



## Real-time detection of plastic part surface defects using deep learning-based object detection model

Miraç Tuba Çelik <sup>a,\*</sup>, Seher Arslankaya <sup>b</sup>, Aytaç Yıldız <sup>a</sup>

<sup>a</sup> Department of Industrial Engineering, Bursa Technical University, Bursa, 16310, Turkey

<sup>b</sup> Department of Industrial Engineering, Sakarya University, Sakarya, 54187, Turkey

### ARTICLE INFO

**Keywords:**

Deep learning  
Quality control  
Defect detection  
Artificial intelligence  
You-Only-Look-Once version 8

### ABSTRACT

In this study, it was aimed to detect defects in plastic parts produced in a company operating in the automotive sub-industry using the YOLOv8 object detection model. The defect types seen in plastic parts were evaluated with the help of Pareto analysis, and scratches, stains and shine were selected as the most common defect types, and data on the three defect types were collected. YOLOv8 models were trained using faulty part images. As a result of the training, the highest mean average precision value of 0.990 was obtained in the YOLOv8s model, and the shortest training time was obtained in the YOLOv8n model. In the YOLOv8s model, which gave the highest mAP value, hyperparameter adjustment was made according to the batch size and learning rate values. The testing phase was carried out with the hyperparameter values that gave the best results and the mAP value was obtained as 0.902.

### 1 Introduction

The rapid increase in the population in the world, technological developments, and the continuous growth trend increase the competition between businesses [1]. Intense competition forces businesses to both keep up with developments and be open to changes. Meeting customer demands and needs in the best possible way is also important for businesses to compete [2]. The most important key for businesses to gain competitive advantage is quality [3]. Businesses that produce quality products/services increase their performance and gain a greater share of the market by providing a competitive advantage [4]. The purpose of businesses to produce quality products is to ensure that they can maintain their assets and resources rather than making more profit. The aim of businesses to increase profitability by producing products at less cost is achieved through quality studies [5]. Definitions have been made for quality by many institutions. American Society for Quality Control (ASQC) defines quality as “all the characteristics of a service or product that reveal their ability to meet the requested needs”. The Japanese Industrial Standards Committee (JISC) defines quality as “the production system that produces the service or product economically and responds to consumer demands” [6]. The International Standard Organization (ISO) defines quality as “the sum of the characteristics of the product or service based on its ability to meet specified or potential

requirements.”. When we look at other definitions of quality in the literature, there are definitions such as: “Quality is fitness for use” [7], “quality is compliance with customer expectations” [8] and “quality is meeting consumer demands” [9]. When defining quality, it is not right to only talk about the quality of goods or products. At the same time, service quality should also be included in this definition [2]. Therefore, quality, in a broad sense, is the system that produces and services in accordance with customer needs based on the intended use of the product, ensures effective and better use of resources, and provides a positive impact when companies fulfill their duties [10]. Purposes of use of the product in the description; It includes elements such as use, ease of maintenance, cost, compliance with environmental conditions, and security [10].

The reason why businesses attach importance to quality is that, thanks to technological developments, the rapid change process affects the diversity of products and services and commercial boundaries disappear. The fact that all these changes and developments cannot be prevented paves the way for the formation of a competitive environment. In this competitive environment, businesses need to attach importance to quality in order to survive and increase their share in the market [11]. It is also possible to talk about quality control at every step where the concept of quality is involved. Quality control is the combination of operations at all stages of production in order to produce at the

\* Corresponding author.

E-mail addresses: [tuba.celik@btu.edu.tr](mailto:tuba.celik@btu.edu.tr) (M.T. Çelik), [aseher@sakarya.edu.tr](mailto:aseher@sakarya.edu.tr) (S. Arslankaya), [aytac.yildiz@btu.edu.tr](mailto:aytac.yildiz@btu.edu.tr) (A. Yıldız).

most economical level and to maintain and improve quality, provided that the customer's satisfaction level is kept at a high level [12,13]. ISO defines quality control in two ways: In the broadest sense, it is the sum of the processes carried out to maintain and improve quality and to make production at the lowest cost that will satisfy the customer. In a narrow sense, it is the process of checking and approving the product's compliance with standards [14]. Quality control is a business function that is given importance to all employees, from the general manager to the workers in production, and is seen at all stages of production [15]. The main purpose of quality control is to prevent poor quality by detecting quality errors in production and trying to minimize or eliminate these errors [2,14].

While carrying out quality control activities, businesses need to create quality policies and take various factors into consideration when creating policies. These factors are known as the 9Ms and greatly affect quality control. These; Money, machinery and equipment, market, motivation, people, modern information methods, materials, management and product parameters are the creation factors [15]. It is very important to determine the error in advance in the quality control process. Errors that cannot be detected at the initial stage progress throughout the production line, which can cause irreversible problems. Once the error is identified, the personnel must be informed and the cause of the error must be investigated and the problem must be resolved. These processes should be seen as a chain and each step should be implemented [16]. After the final product is obtained, it is checked whether the product complies with quality control standards. This process is called the inspection process. The quality control carried out during the inspection process no longer has an impact on the product. That is, it is not an effective stage for improving the quality of the product. In the inspection process, the final product is divided into two as defective or intact, and a rejection or acceptance result is reached [2]. Quality control is of great importance in the automotive industry, as in all sectors. Since the automotive industry has a very important position in the world and contributes greatly to the country's economy in Turkey, as in all other countries of the world, it is important to implement quality control processes effectively. Quality control is of great importance in manufacturing sectors such as automotive, food and glass as it positively affects productivity, market share, competition and product quality. Compared to other sectors, it is known that product defects in the automotive industry will cause huge problems and especially result in loss of life and property. In order to solve this problem, quality control procedures must be applied both in many processes during production and after the final product is obtained.

In this study, defects in plastic parts produced in a company operating in the automotive sub-industry were detected using the YOLOv8 object detection model. The obtained data were trained on all YOLOv8 models ("n-nano", "s-small", "m-medium", "l-large", "x-xlarge") and hyperparameter adjustment was made by considering the model that gave the best results. The model was tested with the best hyperparameter setting and a model recommendation was made.

In the second part of the study, defects in products and defect detection are explained and literature studies are summarized. In the third section, the materials and methods used in the study are explained. In the fourth section, the results of the method determined using the data set are evaluated and discussed. The fifth section includes the results.

## 2. Literature research

It is not possible for the products produced in the companies where the production is made to be at the targeted quality level for many reasons. Products produced in this way are defined as broken or defective products because they contain errors. Although the concepts of broken, defective and faulty products seem to be the same, they are defined as products that have some differences [17]. Turkish Language Association (TDK) defines fault as "a mistake made unknowingly or unintentionally". Defective is defined by TDK as "knowingly or

unknowingly not doing a job properly, unfavorable situation, deficiency" [18]. According to the ISO 9001:2015 Quality Standard (2015), a defect is defined as "a situation where a condition related to a specified or intended use is not fulfilled". From a quality perspective, in the broadest sense, a defect can be defined as "an irregularity or deficiency that eliminates or reduces the benefit it provides." The concepts of fault and defect resemble each other, but any action that does not comply with the determined method during production is a fault, and the result of this fault is a defective product. For example, it is a fault for the operator to make adjustments that go beyond the specified product standard, and as a result of this faulty, a product that cannot fulfill its duty is obtained and this situation appears as a defect. Although the resulting product is often described as faulty, it is clearly seen that this is not conceptually correct [17]. Faulty is the concept that forms the center of all quality control activities [19]. When viewed from another perspective, it is a relative concept that varies according to tolerances. Since it is not possible to completely eliminate faults, the product is produced with a margin of error that does not affect the product in any way. This margin of error is expressed as tolerance. Specified tolerances may vary for various reasons. That is, while a specification requirement may be considered faulty for one product, it may not be faulty for another product. For this reason, it is possible to say that faulty is a relative concept [17]. Although businesses try to produce products with zero fault, this is not that easy. There is a certain rate of faulty production in every business where production is carried out. As a result of this ratio, an undesirable defective product is formed. It is possible to group these faults into four general classes. These are size differences, color differences, superficial differences and pattern differences. Dimensional differences are the differences seen in the dimensions of the product on the three-dimensional axis. Color differences are the changes seen in the features of different parts of the product. Superficial differences include scratches, cracks, roughness, burrs, etc. on the product. is to occur. Finally, pattern differences are the differences in the shapes on the product [20]. Various defects occur in products due to various reasons during production. While holes, tears, color changes and pattern faults are seen in leather and fabric products, breaks, bends and cracks occur in marble, steel and wooden products [21]. Similarly, defects such as scratches on the glass surface, air bubbles inside the glass or on the glass surface, glass breaks and cracks occur [22,23]. One of the developing branches of the forest products industry is the particleboard industry. Frequently encountered defects in particle board production; corner fractures, bumps, cracks and pit formations, slippage of the material after coating, and discoloration [20]. In sectors where parts are manufactured using the injection system, faults occur due to improper raw material storage, incorrect packaging and various reasons throughout production. As a result of these faults, defective products occur. These defects; The formation of burn marks on the product, the formation of flow marks on the product, the appearance of long joining marks on the product surface, the presence of deficiencies in the product, the formation of burrs and the formation of depressions in the product [24]. If the mentioned defects go undetected and reach the customer, it has negative effects on both the business and the customer. Therefore, it is very important to detect these defects using various methods to eliminate the human factor. Studies on defect detection based on deep learning methods in the literature are as follows. Birlutiu et al. [25] presented an automatic defect management system based on machine learning and computer vision that detects and quantifies defects in porcelain products through a convolutional neural network. In the study, logistic regression (Logistic Regression – LR), linear discriminant analysis (LDA), k-nearest neighbors (KNN), CART decision tree (Classification and Regression Tree – CART), naive bayes, Support Vector Machine (SVM), Random Forest (RF) and Convolutional Neural Networks (CNN) algorithms were used and it was stated that the CNN method gave the best result for defect detection with 0.89. Cheng and Wang [26] proposed the Faster R-CNN model to detect 4 types of defects in sewer pipes. The model was trained using a total of 3000 images and

was evaluated in terms of many performance metrics. Additionally, in the study, a new model was created and various hyperparameter settings were made. As a result of the study, defects were detected with a mAP value of 0.83. Wei et al. [27] discussed the traditional fastener positioning method based on image processing for VGG16 fastener defect detection and proposed a defect detection and identification method using Dense-SIFT features. VGG16 fastener is trained for defect detection. As a result of the training, it was concluded that it is possible to detect defects in fasteners with CNN. Finally, Faster R-CNN was used for defect detection. When the results were compared, it was seen that the best result was obtained with Faster R-CNN. Yang et al. [28] proposed a small part defect detection method to achieve a real-time small part defect detection system and solve the problems of manually adjusting the conveyor speed and industrial camera parameters in defect detection for factory products. This proposed system allows detecting defects such as length and point size errors, tightening and deformation of a 0.8 cm knitting needle. In the study, firstly, the effects of the properties of small parts and the environmental parameters of the defect detection system on the stability of the system were evaluated and a correlation model was established between the detection ability coefficient of the part system and the movement speed of the conveyor. Then, a defect detection algorithm based on a single short detector network and deep learning for small parts is proposed. Finally, to solve the problem of missed detections, the industrial real-time detection platform is combined with the missed detection algorithm for mechanical parts based on intermediate variables. Du et al. [29] proposed a deep learning-based X-ray image defect detection system for defect detection of automobile cast aluminum parts. Feature pyramid network algorithm was used to train the model by performing classification. At the end of the study, it was determined that the feature pyramid network algorithm gave better results than Faster R-CNN. Ren et al. [30] used a three-stage deep learning algorithm to detect defects using real X-ray images obtained from the production line. The accuracy of the proposed defect detection method was obtained as 90%. Liu et al. [31] aimed to perform automatic defect detection in injection molding products that can work with small data sets using CNN-based transfer learning. The proposed method was tested with 200 images per category and the detection accuracy was found to be 99%. Shen et al. [32], a physics-informed deep learning approach for bearing fault detection was proposed. This proposed approach consists of a simple threshold model and a deep CNN model. The study was validated using data from 18 roller bearings and ball bearings of the Case Western Reserve University (CWRU) Bearing Data Center. Mery [33] compared the performances of eight different modern object detection models (including YOLOv3, YOLOv5, RetinaNet, EfficientDet) using GDXray images. In the experiments, it was seen that YOLO-based detectors showed the best performance. One of the models, YOLOv5s, was trained in 2.5 h. In the test data set, the average sensitivity was found to be 0.90 and the F1 score was 0.91. Yang et al. [34] used the YOLOv5 algorithm and the Faster R-CNN algorithm for steel pipe weld defect detection and compared the two algorithms. As a result of the study, the mAP value of the YOLOv5x model was found to be 98.7% and the mAP value of the Faster R-CNN model was found to be 78.1%. Xu et al. [35] used SSD and Faster R-CNN methods to detect paint film defects. A total of 2000 images were used in the study as training and testing. As a result of the study, the precision value for the SSD method was found to be 96.96% and the precision value for the Faster R-CNN method was 93.07. At the end of the study, the authors stated that the proposed framework can be easily applied to other industrial application areas such as detection of wood defects, detection of surface defects of parts and detection of welding defects. Zhang et al. [36] proposed an image-based fusion method for defect detection of the spiral cutting edge of the milling cutter. The method first uses the developed YOLOv3-tiny to extract the target cutting edge region and then applies conventional image processing techniques to detect defects. The precision value of the method proposed for the study was found to be 98.86%, the precision value of the traditional image processing method (edge

detection method) was 94%, and the precision value of the deep learning method (R-FCN) was 74.44%. Medak et al. [37] proposed a new deep learning architecture specifically designed for defect detection using UT images in their study. They proposed a feature extractor that improved the sensitivity and efficiency of the detector and modified the detection head to improve the detection of objects with extreme aspect ratios common in UT images. The proposed architecture outperforms the previous state-of-the-art method by 1.7% ( $512 \times 512$  pixel input resolution) and 2.7% ( $384 \times 384$  pixel input resolution) while also significantly reducing the inference time. Yilmazer et al. [38] proposed a new method using the Mask R-CNN deep neural network architecture to detect healthy sleepers and various faults occurring on the sleepers. Model training and testing of the trained model were provided by using real railway images labeled with a total of four different class labels. As a result of the test, the accuracy of the model was determined as 95%. Joshi et al. [39] presented a deep learning-based architecture for automatic detection and segmentation of cracks on the concrete-based material surface. A total of 300 surface crack images were used in the study. The average precision recorded for the trained model is 74.156% and 93.445% at 0.7 and 0.5 IoU thresholds for crack segmentation, respectively. Shafi et al. [40] proposed an image processing-based automatic defect detection and classification approach for cylindrical hollow surfaces. The proposed system uses single shot detector (SSD) and light-weight deep convolutional neural network. The proposed model was found useful for real-time defect classification on hollow cylindrical surfaces with 92% accuracy. Chen et al. [41] proposed a low-contrast defect detection method based on deep learning to detect defects occurring during 3D printing of ceramic parts. The proposed ECANet-Mobilenet SSD network model showed that the prediction accuracy of crack defect reached 94.35% and the prediction accuracy of bulge defect reached 96.72%. Zhang et al. [42] proposed a new deep learning method to detect various defects based on only small data sets. The proposed method is based on deep random chain combined with Faster R-CNN. In the proposed method, the accuracy value at 1000 epochs was found to be 84.23%, the accuracy value at 3000 epochs was 97.04 and the accuracy value at 7000 epochs was found to be 99.07%. Gou et al. [43] proposed an online intelligent detection method for cable pitch defects in their study. The method uses a multi-sensor fusion image acquisition platform consisting of a Doppler laser accelerometer and an industrial line-scan CCD. To verify the accuracy and stability of the online detection method, 6 types of optical cables were prepared for experimental testing. Experimental results show that the detection results meet the accuracy requirements.

Surface defect studies conducted with the YOLO object detection model in the literature are as follows. Zhang et al. [44] proposed a method based on YOLOv2 to detect yarn-dyed fabric defects. In the study, they used 276 fabric defect images and classified the defects as missing thread defects, knot defects and holes. They used YOLO9000, YOLO-VOC and Tiny YOLO algorithms to create fabric defect detection models. They used 72% of the labeled images as the training set, 10% as the test set and 18% as the validation set. With the YOLO-VOC algorithm, they found the sensitivity (recall) value to be 88%, the precision value (precision) to be 86%, the mAP value to be 78.6%, the average IoU value to be 69.45%, and the FPS value to be 23. Jing et al. [45] proposed a real-time fabric defect detection method based on the developed YOLOv3 model to increase the defect detection rate and improve fabric product quality. Firstly, size clustering of target frames was performed by combining fabric defect size and k-means algorithm. Then, low-level features were combined with high-level features and a YOLO detection layer was added to feature maps of different sizes. A total of six classes were used in the study. These; slab error, foreign matter error, folding error, stain error, ribbon thread error and hole error. Of the 6300 images used in the study, 4420 were reserved for training and 1880 for testing. It was concluded that the developed method detects fabric defects better than YOLOv3. Opara et al. [46] developed a method using YOLOv3 to detect defects in asphalt pavements. Four types of defects were used in

the study. Pavement images obtained from National Route 4 were used to test the accuracy of the developed method. As a result of the study, the precision value was found to be 70% and the average IoU value was 50.39%. Yang et al. [47] tried to detect chip defects using the YOLOv3 deep learning method. A total of eight different defect types were used in the study. As a result of the study, optimization methods and hyper-parameter selection were optimized and the final mAP value was found to be 86.36%. Kou et al. [48] developed a defect detection model based on YOLOv3 to detect defects on the surface of steel strip. Anchorless feature selection mechanism was used for the model. At the same time, specially designed dense convolution blocks are included in the model. The mAP value of the proposed model was found to be 71.3% in the GC10-DET dataset and 72.2% in the NEUDET dataset. Wan et al. [49] proposed a detection method based on YOLOv5s developed to detect defects in textured ceramic tiles. In the backbone network, the network layer was deepened and the attention mechanism CBAM was added, a small-scale detection layer was added, and the model was upgraded from a three-output prediction layer to a four-output prediction model. In addition, the network fusion feature in the neck network is strengthened and the original convolution is replaced with depth-separable convolution. When the proposed model was compared with the original YOLOv5s model, it was observed that the accuracy increased by 6.09%, the F1 value increased by 6.53% and the mAP value increased by 6.15%. As a result, the accuracy of the proposed model was found to be 94.27%, the F1 value was 89.95% and the mAP value was 92.80%. Güçlü et al. [50] aimed to detect defects on steel surfaces using computer vision and YOLOv7. The study focuses on post-production defect inspection of steel wires in manufacturing workplaces. To achieve fast detection capability, the inference speed of embedded devices is increased by using TensorRT. Additionally, data augmentation algorithm was used to alleviate the limited data scaling problem. The performance of YOLOv7 was compared with YOLOv5 and high speed and accuracy were achieved by using YOLOv7 for error detection. Li et al. [51] proposed a model that improves the feature fusion module of the YOLOx model by adding an efficient channel to provide higher accuracy and faster detection in detecting defects on wooden surfaces. Four types of defects were used in the study. Test results showed that the mAP value of the improved EAE-YOLOX model was 96.42% and it detected four defects with a detection speed of 46.7 FPS; showed that the lightweight EAE-YOLOX-tiny model had a mAP value of 94.92% and detected four defects with a detection rate of 52.09 FPS. Li et al. [52] proposed the YOLOv4 object detection model developed to detect surface defects of steel strips. In the proposed model, the attention mechanism is embedded in the backbone network structure, and the path aggregation network is transformed into a customized receptive field block structure that strengthens the feature extraction functionality of the network model. As a result of the results obtained, the average sensitivity value increased by 3.87% compared to the original YOLOv4 model, reaching 85.41%. Ying et al. [53] proposed a method based on the YOLOv5 object detection model for the detection of steel wire braided hose defects. In the model, junction boxes were optimized with the k-means++ algorithm, focal loss was used to reduce the effect of sample imbalance, and prediction heads were changed. The results obtained from the study show that the accuracy of the modified YOLOv5s model reached 92.2%. Zhou et al. [54] proposed a detection method based on YOLOv8-2d and BAGAN defect detection model to detect defects on the gear surface. The proposed method accurately locates and classifies multi-scale defects. The results obtained show that the average sensitivity (mAP50) of YOLOv8-2d is 97.3%. As a result of the test performed with a total of 100 defective images, the accuracy (Pr) and sensitivity (Re) of detecting defects were found to be 94.51% and 90.53%. Zhang et al. [55] proposed a model based on the YOLOv5s model to detect surface defects of wind turbines. In the proposed model, the MobileNetV3 network replaces the backbone network in YOLOv5. The neck network is implemented as a weighted two-way feature pyramid network. Convolution block attention module was used as the field and channel attention mechanism.

After the changes made, the new model was compared with the YOLOv5s model and it was observed that the mAP value increased by 5.51% and the detection time decreased by 10.79 frames per second. Guo et al. [56] proposed a method based on the modified YOLO network for defect detection using industrial neutron radiographic images (NRI). A total of 6336 radiographic images were used in the study. To detect small size defects, adaptive spatial feature fusion (ASFF) and convolutional block attention module (CBAM) are introduced into the modified YOLO network. As a result of the study, the proposed method provided an average accuracy of 98.1%. Wang et al. [57] proposed the YOLOv5-CD algorithm to detect steel surface defects. They added a coordination attention mechanism to the backbone network of the proposed algorithm and used a head detector with a separated head component. As a result of the study, the mAP value increased to 80.6%, 5.8% higher than the original YOLOv5 method. Li et al. [58] developed a new deep learning-based automatic defect detection model based on the YOLOv4 object detection model, which provides fast and accurate defect detection for wire and arc additive manufacturing (WAAM). As a result of the developed model, the mAP value was found to be 94.5%. Zhu et al. [59], YOLOv7 algorithm was used to classify surface defects and obtain location information in belt grinding. As a result of the study, it was seen that YOLOv7 performance reached 90.7% mAP. Zhao et al. [60] proposed a model called RDD-YOLO based on YOLOv5 to detect defects on steel surfaces. The backbone component of the proposed model consists of Res2Net blocks. A dual-feature pyramid network is designed for the neck part. A decoupled head was used to provide greater detection sensitivity. As a result of the study, the RDD-YOLO model reached 81.1% mAP value in the NEU-DET data set and 75.2% mAP value in the GC10-DET data set. Li et al. [61] developed the HM-YOLOv5 network for defect detection of hot-pressed light guide plates. Looking at the results of the developed network, it was seen that the mAP value was 98.9% and it provided an improvement of 2.7% and 2.0%, respectively, especially for white point and dark line defects.

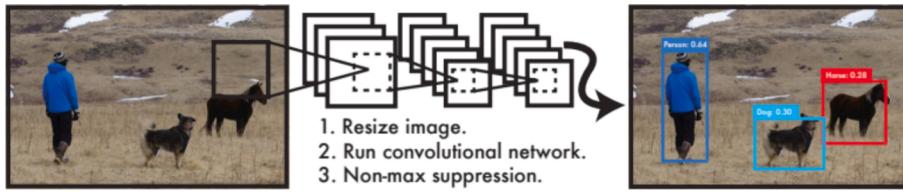
Since the real-time detection of surface defects is a typical application of machine learning algorithms, it is possible to say that some immune-based machine learning algorithms in the literature also have a good reference value for this task such as, Non-linear statistical image watermark detector [62], Deep soft threshold feature separation network for infrared handprint identity recognition and time estimation [63], Floating pollutant image target extraction algorithm based on immune extremum region [64], Immune coordination deep network for hand heat trace extraction [65].

### 3. Material and method

#### 3.1. YOLO object detection method

YOLO, Redmon et al. [66] is a popular real-time object detection algorithm. The new architecture developed treats object detection as a single regression problem, from image pixels to class probabilities and bounding box coordinates. A single convolutional network predicts many bounding boxes and their corresponding class probabilities simultaneously. YOLO consists of a single convolutional end-to-end trainable module that learns from input images during training. This combined model has several advantages over traditional object detection methods [66]. During training and testing, YOLO sees the entire image and thus implicitly encodes contextual information about its appearance in addition to classes. Thanks to this feature, the probability of making errors by confusing background patches in the image with objects is lower compared to Faster R-CNN. By combining separate components of object detection in a single neural network, YOLO is able to simultaneously predict all bounding boxes in all classes for an image. Fig. 1 is an example of the YOLO detection system [66].

YOLO first divides the entire image into grids of SxS size. Although the grid cells are considered as 7x7 in the original YOLO, they can also be selected in different shapes such as 3x3, 5x5 and 19x19. If the center



**Fig. 1.** YOLO detection system [66].

of an object falls in a grid cell, that grid cell is responsible for detecting that object. Each grid cell predicts  $B$  bounding boxes and confidence scores for those boxes. These confidence scores reflect how confident the model is that the box contains an object and how accurately it thinks the box predicts it. If there is no object in the grid cell, the confidence score is zero. Otherwise, the confidence score is desired to be equal to the IoU value between the predicted bounding box and the ground truth. The formula used to calculate the confidence score (CS) is given in equation (3.1) [66].

$$CS = \Pr(\text{object}) * IoU_{\text{pred}}^{\text{truth}} \quad (3.1)$$

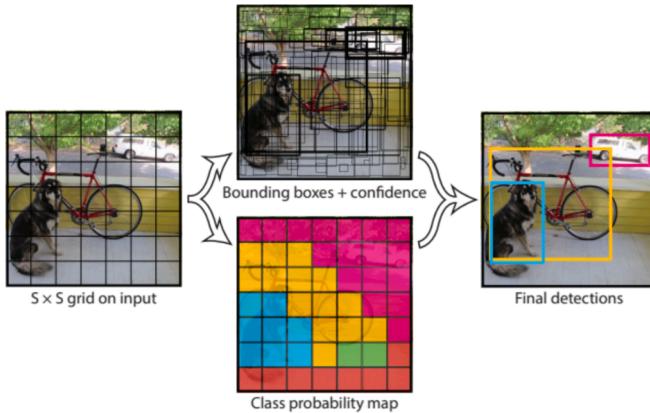
$\Pr(\text{object})$ , the probability of finding the object in the grid,  $IoU_{\text{pred}}^{\text{truth}}$ , represents the intersection value over the union between the bounding box and the ground-truth box.

Each bounding box has 5 components. These components;  $x$ ,  $y$ ,  $w$ ,  $h$  and confidence score. Here  $(x, y)$  represents the center of the bounding box relative to the boundaries of the grid cell,  $(w, h)$  represents the width and height. Width and height are estimated based on the entire image. The confidence estimate represents the IoU between the predicted box and any ground truth box. Each grid cell also estimates  $C$  conditional class probabilities as  $\Pr(\text{class}_i | \text{object})$ . Without depending on the number of  $B$  bounding boxes, only one class probability set is estimated per grid cell. So the probabilities depend on the grid cell containing the object. Equation (3.2) shows the product of conditional class probabilities and individual bin confidence estimates in the test part. This equation returns a class-specific confidence score for each bounding box. This score shows both the probability of that class being in the box and how well the predicted box fits the object [66].

$$\Pr(\text{class}_i | \text{object}) * \Pr(\text{object}) * IoU_{\text{pred}}^{\text{truth}} = \Pr(\text{class}_i) * IoU_{\text{pred}}^{\text{truth}} \quad (3.2)$$

In Fig. 2, the YOLO model divides the input into  $S \times S$  grid cells and estimates  $B$  bounding boxes, confidence of these boxes, and class  $C$  probabilities for each grid cell. These predictions are encoded as an  $S \times S \times (B * 5 + C)$  output tensor. The reason why  $B$  is multiplied by 5 is that it contains the  $(x, y, w, h, \text{trust})$  value for each box [66].

YOLO's architecture is influenced by GoogLeNet in image classification. The network consists of 24 convolution layers and two fully



**Fig. 2.** YOLO model [66].

connected layers. Instead of the starter modules used by GoogLeNet, Lin et al. [67], a  $1 \times 1$  reduction layer and a  $3 \times 3$  convolution layer were used [66]. The network structure is shown in Fig. 3 [68].

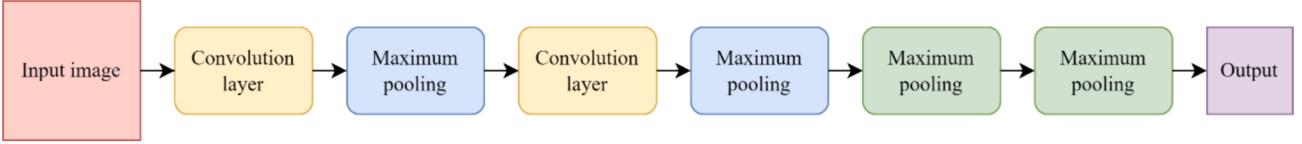
The last layer in the network structure estimates class probabilities and bounding box coordinates. The width and height of the bounding box are normalized according to the image width and height and brought to between 0–1. The bounding box  $x$  and  $y$  coordinates are parameterized so that the distances to a particular grid cell location are between 0–1. A linear activation function is used for the last layer and all other layers use the Leaky ReLU activation function. The Leaky ReLU activation function is shown in Equation (3.3) [66].

$$\phi(x) = \begin{cases} x, & \text{if } x > 0 \\ 0.1x, & \text{otherwise} \end{cases} \quad (3.3)$$

The  $x$  variable represents the outputs in the layers. Total squared error optimization is performed because it is easy to optimize the output of the model. Equal weight was given to localization error and classification error, which may not be ideal. Moreover, many grid cells in each image do not contain any objects. Due to this situation, the confidence scores of the cells are reset and often suppress the trend of cells containing objects. This causes model instability and negatively affects education. To solve this problem, two parameters named  $\lambda_{\text{coord}}$  and  $\lambda_{\text{noobj}}$  are used to increase the loss due to bounding box coordinate estimates and reduce the loss due to confidence estimates for boxes that do not contain objects. The parameters used were set to 5 and 0.5, respectively. At the same time, the total squared error weights errors in large bounding boxes and small bounding boxes equally. The error measurement should show that small deviations in large bounding boxes are less significant than in small boxes. This problem can be partially solved by estimating the square root of the bounding box width and height instead of the width and height directly. YOLO predicts multiple bounding boxes per grid cell. During training, only one bounding box predictor from each object is assigned, which will be responsible for predicting an object based on the prediction with the highest current IoU. This causes specialization among bounding box estimators, whereby each estimator becomes better at predicting certain sizes, aspect ratios, or classes of objects. The loss function optimized during training is given in Equation (3.4) [66].

$$\begin{aligned} & \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^B 1_{ij}^{\text{object}} [(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2] \\ & + \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^B 1_{ij}^{\text{object}} [(\sqrt{w_i} - \sqrt{\hat{w}_i})^2 + (\sqrt{h_i} - \sqrt{\hat{h}_i})^2] \\ & + \sum_{i=0}^{S^2} \sum_{j=0}^B 1_{ij}^{\text{object}} (C_i - \hat{C}_i)^2 \\ & + \lambda_{\text{noobj}} \sum_{i=0}^{S^2} \sum_{j=0}^B 1_{ij}^{\text{noobj}} (C_i - \hat{C}_i)^2 \\ & + \sum_{i=0}^{S^2} 1_i^{\text{object}} \sum_{\text{classes}} (p_i(c) - \hat{p}_i(c))^2 \end{aligned} \quad (3.4)$$

$1_i^{\text{object}}$  indicates whether the object appears in cell  $i$ , and  $1_{ij}^{\text{object}}$  indicates that the  $j$ th bounding box predictor in cell  $i$  is responsible for this prediction. The loss function only penalizes the classification error if there is an object in the relevant grid cell. Additionally, it penalizes the bounding box coordinate error when the estimator is responsible for the ground truth box [66].



**Fig. 3.** YOLO network structure [68].

**Fig. 4** shows the ranking of YOLO versions over time [69].

When **Fig. 4** is examined, it was decided to use the YOLOv8 model in the study because it is the latest version of the YOLO object detection model and when the studies in the literature are examined, it performs much better than other versions. In [Section 3.2](#), the YOLOv8 object detection model is explained in detail.

### 3.2. YOLOv8

YOLOv8, developed by Ultralytics, is an object detection model that has been proven to provide high speed and accuracy in practical applications and has eliminated many disadvantages in previous versions [70]. There are five models for identification, segmentation and classification, namely YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l and YOLOv8x, respectively [71]. YOLOv8's ability to recognize objects in real time is one of its most important features. The latest computer vision methods, such as pattern segmentation, which enables recognition of multiple elements in an image or video, are supported by YOLOv8 [72]. Additionally, it is more effective than previous versions of YOLO because it uses a larger feature map and a more effective convolutional network [73]. The inclusion of a new primary network, an innovative anchor-free detection head, and a dedicated loss function, in addition to its extensibility, have made YOLOv8 a sought-after option for a variety of image segmentation and object identification tasks. Compared to other YOLO versions, YOLOv8 performs the object detection task faster and with higher accuracy and provides a comprehensive framework for object detection, image classification and development of instance segmentation models. One of the key changes in YOLOv8, the anchor-less design, directly estimates the size and position of objects without relying on predetermined anchor boxes, thus providing more accurate detection of objects. The visualization of the junction box is shown in **Fig. 5** [74].

In YOLOv8, the main building block was changed, the C2f module was used instead of the C3 module, and the initial 6x6 transformation of the body was replaced with 3x3. The C2f module is shown in **Fig. 6** [74]. The C2f module used increases detection accuracy by combining high-level features with contextual information [69]. The designed new backbone network and new loss function have the potential to improve the performance of YOLOv8. The backbone network is the main feature extraction element of the neural network. It extracts important features that will enable discrimination of samples, recognition of objects thanks to input data, and downstream classification of images. The loss function measures how well the model performs during training. It compares the ground truth labels with the model's predictions and calculates a missing value to indicate how far they are from the ground truth. The main purpose here is to minimize the loss value [74].

In the output layer of YOLOv8, the sigmoid function is used as the activation function for the objectivity score, which represents the probability that the bounding box contains an object. The softmax

function is used for class probabilities, which represent the probabilities of objects belonging to each possible class [69]. The model has a user-friendly application programming interface that facilitates integration into different applications [72].

Two neural networks, Feature Pyramid Network (FPN) and Path Aggregation Network (PAN), have been added to YOLOv8 along with a new labeling tool that simplifies the annotation procedure. Different features such as shortcut tagging, customizable hotkeys, and automatic tagging are included in this new tagging tool, and thanks to all these features, it is easy to add tags to photos to train the model. In FPN, as the number of feature channels increases, the spatial resolution of the input image continuously decreases. As a result, a feature map is produced that can identify objects of various sizes and resolutions. The PAN architecture can use hop links to mix attributes from many network layers. As a result, the network can better collect information at multiple scales and resolutions, which is necessary for the full recognition of objects of different sizes and shapes [75].

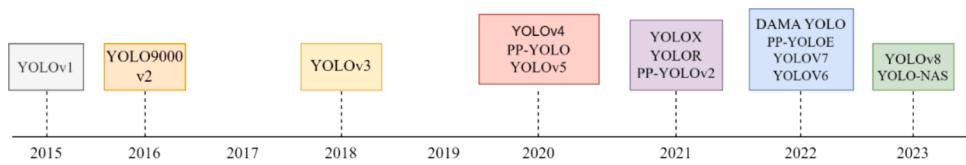
**Fig. 7** shows the comparison of YOLOv8 with other YOLO versions [70].

When **Fig. 7** is examined, when compared with YOLOv8, YOLOv5, YOLOv6 and YOLOv7 trained at 640 image resolution, it is seen that all variations of YOLOv8 provide better efficiency with a similar number of parameters. **Fig. 8** shows the mean average precision value of APs for the RF100 category, which is a dataset of 100 samples from the Roboflow universe, which is a pool of 100,000 datasets of YOLOv8 [76].

According to **Fig. 8**, it can be seen that YOLOv8 performs better than YOLOv5 and YOLOv7 for every RF100 category and has higher mAP values, especially in the video games and real world categories.

YOLOv8 scored 53.7 on a single model single scale on the COCO val2017 dataset for detection using a metric that evaluates average precision across various recall levels. This result is clearly ahead of other YOLO models in terms of speed and accuracy [70]. It has a high accuracy rate measured by COCO and Roboflow 100, it comes with many features that make it easy for developers, from an easy-to-use command-line interface (CLI) to a well-structured Python package, and it achieves strong accuracy in COCO. All these important advantages put YOLOv8 ahead of many models for image processing studies. For example, the YOLOv8m model (mid-size model) achieved 50.2% mAP when measured on the COCO. YOLOv8 scores significantly better than YOLOv5 when evaluated against Roboflow 100, a dataset that specifically evaluates model performance across a variety of task-specific domains. The reason why YOLOv8 is compared here with YOLOv5 rather than with another YOLO version is that the performance and metrics of YOLOv5 are closer to YOLOv8 [74]. **Fig. 9** shows the architecture of YOLOv8 [77].

When some recent studies with the YOLOv8 model are examined; Talaat and ZainEldin [71] in the development of smart fire detection system for smart cities, Li et al. [78] in UAV multi-target detection, Kim et al. [79] in drone detection, Ju and Cai [80] in pediatric wrist trauma



**Fig. 4.** YOLO versions over time [69].

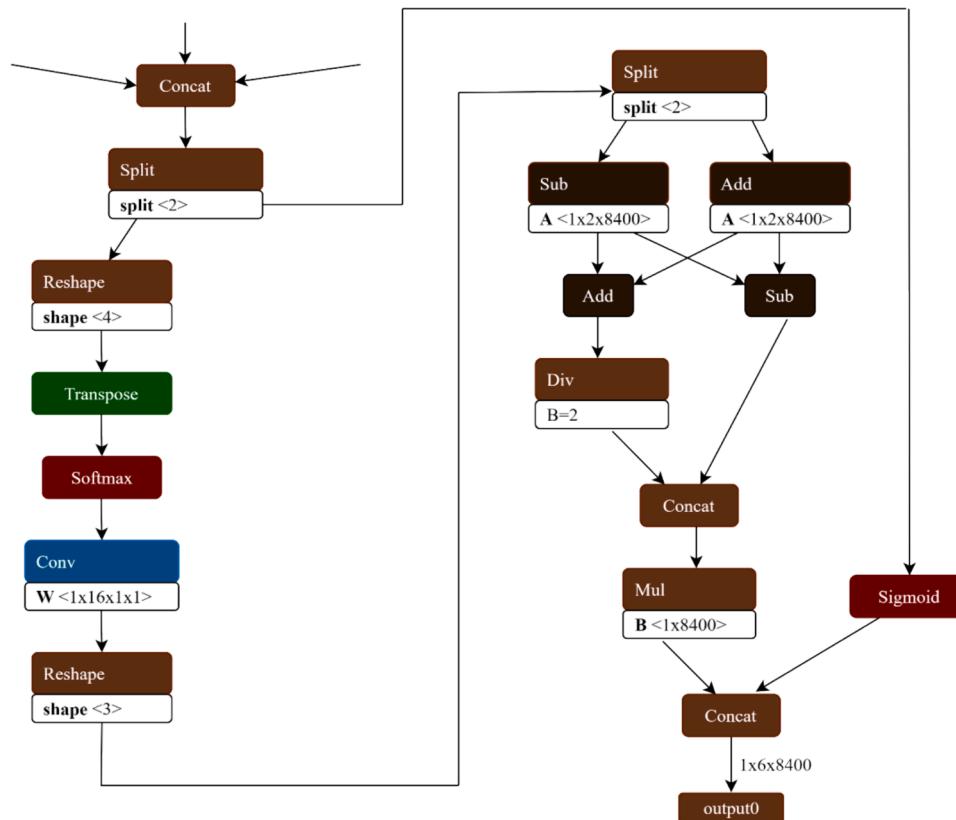


Fig. 5. YOLOv8 anchorless detection [74].

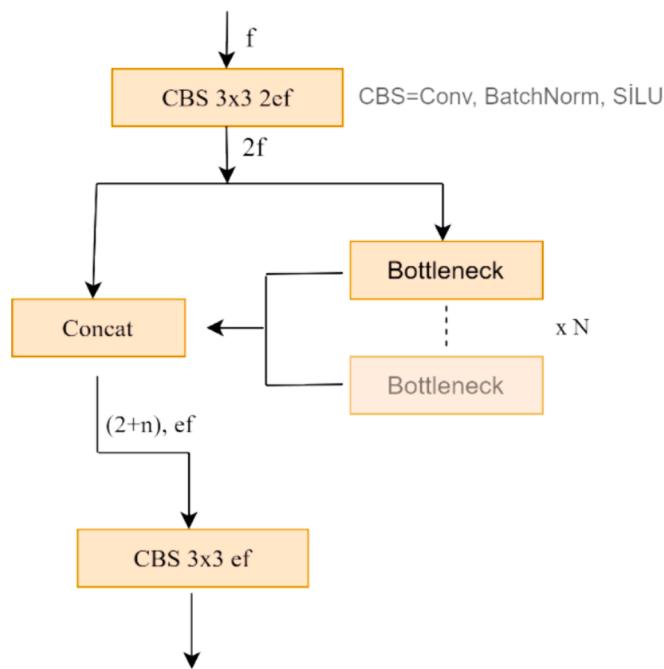


Fig. 6. New YOLOv8 C2f module [74].

x-ray images fracture detection, Xiao et al. [81] in determining fruit maturity, Sharma et al. [82] in parking time violation tracking, Tamang et al. [73] on accurate face mask classification to enhance COVID-19 safety, Yang et al. [83] in cow detection, Zhou et al. [54] in gear surface defect detection, Giakoumoglou et al. [84] to detect whiteflies in agriculture, Aboah et al. [72] for helmet detection, Yang et al. [85] in

tomato detection, Wang et al. [86] in detecting smoking behavior, Oh and lim [87] in detecting driving vehicles and brake light conditions, Akhtar et al. [88] in sediment detection and classification, Chabi Adjobo et al. [89] in detecting dermoscopic features, Li et al. [90] in tomato maturity detection, Zhang et al. [91] for detecting agricultural pests and diseases, Soylu and Soylu [92] for traffic sign detection in autonomous vehicle competition, Yue et al. [93] in classifying healthy and diseased tomato plants, Karna et al. [94] in 3D printer fault detection, Qadri et al. [95] for detection and segmentation of plant diseases, Le et al. [96] in detecting intact surgical instruments in laparoscopic surgery, Jia et al. [97] in detecting and recognizing forest fire using UAV images, Shashank et al. [98] to detect players in the game of cricket and Nandini et al. [99] used the YOLOv8 object detection model to detect the cricket ball. In addition to these studies, studies conducted in different fields in the literature based on the YOLOv8 object detection model and the performance values obtained are shown in Table 1.

According to the literature information given above, in the YOLOv8 object detection model, unlike other methods, the entire image is transmitted over the network only once and object detection is done in real time. With this feature, the method is very successful in producing both fast and accurate results. In addition to these, there are many strategies to reduce the computational demands of deep learning training for YOLOv8, which can help organizations lower their operational costs. These strategies include model simplification, pruning, hardware acceleration, cloud-based solutions, and effective data management. Model simplification can reduce the number of parameters and computations. The pruning technique removes unnecessary, non-informative weights and neurons from the model, thus reducing the complexity and computational costs during training and inference. The implementation of these strategies can help organizations manage costs while deploying advanced deep learning models and even improve performance. For these reasons, the YOLOv8 method was preferred in this study.

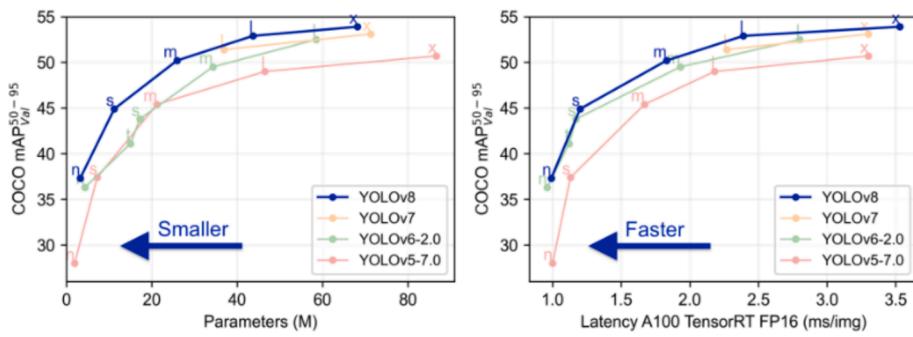


Fig. 7. Comparative analysis of YOLO models versions [70].

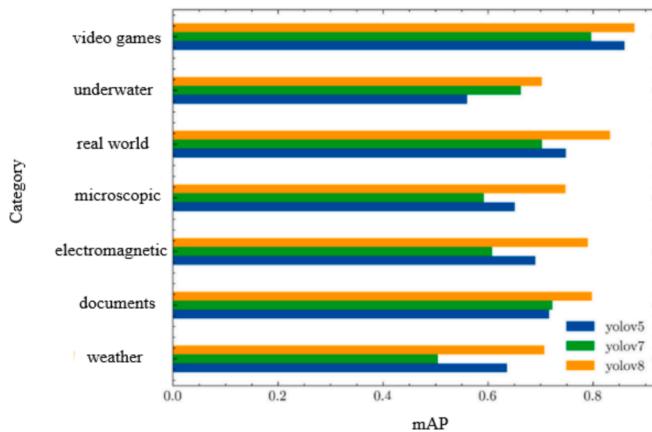


Fig. 8. Average mAP@.50 of YOLO versions by RF100 [76].

### 3.3. Performance metrics

World-accepted methods that serve as metrics to measure the performance and accuracy of the model in the object detection task are called evaluation metrics. Average precision (AP), mean average precision (mAP), Recall (Re), precision (Pr) and F1 are some of these metrics [110]. Evaluation metrics help determine how well a model detects and positions objects in images. Additionally, they help understand how the model handles false positives and false negatives. Thanks to these measurements, they are of great importance in evaluating and improving the performance of the model [111]. Leading competitions such as PASCAL VOC and MSCOCO provide predefined metrics to evaluate how different algorithms for object detection perform on datasets [112].

In order to see how accurately the bounding box detects after object detection, it is necessary to compare the  $IoU_p$  value, which shows the intersection value over the combination between the predicted bounding box and the ground-truth box, and the  $IoU_t$  value used as the threshold value.  $IoU_t$  is used as the threshold value after being calculated as in Fig. 10 [69]. This value is the minimum level required for the predicted bounding box to be considered a successful prediction. If  $IoU_p \geq IoU_t$ , the object is inside the bounding box [68]. The IoU score, ranging from 0 to 1, represents the ratio of intersection to union. A higher IoU indicates more accurate localization [113].

$$IoU_p = \frac{\text{area}(BB \cap GT)}{\text{area}(BB \cup GT)} \quad (3.5)$$

In Equation (3.5), BB represents the bounding box and GT represents the ground-truth box.  $IoU_p$  values above this threshold are considered correct predictions, and below this threshold are considered incorrect predictions. More precisely, predictions are classified as True Positives (TP), False Negatives (FN) and False Positives (FP). TP refers to the

accurate detection of the actually existing object on the image ( $IoU_p \geq IoU_t$ ). FP is the situation where an object that does not actually exist is detected on the image. Additionally, FP may occur due to an IoU score that is below the threshold value when the object is detected despite its presence ( $IoU_p < IoU_t$ ). It is also called localization error. FN indicates that an object that actually exists on the image cannot be detected. True negatives (TN) are not mentioned because it defines the situation in which empty boxes are correctly detected as "non-object". Thanks to the obtained TP, FP, FNs, the Pr and Re values of the model can be calculated [68].

Pr is the probability that the predicted bounding boxes match the ground-truth boxes, also called the positive predictive value. In other words, it is a measure that determines how accurately predictions are made. The mathematical expression used to calculate the Pr value is given in Equation (3.6):

$$Pr = \frac{TP}{TP + FP} \quad (3.6)$$

Re is the probability of correctly detecting ground-truth objects, also called the true positive rate. This metric is a measure that indicates the prediction success of the model. The mathematical expression used to calculate the Re value is given in Equation (3.7):

$$Re = \frac{TP}{TP + FN} \quad (3.7)$$

The F1 score, used as another performance metric, is a function of precision and recall that provides a measure to achieve a balance between precision and recall. A high F1 indicates that both precision and recall are high. A lower F1 score shows a greater imbalance between recall and precision. This value represents the harmonic mean of precision and recall values. The reason for having a harmonic average instead of a simple average is that extreme cases must be ignored. F1 score calculation is given in Equation (3.8):

$$F_1 = 2 \times \frac{Pr \times Re}{Pr + Re} \quad (3.8)$$

Another metric that summarizes both sensitivity and precision and provides a model-wide assessment is the Precision-Sensitivity curve. Since neither measure uses true negatives, this curve is a suitable metric for evaluating the model's performance on unbalanced datasets. The area under this curve gives the AP value, which is one of the most important performance metrics. AP serves as a metric to evaluate the performance of object detectors. A single number metric that encompasses both precision and sensitivity and summarizes the Precision-Sensitivity curve by averaging precision across precision values between 0 and 1. Thanks to this metric, AP values of each class are calculated separately. In other words, AP is calculated as the sum of precisions at each threshold according to increases in recall. IoU is also used when calculating the AP score. During AP calculation, the IoU score is used as a threshold. For instance, AP<sub>50</sub> is the average precision of predictions with an IoU greater than 0.5. AP value is obtained in

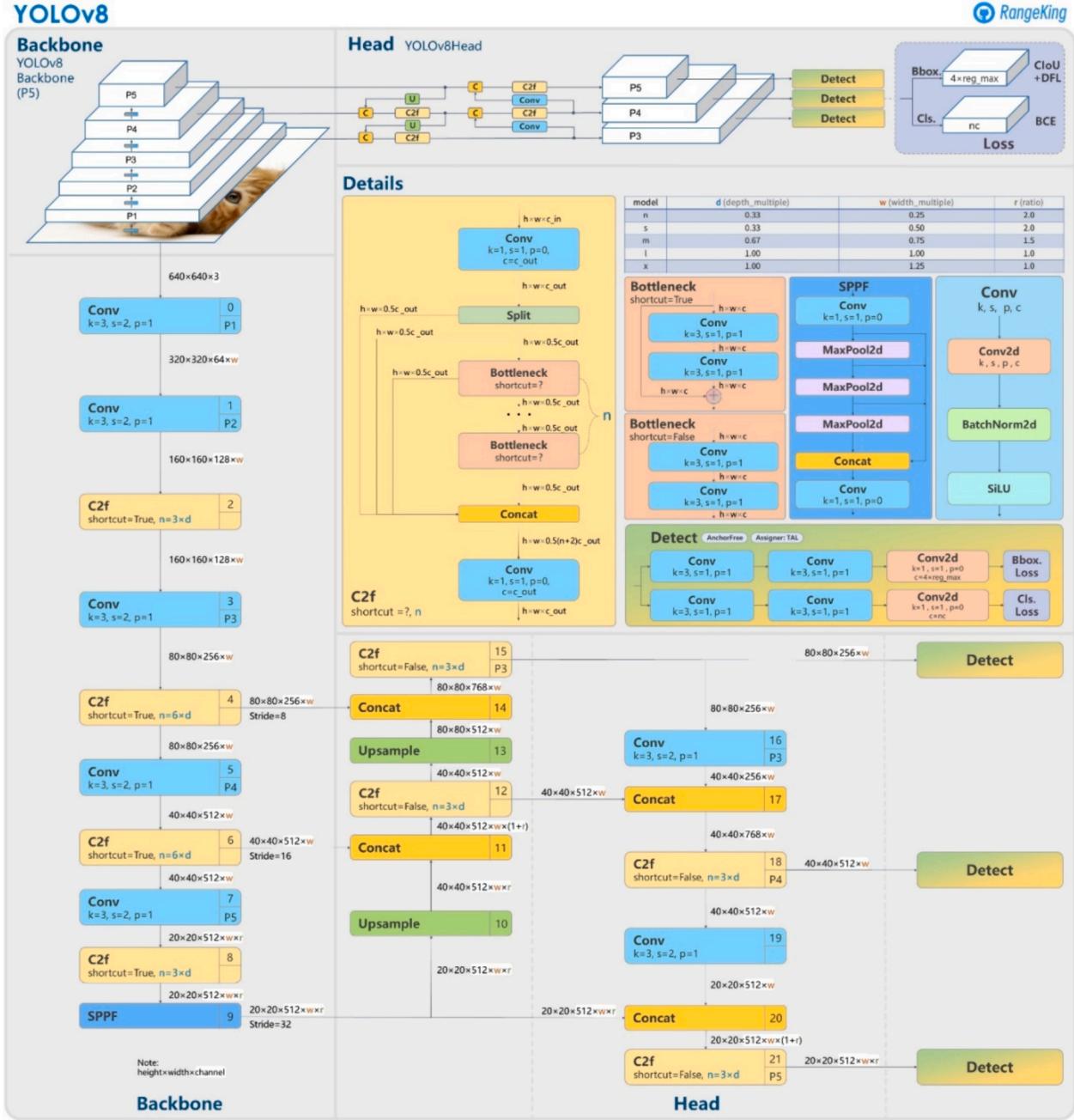


Fig. 9. YOLOv8 architecture [77].

Equation (3.9).

$$AP = \int_0^1 \Pr(Re).dPr \quad (3.9)$$

Finally, the mAP metric, which is widely used in cases where there is more than one class, is obtained by averaging all class APs. The calculation of the mAP value is given in Equation (3.10).

$$mAP = \frac{1}{N} \sum_{i=1}^N AP_i \quad (3.10)$$

N refers to the number of classes in the dataset.

#### 4. Result and discussion

In this section, the results of the YOLOv8 object detection model in

detecting surface defects in plastic parts are given. In the first subheading, where the data was obtained, which defect types were used, which data augmentation techniques were used and the reason, labeling of the data and separation of the data into training and testing were done. In the second subsection, the training results of YOLOv8 models, interpretation of the results using graphs and tables, selection of the model that gives the highest mAP value, which hyper parameters were used in the study, and hyperparameter research was conducted on the model that gave the best mAP value. In the third subsection, the model was tested for the hyperparameter setting that gives the best mAP value and a model recommendation was made. The flow chart of data collection and data pre-processing, training, testing and final evaluation phases of the object detection model is given in Fig. 11.

**Table 1**  
Studies based on the YOLOv8 object detection model.

| Author and year       | Purpose of the study  | mAP   | Pr    | Re    |
|-----------------------|---|-------|-------|-------|
| Ju and Cai [80]       | Fracture detection with YOLOv8 algorithm using pediatric wrist trauma x-ray images  | 63.1  | 69.4  | 59.2  |
| Zhou et al. [54]      | Detection of surface defects of non-standard parts by method based on YOLOv8-2d defect detection model and balancing generative adversarial network | 97.3  | 99.83 | 96.53 |
| Ang et al. [100]      | Real-time arrhythmia detection using YOLOv8   | 99.2  | 98    | 98.1  |
| Wang et al. [101]     | Detection of small objects with the UVA-YOLOv8 object detection model developed for UAV aerial photography scenarios                                | 47    | 54.4  | 45.6  |
| Dumitriu et al. [102] | Automatic detection of hazardous surface water currents using the YOLOv8 object detection model   | 88.94 | 88.27 | 84.67 |
| Luo et al. [103]      | Detection of foreign objects other than coal on the conveyor belt using the algorithm developed based on YOLOv8                                     | 95.6  | 96.9  | 92.7  |
| Tamang et al. [73]    | Accurate face mask classification using YOLOv8 algorithm  | 96    | 95    | 95    |
| Zhang et al. [91]     | Detection of agricultural pests and diseases using the DCF-Yolov8 algorithm   | 60.8  | 53    | 60.4  |
| Ling et al. [104]     | Precise detection of dense printed circuit board (PCB) components with the developed YOLOv8 model   | 87.7  | 89.7  | 83.5  |
| Wang et al. [105]     | Determining miners' behavior to prevent coal mine safety accidents resulting from substandard mining operations using YOLOv8                        | 95.7  | 95.3  | 95.1  |
| Du et al. [106]       | Detecting distracted driving behavior using YOLOv8  | 96.3  | 93.8  | 92.3  |
| Yue et al. [93]       | Real-time and effective segmentation of tomato fruit, surface color and surface features using the developed YOLOv8-Seg network                     | 92.2  | 91.9  | 85.8  |
| Zhai et al. [107]     | Small UAV object detection with optimized YOLOv8 network  | 95.3  | 97    | 89.5  |
| Shah et al. [108]     | Detection of ulcers and similar chronic wounds in medical images using DHuNeT (Dual-phase hyperactive UNet) and YOLOv8 Algorithm                    | 86    | 80.08 | 79.90 |
| Zhang et al. [109]    | Identification of small insulator faults in transmission lines using a fault detection algorithm based on YOLOv8                                    | 91.8  | 93.2  | 84.3  |

$$\text{IoU} = \frac{\text{area of overlap}}{\text{area of union}} = \frac{\text{Intersection}}{\text{Union}}$$

**Fig. 10.** IoU is calculated by dividing the intersection of two boxes by the union of boxes [69].

#### 4.1. Data collection and data pre-processing

The data used in the study consists of images of defective parts produced in an automotive supplier industry company. The company

where the study was conducted has two businesses based in Bursa, is established on a total area of 12 thousand m<sup>2</sup> and employs a total of 415 personnel. The company produces automobile interior, exterior and functional plastic parts. Since a large number of defective parts are produced in the company and manpower is used to control these parts, the defective parts are not noticed and are sent to customers. Since this incident occurs very frequently and customer dissatisfaction is high, it was decided to conduct the study in the company. A Pareto diagram was created using the surface defect types and number of defects seen in the 11,598,030 products produced in total in the company in 2023. The purpose of creating this diagram is to identify the most common defect types in the company. In the Pareto diagram given in Fig. 12, it is observed that 80% of the defects are made up of stains, scratches and glare defects. Therefore, the study was conducted using image data of these three defect types. The glare defect type is more difficult to detect than the other two defect types.

A few of the defective images used in the dataset are shown in Fig. 13.

In total, 257 defective parts and 14 perfect part images were collected. However, this number is not sufficient for education. For this reason, some data augmentation techniques frequently used in the literature were used. Data augmentation techniques called mirroring, noise addition, rotation and shifting were used for each image. In the rotation process, the image was rotated 45°, 90°, 135°, 180°, 225°, 270° and 315° and new images were obtained. With all these data augmentation techniques, 10 new images were obtained from each image, and the number of defective part images increased to 2827. The pixel resolution of the images is 3024x3024 and the images are converted to JPG format. Since the large size of the images obtained increases the training and testing time for data processing and defect detection, the resolution of all images has been reduced. Fig. 14 shows new images created with data augmentation techniques of an image.

Since the defects did not appear or disappeared in a few of the images with noise added and shifting, those images were removed from the data set. In the last case, a total of 2630 defective part images are included in the data set. The images to be used in defect detection training are labeled as shown in Fig. 15.

As seen in Fig. 15, the defective area on the image was boxed and labeled. The point that needs to be taken into consideration at this stage is to label the defect exactly at its borders. MakeSense web-based application was used for labeling. The value "0" was used for scratch defects, the value "1" was used for stain defects, and the value "2" was used for glare defects. The output values of the tagged images are saved in a text file. If there is a defect in the image, the text file consists of a single line, if there are two defects, it consists of two lines. The first value in the text file indicates the class to which the defect belongs, the second and third values specify the coordinates where the defect is located in the image (x and y center coordinates of the bounding box), the fourth value indicates half the width of the box, and the fifth value indicates half the height of the box. Fig. 16 shows images of the labeling process for three classes.

The data set used in the study is divided into 70% training, 10% validation and 20% testing.

#### 4.2. Training phase of YOLOv8 object detection model

In this section, the YOLOv8 model ("n", "s", "m", "l", "x") that shows the best performance using the training data set will be selected. For the training, Python code was taken from Github Ultralytics and made suitable for the study. Training and testing were carried out in the Google Colab environment with the A100 GPU. In all trainings for each model, the number of epochs was selected as 100, the batch size was 16, the learning rate was 0.01 and the IoU value was 0.50. mAP, Pr, Re, F1 performance metrics were used to compare YOLOv8 models. Additionally, training duration was also included in the evaluation. Table 2 shows the performance values obtained after training the YOLOv8

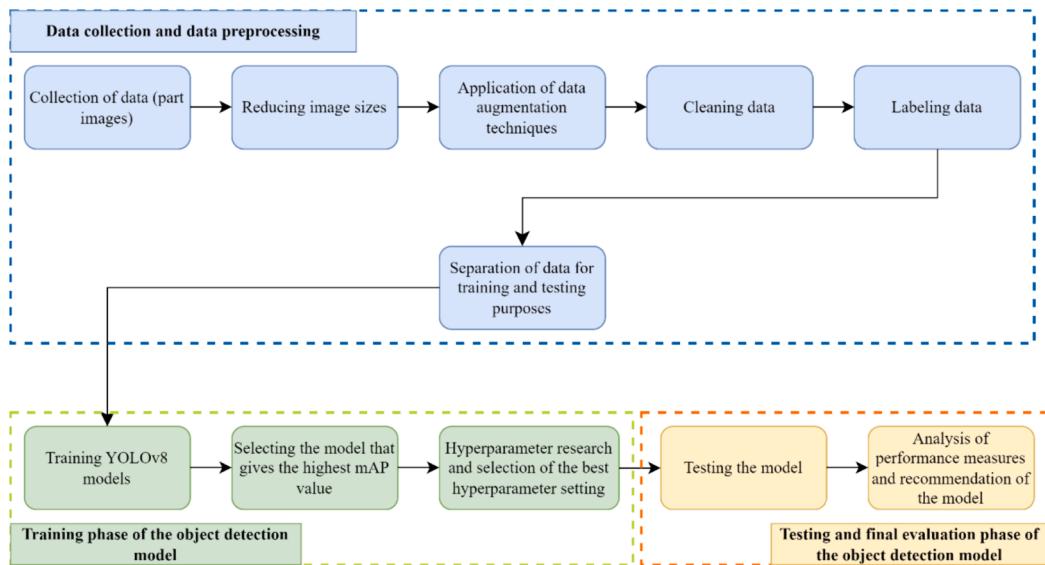


Fig. 11. Flow chart of the study.

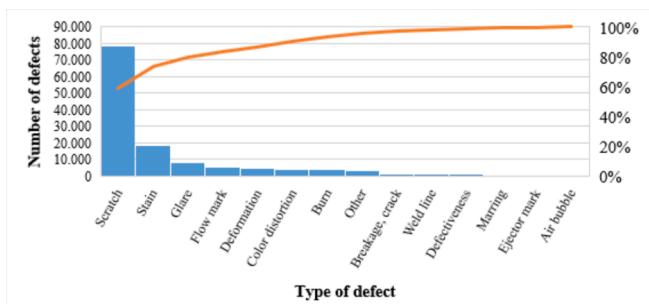


Fig. 12. Pareto diagram.



Fig. 13. A few of the defective images used in the dataset.

models.

When Table 2 is examined, YOLOv8s is the model with the highest performance in all evaluation metrics except training time. In terms of training duration, YOLOv8n was the model that completed the training

in the shortest time. When the models were examined in terms of mAP value, it was seen that YOLOv8n and YOLOv8m models were very close to each other. The mAP50 values of the YOLOv8l and YOLOv8x models were equal. Fig. 17 shows the changing values of mAP, Pr and Re evaluation metrics of the YOLOv8s model over 100 epochs.

Changing mAP values of all YOLOv8 models over 100 epochs are shown in Fig. 18.

When Fig. 18 is examined, the YOLOv8s model started to converge after the 65th Epoch, that is, the learning process slowed down. Training continues after the 70th Epoch for the YOLOv8n model, after the 80th Epoch for the YOLOv8m model, and until the 90th Epoch for the YOLOv8l and YOLOv8x models. The mAP value of the YOLOv8s model increased above 90 between 40–45 epochs. For the same epoch, the mAP value of the YOLOv8n model is below 90, and the mAP value of YOLOv8l and YOLOv8x is below 80. In this case, it is sufficient to train up to 70 epochs for YOLOv8s and YOLOv8n models for the same data set.

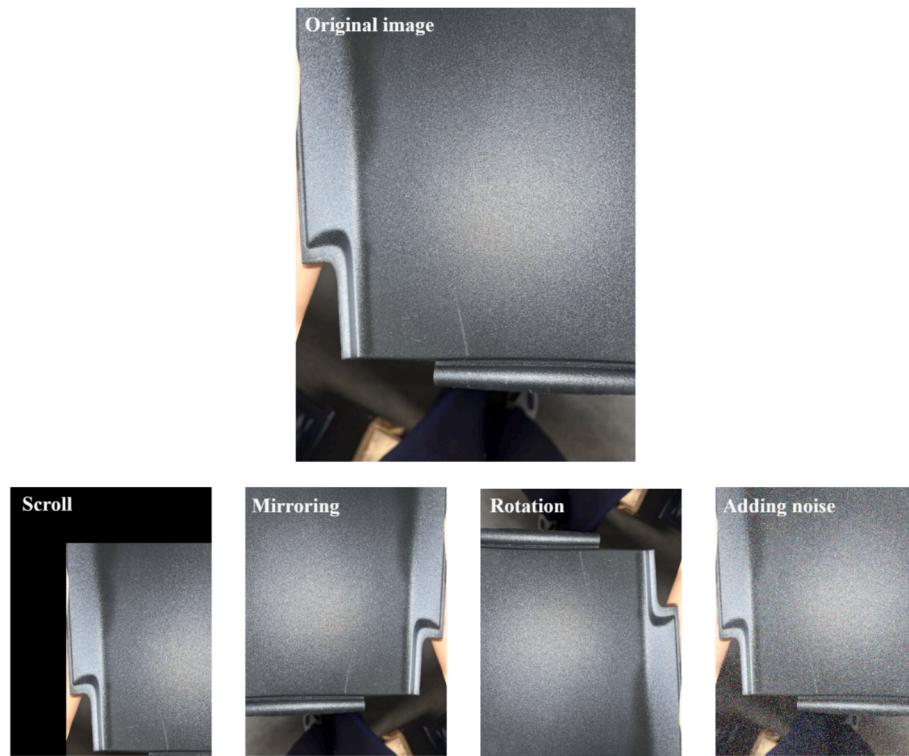
According to the results, the YOLOv8s model was chosen because it had the highest mAP value, and the batch size and learning rate hyperparameters were retrained with different values. All trainings were run for 100 epochs and the IoU value was selected as 0.50. The batch size was determined as 8, 16 and 32, the learning rate was 0.005, 0.01 and 0.02, and training was repeated with nine different hyperparameter settings for the YOLOv8s model in total. All obtained results are given in Table 3.

According to the results given in Table 3, the mAP value is 0.990 at the hyperparameters of batch size 16 and learning rate 0.01. In addition, mAP values were found to be 0.990 for all learning rates in 32 batch sizes. However, when the training times were examined, the best result was chosen for the testing phase because the batch size was 16 and the learning rate was 0.01 hyperparameters.

#### 4.3. Testing and final evaluation phase of the object detection model

A total of 526 defective part images and 154 perfect part images were used during the testing phase. In addition to the accuracy of detecting the class and location of defects in defective parts, it is also of great importance not to find any defects in the images of perfect parts. For this reason, perfect part images are also included in the study. Image examples of test results are shown in Fig. 19.

Table 4 shows the test results of the YOLOv8s model. According to the results, the mAP value was found to be 0.902, the Re value was 0.88, the Pr value was 0.896 and the F1 value was 0.887. The defect detection time from an image in the model is 7 ms.



**Fig. 14.** New images created with data augmentation techniques of an image.



**Fig. 15.** Labeling process for an image.

**Fig. 20** shows the normalized confusion matrix resulting from the testing phase of the YOLOv8 model. In the confusion matrix, the columns show what the actual class is, and the rows show the predicted class.

Interpreting the normalized decision matrix involves evaluating the confusion matrix in terms of ratios. This normalized matrix shows the proportional values of the actual and predicted classes. Additionally, it can be seen that there is a class called background in the matrix. Background class refers to everything outside the three specified classes.

**Fig. 20** shows that about 92% of the scratch defect type was correctly classified as scratch by the model, about 96% of the stain defect type was correctly classified as stain by the model, and about 89% of the glare defect type was correctly classified as scratch by the model. can be interpreted as being classified as glare.

## 5. Conclusion

Defects in products that occur for many reasons are an important

problem for businesses. Defects seen especially in parts used in the automotive industry can lead to great risks. Defects seen on the surfaces of the parts are reflected in the companies as a negative feedback for the customers and cause costs. Therefore, detecting defects in parts plays an important role in the developing competitive environment. However, the fact that most businesses rely on human resources during the detection phase may cause defects not to be detected accurately and quickly. However, due to the developments in technology in recent years, automatic defect detection largely eliminates this problem.

In this study, surface defects on plastic parts produced in a company operating in the automotive sub-industry were detected using the YOLOv8 object detection model. The study was carried out in three stages. In the first stage, three common defect types in the parts were identified and part images were collected. Due to the high size of the collected images, the dimensions were reduced and then data augmentation techniques were applied. Data augmentation techniques called mirroring, noise addition, rotation and shifting were used for each image and 10 images were obtained from each image. The purpose of using



**Fig. 16.** Some of the labelled images.

**Table 2**  
Training performances of YOLOv8 models.

| Model   | mAP50        | mAP50-95     | Pr           | Re           | F <sub>1</sub> | Training duration (hours) |
|---------|--------------|--------------|--------------|--------------|----------------|---------------------------|
| YOLOv8n | 0,986        | 0,675        | 0,974        | 0,952        | 0,962          | <b>0,654</b>              |
| YOLOv8s | <b>0,990</b> | <b>0,729</b> | <b>0,979</b> | <b>0,970</b> | <b>0,974</b>   | 0,714                     |
| YOLOv8m | 0,981        | 0,689        | 0,965        | 0,929        | 0,947          | 0,819                     |
| YOLOv8l | 0,962        | 0,665        | 0,937        | 0,906        | 0,921          | 1,042                     |
| YOLOv8x | 0,962        | 0,668        | 0,947        | 0,901        | 0,923          | 2,177                     |

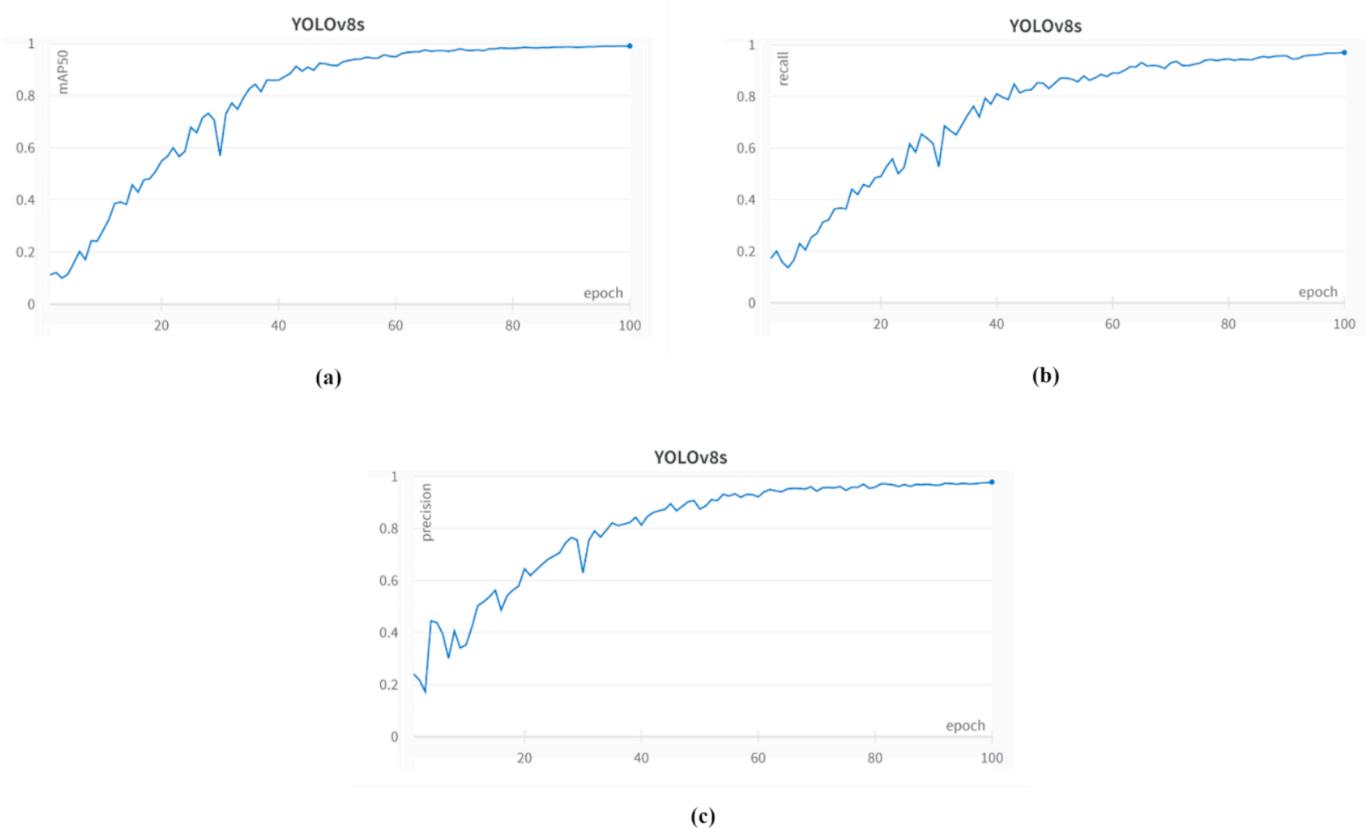
data augmentation techniques in the study is to increase the number of images to a sufficient level for training. After the application of data augmentation techniques, the data was cleaned due to the disappearance of defective areas in some images. Then, labeling was done and the data was divided into training and testing.

In the second stage, training images were trained with YOLOv8 models. After training, it was observed that the YOLOv8s model gave the best mAP value ( $mAP = 0.99$ ) and the YOLOv8n model gave the best training time. The model that gave the highest mAP value was selected and adjustments were made for 2 hyperparameters. The batch size was set as 8, 16, and 32, and the learning rate was set as 0.005, 0.01, and 0.02, and the mAP values were examined. It was observed that the

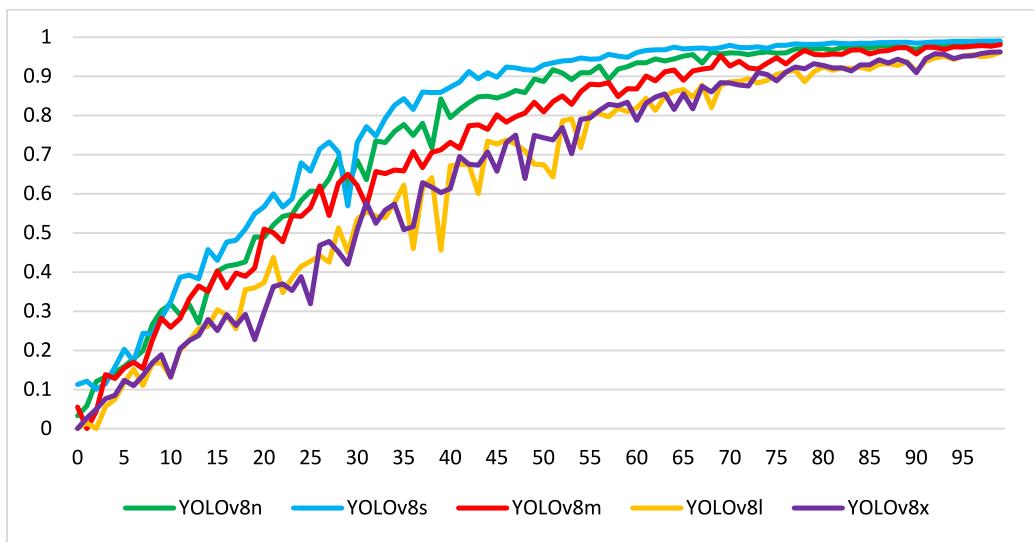
YOLOv8s model with a batch size of 16 and a learning rate of 0.01 gave the best results ( $mAP = 0.99$ ). It can be seen that these hyper parameters are the hyper parameters used in the initial training phase. When the mAP values of the models changing over 100 epochs were examined, it was seen that the YOLOv8s model, which gave the best results, started to converge after the 65th epoch, meaning that learning slowed down. It has been observed that this situation starts later in other models.

In the last stage, the model was tested using the hyper parameter settings that gave the best results and the mAP value was found to be 0.902. When the studies using the YOLOv8 object detection model in Table 1 were examined, it was determined that the mAP value obtained from this study was higher than other studies.

The study is important because it is the first study to detect surface defects on plastic parts in the automotive industry using the YOLOv8s object detection model, and at the same time, it uses real data instead of a ready-made data set and is carried out in university-industry cooperation. The study will enable the company to make its processes more efficient, reduce human errors and reduce operational costs by detecting defects in the parts with a high-accuracy artificial intelligence model without the human factor. At the same time, detecting defective parts will help increase the quality standard and customer satisfaction, and as a result, the company image will increase. All these results will help the business become more efficient, safe and competitive and will allow the



**Fig. 17.** (a) mAP50 metric, (b) Re metric, (c) Pr metric.



**Fig. 18.** Changing mAP values of YOLOv8 models over 100 epochs.

business to improve its business processes. In addition, the high speed and accuracy of the YOLOv8 object detection algorithm in detecting defects in plastic parts will contribute to preventing human error in quality control of the manufactured parts. Therefore, the purpose of this study is to serve as a guide for the specific company involved in the study and generally for other companies that want to detect defects quickly and at low cost.

The method proposed in the study can be used to detect different types of defects on the surfaces of plastic parts. Additionally, the model can be run by making different hyperparameter settings. All these

situations may vary depending on the problem to which the study will be applied and the sector in which the study is applied.

## 6. Discussion and evaluation

The caliber and makeup of the training dataset have a major impact on YOLOv8's performance and generalization, just like it does on other machine learning models. To optimize YOLOv8's performance in a range of object identification tasks, a high-quality, balanced, and diversified dataset is essential. For YOLOv8 to function properly, high-quality

**Table 3**  
YOLOv8s model hyperparameter results.

| Batch size | Learning rate | mAP   | Train duration (hour) |
|------------|---------------|-------|-----------------------|
| 8          | 0,005         | 0,985 | 0,143                 |
|            | 0,01          | 0,985 | 1,134                 |
|            | 0,02          | 0,985 | 1,122                 |
| 16         | 0,005         | 0,988 | 0,870                 |
|            | 0,01          | 0,990 | 0,714                 |
|            | 0,02          | 0,988 | 0,846                 |
| 32         | 0,005         | 0,990 | 0,904                 |
|            | 0,01          | 0,990 | 0,901                 |
|            | 0,02          | 0,990 | 0,852                 |

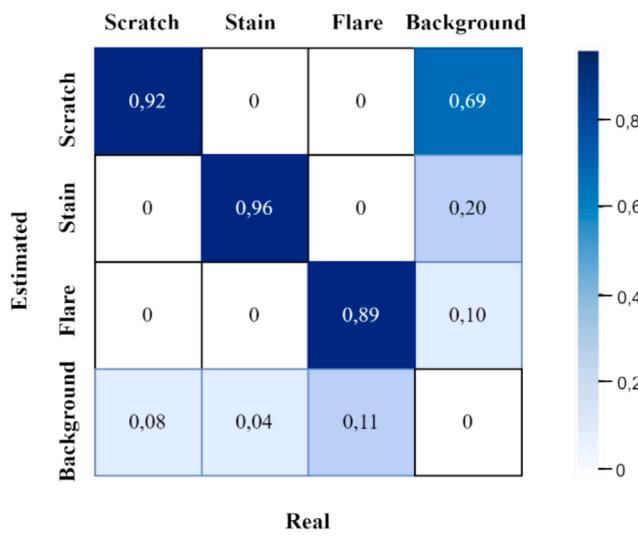
datasets with distinct, well-annotated photos are essential. Inaccurate labels or low-resolution photos are examples of noise in the dataset that might affect model performance and item recognition accuracy. As a result, careful consideration was given to conducting the study with a balanced quantity of data for each of the three categories of faults and making sure the photos had a high quality. Furthermore, the performance of the YOLOv8 model is also highly influenced by the quantity of data points utilized in training the model. When trained with little data,

**Table 4**  
Test results of the YOLOv8s model.

| Model   | mAP50 | mAP50-95 | Pr    | Re   | F <sub>1</sub> | Detection time (ms) |
|---------|-------|----------|-------|------|----------------|---------------------|
| YOLOv8s | 0,902 | 0,449    | 0,896 | 0,88 | 0,887          | 7                   |



**Fig. 19.** Image examples of test results.



**Fig. 20.** Normalized confusion matrix.

deep architectures like YOLOv8 are especially vulnerable to the danger of overfitting. On the training data that it has overadapted, the model performs well; but, on fresh, untested data, it performs worse. To address this issue, various methods such as early stopping and data augmentation, which are applied in the literature, are used.

In addition, the surface properties of the parts examined in the study and the type of defects decided to be used significantly affect the model's detection capability. Detecting irregularly shaped or extremely complex defects, defects that closely match the background material or color of the part, and smaller defects is quite challenging. Furthermore, whether the surface of the part examined is matte or glossy, has complex patterns or textures, or is dirty, worn, or damaged, can reduce detection accuracy by being mistaken for actual defects.

YOLO models have various drawbacks in addition to their notable benefits. One of these is their participation in intricate computations and their need for significant processing capacity to process photos in real-time. The model receives input frames and performs instantaneous processing on them during real-time processing. But usually, this means processing each frame in less than 30 ms. For systems with little resources, such embedded systems or mobile devices, this can be difficult. The amount of memory, energy, and processing power that is accessible is restricted in situations such as mobile devices. When using such devices, YOLO models might not be able to reach their maximum potential, which could result in decreased accuracy or slower processing times.

#### CRediT authorship contribution statement

**Miraç Tuba Çelik:** Writing – original draft, Validation, Software, Methodology, Formal analysis, Data curation. **Seher Arslankaya:** Writing – review & editing, Visualization, Conceptualization. **Aytaç Yıldız:** Writing – review & editing, Methodology, Formal analysis, Conceptualization.

#### Declaration of 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.

#### Data availability

The authors do not have permission to share data.

#### References

- [1] H. Sari-Sarraf, J. Goddard, Vision system for on-loom fabric inspection, *IEEE Trans. Indus. Appl.* 35 (6) (1999) 1252–1259.
- [2] G.A. Bek, Process and Quality Control in a Clothing Mill, Çukurova University, Institute of Science and Technology, Adana, 2008 (MsC Thesis).
- [3] H. Buluklu, Process and Quality Control in Weaving Mills, Çukurova University, Institute of Science and Technology, Adana, 2006 (MsC Thesis).
- [4] F. Duman, The Relationship Between Quality Management and Competitive Management: An Application on Four and Five Star Hotels, Necmettin Erbakan University, Social Sciences Institute, Konya, 2017 (Phd. Thesis).
- [5] F. Yücel, An Analytical Approach for Foreign Trade Flows Between Turkey and Chosen European Union Countries: Before and After Customs Union, Çukurova University, Institute of Social Sciences, Adana, 2006 (Phd. Thesis).
- [6] N. Özil, Quality Costs and Sectoral Quality Cost Analysis in the Total Quality Management System, Ege University, Institute of Science and Technology, Izmir, 1999 (Phd. Thesis).
- [7] S. Madke, P.L. Verma, S. Jain, Enhancement of productivity using statistical quality control tools. a case study in eicher tractor a unit of tafe motors and tractor limited mandideep, *Adv. Eng. Forum* 36 (2020) 47–58.
- [8] W.J. Morse, J.R. Davis, A.L. Hartgraves, *Management Accounting*, third ed., Addison-Wesley Publishing Company Inc., U.S.A, 1991.
- [9] C.D. Heagy, Determining optimal quality costs by considering cost of lost sales, *J. Cost Manag.* 5 (3) (1991) 64–72.
- [10] D. Göktas, Building a Quality and Control System for a Confection Company, Uludağ University, Institute of Science and Technology, Bursa, 2003 (MsC Thesis).
- [11] Ö.İ. Doğan, Kalite uygulamalarının işletmelerin rekabet gücü üzerine etkisi, Dokuz Eylül University, 2000.
- [12] E. Koç, Üretim yönetimi ve organizasyon, Çukurova University Faculty of Engineering and Architecture, Publication number: 33, 2000.
- [13] A. Oztürk, Statistical Quality Control Charts Acceptance Sampling, Selçuk University, Institute of Science and Technology, Konya, 2007 (MsC Thesis).
- [14] Ö. Kurtoğlu, Analizing the Suitable Production and Quality Control Systems for the Stages of Collection Preparation, Ege University, Institute of Science and Technology, Izmir, 2009 (MsC Thesis).
- [15] H. Bircan, S. Özcan, Excel uygulamalı kalite kontrol, Yargı Yayinevi, Sivas, 2003.
- [16] E. Dirgar, Z. Öndoğan, M.C. Erdoğan, The comparison of the modular production system and progressive bundle system in apparel industry, *Textile Apparel.* 15 (2) (2005) 108–113.
- [17] B. Barış, An Expert System Application for Determination of Woven Fabric Defects and Removing Trigerring Factors, Namık Kemal University, Institute of Science and Technology, Tekirdağ, 2018 (Phd. Thesis).
- [18] Url-1 <<https://sozlik.gov.tr/>>, access 15.02.2023.
- [19] M. Akkurt, Kalite Kontrol, second ed., Birsen Yayinevi, İstanbul, 2002.
- [20] İ. Taşçı, Image Processing-Based Defect Detection in Particleboard Production Systems, Süleyman Demirel University, Institute of Science and Technology, Isparta, 2015 (MsC Thesis).
- [21] H. Üzen, Deep Learning Based Texture Defect Detection in Manufacturing Systems, İnönü University, Institute of Science and Technology, Malatya, 2022 (Phd. Thesis).
- [22] Ş. Öztürk, Glass Defect Detection with Image Processing, Selçuk University, Institute of Science and Technology, Konya, 2015 (MsC Thesis).
- [23] G.C. Bükcüci, Defect Detection in Glass Products Using Image Processing Techniques, Tokat Gaziosmanpaşa University, Postgraduate education institute, Tokat, 2021 (Phd. Thesis).
- [24] A. Gökkoruk, Investigation of Production Errors and Remedies that Occur Due to the Mold and Material Used in the Production of Thin Wall Material by Plastic Injection, Munzur University, Postgraduate education institute, Tunceli, 2021 (MsC Thesis).
- [25] A. Birlutiu, A. Burlacu, M. Kadar, D. Onita, Defect Detection in Porcelain Industry Based on Deep Learning Techniques, in: 2017 19th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC), 2018, pp. 263–270.
- [26] J.C.P. Cheng, M. Wang, Automated detection of sewer pipe defects in closed-circuit television images using deep learning techniques, *Autom. Constr.* 95 (2018) 155–171.
- [27] X. Wei, Z. Yang, Y. Liu, D. Wei, L. Jia, Y. Li, Railway track fastener defect detection based on image processing and deep learning techniques: a comparative study, *Eng. Appl. Artif. Intel.* 80 (2019) 66–81.
- [28] J. Yang, S. Li, Z. Wang, G. Yang, Real-Time tiny part defect detection system in manufacturing using deep learning, *IEEE Access* 7 (2019) 89278–89291.
- [29] W. Du, H. Shen, J. Fu, G. Zhang, Q. He, Approaches for improvement of the X-ray image defect detection of automobile casting aluminum parts based on deep learning, *NDT & E Int.* 107 (2019) 102144.
- [30] J. Ren, X. Huang, R. Ren, M. Green, Defect detection from X-ray images using a three-stage deep learning algorithm, in: 2019 IEEE Canadian Conference of Electrical and Computer Engineering (CCECE), 2019, pp. 1–4.
- [31] J. Liu, F. Guo, H. Gao, M. Li, Y. Zhang, H. Zhou, Defect detection of injection molding products on small datasets using transfer learning, *J. Manuf. Process.* 70 (2021) 400–413.
- [32] S. Shen, H. Lu, M. Sadoughi, C. Hu, V. Nemani, A. Thelen, K. Webster, M. Darr, J. Sidon, S. Kenny, A physics-informed deep learning approach for bearing fault detection, *Eng. Appl. Artif. Intel.* 103 (2021) 104295.
- [33] D. Mery, Aluminum casting inspection using deep object detection methods and simulated ellipsoidal defects, *Mach. Vis. Appl.* 32 (3) (2021) 72.

- [34] D. Yang, Y. Cui, Z. Yu, H. Yuan, Deep learning based steel pipe weld defect detection, *Appl. Artif. Intell.* 35 (15) (2021) 1237–1249.
- [35] F. Xu, Y. Liu, B. Zi, L. Zheng, Application of deep learning for defect detection of paint film, in: 2021 6th International Conference on Intelligent Computing and Signal Processing (ICSP), 2021, pp. 1118–1121.
- [36] T. Zhang, C. Zhang, Y. Wang, X. Zou, T. Hu, A vision-based fusion method for defect detection of milling cutter spiral cutting edge, *Measurement* 177 (2021) 109248.
- [37] D. Medak, L. Posilović, M. Subašić, M. Budimir, S. Lončarić, DefectDet: a deep learning architecture for detection of defects with extreme aspect ratios in ultrasonic images, *Neurocomputing* 473 (2022) 107–115.
- [38] M. Yilmazer, M. Karakoç, İ. Aydin, E. Akin, Multiple fault detection in railway components with mask R-CNN deep neural network, *Cukurova Univ. J. Faculty Eng.* 37 (4) (2022) 1103–1112.
- [39] D. Joshi, T.P. Singh, G. Sharma, Automatic surface crack detection using segmentation-based deep-learning approach, *Eng. Fract. Mech.* 268 (2022) 108467.
- [40] I. Shafi, A. Mazahir, A. Fatima, I. Ashraf, Internal defects detection and classification in hollow cylindrical surfaces using single shot detection and MobileNet, *Measurement* 202 (2022) 111836.
- [41] W. Chen, B. Zou, C. Huang, J. Yang, L. Li, J. Liu, X. Wang, The defect detection of 3D-printed ceramic curved surface parts with low contrast based on deep learning, *Ceram. Int.* 49 (2) (2023) 2881–2893.
- [42] T. Zhang, Z. Wang, F. Li, H. Zhong, X. Hu, W. Zhang, D. Zhang, X. Liu, Automatic detection of surface defects based on deep random chains, *Expert Syst. Appl.* 229 (2023) 120472.
- [43] S. Gou, D. Huang, S. Liao, F. Luo, Y. Yuan, L. Liu, X. Wen, Online defect detection method of optical cable pitch based on machine vision technology and deep learning algorithms, *Opt. Laser Technol.* 171 (2024) 110344.
- [44] H.W. Zhang, L.J. Zhang, P.F. Li, D. Gu, Yarn-dyed fabric defect detection with YOLOv2 based on deep convolution neural networks, in: 2018 IEEE 7th Data Driven Control and Learning Systems Conference (DDCLS), 2018, pp. 170–174.
- [45] J. Jing, D. Zhuo, H. Zhang, Y. Liang, M. Zheng, Fabric defect detection using the improved YOLOv3 model, *J. Eng. Fibers Fabr.* 15 (2020).
- [46] J.N. Opara, A.B.B. Thein, S. Izumi, H. Yasuhara, P.J. Chun, Defect detection on asphalt pavement by deep learning, *Int. J. Geomat.* 21 (83) (2021) 87–94.
- [47] X. Yang, F. Dong, F. Liang, G. Zhang, Chip defect detection based on deep learning method, in: 2021 IEEE International Conference on Power Electronics, Computer Applications (ICPECA), 2021, pp. 215–219.
- [48] X. Kou, S. Liu, K. Cheng, Y. Qian, Development of a YOLO-V3-based model for detecting defects on steel strip surface, *Measurement* 182 (2021) 109454.
- [49] G. Wan, H. Fang, D. Wang, J. Yan, B. Xie, Ceramic tile surface defect detection based on deep learning, *Ceram. Int.* 48 (8) (2022) 11085–11093.
- [50] E. Güçlü, İ. Aydin, T.K. Şener, E. Akin, Deep learning based embedded system design for detection of defects on steel surfaces, *EMO Bilimsel Dergi.* 12 (2) (2022) 27–33.
- [51] D. Li, Z. Zhang, B. Wang, C. Yang, L. Deng, Detection method of timber defects based on target detection algorithm, *Measurement* 203 (2022) 111937.
- [52] M. Li, H. Wang, Z. Wan, Surface defect detection of steel strips based on improved YOLOv4, *Comput. Electr. Eng.* 102 (2022) 108208.
- [53] Z. Ying, Z. Lin, Z. Wu, K. Liang, X. Hu, A modified-YOLOv5s model for detection of wire braided hose defects, *Measurement* 190 (2022) 110683.
- [54] F. Zhou, Y. Chao, C. Wang, X. Zhang, H. Li, X. Song, A small sample nonstandard gear surface defect detection method, *Measurement* 221 (2023) 113472.
- [55] Y. Zhang, Y. Yang, J. Sun, R. Ji, P. Zhang, H. Shan, Surface defect detection of wind turbine based on lightweight YOLOv5s model, *Measurement* 220 (2023) 113222.
- [56] W. Guo, S. Qiao, C. Zhao, T. Zhang, Defect detection for industrial neutron radiographic images based on modified YOLO network, *Nucl. Instrum. Methods Phys. Res., Sect. A* (2023) 1056.
- [57] B. Wang, M. Wang, J. Yang, H. Luo, YOLOv5-CD: strip steel surface defect detection method based on coordinate attention and a decoupled head, *Measurement: Sensors* 30 (2023) 100909.
- [58] W. Li, H. Zhang, G. Wang, G. Xiong, M. Zhao, G. Li, R. Li, Deep learning based online metallic surface defect detection method for wire and arc additive manufacturing, *Rob. Comput. Integrat. Manuf.* 80 (2023) 102470.
- [59] B. Zhu, G. Xiao, Y. Zhang, H. Gao, Multi-classification recognition and quantitative characterization of surface defects in belt grinding based on YOLOv7, *Measurement* 216 (2023) 112937.
- [60] C. Zhao, X. Shu, X. Yan, X. Zuo, F. Zhu, RDD-YOLO: a modified YOLO for detection of steel surface defects, *Measurement* 214 (2023) 112776.
- [61] J. Li, Y. Yang, HM-YOLOv5: a fast and accurate network for defect detection of hot-pressed light guide plates, *Eng. Appl. Artif. Intel.* 117 (2023) 105529.
- [62] X. Wang, R. Ma, X. Xu, P. Niu, H. Yang, Non-linear statistical image watermark detector, *Appl. Intell.* 53 (23) (2023) 29242–29266.
- [63] X. Yu, X. Liang, Z. Zhou, B. Zhang, H. Xue, Deep soft threshold feature separation network for infrared handprint identity recognition and time estimation, *Infrared Phys. Technol.* 105223 (2024).
- [64] X. Yu, X. Ye, S. Zhang, Floating pollutant image target extraction algorithm based on immune extremum region, *Digital Signal Process.* 123 (2022) 103442.
- [65] Z. Zhou, B. Zhang, X. Yu, Immune coordination deep network for hand heat trace extraction, *Infrared Phys. Technol.* 127 (2022) 104400.
- [66] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: unified, real-time object detection, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 779–788.
- [67] M. Lin, Q. Chen, S. Yan, Network in network, 2013, ArXiv Preprint ArXiv: 1312.4400.
- [68] İ.E. Parlak, Detection and Classification of Aluminum Casting Defects by Deep Learning Approach, Uludağ University, Institute of Science and Technology, Bursa, 2022 (Phd. Thesis).
- [69] J. Terven, D. Cordova-Esparza, A comprehensive review of YOLO: from YOLOv1 to YOLOv8 and beyond, 2023, ArXiv Preprint arXiv:2304.00501.
- [70] Url-3 <<https://docs.ultralytics.com/#ultralytics-yolov8>>, access 08.09.2023.
- [71] F.M. Talaat, H. ZainEldin, An improved fire detection approach based on YOLO-v8 for smart cities, *Neural Comput. Appl.* 35 (28) (2023) 20939–20954.
- [72] A. Aboah, B. Wang, U. Bagci, Y. Adu-Gyamfi, Real-time multi-class helmet violation detection using few-shot data sampling technique and yolov8, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 5349–5357.
- [73] S. Tamang, B. Sen, A. Pradhan, K. Sharma, V.K. Singh, Enhancing COVID-19 safety: exploring YOLOv8 object detection for accurate face mask classification, *Int. J. Intell. Syst. Appl. Eng.* 11 (2) (2023) 892–897.
- [74] J. Solawetz, F.S. Zuppichini, What is YOLOv8? the ultimate guide, Roboflow Blog, 2023, <https://blog.roboflow.com/whats-new-in-yolov8/#the-yolov8>, Erişim tarihii: 08.09.2023.
- [75] I.P. Sary, S. Andromeda, E.U. Armin, Performance comparison of YOLOv5 and YOLOv8 architectures in human detection using aerial images, *Ultima Comput.: Jurnal Sistem Komputer* 15 (1) (2023) 8–13.
- [76] D. Reis, J. Kupec, J. Hong, A. Daoudi, Real-time flying object detection with YOLOv8, 2023, ArXiv Preprint arXiv:2305.09972.
- [77] Url-4 <<https://github.com/ultralytics/ultralytics/issues/189>>, access 08.09.2023.
- [78] Y. Li, Q. Fan, H. Huang, Z. Han, Q. Gu, A modified YOLOv8 detection network for uav aerial image recognition, *Drones* 7 (5) (2023) 304.
- [79] J.H. Kim, N. Kim, C.S. Won, High-speed drone detection based on Yolo-V8, in: ICASSP 2023–2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2023 pp. 1–2.
- [80] R.Y. Ju, W. Cai, Fracture detection in pediatric wrist trauma x-ray images using YOLOv8 algorithm, 2023, ArXiv Preprint arXiv:2304.05071.
- [81] B. Xiao, M. Nguyen, W.Q. Yan, Fruit ripeness identification using YOLOv8 model, *Multimed. Tools Appl.* 83 (9) (2024) 28039–28056.
- [82] N. Sharma, S. Baral, M.P. Paing, R. Chawuthai, Parking time violation tracking using YOLOv8 and tracking algorithms, *Sensors* 23 (13) (2023) 5843.
- [83] W. Yang, J. Wu, J. Zhang, K. Gao, R. Du, Z. Wu, E. Firkat, D. Li, Deformable convolution and coordinate attention for fast cattle detection, *Comput. Electron. Agric.* 211 (2023) 108006.
- [84] N. Giakoumoglou, E.M. Pechlavani, D. Tzovaras, Generate-paste-blend-detect: synthetic dataset for object detection in the agriculture domain, *Smart Agric. Technol.* 5 (2023) 100258.
- [85] G. Yang, J. Wang, Z. Nie, H. Yang, S. Yu, A lightweight YOLOv8 tomato detection algorithm combining feature enhancement and attention, *Agronomy* 13 (7) (2023) 1824.
- [86] Z. Wang, L. Lei, P. Shi, Smoking behavior detection algorithm based on YOLOv8-MNC, *Front. Comput. Neurosci.* 17 (2023).
- [87] G. Oh, S. Lim, One-stage brake light status detection based on YOLOv8, *Sensors* 23 (17) (2023) 7436.
- [88] S. Akhtar, M. Hanif, H. Malih, Automatic urine sediment detection and classification based on YoloV8, in: International Conference on Computational Science and Its Applications, 2023, pp. 269–279.
- [89] E. Chabi Adjobo, A.T. Sanda Mahama, P. Gouton, J. Tossa, Automatic localization of five relevant dermoscopic structures based on YOLOv8 for diagnosis improvement, *J. Imag.* 9 (7) (2023) 148.
- [90] P. Li, J. Zheng, P. Li, H. Long, M. Li, L. Gao, Tomato maturity detection and counting model based on MHSAs-YOLOv8, *Sensors* 23 (15) (2023) 6701.
- [91] L. Zhang, G. Ding, C. Li, D. Li, DCF-Yolov8: an improved algorithm for aggregating low-level features to detect agricultural pests and diseases, *Agronomy* 13 (8) (2023) 2012.
- [92] E. Soylu, T. Soylu, A performance comparison of YOLOv8 models for traffic sign detection in the Robotaxi-full scale autonomous vehicle competition, *Multimed. Tools Appl.* 83 (8) (2024) 25005–25035.
- [93] X. Yue, K. Qi, X. Na, Y. Zhang, Y. Liu, C. Liu, Improved YOLOv8-Seg network for instance segmentation of healthy and diseased tomato plants in the growth stage, *Agriculture* 13 (8) (2023) 1643.
- [94] N. Karna, M.A.P. Putra, S.M. Rachmawati, M. Abisado, G.A. Sampredo, Towards accurate fused deposition modeling 3d printer fault detection using improved YOLOv8 with hyperparameter optimization, *IEEE Access* (2023) 74251–74262.
- [95] S.A.A. Qadri, N.F. Huang, T.M. Wani, S.A. Bhat, Plant disease detection and segmentation using end-to-end YOLOv8: a comprehensive approach, in: 2023 IEEE 13th International Conference on Control System, Computing and Engineering (ICCSCE), 2023, pp. 155–160.
- [96] H.B. Le, T.D. Kim, M.H. Ha, A.L. Q. Tran, D.T. Nguyen, X.M. Dinh, (2023). Robust surgical tool detection in laparoscopic surgery using YOLOv8 model, in: 2023 International Conference on System Science and Engineering (ICSSE), Ho Chi Minh, Vietnam, July 27–28, pp. 537–542.
- [97] X. Jia, Y. Wang, T. Chen, Forest fire detection and recognition using YOLOv8 algorithms from UAVs images, in: 2023 IEEE 5th International Conference on Power, Intelligent Computing and Systems (ICPICS), 2023, pp. 646–651.
- [98] K.S. Shashank, N.P. Prasad, K.S. Reddy, L.S. Rao, Upload cricket match video to generate audio commentary by YOLOv8 and transformer, in: 2023 International Conference on Sustainable Computing and Smart Systems (ICSCSS), 2023, pp. 1152–1157.

- [99] N. Nandini, D. Hemashree, B.R. Nandish Rao, Fast moving object detection in cricket, *Int. Res. J. Modern. Eng. Technol. Sci.* 5 (6) (2023) 2919–2925.
- [100] G.J.N. Ang, A. K. Goil, H. Chan, J.J. Lew, X. C. Lee, R.B.A. Mustaffa, T. Jason, Z.T. Woon, B. Shen, 2023, A novel application for real-time arrhythmia detection using YOLOv8, *ArXiv Preprint arXiv:2305.16727*.
- [101] G. Wang, Y. Chen, P. An, H. Hong, J. Hu, T. Huang, UAV-YOLOv8: a small-object-detection model based on improved YOLOv8 for UAV aerial photography scenarios, *Sensors* 23 (16) (2023) 7190.
- [102] A. Dumitriu, F. Tatui, F. Miron, R.T. Ionescu, R. Timofte, Rip current segmentation: a novel benchmark and YOLOv8 baseline results, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 1261–1271.
- [103] B. Luo, Z. Kou, C. Han, J. Wu, A “Hardware-Friendly” foreign object identification method for belt conveyors based on improved YOLOv8, *Appl. Sci.* 13 (20) (2023) 11464.
- [104] Q. Ling, N.A.M. Isa, M.S.M. Asaari, Precise detection for dense pcb components based on modified YOLOv8, *IEEE Access* (2023).
- [105] Z. Wang, Y. Liu, S. Duan, H. Pan, An efficient detection of non-standard miner behavior using improved YOLOv8, *Comput. Electr. Eng.* 112 (2023) 109021.
- [106] Y. Du, X. Liu, Y. Yi, K. Wei, Optimizing road safety: advancements in lightweight YOLOv8 models and ghostc2f design for real-time distracted driving detection, *Sensors* 23 (21) (2023) 8844.
- [107] X. Zhai, Z. Huang, T. Li, H. Liu, S. Wang, YOLO-Drone: an optimized YOLOv8 network for tiny UAV object detection, *Electronics* 12 (17) (2023) 3664.
- [108] S.M.A.H. Shah, A. Rizwan, G. Atteia, M. Alabdulhafith, CADFU for dermatologists: a novel chronic wounds & ulcers diagnosis system with DHuNeT (Dual-Phase Hyperactive UNet) and YOLOv8 algorithm, *Healthcare*. 11 (21) (2023) 2840.
- [109] Y. Zhang, Z. Wu, X. Wang, W. Fu, J. Ma, G. Wang, Improved YOLOv8 insulator fault detection algorithm based on BiFormer, in: 2023 IEEE 5th International Conference on Power, Intelligent Computing and Systems (ICPICS), 2023, pp. 962–965.
- [110] R. Padilla, S.L. Netto, E.A. Da Silva, A survey on performance metrics for object-detection algorithms, in: 2020 International Conference on Systems, Signals and Image Processing (IWSSIP), 2020, pp. 237–242.
- [111] Url-2 <<https://docs.ultralytics.com/guides/yolo-performance-metrics/>>, access 05.09.2023.
- [112] İ.E. Parlak, E. Emel, Deep learning-based detection of aluminum casting defects and their types, *Eng. Appl. Artif. Intel.* 118 (2023) 105636.
- [113] E. Ekici, Deep Learning Based Fruit And Vegetable Recognition for Android Pos Devices, Istanbul Technical University, Graduate School, Istanbul, 2022 (M.Sc Thesis).