The Influence of YOLOv5 Hyperparameters for Construction Details Detection

Tautvydas Kvietkauskas

Vilnius Gediminas Technical University, Department of Information Technology,
Saulėtekio al. 11, LT-10223 Vilnius, Lithuania
tautvydas.kvietkauskas@stud.vilniustech.lt

Abstract. Computer vision has become a fundamental area of interest in recent decades. Each area has unique data which object detection methods can analyse. However, it is important to find the most suitable parameters for the model that detects different object groups. In this research has been investigated the influence of pre-trained YOLOv5 (nano (n), small (s), medium (m), large (l), extra-large (x)) models, hyperparameters (learning rate, momentum, and weight decay) and different image augmentation (hsv_h, degrees, translate, flipud, mosaic, mixup, shear, perspective) efficiency for similar construction details detection. A newly collected dataset with twenty-two labelled categories of construction details was prepared. A total of 270 models were trained and evaluated. Every model was evaluated with 3,300 test images which backgrounds were mixed, neutral, and white backgrounds. The most accurate model was YOLOv5l with learning rate – 0.001, momentum – 0.950 and weight decay – 0.0001. This model achieved – 0.5015 (50.15%) accuracy.

Keywords: YOLOv5, object detection, hyperparameters, constructions details.

1 Introduction

Computer vision is a rapidly developing field of artificial intelligence designed to enable machines to interpret and understand visual information from the surrounding environment. By simulating the human visual system, computer vision systems can extract meaningful insights from images and videos, revolutionising various industries. From house number recognition [17] to medical pills [1] and drones [2], computer vision algorithms play an important role in analysing and interpreting visual data.

To get the highest object detection results, it is necessary to have the proper data set. The most difficult problem in object recognition is similar-looking objects. Similarity can be seen in shape, colour, and size. Object detection, which is used to detect what fruit [3] is bought at self-service

DOI: https://doi.org/10.15388/LMITT.2024.11

checkouts, can easily make mistakes regarding the type of apples because of similarities in appearance. Image shooting angles, shadows, lighting, distance, and other additional factors make different types of objects look the same. The same problem exists when trying to detect construction details.

This study looked at several types of pre-trained YOLOv5 models using newly gathered construction detail datasets [12]. The training dataset includes 440 photos (22 categories, each with 20 images). For testing, 3300 photos (22 construction details on mixed, neutral, and white backgrounds; 50 photographs for each group). The experimental inquiry was divided into two stages: main and additional. In the main

stage, 135 experiments were carried out using the YOLOv5 models nano, small, medium, large, and extra-large. The optimal learning rate, weight decay, and momentum were observed. According to the main stage learning curves and accuracy, an additional 135 models were built and evaluated for potential accuracy improvements. The originality of this work is a thorough investigation of 270 experiments in which various YOLOv5 hyperparameters were analysed. The findings may be useful in other applications requiring the detection of similar feature items. Furthermore, experimental results can help build the construction recommendation model. It can be practically applied in a smartphone app that suggests various constructions based on observed details in real time.

2 Objects Detection Methods Review

Popular object detection models are SSD [4], Faster R-CNN [5] and YOLO group [6-10]. SSD as a single-shot detector, efficiently predicts bounding boxes and class probabilities simultaneously, striking a balance between speed and accuracy suitable for real-time applications. In contrast, Faster R-CNN adopts a two-stage architecture, leveraging a Region Proposal Network (RPN) to generate region proposals before refining and classifying them. While offering higher accuracy, this approach sacrifices speed and is more suitable for tasks requiring precision. YOLO predicts bounding boxes and class probabilities for each grid cell, making it incredibly fast and ideal for real-time applications, particularly for small objects.

Three different object detection methods have been examined using medication pills. Correct identification is essential for safe medicine administration. In a real-time pill recognition investigation, Faster R-CNN, SSD, and YOLOv3 recognition algorithms were employed to assess recognition accuracy and speed. The tablets were randomly arranged, and 5,131 photos were captured. The dataset contains 70 capsules and 191 non-capsules. The training parameters for each algorithm have been adjusted to 64 batches, 16 sub-divisions, 0.001 learning rate, 0.9 momentum, and 0.0001 weight decay. Based on these findings, researchers determined that YOLOv3 is faster than SSD and Faster-R-CNN. According to the mAP indication, Faster R-CNN appears to be the highest (82.89%), however, its detection rate is just 17 frames per second. The SSD-based model achieved an average of 32 frames per second and 82.71% mAP. Compared to recent models, the YOLOv3 achieves only 80.69% mAP, but it can significantly improve detection rates and attain real-time performance at 51 frames per second. As a result, it was determined that the YOLO group model would be appropriate for real-time pill detection because it can recognize pills quickly and with reasonable accuracy [1].

To efficiently use recognition to identify road traffic items, the optimum object identification method must be discovered. Many object identification algorithms have recently been released, although there is little material comparing algorithms, such as YOLOv5, which is focused on road traffic objects. The article investigates SSD MobileNetv2, YOLOv3, YOLOv4, and YOLOv5 for real-time street-level item recognition. The dataset comprised 3,169 pictures with 24,102 annotations. Five classes were identified: automobiles (16446 comments), traffic lights (4790), crossings (1756), trucks (761), and motorcyclists (349). The dataset was separated into three parts: training (2010), validation (586), and testing (573). Each image was rescaled with HSV scaling (-25 to 25), noise augmentation (up to 5% of pixels), and cut-out (3 cells at 10% each) was also used. During the training phase of YOLO group algorithms, the SGD optimizer was set together with 100 epochs. Meanwhile, 32000 training steps were scheduled for the SSD MobileNetv2 FPN-lite. YOLOv4 had comparatively lower F1 scores, accuracy, and mAP than YOLOv5l and YOLOv3. The data suggest that YOLOv5l is the most accurate (Precision – 0.780) algorithm for this experiment. However, when compared to the other YOLO models, the mAP rates were not significantly different (SSD – 0.315, YOLOv3 – 0.313, YOLOv4 – 0.304, YOLOv5l – 0.313, YOLOv5s – 0.260). Also, YOLOv4 was the slowest of the models. Meanwhile, YOLOv5 performs better than previous YOLO versions in terms of mAP@.5 and inference time. SSD MobileNetv2 FPN-lite had the lowest mAP@.5

performance of any of the object identification algorithms tested in this trial, with a score of 0.315. However, it is the fastest algorithm in the trial, taking 6.3 milliseconds. The second quickest object detection technique is YOLOv5s – 8.50 milliseconds, an F1-Score of 0.579, and a mAP@.5 of 0.530, which is just 11% worse, and mAP@.5:95 is 17% poorer than YOLOv5l, the most accurate model in this experiment. In conclusion, YOLOv5l is the most accurate algorithm [8].

According to other research about SSD, YOLO, and Faster R-CNN, it was decided to choose YOLOv5 due to the training time and accuracy ratio, the lowest detection loss, prediction time and the best stability in the YOLO group.

3 Analysis of New Dataset

The first dataset of four [12] has been used only for the training phase. Each of the different twenty-two classes of details was photographed only on a white background (Figure 1, first image). Every detail was rotated twenty times and photos from the new angle were taken. The training data set contains 440 images because pre-trained [13, 14] YOLOv5 models were used. For the testing dataset, each construction detail was photographed from fifty different angles on white (W), neutral (N) and mixed (M) backgrounds (Figure 1, second image). These three different backgrounds simulated the possible real-world environment. In general, 1100 images for each background have been prepared, in total 3300 different images.

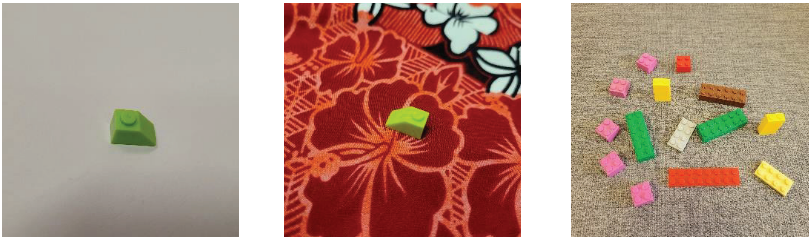

Figure 1. The samples of the datasets.

4 Experiment Methodology

Investigation of various hyperparameters efficiency for the accuracy of YOLOv5 has been done. In the main research, 135 models were trained and evaluated to find the highest accuracy in construction details detection.

additional research in which other 135 models were trained and evaluated according to main research training specifications, statistics and learning curves. The research workflow is in Figure 2.

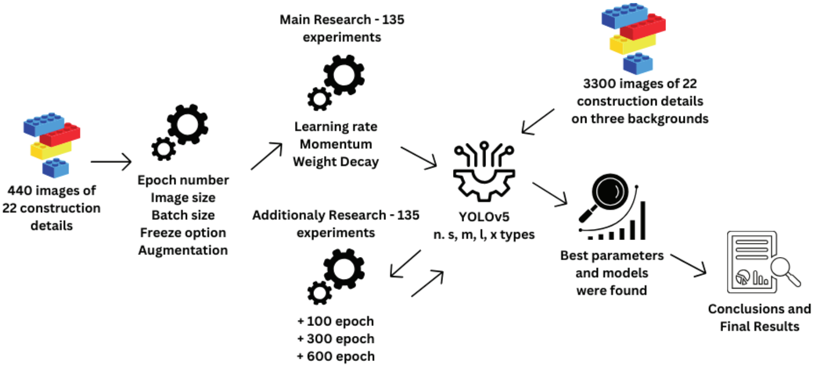
440 images of 22 construction details
Epoch number
Image size
Batch size
Freeze option
Augmentation
Main Research - 135 experiments
Learning rate
Momentum
Weight Decay
Additionaly Research - 135 experiments
+ 100 epoch
+ 300 epoch
+ 600 epoch
YOLOv5
n. s, m, l, x types
3300 images of 22 construction details on three backgrounds
Best parameters and models were found
Conclusions and Final Results

Figure 2. The methodology of the experimental investigation.

The main stage of experiments was focused on hyperparameters, while the additional stage was for additional epochs. For every experiment, pre-trained YOLOv5 [13] was used. The models which were used are already pre-trained with the COCO2017 dataset [14]. The dataset contains 164,000 labelled images of 80 different objects. The models have been trained using 118,000 images, for validation of 5,000 images and testing 41,000 images. During the experiment, every model was trained using Intel(R) Xeon(R) CPU E5-2630 v4 @ 2.20GHz (20 Threads, 10 Cores). Hardware has been used Linux operating system with 32-GB DDR4 RAM and GPU - Tesla P100 PCIe 12GB.

The mean Average Precision (mAP) with a predefined IoU (Intersection Over Union) threshold usually evaluates object detection results during the training stage. Across experiments, models with low accuracy had low mean Average Precision (mAP), and vice versa. After examining the learning curves, it was decided to select models based on correct detection accuracy. This is because models with similar training results showed significantly different accuracy. Differences between models ranged from 100 to 200 detected construction details after testing.

5 Results of the Main and Additional Research

Based on other research [17-19] and our pilot studies, the results have shown that augmentation has a positive impact on detection accuracy. Furthermore, experiments have shown that by freezing backbones, the accuracy increases about 1.5 times. According to the overall results, the detection accuracy is much better on a neutral background. However, all models have been trained with images in which the background was only white. In general, the pilot parameters became like this: image size – 320, batch size – 32, epoch number – 300, layers freeze option – 10, 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.

The analysis of related works showed that most researchers focus on learning rate, momentum, and weight loss [16, 17]. In the main research, the nano (n), small (s), medium (m), large (l), and extra-large (x) versions of the YOLOv5 have been trained using the parameters of the pilots research. A total of 270 models were trained and evaluated. The parameters used in the main research: learning rate – 0.01, 0.001, 0.0001; momentum – 0.9, 0.937, 0.95; weight decay – 0.0001, 0.0005, 0.0007. The other values of the parameters were default. After the main research (135 trained models), additional pieces of training were done because according to results, and learning charts, some models are underfitting. Therefore, models that trained with a 0.01 learning rate were trained additionally with 100, 0.001 - 300 and 0.0001 - 600 epochs. The results have shown that the highest accuracy of the main research is equal to 0.5012 (50.12%). It was achieved with the YOLOv5l model with a learning rate equal to 0.001, a momentum is 0.95, and a weight decay is 0.0007. In some cases, the correct detection ratio was equal to 0. It happens because of the too short training time. Therefore, additional trainings were made. However, for models whose accuracy was 0, it became higher but overall did not make much of an impact. The highest accuracy additional trained model was also YOLOv5l, learning rate - 0.001 and momentum - 0.950; however, the weight decay is 0.0001. This model achieved slightly better results – 0.5015 (50.15%). On the other hand, the most accurate models of YOLOv5 nano (n), medium (m), and extra-large (x) achieved slightly lower results than models of the main experiment (Figure 3).

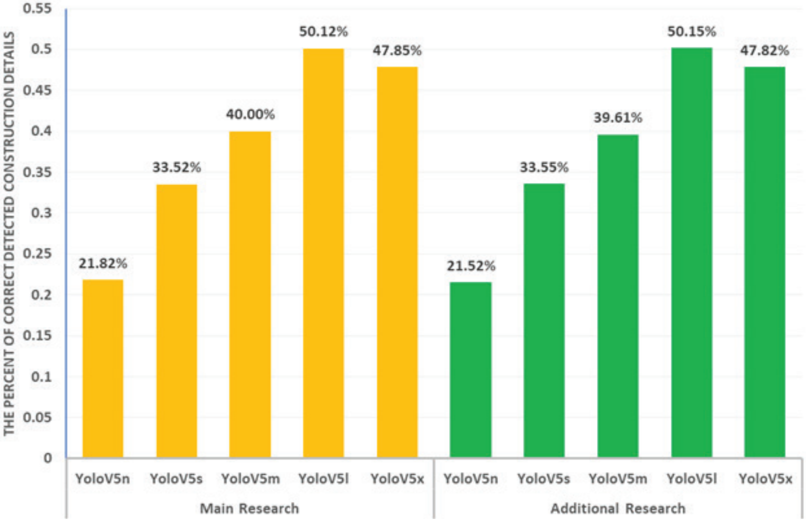

Figure 3. The percentage of correct detection ratio of each nano, small, medium, large, and extra-large model.

As Figure 3 illustrates, according to the main research, YOLOv5n, which has the lowest number of CNN, shows the lowest precision in the detection ratio – 21.82%, while YOLOv5x with the highest amount of CNN achieves 47.85%. The different results for YOLOv5s (33.52%) and YOLOv5m (40%) is 6.48%. Similar ratios were achieved after additional experiments. YOLOv5 nano (n), medium (m) and extra-large (x) have slighter lower accuracies, while small (s) and large (l) versions achieved slighter higher accuracies. In both experiments, YOLOv5l showed the highest results. The main research YOLOv5l – 1654 (mix – 497, neutral – 562, white – 595), while the additional research YOLOv5l – 1655 (mix – 496, neutral – 561, white – 598).

6 Conclusions

This study examined the impact of the training parameters and hyperparameters on the identification of construction details. When analysing similar feature data, the task complexity led to the selection of construction details. Recognition is dependent on the shot's angle, which is determined by the camera's point of view. Throughout the study, the five pre-trained YOLOv5 models were examined. In total, 270 models have been trained and evaluated. Three different complexity backgrounds containing a total of 3300 photos were used to assess the efficiency models. Learning rate, momentum, and weight decay were examined. Every parameter was used in various combinations. According to the findings of the experimental investigation, coloured images, an image size of 320, a batch size of 32, epoch number of 300, an option of layer freeze of 10, data enhancement is used, learning rate of 0.001, momentum of 0.95, and a weight decay of 0.0007 are the optimal parameters for the detection of construction details. Another optimal parameter with almost similar accuracy can be the same as it was mentioned but with a learning rate – 0.0001 and 900 epochs. Regardless of the background chosen, the proportion of proper detection in this case is ~ 50%. The results of the experimental investigation indicate that the use of a mixed background yields the least detection results. The primary cause is that some details become lost in the background, making it impossible for the models to identify any details at all.

References

[1] L. Tan, T. Huangfu, L. Wu, and W. Chen, “Comparison of RetinaNet, SSD, and YOLO v3 for real-time pill identification,” BMC Medical Informatics and Decision Making, vol. 21, no. 1, Nov. 2021, doi: 10.1186/s12911-021-01691-8.

[2] S. M. Alkentar, B. Alsahwa, A. Assalem, and D. Karakolla, “Practical comparison of the accuracy and speed of YOLO, SSD and Faster RCNN for drone detection,” Maǧallaẗ Al-handasaẗ, vol. 27, no. 8, pp. 19–31, Aug. 2021, doi: 10.31026/j.eng.2021.08.02.

[3] 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. Applied Sciences, 2020, 10(23), 8667.

[4] W. Liu et al., “SSD: Single Shot MultiBox Detector,” in Lecture Notes in Computer Science, 2016, pp. 21–37. doi: 10.1007/978-3-319-46448-0_2.

[5] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 6, pp. 1137–1149, Jun. 2017, doi: 10.1109/tpami.2016.2577031.

[6] C. Li et al., “YOLOV6: A Single-Stage Object Detection Framework for Industrial Applications,” arXiv.org, Sep. 07, 2022. https://arxiv.org/abs/2209.02976

[7] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, “YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors,” arXiv.org, Jul. 06, 2022. https://arxiv.org/abs/2207.02696

[8] U. Nepal and H. Eslamiat, “Comparing YOLOV3, YOLOV4 and YOLOV5 for autonomous landing spot detection in faulty UAVs,” Sensors, vol. 22, no. 2, p. 464, Jan. 2022, doi: 10.3390/s22020464.

[9] D. Reis, J. Kupec, J. Hong, and A. Daoudi, “Real-Time Flying Object Detection with YOLOv8,” arXiv.org, May 17, 2023. https://arxiv.org/abs/2305.09972

[10] C.-Y. Wang, I.-H. Yeh, and H.-Y. M. Liao, “YOLOV9: Learning what you want to learn using programmable gradient information,” arXiv.org, Feb. 21, 2024. https://arxiv.org/abs/2402.13616

[11] J. Kim, J.-Y. Sung, and S.-H. Park, “Comparison of Faster-RCNN, YOLO, and SSD for Real-Time Vehicle Type Recognition,” 2020 IEEE International Conference on Consumer Electronics - Asia (ICCE-Asia), Nov. 2020, doi: 10.1109/icce-asia49877.2020.9277040.

[12] Data set of Construction details. https://app.box.com/s/j420ld0wo89hvh6np1rc3z-9t1e65yg2k.

[13] Jocher, G. YOLOv5 by Ultralytics (Version 7.0), Computer software, 2020, https://doi.org/10.5281/zenodo.3908559.

[14] T.-Y. Lin et al., “Microsoft COCO: Common Objects in context,” arXiv.org, May 01, 2014. https://arxiv.org/abs/1405.0312.

[15] S.-H. Gao, Y.-Q. Tan, M.-M. Cheng, C. Lu, Y. Chen, and S. Yan, “Highly Efficient Salient Object Detection with 100K Parameters,” arXiv.org, Mar. 12, 2020. https://arxiv.org/abs/2003.05643

[16] K. Nakamura and B.-W. Hong, “Adaptive weight decay for deep neural networks,” IEEE Access, vol. 7, pp. 118857–118865, Jan. 2019, doi: 10.1109/access.2019.2937139.

[17] M. Taşyürek and C. Öztürk, “A fine-tuned YOLOv5 deep learning approach for real-time house number detection,” *PeerJ*, vol. 9, p. e1453, Jul. 2023, doi: 10.7717/peerj-cs.1453.

[18] Y. Y. Liau and K. Ryu, “Status recognition using Pre-Trained YOLOV5 for Sustainable Human-Robot Collaboration (HRC) system in Mold assembly,” *Sustainability*, vol. 13, no. 21, p. 12044, Oct. 2021, doi: 10.3390/su132112044.

[19] T. Kvietkauskas and P. Stefanovič, “Influence of training parameters on Real-Time similar object detection using YOLOV5S,” *Applied Sciences*, vol. 13, no. 6, p. 3761, Mar. 2023, doi: 10.3390/app13063761.