Automated surface crack detection in historical constructions with various materials using deep learning-based YOLO network

Narges Karimi[1], Mayank Mishra[*2], Paulo B. Lourenço[3]

[1]PhD Student, University of Minho, ISISE, ARISE, Department of Civil Engineering, Guimarães, 4800-058, Portugal, Email: nkarimi@civil.uminho.pt

[2]Marie Skłodowska-Curie Individual Fellow, University of Minho, ISISE, ARISE, Department of Civil Engineering, Guimarães, 4800-058, Portugal, Email: mayank@civil.uminho.pt

[3]Professor, University of Minho, ISISE, ARISE, Department of Civil Engineering, Guimarães, 4800-058, Portugal, Email: pbl@civil.uminho.pt

Corresponding author Email*: mayank@civil.uminho.pt

ABSTRACT Cultural heritage (CH) constructions involve the use of diverse masonry materials. Under natural and human influences, masonry materials can undergo various types of damages, with crack damages being most prevalent. Developing a robust model capable of detecting cracks in various CH materials is crucial for applying deep learning (DL) methods. In this study, we compared the performance of the DL method You Only Look Once (YOLO) object detection network based on images in different masonry materials (stone, brick, cob, and tile) with that in a modern material (concrete). The dataset used in the study comprised 1213 brick, 1116 concrete, 955 cob, 882 stone, and 208 tile images. YOLOv5 architecture, transfer learning, and object detection models were utilized for detecting cracks to observe and compare their performance in different materials. This study represents the first comparison of this kind using an original dataset. The model achieved mean average precision values of 94.4%, 93.9%, 92.7%, 87.2%, 83.4%, 81.6%, and 70.3% for concrete; concrete and cob, cob; stone; stone and brick; brick; and tile, respectively. The findings of this study indicate considerable potential for the widespread use of DL techniques in identifying cracks from images and detecting more damages across various materials.

Keywords Historic buildings, Automatic crack detection, YOLO, Deep learning, Convolutional neural networks

1 Introduction

A crack is a defect resulting from linear failure in brittle materials, such as those used for masonry, for example, concrete, stone, brick, and tiles. Cracks are formed under the effect of various factors, including stresses induced by shrinkage, loading, chemical reactions, poor adhesion, and aging (Davies and Jokiniemi, 2008). They are one of the many concerns for ensuring the safety, durability, stability, operability, and serviceability of building structures.

When cracks develop, they tend to reduce the effective loading area, leading to an increase in stress, eventually resulting in structure failure (Talab et al., 2016). In historical buildings, which often present highly complicated scenarios, neglecting damages such as cracks can worsen the condition. This situation is aggravated when the cracks are deepened under the action of harmful chemicals, impacting the integrity, beauty, originality, and value of the CH (Adhikari et al., 2014). A recent evacuation of "Joshimath town" in India in 2022-2023, owing to crack propagation over years is an example of cracks worsening the situation.

In historical buildings in many countries, cracks are mostly detected manually during inspections. However, this method is time consuming and strenuous as it necessitates measurement, collection, and processing of data. The assessment of deterioration in CH depends on the expertise, knowledge, and skills of inspectors. Relying on this method may result in potentially inaccurate outcomes for pinpointing crack severity (Wang et al., 2019b).

Several methods have been presented for monitoring and detecting cracks in historic buildings. Traditional approaches include using plaster bridges for crack monitoring (Yan et al., 2016), which are relatively simple but often yield unreliable results. The so called "plaster bridges" only give an indication of crack propagation via means of plaster bridge fractures and do not quantify width of the cracks. In addition, various digital, electrical, graphic, and optical sensors, such as fissurometers, Crack gauges based on moiré effect (Ratnam et al., 2019), are available that can be connected to data recording devices and measure the extent of the crack. De Sanjosé Blasco et al. (2020) on carried out crack analysis using fissurometer and laser scanning on a case study Church of San Ildefonso in Salorino and concluded that both crack-measuring geomatic techniques can compliment each other. However, data collection process using the laser scanning is expensive at site and requires high computational cost and might not be feasible for many buildings owing to its costs. This setup allows for automatic data extraction and subsequent analysis by an expert, providing more reliable results. Glisic et al. (2007) in their study installed long-gage fiber optic interferometric sensors in several case study churches to monitor crack openings. Visualisation of cracks by transmitting images via low cost endoscope is also popular in CH structures (Gil et al., 2021). The aforementioned methods work if the cracks are present on the surface of the structure. Some case study applications do exist to determine cracks using millimeter wave-based non-invasive crack detection for ceramic tiles (Agarwal and Singh, 2015).

In addition to the traditional methods, non-destructive wave-based techniques have also been employed for crack detection in historic buildings and historic artefacts. These tests include sonic and ultrasonic testing using sound waves. Pascale and Lolli (2015) based on ultrasonic tests assessed crack patterns and crack depth by extensive statistical analysis on Michelangelo's David in Florence Italy. Tavukçuoğlu et al. (2010) deployed infrared thermography (IRT) and ultrasonic pulse velocity tests to capture radiation and represent it as an image for qualtifying both superficial and deep cracks. Orlando (2007) showed that multicomponent ground-penetrating radar (GPR) data for direct crack detection and monitoring their time-lapse in historic building. A comparitive study was performed on columns to determine the effectiveness of GPR radiography testing and other NDT techniques (such as seismic tomography, endoscopy and compression tests) by Santos-Assunçao et al. (2014). Thus the review summrises some NDT techniques deployed for both external and internal crack measurements and a need for using multidisciplinary approach for crack quantification.

Several applications of machine learning do exist where DL-based techniques are deployed as an non-destructive techniques to identify defects such as cracks in cultural heritage structure. Pratibha et al. (2023) used DL-technique YOLO to determine cracks on masonry structures in case study database of various structures in Bhubaneswar India. Apart from cracks, several studies do deal with identification of defects in addition to cracks in CH structures such as spalling, discoloration, exposed bricks and other surface damages in bricks, detachment, sabotage issues, tile-defects in façades etc (Wang et al., 2019a; Kwon and Yu, 2019; Samhouri et al., 2022; Mishra et al., 2022; Mansuri and Patel, 2022; Li et al., 2023; Wei et al., 2023) in various materials of CH properties. Thus based on review in this paragraph, current approach can help engineers to detect cracks and thus carry out monitoring services such as installation of crack-meters at detected places that are critical and future-crack prone areas.

Researchers have utilized ML in conjunction with DIP (Mishra and Lourenço, 2024), exploring the use of various ML techniques to identify cracks. In ML methods, features in cracks such as edges are extracted by experts, and the entire process is not fully automated. Therefore, DL methods have been widely employed for crack detection. Crack detection using DL methods can be categorized into three groups (Fig. 1). The first category is classification, where the objective is to identify whether an image contains a crack or not. In this approach, images are classified into binary categories: "crack" or "intact" (Xu et al., 2019). In another approach, researchers classify multi-class damages, among which cracks are included as one class (Karimi et al., 2023).

In the second category, object detection, the task is to identify and localize cracks within an image. Crack detection through object detection can be performed using two approaches, which differ in how they represent and detect cracks within an image. In the first approach, an image is divided into a grid of cells, with each cell connected to a group of anchor boxes, which are utilized for detecting cracks. Then, the model predicts whether a crack is present in each anchor box and adjusts coordinates of the box. Examples of anchor-based methods include R-CNN (Zhao et al., 2022), mask-RCNN (Kim et al., 2022), faster R-CNN (Xu et al., 2022), Single Shot Detector (SSD) (Yan and Zhang, 2021), and YOLOv1-v8 (Jia et al., 2023; Qiu and Lau, 2023; Su et al., 2024). The second approach enables direct prediction of crack locations and sizes using a deep neural network (DNN) and involves anchor-free methods. Examples of these methods include CenterNet (Jia et al., 2020) and CornerNet.

The third category is semantic segmentation, which determines whether each pixel belongs to a part of a crack. In this method, researchers have utilized several models to detect cracks. The first approach involved the use of encoder–decoder models such as full convolution network (FCN) and SegNet (Dung et al., 2019; Tabernik et al., 2020). The second approach involved using models without pooling layers to maintain the quality of resolution (Fei et al., 2019; Islam and Kim, 2019). In the third approach, fully connected (FC) layers were combined with techniques such as adaptive thresholding (Ai et al., 2020). The fourth approach employed recurrent neural networks (Zhang et al., 2018a), and the fifth approach utilized generative adversarial networks that employed self-supervised learning for achieving crack semantic segmentation (Zhang et al., 2020).

Based on the preceding literature review, further investigation is crucial to address gaps in crack detection in different masonry materials in CH using DL. Most studies have focused on crack detection in modern materials such as asphalt, concrete, and metal. However, an increasing number of studies are focusing on masonry materials mostly using different deep learning (DL) methods to detect cracks in a specific material within customized datasets. Crack detection in historical buildings made of masonry materials is more challenging than in those made of modern materials. Moreover, there is a lack of datasets related to crack detection in various masonry materials.

In this study, cracks in four different masonry materials (cob, brick, stone, and tiles) were collected, and their results were compared with a concrete dataset similar to modern materials. The images of concrete and cob were combined into one dataset, while stone and brick were grouped into separate datasets. Six distinct datasets were utilized and trained using this model. These data were used in DL models and assist experts in prompt detection of cracks, enabling more efficient and timely repairs of CH.

2 Materials and methodology

2.1 Data collection and preprocessing

Due to the diverse nature of the materials considered herein, datasets were gathered separately from different locations. The images of stones and bricks were captured at the historic bridges in Isfahan (Karimi et al., 2023). Stone photos depict the foundations of bridges, while brick photos show the main structures. The cob dataset was collected from historical buildings in villages located in the desert area around Isfahan, Iran. Tile images were collected from structural ornaments in northern Portuguese cities including Porto, Barcelos, Braga, and Guimarães. Photos of concrete bridges in Isfahan city were included in the concrete dataset (Fig. 2).

All images were captured using a 64-megapixel Samsung Galaxy A32 camera, ensuring the right angle and distance from the cracks. To prevent model overfitting, photos were taken under varying weather conditions, including cloudy, sunny, and shadowy. A total of 4,912 photos were collected first. After removing low-quality or irrelevant images, the dataset was meticulously cleaned, resulting in 1397, 1344, 998, 992, 241, 2389, and 2342 photos for the concrete; brick; stone; cob; tile; concrete and cob; and stone and brick datasets, respectively. All pictures had a resolution of 3456×3456 pixels, and to ensure model fitting, the resolution was scaled down to 416 × 416.

The images were taken such that we captured not only crack damage but also other issues such as higher plants, vegetation, efflorescence, spalling, and glaze detachment. These additional elements complicated the object detection process for the model. The presence of mortar boundaries can often be a challenge for the model, especially in distinguishing between cracks and joints. To address these challenges, 45° image rotation was applied as a data augmentation technique, ensuring robustness of the model. Image processing techniques were applied to modify the brightness and contrast of the images, distinguishing cracks from the background and improving damage visibility. This adjustment ensured that cracks were more noticeable and easily detectable by the model. The bounding boxes used to annotate the data were rectangular and covered the cracks in the images. Each rectangular box was manually drawn around individual cracks, tightly enclosing the damages within these boxes using Roboflow (Fig. 3).

2.2 Model implementation

ML and DL networks are complex neural networks that learn input data characteristics and produce desired outputs (LeCun et al., 2015; Mohammadi et al., 2023). The main tasks in DL are classification, object detection, and semantic segmentation (Liu et al., 2020). In this study, we focused on object detection for identifying and detecting damage. R-CNN, Fast R-CNN, and Faster R-CNN are DL-based object detection algorithms (Girshick, 2015). RetinaNet, SSD, and Feature-fusion SSD (FSSD) are other faster object detection algorithms (Bai et al., 2022). YOLO is a one- stage algorithm known for its real-time capabilities, which has contributed to its popularity. Several versions of YOLO have been developed, ranging from YOLOv1 to YOLOv8. In this study, YOLOv5 was selected because it yielded superior results compared with its higher versions.

Fig. 4 shows the general structure of the applied YOLOv5 algorithm for crack detection.

It comprised three components: the backbone, neck, and head layers. The crack image was fed into the model through the input layer. A pre-trained network serving as the backbone extracted intricate feature representations of cracks. This process diminished the spatial resolution of the image while enhancing its feature resolution, which was particularly beneficial for refining edge details within cracks. Subsequently, the model's neck was employed to generate feature pyramids, enhancing its ability to generalize across various sizes and scales of cracks and their features. Lastly, the model's head was used by applying anchor boxes to feature maps, culminating in the final output: classes (indicating the presence or absence of a crack class in the image), objectness scores, and bounding boxes (indicating the percentage of predicted bounding boxes for cracks that align with the annotations).

The dataset (Section 2.1) was partitioned into three subsets: training, testing, and validation sets, maintaining a ratio of 70:20:10. The total data counts for brick; cob; concrete; stone; tile; concrete and cob; and stone and brick were 1213, 955, 1,116, 882, 208, 2,071, and 2,905, respectively (Table 1). The data augmentation technique was also applied to the training data, involving 45° image rotation, effectively doubling the amount of training data (Fig. 5).

The model was trained using the COCO dataset, with 100 epochs, a batch size of 16, using pre-trained `yolov5S` weights. The dataset was trained using the custom YOLOv5 algorithm. It contains a total of 214 layers, 7022326 parameters, and gradients. All experiments were performed using the Pytorch library in Google Colaboratory (COLAB) with the available GPU memory (NVIDIA Tesla T4, 16GB). Table 2 displays the model information and hyperparameters used for training.

2.3 Optimizer

The model utilizes a stochastic gradient descent (SGD) optimizer. Typically, parameters such as biases and normalization layers (e.g., batch normalization layers) should not undergo weight decay. to eliminate weight decay for biases and normalization layers, these specific parameters were optimized using SGD. SGD was utilized to minimize the loss function of the model by updating the model parameters along the direction of the negative gradient (Zhang et al., 2018b), as represented using Equation 1:

$$\theta_{t+1} = \theta_t - \mathrm{lr}\nabla_\theta \mathrm{Lf}(\theta_t; x_t, y_t) \tag{1}$$

where θ represents the model's parameters that should be updated during training to minimize the loss and improve the model's performance on crack detection. lr is the learning rate that determines the size of steps used when training the crack detection model. Lf is the loss function that measures the model's confidence by comparing the prediction of a crack within a bounding box with the actual label. x_t and y_t denote the input and output of the mini-batch at iteration t, respectively, and ∇_θ represents the gradient operator with respect to θ.

2.4 Performance metrics

The custom YOLOv5 model's performance was evaluated in terms of precision (or positive predictive value) and recall (or sensitivity) by using equations 3 and 4, respectively.

$$\text{Precision} = \frac{TP}{TP + FP} \tag{3}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{4}$$

where True Positive (TP) indicates the number of cases where the model correctly identifies crack, False Positive (FP) represents the number of background images that are incorrectly identified as crack, and False Negative (FN) refers to the number of images incorrectly identified as the background instead of a crack.

The robustness of the proposed YOLOv5 algorithm was measured using the average precision (AP) and mean average precision (mAP) values. For a single class (corresponding to the number of classes being 1), the metrics of precision (P), AP, and mAP were equal. Therefore, we considered mAP50 (IoU = 0.5) and mAP50:95 (IoU threshold ranging from 0.5 to 0.95).

The bounding boxes predicted for identified cracks were evaluated against the ground-truth bounding boxes (annotations) by using a metric known as intersection over union (IoU). The threshold value of IoU was set to 0.5; accordingly, only the predicted bounding boxes with IoU ≥ 0.5 were considered as correct crack detection. This metric measured the extent of overlap between the two boxes (Equation 7).

$$\mathrm{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}} \tag{7}$$

The loss functions of YOLOv5 were as follows: first, bounding box regression loss (L_{bbox}) measured how well the predicted bounding boxes matched the ground-truth boxes drawn around the cracks. Objectness loss (L_{obj}) is the second loss function, which measures the model's confidence level in predicting the existence of a crack in each bounding box. The classification loss (L_{cls}) was calculated similarly, but it focused on the predicted class probabilities. Because the number of classes is one, this loss function is always 0.

3 Results and discussion

3.1 Crack detection results and performance metrics for the different material datasets

From the results obtained, some general outcome can be derived. These outcomes are presented in Table 3 and Fig. 6 along with the associated visual attention (Fig. 6a). The proposed model obtained a mAP50 of 94.4%, 93.9%, 92.7%, 87.2%, 83.4%, 81.6%, and 70.3% for concrete, concrete and cob, cob, stone, stone and brick, brick, and tile, respectively.

Based on Table 3 and Fig. 8, the precision values for concrete and cob, concrete, cob, stone, brick, stone and brick, and tile were 93.2%, 92.6%, 92%, 86.3%, 82.6%, 80.6% and 71.2%, and 90%, respectively. The recall values for concrete and cob, concrete, cob, stone, stone and brick, brick, and tile were 90%, 89.2%, 87.9%, 80.3%, 77.5%, 77%, and 67.4%, respectively (Fig. 8b).

The precision, recall, and mAP_0.5 values for concrete surpassed those of other materials, indicating that the model successfully detected cracks in concrete. This may be attributed to two factors: first, the number dataset of concrete images, and second, the background of concrete was more uniform compared to that of other materials. The model was thus more capable of recognizing cracks against a plain background compared to a patterned one. The concrete and cob datasets ranked second in terms of precision, recall, and mAP_0.5. Cob ranked third with a difference of 1%. Despite the lower quantity of concrete and cob images in the dataset compared to brick, the model exhibited better performance in detecting cracks on concrete and cob. This observation suggests that in the context of object detection using the YOLOv5 model, the nature of the background plays a crucial role, potentially outweighing the influence of the quantity of images in the dataset.

Stone, brick, and tiles exhibited more intricate backgrounds than concrete and cob. The recall rate for stone and brick materials was 80% and 77%, respectively, whereas that for concrete and cob was close to 90%. The presence of joints in stone and brick materials might contribute to the model's inability to correctly identify some joint parts as cracks. This issue became more challenging for tiles because in addition to joints, tiles also include feature patterns. Incorrectly identifying parts of the pattern and the overall tile design as cracks can lead to inaccuracies, resulting in a lower recall rate for tiles (67%). This observation underscores the complexity of accurately detecting cracks in materials with intricate backgrounds or patterns.

When a new dataset was created by combining concrete and cob images, the model's mAP_0.5 was lower than that of concrete but higher than that of cob. The difference in mAP_0.5 was close to 1%. This pattern was also observed in the brick and stone dataset, where the mAP_0.5 was intermediate to those of the brick and stone datasets. Therefore, we anticipated that by blending data with homogeneity in terms of background and structure, higher mAP_0.5 can be attained.

Fig. 7b presents the values of mAP_50:95, representing the average mAP across various IoU thresholds ranging from 0.5 to 0.95. This indicated that the YOLOv5 model achieved a better overlap between the predicted and ground-truth bounding boxes around cracks for concrete, concrete and cob, cob, stone, stone and brick, brick, and tile, respectively. For instance, the model attained an mAP_50:95 of 77% for concrete, showcasing a commendable level of precision across a range of IoU thresholds when detecting cracks. This suggests the model's effectiveness in precisely locating and categorizing cracks in this particular material.

The loss function, depicted in Figs. 9, 10a ,and 10b and table 4, consistently decreased as the model's crack detection accuracy improved. The reduction in both objectness and bounding box losses indicated that the YOLOv5 model could identify cracks and establish bounding boxes around them. Moreover, with model training, the loss associated with crack detection and bounding boxes decreased. The loss was maximum for crack detection in tiles and minimum for that in concrete. This observation aligned logically with the obtained precision metrics.

As depicted in Fig. 11a-c, the model successfully detected cracks in previously unseen concrete images. The confidence levels for these images were 93%–81% and 76%, respectively. The 93% confidence level (Fig. 11c) corresponds to an image where the crack was clearly visible, indicating that the model identified the crack with higher certainty. The confidence level of 76% (Fig. 11b) was attributed to an image where cracks were less visible or showed very close resemblance to the background. In an image containing two cracks, the model failed to detect one of them, indicating one of its limitations.

For cob materials, the model effectively detected cracks in previously unseen images with confidence levels in the range 52%–82% (Fig. 11d-f). However, identifying thin cracks and cracks that have branched into thinner ones remained challenging for the model.

The model achieved confidence levels ranging from 61% to 78% (Fig. 11j-l) for unseen images of brick material, signifying its capability to identify cracks. However, in a specific instance, the model inaccurately identified a crack in the joint between the bricks and failed to detect a small crack branching from the joint.

For stone, the model demonstrated noteworthy success in its outcomes for unseen photos. However, an exception was noted in one instance, where the model incorrectly identified a joint between stones instead of a crack. The confidence levels ranged from the maximum of 91% (Fig. 11g) to the minimum of 65% (Fig. 11h).

The model did not achieve a high level of confidence for unseen images of the tile crack, and it failed to detect cracks in two instances (Fig. 11m and Fig. 11n). However, in a case where the tile design was simple, it successfully identified the cracks, despite confidence levels of 48% and 51% (Fig. 11o) and 78% (Fig. 11j). Despite the relatively low confidence, the model accurately detected the presence of crack damage.

3.2 Comparative analysis of YOLOv5 model with other researches

This study aimed to detect cracks in various materials of historical buildings through the application of the Yolov5 model. Historical buildings are often made of diverse materials with different backgrounds, which pose a challenge in developing and training a Yolov5 model for crack identification. Previous research has mostly concentrated on identifying cracks in a single material type, such as concrete or asphalt, but not in different materials.

Different versions of the YOLO method have been extensively employed for detecting cracks in concrete materials. Utilizing the YOLOv5 model, Mishra et al. (2023), and Zhu et al. (2023) achieved mAP50 scores of 95.02%, and 91.2%, respectively. This model has also been applied in crack detection for asphalt and pavement. Employing YOLOv3, Nie and Wang (2019) attained the detection accuracy of 88%. YOLOv4 and YOLOv2 exhibited the mAP50 values of 94.54% and 91.74%, respectively, in identifying cracks in pavement tiles, utilizing a dataset collected through drones. This method has been seldom applied to building materials, particularly historical structures. Similarly, utilizing the YOLO method, Pratibha et al. (2023) identified cracks in brick materials in historical monuments in India, achieving the mAP value of 92%.

Unlike previous studies that have predominantly focused on concrete and asphalt materials, the present study focused on building materials, particularly those used in historical structures. Cracks in historical buildings exhibit more intricate shapes than those in concrete structures, posing a greater challenge for the detection model. Moreover, building materials like brick, stone, and tile feature seams, doubling the complexity of crack detection. Unlike prior studies that lacked a focus on diverse materials, in this study, we trained the YOLOv5 model on seven datasets with five different materials, achieving satisfactory results. Based on the results of this study, we infer that the YOLOv5 model can be reliably used for detecting cracks in various materials in historical buildings.

4 Conclusions, limitations, and future scope

CH structures often use different materials, such as stone, brick, tile, and plaster. Structural damages, particularly cracks, are detrimental to the integrity of buildings. Thus, it is essential to develop a DL model capable of detecting crack damage in diverse materials using image data. Previous research has left a substantial gap in this domain.

In this study, we present a crack detection model using DL methods specifically tailored for CHs built using diverse materials such as brick, stone, tile, cob, and even a modern material like concrete. In addition to identifying the optimal model for crack detection in these materials, we used a mixed dataset of homogeneous materials (concrete-cob and stone-brick) and compared the outcomes with the previous unmixed datasets.

The YOLOv5 model was employed , which could successfully detect cracks across seven datasets encompassing different materials, achieving high accuracy (mAP) values of 94.4%, 93.9%, 92.7%, 87.2%, 83.4%, 81.6%, and 70.3% for concrete, concrete-cob, cob, stone, stone-brick, brick, and tile, respectively. Furthermore, the model could accurately detect individual cracks in images containing multiple instances.

To enhance the model's accuracy, we recommend increasing the dataset size and including more images. Using the YOLOv5 model was proven to be a time-saving approach, aiding restoration professionals in reducing maintenance costs for CH within their preservation plans. Furthermore, this model's applicability extends beyond diverse materials and different types of deteriorations in CH.

The primary challenge and limitation faced in this study were the processes of collecting data and taking images of cracks in diverse materials, as they are highly time-consuming tasks. Labeling this dataset was another time-intensive challenge. The presence of joints between bricks, stones, and tiles as masonry materials further complicated the model's ability to distinguish between cracks and joints. Moreover, the intricate designs on tiles posed additional difficulties for the model in detecting cracks.

Future research can use drones to gather images of cracks from different materials in CHs, thereby allowing the acquisition of a larger number of crack images and the use of the model for real-time crack detection. The use of various data can improve the accuracy of the DL models, enhancing their reliability and aiding CH experts in identifying issues in inaccessible locations using drones. This method can be highly valuable during critical situations and natural calamities such as floods or earthquakes.

Since bounding boxes cannot precisely measure the size and dimensions of cracks, future studies could shift focus toward semantic segmentation models. These models can determine geometric dimensions of cracks, enabling experts to prioritize the risk of cracking, eventually assisting them in restoring and protecting CH structures. The current inspection technique via YOLO model helps in identification/pinpointing of cracks from digital images and can be a starting point to determine the areas needing more extensive monitoring via other techniques such as laser scanning and installation of fissurometers at critical locations.

Data Availability statement

The Mendeley database (Mendeley Data, doi: 10.17632/3t3dk43bv9.2) contains some of the tile image data collected by authors that were used for the YOLOv5 models. The authors can provide the Python code for the replication of results upon reasonable request.

Funding

This research has been funded by the European Unions Horizon research and innovation program under the Marie Skodowska-Curie grant agreement No 101063722. This work was partly financed by FCT/MCTES through national funds (PIDDAC) under the R & D Unit Institute for Sustainability and Innovation in Structural Engineering (ISISE), under reference UIDB / 04029/2020 (doi.org/10.54499/UIDB/04029/2020), and under the Associate Laboratory Advanced Production and Intelligent Systems ARISE under reference LA/P/0112/2020.

Declaration of Competing Interest

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

References

R. Adhikari, O. Moselhi, and A. Bagchi. Image-based retrieval of concrete crack properties for bridge inspection. *Automation in construction*, 39:180–194, 2014.

S. Agarwal and D. Singh. An adaptive statistical approach for non-destructive underline crack detection of ceramic tiles using millimeter wave imaging radar for industrial application. *IEEE Sensors Journal*, 15(12):7036–7044, 2015.

D. Ai, G. Jiang, S.-K. Lam, P. He, and C. Li. Automatic pixel-wise detection of evolving cracks on rock surface in video data. *Automation in Construction*, 119:103378, 2020.

D. Bai, Y. Sun, B. Tao, X. Tong, M. Xu, G. Jiang, B. Chen, Y. Cao, N. Sun, and Z. Li. Improved single shot multibox detector target detection method based on deep feature fusion. *Concurrency and Computation: Practice and Experience*, 34(4):e6614, 2022.

N. Davies and E. Jokiniemi. *Dictionary of architecture and building construction*. Routledge, 2008.

J. J. De Sanjosé Blasco, M. Sánchez-Fernández, A. D. Atkinson, and A. M. Marra Recuero. The analysis of pathologies in the church of san ildefonso in salorino (cáceres) using geomatic techniques. In *Advances in Design Engineering: Proceedings of the XXIX International Congress INGEGRAF, 20-21 June 2019, Logroño, Spain*, pages 507–517. Springer, 2020.

C. V. Dung et al. Autonomous concrete crack detection using deep fully convolutional neural network. *Automation in Construction*, 99:52–58, 2019.

Y. Fei, K. C. Wang, A. Zhang, C. Chen, J. Q. Li, Y. Liu, G. Yang, and B. Li. Pixel-level cracking detection on 3d asphalt pavement images through deep-learning-based cracknet-v. *IEEE Transactions on Intelligent Transportation Systems*, 21(1):273–284, 2019.

E. Gil, Á. Mas, C. Lerma, M. E. Torner, and J. Vercher. Non-destructive techniques methodologies for the detection of ancient structures under heritage buildings. *International Journal of Architectural Heritage*, 15(10):1457–1473, 2021.

R. Girshick. Fast r-cnn object detection with caffe. *Microsoft Research*, 2015.

B. Glisic, D. Inaudi, D. Posenato, A. Figini, and N. Casanova. Monitoring of heritage structures and historical monuments using long-gage fiber optic interferometric sensors–an overview. In *The 3rd international conference on structural health monitoring of intelligent infrastructure, Vancouver*, pages 13–16, 2007.

M. M. Islam and J.-M. Kim. Vision-based autonomous crack detection of concrete structures using a fully convolutional encoder–decoder network. *Sensors*, 19(19):4251, 2019.

X. Jia, X. Yang, X. Yu, and H. Gao. A modified centernet for crack detection of sanitary ceramics. In *IECON 2020 The 46th Annual Conference of the IEEE Industrial Electronics Society*, pages 5311–5316. IEEE, 2020.

Z. Jia, X. Su, G. Ma, T. Dai, and J. Sun. Crack identification for marine engineering equipment based on improved ssd and yolov5. *Ocean Engineering*, 268:113534, 2023.

N. Karimi, N. Valibeig, and H. R. Rabiee. Deterioration detection in historical buildings with different materials based on novel deep learning methods with focusing on isfahan historical bridges. *International Journal of Architectural Heritage*, pages 1–13, 2023.

C. Kim, S. Hwang, and H. Sohn. Weld crack detection and quantification using laser thermography, mask r-cnn, and cyclegan. *Automation in Construction*, 143:104568, 2022.

D. Kwon and J. Yu. Automatic damage detection of stone cultural property based on deep learning algorithm. *The International Archives of the photogrammetry, remote sensing and spatial information sciences*, 42:639–643, 2019.

Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

Q. Li, L. Zheng, Y. Chen, L. Yan, Y. Li, and J. Zhao. Non-destructive testing research on the surface damage faced by the shanhaiguan great wall based on machine learning. *Frontiers in Earth Science*, 11:1225585, 2023.

L. Liu, W. Ouyang, X. Wang, P. Fieguth, J. Chen, X. Liu, and M. Pietikäinen. Deep learning for generic object detection: A survey. *International journal of computer vision*, 128:261–318, 2020.

L. E. Mansuri and D. Patel. Artificial intelligence-based automatic visual inspection system for built heritage. *Smart and Sustainable Built Environment*, 11(3):622–646, 2022.

M. Mishra and P. B. Lourenço. Artificial intelligence-assisted visual inspection for cultural heritage: State-of-the-art review. *Journal of Cultural Heritage*, 66:536–550, 2024.

M. Mishra, T. Barman, and G. Ramana. Artificial intelligence-based visual inspection system for structural health monitoring of cultural heritage. *Journal of Civil Structural Health Monitoring*, pages 1–18, 2022.

M. Mishra, V. Jain, S. K. Singh, and D. Maity. Two-stage method based on the you only look once framework and image segmentation for crack detection in concrete structures. *Architecture, Structures and Construction*, 3(4):429–446, 2023.

A. Mohammadi, S. Karimzadeh, S. Yaghmaei-Sabegh, M. Ranjbari, and P. B. Lourenço. Utilising artificial neural networks for assessing seismic demands of buckling restrained braces due to pulse-like motions. *Buildings*, 13(10):2542, 2023.

M. Nie and C. Wang. Pavement crack detection based on yolo v3. In *2019 2nd international conference on safety produce informatization (IICSPI)*, pages 327–330. IEEE, 2019.

L. Orlando. Using gpr to monitor cracks in a historical building. In *2007 4th International Workshop on, Advanced Ground Penetrating Radar*, pages 45–48. IEEE, 2007.

G. Pascale and A. Lolli. Crack assessment in marble sculptures using ultrasonic measurements: laboratory tests and application on the statue of david by michelangelo. *Journal of Cultural Heritage*, 16(6):813–821, 2015.

K. Pratibha, M. Mishra, G. Ramana, and P. B. Lourenço. Deep learning-based yolo network model for detecting surface cracks during structural health monitoring. In *International Conference on Structural Analysis of Historical Constructions*, pages 179–187. Springer, 2023.

Q. Qiu and D. Lau. Real-time detection of cracks in tiled sidewalks using yolo-based method applied to unmanned aerial vehicle (uav) images. *Automation in Construction*, 147:104745, 2023.

M. M. Ratnam, B. Y. Ooi, and K. S. Yen. Novel moiré-based crack monitoring system with smartphone interface and cloud processing. *Structural Control and Health Monitoring*, 26 (10):e2420, 2019.

M. Samhouri, L. Al-Arabiat, and F. Al-Atrash. Prediction and measurement of damage to architectural heritages facades using convolutional neural networks. *Neural Computing and Applications*, 34(20):18125–18141, 2022.

S. Santos-Assunçao, V. Perez-Gracia, O. Caselles, J. Clapes, and V. Salinas. Assessment of complex masonry structures with gpr compared to other non-destructive testing studies. *Remote Sensing*, 6(9):8220–8237, 2014.

P. Su, H. Han, M. Liu, T. Yang, and S. Liu. Mod-yolo: Rethinking the yolo architecture at the level of feature information and applying it to crack detection. *Expert Systems with Applications*, 237:121346, 2024.

D. Tabernik, S. Šela, J. Skvarč, and D. Skočaj. Segmentation-based deep-learning approach for surface-defect detection. *Journal of Intelligent Manufacturing*, 31(3):759–776, 2020.

A. M. A. Talab, Z. Huang, F. Xi, and L. HaiMing. Detection crack in image using otsu method and multiple filtering in image processing techniques. *Optik*, 127(3):1030–1033, 2016.

A. Tavukçuoğlu, S. Akevren, and E. Grinzato. In situ examination of structural cracks at historic masonry structures by quantitative infrared thermography and ultrasonic testing. *Journal of Modern Optics*, 57(18):1779–1789, 2010.

N. Wang, X. Zhao, L. Wang, and Z. Zou. Novel system for rapid investigation and damage detection in cultural heritage conservation based on deep learning. *Journal of Infrastructure Systems*, 25(3):04019020, 2019a.

N. Wang, X. Zhao, P. Zhao, Y. Zhang, Z. Zou, and J. Ou. Automatic damage detection of historic masonry buildings based on mobile deep learning. *Automation in Construction*, 103:53–66, 2019b.

G. Wei, F. Wan, W. Zhou, C. Xu, Z. Ye, W. Liu, G. Lei, and L. Xu. Bfd-yolo: A yolov7-based detection method for building façade defects. *Electronics*, 12(17):3612, 2023.

H. Xu, X. Su, Y. Wang, H. Cai, K. Cui, and X. Chen. Automatic bridge crack detection using a convolutional neural network. *Applied Sciences*, 9(14):2867, 2019.

X. Xu, M. Zhao, P. Shi, R. Ren, X. He, X. Wei, and H. Yang. Crack detection and comparison study based on faster r-cnn and mask r-cnn. *Sensors*, 22(3):1215, 2022.

K. Yan and Z. Zhang. Automated asphalt highway pavement crack detection based on deformable single shot multi-box detector under a complex environment. *IEEE Access*, 9: 150925–150938, 2021.

P. Yan, Y. Zou, J. Zhou, W. Lu, Y. Zhang, and L. Liu. Assessment of seismic impact on residences during blasting excavation of a large-scale rock slope in china. *Environmental Earth Sciences*, 75:1–15, 2016.

A. Zhang, K. C. Wang, Y. Fei, Y. Liu, S. Tao, C. Chen, J. Q. Li, and B. Li. Deep learning–based fully automated pavement crack detection on 3d asphalt surfaces with an improved cracknet. *Journal of Computing in Civil Engineering*, 32(5):04018041, 2018a.

K. Zhang, Y. Zhang, and H. Cheng. Self-supervised structure learning for crack detection based on cycle-consistent generative adversarial networks. *Journal of Computing in Civil Engineering*, 34(3):04020004, 2020.

N. Zhang, D. Lei, and J. Zhao. An improved adagrad gradient descent optimization algorithm. In *2018 Chinese Automation Congress (CAC)*, pages 2359–2362. IEEE, 2018b.

M. Zhao, P. Shi, X. Xu, X. Xu, W. Liu, and H. Yang. Improving the accuracy of an r-cnn-based crack identification system using different preprocessing algorithms. *Sensors*, 22(18):7089, 2022.

Y. Zhu, W. Xu, C. Cai, and W. Xiong. Relative-breakpoint-based crack annotation method for lightweight crack identification using deep learning methods. *Applied Sciences*, 13(15): 8950, 2023.

Table 1: Number of model training/validation/test images

	Concrete	Concrete and cob	Cob	Stone	Stone and brick	Brick	Tile
Train	1005	1865	860	788	2693	1095	167
Test	111	206	95	94	212	118	41
Total	1116	2071	955	882	2905	1213	208

Table 2: Used hyperparameters for training the YOLOv5 model

Parameter	Amount
Input shape	416 × 416
Batch size	16
Learning rate	0.01
Optimizer	SGD
Epochs	100
Data augmentation	Image rotation 45°
Pre-trained weight	COCO
Loss function	Box loss function, Obj loss function
Performance metrics	Precision, recall, mAP
Environment	Google Colab

Table 3: Performance metrics corresponding to crack detection for each material

Materials	Precision (%)	Recall (%)	mAP50 (%)	mAP50:95 (%)
Concrete	92.6	89.2	94.4	76.5
Concrete and cob	93.2	90	93.9	69.2
Cob	92.0	87.9	92.7	65.8
Stone	86.3	80.3	87.2	63.5
Stone and brick	80.6	77.5	83.4	59.4
Brick	82.6	77.0	81.6	55.2
Tile	71.2	67.4	70.3	47.1

Table 4: Loss function of YOLOv5 corresponding to crack detection for each material

Materials	box loss	obj loss	Time (minutes)
Concrete	0.02462	0.01415	21
Concrete and cob	0.02809	0.01704	39
Cob	0.03064	0.01968	18
Stone	0.02891	0.02146	17.5
Stone and brick	0.03119	0.0209	40
Brick	0.03307	0.02008	21.5
Tile	0.05102	0.04512	10

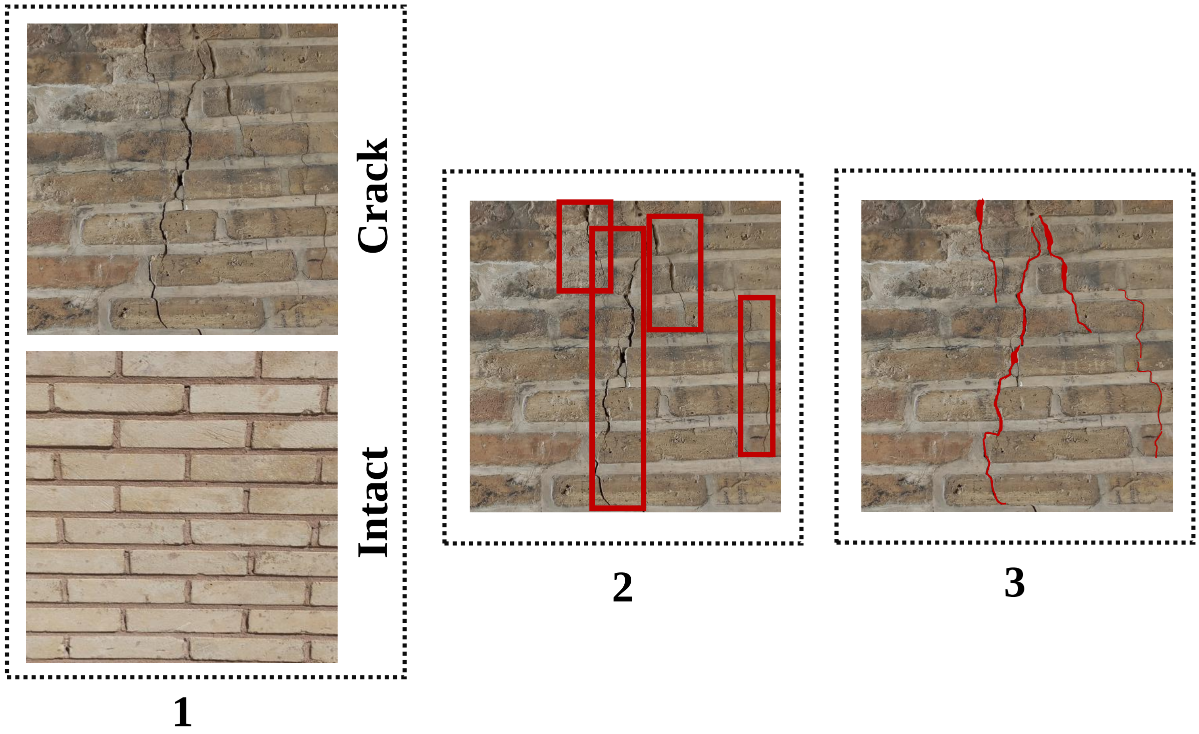


Figure 1: Three categories of crack detection by DL (1. classification, 2. object detection, and 3. semantic segmentation)

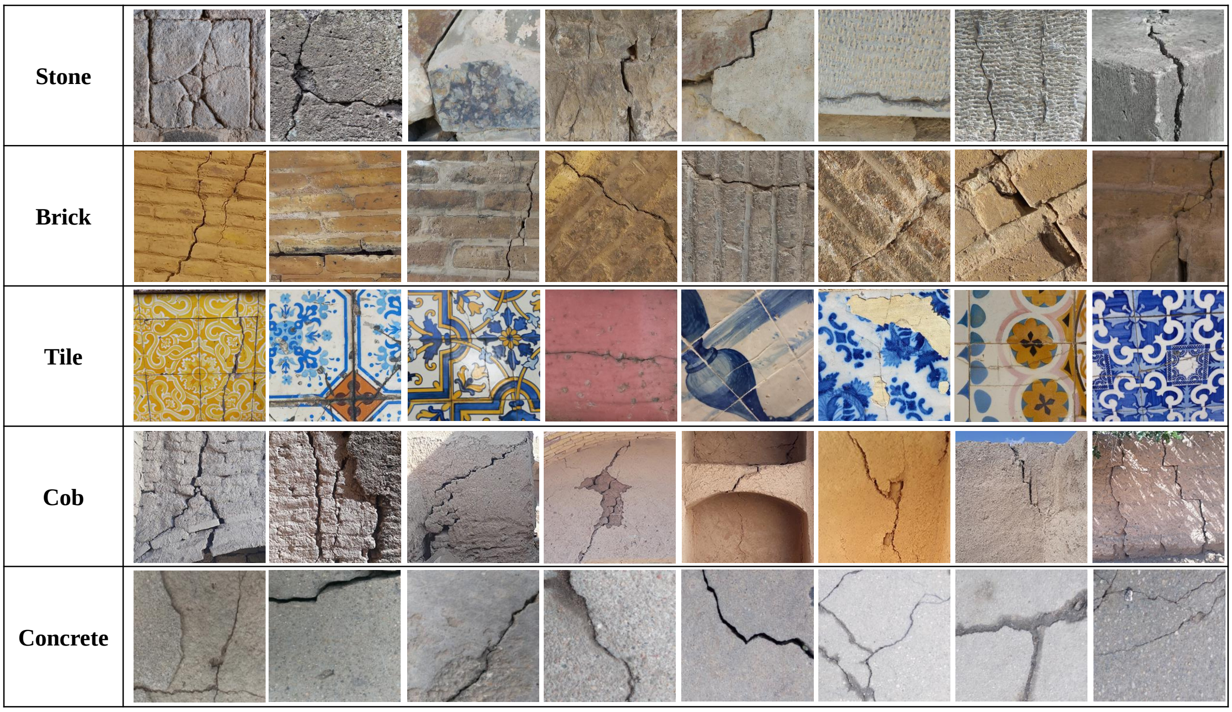


Figure 2: Sample images from five datasets showcasing various materials with cracks (stone, brick, tile, cob, and concrete)

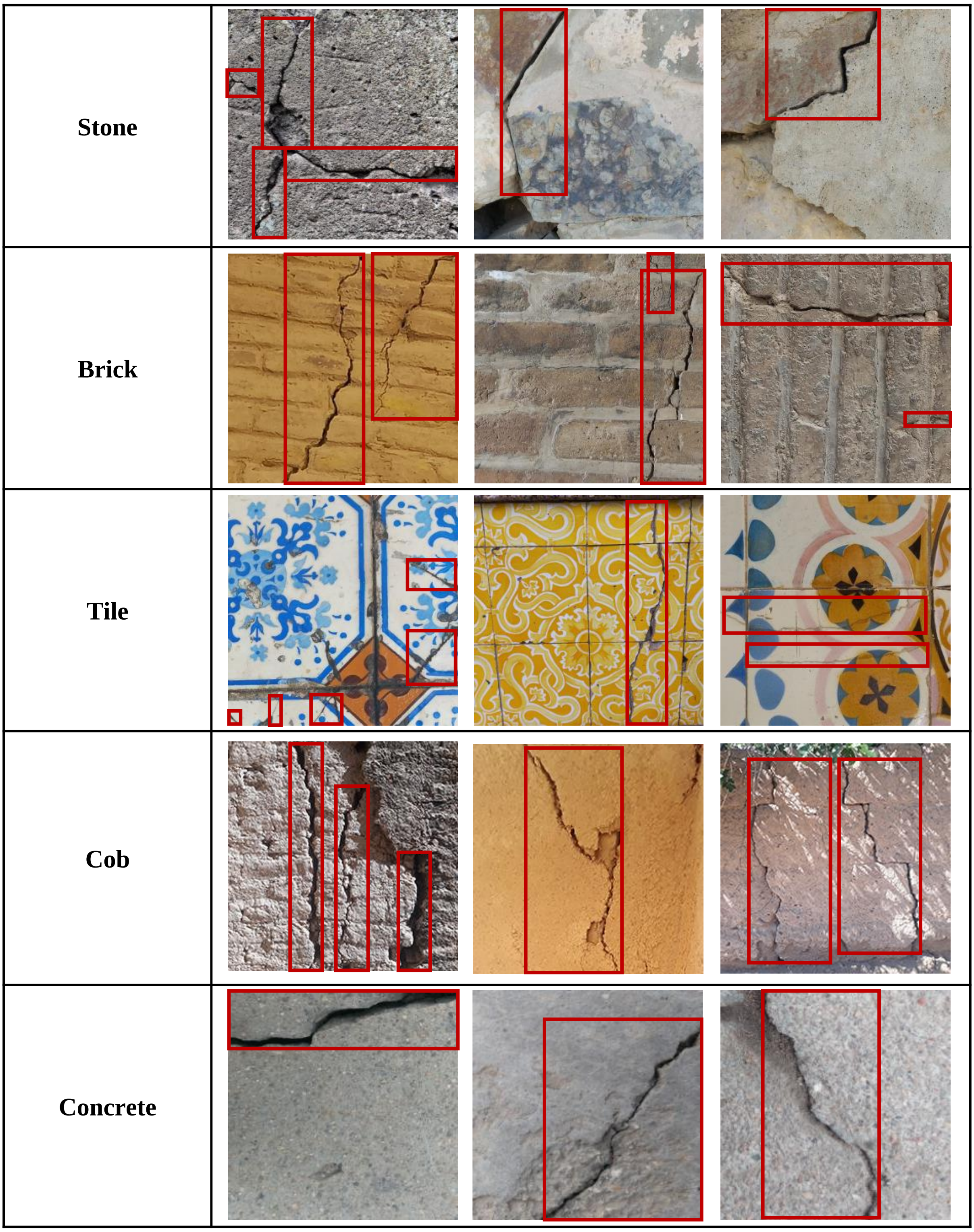


Figure 3: Sample annotated images of cracks by bounding boxes in different datasets for the YOLOv5 model

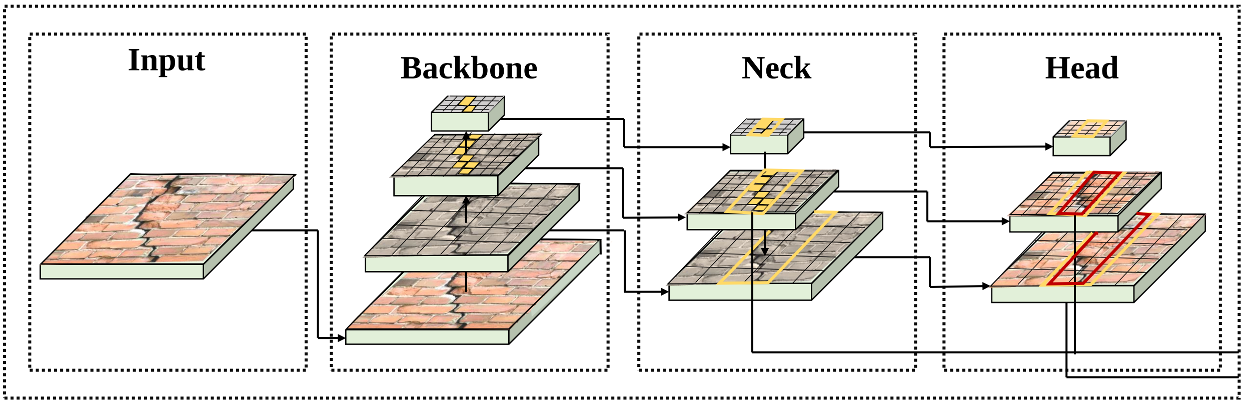


Figure 4: Applied YOLOv5 structure for crack detection

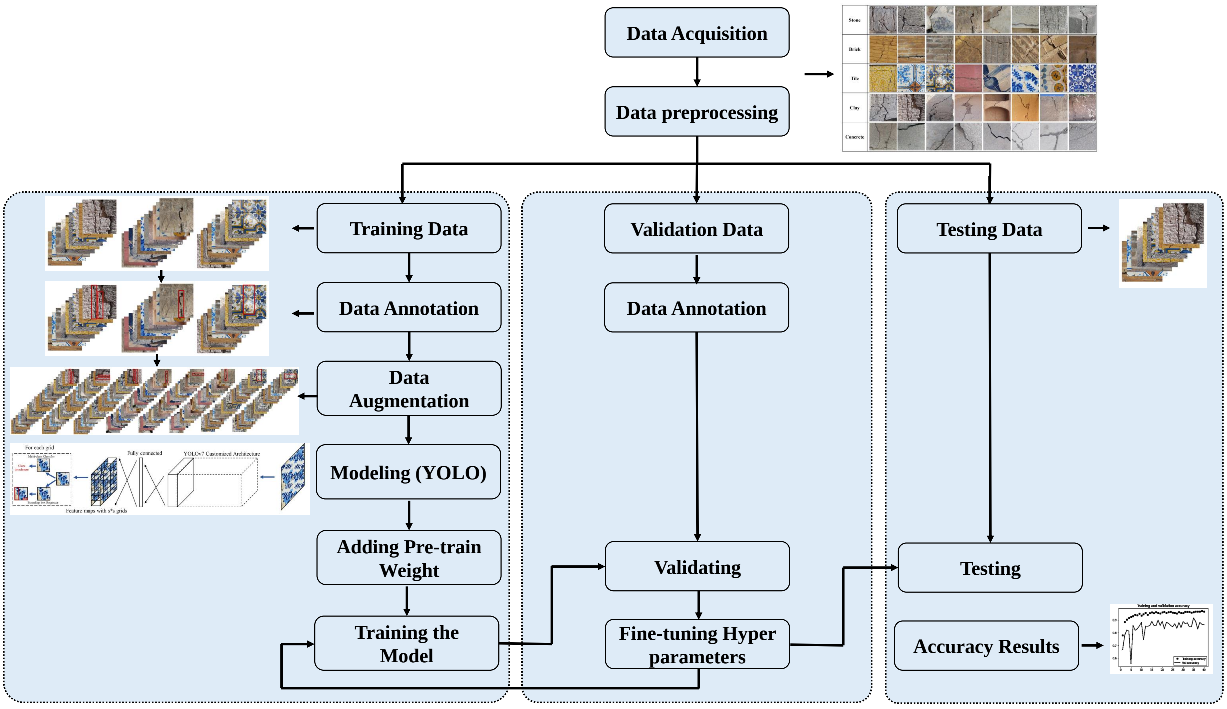


Figure 5: Flowchart of the YOLOv5 method applied for crack detection in different materials

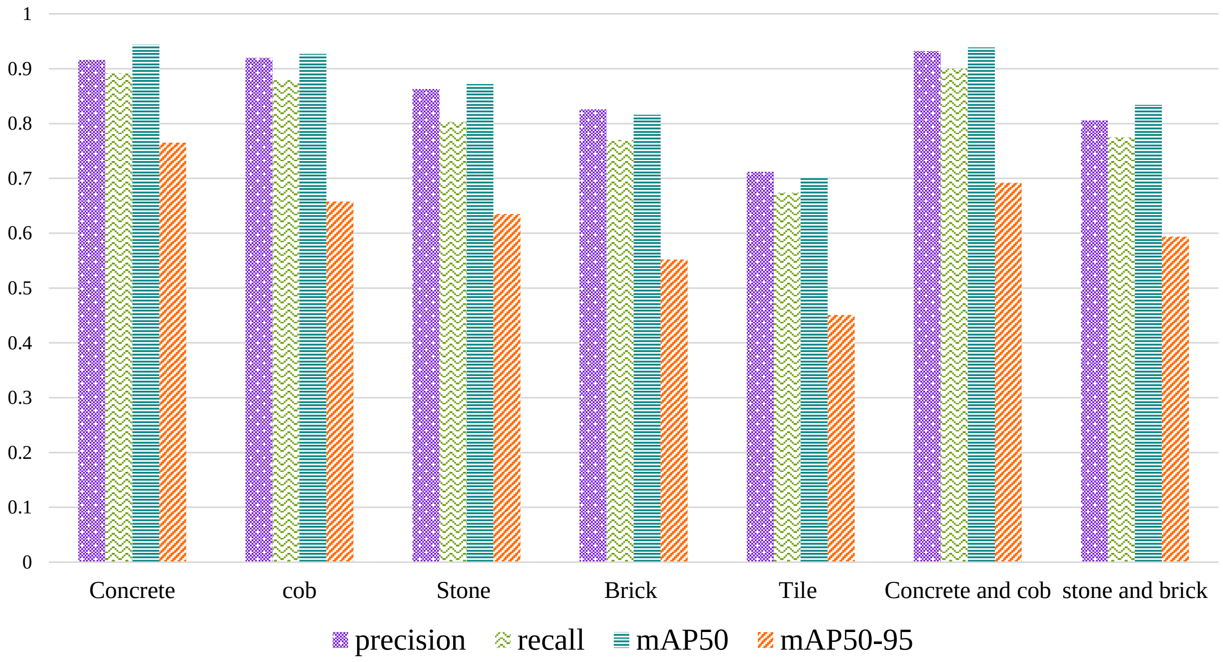


Figure 6: Comparison of performance metrics for YOLOv5 across different materials

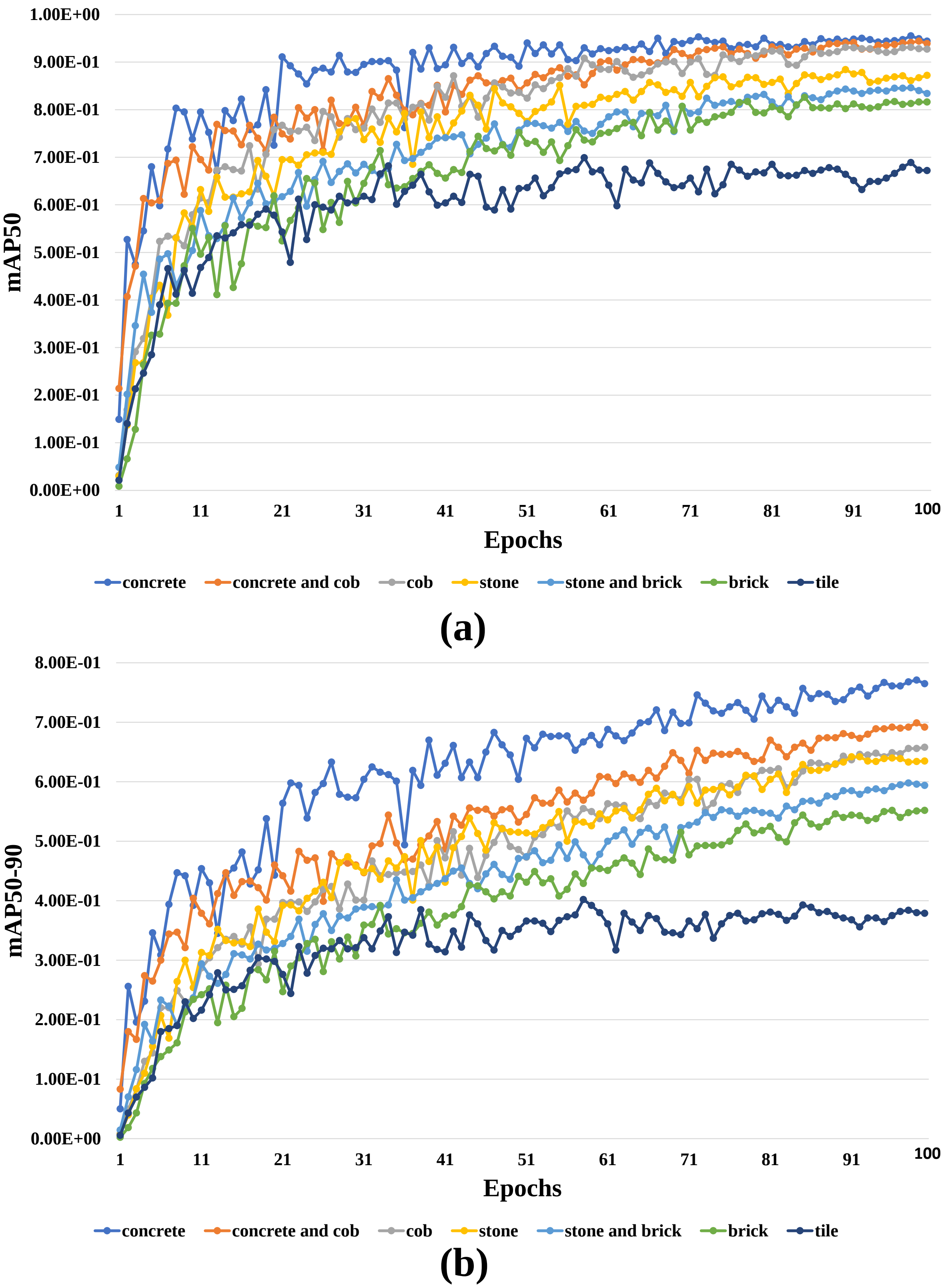


Figure 7: (a) mAP_0.5 and (b) mAP_0.5:95 metrics for different materials (concrete, brick, cob, stone, and tile) achieved using YOLOv5

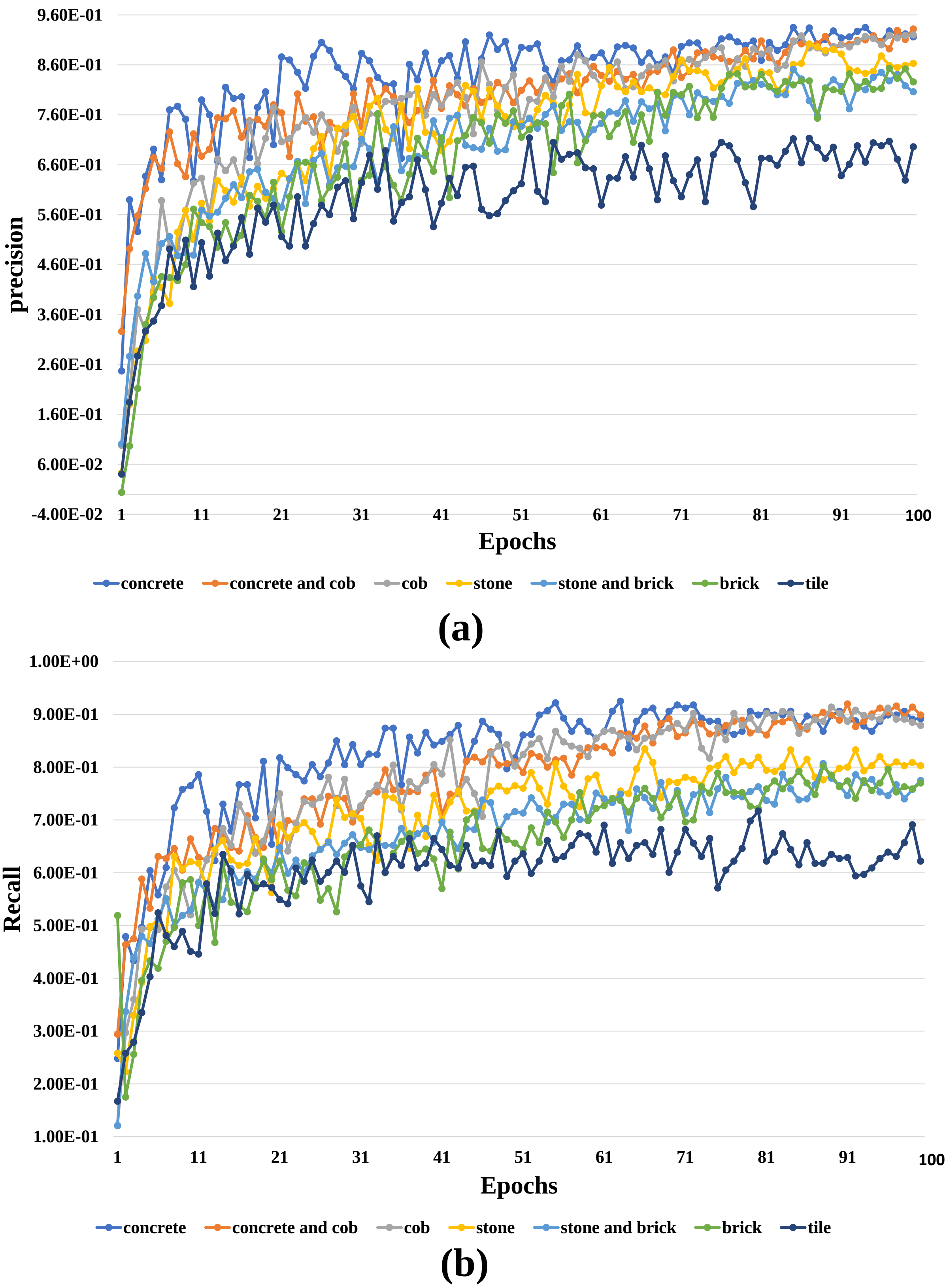


Figure 8: Comparison of (a) recall (b) precision metrics for YOLOv5 across different materials (concrete, brick, cob, stone, and tile)

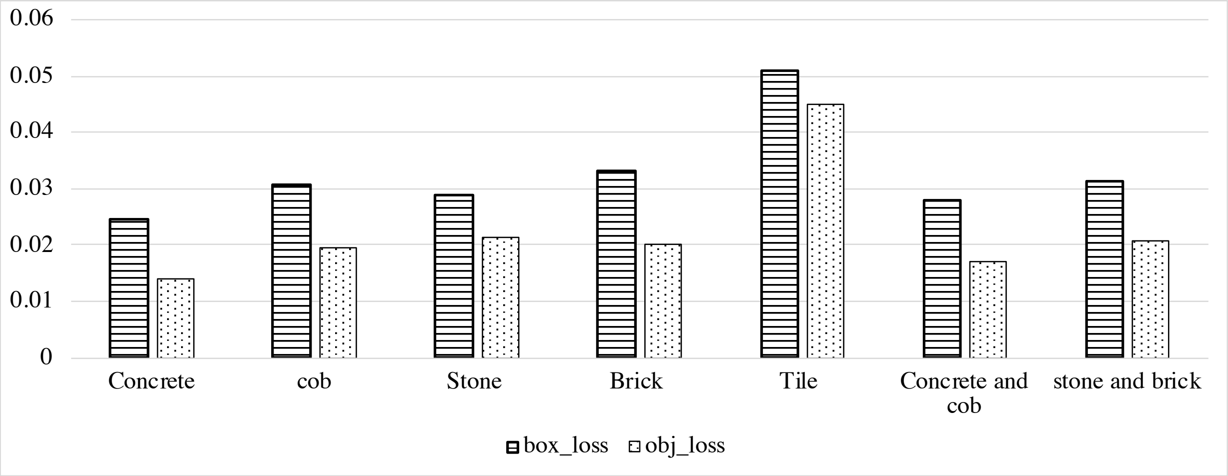
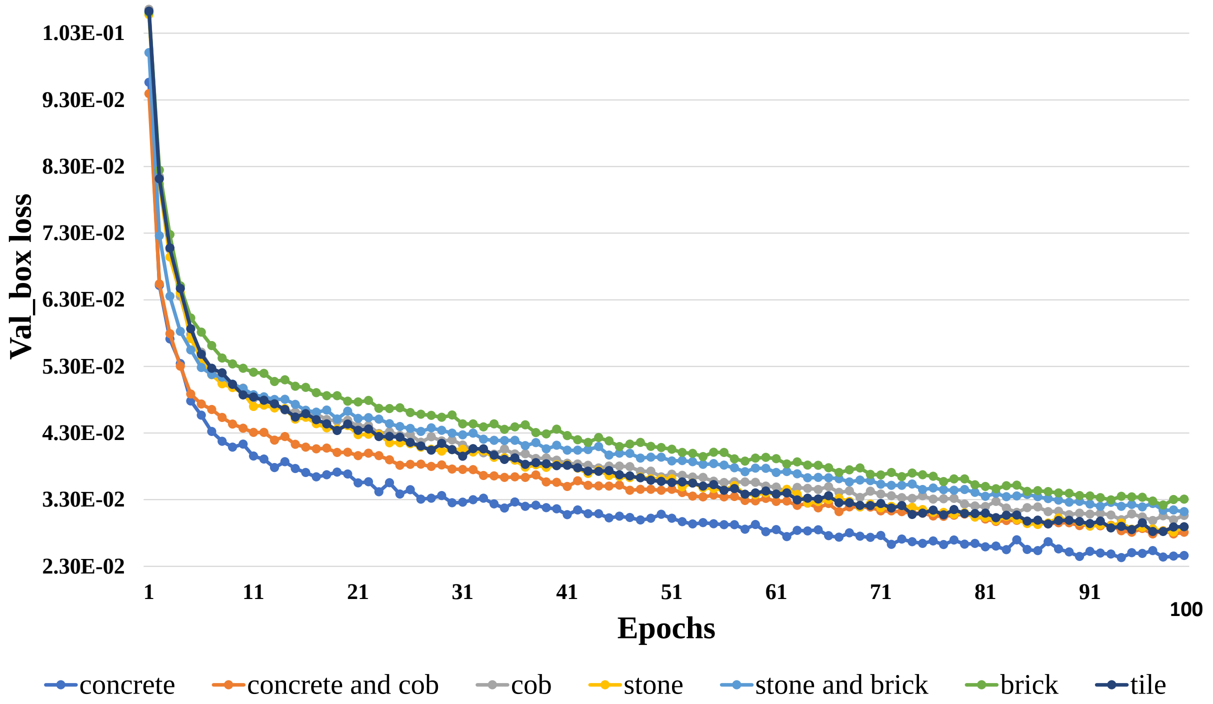
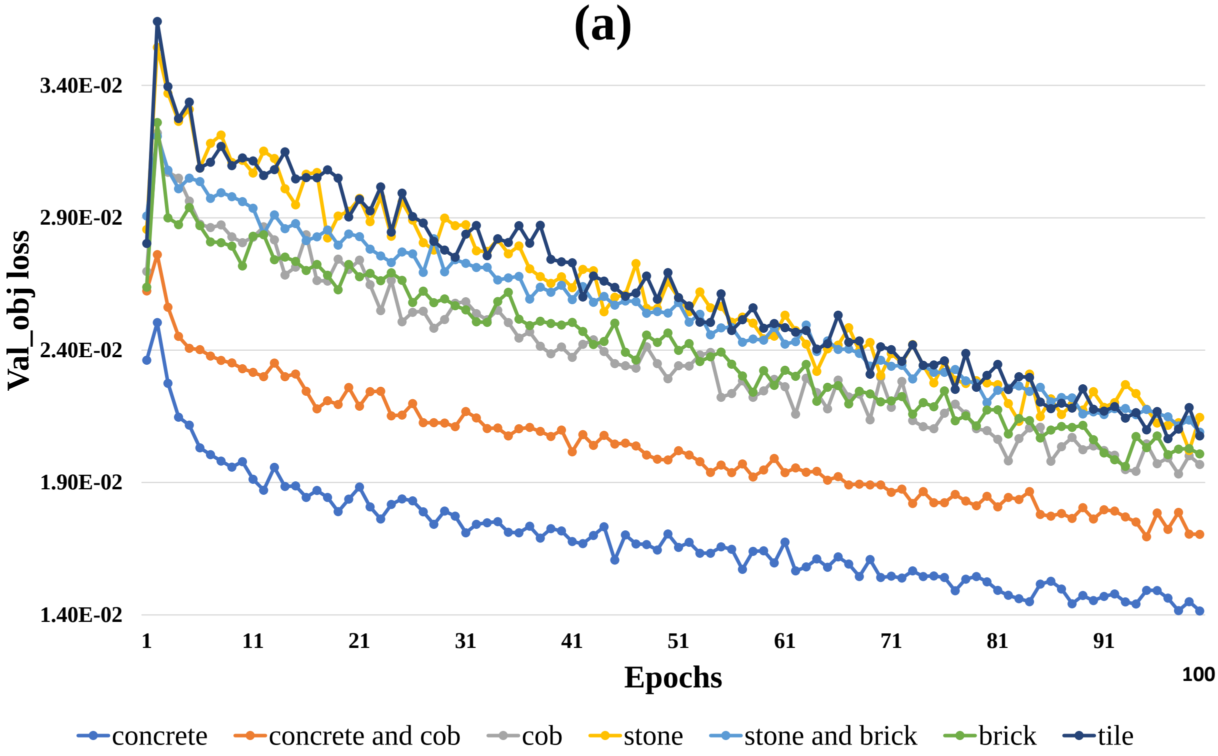


Figure 9: Comparison of loss metrics for YOLOv5 across different materials



(a)



(b)

Figure 10: (a) Val_box loss (b) Val_obj loss comparison for YOLOv5 across different materials (concrete, brick, cob, stone, and tile)

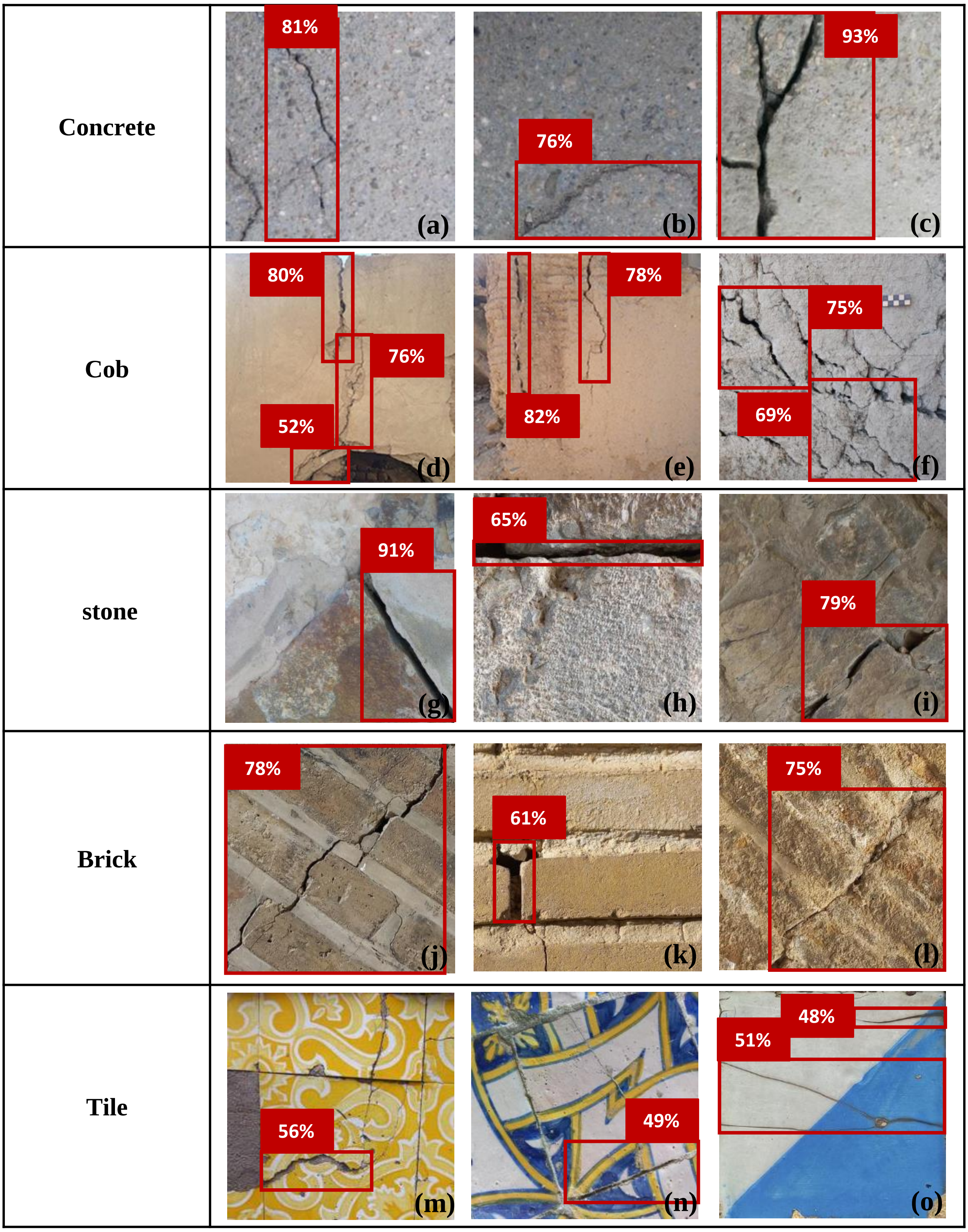


Figure 11: Crack detection results obtained using YOLOv5 for unseen images across different materials (concrete, brick, cob, stone, and tile)