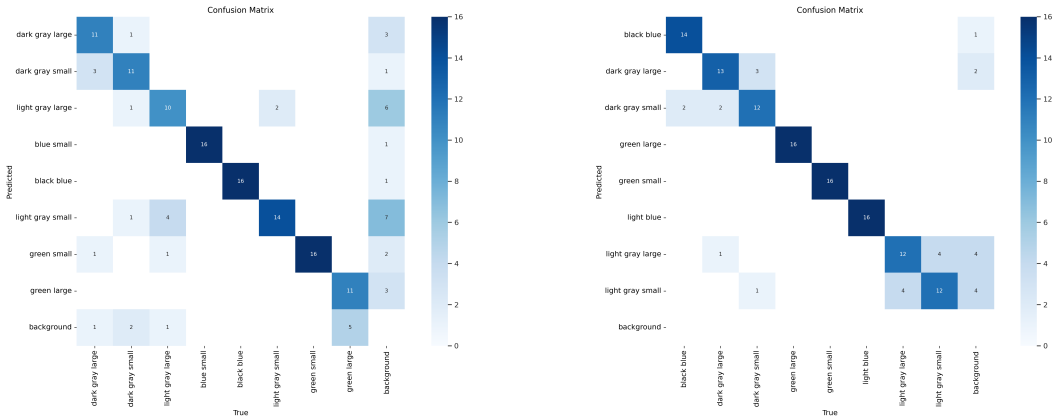
To evaluate the performance of the collected datasets they were trained using YOLOv8 object detection model. Below, the table details the mean average precision (mAP) for the manual collected dataset by Wang et al. [1] trained in both YOLOv5 and YOLOv8 compared to the work carried out in this thesis: manual vs automatic annotation. Interestingly, the results indicate that the manual annotation mAP outperforms the automatic approach. The reasons behind this outcome are discussed in more detail in the Discussions section (Chapter 5).

Table 4.3: The mean average precision (%) of YOLOv5 in [1], YOLOv8 in [1] and YOLOv8 with the benchmark dataset presented in this thesis (automatic and manual annotated datasets)

	mAP_{50}	mAP_{50-95}
YOLOv5 [1]	88.5	82.1
YOLOv8 [1]	87.1	84.4
YOLOv8 - Automatic annotation	74.4	54.9
YOLOv8 - Manual annotation	**93.5**	**87.6**

4.5.2 Confusion Matrices

Below, the confusion matrices for the evaluation of the collected datasets, manually and automatic annotated are depicted. The main diagonal elements represent the correctly predicted classes, whereas the off-diagonal elements represent the misclassifications. These matrices help us visualize which classes are being misclassified.

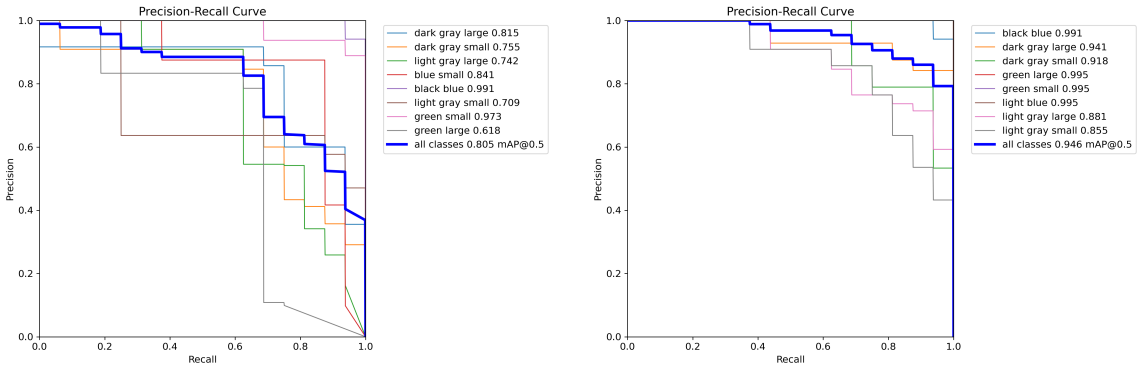


(a) Automatic annotation **(b)** Manual automation

Figure 4.6: Confusion matrices for automatic (4.6a) and manual (4.6b) annotation.

4.5.3 Precision-Recall Curves

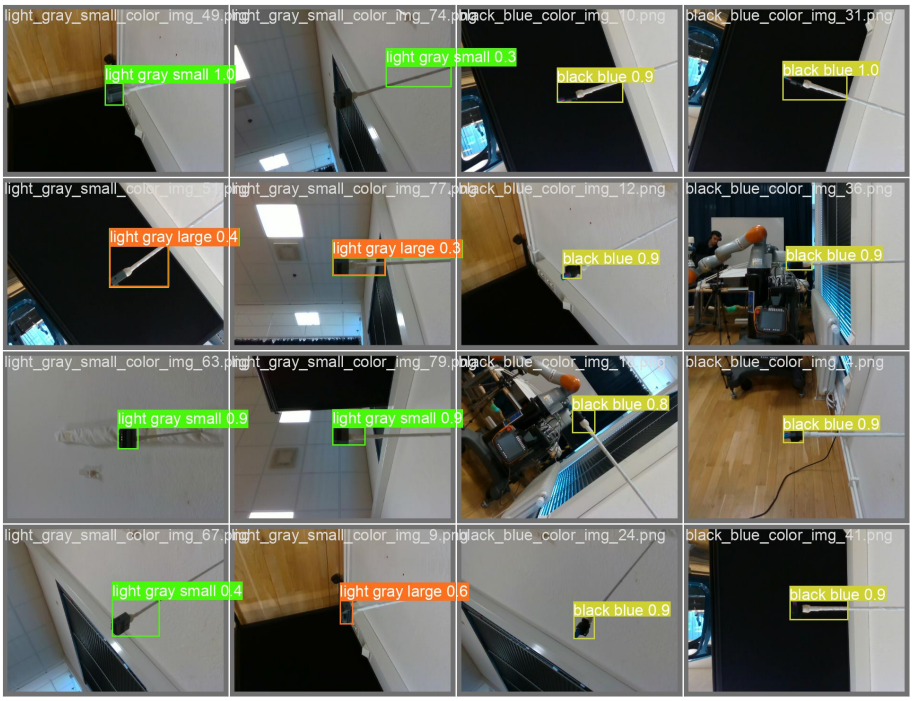
To further evaluate the performance of the object detection models, Precision-Recall (P-R) curves were generated. P-R curves are valuable tools for understanding the trade-off between precision (the accuracy of the positive predictions) and recall (the ability to capture all positive instances) at different threshold settings. They provide a comprehensive view of the model's performance.



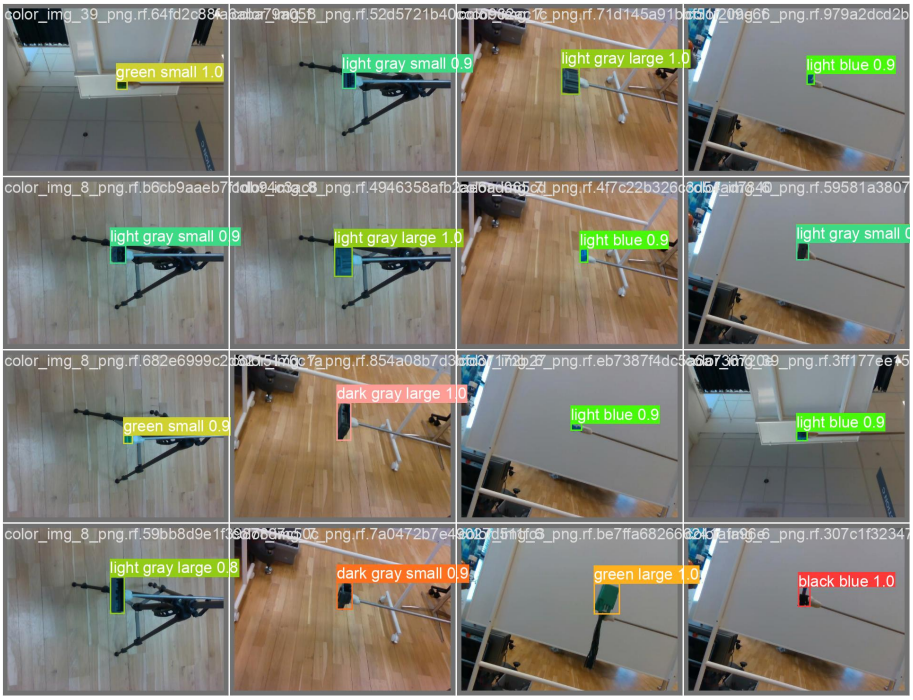
(a) Automatic annotation **(b)** Manual automation

Figure 4.7: Precision-Recall curves for automatic (4.7a) and manual (4.7b) annotation.

4.5.4 Predictions



(a) Automatic annotation



(b) Manual annotation

Figure 4.8: Predictions for two validation batches for automatic (4.8a) and manual (4.8b) annotation.

5
Discussion

5.1 Results

The results obtained in this thesis 4.3, also demonstrated an increase in the accuracy, with the model achieving 93.5% accuracy compared to the 87.1% obtained by Wang et al. This improvement can be partly attributed to the smaller number of classes, as training with fewer classes generally yields higher accuracy. Nonetheless, the enhanced accuracy and efficiency highlights the benefits of the automated approach. It is important to point out the results obtained for the automated annotated dataset, with an mAP of 74.4%. This accuracy is not high as expected since during the automatic annotation process some connectors are mislabeled as a result of the inconsistent and uncontrollable lighting conditions. However, the setup could be improved in further works by looking for a better illuminated setup.

From 4.6, if both confusion matrices are compared, it can be seen that the manual annotation method correctly classifies more classes than the automatic tool, as it has more main diagonal elements than the automatic tool's confusion matrix. However, from both matrices, it can be observed that there is some confusion between the gray connectors. Misclassifications often occur between dark gray small and large classes, as well as between light gray small and large classes.

As shown in Figure 4.7, the model achieved both high recall and high precision, indicating an optimal trade-off. In contrast, when recall approaches 1.0, precision tends to be low, nearing 0.0, and vice versa. This behavior is expected, as it is difficult to maximize both metrics simultaneously. Given that, precision focuses on identifying all relevant objects, while recall aims to detect all ground-truth bounding boxes.

5.2 Automatic Annotation tool

The initial attempt in this thesis to label the dataset with bounding boxes was based on a setup with a white background. However, it proved unsuccessful due to issues with lighting and shadows, which were a major challenge encountered in the automated dataset collection because of the inability to control lighting conditions consistently. Consequently, an alternative method was explored. It involved using depth values to identify object pixels within the context and applying white filtering to remove the white rod used to hold the connectors. This method, although more

complex, provided a viable solution to initial challenges encountered and highlighted areas for further improvement in future research. Still, even using this alternative automatic annotation tool, light and shadows had a big impact on the dataset evaluation results. The robot's movement during image capture invariably introduced shadows within the camera frame, adversely affecting the dataset's quality. These varying lighting conditions and shadows significantly contributed to the low mean Average Precision (mAP) results. While the automated system facilitated faster data collection, the fluctuations in lighting conditions hindered the accuracy of this thesis work object detection model, resulting in performance below initial expectations.

5.3 Previous work

Comparing this thesis to previous research work conducted by Wang et al. [1], several key differences and improvements were identified. Wang et al. collected a dataset comprising of 20 different connectors, whereas this thesis focused on 8 connectors. Despite the smaller number of classes in the dataset, the process was enhanced by automating both collection and labelling. The automated system reduced the time required to collect and annotate images, achieving a total of 80 images per connector in approximately 3 minutes. In contrast, Wang et al. manually collected 15 images per connector, indicating a significant improvement in efficiency through automation. Furthermore, these manually collected images were manually annotated, which entails that the solution proposed by Wang et al. is not efficient in regards to scalability.

5.4 The automated system

The automated system presented in this thesis offers several advantages over manual methods, particularly concerning speed and scalability. the decreased time and labor demands for dataset collection enable the addition of new connectors to the dataset with minimal effort. This adaptability holds significant value for applications necessitating frequent dataset updates or expansions. Moreover, through the automation of the labeling process, human error and variability were minimized, thereby enhancing the consistency and reliability of the data.

6
Future Work

There is still room for work to be done in the field of detection of electrical connectors, as well as, in dataset benchmarking, precisely in automatic data annotation. Existing works annotate and estimate the 6D pose of an object utilizing a CAD model or synthetic data. Using 3D models allows them to have a 2D-3D correspondence which is highly valuable when it comes to annotate and estimate accurately the position of an object in three-dimensional space. A natural continuation of this thesis would be to make use of both RGB-D and 3D synthetic data, to enable accurate pose annotation.

Furthermore, in this thesis work we have only carried out data collection for a fixed distance of $310mm$ from camera lens to object. This may be limited when evaluating the performance of YOLOv5 in shorter or longer distances. Therefore, future research may consider to increase the data collection for different distances from the object in context.

Apart from object detection, it is valuable to consider the performance of using object segmentation when annotating and estimating the 6D pose of an object. Object segmentation goes a step further than detection by identifying the exact pixels that belong to each detected object, this is why for more accurate and detailed pose estimation, object segmentation often provides better results due to its pixel-level precision.

7
Conclusion

This thesis work investigates an approach using a robotic set-up of carrying out data collection in an automated fashion. This benchmark includes the data acquisition utilizing a UR5 robot plus an Intel RealSense D435 camera; the robot-camera communication using a Raspberry Pi 4b as a bridge between robot and camera and an automatic labeling tool. The resultant dataset is evaluated using YOLOv8, a deep-learning based model. The collected dataset comprises of 8 different connectors that are commonly used in automotive wire harnesses.

During the development of this master thesis, answers have been found for the research questions proposed in the beginning. Below, the research questions are presented together with the findings and their answers:

RQ1. What are the current state-of-the-art methodologies for benchmark dataset collection?
The current state-of-the-art methodologies for benchmark dataset collection and 6D pose annotation still rely on manual annotation for 2D datasets like ImageNet, Objects365 and COCO, while efforts are being made to automate data collection using robotic systems. For 3D object detection, methods include depth images, point-clouds and augmented reality with semi-automatic tools like SALT and ARS being developed.

RQ2. How can we automate the benchmark dataset collection that includes both 2D and 3D data, and evaluate the 2D dataset?
The development of an automated system was successfully achieved by integrating a UR5 robot with an Intel RealSense D435 camera. This setup, combined with a Raspberry Pi 4b for communication, enabled automated data collection and labeling. The automation process reduced human error and variability, thus enhancing the consistency and reliability of the data. The automated system's adaptability allows for the easy addition of new connectors to the dataset with minimal effort, demonstrating significant value for applications requiring frequent dataset updates or expansions.

RQ3. What are the advantages of employing a semi-automated data collection methodology compared to traditional manual data collection processes?
The automated system demonstrated several advantages over manual methods. Notably, it offered increased speed and scalability, reducing time and labor de-

mands for dataset collection. However, the evaluation results indicated that while the manually annotated dataset achieved higher mean average precision (mAP) than the automated approach, future improvements could focus on controlling consistent lighting conditions to enhance labeling accuracy during data collection.

Bibliography

[1] Hao Wang and Björn Johansson. 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, 2023.

[2] Fayez Alanazi. Electric vehicles: Benefits, challenges, and potential solutions for widespread adaptation. *Applied Sciences*, 13(10), 2023.

[3] Gabriel E Navas-Reascos, David Romero, Ciro A Rodriguez, Federico Guedea, and Johan Stahre. Wire harness assembly process supported by a collaborative robot: A case study focus on ergonomics. *Robotics*, 11(6):131, 2022.

[4] Omkar Salunkhe. *Designing Collaborative Robot Workstations for Human Centred Automation in Final Assembly: A Task Allocation Approach.* Chalmers University of Technology, 2023.

[5] Baiqing Sun, Fei Chen, Hironobu Sasaki, and Toshio Fukuda. Robotic wiring harness assembly system for fault-tolerant electric connectors mating. In *2010 International Symposium on Micro-NanoMechatronics and Human Science*, pages 202–205, 2010.

[6] Pei Di, Jian Huang, Fei Chen, Hironobu Sasaki, and Toshio Fukuda. Hybrid vision-force guided fault tolerant robotic assembly for electric connectors. In *2009 International Symposium on Micro-NanoMechatronics and Human Science*, pages 86–91, 2009.

[7] Pei Di, Fei Chen, Hironobu Sasaki, Jian Huang, Toshio Fukuda, and Takayuki Matsuno. Vision-force guided monitoring for mating connectors in wiring harness assembly systems. volume 24, pages 666–676. Fuji Technology Press, 2012.

[8] Hee-Chan Song, Young-Loul Kim, Dong-Hyeong Lee, and Jae-Bok Song. Electric connector assembly based on vision and impedance control using cable connector-feeding system. *Journal of Mechanical Science and Technology*, 31:5997–6003, 12 2017.

[9] Tomoki Tamada, Yuji Yamakawa, Taku Senoo, and Masatoshi Ishikawa. High-speed manipulation of cable connector using a high-speed robot hand. In *2013 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 1598–1604, 2013.

[10] Hang Zhou, Shunchong Li, Qi Lu, and Jinwu Qian. A practical solution to deformable linear object manipulation: A case study on cable harness connection. In *2020 5th International Conference on Advanced Robotics and Mechatronics (ICARM)*, pages 329–333, 2020.

[11] Francisco Yumbla, Meseret Abeyabas, Tuan Luong, June-Sup Yi, and Hyungpil Moon. Preliminary connector recognition system based on image processing for

wire harness assembly tasks. In *2020 20th International Conference on Control, Automation and Systems (ICCAS)*, pages 1146–1150, 2020.

[12] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

[13] Shuai Shao, Zeming Li, Tianyuan Zhang, Chao Peng, Gang Yu, Xiangyu Zhang, Jing Li, and Jian Sun. Objects365: A large-scale, high-quality dataset for object detection. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8429–8438, 2019.

[14] Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John M. Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88:303–338, 2010.

[15] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollar, and Larry Zitnick. Microsoft coco: Common objects in context. In *ECCV*. European Conference on Computer Vision, September 2014.

[16] Piotr Dollar, Christian Wojek, Bernt Schiele, and Pietro Perona. Pedestrian detection: A benchmark. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 304–311, 2009.

[17] Andreas Danzer, Thomas Griebel, Martin Bach, and Klaus Dietmayer. 2d car detection in radar data with pointnets. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pages 61–66. IEEE, 2019.

[18] Constantine P Papageorgiou, Michael Oren, and Tomaso Poggio. A general framework for object detection. In *Sixth international conference on computer vision (IEEE Cat. No. 98CH36271)*, pages 555–562. IEEE, 1998.

[19] Valery Ilin, Ivan Kalinov, Pavel Karpyshev, and Dzmitry Tsetserukou. Deepscanner: a robotic system for automated 2d object dataset collection with annotations. In *2021 26th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, pages 01–08. IEEE, 2021.

[20] Will Knight. The foundations of ai are riddled with errors. *Wired*, 2021.

[21] Rakesh Shrestha, Siqi Hu, Minghao Gou, Ziyuan Liu, and Ping Tan. A real world dataset for multi-view 3d reconstruction. In *European Conference on Computer Vision*, pages 56–73. Springer, 2022.

[22] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5828–5839, 2017.

[23] A Geiger and Urtasun R Lenzp. Arewereadyfor autonomousdriving, 2012.

[24] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1912–1920, 2015.

[25] Colin Rennie, Rahul Shome, Kostas E Bekris, and Alberto F De Souza. A dataset for improved rgbd-based object detection and pose estimation for warehouse pick-and-place. *IEEE Robotics and Automation Letters*, 1(2):1179–1185, 2016.

[26] Arjun Singh, James Sha, Karthik S Narayan, Tudor Achim, and Pieter Abbeel. Bigbird: A large-scale 3d database of object instances. In *2014 IEEE international conference on robotics and automation (ICRA)*, pages 509–516. IEEE, 2014.

[27] Mikaela Angelina Uy, Quang-Hieu Pham, Binh-Son Hua, Thanh Nguyen, and Sai-Kit Yeung. Revisiting point cloud classification: A new benchmark dataset and classification model on real-world data. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1588–1597, 2019.

[28] Alexander Kasper, Zhixing Xue, and Rüdiger Dillmann. The kit object models database: An object model database for object recognition, localization and manipulation in service robotics. *The International Journal of Robotics Research*, 31(8):927–934, 2012.

[29] Andy Zeng, Kuan-Ting Yu, Shuran Song, Daniel Suo, Ed Walker, Alberto Rodriguez, and Jianxiong Xiao. Multi-view self-supervised deep learning for 6d pose estimation in the amazon picking challenge. In *2017 IEEE international conference on robotics and automation (ICRA)*, pages 1386–1383. IEEE, 2017.

[30] ZhiHong Jiang, JinHong Chen, YaMan Jing, Xiao Huang, and Hui Li. 6d pose annotation and pose estimation method for weak-corner objects under low-light conditions. *SCIENCE China Technological Sciences*, 66(3):630–640, 2023.

[31] Hao Zhang, Pinxin Long, Dandan Zhou, Zhongfeng Qian, Zheng Wang, Weiwei Wan, Dinesh Manocha, Chonhyon Park, Tommy Hu, Chao Cao, et al. Dorapicker: An autonomous picking system for general objects. In *2016 IEEE International Conference on Automation Science and Engineering (CASE)*, pages 721–726. IEEE, 2016.

[32] Mostafa Ibrahim, Qiong Liu, Rizwan Khan, Jingyu Yang, Ehsan Adeli, and You Yang. Depth map artifacts reduction: a review. *IET Image Processing*, 14, 09 2020.

[33] Radu Bogdan Rusu and Steve Cousins. 3d is here: Point cloud library (pcl). In *2011 IEEE international conference on robotics and automation*, pages 1–4. IEEE, 2011.

[34] Yilin Wang and Jiayi Ye. An overview of 3d object detection. *arXiv preprint arXiv:2010.15614*, 2020.

[35] Dennis Stumpf, Stephan Krauß, Gerd Reis, Oliver Wasenmüller, and Didier Stricker. SALT: A semi-automatic labeling tool for RGB-D video sequences. *CoRR*, abs/2102.10820, 2021.

[36] Daniele De Gregorio, Alessio Tonioni, Gianluca Palli, and Luigi Di Stefano. Semiautomatic labeling for deep learning in robotics. *IEEE Transactions on Automation Science and Engineering*, 17(2):611–620, 2019.

[37] Stefan Hinterstoisser, Vincent Lepetit, Slobodan Ilic, Stefan Holzer, Gary Bradski, Kurt Konolige, and Nassir Navab. Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes. In *Computer Vision–ACCV 2012: 11th Asian Conference on Computer Vision, Daejeon, Korea, November 5-9, 2012, Revised Selected Papers, Part I 11*, pages 548–562. Springer, 2013.

[38] Yu Xiang, Tanner Schmidt, Venkatraman Narayanan, and Dieter Fox. Posecnn: A convolutional neural network for 6d object pose estimation in cluttered

scenes. *arXiv preprint arXiv:1711.00199*, 2017.

[39] Yuan Liu, Yilin Wen, Sida Peng, Cheng Lin, Xiaoxiao Long, Taku Komura, and Wenping Wang. Gen6d: Generalizable model-free 6-dof object pose estimation from rgb images. In *European Conference on Computer Vision*, pages 298–315. Springer, 2022.

[40] Berk Calli, Arjun Singh, Aaron Walsman, Siddhartha Srinivasa, Pieter Abbeel, and Aaron M. Dollar. The ycb object and model set: Towards common benchmarks for manipulation research. In *2015 International Conference on Advanced Robotics (ICAR)*, pages 510–517, 2015.

[41] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection, 2017.

[42] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection, 2018.

[43] Glenn Jocher, Ayush Chaurasia, and Jing Qiu. Ultralytics YOLO, January 2023.

[44] Martin Pollák, Marek Kočiško, Dušan Paulišin, and Petr Baron. Measurement of unidirectional pose accuracy and repeatability of the collaborative robot ur5. *Advances in Mechanical Engineering*, 12:168781402097289, 12 2020.

[45] Xi Vincent Wang, A Seira, and Lihui Wang. Classification, personalised safety framework and strategy for human-robot collaboration. In *Proceedings of International Conference on Computers & Industrial Engineering, CIE*, 2018.

[46] Parvinder Kaur, Baljit Singh Khehra, and Er Bhupinder Singh Mavi. Data augmentation for object detection: A review. In *2021 IEEE International Midwest Symposium on Circuits and Systems (MWSCAS)*, pages 537–543. IEEE, 2021.

[47] Jan Flusser, Sajad Farokhi, Cyril Höschl, Tomáš Suk, Barbara Zitova, and Matteo Pedone. Recognition of images degraded by gaussian blur. *IEEE transactions on Image Processing*, 25(2):790–806, 2015.

[48] Shashank Bujimalla, Mahesh Subedar, and Omesh Tickoo. Data augmentation to improve robustness of image captioning solutions, 2021.

[49] Pierre Kornprobst, Jack Tumblin, and Frédo Durand. Bilateral filtering: Theory and applications. *Foundations and Trends in Computer Graphics and Vision*, 4:1–74, 01 2009.

[50] Sebastian Villar, Sebastian Torcida, and Gerardo Acosta. Median filtering: A new insight. *Journal of Mathematical Imaging and Vision*, 58:1–17, 05 2017.

[51] José A. Rodríguez-Rodríguez, Ezequiel López-Rubio, Juan A. Ángel Ruiz, and Miguel A. Molina-Cabello. The impact of noise and brightness on object detection methods. *Sensors*, 24(3), 2024.

DEPARTMENT OF SOME SUBJECT OR TECHNOLOGY
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden
www.chalmers.se

CHALMERS
UNIVERSITY OF TECHNOLOGY