applied sciences

MDPI

Article

The Efficiency of YOLOv5 Models in the Detection of Similar Construction Details

Tautvydas Kvietkauskas [1], Ernest Pavlov [2], Pavel Stefanovič [3,*] and Birutė Pliuskuvienė [3]

[1] Department of Information Technology, Vilnius Gediminas Technical University, Saulėtekio al. 11, LT-10223 Vilnius, Lithuania; tautvydas.kvietkauskas@stud.vilniustech.lt
[2] Department of Electronic Systems, Vilnius Gediminas Technical University, Saulėtekio al. 11, LT-10223 Vilnius, Lithuania
[3] Department of Information Systems, Vilnius Gediminas Technical University, Saulėtekio al. 11, LT-10223 Vilnius, Lithuania; birute.pliuskuviene@vilniustech.lt
* Correspondence: pavel.stefanovic@vilniustech.lt

Abstract: Computer vision solutions have become widely used in various industries and as part of daily solutions. One task of computer vision is object detection. With the development of object detection algorithms and the growing number of various kinds of image data, different problems arise in relation to the building of models suitable for various solutions. This paper investigates the influence of parameters used in the training process involved in detecting similar kinds of objects, i.e., the hyperparameters of the algorithm and the training parameters. This experimental investigation focuses on the widely used YOLOv5 algorithm and analyses the performance of different models of YOLOv5 (n, s, m, l, x). In the research, the newly collected construction details (22 categories) dataset is used. Experiments are performed using pre-trained models of the YOLOv5. A total of 185 YOLOv5 models are trained and evaluated. All models are tested on 3300 images photographed on three different backgrounds: mixed, neutral, and white. Additionally, the best-obtained models are evaluated using 150 new images, each of which has several dozen construction details and is photographed against different backgrounds. The deep analysis of different YOLOv5 models and the hyperparameters shows the influence of various parameters when analysing the object detection of similar objects. The best model was obtained when the YOLOv5l was used and the parameters are as follows: coloured images, image size—320; batch size—32; epoch number—300; layers freeze option—10; data augmentation—on; learning rate—0.001; momentum—0.95; and weight decay—0.0007. These results may be useful for various tasks in which small and similar objects are analysed.

Keywords: YOLOv5; object detection; construction details; similar objects; hyperparameters

Citation: Kvietkauskas, T.; Pavlov, E.; Stefanovič, P.; Pliuskuvienė, B. The Efficiency of YOLOv5 Models in the Detection of Similar Construction Details. *Appl. Sci.* **2024**, *14*, 3946. https://doi.org/10.3390/app14093946

Academic Editor: Andrea Prati

Received: 22 February 2024
Revised: 15 April 2024
Accepted: 4 May 2024
Published: 6 May 2024

1. Introduction

Over the past decade, the application of artificial intelligence has grown in various areas. Many different methods of artificial intelligence can be used for different types of data analysis, such as those involving numbers, texts, sounds, and images. Deep learning methods play a significant role in various scientific research. This is due to the possibility of using not only the CPU, but also the GPU in model building. One field of artificial intelligence based on deep learning methods is computer vision. The most popular computer vision tasks are image classification, segmentation, and object detection. For example, in medicine, image data can be used to predict different diseases, such as cancer [1,2], glaucoma [3,4], and pneumonia [5,6]. Object detection models can be used in systems for travel direction recommendation [7], in industry for solutions to robotization tasks [8,9], in face detection for different applications [10,11], or other fields [12–16]. Usually, in all research, various computer vision methods or combinations are used to solve the

specific problem. This is because there is no unambiguously appropriate method for all of them and, as a result, the results may depend on various factors.

One of the factors in building successful artificial intelligence models is properly prepared data for model training. A large amount of research has sought to analyse the data showing features that are distinctly different, and for which the natural size of the object is large in the real world [17,18]. In this case, the obtained results of object detection are high. A more complex task is to detect a similar object within an image, especially when the object is small in the real world. Objects that are similar can be described by the following characteristics: shape, colour, size, etc. For example, if we analyse medical pill detection, some of the pills can look identical in different images, depending on the angle, distance, lighting, shadows, and other external factors. At self-service checkouts [19], object detection methods have been implemented to detect fruits. It is difficult for models to determine what type of apple the customer is trying to buy due to the similarity between fruits. The same problem occurs in the construction detail analysis because some details can look identical. Detecting similar objects requires a much deeper analysis.

In this investigation, the efficiency of YOLOv5 has been investigated using the newly collected construction details dataset [20]. In construction detail analysis, datasets have a large number of categories, and items have similar features. It is therefore important to investigate the parameters of the dataset and find which of them has the highest influence on object detection. Additionally, one must take into account the size of the chosen YOLOv5 model and the selected training hyperparameters. The training dataset used in the experimental investigation consisted of 440 images (22 construction details on a white background, with 20 images in each category). Additionally, a test dataset consisting of 3300 images (22 construction details on 3 different backgrounds, with 50 images belonging to each category). Because training each model costs a large amount of time, the experimental investigation was performed in two stages. In the first stage, primary research was performed to determine the influence of epoch numbers, image size, batch size, layer freeze option, and data augmentation on object detection results. A total of 50 experiments were performed using the most popular and widely used models of other researchers—YOLOv5s and YOLOv5m. During the primary experiments, the best parameters were found and used in the second stage. In the second stage, a total of 135 experiments were performed using all five models of YOLOv5 (n, s, m, l, x). The main aim was to find the best training hyperparameters, such as learning rate, weight decay, and momentum. The main contributions of the paper are as follows:

(1) The newly collected dataset has been prepared, is publicly available, and can be used in various computer vision tasks.
(2) The five YOLOv5 models of different sizes have been experimentally investigated using the newly collected construction details. A total of 185 experiments have been performed, in which various combinations of the training and algorithm parameters have been analysed.
(3) The results of the experimental investigation have shown the efficiency of different models, which allows us to see which nondefault parameters help to achieve higher object detection results. This could be useful for other researchers when analysing similar featured data.
(4) The models could be used in the recommendation systems that allow the recommendation of a possible construction by detecting several dozen construction details in one image.

The structure of this paper is as follows. In Section 2, the related works are reviewed. No research has yet been able to solve the problem we are addressing in terms of the detection of similar construction details, nevertheless most similar research is herein overviewed. In addition, a brief overview of the most popular object detection algorithms is presented. In Section 3, an experimental investigation scheme is presented and described in detail. All steps from data collection and data preprocessing to model training and evaluation

are presented. In Section 4, the discussion and limitations of the research are presented. Section 5 concludes the paper.

2. Related Works

The literature analysis has shown that, due to the complexity of the task, there is a lack of research that focuses on the detection of similar objects. Therefore, it is difficult to perform a comparative analysis of such research results. Usually, in such types of object detection tasks, the accuracy of the obtained model is smaller compared with other types of data. Therefore, in various investigations, different object detection algorithms and their parameters are changed in order to increase the model's accuracy. Several research studies have been published that deal with the problem of similar object detection, though they have used different kinds of data.

In the investigation by Kwon et al. [21], the detection of medical pills was analysed using a deep learning algorithm. The authors proposed a two-step model based on a mask region-based convolutional neural network (Mask R-CNN) [22] that improved the detection performance of medical pills. In the first step, the object localization problem was solved in order to detect the medical pill in the image, and, in the second step, the multiclass classification was solved in order to detect the possible type of the medical pill. According to the testing results of the proposed model and YOLOv3 [23], experiments have shown that the accuracy of the proposed Mask R-CNN model (91%) is 18% higher than the results obtained using YOLOv3 (73%). The results obtained have shown that the proposed model can be applied in cases when a small amount of data are used to train the object detection models. Another study, which also focused on the real-time detection of medical pills, was performed by Tan et al. [24]. In this research, the efficiencies of the following three object detection algorithms were investigated: RetinaNet, Single Shot Multi-Box Detector (SSD), and YOLOv3. The results of the experimental investigation show that RetinaNet is not suitable for real-time medical pill detection due to slow performance (FPS–17), but that the accuracy, when compared with the other analysed algorithms, was the highest (82.89%). The highest speed performance was obtained by YOLOv3 (FPS–51), but the accuracy is smaller (80.69%) compared with RetinaNet and SSD. Intermediate performance was obtained by the SSD algorithm, where the accuracy was equal to 82.71% (slightly smaller when compared with the RetinaNet) and the speed was equal to 32 (FPS). By concluding the results, the authors state that YOLOv3 is more suitable for similar object detection tasks when the medical pills are analysed. In the research by Ou et al., models based on convolutional neural networks were used to detect and classify medical pills in images. In 2018 [25], an improved model of Inceptionv3 [26] was used, wherein models were trained using a newly collected dataset. The prepared dataset consisted of more than 470,000 images, where each category (different types of medical pills, for a total of 131 categories) had approximately 3600 images, taken from various angles. During the experimental research, the resolution of the images was transformed to 299 $\times$ 299. The accuracy of the model was evaluated using additional images of medical pills, which contain 400 images with 2825 annotations. The proposed model achieved 79.4% accuracy. Later, in 2020, Ou et al. [27] used Inception-ResNetv2 for the medical pill classification task due to its experimental performance. The same type of dataset was used, but with a larger amount of medical pill images (612 categories) having been prepared for the model training process. Furthermore, the authors analysed the efficiency of various classifiers (VGG-16, VGG-19, ResNet-50, ResNet-101, Inceptionv3, Inceptionv4, Xception, Inception-ResNetv1, Inception-ResNetv2). The highest accuracy (82.1%) was achieved using Inception-ResNetv2, and the smallest accuracy was obtained using VGG 16 (40.5%).

Saeed et al. [28] proposed an approach for the detection of small industrial objects using an improved faster regional convolutional neural network (Improved Faster RCNN). The main aim of their research was to detect and recognize screws in images. This problem is also related to the problem of similar object detection because, in some images taken from different angles, the various screw types may look the same. To train the models, the

authors collected a new dataset from many images of industrial products in which screws could be found. A total of 917 original images of four different types (325, 163, 251, 178) of screws were taken. An augmentation of the dataset was applied and a total of 63,013 images were used in the experimental investigation. The efficiency of the proposed improved model of Faster RCNN was compared with RCNN, Fast RCNN, and Faster RCNN. The experimental results show that the highest accuracy was achieved using the improved Faster RCNN (~91%), followed by the Faster RCNN (~89%), Fast RCNN (~84%), and RCNN (~83%). In the research by Yildiz et al. [29], the authors proposed the combination of the Xception and Inceptionv3 models in order to detect screws in automated disassembly processes. The main objective of the research was to detect screws during hard disk disassembly. All images analysed in the training process were transformed to greyscale. In the research, the efficiencies of Xception, Inceptionv3, ResneXt101, InceptionResnetv2, Densenet201, and Resnet101v2 were evaluated. All analysed models achieved an accuracy greater than 96%, but the highest accuracy was obtained by Inceptionv3 (98.8%), followed by InceptionResnetv2 (98.6%), and ResneXt101 with Xception (98.5%). The lowest accuracy was obtained using Resnet101v2 (96.9%). The authors decided to combine two models with the highest accuracy to increase the accuracy of the combined classifier. For this reason, the results of the models were combined using some chosen weights, and the final prediction results were calculated. The combination of the proposed models achieved 99% accuracy when analysing the selected dataset. In the research by Mangold et al. [30], the YOLOv5 models were used to detect the screw head for automated disassembly and remanufacturing. The authors investigate two types of YOLOv5 group models—YOLOv5s and YOLOv5m. The dataset used in the investigation was pre-processed, and the size of the images reduced to 640 × 640 (the original size of the images was 1200 × 1200). During model training, the batch size was equal to 32. The results of the experimental investigation performed in the research show that the highest accuracy was obtained using the YOLOv5s model (mAP@0.5—98.4% and mAP@0.5:0.95—83.4%). A slightly smaller accuracy was obtained using the YOLOv5m (mAP@0.5—98% and mAP@0.5:0.95—82.6%) but the difference between the different models' accuracy is not significant. The trained YOLOv5s model was evaluated using the real environment, where 20 small and 7 large motor images were passed to the model in order for it to detect screws. The testing results show that 39 out of 45 screws were correctly detected in the images of the small motors and 15 out of 17 screws were correctly detected in the images of the large motors.

This literature review has shown that many object detection algorithms exist and are used in various fields, for example, RCNN, Faster RCNN, SSD, YOLO, etc. [31–33]. Nowadays, one of the most popular object detection algorithm groups is YOLO, which can be used in real-time object detection tasks and the group algorithms of which allow one to obtain promising results in different areas. Of course, there exist many versions of the YOLO algorithm, from the first original version of YOLO to YOLOv8, YOLO-NAS, and YOLO with transformers [34]. The newest versions of YOLO, starting from YOLOv6, are still in the development process, so there are different issues with their practical use. One of the most stable recent versions is YOLOv5, which is widely used in scientific research, such as small and similar object detection [35]. YOLOv5 differs from previous versions of the YOLO algorithm because it uses the PyTorch framework, rather than Darknet, and because it uses CSPDarknet53 as the backbone. The YOLOv5 architecture uses the path aggregation network (PANet) as a neck by which to increase the flow of information. The head of YOLOv5 is the same as that of YOLOv3 and YOLOv4, which generates three different feature map outputs to achieve multiscale prediction. This helps to effectively increase the prediction of small and large objects in the model. The output layer generates the results. In the manuscript by Dlužnevskij et al. [36], experimental research has been performed to investigate the efficiency of YOLOv5 using a mobile device with real-time object detection tasks. Four different models of YOLOv5 have been analysed (YOLOv5s, YOLOv5m, YOLOv5l, YOLOv5x). The experiments were conducted using the original COCO dataset, reducing it to fit the requirements of the mobile environment. The results of

the experimental investigation show that the performance of the model is highly influenced by the hardware architecture and the system in which the model is used.

In our previous research [37], the influence of training parameters on the detection of real-time construction details using YOLOv5s was analysed. Parameters, such as image resolution, batch size, iteration number, and colour of images, were investigated. The focus was only on the one YOLOv5s model that is usually suitable for real-time object detection using a limited technical environment, such as mobile phones. The results of the experimental investigation have shown that, in many cases, the optimal resolution of the construction details images should be 320 × 320 or 640 × 640 and that colour images allow slightly better results compared with greyscale images. Choosing the higher resolution image leads to a lower accuracy of construction detail detection. Furthermore, during model training, the batch size should be chosen as 16 or 32 to achieve higher model accuracy. The limitation of the research was that the other versions of YOLOv5 (n, m, l, x) were not analysed. Additionally, the hyperparameters were not changed during the model training process; instead, only the best hyperparameters are used based on the analysis of related works. The results of related works [38,39] have shown that, generally, other similar research has focused only on a small number of YOLOv5 hyperparameters, such as learning rate, momentum, augmentation parameters, and weight selection, and that other hyperparameters are usually not changed. In the research, the dataset used in the training process was not balanced, and this is important to consider when evaluating the models. Therefore, it is necessary to investigate the influence of different versions of YOLOv5 and hyperparameters on the detection of construction details.

Related works have shown that there is no single best model for object detection and that the results depend on various factors. One of the most important factors is the dataset being analysed. By analysing similar objects, such as medical pills, screws, or construction details, the correct detection depends on the angle of the camera, the lighting, and the position. In some cases, one object can look similar to another. Image pre-processing, such as that involved in the colour of the image or the size of the resolution, also influences the detection results. During the training of the models, it is important to select suitable hyperparameters. However, in computer vision, each new combination of hyperparameters costs a lot of training time because of the image analysis tasks and the model complexities. The object detection model selection is also one of the hardest parts, because related works have shown that older models, such as RCNN, Fast RCNN, or Faster RCNN, can be used to obtain an accuracy that is not inferior to the latest models. In addition, there are many versions or modifications of the object detection model in the scientific literature. All these facts show that it is important to investigate the efficiency of the object detection models using various factors and to find the best combination for each specific domain.

3. Experimental Investigation

To investigate the influence of various training parameters on different models of YOLOv5, an experimental investigation was performed. YOLOv5 is a large step forward in object identification algorithms, departing from its predecessors by leveraging the PyTorch framework and incorporating the CSPDarknet53 backbone with a new pooling architecture. This architecture solves feature fusion and computational efficiency concerns, improving object localisation accuracy while reducing model size. The focus layer enhances memory use and propagation efficiency [40]. The different combinations of training parameters have been used to find the highest accuracy in construction detail detection and, for this reason, a total of 185 models have been created and evaluated. The research workflow is presented in Figure 1.

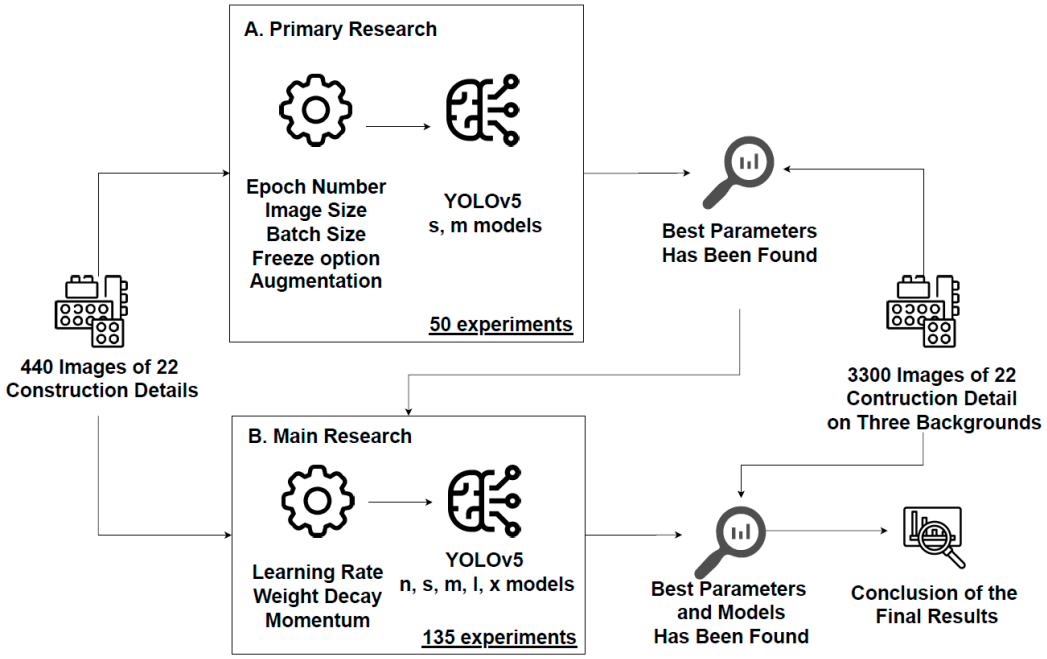

Figure 1. The workflow of the experimental investigation.

The research was performed in two stages. The first stage focuses on training parameters and the second stage focuses on the hyperparameters of the pre-trained YOLOv5 models [40]. All of the YOLOv5 models presented in Table 1 have been trained using the well-known COCO2017 dataset, which was collected and prepared for object detection and segmentation tasks. The COCO2017 dataset is the subset of the MS COCO dataset (containing 164,000 images of 80 different objects with bounding boxes and segmentation masks for each data item). The models were trained using 118,000 images, and the remainder of the dataset was used for validation (5000) and testing (41,000) of the models.

Table 1. The specification of the pre-trained YOLOv5 models [40].

Model	Image Size (pixels)	mAP^{val} (50–95)	mAP^{val} (50)	Speed (ms) CPU bl	Speed (ms) V100 bl	Speed (ms) V100 b32	Params (M)	FLOPs @640 (B)
YOLOv5n	640	28.0	45.7	45	6.3	0.6	1.9	4.5
YOLOv5s	640	37.4	56.8	98	6.4	0.9	7.2	16.5
YOLOv5m	640	45.4	64.1	224	8.2	1.7	21.2	49.0
YOLOv5l	640	49.0	67.3	430	10.1	2.7	46.5	109.1
YOLOv5x	640	50.7	68.9	766	12.1	4.8	86.7	205.7

All of the steps of the experimental investigation that were performed, from data preparation to model training and evaluation, are described in this section in more detail.

During the experimental investigation, all models were trained in an environment with the following specifications: Intel(R) Xeon(R) CPU E5-2630 v4 @ 2.20GHz (20 Threads, 10 Cores). The environment used a Linux operating system with 32 GB DDR4 RAM and a Tesla P100 PCIe 12GB GPU.

3.1. Results of the Primary Research

The newly collected construction details dataset was used in the experimental investigation [20]. The dataset was constructed in such a way that it is divided into three parts in order to be used in three stages. In the first stage, the dataset of 440 images was collected to train the models. The dataset consists of 22 different categories of images of construction details that were photographed on a white background. Each construction detail has been

rotated 20 times in order for each picture to show a new angle. Each item of the dataset has been manually annotated. The number of images in the dataset is not high because the pre-trained models of YOLOv5 have been used. A sample of the analysed dataset is presented in Figure 2.

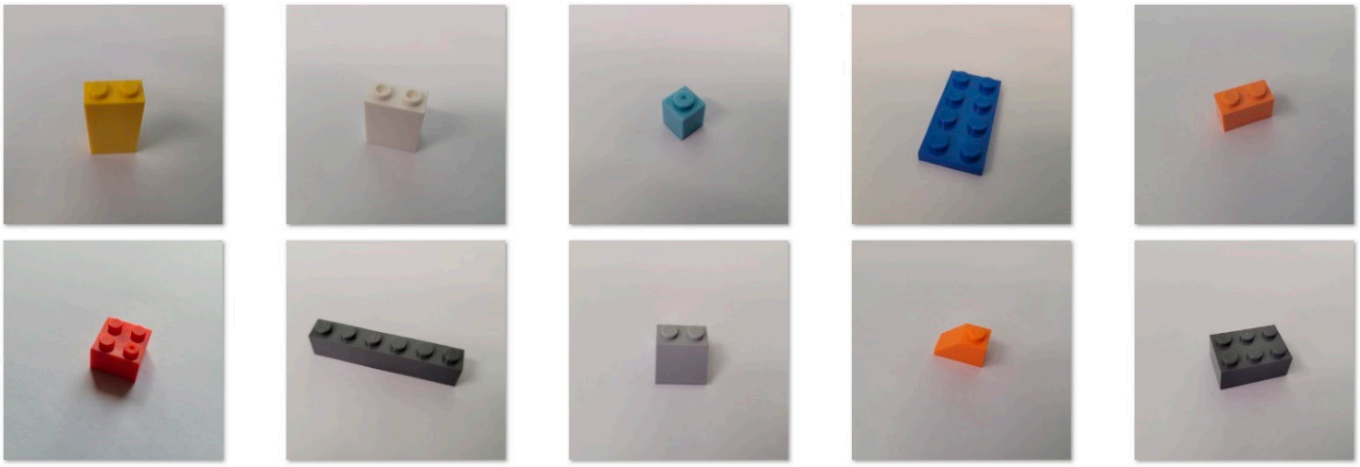

Figure 2. A sample of the dataset used to train the YOLOv5 models.

To evaluate the efficiency of the models, a larger number of construction details have been prepared. Each construction detail used in the training process has been photographed 50 times from different angles using 3 different backgrounds: white (W), neutral (N), and mixed (M) (Figure 3). The main reason for using three different backgrounds is to simulate the efficiency of the models in a real environment. On a neutral background, all analysed construction details can be clearly observed. In contrast, on a white background, all details usually stand out and are highlighted from the background, except the construction details of the white colour. On the third background, which is mixed, the pattern can be considered as noise. In this case, it is more difficult to correctly detect the object compared with the white and neutral backgrounds. A total of 3300 images were prepared (1100 images on each background).

Figure 3. A sample of the dataset used to evaluate the YOLOv5 models.

The last part of the dataset which was used in the experimental investigation is formed by 150 images. In these, several dozen construction details are placed on the three backgrounds (Figure 4). The main idea of these images is to evaluate the efficiency of the YOLOv5 models that obtained the highest accuracy.

As mentioned above, selecting different parameters during the training process can lead to different results. It is important to investigate not only the hyperparameters of the YOLOv5 models, but also the other important parameters that could influence the final detection results. Training each model can be time consuming, so the experimental study was divided into two stages. The first was called primary research, and the second was called main research. In the primary research, the influence of the following five parameters was investigated: epoch numbers (one complete forward and backward pass of all training

examples), batch size (number of images processed simultaneously in a forward pass), image size, layer freezing option, and different data augmentation options. Related works have shown the efficiency of YOLOv5s and YOLOv5m when used for the detection of small objects with similar features. In this case, these two models have been chosen in primary research. Various combinations of training parameters have been used in the training process (Table 2) and a total of 50 models were created and evaluated.

Figure 4. A sample of the dataset used to evaluate the best YOLOv5 model, with several dozen details in one image.

Table 2. Parameters which were investigated in primary research.

Name of the Parameter	Value of the Parameters	Comment
Epoch number	300, 600	The results of our previous research [37] have shown that these parameter options allow for the highest object detection results.
Image size	320, 640 (pixels)	
Batch size	16, 32	
Layers freeze option	10	The layer freeze option [41] is a feature in which the backbone and head layers can be unused in training mode. Primary research has shown that after 10 backbone layers were frozen, training times were reduced by approximately 2 times and construction detail recognition accuracy improved by approximately 1.5 times.
Augmentation	13 options	The different options for data augmentation have been experimentally chosen and analysed [42–44]: **hsv_h**—HSV-Hue augmentation of the image. **hsv_s**—HSV-Saturation augmentation of the image. **hsv_v**—HSV-Value augmentation of the image. **degrees**—rotation (+/− degrees) of the image. **translate**—shifting or moving the objects within the image. **scale**—resizing the input images to different scales. **shear**—geometric deformations by tilting or skewing the images along the x or y axes. **perspective**—simulates perspective changes. **flipud**—flips the image vertically, the top becomes the bottom, and vice versa. **fliplr**—flips the image horizontally, the left side becomes the right side, and vice versa. **mosaic**—combines several images to create a single training sample with a mosaic-like appearance. **mixup**—combines pairs of images and their corresponding object labels to create new training examples. **copy_paste**—involves randomly selecting a portion of one image and pasting it onto another image while maintaining the corresponding object labels.

To find the best parameter combination, the 50 models have been evaluated using 3300 images. The experiments have been named according to the parameters used in the training process. For example, the name of the model *Yolov5s_320_16_300_DefAugm* means that the YOLOv5s model has been used and that the parameters are as follows: image

size—320; batch size—16; epoch number—300; default parameters of data augmentation. The results of the experimental investigation show that, without data augmentation, the detection results are much lower when compared with the results using augmentation. This is true regardless of whether the default or custom options of the data augmentation have been used. Additionally, the first experiments have shown that object detection accuracy increases significantly using the option of 10 backbone layer freeze.

The influence of different combinations of data augmentation options has been analysed. The results of the experiment show that the best detection ratio achieved is equal to 0.4 (40% of correct detection). In this case, the highest number of construction details has been detected no matter which background has been used (322 construction details on the mixed (M) background; 509 construction details on the neutral (N) background; 485 construction details on the white (W) background). Overall, results show that almost every model better detects the construction details on a neutral background. However, it is important to mention that the models have been trained with the construction details, which were placed on a white background. During the primary research, the best parameters to allow one to achieve the highest ratio were found, and are presented in Table 3. These parameters will be used in the main research. The results of all of the experiments are presented in Table 4.

Table 3. The best parameters obtained in the primary research.

Parameter	Value of the Parameter
Image size	320
Batch size	32
Epoch number	300
The layers freeze option	10
Augmentation	hsv_h—0.09; hsv_s—0.7; hsv_v—0.4; degrees—0.125; translate—0; scale—0.5; shear—0.9; perspective—0; flipud –0.5; fliplr—0.5; mosaic—0; mixup—0; copy_paste—0.

3.2. Results of the Main Research

During the training process of YOLOv5, there is the possibility to choose various hyperparameters that could influence the results of object detection. The analysis of related works has shown that many researchers have focused on the following three main parameters: learning rate, momentum, and weight decay. The various values of these parameters are used in scientific papers. Based on other research, our main experiments investigate several combinations of hyperparameters. In the main research, five versions of the YOLOv5 have been trained using the parameters obtained from the primary research results. The hyperparameters used in the main research are presented in Table 5. A total of 135 models have been trained and evaluated.

Table 4. The results of primary research (background: white (W), neutral (N), and mixed (M)).

The Name of the Model	Augmentation Options	Correct Detection on Different Background			Overall Ratio of the Correct Detection
		M	N	W	
Yolov5s_320_16_300_NoAugm	Data augmentations have not been used.	122	133	203	0.14
Yolov5s_320_32_300_NoAugm		118	141	196	0.14
Yolov5s_320_16_600_NoAugm		100	108	199	0.12
Yolov5s_640_16_300_NoAugm		39	12	158	0.06
Yolov5m_320_16_300_NoAugm		126	186	255	0.17
Yolov5m_320_32_300_NoAugm		156	187	226	0.17
Yolov5m_320_16_600_NoAugm		153	139	241	0.16
Yolov5m_640_16_300_NoAugm		28	149	207	0.12
Yolov5s_320_16_300_DefAugm	hsv_h—0.015; hsv_s—0.7; hsv_v—0.4; degrees—0; translate—0.1; scale—0.5; shear—0; perspective—0; flipud—0; fliplr—0.5; mosaic—1; mixup—0; copy_paste—0.	136	140	307	0.18
Yolov5s_320_32_300_DefAugm		173	215	287	0.20
Yolov5s_320_16_600_DefAugm		115	143	359	0.19
Yolov5s_640_16_300_DefAugm		21	70	305	0.12
Yolov5m_320_16_300_DefAugm		82	257	327	0.20
Yolov5m_320_32_300_DefAugm		111	253	305	0.20
Yolov5m_320_16_600_DefAugm		128	166	326	0.19
Yolov5m_640_16_300_DefAugm		51	171	321	0.16
Yolov5m_320_32_600_DefAugm		116	162	353	0.19
Yolov5m_640_32_300_DefAugm		81	145	379	0.18
Yolov5m_640_32_600_DefAugm		94	119	342	0.17

Table 4. *Cont.*

The Name of the Model	Augmentation Options	Correct Detection on Different Background			Overall Ratio of the Correct Detection
		M	N	W	
Yolov5s_320_16_300_Frz_CusAugm	hsv_h—0.5; hsv_s—0.7; hsv_v—0.4; degrees—0.125; translate—0; scale—0.5; shear—0; perspective—0; flipud—0.5; fliplr—0.5; mosaic—1; mixup—0; copy_paste—0.	158	244	309	0.22
Yolov5s_320_32_300_Frz_CusAugm		211	250	329	0.24
Yolov5s_320_32_600_Frz_CusAugm		163	215	325	0.21
Yolov5s_320_16_600_Frz_CusAugm		148	231	313	0.21
Yolov5s_640_32_600_Frz_CusAugm		77	142	278	0.15
Yolov5s_640_32_300_Frz_CusAugm		80	168	306	0.17
Yolov5s_640_16_600_Frz_CusAugm		76	161	280	0.16
Yolov5m_320_16_300_Frz_CusAugm		243	355	347	0.29
Yolov5m_320_32_300_Frz_CusAugm		274	341	361	0.30
Yolov5m_320_16_600_Frz_CusAugm		259	372	362	0.30
Yolov5m_640_32_600_Frz_CusAugm		93	242	321	0.20
Yolov5m_320_32_600_Frz_CusAugm		257	338	374	0.29
Yolov5m_640_32_300_Frz_CusAugm		69	260	347	0.20
Yolov5s_320_16_300_Frz_CusAugm	hsv_h—0.09; hsv_s—0.7; hsv_v—0.4; degrees—0.125; translate—0; scale—0.5; shear—0; perspective—0; flipud—0.5; fliplr—0.5; mosaic—1; mixup—0; copy_paste—0.	160	215	316	0.21
Yolov5s_320_32_300_Frz_CusAugm		152	255	312	0.22
Yolov5m_320_16_300_Frz_CusAugm		271	355	352	0.30
Yolov5m_320_32_300_Frz_CusAugm		225	332	330	0.27
Yolov5m_320_16_600_Frz_CusAugm	hsv_h—0.015; hsv_s—0.7; hsv_v—0.4; degrees—0.125; translate—0; scale—0.5; shear—0; perspective—0; flipud—0.5; fliplr—0.5; mosaic—1; mixup—0; copy_paste—0.	267	362	362	0.30
Yolov5m_320_32_600_Frz_CusAugm		216	324	323	0.26
Yolov5m_320_16_300_Frz_CusAugm		269	347	370	0.30
Yolov5m_320_32_300_Frz_CusAugm		243	377	331	0.29
Yolov5m_320_32_300_Frz_CusAugm	hsv_h—0.09; hsv_s—0.7; hsv_v—0.4; degrees—0.125; translate—0; scale—0.5; shear—0; perspective—0; flipud—0.5; fliplr—0.5; mosaic—0.5; mixup—0; copy_paste—0.	264	411	441	0.34

Table 4. *Cont.*

The Name of the Model	Augmentation Options	Correct Detection on Different Background			Overall Ratio of the Correct Detection
		M	N	W	
Yolov5m_320_32_300_Frz_CusAugm	hsv_h—0.09; hsv_s—0.7; hsv_v—0.4; degrees—0.125; translate—0; scale—0.5; shear—0; perspective—0; flipud—0.5; fliplr—0.5; mosaic—1; mixup—0.5; copy_paste—0.	107	355	378	0.25
Yolov5m_320_32_300_Frz_CusAugm	hsv_h—0.09; hsv_s—0.7; hsv_v—0.4; degrees—0.125; translate—0; scale—0.5; shear—0; perspective—0; flipud—0.5; fliplr—0.5; mosaic—1; mixup—0; copy_paste—0.	138	253	329	0.22
Yolov5m_320_32_300_Frz_CusAugm	hsv_h—0.09; hsv_s—0.7; hsv_v—0.4; degrees—0.125; translate—0; scale—0.5; shear—0; perspective—0; flipud—0.5; fliplr—0.5; mosaic—0; mixup—0; copy_paste—0.	281	497	439	0.37
Yolov5m_320_32_300_Frz_CusAugm	hsv_h—0.09; hsv_s—0.7; hsv_v—0.4; degrees—0.125; translate—0; scale—0.5; shear—0; perspective—0; flipud—0.5; fliplr—0.5; mosaic—0.2; mixup—0; copy_paste—0.	285	453	411	0.35
Yolov5m_320_32_300_Frz_CusAugm	hsv_h—0.09; hsv_s—0.7; hsv_v—0.4; degrees—0.125; translate—0; scale—0.5; shear—0; perspective—0; flipud—0.5; fliplr—0.5; mosaic—0.4; mixup—0; copy_paste—0.	268	400	360	0.31
Yolov5m_320_32_300_Frz_CusAugm	hsv_h—0.09; hsv_s—0.7; hsv_v—0.4; degrees—0.125; translate—0; scale—0.5; shear—0.5; perspective—0; flipud—0.5; fliplr—0.5; mosaic—0; mixup—0; copy_paste—0.	279	503	471	0.38
Yolov5m_320_32_300_Frz_CusAugm	hsv_h—0.09; hsv_s—0.7; hsv_v—0.4; degrees—0.125; translate—0; scale—0.5; shear—0.7; perspective—0; flipud—0.5; fliplr—0.5; mosaic—0; mixup—0; copy_paste—0.	230	472	456	0.35
Yolov5m_320_32_300_Frz_CusAugm	**hsv_h—0.09; hsv_s—0.7; hsv_v—0.4; degrees—0.125; translate—0; scale—0.5; shear—0.9; perspective—0; flipud—0.5; fliplr—0.5; mosaic—0; mixup—0; copy_paste—0.**	**322**	**509**	**485**	**0.40**
Yolov5m_320_32_300_Frz_CusAugm	hsv_h—0.09; hsv_s—0.7; hsv_v—0.4; degrees—0.125; translate—0; scale—0.5; shear—1; perspective—0; flipud—0.5; fliplr—0.5; mosaic—0; mixup—0; copy_paste—0.	233	492	459	0.36

Table 5. Hyperparameters used in the main research.

Name of the Parameter	Value of the Parameters
Learning rate (lr0)	0.01, 0.001, 0.0001
Momentum (m)	0.9, 0.937, 0.95
Weight decay (wd)	0.0001, 0.0005, 0.0007
Other options	The other values of the parameters have been left as default: lrf—0.01; warmup_epochs—3; warmup_momentum—0.8; warmup_bias_lr—0.05; box—0.05; cls—0.5; cls_pw—1; obj—1; obj_pw—1; iou_t—0.2; anchor_t—4; anchors—3; fl_gamma—0.

The results of the main research show that, when using various combinations of hyperparameters, the highest obtained correct detection ratio of construction details is equal to 0.5012 (50%). In this case, the YOLOv5l model was used. The model was trained with a learning rate equal to 0.001, a momentum of 0.95, and a weight decay of 0.0007. In some cases, the correct detection ratio is equal to 0. The lowest correct detection ratio was obtained using YOLOv5n. The highest correct detection ratio obtained for each YOLOv5 model (n, s, m, l, x) are presented in Figure 5.

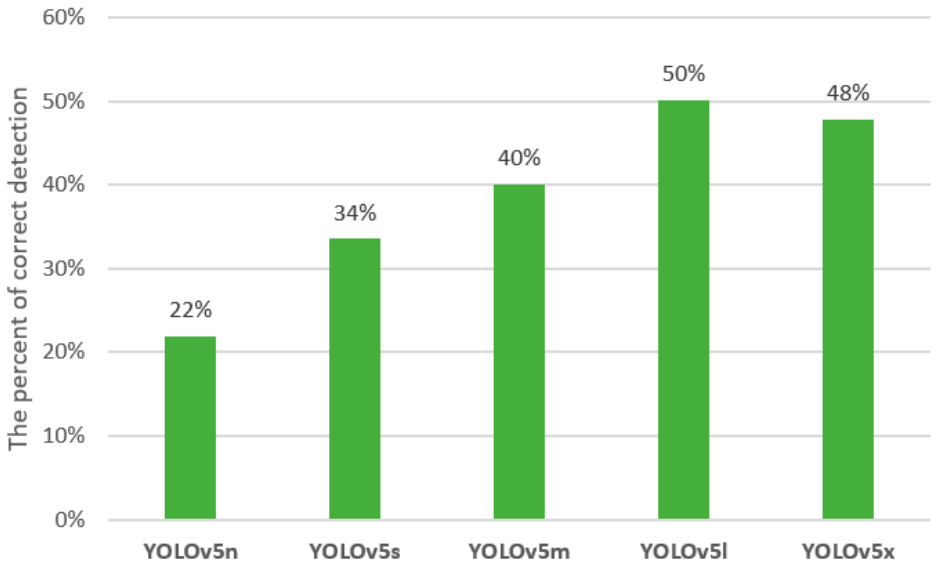

Figure 5. The highest correct detection ratio of each YOLOv5 model.

As one can see in Figure 5, the smallest correct detection ratio was obtained using YOLOv5n (22%). The difference between the results of YOLOv5s (34%) and YOLOv5m is equal to 6%, while better results were obtained using YOLOv5m (40%). The results obtained using YOLOv5x (48%) are slightly lower compared with the results of YOLOv5l (50%). In addition, in Figure 6, the curves of precision, recall, mAP@0.5, and map@0.5:0.95 are presented.

One can see (Figure 6) that, until approximately 200 epochs, the model is still training, and after 200 epochs there is no progress. The recall and the precision metrics of the model are close to 1. In the case of the map@0.5 metric, the model is close to value 1 after 100 epochs. The map@0.5:0.95 metric shows that the accuracy during all 300 training epochs continues to increase.

The results of all of the main research experiments are presented in Table 6. In Figure 7 the confusion matrices of the best model on three different backgrounds are presented. As one can see, the smallest number of correct detections was on the mixed background (497). Using this background, two details were not detected at all and were recognized as different construction details. Furthermore, in this case, many details were not assigned to

any classes at all, which shows that details merge in the mixed background. On the neutral background, the number of correct detections is larger (562), though the same construction detail as in the case of the mixed background was nevertheless recognized incorrectly. All of the details have been correctly recognized on the white background at least once. On a white background, the number of correct detections is largest (595), therefore in this case, 54% of the construction details were recognized correctly.

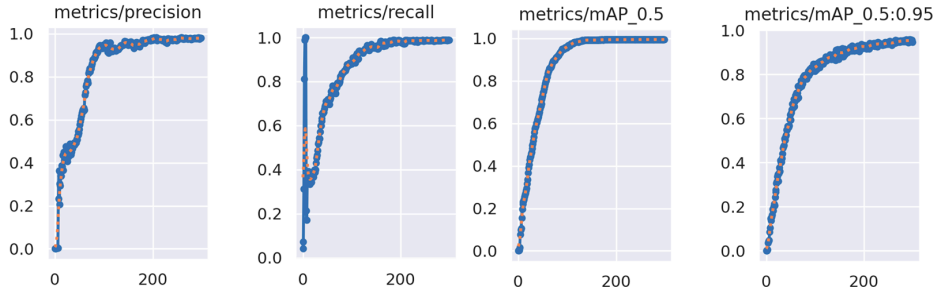

Figure 6. The evaluation of the YOLOv5l model.

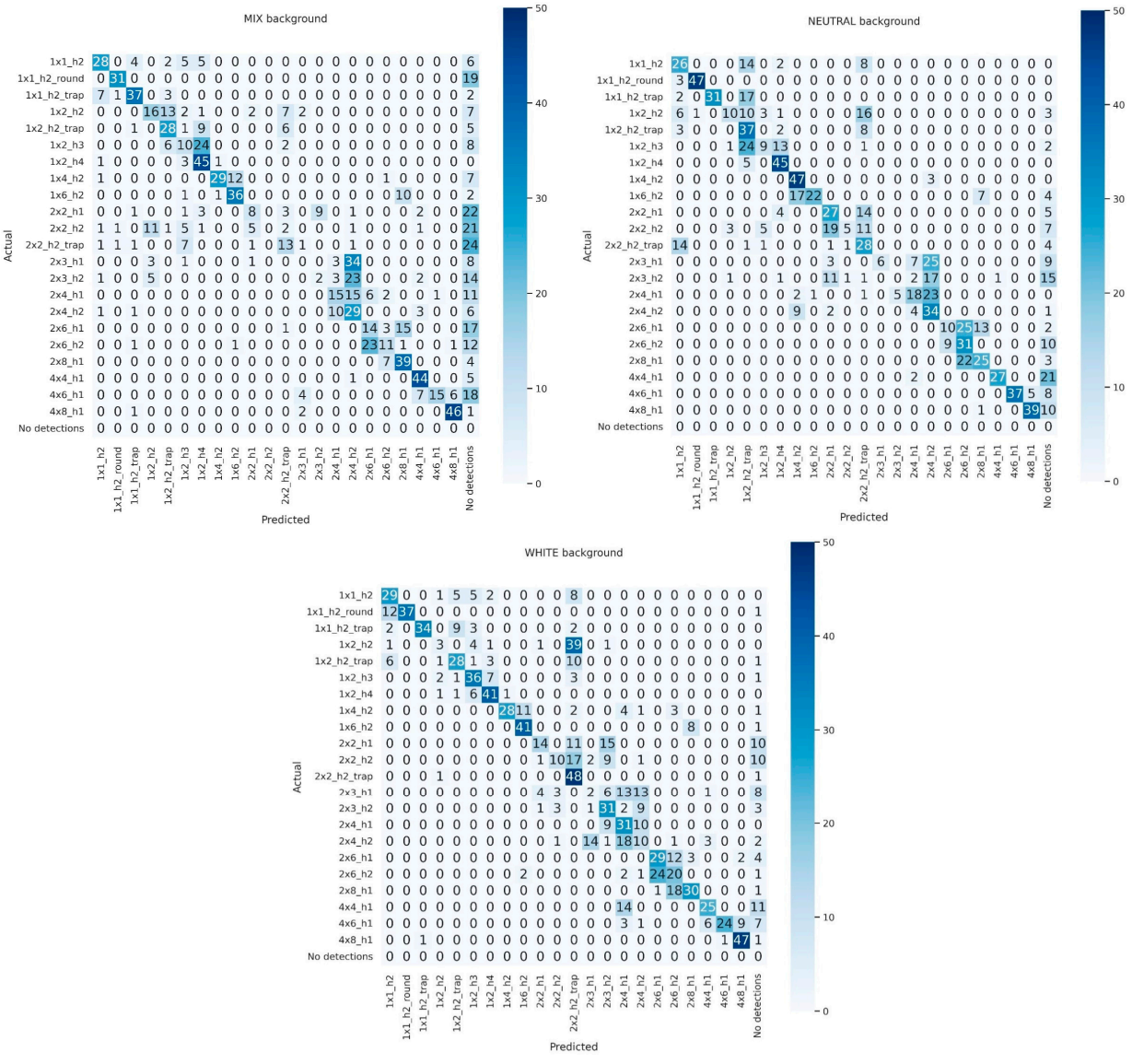

Figure 7. The confusion matrices of the best obtained model.

Table 6. The results of the main research (parameters: learning rate (lr0), momentum (m), weight decay (wd). Background: white (W), neutral (N), and mixed (M)).

Parameters			YOLOv5n				YOLOv5s				YOLOv5m				YOLOv5l				YOLOv5x			
lr0	*m*	*wd*	Correct Detection on Different Background			Overall Ratio of the Correct Detection	Correct Detection on Different Background			Overall Ratio of the Correct Detection	Correct Detection on Different Background			Overall Ratio of the Correct Detection	Correct Detection on Different Background			Overall Ratio of the Correct Detection	Correct Detection on Different Background			Overall Ratio of the Correct Detection
			M	N	W		M	N	W		M	N	W		M	N	W		M	N	W	
0.01	0.937	0.0007	111	136	393	0.1939	252	357	497	0.3352	215	494	458	0.3536	410	547	525	0.4491	432	457	458	0.4082
0.01	0.937	0.0005	142	107	414	0.2009	216	318	469	0.3039	325	509	486	0.4000	405	529	513	0.4385	433	510	451	0.4224
0.01	0.937	0.0001	139	165	416	0.2182	215	364	508	0.3294	280	491	481	0.3794	392	515	492	0.4239	420	497	474	0.4215
0.01	0.95	0.0007	158	163	378	0.2118	233	366	503	0.3339	314	504	449	0.3839	339	497	507	0.4070	436	488	468	0.4218
0.01	0.95	0.0005	149	137	384	0.2030	194	329	455	0.2964	239	513	438	0.3606	392	483	525	0.4242	437	507	522	0.4442
0.01	0.95	0.0001	124	123	386	0.1918	225	357	482	0.3224	268	475	462	0.3652	367	510	500	0.4173	438	480	466	0.4194
0.01	0.9	0.0007	129	109	396	0.1921	197	336	496	0.3118	296	470	463	0.3724	356	527	492	0.4167	469	487	466	0.4309
0.01	0.9	0.0005	142	127	376	0.1955	197	352	502	0.3185	299	502	509	0.3970	395	535	541	0.4458	450	486	476	0.4279
0.01	0.9	0.0001	108	127	364	0.1815	181	344	492	0.3082	245	457	482	0.3588	420	533	535	0.4509	458	516	460	0.4345
0.001	0.937	0.0007	64	28	218	0.0939	117	205	336	0.1994	303	449	509	0.3821	490	542	576	0.4873	453	569	544	0.4745
0.001	0.937	0.0005	65	35	213	0.0948	115	197	337	0.1967	290	443	506	0.3755	494	543	571	0.4873	466	565	536	0.4748
0.001	0.937	0.0001	61	36	217	0.0952	110	194	333	0.1930	294	435	494	0.3706	494	541	571	0.4867	462	554	538	0.4709
0.001	0.95	0.0007	76	35	261	0.1127	132	217	360	0.2148	307	460	523	0.3909	497	562	595	**0.5012**	467	571	537	0.4773
0.001	0.95	0.0005	85	38	264	0.1173	135	214	350	0.2118	312	452	527	0.3912	498	557	584	0.4967	475	572	532	0.4785
0.001	0.95	0.0001	75	36	268	0.1148	140	216	365	0.2185	323	451	533	0.3961	492	560	587	0.4967	464	565	531	0.4727
0.001	0.9	0.0007	23	12	151	0.0564	73	121	248	0.1339	245	392	417	0.3194	422	503	526	0.4397	442	535	525	0.4552
0.001	0.9	0.0005	23	12	154	0.0573	69	120	244	0.1312	248	390	410	0.3176	421	499	516	0.4352	467	533	547	0.4688
0.001	0.9	0.0001	20	13	152	0.0561	68	117	248	0.1312	236	390	418	0.3164	420	495	523	0.4358	452	545	534	0.4639
0.0001	0.937	0.0007	0	0	0	0.0000	0	4	16	0.0061	8	32	51	0.0276	47	33	35	0.0348	39	100	82	0.0670
0.0001	0.937	0.0005	0	0	0	0.0000	0	4	16	0.0061	8	33	49	0.0273	46	33	35	0.0345	34	101	82	0.0658
0.0001	0.937	0.0001	0	0	0	0.0000	1	5	19	0.0076	8	32	50	0.0273	47	34	39	0.0364	37	103	83	0.0676
0.0001	0.95	0.0007	0	0	0	0.0000	1	5	27	0.0100	14	53	73	0.0424	66	104	92	0.0794	82	193	126	0.1215
0.0001	0.95	0.0005	0	0	0	0.0000	1	5	28	0.0103	12	55	73	0.0424	67	105	93	0.0803	82	188	127	0.1203
0.0001	0.95	0.0001	0	0	0	0.0000	0	4	21	0.0076	14	54	72	0.0424	66	105	93	0.0800	82	193	125	0.1212
0.0001	0.9	0.0007	0	0	0	0.0000	0	2	4	0.0018	1	6	20	0.0082	14	3	1	0.0055	10	34	26	0.0212
0.0001	0.9	0.0005	0	0	0	0.0000	0	2	4	0.0018	1	5	22	0.0085	13	3	1	0.0052	11	32	24	0.0203
0.0001	0.9	0.0001	0	0	0	0.0000	1	3	4	0.0024	1	6	22	0.0088	13	3	1	0.0052	11	32	25	0.0206

4. Discussion

This experimental investigation has shown the importance of training parameters and hyperparameter selection in the model training process. In this investigation, a total of 185 models were trained. The main problem with object detection tasks is that there are many different options for how to train the models, so it is hard to consider all of them. This is especially so when training each model takes a long time. In this research, many different parameter combinations were evaluated. The results may be useful for tasks related to the detection of objects with similar features. The analysed dataset has 22 categories. Some of the construction details could look identical to different categories due to the different photoshoot angles. This means that the results are not as good, but they are still promising and are valuable for future research. Due to the complexity of the task, the detection of construction details may be useful when evaluating the efficiency and performance of the model.

The model obtained in the main research could be used to develop a recommendation for building construction. It would detect details from an image and suggest possible construction. The system or application could be implemented in a mobile environment. An additional experiment was performed in which 150 new images were fed to the best obtained models. As mentioned, images of several dozen construction details were placed and photographed on the three different backgrounds that were used in the primary and main research. A sample of the construction details detection results is presented in Figure 8.

Figure 8. A sample of construction detail detection in real-world simulation.

The five best YOLOv5 models obtained in the main research (Figure 5) were evaluated using 150 images. The full results of the correct detection ratio of each construction detail are presented in Table 7. As one can see, the worst detection results are obtained when a mixed background is used. Only in the case of YOLOv5m, was the correct detection ratio larger than 0 and almost all details were recognized at least once. The highest correct detection ratio is obtained when using the white background. Overall, results show that some construction details, like 1x1_h2_round, 2x3_h1 and 2x3_h2, were not detected at all or detected by only few models. The details 2x3_h1 and 2x3_h2 are differed in terms of their height but can look identical from other angles. Some of the construction details were recognized correctly all the time by the YOLOv5m model, for example, 2x8_h1, and 4x4_h1 when the neutral background is used. Summarized results of the additional research show that the YOLOv5m model recognizes the highest number compared with the other four models.

Table 7. Correct detection ratio of each model type on white (W), neutral (N), and mixed (M) backgrounds using 150 images.

Name of the Construction Detail	YOLOv5n			YOLOv5s			YOLOv5m			YOLOv5l			YOLOv5x		
	M	N	W	M	N	W	M	N	W	M	N	W	M	N	W
2x2_h2	0.00	0.00	0.13	0.00	0.04	0.00	0.07	0.14	0.19	0.00	0.00	0.00	0.01	0.00	0.00
1x2_h2	0.00	0.00	0.01	0.00	0.01	0.05	0.01	0.01	0.04	0.00	0.02	0.00	0.00	0.02	0.00
2x3_h1	0.00	0.00	0.00	0.00	0.15	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
2x4_h1	0.02	0.02	0.14	0.00	0.02	0.09	0.05	0.16	0.23	0.00	0.02	0.18	0.00	0.09	0.07
2x4_h2	0.03	0.04	0.70	0.03	0.11	0.59	0.11	0.21	0.62	0.22	0.38	0.53	0.01	0.07	0.41
2x2_h2_trap	0.00	0.00	0.18	0.04	0.05	0.44	0.08	0.33	0.38	0.01	0.08	0.37	0.03	0.11	0.55
2x3_h2	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.16	0.00	0.00	0.11	0.00	0.00	0.08
1x2_h2_trap	0.00	0.00	0.21	0.00	0.00	0.11	0.00	0.16	0.21	0.00	0.16	0.00	0.11	0.05	0.05
2x2_h1	0.00	0.00	0.10	0.00	0.30	0.40	0.00	0.20	0.30	0.10	0.00	0.10	0.00	0.00	0.20
1x2_h3	0.00	0.00	0.05	0.00	0.00	0.05	0.00	0.00	0.00	0.00	0.00	0.05	0.00	0.00	0.09
1x4_h2	0.00	0.00	0.06	0.00	0.22	0.17	0.11	0.22	0.17	0.06	0.17	0.06	0.06	0.06	0.17
4x6_h1	0.00	0.00	0.00	0.00	0.00	0.00	0.50	0.50	0.50	0.00	0.00	0.00	0.00	0.00	0.00
1x1_h2	0.00	0.00	0.03	0.00	0.08	0.11	0.14	0.19	0.16	0.00	0.14	0.05	0.05	0.00	0.16
2x6_h2	0.00	0.00	0.15	0.00	0.00	0.12	0.15	0.09	0.18	0.00	0.09	0.09	0.03	0.12	0.09
2x8_h1	0.00	0.13	0.50	0.13	0.25	0.25	0.38	1.00	0.50	0.38	0.63	0.00	0.25	0.75	0.38
1x6_h2	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.33	0.00	0.00	0.67
2x6_h1	0.00	0.00	0.17	0.00	0.17	0.17	0.00	0.17	0.33	0.00	0.00	0.00	0.00	0.17	0.17
1x1_h2_round	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
1x2_h4	0.00	0.00	0.50	0.00	0.67	0.67	0.33	0.83	0.83	0.00	0.33	0.17	0.00	0.00	0.17
4x8_h1	0.00	0.00	0.00	0.00	0.00	0.50	0.00	0.50	0.50	0.50	0.00	0.50	0.00	0.00	0.00
1x1_h2_trap	0.00	0.00	0.00	0.00	0.00	0.00	0.17	0.00	0.67	0.17	0.00	0.00	0.00	0.00	0.17
4x4_h1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.50	0.00	0.50	0.00	0.00	0.50
2x2_h2	0.00	0.00	0.13	0.00	0.04	0.00	0.07	0.14	0.19	0.00	0.00	0.00	0.01	0.00	0.00
1x2_h2	0.00	0.00	0.01	0.00	0.01	0.05	0.01	0.01	0.04	0.00	0.02	0.00	0.00	0.02	0.00

This research has some limitations, because the results have not been compared with the other object detection models the experimental investigation has been based only on the YOLOv5 algorithm. Additionally, it is not possible to ensure that the same results could be obtained using another dataset that has similar features. The results can still depend on many different aspects, for example, the angle of the image taken, the noise appearing around the construction details, etc. However, the results may still be useful to other researchers. The experimental investigation has shown the importance of the freeze option in the training process and the use of nondefault parameters to obtain higher object detection results.

5. Conclusions

In this paper, the influences of the training parameters and hyperparameters of YOLOv5 on the detection of construction details were analysed. Construction details were chosen due to the task complexity when similar feature data are analysed. In some cases, the construction details appear to be identical. All depends on the angle of the shot used, which in turn depends on the point of view of the camera. During the research, five models of YOLOv5 were analysed. A total of 185 models were trained and evaluated. Model efficiencies were tested using a total of 3300 images placed on 3 different complexity backgrounds. The influence of five training parameters (image size, batch size, epoch size, layer freeze option, and data augmentation) and three hyperparameters (learning rate,

momentum, and weight decay) was analysed. All of the parameters mentioned were used in various combinations.

The results of the experimental investigation show that the best parameters for the detection of construction details are as follows: coloured images; image size—320; batch size—32; epoch number—300; layer freeze option—10; data augmentation—on; learning rate—0.001; momentum—0.95; and weight decay—0.0007. In this case, the percentage of correct detection is equal to 50%, regardless of which background is used. The correct detection results of the model only on the white background are equal to 54%. Experimental investigation has shown that the smallest detection results are obtained when a mixed background is used. The main reason for this is that some details merge with the background and that, therefore, the models cannot detect the construction details. Additional research using several dozen construction details in the same image (on three different backgrounds) have shown that the YOLOv5m model correctly recognizes the highest number of structural details.

The number of correct detection results can be increased if the YOLOv5 model is used to localize the structure details in the image. A second step would be to use an additional binary classification to find the correct details of the structure. This could be implemented in the future to find the best way in which to detect similar construction details at different angles.

Author Contributions: Conceptualization, T.K. and P.S.; methodology, P.S.; validation, T.K. and E.P.; formal analysis, T.K. and P.S.; data curation, T.K. and P.S.; writing—original draft preparation, T.K., E.P., P.S. and B.P.; writing—review and editing P.S. and B.P.; visualization, T.K., E.P. and P.S.; supervision, P.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author due to the large capacity of the data (7.21 GB).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Jucevičius, J.; Treigys, P.; Bernatavičienė, J.; Briedienė, R.; Naruševičiūtė, I.; Trakymas, M. Investigation of MRI prostate localization using different MRI modality scans. In Proceedings of the 2020 IEEE 8th Workshop on Advances in Information, Electronic and Electrical Engineering (AIEEE), Vilnius, Lithuania, 22–24 April 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 1–5.
2. Wang, X.; Chen, H.; Gan, C.; Lin, H.; Dou, Q.; Tsougenis, E.; Heng, P.A. Weakly supervised deep learning for whole slide lung cancer image analysis. *IEEE Trans. Cybern.* **2019**, *50*, 3950–3962. [CrossRef]
3. Shabbir, A.; Rasheed, A.; Shehraz, H.; Saleem, A.; Zafar, B.; Sajid, M.; Shehryar, T. Detection of glaucoma using retinal fundus images: A comprehensive review. *Math. Biosci. Eng.* **2021**, *18*, 2033–2076. [CrossRef] [PubMed]
4. Elangovan, P.; Nath, M.K. Glaucoma assessment from color fundus images using convolutional neural network. *Int. J. Imaging Syst. Technol.* **2021**, *31*, 955–971. [CrossRef]
5. Amyar, A.; Modzelewski, R.; Li, H.; Ruan, S. Multi-task deep learning based CT imaging analysis for COVID-19 pneumonia: Classification and segmentation. *Comput. Biol. Med.* **2020**, *126*, 104037. [CrossRef]
6. Toğaçar, M.; Ergen, B.; Cömert, Z.; Özyurt, F. A deep feature learning model for pneumonia detection applying a combination of mRMR feature selection and machine learning models. *IRBM* **2020**, *41*, 212–222. [CrossRef]
7. Stefanovič, P.; Ramanauskaitė, S. Travel Direction Recommendation Model Based on Photos of User Social Network Profile. *IEEE Access* **2023**, *11*, 28252–28262. [CrossRef]
8. Li, C.; Li, L.; Jiang, H.; Weng, K.; Geng, Y.; Li, L.; Wei, X. YOLOv6: A single-stage object detection framework for industrial applications. *arXiv* **2022**, arXiv:2209.02976.
9. Zhou, X.; Xu, X.; Liang, W.; Zeng, Z.; Shimizu, S.; Yang, L.T.; Jin, Q. Intelligent small object detection for digital twin in smart manufacturing with industrial cyber-physical systems. *IEEE Trans. Ind. Inform.* **2022**, *18*, 1377–1386. [CrossRef]
10. Li, C.; Wang, R.; Li, J.; Fei, L. Face detection based on YOLOv3. In *Recent Trends in Intelligent Computing, Communication and Devices: Proceedings of ICCD 2018*; Springer: Singapore, 2020; pp. 277–284.
11. Chen, W.; Huang, H.; Peng, S.; Zhou, C.; Zhang, C. YOLO-face: A real-time face detector. *Vis. Comput.* **2021**, *37*, 805–813. [CrossRef]

12. Ye, X.; Liu, Y.; Zhang, D.; Hu, X.; He, Z.; Chen, Y. Rapid and Accurate Crayfish Sorting by Size and Maturity Based on Improved YOLOv5. *Appl. Sci.* **2023**, *13*, 8619. [CrossRef]
13. Shi, H.; Xiao, W.; Zhu, S.; Li, L.; Zhang, J. CA-YOLOv5: Detection model for healthy and diseased silkworms in mixed conditions based on improved YOLOv5. *Int. J. Agric. Biol. Eng.* **2024**, *16*, 236–245. [CrossRef]
14. Hui, Y.; You, S.; Hu, X.; Yang, P.; Zhao, J. SEB-YOLO: An Improved YOLOv5 Model for Remote Sensing Small Target Detection. *Sensors* **2024**, *24*, 2193. [CrossRef] [PubMed]
15. Zhang, J.; Xie, J.; Zhang, F.; Gao, J.; Yang, C.; Song, C.; Rao, W.; Zhang, Y. Greenhouse tomato detection and pose classification algorithm based on improved YOLOv5. *Comput. Electron. Agric.* **2024**, *216*, 108519. [CrossRef]
16. Feng, S.; Qian, H.; Wang, H.; Wang, W. Real-time object detection method based on YOLOv5 and efficient mobile network. *J. Real-Time Image Process.* **2024**, *21*, 56. [CrossRef]
17. Reddy, B.K.; Bano, S.; Reddy, G.G.; Kommineni, R.; Reddy, P.Y. Convolutional network based animal recognition using YOLO and Darknet. In Proceedings of the 2021 6th International Conference on Inventive Computation Technologies (ICICT), Coimbatore, India, 20–22 January 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 1198–1203.
18. Dewi, C.; Chen, R.C.; Jiang, X.; Yu, H. Deep convolutional neural network for enhancing traffic sign recognition developed on Yolo V4. *Multimed. Tools Appl.* **2022**, *81*, 37821–37845. [CrossRef]
19. Hameed, K.; Chai, D.; Rassau, A. A sample weight and adaboost cnn-based coarse to fine classification of fruit and vegetables at a supermarket self-checkout. *Appl. Sci.* **2020**, *10*, 8667. [CrossRef]
20. Construction Details Dataset. Available online: https://app.box.com/s/j420ld0wo89hvh6np1rc3z9t1e65yg2k (accessed on 13 January 2024).
21. Kwon, H.J.; Kim, H.G.; Lee, S.H. Pill detection model for medicine inspection based on deep learning. *Chemosensors* **2021**, *10*, 4. [CrossRef]
22. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask R-CNN. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *42*, 386–397. [CrossRef]
23. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
24. Tan, L.; Huangfu, T.; Wu, L.; Chen, W. Comparison of RetinaNet, SSD, and YOLO v3 for real-time pill identification. *BMC Med. Inform. Decis. Mak.* **2021**, *21*, 324. [CrossRef]
25. Ou, Y.Y.; Tsai, A.C.; Wang, J.F.; Lin, J. Automatic drug pills detection based on convolution neural network. In Proceedings of the 2018 International Conference on Orange Technologies (ICOT), Nusa Dua, Indonesia, 23–26 October 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 1–4.
26. Chollet, F. Xception: Deep learning with depthwise separable convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 1251–1258.
27. Ou, Y.Y.; Tsai, A.C.; Zhou, X.P.; Wang, J.F. Automatic drug pills detection based on enhanced feature pyramid network and convolution neural networks. *IET Comput. Vis.* **2020**, *14*, 9–17. [CrossRef]
28. Saeed, F.; Ahmed, M.J.; Gul, M.J.; Hong, K.J.; Paul, A.; Kavitha, M.S. A robust approach for industrial small-object detection using an improved faster regional convolutional neural network. *Sci. Rep.* **2021**, *11*, 23390. [CrossRef] [PubMed]
29. Yildiz, E.; Wörgötter, F. DCNN-based screw detection for automated disassembly processes. In Proceedings of the 2019 15th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS), Sorrento, Italy, 26–29 November 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 187–192.
30. Mangold, S.; Steiner, C.; Friedmann, M.; Fleischer, J. Vision-based screw head detection for automated disassembly for remanufacturing. *Procedia CIRP* **2022**, *105*, 1–6. [CrossRef]
31. Xiao, Y.; Tian, Z.; Yu, J.; Zhang, Y.; Liu, S.; Du, S.; Lan, X. A review of object detection based on deep learning. *Multimed. Tools Appl.* **2020**, *79*, 23729–23791. [CrossRef]
32. Zou, X. A review of object detection techniques. In Proceedings of the 2019 International Conference on Smart Grid and Electrical Automation (ICSGEA), Xiangtan, China, 10–11 August 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 251–254.
33. Li, K.; Cao, L. A review of object detection techniques. In Proceedings of the 2020 5th International Conference on Electromechanical Control Technology and Transportation (ICECTT), Nanchang, China, 15–17 May 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 385–390.
34. Terven, J.; Cordova-Esparza, D. A comprehensive review of YOLO: From YOLOv1 to YOLOv8 and beyond. *arXiv* **2023**, arXiv:2304.00501.
35. Zhao, Y.; Shi, Y.; Wang, Z. The improved YOLOV5 algorithm and its application in small target detection. In Proceedings of the International Conference on Intelligent Robotics and Applications, Kuala Lumpur, Malaysia, 5–7 November 2020; Springer: Berlin/Heidelberg, Germany, 2022; pp. 679–688.
36. Dlužnevskij, D.; Stefanovič, P.; Ramanauskaite, S. Investigation of YOLOv5 Efficiency in iPhone Supported Systems. *Balt. J. Mod. Comput.* **2021**, *9*, 333–344. [CrossRef]
37. Kvietkauskas, T.; Stefanovič, P. Influence of Training Parameters on Real-Time Similar Object Detection Using YOLOv5s. *Appl. Sci.* **2023**, *13*, 3761. [CrossRef]
38. Isa, I.S.; Rosli, M.S.A.; Yusof, U.K.; Maruzuki, M.I.F.; Sulaiman, S.N. Optimizing the hyperparameter tuning of YOLOv5 for underwater detection. *IEEE Access* **2022**, *10*, 52818–52831. [CrossRef]

39. Mantau, A.J.; Widayat, I.W.; Adhitya, Y.; Prakosa, S.W.; Leu, J.S.; Köppen, M. A GA-Based Learning Strategy Applied to YOLOv5 for Human Object Detection in UAV Surveillance System. In Proceedings of the 2022 IEEE 17th International Conference on Control & Automation (ICCA), Naples, Italy, 27–30 June 2022; IEEE: Piscataway, NJ, USA, 2022; pp. 9–14.
40. Jocher, G.; Chaurasia, A.; Stoken, A.; Borovec, J.; NanoCode012; Kwon, Y.; Michael, K.; Xie, T.; Fang, J.; imyhxy; et al. *ultralytics/yolov5: V7.0—YOLOv5 SOTA Realtime Instance Segmentation*; Zenodo: Geneva, Switzerland, 2021. [CrossRef]
41. Huang, Q.; Zhou, Y.; Yang, T.; Yang, K.; Cao, L.; Xia, Y. A Lightweight Transfer Learning Model with Pruned and Distilled YOLOv5s to Identify Arc Magnet Surface Defects. *Appl. Sci.* **2023**, *13*, 2078. [CrossRef]
42. Ultralytics. Hyperparameter Tuning. Ultralytics YOLOv8 Docs. 3 March 2024. Available online: https://docs.ultralytics.com/guides/hyperparameter-tuning (accessed on 13 January 2024).
43. Ultralytics. "Train". Ultralytics YOLOv8 Docs. 30 March 2024. Available online: https://docs.ultralytics.com/modes/train/#train-settings (accessed on 24 January 2024).
44. Ruman. YOLO Data Augmentation Explained–Ruman–Medium. Medium. 4 June 2023. Available online: https://rumn.medium.com/yolo-data-augmentation-explained-turbocharge-your-object-detection-model-94c33278303a (accessed on 24 January 2024).