Journal of Intelligent Manufacturing
https://doi.org/10.1007/s10845-024-02411-5

Flexible and robust detection for assembly automation with YOLOv5: a case study on HMLV manufacturing line

Alexej Simeth[1] · Atal Anil Kumar[1] · Peter Plapper[1]

Received: 25 August 2023 / Accepted: 23 April 2024

Abstract

Automating assembly processes in High-Mix, Low Volume (HMLV) manufacturing remains challenging, especially for Small and Medium-sized Enterprises (SMEs). Consequently, many companies still rely on a significant amount of manual operations with an overall low degree of automation. The emergence of artificial intelligence-based algorithms offers potential solutions, enabling assembly automation compatible with multiple products and maintaining overall production flexibility. This paper investigates the application of the YOLO (You Only Look Once) object detection algorithm in an HMLV production line within an SME. The performance of the algorithm was tested for different cases, namely, (a) on different products having similar product features, (b) on completely new products, and (c) under different lighting conditions. The algorithm achieved precision and recall greater than 98% and mAP50:95 greater than 97%.

Keywords High-mix low-volume (HMLV) · Assembly automation · Artificial intelligence (AI) · You only look once (YOLO)

Introduction

The manufacturing industry is undergoing significant innovation and changes due to the integration of sensors and the Internet of Things (IoT), increased data availability, and advancements in robotics and automation (Vaidya et al., 2018; Zhong et al., 2017). These changes have created demands for product differentiation and personalisation, due to which the manufacturing sector is moving from high-volume, low-mix production to high-mix, low-volume (HMLV) production (Abu-Samah et al., 2017; Alduaij & Hassan, 2020). HMLV manufacturing is a production strategy that involves the simultaneous production of a large variety of products in the same production line, each in relatively small quantities. This approach contrasts traditional mass production methods, typically producing a limited number of products in high quantities (Fernandes et al., 2012; Holtewert & Bauernhansl, 2016). HMLV manufacturing is commonly used in aerospace, defence, and medical device manufacturing industries, where the demand for products is highly diversified, and the need for customisation is high (Tahmina et al., 2022). HMLV manufacturing requires a high degree of flexibility and agility in the manufacturing process and robust systems for product design and engineering, production planning, and supply chain management. It also requires advanced manufacturing technologies such as 3D printing, robotics, and digital twinning (Johansen et al., 2021). The goal of HMLV manufacturing is to achieve a balance between cost-effectiveness, product quality, and time-to-market.

HMLV production is predominantly implemented in small and medium-sized enterprises (SMEs). Most of those SMEs still rely on a significant amount of manual operations, and the degree of automation is often low or nonexistent (Downs et al., 2021; Karaulova et al., 2019; Kleindienst & Ramsauer, 2015). However, the changing market dynamics and the competition among various producers may require companies to have a high degree of automation to remain competent (Transeth et al., 2020). This is especially challenging for SMEs in high-wage countries, as these companies face a

✉ Alexej Simeth
alexej.simeth@uni.lu

✉ Atal Anil Kumar
atal.kumar@uni.lu

Peter Plapper
peter.plapper@uni.lu

[1] Department of Engineering, University of Luxembourg, 6 Rue Richard Coudenhove-Kalergi, 1359 Kirchberg, Luxembourg

Published online: 23 May 2024

trade-off between the increased output and productivity of automated assembly systems and the flexibility and versatility of manual assembly. With the spread of the Industry 4.0 paradigm and the implementation of advanced technologies, there is a broad scope for SMEs to adapt their HMLV production line to manage the market demands successfully. Among these technologies, artificial intelligence (AI) (including machine learning (ML) and deep learning (DL)) and computer vision (CV) have been gaining much relevance due to their direct influence on intelligent manufacturing systems (Pierleoni et al., 2020). With the availability of advanced camera systems and improved AI algorithms, companies can move towards the desired level of performance for their production line.

CV stands as a robust domain that garners substantial attention from researchers worldwide. It enables machines to interpret, perceive, and scrutinise complex visual scenarios. Within the realm of CV, a wide array of tasks exist, spanning from object recognition, object detection, video tracking, object segmentation, pose estimation, and motion estimation. Among these tasks, object detection has emerged as a focal point for researchers due to its incorporation of classification, localisation, and segmentation elements (Kaur & Singh, 2022).

Many AI/ML/DL models have been developed for object detection (Zou et al., 2023). They have been widely used in real-world applications such as vision for robotics, continuous monitoring using video, autonomous driving, etc. Several literature reviews are available highlighting the working of these models and their scope (Diwan et al., 2023; Ren et al., 2022; Zou et al., 2023). More recently, they have been used in the manufacturing industry in the context of Industry 4.0 for different applications such as defect detection, quality, or logistics (Yi et al., 2021; Zheng et al., 2021). However, several challenges still need to be addressed, especially when it comes to implementing such algorithms in real-time for their widespread use in the industry. One such challenge is the implementation of different algorithms in an HMLV production line as one of the major requirements of an HMLV setup is that the production line must be advanced enough to accommodate divergences from initial production schedules, adeptly recognise and differentiate among the various items being handled on the factory floor, perform physical changes of physical configuration and demonstrate the ability to adapt to new models as they are progressively introduced into the assembly line.

This work aims to investigate the use of the state-of-the-art YOLO object detection algorithm (version 5) for enabling the automation of assembly tasks in SMEs consisting of HMLV production lines. To investigate its performance in different real-world production scenarios, the algorithm is tested with changing products having similar features and under different illumination conditions.

The scientific contributions of the work are:

1. Implement YOLOv5 for object detection in HMLV production lines to help SMEs automate assembly tasks.
2. Systematic evaluation of the robustness of the algorithm in real-world scenarios characterised by changing products with similar features.
3. Performance of the algorithm under different illumination conditions to highlight its resilience to lighting variations commonly encountered in manufacturing environments.

The paper is organised as follows: Sect. 2 presents a brief background on the current state-of-the-art object detectors for the manufacturing sector. Section 3 explains the methodology followed in this work, including the experimental setup and the design of experiments. Section 4 presents the experimental results of the various cases defined in the work. Section 5 presents the discussion. Section 6 presents the conclusion and future work.

Background

Traditional object detectors

The first object detector was published in 2001 and was named the Viola-Jones Object Detector (Viola & Jones, 2001, 2004). The authors achieved real-time detection of human faces without constraints using sliding windows, i.e., to go through all possible locations and scales in an image to see if any window contained a human face. Following this, in 2005, N. Dalal and B. Triggs developed the Histogram of Oriented Gradients (HOG) feature descriptor (Dalal & Triggs, 2005) as an improvement of the scale-invariant feature transform (Lowe, 1999). The HOG detector has been an essential foundation of many object detectors (Felzenszwalb et al., 2008, 2010; Malisiewicz et al., 2011) and a large variety of CV applications for many years. Another well-known object detector is the Deformable Part-based Model (DPM), proposed by Felzenszwalb (2008).

Following this, the field of object detection accelerated rapidly due to the advancement of AI, ML, and DL algorithms. One of the first important approaches in this field was proposed by R. Girschick et al., who developed the Regions with Convolutional Neural Network (RCNN) in 2014 (Girshick et al., 2014, 2015). Since then, object detection algorithms evolved into two categories: two-stage and one-stage detectors. Two-stage detectors formulate the detection as a "coarse-to-fine" process, while the one-stage detectors complete the process in a single step (Zou et al., 2023). A summary of these detectors is presented in the following subsections.

Two-stage object detectors

The first stage of two-stage detectors generates the regions of interest (ROI) using the region proposal network (RPN). The second stage predicts the objects and bounding boxes for the proposed regions (Diwan et al., 2023). Some of the well-known two-stage detectors are RCNN, Fast RCNN (Girshick, 2015), Faster RCNN (Ren et al., 2015), Mask RCNN (He et al., 2017) and others. One of the major drawbacks of two-stage detectors is that they have reduced speed and are more complex to implement; however, they achieve a high accuracy (Zou et al., 2023).

One-stage object detectors

One-stage detectors retrieve all the objects in one-step inference, i.e., the bounding boxes are predicted over the images without the region proposal step, as a result of which the speed of detection is increased. Figure 1 shows the generic schematic architecture of one-stage object detectors.

Various one-stage detectors include you only look once (YOLO) (Redmon et al., 2016), YOLOv2-v8, YOLO-NAS (Bochkovskiy et al., 2020; Redmon & Farhadi, 2017, 2018; Terven et al., 2023), Single Shot MultiBox Detector (SSD) (Liu et al., 2016), RetinaNet (Lin et al., 2017), M2Det (Zhao et al., 2019), RefineDet (Zhang et al., 2018), DCN (Dai et al., 2017). Amongst these, YOLO and its variants have been accepted as one of the best object detection approaches due to its inference speed, accuracy, simple architectural design and generalisability (Park et al., 2021).

Working principle of YOLO

The developers of YOLO reframed the task of object detection as a regression problem instead of a classification problem, where a single CNN directly predicts the bounding box coordinates and class probabilities of multiple objects in a single step. The algorithm was named You Only Look Once since it identifies the objects and their positions with the help of bounding boxes by looking at the image only once (Diwan et al., 2023).

As a first step, the image is divided into $S \times S$ grid of cells. Each grid cell predicts a fixed amount of anchor boxes and returns the bounding box coordinates relative to the cell's position, width, height, and class probabilities for each anchor box. The fundamental concept behind detecting an object by any cell is that the centre of an object should lie inside the grid cell. The cell can detect the particular object using any suitable bounding box.

The confidence score is computed for each bounding box per grid by multiplying the class probability with Intersection over Union (IoU) between the ground truth and predicted bounding box. This helps in understanding if the object exists in the grid cell. The next step calculates the class-specific score for each bounding box of all the grid cells, including the probability of the class appearing in that box and how well the predicted box fits the object.

Each bounding box is associated with a class-specific score, box coordinates (centre position, width, height), and a classification output category at this stage. Based on the class-specific score, the non-useful boxes will be discarded based on some predefined threshold value (generally 0.5). Further reduction in the number of bounding boxes is done using non-maximum suppression, which is based on IoU and eliminates redundant and overlapping bounding boxes, ensuring the retention of only the most confident and relevant predictions.

Popular dataset and characteristics

There are several datasets available for implementing various AI algorithms. One of the most widely used datasets for object detection tasks using ML and DL-based approaches is the MicroSoft Common Objects in COntext (MSCOCO) dataset. It includes 91 different categories of objects with different characteristics per image (Lin et al., 2014). The other commonly used dataset is the PAscal Visual Object Classes (PAscal VOC) (Everingham et al., 2015). It consists of various classes and is commonly used in ML/DL object detection algorithms.

Existing works using YOLO in manufacturing

Based on the literature, YOLO-based detection frameworks have been primarily used in detection (defects) and localisation tasks. Some of the existing works are mentioned here.

The authors in (Mo et al., 2019) used YOLOv3 to identify and detect the position of the solder joints in the automobile door panel welding production line. They could accurately detect the solder joints' position with an average detection time of less than 0.2 seconds. The authors in (Jiao et al., 2022) proposed an improved object detection method based on YOLOv4 to detect a vehicle wheel weld in an image. They obtained higher accuracy in detecting the wheel weld (up to 4% higher than the baseline model). Li et al. in (Li et al., 2019) achieved an accuracy of 90% using YOLOv3 for detecting the electrical components on a printed circuit board (PCB). The authors in (Chen & Shiu, 2022) used the family of YOLO algorithms (from v2 to v5) in their automatic optical inspection (AOI) system to inspect the surface of electroplated products. They obtained an accuracy rate of around 70% for detecting real-time video data in production lines. They also demonstrated that YOLOv3 and YOLOv5 performed better than the other versions. Their system was more accurate, with YOLOv2 detecting larger objects (pock

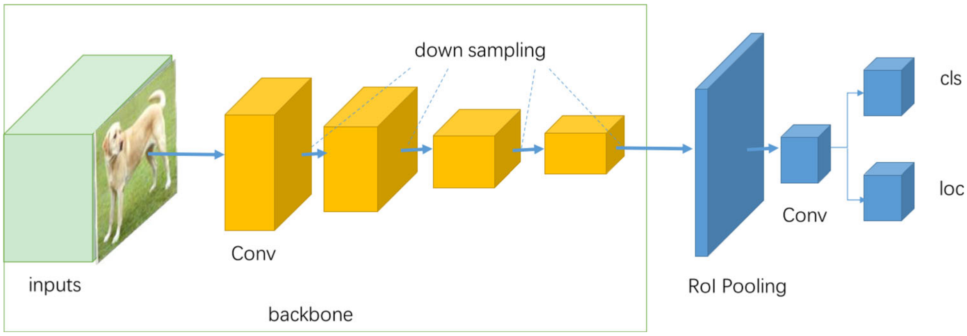

Fig. 1 Working principle of one-stage object detectors (Jiao et al., 2019)

detection). Despite these works, applying YOLO for object detection in an HMLV line is still not available.

Challenges in HMLV production line

HMLV manufacturing describes a production involving many diverse products in relatively small quantities. High-mix refers to the variety of products produced, while low-volume refers to the small quantity of each product produced.

Several challenges exist for an HMLV production line for implementing object detection algorithms.

1. Diverse product types and adaptive models
   HMLV involves several product variants with distinct shapes, sizes, and appearances. Adapting object detection models to this diversity requires detection models that are flexible, resilient, and capable of recognizing a broad array of product characteristics without compromising accuracy.
2. Scarcity of labeled data
   Since the production volume for each product is limited, labelled data are scarce. Traditional object detection models rely heavily on extensive labelled datasets. Hence, there is a need to implement recently developed object detection algorithms to overcome these limitations.
3. Real-time adaptability
   Achieving real-time adaptability is crucial in an HMLV production line to reduce downtime and increase productivity. Most of the existing object detection algorithms are not real-time and hence have limited application for a manufacturing line.
4. Environmental variability and robust models
   The object detection models should consider environmental variability, such as changes in lighting conditions and variations in product positioning. This challenge requires the models to enhance their resilience to environmental changes and improve generalization across diverse manufacturing scenarios.

Reasons for choosing YOLO

Based on the literature and the domain of application, the following are the reasons that positions YOLO as a compelling choice for our study:

1. Real-time processing and efficiency
   YOLO's architecture is optimised for real-time processing, a critical requirement in manufacturing applications where timely detection of objects is imperative. The algorithm's ability to process entire images in a single pass through the neural network enhances its efficiency, making it well-suited for scenarios demanding rapid decision-making.
2. Unified model architecture
   YOLO's single, unified model architecture supports the simultaneous detection of multiple object classes. In HMLV manufacturing, where diverse objects may need identification within a single scene, this feature simplifies model management and deployment, offering a practical solution for complex SME environments.
3. Confidence score and minimal false positives
   YOLO incorporates a confidence score in its predictions, providing a measure of confidence in the presence of a detected object. This characteristic reduces false positives, a critical consideration in manufacturing applications where precision and accuracy are essential to remain profitable.
4. Versatility and generalisation
   The algorithm's versatility in detecting various object classes makes it well-suited for HMLV environments characterised by diverse processes and products. YOLO's ability to generalise across different object categories enhances its adaptability to the dynamic nature of production settings.
5. Adaptability to different scales
   YOLO is designed to accommodate objects of varying sizes, a feature particularly advantageous in manufacturing where objects may exhibit considerable scale differences. This adaptability ensures the algorithm's

effectiveness across various manufacturing processes and product types.

Methodology

This paper showcases the suitability of the neural network-based object detector YOLO version 5 for application in HMLV processes. The target is to use the YOLOv5 model predictions, i.e., the object localisation information in the image, in manufacturing processes such as manipulation, inspection, or positioning. The novelty of this paper is the application and testing of the learning-based CV model YOLOv5 for such tasks in an HMLV manufacturing scenario. So far, YOLOv5 has not been available for such tasks due to the challenges implied by the HMLV environment. The high mix of products produced leads to a changing product portfolio, including new products. The detection model must find decisive product features on all products in the mix. The production facilities strongly affect the object detection process since illumination and lighting mainly determine the quality of images that the detection model processes. In SMEs, the typical company producing in HMLV, the facilities commonly have windows, natural light sources, and limited potential for encapsulation. Therefore, the following detection model capabilities are investigated:

1. Performance on changing & new products
2. Performance under changing environment conditions (lighting)
3. Accuracy and precision of object localisation

The following subsections present the experimental setup, the selected HMLV products, the datasets, and the design of the experiments.

Experimental setup

The overall setup is depicted in Fig. 2a. An ABB IRB 120 industrial robot is the main manipulator with the ABB IRC5 compact controller. A Multi-Funtionalool End Effector (MFEE) is installed on the robot. The MFEE comprises a camera module, a suction gripper module, a glue module, and two ring light modules (see Fig. 2b). The installed industrial camera, type DFK 33GX264 from TheImageSource, is a 5MP colour camera with a fixed 8*mm* lens. The working distance between the camera and the board is fixed during the individual experiments. The camera axis is perpendicular to the surface of the workspace. The camera is running in an automatic mode. The camera system is calibrated with a chessboard pattern following the method presented by (Zhang, 2000). The transformation of pixel locations into real-world coordinates is performed with camera flange calibration (Müller et al., 2019). For both calibrations, 20 images of the chessboard pattern (see Fig. 2b) from different angles are taken and stored together with the correlating robot pose.

The ring light concentric with the camera axis is referred to as the camera light, and the other as glue light. Each ring light consists of 24 RBG-LEDs that are controllable in an intensity range of 0-255 for each colour channel. In conducted experiments, the ring lights always have white light with the same intensity for each channel. The demonstrator is placed in the corner of the laboratory next to a window, a natural light source (see Fig. 2c). The laboratory has two light sources on the ceiling. The light source closer to the demonstrator is room light A and the other room light B. The lights in the room can be turned on or off.

The selected products for this study are four different boards (see Fig. 3a–d). Board 1 relates to Fig. 3a, board 2 to Fig. 3b, etc. The boards have a top, core, and bottom layers. Colour, structure, appearance, and reflectivity change between the boards. Furthermore, the holes in the boards differ due to the change in the material of the core layer. Each hole carries a specific component, which must be inserted during assembly. However, unlike the depicted research boards, every board has different dimensions and features a different hole pattern in reality. Thus, the assembly process is unique for each board conducted in HMLV.

Utilised datasets

Two different datasets are used for the experiments in this study. The first dataset contains a total of 1200 images. For each board, a series of 300 images is taken. Between each image, the position of the board changes (translation, rotation). Changes in lighting may occur due to the natural light source next to the test rig (window; see Fig. 2a, c). The working distance between the camera and the board is kept constant. For each 5MP image, a central square is cropped in 768x768 pixels to reduce the data size. The holes depicted on each cropped image are then annotated with bounding boxes using the Python program Label Studio (Tkachenko et al. 2020). All holes in all boards have the same label category. One image for each board in this dataset is depicted in Fig. 3a–d. This dataset is named dataset A.

The second dataset consists of images of board 1 only (s. Fig. 3a, e–h). The four different light sources are changed systematically, and five images are taken for each configuration. The board is not moved throughout the entire process. All images contain the same scenery as board 1 on the demonstrator but in different lighting situations. Nine hundred eighty images of board 1 were generated under different lighting conditions. A detailed description of the procedure and the experiment are given in the following subsection. In this paper, this dataset is named dataset B.

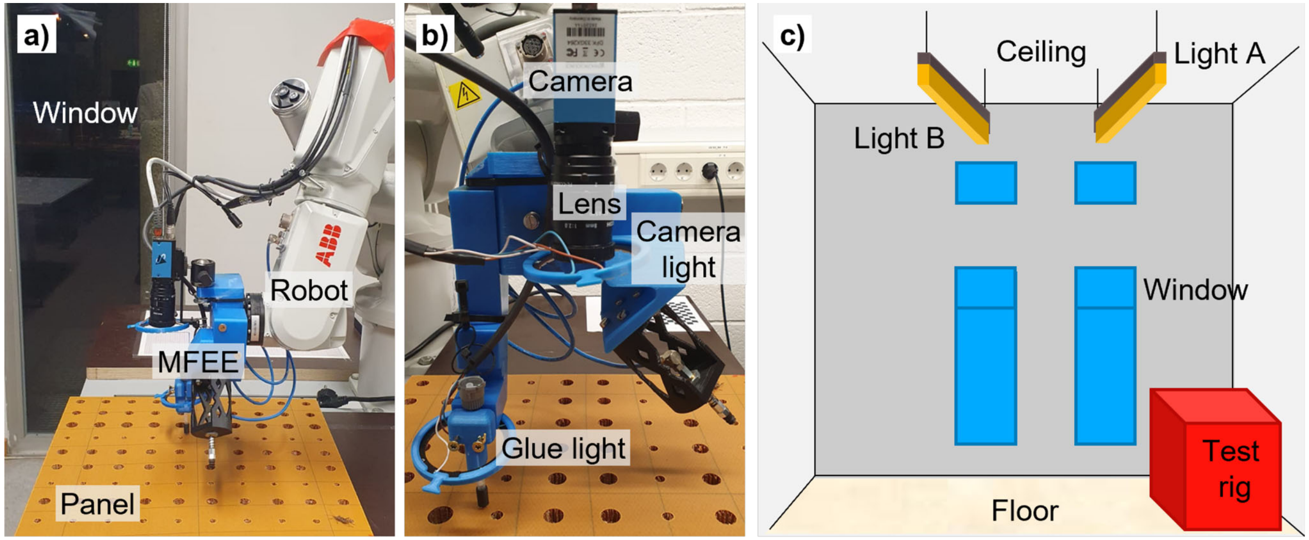

Fig. 2 **a** Side view of the overall experimental setup with the robot, MFEE, board. **b** MFEE with camera module and two ring lights. The camera light is the upper right ring around the lens. The chessboard at the right back of the test rig is used for calibration. **c** Schematic of the laboratory with the lights in the room mounted on the ceiling and the red cube representing the robot setup. The window in the bottom right is the same window depicted in image (**a**)

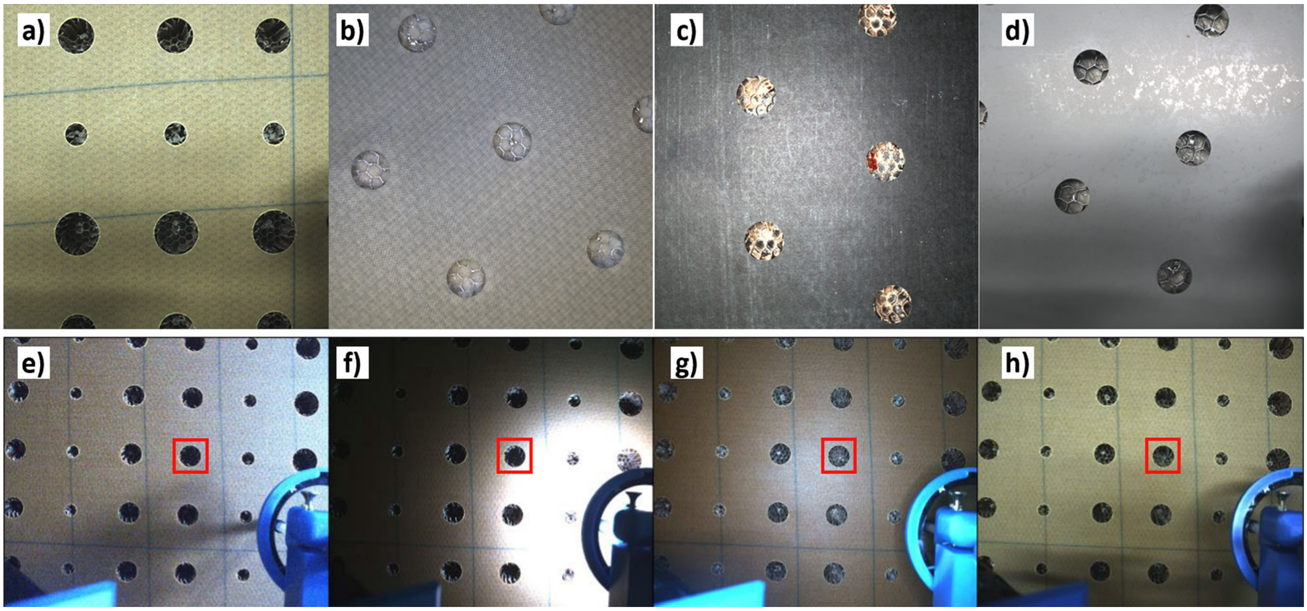

Fig. 3 **a–d** Images of the different boards 1-4 used in this study taken from dataset A. **e–h** Example images of board **a** in different lighting situations drawn from dataset B. The red bounding box indicates the most central hole used to compare the results

Design of experiments

The functionality of the state-of-the-art object detector YOLOv5 is investigated in three experiments. The models are trained and validated in Experiment A. Their localisation performance is analysed in Experiment B for different products and in Experiment C under changing lighting conditions.

Experiment A

The first experiment is designed to assess the general performance of the selected CV model on dataset A. The objective is to generate a performance reference and identify any challenges with the model training. Therefore, dataset A is divided into training, validation, and test sets in the ratio 80%:10%:10% evenly over the four boards. Then, five different reference models are trained. The first model is trained on the complete training set with all four boards. The other four models are trained on three boards, with one of the four selected boards removed from the dataset A (training and validation set). The performance of the models is determined with the validation sets.

Experiment B

The second experiment examines the models' product flexibility by testing the performance of the trained models from Experiment A on changing and new products. Here, the objective is to identify the model's ability to generalise and make predictions on unknown and new data. Therefore, the following steps are conducted:

1. The model trained on all boards is tested on the test set from dataset A.
2. The models trained on three boards only are tested on a reduced test set containing only the three boards used for training.
3. The models trained on three boards are tested on all board images of the fourth board not included in the training set.

Furthermore, the precision of the object localisation of the trained models on different subsets of dataset A is investigated. First, the aim is to quantify the performance of YOLOv5 on new products when trained on existing products only. Furthermore, the possibility of using the information from the bounding box for subsequent manufacturing processes, i.e., the insertion of the insert, shall be showcased. Therefore, the model predictions on the test sets of dataset A subsets are compared with the ground truth annotations of the test sets. The parameters IOU of the bounding boxes of the most central hole and the Euclidean distance between their centre points are used for comparison.

Experiment C

In the third experiment, we simulate working under changing lighting conditions. Images of the same scenery with changed lighting configuration are analysed by the model trained in experiment A. With this experiment, we evaluate the impact of changing lighting situations on the ability of the model to identify product features, that is, holes, and to predict their location precisely. This time, full-resolution images (2448x2048 pixels) are fed to the model. The detection results for the most central object are used for comparison (see Fig. 3e–h).

Four different lighting cases can be created by alternating the room light status. The ring lights can also be turned off or turned on at a defined light intensity. We incremented the light intensity of the ring lights by 25 for each channel, totalling ten different intensities, i.e. (25, 25, 25), (50, 50, 50), ..., (250, 250, 250). An overview of the lighting scenarios used is given in Table 1. The images were taken with the following procedure:

Algorithm 1 Image generation procedure for dataset B

```
For each Case 1-4:
    For each Subcase 1-6:
        Do until specified intensity reached:
            Take five images
            Increment ring light(s) intensity by 25 units
```

YOLOv5 object detector is run on each image, and the bounding box parameters of the most central hole are used for comparison with a reference bounding box from the annotation. The model trained on the entire dataset A is applied for object detection.

Performance metrics

The model training is performed only on images from the dataset A. The used set of images is always divided into three parts: training, validation, and testing. Therefore, the images in the test set have not been used during model training and are unknown to the models. All images of dataset B (lighting) are unknown to all trained models. The evaluation is performed by comparing the metrics precision (P), recall (R), and medium average precision (mAP) at different IoU thresholds.

The primary metrics to determine the localisation accuracy of the models are based on the bounding box information of the most central element detected on each image. The four parameters of the bounding box are the centre coordinates, the width, and the height (x, y, w, h). Based on x and y, the spatial accuracy of the bounding box on the image is determined by calculating the Euclidean distance between the reference centre and the predicted centre. Additionally, the inference time is tracked.

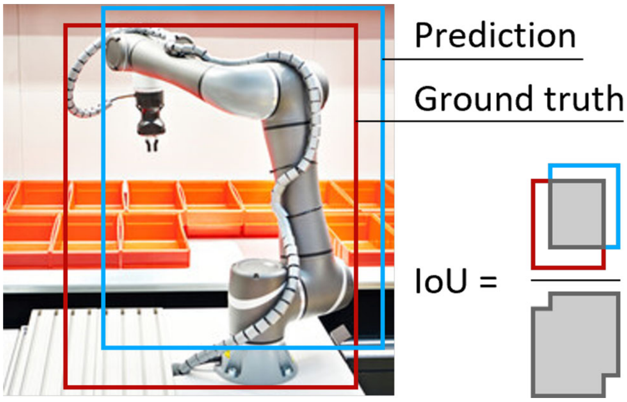

Fig. 4 Intersection over Union adapted from (Elgendy, 2020). Picture source: (Adobe, xxx)

Intersection over Union (IoU), also known as the Jaccard index, is a widely adopted measure for evaluating the similarity between two arbitrary shapes. It captures the distinctive shape attributes of the compared objects, such as the widths, heights, and positions of their respective bounding boxes. This data is incorporated into the spatial property, leading to the computation of a standardised value that emphasises their shared areas (or volumes). This characteristic renders IoU unaffected by the scale of the problem being examined. Thanks to this attractive characteristic, every evaluation metric employed for tasks like segmentation (Cordts et al., 2016; Zhou et al., 2017), object detection (Lin et al., 2014; Everingham et al., 2010), and tracking (Lin et al., 2017) depends on this measurement.

It is calculated by the overlap between the ground truth bounding box G and the predicted bounding box P divided by their union (Eq. 1). The value of IoU ranges between 0 (no overlap) and 1 (100% overlap). A prediction is counted as true positive (TP) if the IoU value exceeds a defined threshold, usually 0.5.

$$\mathrm{IoU}(G, P) = \frac{G \cap P}{G \cup P} \tag{1}$$

Precision measures the number of correct instances retrieved divided by all retrieved instances (Eq. 2). *Recall* measures the number of correct instances retrieved divided by all correct instances (Eq. 3).

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \tag{2}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}} \tag{3}$$

Here, *TP*, also known as true positive, represents a positive sample correctly classified; *FP*, also known as false positive, represents positive outcomes that the model predicted incorrectly; *FN*, also called false negative, represents a negative outcome that the model predicted incorrectly.

Table 1 Overview of the lighting scenarios by lighting case

Lighting case	Room A	Room B	Camera	Ring	# images
Case 1	Off	Off	–	–	245
Case 2	On	Off	–	–	245
Case 3	Off	On	–	–	245
Case 4	On	On	–	–	245
Subcase 1	–	–	Off	Off	5
Subcase 2	–	–	25-250[1]	Off	50
Subcase 3	–	–	Off	25-250	50
Subcase 4	–	–	25-250	25-250	50
Subcase 5	–	–	On (=250)	25-225	45
Subcase 6	–	–	25-225	On (=250)	45

[1] 25-250 and 25-225 indicate incrementing the intensity from 25 to 250/225 with an incrementation of 25

The mAP is used as a measure of the CV model performance and is equal to the mean value of the average precision metrics of all individual model classes (Eq. 4) (Francies et al., 2022).

$$mAP = \frac{1}{n}\sum_{k=1}^{k=n} AP_k \tag{4}$$

where, AP_k represents the average precision of the individual class, and n is the total number of classes.

Model implementation

The object detection model YOLOv5 is implemented in version v7.0-70-g589edc7, and the pre-trained weights *yolov5x.pt* with 86.7 million parameters are used as the basis for the training of object detectors (Jocher et al. 2022). In total, five different YOLOv5 object detection models are generated by retraining the same pretrained weights *yolo5x.pt* on five different compositions of dataset A. The training hyperparameters, such as the number of epochs, optimiser, batch size, and augmentation, are kept constant for each detection model training. All experiments are run on Google Colabs using the PyTorch framework and GPU (Tesla T4). A stochastic gradient descent optimiser is selected with a learning rate of 0.01. Each model is trained for 100 epochs with a batch size of 16. The images are passed to the model in size 640 × 640 × 3. Augmentation is applied, and the parameters include augmentation of the hue, saturation, and value, translation, scaling, vertical reflection, and image mosaic creation. The model fitness after each training epoch is calculated based on the weighted average of the metrics mAP50 and mAP50:95 in the ratio 10%:90%. The model with the highest fitness is selected for later detection.

Results

In experiment A, all object detection models used in this paper are trained and validated. Their spatial accuracy is analysed in experiment B for different products and in experiment C for different lighting conditions.

Experiment A: general model performance

From dataset A, five different training subsets are created and applied to train five object detectors. The models and datasets are named based on the boards used for training. The dataset name D1234 indicates that images from all four boards are included. The model name M1234 is trained on dataset D1234, including all 1200 images of dataset A. The dataset D134 consists of the images of boards 1, 3, and 4, in total 900, and is used to train model M134. The specification of the datasets is given in Table 2. Although 300 images are taken per board, the number of instances on board 1 is higher than on the other boards. Due to a different hole pattern, board 1 features approximately ten instances per image. The other boards 2, 3, and 4, have six on average.

The training results are given in Table 3. All models are trained with GPU for 100 epochs. The training took around 98 minutes for the smaller datasets and 122 minutes for training on the complete dataset A training set. The model with the highest weighted average of mAP50 and mAP50:95 on the validation set is selected for later application. For the models M1234 and M234, the best results are achieved after epochs 98 and 87, respectively. The others achieve the highest performance after 100 epochs of training. The results of all trained models show high performance on precision, recall and mAP50 parameters, which are all larger than 0.988. Almost all objects are found and predicted correctly. Also, for high IoU thresholds, all detectors predict objects with an mAP50:95 of 0.974 and higher. The inferior predictions occur mostly on divided holes at the edge of the image.

Table 2 Datasets used in Experiment A drawn from dataset A

Dataset	Total images	Instances	Board 1	Board 2	Board 3	Board 4
D1234	1200	8562	X	X	X	X
D123	900	6676	X	X	X	
D124	900	6747	X	X		X
D134	900	6712	X		X	X
D234	900	5551	X	X	X	

All datasets have the split training-validation-testing: 80%:10%:10%

Table 3 Training results of YOLOv5 object detectors

Model name	Validation images	instances	Best epoch	Time [min]	Precision	Recall	mAP50	mAP50:95
M1234	120	841	98	122	**0.991**	0.992	0.994	0.978
M123	90	648	100	97.6	0.989	0.996	0.993	0.975
M124	90	660	100	98.6	0.990	0.991	0.993	0.974
M134	90	648	100	97.7	**0.991**	**0.997**	0.994	**0.98**
M234	90	567	**87**	**94.7**	0.988	0.996	0.995	**0.98**

The bold values indicate the best results obtained for the respective metrics

Model: YOLOv5 v7.0-70-g589edc7. Weights: yolov5x.pt. Augmentation: hyp.scratch-med.yaml. Python version: 3.8.16. Framework: torch-1.13.0 + cu116. GPU: Tesla T4

The learning curves for all models are given in Fig. 5. Precision, recall, mAP, box loss, and object loss of each model are plotted for each epoch. While precision, recall, and mAP are relative numbers in the interval [0, 1], the losses are absolute values and calculated with the according loss functions. The objectness loss describes whether an object is present in a particular grid cell or not and is calculated with the Binary Cross-Entropy loss function (Redmon & Farhadi, 2018). The box loss measures the error in localising the object within the grid cell and is calculated using Complete IoU loss (CIoU) (Zheng et al. 2019). The curves are close to each other and show a similar profile. The box and object loss decrease with a high gradient until epoch 12. Then, the curves become more flat. By then, precision, recall, and mAP50 are already close to 1 and do not increase significantly throughout the remaining epochs. Both losses decrease slowly until epoch 100, while the mAP50:95 continuously improves. All curves indicate that there are no issues throughout the training process.

Experiment B: product flexibility

The ability of the models to generalise is analysed by running them on new, unknown data. Therefore, the performance is measured on the test sets. Additionally, the models trained on only three boards are tested on the complete set of images of the board removed from training (s. Table 4). The performances of all models are again high. The models produce similar results on both the validation and test sets. Models trained on three boards achieve slightly reduced performance on the fourth board. The largest difference is visible in the mAP50:95. Still, results on the fourth board for mAP50:95 reach at least 0.945. The inference time varies between 50 and 56.8 ms.

The performance on localisation accuracy is determined by comparing the prediction and ground truth bounding box data of the central hole of each image. The characteristics of both IoU and centre distance are depicted in the box plot charts in Figs. 6 and 7. For all tested model test set combinations, the results are close to each other. Regarding the IoU, the five models trained and tested on three or four boards do not vary much. Boxes, whiskers, and outliers are in the same range, with minimal IoU values at 0.93. The upper limit for each observation is a perfect bounding box prediction with 100% overlap. Models trained on three boards and tested on the fourth board not included in the training data show a higher variation in IoU. The number of outliers and their significance are stronger pronounced in those results. The average IoU across all predictions of all combinations tested is 0.9712. An example of bounding boxes with this IoU is given in Fig. 8 on the left. Since all images in one test set are different with different locations of the central hole, a grey-filled circle is drawn to represent an average hole. In the same Fig. 8, the worst prediction in terms of IoU is given. The image to this prediction is from dataset D3 and is not sharp, but it shows a board in motion with a blurred hole. The same is the case for the two outliers of M134-D2 observations with the lowest IoU. Here again, the translation and rotation process was not entirely finished, and the image capture had already started during the board repositioning. In these images, the predicted bounding box is smaller than the annotation. The soft edges are not included in the prediction. These images are only part of the complete sets of 300

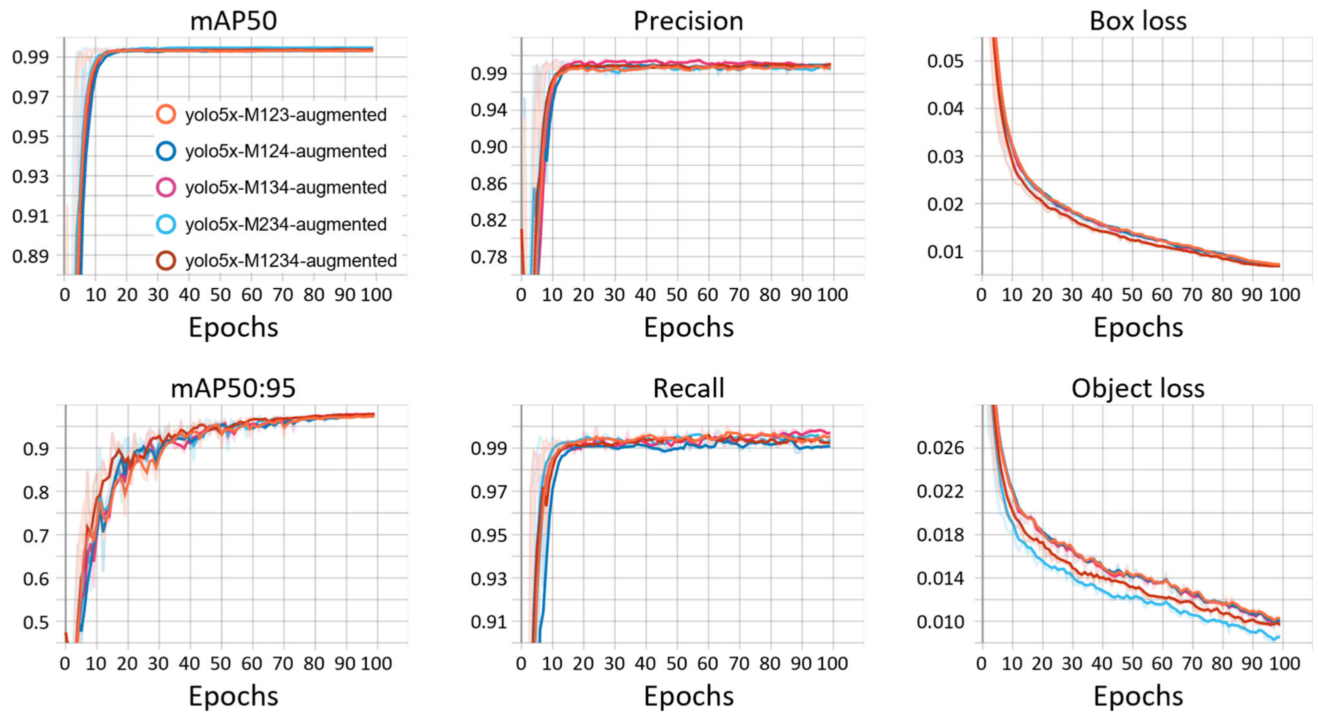

Fig. 5 Learning curves (with smoothing) for the parameters mAP50, mAP50:95, precision, recall, box loss, and object loss for all models plotted against the number of epochs. mAP50, mAP50:95, precision, and recall are unitless relative values between 0 and 1. The box and object loss are the absolute losses calculated with the according loss function

Table 4 Testing results of YOLOv5 object detectors

Model	Set	Images	Instances	Precision	Recall	mAP50	mAP50:95	Inference
M1234	D1234	120	836	0.99	0.999	**0.995**	**0.979**	54.0 ms
M123	D123	90	652	0.989	0.997	**0.995**	0.978	53.0 ms
	D4	300	1886	**0.993**	0.996	**0.995**	0.976	53.5 ms
M124	D124	90	651	0.989	**1**	**0.995**	0.977	54.3 ms
	D3	300	1815	0.985	0.994	**0.995**	0.965	**50.0** ms
M134	D134	90	641	0.992	0.996	**0.995**	0.975	56.8 ms
	D2	300	1850	0.989	0.994	**0.995**	0.945	55.2 ms
M234	D234	90	564	**0.993**	**1**	**0.995**	0.978	54.7 ms
	D1	300	3011	0.987	0.996	0.994	0.958	54.0 ms

The bold values indicate the best results obtained for the respective metrics

images per board and are not included in the test sets, i.e., the 10% selected for testing.

The measurements of the distance between the predicted and ground truth bounding box are given in Fig. 7. The overall variation without outliers is low. All observations range from 2 pixels to 0, i.e., no distance between centres. Model M1234 is the most precise, and the box of the plot is distributed in the interval [0.5, 0.7]. For most predictions, the x and y coordinates differ from the annotation by not more than 0.5 pixels. In the results of the other models, there is no significant difference. Especially, the behaviour of models trained on three boards is unchanged regarding the centre distance for known products and unknown products. Although the blurred images are still in the single board sets, this does not have the same effect as it has on the IoU. The largest distance is predicted on a sharp image drawn from the board 3 dataset (see Fig. 8, right). The largest distance is close to 2.7 pixels, which is in the setting of experiment B 0.41 mm. The average predicted bounding box centre is 0.73 pixels or 0.11 mm away from the annotated centre. In the example given for the average distance, the centre markers and the drawn circles are aligned to a very large degree.

Experiment C: robustness

For experiment C, 980 images of the same scene are taken in 24 different lighting scenarios. An image with fairly uniform lighting of the central hole is selected to create the reference annotation. The reference image is done in Case 1, Subcase 1, with both ring lights turned off (see Fig. 9). The bounding box is square with an edge length of 164 pixels, which is proportional to a 14.7 mm hole diameter. The bounding box parameters (centre coordinates, width, height) are equal for all images from dataset B. The predictions for the central bounding box for all images of dataset B are generated with the M1324 model trained on the complete dataset A training data. The predicted bounding box is then compared with the reference bounding box (see Table 3).

Four main lighting cases and six subcases have been defined to investigate the influence of different lighting on the prediction results. However, there is no significant change in prediction performance for the different cases. Figure 10 compares the predicted and reference bounding boxes. For all images of dataset B, exactly one instance is predicted for the central hole. Thus, recall and precision values are one for the considered hole since it is always detected at the correct location exactly once. The difference in x-coordinates between the prediction and the reference ranges from 0 to − 1.5 pixels. The median line is close to the upper box line, equal to − 0.5 pixels. The average for the x-centre coordinate is at approx. − 0.87 pixel (<0.1 mm). Overall, the x-coordinate prediction is very precise with low variation. This is valid for every lighting case. The y-coordinate prediction is again very precise, with all values in the range of 1.5 pixels. The accuracy is slightly reduced, and the difference compared to the reference is larger, with the average at − 3.18 pixels (0.28 mm) and the median at − 3 pixels (0.27 mm). Again, the results are similar for all lighting cases. The predicted bounding box centre is shifted to the top left of the image. The maximum Euclidean distance between reference and prediction is 4.27 pixels, which is in the setup of experiment C less than 0.4 mm.

The overlap of the predicted bounding box with the reference is measured using the IoU. The corresponding box plot shows a minimum IoU of greater than 0.935. Most predictions have an IoU above 0.95, which is commonly the most significant threshold for accepting a positive prediction. Overall, the overlap of the bounding boxes is considerable. The worst and average predictions in terms of centre distance and IoU are shown in Fig. 11. The reference is green with a plus marker, and the prediction is red with an x-marker. To emphasise the centre distance, a circle is plotted based on the height and width of the bounding box. Overall, the detection results are high quality, with only minor deviations from the reference. The bounding box, circle, and marker added to the images in original resolution can only be plot-

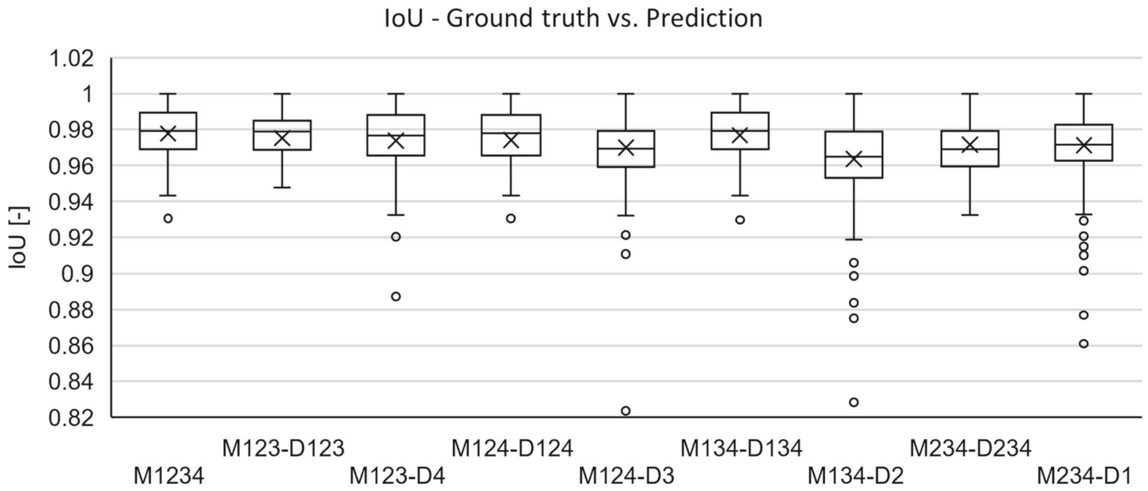

Fig. 6 Comparison of IoU-calculations for the central hole predictions for all models and test sets combinations

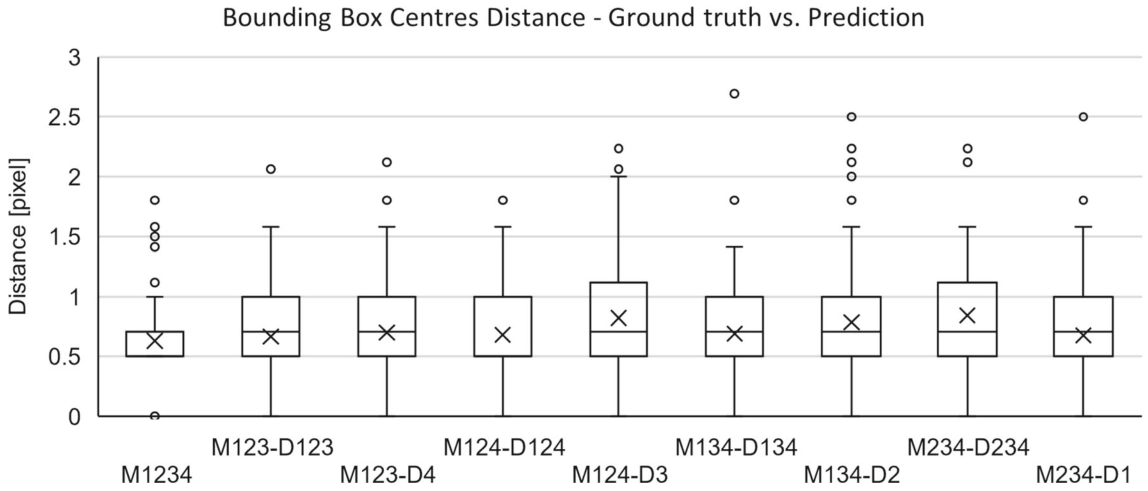

Fig. 7 Comparison of distance calculations in pixels for central hole predictions for all models and test sets combinations

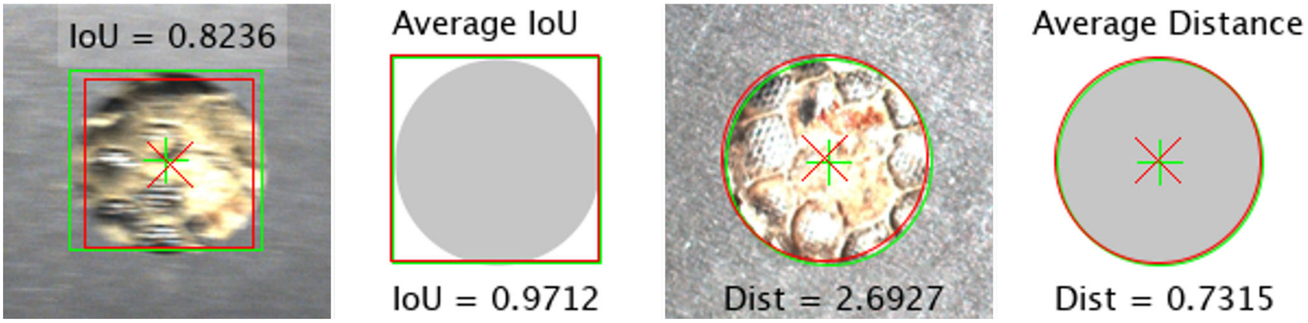

Fig. 8 Worst and average prediction result for IoU (left) and centre distance in pixel (right). Predictions are indicated with the red line and the x-marker, and annotation with the green line and the plus-marker (Color figure online)

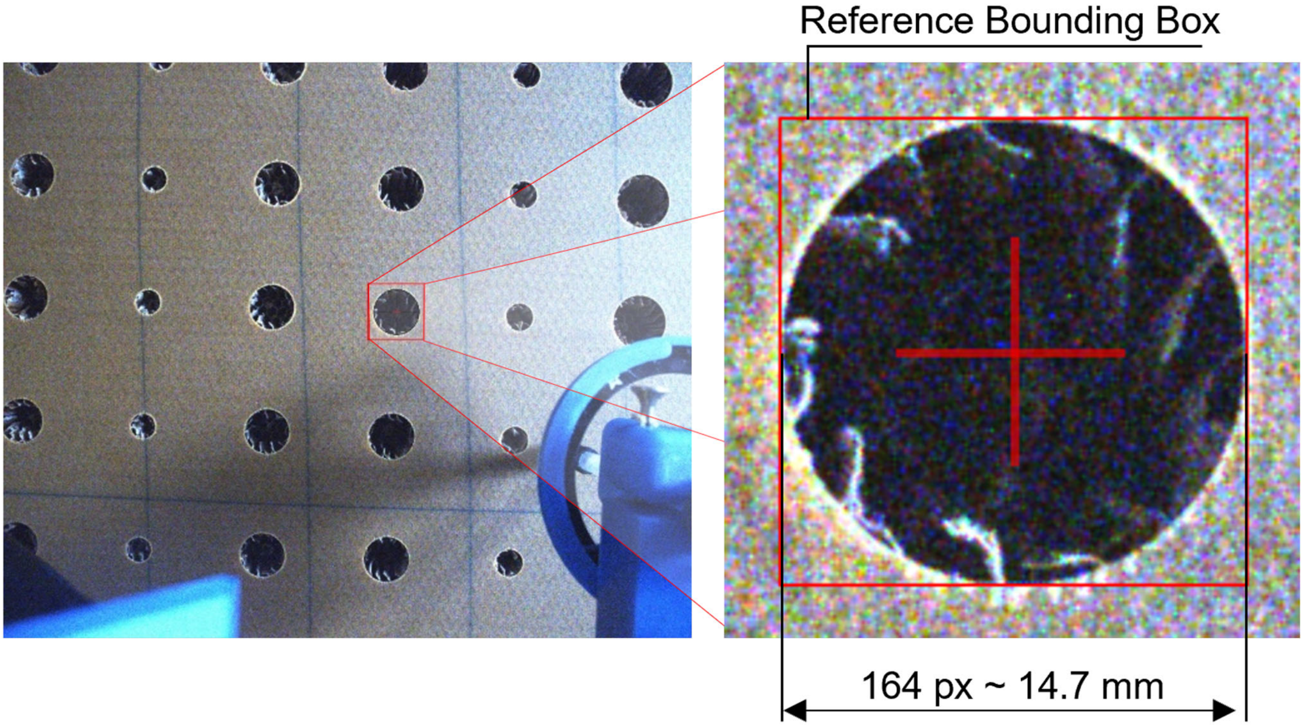

Fig. 9 Reference image for experiment B with a close-up of the hole closest to the image centre. The reference bounding box and the centre of the bounding box are highlighted in red (Color figure online)

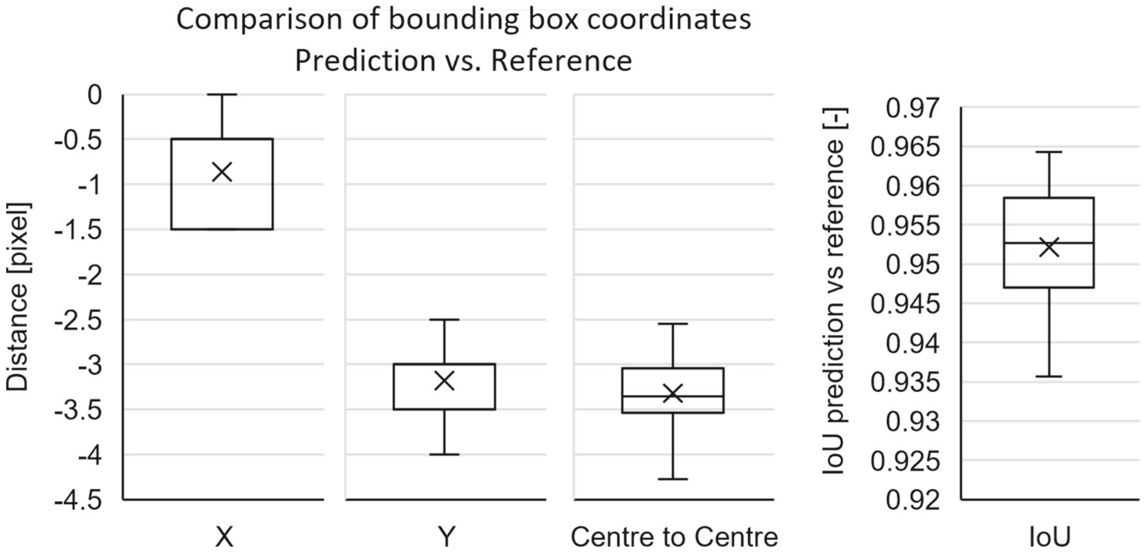

Fig. 10 Comparison of the coordinates of the central bounding box and IoU of the predicted and reference bounding box

ted on full-pixel level. Calculations and later transformation into Cartesian coordinates are on the sub-pixel level.

In experiment C, the images are input into the model in full size (5MP). 28 holes are visible in each image. The inference time increases to 81.8 ms.

Discussion

Experiment A involved training the YOLOv5 models on datasets of different products with similar product features. The products selected for this study are four multi-layer boards made from different materials with varying number of holes and their position. Several datasets containing three or four boards were created and used to train several YOLOv5

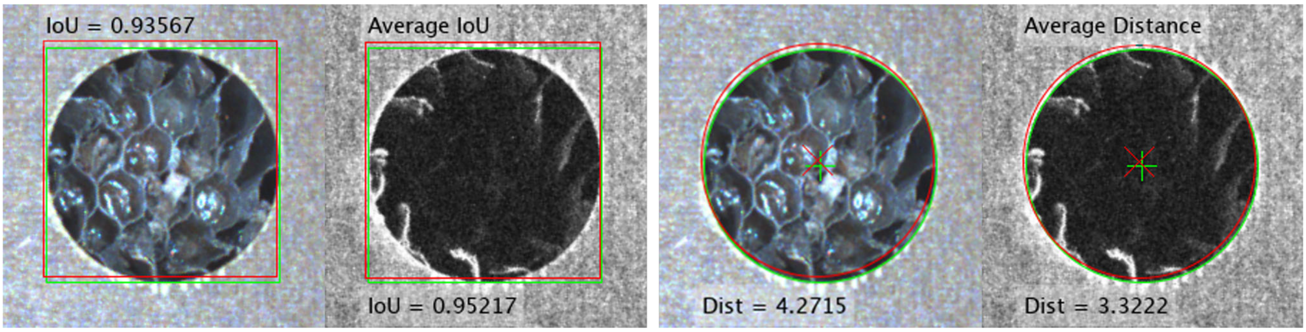

Fig. 11 Worst and average prediction results for IoU (left) and centre-distance (right). Predictions are indicated with the red line and the x-marker, references with the green line and the plus-marker (Color figure online)

models. The validation results were good for all models, with precision and recall greater than 98% and mAP50:95 greater than 97%.

In experiment B, the trained models were tested on new data of the same distribution or completely new data. Here, we want to simulate the change in product lineup with existing and new products. The YOLOv5 models generalised well in our experiments and could predict similar features on completely unknown products. For the object detected closest to the image centre, precision and recall were 1, meaning the object was always found exactly once. Additionally, the spatial accuracy of the predictions was very high. The predicted bounding boxes had a very high overlap with the annotations (IoU on average 0.97). The average distance of the bounding box centre between annotation and prediction was less than one pixel, which is, in this setup, approximately 0.15 mm. This level of accuracy makes it possible to use YOLOv5 for manufacturing processes such as insertion, screwing, and glueing, as well as for guiding robots to the point of interest. Additionally, the inference time of 55 ms is fast enough for many applications.

With different lighting scenarios in experiment C, we simulated changing environments in production lines induced by natural light through windows, artificial light sources, or moving objects, which has, in general, a significant impact on CV systems. We found that the results were similar to those of experiment B. Precision and recall were 1 for the central hole, and the IoU was, on average, 0.95, indicating a high level of overlap. The maximum distance between predicted and annotated bounding box centres was 4.7 pixels, which is still less than 0.4 mm since a larger image size and changed working distance were used. The larger input data increased the inference time to 81.8 ms, which is still very quick.

Conclusion

A significant challenge in HMLV manufacturing is the high degree of flexibility and agility required in the manufacturing processes. In this paper, we analysed the applicability of the neural network-based object detector YOLOv5-v7.70 for those environments. To investigate this, we conducted several experiments to test the performance of YOLOv5 on changing products and under varying lighting conditions. Furthermore, we assessed the potential to utilise the prediction information to execute manufacturing processes.

Based on these experiments, we have found that YOLOv5 is a robust and precise model for object localisation and can detect multiple products without interchange. If a new product of a similar family is launched, the potential is high that the model can be used directly or with minor adjustments or new data. The model also shows superior robustness against changing environmental conditions, including strong changes in lighting or corrupted data as blurred images taken in motion.The analysis also found that the influence of lighting conditions on YOLOv5's predictions was minor compared to the board change. Overall, state-of-the-art neural network-based object detectors are suitable for their application in real manufacturing environments, at least for the product selected for this study. However, their application may only be efficient for some types of HMLV production and needs to be analysed case by case.

Future work includes conducting experiments on the actual equipment with subsequent execution of manufacturing tasks, using a more complex dataset with different geometries and multiple object classes, implementing newer versions of YOLO (version 6, 7, and 8, YOLO-NAS) released in 2022 and 2023 and achieving higher performance on benchmark datasets.

Author contributions AS: Conceptualisation, methodology, software, investigation, data collection, experimentation, writing. AAK: Writing, methodology, investigation. PP: Supervision, review and editing.

Funding Not applicable.

Data availability The data that support the findings of this study are available from the corresponding author upon request.

Declarations

Competing interest The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Abu-Samah, A., Shahzad, M. K., & Zamai, E. (2017). Bayesian based methodology for the extraction and validation of time bound failure signatures for online failure prediction. *Reliability Engineering & System Safety, 167*, 616–62. https://doi.org/10.1016/j.ress.2017.04.016

Adobe. Lizenzfreie Stockfotos und Bilder. Retrieved from https://stock.adobe.com/de/photos

Alduaij, A., & Hassan, N. M. (2020). Adopting a circular open-field layout in designing flexible manufacturing systems. *International Journal of Computer Integrated Manufacturing, 33*(6), 572–589. https://doi.org/10.1080/0951192X.2020.1775300

Bochkovskiy, A., Wang, C. Y., & Liao, H. Y. M. (2020). Yolov4: Optimal speed and accuracy of object detection. arXiv preprint arXiv:2004.10934.

Chen, Y. W., & Shiu, J. M. (2022). An implementation of YOLO-family algorithms in classifying the product quality for the acrylonitrile butadiene styrene metallization. *The International Journal of Advanced Manufacturing Technology, 119*(11–12), 8257–826. https://doi.org/10.1007/s00170-022-08676-5

Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., et al. (2016). The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3213–3223).

Dai, J., Qi, H., Xiong, Y., Li, Y., Zhang, G., Hu, H., et al. (2017). Deformable convolutional networks. In *Proceedings of the IEEE international conference on computer vision* (pp. 764–773).

Dalal, N., & Triggs, B. (2005). Histograms of oriented gradients for human detection. In *2005 IEEE computer society conference on computer vision and pattern recognition* (Vol. 1, pp. 886–893). IEEE.

Diwan, T., Anirudh, G., & Tembhurne, J. V. (2023). Object detection using YOLO: Challenges, architectural successors, datasets and applications. *Multimedia Tools and Applications, 82*(6), 9243–927. https://doi.org/10.1007/s11042-022-13644-y

Downs, A., Kootbally, Z., Harrison, W., Pilliptchak, P., Antonishek, B., Aksu, M., et al. (2021). Assessing industrial robot agility through international competitions. *Robotics and Computer-Integrated Manufacturing, 70*, 10211. https://doi.org/10.1016/j.rcim.2020.102113

Elgendy, M. (2020). *Deep learning for vision systems*. Simon and Schuster.

Everingham, M., Eslami, S. A., Van Gool, L., Williams, C. K., Winn, J., & Zisserman, A. (2015). The pascal visual object classes challenge: A retrospective. *International Journal of Computer Vision, 111*, 98–136. https://doi.org/10.1007/s11263-014-0733-5

Everingham, M., Van Gool, L., Williams, C. K., Winn, J., & Zisserman, A. (2010). The pascal visual object classes (voc) challenge. *International Journal of Computer Vision, 88*, 303–33. https://doi.org/10.1007/s11263-009-0275-4

Felzenszwalb, P., McAllester, D., & Ramanan, D. (2008). A discriminatively trained, multiscale, deformable part model. In *2008 IEEE conference on computer vision and pattern recognition* (pp. 1–8). IEEE.

Felzenszwalb, P. F., Girshick, R. B., & McAllester, D. (2010). Cascade object detection with deformable part models. In *2010 IEEE computer society conference on computer vision and pattern recognition* (pp. 2241–2248). IEEE.

Fernandes, R., Gouveia, J. B., & Pinho, C. (2012). Product mix strategy and manufacturing flexibility. *Journal of Manufacturing Systems, 31*(3), 301–31. https://doi.org/10.1016/j.jmsy.2012.02.001

Francies, M. L., Ata, M. M., & Mohamed, M. A. (2022). A robust multiclass 3D object recognition based on modern YOLO deep learning algorithms. *Concurrency and Computation: Practice and Experience, 34*(1), e651. https://doi.org/10.1002/cpe.6517

Girshick, R. (2015). Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision* (pp. 1440–1448).

Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 580–587).

Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2015). Region-based convolutional networks for accurate object detection and segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 38*(1), 142–158. https://doi.org/10.1109/TPAMI.2015.2437384

He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision* (pp. 2961–2969).

Holtewert, P., & Bauernhansl, T. (2016). Interchangeable product designs for the increase of capacity flexibility in production systems. *Procedia CIRP, 50*, 252–257. https://doi.org/10.1016/j.procir.2016.04.129

Jiao, L., Zhang, F., Liu, F., Yang, S., Li, L., Feng, Z., et al. (2019). A survey of deep learning-based object detection. *IEEE Access, 7*, 128837–12886. https://doi.org/10.1109/ACCESS.2019.2939201

Jiao, L. T., Guo, P. W., Hong, B., & Feng, P. (2022). Vehicle wheel weld detection based on improved YOLO v4 algorithm. *Computer Optics, 46*(2), 271–279.

Jocher, G., Chaurasia, A., Stoken, A., Borovec, J., Kwon, Y., Michael, K., et al. (2022) ultralytics/yolov5: v7.0—YOLOv5 SOTA realtime instance segmentation. Retrieved from https://zenodo.org/record/7347926

Johansen, K., Rao, S., & Ashourpour, M. (2021). The role of automation in complexities of high-mix in low-volume production-a literature review. *Procedia CIRP, 104*, 1452–1457. https://doi.org/10.1016/j.procir.2021.11.245

Karaulova, T., Andronnikov, K., Mahmood, K., & Shevtshenko, E. (2019). Lean automation for low-volume manufacturing environment. In B. Katalinic (Ed.), *Proceedings of the 30th DAAAM international symposium* (pp. 0059–0068). DAAAM International.

Kaur, J., & Singh, W. (2022). Tools, techniques, datasets and application areas for object detection in an image: A review. *Multimedia Tools and Applications, 81*(27), 38297–3835. https://doi.org/10.1007/s11042-022-13153-y

Kleindienst, M., & Ramsauer, C. (2015). Der Beitrag von Lernfabriken zu Industrie 4.0-Ein Baustein zur vierten industriellen Revolution bei kleinen und mittelständischen Unternehmen. *Industrie-Management, 3*, 41–44.

Li, J., Gu, J., Huang, Z., & Wen, J. (2019). Application research of improved YOLO V3 algorithm in PCB electronic component detection. *Applied Sciences, 9*(18), 375. https://doi.org/10.3390/app9183750

Lin, T. Y., Goyal, P., Girshick, R., He, K., & Dollár, P. (2017). Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision* (pp. 2980–2988).

Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P, Ramanan, D., et al. (2014). Microsoft coco: Common objects in context. In *Computer vision–ECCV 2014: 13th European conference*, Zurich, Switzerland, September 6–12, 2014, Proceedings, Part V 13 (pp. 740–755). Springer.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., et al. (2016). Ssd: Single shot multibox detector. In *Computer vision–ECCV 2016: 14th European conference*, Amsterdam, The Netherlands, October 11–14, Proceedings, Part I 14 (pp. 21–37). Springer.

Lowe, D. G. (1999). Object recognition from local scale-invariant features. In *Proceedings of the seventh IEEE international conference on computer vision* (Vol. 2, pp. 1150–1157). IEEE.

Malisiewicz, T., Gupta, A., & Efros, A. A. (2011). Ensemble of exemplar-svms for object detection and beyond. In *2011 international conference on computer vision* (pp. 89–96). IEEE.

Mo, Z., Chen, L., & You, W. (2019). Identification and detection of automotive door panel solder joints based on YOLO. In *Chinese control and decision conference (CCDC)* (pp. 5956–5960). IEEE.

Müller, R., Vette-Steinkamp, M., & Kanso, A. (2019). Position and orientation calibration of a 2D laser line sensor using closed-form least-squares solution. *IFAC-PapersOnLine, 52*(13), 689–694. https://doi.org/10.1016/j.ifacol.2019.11.136

Park, S. S., Tran, V. T., & Lee, D. E. (2021). Application of various YOLO models for computer vision-based real-time pothole detection. *Applied Sciences, 11*(23), 1122. https://doi.org/10.3390/app112311229

Pierleoni, P., Belli, A., Palma, L., & Sabbatini, L. (2020). A versatile machine vision algorithm for real-time counting manually assembled pieces. *Journal of Imaging, 6*(6), 4. https://doi.org/10.3390/jimaging6060048

Redmon, J., & Farhadi, A. (2018). Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767.

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 779–788).

Redmon, J., & Farhadi, A. (2017). YOLO9000: Better, faster, stronger. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 7263–7271).

Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in Neural Information Processing Systems, 28*.

Ren, Z., Fang, F., Yan, N., & Wu, Y. (2022). State of the art in defect detection based on machine vision. *International Journal of Precision Engineering and Manufacturing-Green Technology, 9*(2), 661–69. https://doi.org/10.1007/s40684-021-00343-6

Tahmina, T., Garcia, M., Geng, Z., & Bidanda, B. (2022). A survey of smart manufacturing for high-mix low-volume production in defense and aerospace industries. In: *International conference on flexible automation and intelligent manufacturing* (p. 237–245). Springer.

Terven, J., Córdova-Esparza, D. M., & Romero-González, J. A. (2023). A comprehensive review of YOLO architectures in computer vision: From YOLOv1 to YOLOv8 and YOLO-NAS. *Machine Learning and Knowledge Extraction, 5*(4), 1680–1716. https://doi.org/10.3390/make5040083

Tkachenko, M., Malyuk, M., Holmanyuk, A., & Liubimov, N. (2020) Label studio: Data labeling software. Retrieved from https://github.com/heartexlabs/label-studio

Transeth, A. A., Stepanov, A., Linnerud, Å. S., Ening, K., & Gjerstad, T. (2020). Competitive high variance, low volume manufacturing with robot manipulators. In *3rd international symposium on small-scale intelligent manufacturing systems (SIMS)* (pp. 1–7). IEEE.

Vaidya, S., Ambad, P., & Bhosle, S. (2018). Industry 4.0–a glimpse. *Procedia Manufacturing, 20*, 233–238. https://doi.org/10.1016/j.promfg.2018.02.034

Viola, P., & Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. In *Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition* (Vol. 1, pp. I–I). IEEE.

Viola, P., & Jones, M. J. (2004). Robust real-time face detection. *International Journal of Computer Vision, 57*, 137–154. https://doi.org/10.1023/B:VISI.0000013087.49260.fb

Yi, L., Siedler, C., Kinkel, Y., Glatt, M., Kölsch, P., & Aurich, J. C. (2021). Object detection in factory based on deep learning approach. *Procedia CIRP, 104*, 1029–103. https://doi.org/10.1016/j.procir.2021.11.173

Zhang, S., Wen, L., Bian, X., Lei, Z., & Li, S. Z. (2018). Single-shot refinement neural network for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4203–4212).

Zhang, Z. (2000). A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 22*(11), 1330–1334. https://doi.org/10.1109/34.888718

Zhao, Q., Sheng, T., Wang, Y., Tang, Z., Chen, Y., Cai, L., et al. (2019). M2det: A single-shot object detector based on multi-level feature pyramid network. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 33, pp. 9259–9266).

Zheng, X., Chen, J., Wang, H., Zheng, S., & Kong, Y. (2021). A deep learning-based approach for the automated surface inspection of copper clad laminate images. *Applied Intelligence, 51*, 1262–1279. https://doi.org/10.1007/s10489-020-01877-z

Zheng, Z., Wang, P., Liu, W., Li, J., Ye, R., & Ren, D. (2019) Distance-IoU loss: Faster and better learning for bounding box regression. Retrieved from http://arxiv.org/abs/1911.08287

Zhong, R. Y., Xu, X., Klotz, E., & Newman, S. T. (2017). Intelligent manufacturing in the context of industry 4.0: A review. *Engineering, 3*(5), 616–63. https://doi.org/10.1016/J.ENG.2017.05.015

Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., & Torralba, A. (2017). Scene parsing through ade20k dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 633–641).

Zou, Z., Chen, K., Shi, Z., Guo, Y., & Ye, J. (2023). Object detection in 20 years: A survey. *Proceedings of the IEEE, 111*(3), 257–276. https://doi.org/10.1109/JPROC.2023.3238524