Multiple structural defect detection for reinforced concrete buildings using YOLOv5s

Chaobin Li[1], Wei Pan[1], Pong Chi Yuen[2] and Ray K L Su[1]

[1] Department of Civil Engineering, The University of Hong Kong, Hong Kong, People's Republic of China
[2] Department of Computer Science, Hong Kong Baptist University, Hong Kong, People's Republic of China

ABSTRACT

Building inspection and maintenance are becoming increasingly essential means by which to consider the deterioration problems of old, reinforced concrete (RC) buildings. While such inspection work can be conducted with the aid of computer vision-based technology, this technology remains challenged, since real-world structural defects and environmental conditions are varied and complex. In recent years, object detection algorithms have improved to achieve greater speed and accuracy with the help of deep learning. In this paper, an advanced object detector, YOLOv5s, was successfully applied to the recognition of common structural defects including cracks, delamination, exposed reinforcement, rust stains, spalling, tile cracks, tile delamination, and tile loss. Compared with the other advanced object detectors of YOLO (i.e., YOLOv5m, YOLOv5l, YOLOv5x, YOLOv4, and YOLOv3) based on the built data set, the YOLOv5s algorithm shows an obvious advantage for defect detection, achieving 64.5% and 67.0% mean average precision (mAP) for training and testing, respectively. It also takes less than 0.1 seconds to detect a defect on an image. The lightweight and high detection performance of the YOLOv5s algorithm shows great promise for potential deployment on an onboard inspection device, such as an unmanned aerial vehicle (UAV) or a robot, to achieve real-time structural inspection.

KEYWORDS Building inspection; deep learning; object detection; YOLOv5
CONTACT Ray K L Su klsu@hku.hk
Received 9 September 2021

1. Introduction

In June 2021, a 40-year-old reinforced concrete (RC) building in Miami suddenly collapsed without warning (Wikipedia, 2021), causing the deaths of at least 98 residents and arousing concern about housing safety. An early inspection report in 2018 showed that the building had seriously deteriorated due to water leakage and consequent concrete damage, and this was considered by many building engineers to be one possible cause for the collapse. Many older RC buildings in Hong Kong also have a deterioration problem, mainly due to carbonation and the impact of marine chloride ion erosion. Building deterioration can cause a variety of structural defects; hence, regular building inspection and maintenance are necessary and urgent.

Manual visual inspection is the most common means of structural inspection in Hong Kong; however, this can be labour intensive, subjective, and unreliable. By comparison, computer vision-based technology can achieve automatic inspection and thus has become a research hotspot. In the past, research has centred around image processing techniques (IPTs), such as histogram transformation, filtering, and texture identification (Koch et al., 2015), which have been adopted for detecting structural defects including cracks (Yamaguchi and Hashimoto, 2010; Yeum and Dyke, 2015), spalling (German et al., 2012; Dawood et al., 2017), rusting (Koch et al., 2014), and corrosion (O'Byrne et al., 2013). However, conventional IPTs are not always effective when dealing with real-world images with complex defects and backgrounds due to limitations such as being time consuming and their ineffectiveness in dealing with background noise.

In recent years, IPTs using deep learning methods have gained much attention thanks to the revival of convolutional neural networks (CNNs) (Krizhevsky et al., 2012). Over the past decade, CNN-based machine learning algorithms have become increasingly more powerful in dealing with real-world images, and there have been a number of successful applications regarding structural inspection. Some researchers have proposed CNN-based classification methods for concrete defect classification (Li et al., 2018), structural component identification (Liang, 2019), and post-disaster damage detection (Pan and Yang, 2020). Others have developed CNN-based object detection methods for recognising cracks, pop-outs, spalling, rebar exposure, efflorescence, and defacement (Zhang et al., 2020; Mondal et al., 2020; Jiang et al., 2021; Kung et al., 2021), where visible structural defects can be identified and localised with bounding frames and classified as the pre-set category. Using the object detection method, multiple structural defects can be recognised from a given image.

There are currently two main deep learning–based object detection algorithms: two-stage algorithms such as Regions with CNN features (R-CNN) (Girshick et al., 2014), Fast R-CNN (Girshick, 2015), and Faster R-CNN (Ren et al., 2017), which generate potential target locations in advance and then apply CNN to classify and detect separately; and one-stage algorithms such as You Only Look Once (YOLO) (Redmon et al., 2016), Single Shot MultiBox Detector (SSD) (Liu et al., 2016),

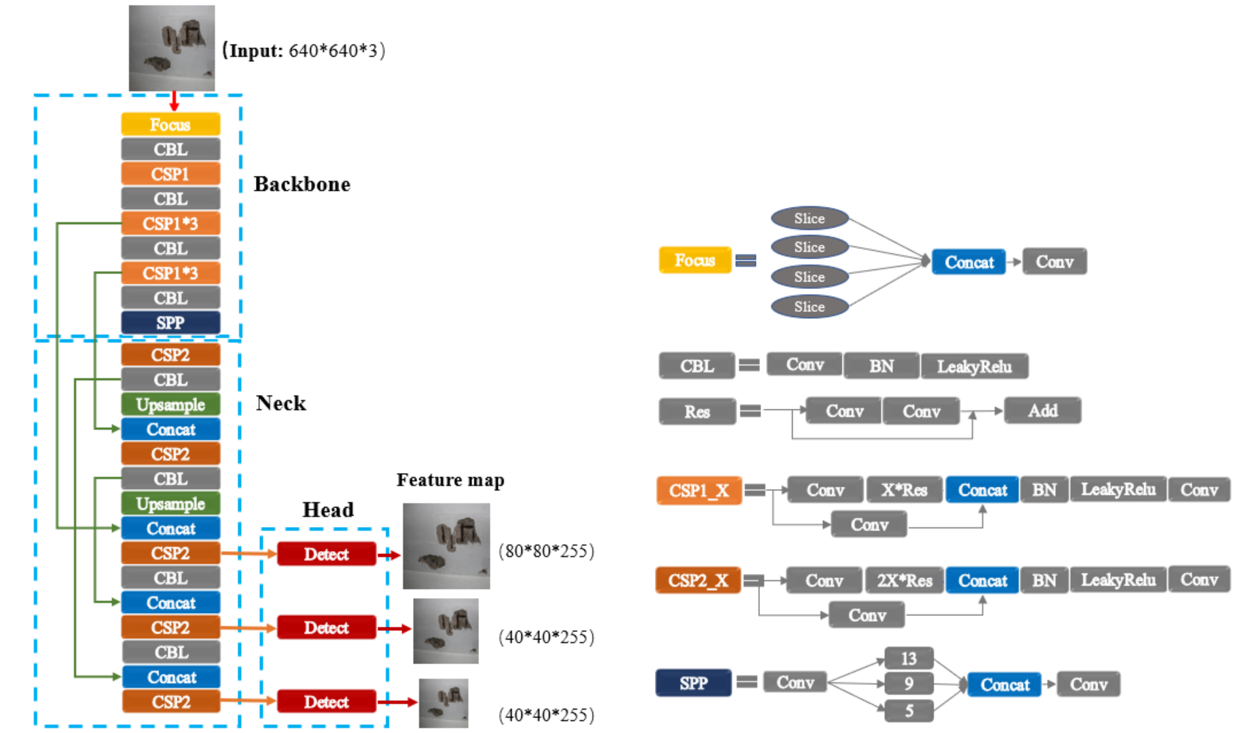

Figure 1. YOLOv5s architecture.

and RetinaNet (Lin et al., 2017), which utilise regression networks to achieve target positioning and classification simultaneously. Although two-stage algorithms can achieve high accuracy, they are comparatively slow in terms of real-time performance. In contrast, one-stage methods can significantly speed up object detection by simplifying the detection processes, which is preferred in structural defect detection. However, research into structural inspection based on one-stage object detectors is limited.

Representative of one-stage detectors, YOLO is superior to other one-stage detectors such as SSD and RetinaNet in terms of accuracy and speed (Redmon and Farhadi, 2017; Redmon and Farhadi, 2018; Bochkovskiy et al., 2020; Ultralytics, 2020). With YOLO, the input image is first divided into grids, then the grid cell, whose region contains the centre point of the object, takes responsibility for prediction. Thereafter, coordinate information of the bounding frame of the predicted target, as well as the confidence score, forms the output. It should be noted that the initial version of YOLO (YOLOv1) has difficulty detecting small objects, with detection results producing a greater number of localisation mistakes and lower recall compared with two-stage algorithms. To address these drawbacks, many improvements have been introduced to the YOLO network, such as batch normalisation, advanced backbone network, convolution with anchor frames, dimension clusters, multi-scale training, and mosaic data augmentation (Redmon and Farhadi, 2017; Redmon and Farhadi, 2018; Bochkovskiy et al., 2020; Ultralytics, 2020). The latest version of YOLO (YOLOv5) can realise real-time and accurate target recognition. Furthermore, the light-weight characteristics of YOLOv5 makes it possible for it to be deployed in an inspection device, such as a UAV or robot, for automatic detection of multiple structural defects.

This paper uses YOLOv5s – the smallest version of YOLOv5 – to conduct experiments in the detection of common structural defects. There are three main contributions of this paper: (1) 1,907 inspection images from old RC structures in Hong Kong were collected to establish the data set, considering eight types of common structural defects including cracks, delamination, exposed reinforcement, rust stains, spalling, tile cracks, tile delamination, and tile loss; (2) a state-of-the-art, one-stage algorithm, YOLOv5s, was successfully developed and applied for structural defect detection; and (3) the performance of a multiple structural defect detection model using YOLOv5s was compared with other advanced object detectors of YOLO, and further optimisation has been proposed.

2. YOLOv5 algorithm

YOLOv5 (Ultralytics, 2020) achieves greater speed and higher accuracy from a smaller model size compared with its predecessors in two authoritative data sets: Pascal VOC (Everingham et al., 2015) and Microsoft COCO (Lin et al., 2014). There are four versions of YOLOv5, namely

YOLOv5s, YOLOv5m, YOLOv5l, and YOLOv5x. The versions all have a similar network architecture but different numbers of feature extraction modules and convolution kernels in the corresponding location of the network. As the version names suggest, the size and number of parameters of the four models increase in turn. YOLOv5s is the most light-weight and achieves fast inference speed but relatively not greater accuracy compared to the other three. In terms of building an inspection application, the goal of this study is to embed the algorithm in onboard inspection devices for timely detection. Special attention should be paid to algorithm size and inference speed, especially for resource-poor edge devices. Hence, in this paper, the structural defect detection experiment is carried out based on the YOLOv5s algorithm.

As shown in Figure 1, YOLOv5s can be divided into three main components: the backbone, which is responsible for calculating feature maps taken from the input image; the neck, which is designed to fuse features obtained by the backbone and pass them to the head for prediction; and the head, which is responsible for predicting image features and performing bounding frame regression and classification. Considering the input element of YOLOv5, the mosaic data augmentation method is adopted to enhance the algorithm's detection accuracy for small objects, and the adaptive anchor box calculation is added to search for the optimal anchor during training. Considering the backbone element of YOLOv5, the focus module, CSP1 module, and SPP module are designed to better calculate the feature maps of the input image, which also ensures detection speed and accuracy, as well as a lower algorithm size. In terms of the neck element of YOLOv5, application of the FPN-PAN structure and CSP2 module has greatly strengthened the propagation and integration of features obtained from the backbone. In the head element of YOLOv5s, three feature maps of different size are formed for the multi-scale prediction in order to consider small, medium, and large targets in the given image. Additionally, the G_IOU loss is employed for calculating bounding box loss.

3. Data set

3.1. Data set production

Due to natural deterioration and aging, many common types of defects can be found in old RC buildings. Some of them may endanger the safety of the structure, such as steel corrosion-induced cracking, delamination, spalling, and water leakage. Most of them are widely distributed and may affect the serviceability of a building, such as damage to a wall finish system. Hence, it is valuable to collect evidence of these defects to effectively assess the building's condition.

From April to July 2021, structural defect images were randomly collected in Hong Kong from Waterfall Bay, Kennedy Town, Sai Ying Pun, Tsuen Wan, To Kwa Wan, and Shatin. Across the wide range of structural

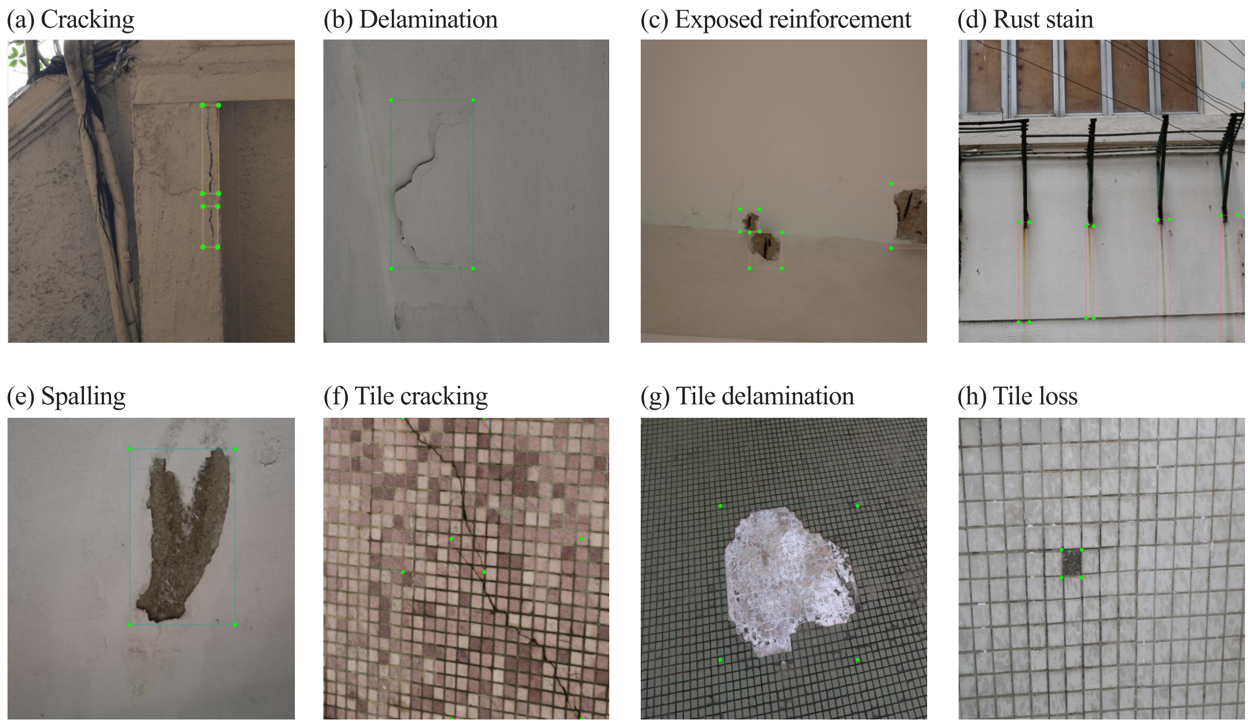

Figure 2. Sample images of structural defects.

defects, eight common types are considered in this paper: cracks, delamination, exposed reinforcement, rust stains, spalling, tile cracks, tile delamination, and tile loss. Most of these are located in car parks and old RC residential buildings varying in age from 30 to 50 years. A Canon EOS 70D camera and iPhone 12 were used for image taking, and the images were saved in JPG format at a resolution of 5,472×3,648 and 2,523×1,170, respectively. In order to improve the training, the image data were manually screened and cropped. Free annotation software, LabelImg, was then used to carry out manual annotations for the images by drawing outer rectangular boxes around each structural defect. The YOLOv5 format files were saved upon completion of the annotation. In total, 1,907 images of a variety of defects were labelled: 684 objects were labelled as cracks (17.7%), 462 as delamination (12.0%), 692 as exposed reinforcement (17.8%), 489 as rust stains (12.6%), 493 as spalling (12.7%), 364 as tile cracks (9.4%), 254 as tile delamination (6.6%), and 433 as tile loss (11.2%). Some labelled samples are shown in Figure 2.

3.2. Data pre-processing

To facilitate the training of the YOLOv5s model, data pre-processing is needed for the labelled image data set. The obtained data file was uploaded to Roboflow (https://app.roboflow.com), which is a free website for data pre-processing.

First, the image was resized to 640×640, which is the default input image size of YOLOv5s. The data set was randomly divided into a training set (80%), validation set (10%), and test set (10%); that is, there were 1,526 images for training, 191 images for validation, and 190 images for testing. To improve the training results and enrich the image data, a data augmentation process was carried out on the training set. Since the goal of this study is to detect real-world images, the detection model must be sufficiently robust to a variety of factors during shooting, such as weather conditions, camera angle, camera shake, and image noise produced by the camera. Hence, several data enhancement methods were utilised, including enhancement and reduction of image brightness, horizontal and vertical mirroring, rotation of the original image (90°, 180°, 270°), rectification of the blur of the original image, and the addition of Gaussian noise to the raw image. The training set was tripled by those processes; that is, there were 4,578 images for training, 191 images for validation, and 190 images for testing. It should be noted that there was no intersection between the training set and test set. The obtained data set documents were saved in YOLOv5 Pytorch format in readiness for application to the developed detection model.

4. Experiment and discussion

4.1. Experiment setup

The YOLOv5s model was trained on the Google Colab platform with a Tesla T4 GPU. Google Colab is a user-friendly online Jupyter notebook environment that offers free and powerful GPUs. As users are not required to undertake any configuration, it is very suitable for training deep learning models. The YOLOv5 notebook used in this paper was developed by Roboflow (2020). To avoid overfitting and accelerate convergence of YOLOv5s, transfer learning was adopted by employing the pre-trained Microsoft COCO weight file (Ultralytics, 2020) to initialise the training model. Related parameters were set as follows: the input image size was 640×640, the total number of training epochs was 150, the batch size was 64, and all other hyperparameters were set as default.

4.2. Evaluation indicators

The precision-recall (P-R) curve is employed for judging the quality of the YOLOv5 detector, where precision can be interpreted as the accuracy at which the detector can localise the object, while recall relates to how complete the detection results can be. Average precision (AP) denotes the area under the curve. A large area (i.e., a larger AP) represents enhanced performance of the detector for the given class. Mean average precision (mAP) reflects the mean value of APs for total classes. mAP@0.5 denotes the mAP value with IoU =0.5, where the value of IoU (Intersection over Union) shows how well the predicted bounding frame overlaps with the ground-truth (GT). mAP@0.5:0.95 denotes the average mAP value form IoU=0.5 to IoU=0.95. In addition, the value change of loss function, which is composed of three components – localisation loss (box loss), confidence loss (objectness loss), and classification loss – can also reflect the training performance of YOLOv5.

4.3. Training results and discussion

Figure 3 sets out some predictions during model training using the validation set. From the comparisons with the GT labels, it is evident that only minor mistakes have occurred in the prediction results, which shows the powerful detection ability of YOLOv5s. Figure 4 shows the P-R curve, the AP for each structural defect, and the mAP@0.5 for all classes. It can be seen that the YOLOv5s model can achieve a mAP of 0.645 for all the structural defects. The AP for detecting tile delamination is highest, followed by exposed reinforcement, and the AP for detecting cracks is lowest, followed by tile cracks. However, the number of labelled tile delamination for training is much lower than for cracks. This is because the YOLOv5 algorithm more accurately detects large objects with distinctive characteristics and may not perform well where there exists a large imbalance between targets and background in the image. To address these issues, we can add more training data and specifically design the algorithm to be more sensitive to those structural defects with a low AP. Figure 5 demonstrates how the curves of precision, recall, mAP@0.5, and mAP@0.5:0.95 increase quickly until the epoch reaches roughly 50. The curves of the loss of box, objectness, and classification in the training data and validation data also show a significant decrease until the epoch reaches roughly 50. When the training epoch is roughly 150, most of the parameters change very little, meaning the model tends to stabilise.

(a) Ground-truth labels

(b) YOLOv5 predictions

Figure 3. Predictions from the validation image data.

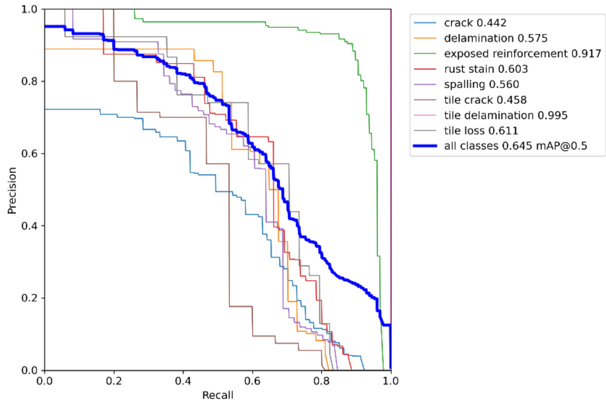

Figure 4. P-R curves for the training data.

4.4. Comparison with other object detectors

Currently, YOLOv3 (Redmon and Farhadi, 2018) and YOLOv4 (Bochkovskiy et al., 2020) are still adopted by many researchers, since they are also effective for target detection by integrating the most advanced technology (Huang et al., 2021; Xin et al., 2021). In fact, the principle and architecture of YOLOv5 are very similar to those of YOLOv3 and YOLOv4. To examine the performance of YOLOv5s, experiments employing three different versions of YOLO, i.e., YOLOv5m, YOLOv5l, YOLOv5x, YOLOv4 and YOLOv3, were conducted as a comparison. All the models were trained based on the same data set. Transfer learning was also applied to the three types of algorithms to avoid overfitting by employing pre-trained Microsoft COCO weight files. The parameter configurations for the different YOLOv5 algorithms were the same. The hyperparameters of YOLOv3 and YOLOv4 were set as default. The YOLOv3 model was trained for 100 epochs, and the training details are shown in Figure 6. The YOLOv4 model was trained for 2,000 iterations, and the average loss during training is shown in Figure 7. The specific training results of each model are summarised in Table 1.

In Figure 6, the training loss of YOLOv3 for the training set and validation set decreases significantly, while the performance metrics increase quickly until the epoch reaches roughly 50. Thereafter, all these parameters become stable, which shows that 100 epochs are enough for model convergence. In Figure 7, the loss value of YOLOv4 during training shows an apparent decrease before the 200th iteration. It then becomes stable at roughly three, which shows the number of iteration steps are enough for model convergence.

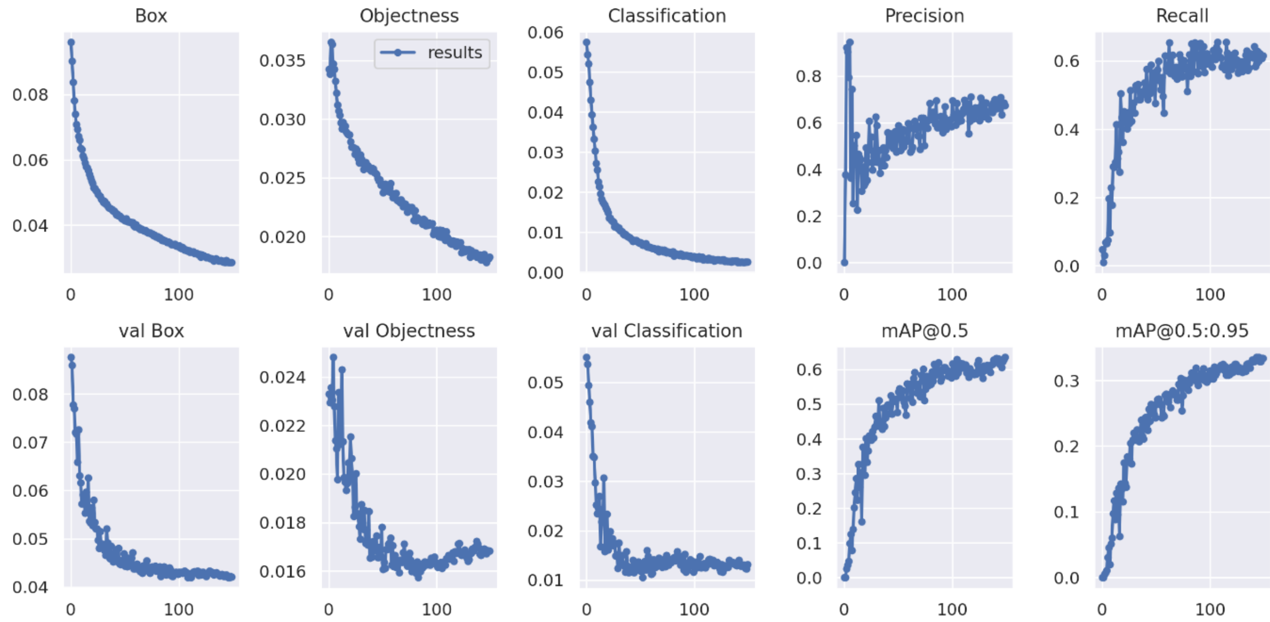

Figure 5. Training details of the YOLOv5s model.

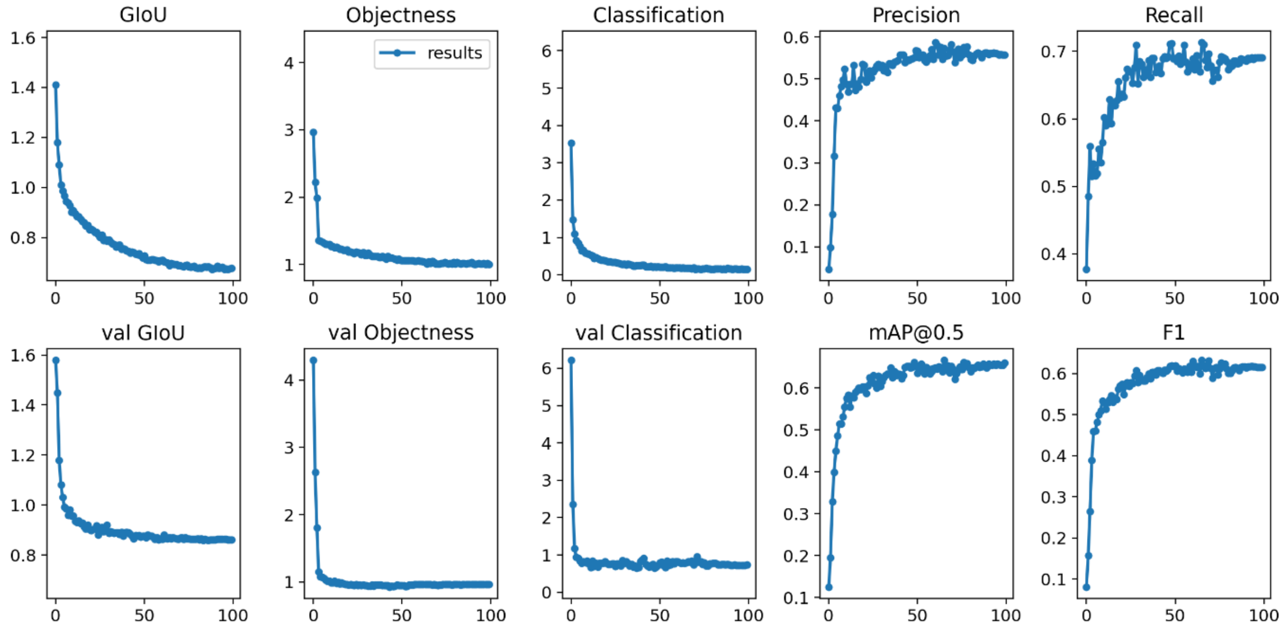

Figure 6. Training details of the YOLOv3 model.

Table 1. Training results of each YOLO model.

Model	P	R	mAP@0.5	FPS	Inference time (milliseconds)	Weights (MB)
YOLOv5s	0.781	0.583	0.645	111	9	56.0
YOLOv5m	0.716	0.64	0.674	91	11	161.0
YOLOv5l	0.696	0.686	0.687	71	14	356.9
YOLOv5x	0.680	0.713	0.691	42	24	667.2
YOLOv4	0.74	0.58	0.615	77	13	244.3
YOLOv3	0.558	0.702	0.642	50	20	236.68

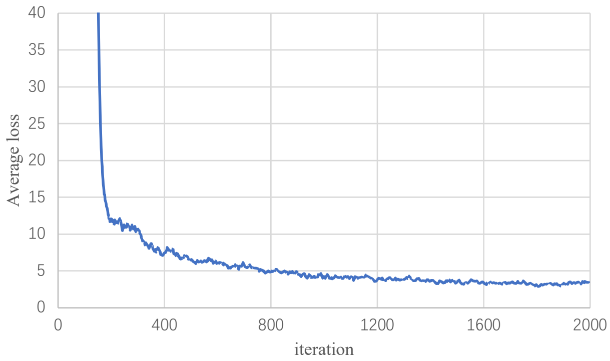

Figure 7. Average loss of the YOLOv4 model during training.

In Table 1, the relevant performance metrics of the different trained models are compared. As can be seen, the YOLOv5s model is the most lightweight, with the highest detection speed of 111 frames per second (FPS), followed by the YOLOv5m model. However, from the perspective of detection accuracy (i.e., Precision (P), Recall (R), and mAP), YOLOv5s is not the best. The mAP value of the YOLOv5s model is lower than the other YOLOv5 models, but better than the YOLOv3 and YOLOv4 models. It should be noted that detection accuracy and inference speed are the primary considerations for real-time, onboard object detection. Hence, from the comparison results, both YOLOv5s and YOLOv5m are the most suitable for onboard structural detection, as they show a similar mAP and inference speed. All of the experiments are done using a high-performance Tesla GPU T4, since the detection speed of an object detector is affected by a GPU's performance. A detection experiment using a Tesla K80 (a GPU with lower computing power) was also conducted. It was found that the inference time of YOLOv5s, YOLOv5m, YOLOv5l, YOLOv5x, YOLOv4, and YOLOv3 is 25 milliseconds, 57 milliseconds, 101 milliseconds, 223 milliseconds, 107 milliseconds, and 121 milliseconds, respectively, showing that YOLOv5s is the fastest object detector. Furthermore, YOLOv5s is the only algorithm that can scan video in real-time, since video is generally 30FPS or 33 milliseconds.

It should be noted that the resources of an onboard inspection device are limited. Ultimately, the YOLOv5s algorithm is chosen for building defect detection, as the file weights of YOLOv5s are three times less than YOLOv5m; although the mAP score is a little lower than YOLOv5m.

4.5. Testing results and discussion

To test the robustness of the trained YOLOv5s model, it needs to make predictions for unseen images. Therefore, the best weight file during training was employed for image inference on the test set. The input image size for testing was 640×640 and the confidence score was 0.4. The results show that most of the structural defects can be correctly recognised, even when the background of the image is complex; indeed, the trained YOLOv5s model achieved a detection performance of mAP=67% for the test set. Moreover, the inference speed per image is less than 0.1 seconds with the Tesla P100 GPU, which meets the needs of structural inspection. Some detection samples are listed in Figure 6 for further discussion.

In Figure 8(a)–(g), the detection model shows enhanced class-specific confidence scores when detecting exposed reinforcement and tile delamination, but lower scores when detecting cracks, tile cracks, and delamination, which is consistent with the training results. In addition, some new problems have emerged. As shown in Figure 8(h)–(j), the returned positioning frame is not accurate for cracks and delamination. In addition, Figure 8(k) shows that the algorithm was unable to detect an obvious tile crack. These problems may be due to insufficient training data and inaccurate data annotation, since these elements directly determine the model's performance. In Figure 8(l), the algorithm mistakenly recognises delamination as cracking, because they have similar characteristics. Additional data training should be implemented to address these problems. The above test results show that the model remains insufficiently robust; hence, in the future, additional data will be required for model training, and the YOLOv5s algorithms will need to be optimised.

5. Conclusion and future work

Structural inspection is essential in light of the deterioration problem for many old RC structures in Hong Kong. With the help of computer vision-based technology, structural inspection can be carried out automatically using onboard inspection devices. The YOLO series of computer vision algorithms are skilled at detecting objects in real time. Hence, this paper carried out an experiment based on YOLOv5s in order to achieve the accurate and fast identification of structural defects.

As an advanced, single-stage object detector, YOLOv5 employs a CSP module, SPP module, and FPN-PAN structure to realise enhanced speed, higher accuracy, and a smaller model size compared with its predecessors. YOLOv5s is the most light-weight version of YOLOv5 and has a fast inference speed, making it the most suitable to be deployed in an onboard inspection device, hence it was adopted for this experiment. A total of 1,907 real-world building inspection images were collected from different regions of Hong Kong, taking into consideration eight types of structural defect, namely cracks, delamination, exposed reinforcement, rust stains, spalling, tile cracks, tile delamination, and tile loss. Data augmentation was also employed to consider various uncertain factors during shooting and to improve detection performance. Five other object detectors (i.e., YOLOv5m, YOLOv5l, YOLOv5x, YOLOv4, YOLOv3) were used as a comparison. The results show that the trained YOLOv5s model is good

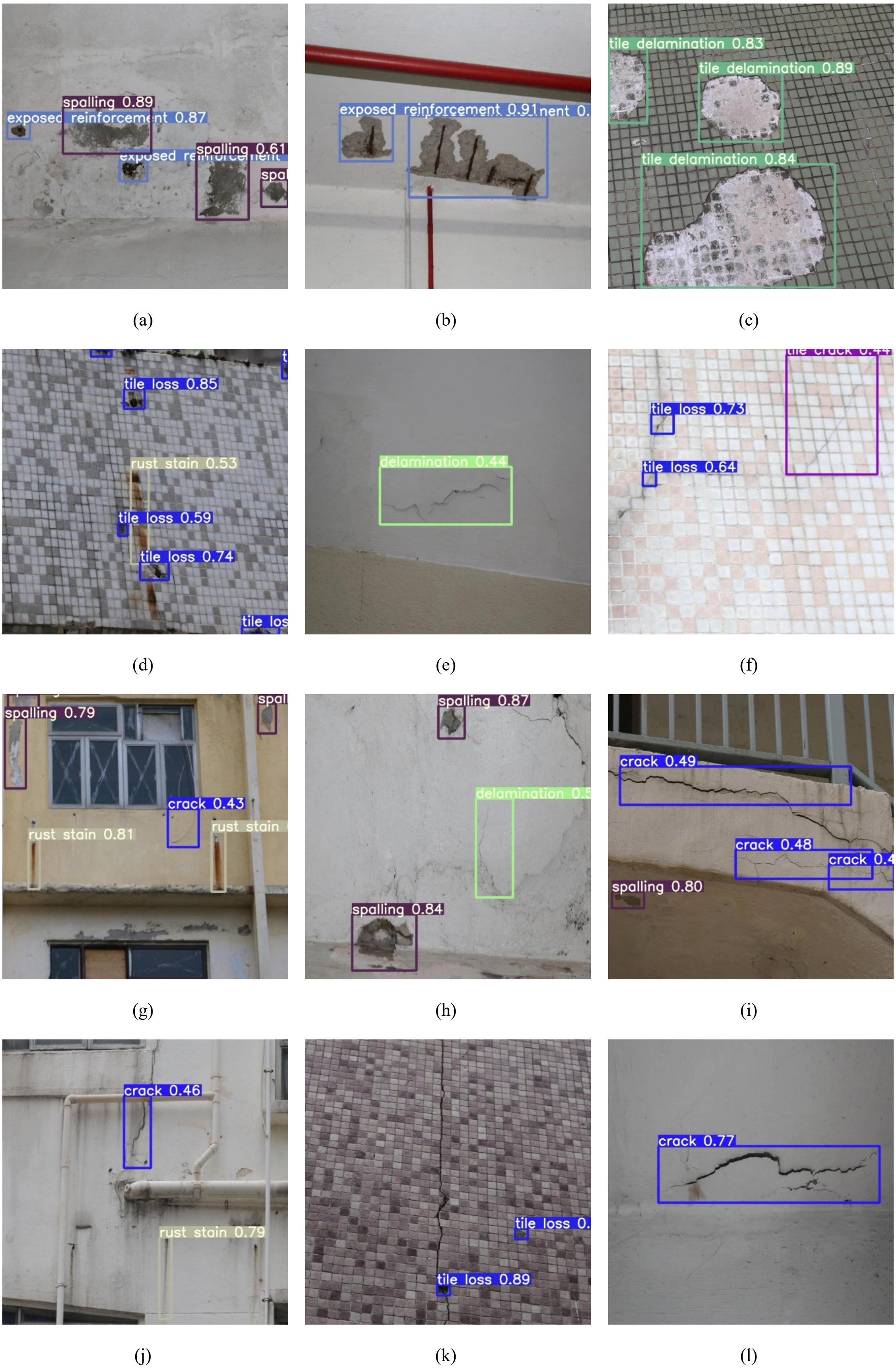

Figure 8. Sample images during testing using YOLOv5s.

enough for building defect detection, achieving a detection mAP of 64.5% and 67.0% for training and testing, respectively. Additionally, using a Tesla T4 GPU, the inference speed per image was less than 0.1 seconds, evidencing successful application in structural defect detection. In addition, the detection algorithm is reliable at detecting large-area defects such as exposed reinforcement and tile delamination, while sometimes missing small-area defects such as cracks and tile cracks.

In the future, sensors such as stereo cameras and laser scanners will be equipped in inspection devices, allowing for the collection of dimensional information such as area and depth of the detected structural defect. With sufficient defect information, building condition can be evaluated more effectively. In order to keep the detection results reliable, further optimisation of YOLOv5s is needed. In the next step, we plan to use additional training data and data balance technology, as well as pay more attention to the quality of the training data. Furthermore, the algorithm architecture will be carefully investigated and modified to improve performance in the structural defect detection task.

Acknowledgements

The authors would like to express their gratitude for the financial support of the Hong Kong Research Grants Council Research Impact Fund (R7027-18) entitled "Modular Integrated Construction 2.0+" for Quality and Efficient Tall Residential Buildings through Advanced Structural Engineering, Innovative Building Materials, and Smart Project Delivery.

Notes on contributors

Mr Chaobin Li is a Ph.D. candidate in the Department of Civil Engineering at The University of Hong Kong. His research interests include damage detection with advanced optical methods and machine learning techniques.

Prof Wei Pan is a Professor in the Department of Civil Engineering at The University of Hong Kong. He received his Ph.D. from Loughborough University. His research interests include sustainable construction and infrastructure, zero carbon building, productivity, offsite prefabrication and modular integrated construction, smart construction, and technological innovation.

Prof Pong Chi Yuen is a Chair Professor in the Department of Computer Science at Hong Kong Baptist University. He received his B.Sc. in Electronic Engineering with First Class Honours in 1989 from City Polytechnic of Hong Kong, and his Ph.D. in Electrical and Electronic Engineering in 1993 from The University of Hong Kong. His research interests include human face recognition, biometric security, and privacy and intelligent video surveillance.

Dr Ray K L Su is an Associate Professor in the Department of Civil Engineering at The University of Hong Kong. His research interests include seismic resistant design of reinforced concrete buildings, structural strengthening, fracture mechanics, and durability of concrete structures.

References

[1] Bochkovskiy A, Wang C and Liao H M (2020). YOLOv4: Optimal speed and accuracy of object detection.

[2] Dawood T, Zhu Z and Zayed T (2017). Machine vision-based model for spalling detection and quantification in subway networks. *Automation in Construction*, 81, pp. 149-160.

[3] Everingham M, Eslami S, Gool L, Williams C, Winn J and Zisserman A (2015). The pascal visual object classes challenge: a retrospective. *International Journal of Computer Vision*, 111(1), pp. 98-136.

[4] German S, Brilakis I and DesRoches R (2012). Rapid entropy-based detection and properties measurement of concrete spalling with machine vision for post-earthquake safety assessments. *Advanced Engineering Informatics*, 26(4), pp. 846-858.

[5] Girshick R, Donahue J, Darrell T and Malik J (2014). *Rich feature hierarchies for accurate object detection and semantic segmentation*. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition. Columbus, pp. 580-587.

[6] Girshick R (2015). *Fast R-CNN*. In: 2015 IEEE International Conference on Computer Vision, Santiago, pp. 1440-1448.

[7] Huang J, Zhang H, Wang L and Zhang Z (2021). Improved YOLOv3 model for miniature camera detection. *Optics & Laser Technology*, 142.

[8] Jiang Y, Pang D and Li C (2021). A deep learning approach for fast detection and classification of concrete damage. *Automation in Construction*, 128.

[9] Koch C, Georgieva K, Kasireddy V, Akinci B and Fieguth P (2015). A review on computer vision based defect detection and condition assessment of concrete and asphalt civil infrastructure. *Advanced Engineering Informatics*, 29(2), pp.196-210.

[10] Koch C, Paal S, Rashidi A, Zhu Z, König M and Brilakis I (2014). Achievements and challenges in machine vision-based inspection of large concrete structures. *Advances in Structural Engineering*, 17(3), pp. 303-318.

[11] Krizhevsky A, Sutskever I and Hinton G (2012). ImageNet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*, pp. 1097-1105.

[12] Kung RY, Pan NH, Wang C and Lee P (2021). Application of deep learning and unmanned aerial vehicle on building maintenance. *Advances in Civil Engineering*, pp. 1-12.

[13] Lee K, Hong G, Sael L, Lee S and Kim H (2020). MultiDefectNet: multi-class defect detection of building facade based on deep convolutional neural network. *Sustainability*, 12(22).

[14] Li R, Yuan Y, Zhang W and Yuan Y (2018). Unified vision-based methodology for simultaneous concrete defect detection and geolocalization. *Computer-Aided Civil and Infrastructure Engineering*, 33(7), pp. 527-544.

[15] Liang X (2019). Image-based post-disaster inspection of reinforced concrete bridge systems using deep learning with Bayesian optimization. *Computer-Aided Civil and Infrastructure Engineering*, 34(5), pp. 415 430.

[16] Lin T, Goyal P, Girshick R, He K and Dollar P (2017). Focal loss for dense object detection. IEEE Transactions on Pattern Analysis & Machine Intelligence, 42(2), pp.318-327.

[17] Lin T, Maire M, Belongie S, Bourdev L, Girshick R, Hays J, Perona P, Ramanan D, Zitnick C and Dollar P (2014). Microsoft COCO: Common Objects in Context. *Springer International Publishing*.

[18] Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu C and Berg A (2016). *SSD: Single shot multibox detector*. In: 14th European Conference on Computer Vision (ECCV), 9905, pp.21-37 .

[19] Mondal TG, Jahanshahi MR, Wu R and Wu Z (2020). Deep learning-based multi-class damage detection for autonomous post-disaster reconnaissance. *Structural Control and Health Monitoring*, 27(4).

[20] O'Byrne M, Schoefs F, Ghosh B and Pakrashi V (2013). Texture analysis based damage detection of ageing infrastructural elements. *Computer-Aided Civil and Infrastructure Engineering*, 28(3), pp. 162-177.

[21] Pan X and Yang TY (2020). Postdisaster image-based damage detection and repair cost estimation of reinforced concrete buildings using dual convolutional neural networks. *Computer-Aided Civil and Infrastructure Engineering*, 35(5), pp. 495-510.

[22] Redmon J, Divvala S, Girshick R and Farhadi A (2016). You only look once: unified, real-time object detection. *IEEE*.

[23] Redmon J and Farhadi A (2017). YOLO9000: better, faster, stronger. *IEEE*, pp. 6517-6525.

[24] Redmon J and Farhadi A (2018). YOLOv3: an incremental improvement. *arXiv*, e-prints.

[25] Ren S, He K, Girshick R and Sun J (2017). Faster R-CNN: towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 39(6), pp. 1137 1149.

[26] Ultralytics (2020). YOLOv5. Available at: <https://github.com/ultralytics/yolov5>. [Accessed on 18 May 2020].

[27] Roboflow (2020). *How to train YOLOv5 on custom objects*. Available at: <https://colab.research.google.com/drive/1gDZ2xcTOgR39tGGs-EZ6i3RTs16wmzZQ>. [Accessed on 1 August 2021].

[28] Wikipedia (2021). *Surfside condominium collapse*. [online] Available at: <https://en.wikipedia.org/wiki/Surfside_condominium_collapse>. [Accessed on 20 August 2021].

[29] Xin H, Chen Z and Wang B (2021). PCB electronic component defect detection method based on improved YOLOv4 algorithm. *Journal of Physics: Conference Series*, 1827(1):012167 (12pp).

[30] Yamaguchi T and Hashimoto S (2010). Fast crack detection method for large-size concrete surface images using percolation-based image processing. *Machine Vision and Applications*, 21(5), pp. 797-809.

[31] Yeum CM and Dyke SJ (2015). Vision-based automated crack detection for bridge inspection. *Computer-Aided Civil and Infrastructure Engineering*, 30(10), pp. 759-770.

[32] Zhang C, Chang C and Jamshidi M (2020). Concrete bridge surface damage detection using a single stage detector. *Computer-Aided Civil and Infrastructure Engineering*, 35(4), pp.389-409.