Waste Management 167 (2023) 194–203

Contents lists available at ScienceDirect

Waste Management

journal homepage: www.elsevier.com/locate/wasman

Research Paper

Real-time construction demolition waste detection using state-of-the-art deep learning methods; single–stage vs two-stage detectors

Demetris Demetriou [a,*], Pavlos Mavromatidis [b], Ponsian M. Robert [b], Harris Papadopoulos [b], Michael F. Petrou [a], Demetris Nicolaides [b,c]

[a] Department of Civil & Environmental Engineering, University of Cyprus, Nicosia 1303, Cyprus
[b] Frederick Research Centre, Nicosia 1036, Cyprus
[c] Frederick University, Nicosia 1036, Cyprus

ARTICLE INFO

Keywords:
Construction and Demolition Waste
Object detection
Waste sorting
Deep learning
Convolutional neural networks

ABSTRACT

Central to the development of a successful waste sorting robot lies an accurate and fast object detection system. This study assesses the performance of the most representative deep-learning models for the real-time localisation and classification of Construction and Demolition Waste (CDW). For the investigation, both single-stage (SSD, YOLO) and two-stage (Faster-RCNN) detector architectures coupled with various backbone feature extractors (ResNet, MobileNetV2, efficientDet) were considered. A total of 18 models of variable depth were trained and tested on the first openly accessible CDW dataset developed by the authors of this study. This dataset consists of images of 6600 samples of CDW belonging to three object categories: brick, concrete, and tile. For an in-depth examination of the performance of the developed models under working conditions, two testing datasets containing normally and heavily stacked and adhered samples of CDW were developed. A comprehensive comparison between the different models yields that the latest version of the YOLO series (YoloV7) attains the best accuracy ($mAP_{50:95} \approx 70\%$) at the highest inference speed (<30 ms), while also exhibiting enough precision to deal with severely stacked and adhered samples of CDW. Additionally, it was observed that despite the rising popularity of single-stage detectors, apart from YoloV7, Faster-RCNN models remain the most robust in terms of exhibiting the least *mAP* fluctuations over the testing datasets considered.

1. Introduction

Construction and Demolition Waste (CDW) makes up more than one-third of all waste produced in the European Union (EU) and is the largest waste stream in volume (Bilsen et al., 2018). It is thus not surprising that the EU has identified CDW as a priority waste stream, placing the implementation of appropriate management of CDW in the centre of the European agenda for Circular Economy, and at the core of efforts toward greening the built environment.

CDW consists of a wide variety of materials such as concrete, brick, wood, glass, metal, and plastic, with a significant resource value. This value can be unlocked by recycling and reusing the material in new products and high-value applications. One of the common inhibitors to recycling and reusing CDW is the lack of confidence in the quality of the recycled material, which ultimately restricts the demand for CDW.

Conventional recycling methods involve breaking down coarse CDW by jaw crushers and screening the material for the removal of soil, residue, and gravel. The screened material is then subjected to magnetic and air separation for recovering metals and removing lighter substances such as plastic bags, cardboard, etc. Manual sorting of CDW is performed at the final stage for separating the waste into its individual constituents. This step is critical for ensuring the purity of CDW, thus increasing its value and recyclability. Manual sorting of CDW, nonetheless, is found to be inconsistent, unreliable, expensive (Davis et al., 2021), and hazardous for the people involved in the sorting process (Sarc et al., 2019). For these reasons, automated sorting of CDW using robots has been proposed for on-site (Chen et al., 2022; Wang et al., 2019, 2020) and off-site recycling (Bosoc et al., 2021; Lukka et al., 2014; Xiao et al., 2020), and can be arguably considered a significant step toward implementing effective waste management and aiding greening attempts in the building sector. Yet obviously, the success of a sorting robot heavily relies on its capacity to locate and classify waste.

Convolutional Neural Networks (CNN) exhibit excellent object detection and classification characteristics and have been successfully

* Corresponding author.
E-mail address: demetriou.c.demetris@ucy.ac.cy (D. Demetriou).

https://doi.org/10.1016/j.wasman.2023.05.039
Received 23 December 2022; Received in revised form 5 May 2023; Accepted 24 May 2023
Available online 1 June 2023

used for the detection of solid waste. In recent years, the TrashNet benchmark dataset proposed by Yang and Thung (2016) serves as a baseline for the development of various object detection models on mainstream CNN architectures with great reported success (Aral et al., 2018; Mao et al., 2021). Still, it is thought that TrashNet is not capable of capturing the complexity of solid waste in working conditions which is typically irregular and covered with dust (Li et al., 2022b; Yu et al., 2020). This problem of heterogeneity and surface contamination is clearly amplified in the case of CDW. In an attempt to mitigate this effect, Li et al. (2022b) proposed the fusion of RGB and depth images within a mask-Region-based CNN (RCNN) framework to improve the detection capacity of CDW detection models and examined its performance on a proprietary CDW dataset. The fusion of RGB and depth images was shown to increase the robustness of the model to the complex solid waste conditions by increasing mean Average Precision (*mAP*) by 2% compared to an equivalent RGB mask-RCNN model. Ku et al. (2021) proposed an RCNN model for improving the grasping efficiency of a CDW waste sorting robot, reporting 80% accuracy at test time. Na et al. (2022) investigated the effect of data augmentation on the development of CNN models, quantitively capturing the effect of each data augmentation procedure (e.g. noise, blur, sigmoid contrast, brightness, etc.) on improving CDW model performance. Wang et al. (2020) developed an RCNN and a Faster-RCNN model for incorporation into a CDW sorting robot. Their investigation demonstrated that Faster-RCNN models can quickly identify CDW, while mask-RCNN models can be more accurate in recognizing the pose of the target object, thus increasing the success of the grasping system.

The presented studies suggest a consensus regarding the use of deep neural networks for mining semantic information contained in CDW images. However, deep network architectures when deployed on two-stage detector heads, such as the ones comprising the RCNN, mask-RCNN and Faster-RCNN networks make the inference process slow, increasing the difficulty and cost of performing CDW sorting in real-time, particularly when considering that conveyor transportation speeds should be kept as high as possible for sorting the large volumes of waste. Slow inference is a result of the complex pipeline of two-stage detectors, which are required i) to identify regions where objects are expected to be found through a Region Proposal Network (RPN) and ii) to identify objects within the proposed regions using fully convolutional networks. To overcome this limitation, object detectors such as the SSD (Single-Shot-Detector) (Liu et al., 2016) and YOLO (You-only-look-once) (Bochkovskiy et al., 2020; Jocher, 2021; Li et al., 2022a; Redmon et al., 2015; Redmon and Farhadi, 2016, 2018; Wang et al., 2022) have been proposed for performing the object detection task in a single pass (hence 'single-shot' and 'look once'). The performance of single-stage and two-stage detectors has been exhaustively investigated on large benchmark datasets such the Common-Objects in Context (COCO) (Lin et al., 2014) on which detectors are required to distinguish between 80 different classes, and the findings exhibit a clear superiority of the latter in terms of *mAP* (mean Average Precision for the 80 classes). Still, accuracy gain margins between the two detector types might be narrower for problem specific domains (García et al., 2021), where a detector is required to distinguish between fewer classes and does not need to generalise to different types of objects such as in the COCO. Additionally, apart from generalised metrics of accuracy, other measures of performance are essential in the CDW sorting scenario, such as computational resource consumption and inference speed. Consequently, a comparison of single-stage and two-stage detectors at different configurations is rather meaningful for drawing conclusions on the trade-off between speed and accuracy between the two object detection frameworks for the CDW sorting task.

1.1. Research significance

Motivated by the need of increasing current CDW recycling rates and improving the purity of the recovered material, this study makes the following contributions:

1) It develops and presents the first open access CDW dataset, providing the community with a baseline and benchmark for comparison. The dataset can be accessed on Mendeley Data (https://doi.org/10.17632/24d45pf8wm.1) (Demetriou et al., 2022) where training and testing images along with their annotations are available to download.
2) It proposes the incorporation and utilisation of secondary and independent testing dataset containing heavily stacked and adhered samples of CDW as a more representative way of evaluating the performance of the detectors in a real-life setting.
3) It provides an exhaustive investigation and comparison of the performance of the most representative (and state-of-the-art) single-stage and two-stage detectors for quantifying the effect of framework selection on the accuracy and inference speed, thus guiding efforts in the most promising direction.
4) It implements the investigated object detection models on a CDW small-scale sorting prototype where the investigated models provide all the necessary information to the mechatronic sorting system for executing the CDW sorting task.

2. Methodology

2.1. Dataset description

This investigation focuses on the performance evaluation of the most representative single-stage and two-stage deep-learning model architectures for the detection of concrete, brick, and tile in the CDW stream. These object categories were specifically selected based on their intrinsic value and high potential for valorisation in several applications, such as recycled concrete aggregate, fire insulation/proofing, and geopolymers. It is important to note that this work is part of the Development of an innovative insulation fire resistant façade from the Construction and Demolition Waste (DEFEAT) project, co-funded by the European Union, which aims to promote the sustainable reuse of CDW by developing innovative technologies for valorising recycled brick and tiles in geopolymerisation processes. Indeed, many recent studies have focused on the utilisation of recycled concrete, brick and tile in the development of new products such as recycled concrete aggregate (Dimitriou et al., 2018; Oikonomopoulou et al., 2020a, 2022b; Savva et al., 2021), as well as fire insulation/proofing and geopolymers type applications (Luhar et al., 2021; Robert et al., 2023; Valanides et al., 2023), highlighting the apparent environmental and economic benefit from their recycling and reuse. It is important to highlight that these three object categories are extracted from the final CDW sorting stage which is typically performed manually owing to the similarity of material density and composition which prohibits the use of conventional automated methods such as air and magnetic separation.

For the development of the training and testing datasets, samples of concrete, brick, and tile were randomly extracted from piles of sorted CDW at a recycling facility in Cyprus. The samples used in this study were collected from multiple sites, reflecting the legal requirement of the sorting facility to receive waste from various sources. It is also noteworthy that typical structures in Cyprus are constructed using reinforced concrete, brick, and roof tiles, manifesting the abundance of these materials in the waste stream. To ensure that the material was representative of what is typically found in the waste stream, all samples were used as received. However, it should be noted that since the proposed sorting method is currently capable of sorting at a small-scale prototype level at a maximum sorting capacity of 14 aggregates/minute (limited by the capacity of the robot's actuators), for cases where the material was larger than the robotic manipulator's grippers (70 mm), the material was further crushed, ensuring efficient picking during the sorting process. Finally, digital images of the material were recorded in a controlled environment on the conveyor belt of the prototype platform

Fig. 1. (a) Photograph and (b) schematic Illustration of the prototype CDW sorting station comprised of (A) an RGB camera, (B) a robotic arm manipulator and (C) a conveyor belt.

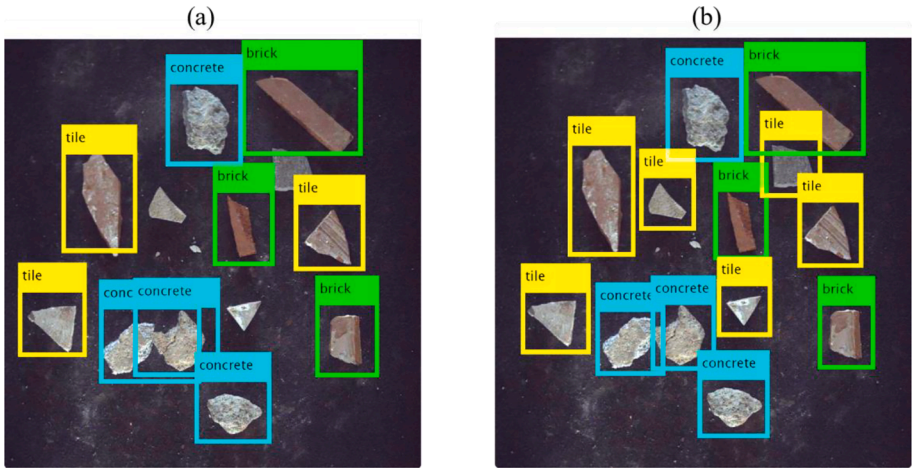

Fig. 2. (a) Labelling objects using teacher models, (b) final labels post adjustment.

shown in Fig. 1. The prototype consists of an industrial camera overlooking the workspace, a conveyor belt that carries the material, two robotic manipulators for sorting the waste and a dedicated local computer. Upon placement of the material on the conveyor belt of the prototype platform, images of CDW were recorded with the use of a HIKROBOT MV-CA023-10GC camera. The original size of the captured images was 1920 × 1200 × 3, with the colour channel being RGB. A collection of 500 multi-class images, containing approximately 4 samples of each object class (brick, concrete and tile) were recorded, resulting in a total of approximately 2000 samples of each object category. Maintaining the same number of samples of each object class in every image balances the dataset and mitigates the overfitting and underfitting phenomena observed owing to class overrepresentation and underrepresentation respectively. This ultimately allows for a more meaningful and unbiased evaluation of the final detection models.

For labelling the objects, a semi-automated method was adopted (Li et al., 2022b). Firstly, manual labelling was performed on the first 100 images and a coarse "teacher" model was trained for 50 epochs on the compressed and faster version of the YoloV4 network, namely YoloV4-tiny (Bochkovskiy et al., 2020) comprising of a compressed version of the DarkNet-53 backbone. This model was used to predict approximate bounding box locations and object labels on the succeeding 50 images. Manual adjustment of the bounding boxes and re-labelling was performed on the misclassified and unidentified objects of these 50 images (Fig. 2a). Subsequently, a new model was trained, using the previous model weights as a starting point, for an additional 50 epochs on the 150 now correctly labelled images, resulting in an improved model. This process was performed iteratively for every subsequent batch of 50 images until all 500 images were labelled and adjusted. Fig. 2(a) and Fig. 2(b) present a sample of the labels obtained from the 'teacher' models and the final labels utilised as ground truth after manual adjustment.

In accordance with good machine learning practice, 70% of the samples (4230 samples) in the labelled images were used in the training dataset for developing the object detection models, while the remaining 30% of samples in the labelled images (1727 samples) were retained for developing the first testing dataset (testing set_1). This dataset represents an idealised case of CDW placement where the items are sparsely spaced on the conveyor belt (Fig. 2). This dataset can serve as a baseline indicator of model performance; however, it is anticipated that degradation will occur because of the presence of stacking and adherence of samples. Stacking occurs when samples are placed on top of each other, while adherence occurs when samples are in contact. For the former, least sophisticated object detectors tend to misclassify or even completely miss stacked objects, while for the latter they tend to place a single bounding box to the adhered samples, particularly if the samples belong to the same class. An example of adherence and poor model performance can be observed in Fig. 2(a) where the teacher model significantly overlapped the two adhered concrete samples.

To address the issue of stacking and adherence, a secondary testing

Fig. 3. Testing set_2: Adhered and stacked samples of CDW on the conveyor belt.

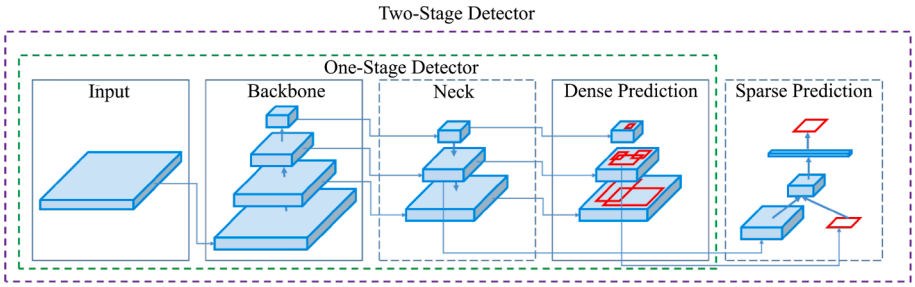

Fig. 4. One and two-stage object detector architecture (Bochkovskiy et al., 2020).

dataset (testing set_2) containing 596 heavily stacked and adhered samples (Fig. 3) was developed and used to give a more representative evaluation of the models under working conditions. While it is possible to merge the two testing datasets into a single test set, which is often the case in most object detection development scenarios, the authors strongly believe that separating the two gives a deeper insight into the performance of the models and the complexity required to achieve good results particularly in heavily stacked and adhered cases.

2.2. Deep neural networks for object detection

Object detectors consist of three main parts, a backbone, a neck, and a head (Fig. 4). The backbone is responsible for extracting semantic information such as the shape, edges, and colour from the input and converting this information into high-dimensional feature mappings. Heavyweight backbone feature extractors intended to run on GPUs include among others the popular VGG (Simonyan and Zisserman, 2014), ResNet (He et al., 2016), and DenseNet (Huang et al., 2017) network architectures, while lightweight backbone feature extractors capable of running on CPUs include the SqueezeNet (Iandola et al., 2016), MobileNet (Andrew G Howard et al., 2017) and ShuffleNet (Zhang et al., 2018) network architectures.

In modern detectors, several layers are often inserted between the backbone and the head and are used to gather feature maps from different levels in the network. These layers are known as the neck of the detector and consist of numerous top-down and bottom-up pathways. Most representative networks incorporating such layers include the Feature Pyramid Network (FPN) (Lin et al., 2017), and Path Aggregation Network (PAN) (Liu et al., 2018).

Regarding the head, the part responsible for the prediction of bounding boxes and classes of objects, object detectors are classified as two-stage and single-stage detectors. As the name suggests, a two-stage detector performs the prediction of bounding boxes and object classes in two steps. Firstly, it generates regions of interests (RoIs) using an RPN. The RPN outputs a predefined number of bounding box proposals with scores representing the probability of an object being contained at each location proposal. In the second step, it predicts bounding boxes and classes for the proposed regions. On the contrary, single-stage detectors

Table 1
Description of investigated object detectors.

Detection head type	Model	Reference Paper	Network size (Mb)	Input layer size
Faster_RCNN	Resnet50 + C4	(Ren et al., 2016)	221	640 × 640
	Resnet101 + C4		370	640 × 640
Faster_RCNN	ResNet50 + DC5	(Dai et al., 2017)	638	640 × 640
	ResNet101 + DC5		705	640 × 640
Faster_RCNN	ResNet50 + FPN	(Lin et al., 2017)	160	640 × 640
	ResNet101 + FPN		232	640 × 640
SSD	ResNet50 FPN (RetinaNet50)	(Lin et al., 2017b)	158	640 × 640
	ResNet101FPN (RetinaNet101)		228	640 × 640
SSD	EfficentDet_D0	(Tan et al., 2020)	45	512 × 512
	EfficientDet_D1		70	640 × 640
SSD	MobilenetV2	(Sandler et al., 2019)	37	320 × 320
	MobilenetV2_FPN		20	640 × 640
YOLO	YoloV5s	(Jocher, 2021)	15	416 × 416
	YoloV5l		93	640 × 640
YOLO	YoloV6s	(Li et al., 2022a)	38	640 × 640
	YoloV6l		118	640 × 640
YOLO	YoloV7	(Wang et al., 2022)	73	640 × 640
	Yolo7x		139	640 × 640

focus on all the spatial region proposals for object detection in one single pass through the image. From this description, it can be realised that the performance of any object detector, that is detection accuracy and inference time, is influenced by two main aspects; i) the complexity and number of learnable parameters in the backbone architecture, and ii) the type of detection head used. Generally, deeper backbone architectures benefit from improved detection accuracy while suffering from slower inference times. Similarly, two-stage detection heads improve detection accuracy at the expense of inference speed.

This study investigates the performance of the most representative single-stage and two-stage deep-learning models on the CDW dataset described in section 2.1. The models are developed on three frameworks, namely Faster-RCNN, SSD and YOLO. To capture the sensitivity of the models to the depth of the backbone, each model is coupled to one shallow and one deep backbone feature extractor. Accordingly, Faster-RCNN models with ResNet conv4 (C4), conv5 (DC5) and Feature Pyramid Network (FPN) backbone combinations are developed and tested. The study also investigates SSD models including Facebook's AI Research popular RetinaNet50 and RetinaNet101 composed of the ResNet50 and ResNet101 backbones respectively, as well as SSD models composed of the lightweight MobileNetV2 and MobilenetV2 FPN architectures. Finally, in terms of models employing the YOLO meta-architectures the study investigates the most recent versions (e.g., YoloV5, YoloV6, and YoloV7) of the YOLO series which have shown great promise on the COCO dataset. From these combinations, a total of 18 models were developed. These are summarised in Table 1. It should be noted, however, that this model list is not exhaustive as other detectors such as the fully convolutional anchor-free one-stage object detector (FCOS) (Tian et al., 2020) also exist and could be investigated in future studies. Still, the scope of this study is limited to the three presented frameworks for the sake of clarity and comparability.

2.3. Deep network implementation details

The training of all deep neural networks was performed on Google's Collaboratory on an Nvidia Tesla T4 (16 GB memory) GPU. The stochastic gradient descent with momentum (SGDM) optimizer was used in all training experiments. The momentum of the optimizer was set to 0.9 and an initial learning rate of 0.001 was selected. A cosine learning rate decay with a warmup of 2500 steps was applied and a base learning rate of 8E-2 was chosen.

To satisfy GPU memory constraints, for the case of models employing heavyweight backbone architectures, e.g., EfficientDet_1, ResNet50 and ResNet101 a batch size of 8 was selected. The remaining models were trained with a batch size of 16 while the 'ultra-lightweight' model SSD MobilenetV2 with input resolution 320 × 320 was benefitted from an increased batch size of 32. All models were trained for at least 100 epochs, or until their training loss converged. To maintain the fairness of comparison by accounting for differences in the batch size, the number of training iterations was calculated by multiplying the targeted epoch number by the quotient of the training samples and the batch size (e.g., number of iterations = target epochs × (training samples/batch size)).

For speeding up model training, transfer learning i.e., the use of pre-trained model weights as a starting point for the training of new models for the CDW object detection task was performed. Specifically, pre-trained weights of models trained on the COCO dataset were utilised. In the case of SSD and Faster-RCNN models, configurations and associated pre-trained weights were obtained from Tensorflow's 2 (Huang et al., 2020) and Detectron's 2 (Wu et al., 2019) model zoos respectively, while in the case of YOLO model configurations and pre-trained weights were accesses through their corresponding GitHub repositories.

To prevent overfitting, which can impair the performance of deep learning models, several techniques were employed. For the case of TensorFlow 2 models, L2 regularization was applied during training with the weights presented in the corresponding model configuration files while for the case of YOLO models, batch normalization (instead of dropout) was used during training as a method of preventing overfitting.

To further enhance the performance of the object detection models and assist with mitigating overfitting phenomena, various data augmentation techniques were incorporated in the training pipeline. Specifically, the default data augmentation procedures included in the model configurations in TensorFlow 2 were utilized. These procedures included random horizontal flip and random image cropping. It is noted that the actual values attributed to each data augmentation procedure can be found in the configuration files of each model in TensorFlow 2. By applying these techniques, a new image was generated at each training iteration, which helped to prevent overfitting and increase the robustness of the models. It is noted that models belonging to the YOLO series employed a different data augmentation procedure, which is described in the "bag-of-freebies" training tricks as presented in the respective papers (Bochkovskiy et al., 2020; Jocher, 2021; Li et al., 2022a; Wang et al., 2022). The YOLO data augmentation procedure consists of: HSV-Hue augmentation, HSV-Saturation augmentation, image HSV-Value augmentation, image translation, scale, random rotation at 90 degrees, mix-up and mosaic augmentation. The GitHub repositories for YOLO5, YOLO6, and YOLO7 provide the default values attributed to each of the considered augmentations.

For all models, the default anchor-box configurations as presented in the original papers was retained and non-maximum suppression with a confidence threshold of 0.7 was applied as a post-processing step for the derivation of the final bounding boxes.

3. Evaluation

The most popular evaluation metrics in object detection are Average Precision (*AP*) and mean Average Precision (*mAP*). *AP* is defined as the average detection precision under various recalls and is evaluated independently for each considered object class. The average *AP* between all considered classes, *mAP*, allows a more comprehensive evaluation of the detector's performance.

AP is calculated through numerical integration of the precision-curve:

$$AP = \int_0^1 p(r)dr \tag{1}$$

where p is precision and r denotes recall.

As a first step toward obtaining the precision-recall curve, classification predictions and the intersection-over-union (IOU) between the predicted (B_p) and ground truth (B_{gt}) bounding boxes needs to be calculated for every image in the testing datasets as:

$$IOU = \frac{area(B_p \cap B_{gt})}{area(B_p \cup B_{gt})} \tag{2}$$

Depending on the *IOU* and classification prediction, true positive, false positive and false negative detections are defined as follows:

True Positive (*TP*): Correct prediction (IOU > threshold + correct classification).

False positive (*FP*): Incorrect prediction (IOU < threshold).

False Negative (FN): Object not detected.

Accordingly, precision is defined as the percentage of correct detection with respect to all detections and is given by:

$$p = \frac{TP}{TP + FP} \tag{3}$$

Recall is the percentage of correct detections with respect to the ground truth and is given by:

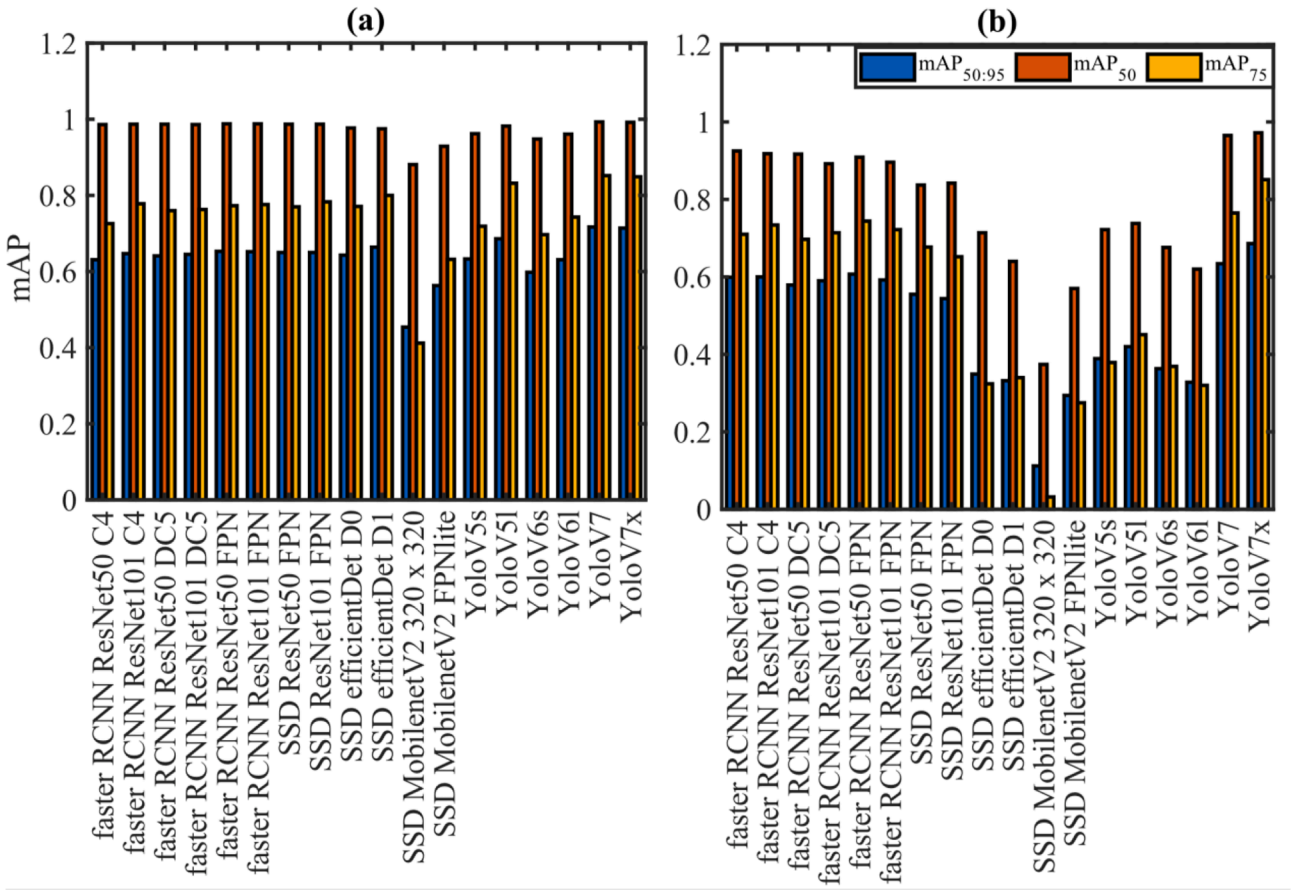

Fig. 5. *mAP* of investigated models for (a) testing set_1 and (b) testing set_2.

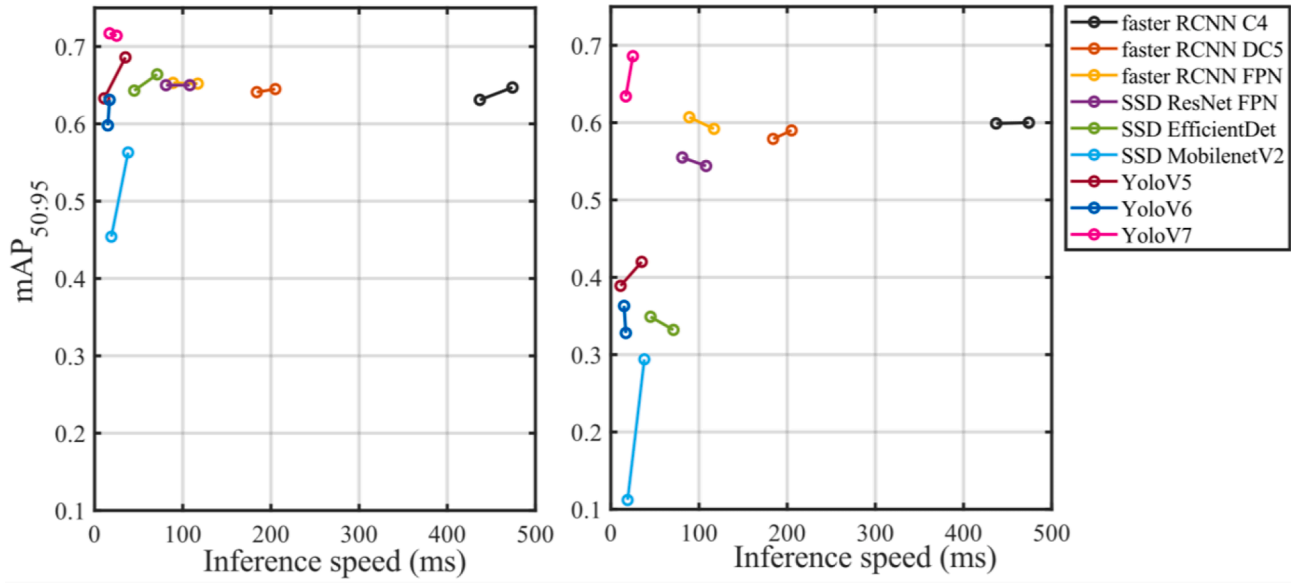

Fig. 6. Precision vs inference speed for (a) testing set_1 and (b) testing set_2 for each object detector couple comprised of a shallow and deep backbone.

$$r = \frac{TP}{TP + FN} \tag{4}$$

It is noted that precision is calculated as the number of accumulated *TP* divided by the sum of accumulated *TP* and accumulated *FP*, while recall is calculated as the number of accumulated *TP* divided by the sum of all ground truths. From this, it is realised that recall will progressively increase as more images in the testing dataset are evaluated (with a maximum potential recall value of 1 if all ground truths are correctly identified), while precision will move in a zig-zag pattern as *FP* predictions drag precision down whilst *TP* predictions bring it back up.

In this study, *AP* is calculated for several *IOU* thresholds in line with the evaluation metrics used in the Pascal VOC and COCO competitions.

Fig. 7. Inference results on testing set_2 from YoloV7x model (top), YoloV5l model (bottom).

In this regard, *AP* is calculated at the baseline (Pascal VOC metric) and strict *IOU* thresholds of 0.50 and 0.75, yielding $AP_{0.5}$ and $AP_{0.75}$ respectively. Furthermore, 10 *IOU* thresholds ranging between 0.50 and 0.95 at an *IOU* interval of 0.05 are considered and averaged to obtain $AP_{55:95}$ for each class. Similarly, $mAP_{50,}$ mAP_{75} and $mAP_{50:95}$ correspond to the averaged *AP* at different thresholds for all the classes considered.

4. Results

Object detectors of different configurations were exposed to the two CDW testing datasets (testing set_1 and testing set_2) and their responses were used to construct precision-recall curves from which *mAPs* at different *IOU* thresholds were calculated. Fig. 5 summarises the achieved *mAP* of each detector on the two testing datasets at different *IOU* thresholds. Fig. 6 complements the findings of the analysis by exhibiting the performance of each detector couple (i.e., detector configuration at one shallow and one deep backbone combination) with respect to the inference speed. The reader is referred to Table A1 and Table A2 in the appendix for a more elaborate examination of each detector's performance.

With reference to Fig. 5 and Fig. 6 the following remarks can be made: i) All models except for the low-resolution SSD_mobilenetV2 320 × 320 model exhibit good performance regarding the achieved *mAP* on testing set_1. ii) For the problem domain in consideration, *mAP* is not significantly influenced by the depth of the backbone architecture. In fact, the complexity of the deeper network architectures not only increased training and inference times but in some cases (SSD EfficientDet_1, SSD ResNet101 FPN and Faster- RCNN ResNet101 FPN) it even reduced *mAP*. While not definitive, this might be attributed to model overfitting, as the authors observed a further drop in performance at higher epoch numbers. iii) For most models, a drop in performance is observed between the *mAP* achieved on testing set_1 and testing set_2, highlighting the fact that testing set_1 is not capable of capturing the accuracy of object detectors under working conditions. iv) With the exception of the low-resolution SSD_mobilenetV2 320 × 320, model performance did not degrade on images containing samples primarily belonging to a single class. This observation is supported by the high mAP achieved by the models on testing_set 1, which includes several images of samples belonging to a single class (the reader is referred to Fig. A1 in the appendix).v) From the three considered frameworks, Faster-RCNN based models exhibit the most robustness, that is the least *mAP* fluctuation between the two testing datasets. Still, it can be observed that this robustness comes at the expense of inference speed. Among the three Faster-RCNN configurations, the configurations employed with FPN are shown to achieve the best accuracy/speed performance. On the contrary, the original configuration of the Faster RCNN employing a conv4 (C4) backbone is shown to be the slowest among the assembly. vi) In terms of absolute *mAP* performance, the

latest version of Yolo, namely YoloV7 and YoloV7x achieved higher *mAP* on testing set_1 and testing set_2 respectively, with a significant margin over its counterparts. It is to the authors' understanding that the superior performance of YoloV7 and YoloV7x models is attributed to the model's novel architecture and the advanced training techniques implemented in the bag-of-freebies, a collection of state-of-the-art methods constantly evolved through the YOLO iterations for improving object detection. Still, the actual reason why this is the case remains to be investigated.vii) With regard to the trade-off between accuracy and speed, YoloV7x is evidently the best candidate as it achieves the highest *mAP* at <30 ms inference time. It is also noted that a good compromise between accuracy and speed is achieved by the RetinaNet models (e.g., SSD ResNet 50 FPN and SSD ResNet 101 FPN).

To gain a better appreciation of the performance of the developed models and illustrate the effect of performance deterioration between testing set_1 and testing set_2, detection results on heavily stacked and adhered images of CDW are presented for the case of the YoloV5l and YoloV7x in Fig. 7.

It is evident that the YoloV5l detector, despite achieving the best-of-the-rest performance on testing set_1, its performance has deteriorated significantly on images containing adhered and stacked samples. With reference to Fig. 7, YoloV5l when compared to YoloV7x achieves in general a lower confidence score on all the identified objects. Finally, compared to YoloV7x, the YoloV5l detector has misclassified, placed multiple and incorrect bounding boxes, and even completely missed several objects.

The findings of this study provide an important contribution to the field of object detection by evaluating the performance of various state-of-the-art models under a specific problem domain. While there have been numerous studies that evaluate object detection models, the findings of this study provide unique insights into the impact of model selection on the detection performance for cases of normally and heavily stacked and adhered samples of CDW. Additionally, the study highlights the importance of not simply relying on increasing the complexity of the model for better performance, as some of the deeper network architectures did not necessarily lead to better mAP performance. This is a departure from some prior studies that have focused on increasing model complexity as a means of improving performance. Additionally, the findings suggest that the YoloV7 and YoloV7x models achieve superior performance possibly owing to its innovative architecture and advanced training techniques employed within the YoloV7 framework. This adds to the growing body of literature on the effectiveness of state-of-the-art methods for improving detection performance and the results of this study provide important insights that can help guide the selection of object detection models for the CDW domain.

While this study offers valuable insights into object detection for CDW sorting, it is important to note some limitations. First, the dataset used in this study is region-specific, meaning that model developers would need to retrain their models on their own dataset to account for any regional differences. Still, the provided dataset can serve as a starting point and facilitate the process of transfer learning. Second, the lack of reliable uncertainty quantification is also acknowledged by the authors, which is crucial in real-world applications. Therefore, the authors' plan is to extend the YoloV7x model by incorporating the Conformal Prediction framework (Papadopoulos, 2008) to enhance predictions with valid confidence measures, as demonstrated in Eliades et al. (2019) and Maltoudoglou et al. (2022). It should be also noted that the scope of this study was limited to a subset of state-of-the-art models, and other recent models detectors such as the FCOS were not investigated. However, the need to investigate such models should not be overlooked, as object detection is a rapidly evolving field, and new models are constantly emerging. Finally, the addition of more object classes to the dataset is a promising prospect for the wider dissemination of the proposed method. The inclusion of more classes would enable the detection of a greater range of materials in the CDW stream and could enhance the sorting process. Future studies could explore the feasibility and performance of adding more object classes to the dataset, which would require collecting and annotating additional data.

5. Conclusions

This study presented an experimental investigation of the performance of deep learning models for the CDW object detection task. As a first step, the study presented the first openly accessible CDW dataset (https://doi.org/10.17632/24d45pf8wm.1), emphasising on the complexity of CDW in working conditions through an independent severely stacked and adhered dataset. Secondly, the study used the dataset to develop multiple model combinations based on popular architectures (Faster-RCNN, SSD and YOLO) and backbone feature extractors (ResNet, EfficientDet, MobileNetV2) of variable depth. Thirdly, an evaluation of the detectors on popular COCO metrics of precision and inference speed was performed, and the results showed a clear superiority of the YoloV7x model which achieves the highest precision ($mAP_{50:95} \approx 70\%$) at the lowest inference times (<30 ms) with a significant margin over its counterparts. While counterintuitive it was also shown that for the CDW sorting scenario in consideration, deeper backbone architectures show little to no benefit on the considered models, rather, performance gains are shown to be primarily attributed to the type of the detection head considered. Additionally, it is found that beyond YoloV7 and YoloV7x, legacy Faster-RCNN models achieve the most robustness and least *mAP* fluctuations between the normally and heavily stacked and adhered datasets. Finally, it was found that RetinaNet models achieve a sound compromise between speed and accuracy.

Declaration of Competing Interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Demetris Nicolaides reports financial support was provided by European Regional Development Fund. Demetris Nicolaides reports financial support was provided by Cyprus Research and Innovation Foundation.

Acknowledgements

The authors would like to express their sincere gratitude to the Cyprus Research & Innovation Foundation (RIF) and the European Regional Development Fund (ERDF), for funding the research project entitled "Development of an Innovative Insulation Fire Resistant Façade from the Construction and Demolition Wastes" (Contract Number: INTEGRATED/0918/0052). Additionally, the authors gratefully acknowledge Frederick University and the University of Cyprus for providing access to their facilities and data. Finally, the authors would like to express their gratitude to S. Netiates & H.Xenis Epixeiriseis LTD who provided the recycled material to support the efforts in this study.

Funding source:

This research was funded by the Cyprus Government through the Research & Innovation Foundation (RIF) and the European Regional Development Fund (ERDF) in the frame of the RESTART Programmes for Research, Technological Development and Innovation 2016–2020 (Project Budget: €1,098,880, Project Contract Number: INTEGRATED/0918/0052).

Appendix A

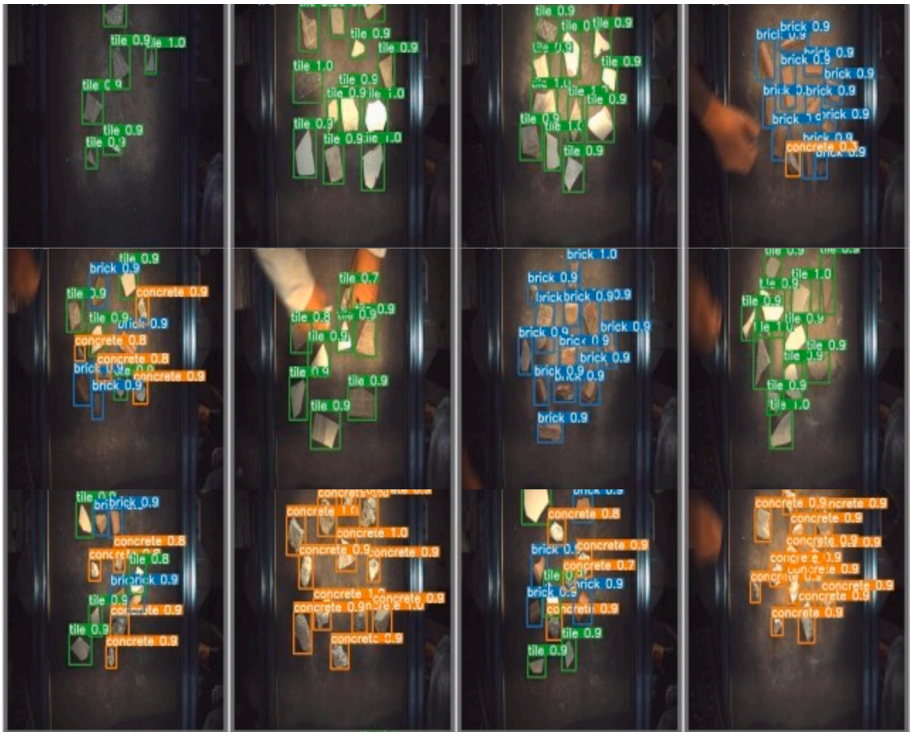

Fig. A1. Inference results on images containing class imbalanced samples (extracted from testing set_1).

Table A1
Performance of object detectors on testing set_1.

Model	$mAP_{50:95}$	mAP_{50}	mAP_{75}	Training Speed(s/it)	Inference Speed(ms)	Batch Size
Faster_RCNN_Resnet50 + C4	0.631	0.986	0.726	1.274	437	8
Faster_RCNN_Resnet101 + C4	0.647	0.987	0.778	1.881	474	8
Faster_RCNN_Resnet50 + DC5	0.641	0.987	0.760	1.817	184	8
Faster_RCNN_Resnet101 + DC5	0.645	0.986	0.763	2.464	205	8
Faster_RCNN_Resnet50 + FPN	0.653	0.988	0.773	1.434	89	8
Faster_RCNN_Resnet101 + FPN	0.652	0.988	0.776	1.970	117	8
SSD_ResNet50 + FPN	0.650	0.987	0.770	1.372	81	8
SSD_ResNet101 + FPN	0.650	0.987	0.783	1.873	108	8
SSD_efficentDet_D0	0.643	0.977	0.771	1.014	45	16
SSD_eddicientDet_D1	0.664	0.975	0.800	1.015	71	8
SSD_MobilenetV2 320 × 320	0.454	0.881	0.412	0.265	19	32
SSD_MobilenetV2 640 × 640	0.563	0.929	0.632	0.954	38	16
YoloV5s	0.633	0.962	0.719	**0.224**	**11**	16
YoloV5l	0.686	0.982	0.832	0.575	35	16
YoloV6s	0.598	0.948	0.697	0.446	15	16
YoloV6l	0.631	0.961	0.743	0.877	17	16
YoloV7	**0.717**	**0.993**	**0.852**	0.91	17	16
YoloV7x	0.714	0.992	0.849	1.27	25	16

Table A2
Performance of object detectors on testing set_2.

Model	$mAP_{50:95}$	mAP_{50}	mAP_{75}
Faster_RCNN_Resnet50 + C4	0.599	0.925	0.710
Faster_RCNN_Resnet101 + C4	0.600	0.918	0.734
Faster_RCNN_Resnet50 + DC5	0.579	0.917	0.697
Faster_RCNN_Resnet101 + DC5	0.590	0.892	0.714
Faster_RCNN_Resnet50 + FPN	0.607	0.909	0.744
Faster_RCNN_Resnet101 + FPN	0.592	0.896	0.722
SSD_ResNet50 FPN (RetinaNet50)	0.555	0.837	0.677
SSD_ResNet101 FPN (RetinaNet101)	0.544	0.842	0.652
SSD_efficentDet_D0	0.349	0.714	0.324
SSD_efficientDet_D1	0.332	0.640	0.340
SSD_MobileNetV2 320 x320	0.112	0.374	0.032
SSD_MobilenetV2 640 640	0.294	0.570	0.275
YoloV5s	0.389	0.722	0.379
YoloV5l	0.420	0.738	0.451
YoloV6s	0.363	0.676	0.369
YoloV6l	0.328	0.620	0.320
YoloV7	0.634	0.965	0.765
YoloV7x	**0.686**	**0.972**	**0.851**

Appendix B. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.wasman.2023.05.039.

References

Aral, R.A., Keskin, S.R., Kaya, M., Hacıömeroğlu, M., 2018. Classification of TrashNet Dataset Based on Deep Learning Models. 2018 IEEE International Conference on Big Data (Big Data).

Bilsen, V., Kretz, D., Padilla, P., M.V, A., J.V., O., Izdebska, O., Hansen, M.E., Bergmans, J., Szuppinger, P., 2018. Development and implementation of initiatives fostering investment and innovation in construction and demolition waste recycling infrastructure. European Commission.

Bochkovskiy, A., Wang, C.-Y., Liao, H.-Y.M., 2020. YOLOv4: Optimal Speed and Accuracy of Object Detection. arXiv:2004.10934.

Bosoc, S., Suciu, G., Scheianu, A., Petre, I., 2021. Real-time sorting system for the Construction and Demolition Waste materials. 13th International Conference on Electronics, Computers and Artificial Intelligence (ECAI).

Chen, X., Huang, H., Liu, Y., Li, J., Liu, M., 2022. Robot for automatic waste sorting on construction sites. Autom. Constr. 141.

Dai, J., Qi, H., Xiong, Y., Li, Y., Zhang, G., Hu, H., Wei, Y., 2017. Deformable Convolutional Networks. arXiv:1703.06211.

Davis, P., Aziz, F., Newaz, M.T., Sher, W., Simon, L., 2021. The classification of construction waste material using a deep convolutional neural network. Autom. Constr. 122.

Demetriou, D., Mavromatidis, P., Mwombeki, R., Papadopoulos, H., Petrou, M., Nicolaides, D., 2022. Construction and demolition waste object detection dataset. Mendeley Data.

Dimitriou, G., Savva, P., Petrou, M.F., 2018. Enhancing mechanical and durability properties of recycled aggregate concrete. Constr. Build. Mater. 158, 228–235.

Eliades, C., Lenc, L., Kral, P., Papadopoulos, H., 2019. Aytomatic face recognition with well-calibrated confidence measures. Mach. Learn. 108.

García, M.C., Mateo, J.T., Benítez, P.L., Gutiérrez, J.G., 2021. On the performance of one-stage and two-stage object detectors in autonomous vehicles using camera data. Remote Sens. (Basel) 13.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778.

Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, Adam, a.H., 2017. MobileNets: Efficient convolutional neural networks for mobile vision applications. arXiv:1704.04861.

Huang, J., Rathod, V., Sun, C., 2020. Tensorflow Object Detection API. 2020.

Huang, G., Liu, Z., Maaten, L.V.D., Weinberger, K.Q., 2017. Densely connected convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4700–4708.

Iandola, F.N., Han, S., Moskewicz, M.W., Ashraf, K., Dally, W.J., Keutzer, K., 2016. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size. arXiv:1602.07360.

Jocher, G., 2021. ultralytics/yolov5: v6.0 - YOLOv5n 'Nano' models, Roboflow integration, TensorFlow export, OpenCV DNN support.

Ku, Y., Yang, J., Fang, H., Xiao, W., Zhuang, J., 2021. Deep learning of grasping detection for a robot used in sorting construction and demolition waste. J. Mater. Cycles Waste Manage. 23, 84–95.

Li, J., Fang, H., Fan, L., Yang, J., Ji, T., Chen, Q., 2022b. RGB-D fusion models for construction and demolition waste detection. Waste Manage. 139.

Li, C., Li, L., Jiang, H., Weng, K., Geng, Y., Li, L., Ke, Z., Li, Q., Cheng, M., Nie, W., Li, Y., Zhang, B., Liang, Y., Zhou, L., Xu, X., Chu, X., Wei, X., Wei, X., 2022. YOLOv6: A Single-Stage Object Detection Framework for Industrial Applications. arXiv: 2209.02976.

Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S., 2017. Feature pyramid networks for object detection. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2117–2125.

Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P., 2017b. Focal Loss for Dense Object Detection. arXiv:1708.02002.

Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C., 2014. Microsoft COCO: Common objects incontext, ECCV 2014. Zurich, Switzerland, pp. 740–755.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., 2016. SSD: Single Shot MultiBox Detector. arXiv:1512.02325.

Liu, S., Qi, L., Qin, H., Shi, J., Jia, J., 2018. Path aggregation network for instance segmentation. IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 8759–8768.

Luhar, S., Nicolaides, D., Luhar, I., 2021. Fire resistance behaviour of geopolymer concrete: an overview. Buildings 11.

Lukka, T.J., Tossavainen, T., Kujala, V.J., Raiko, T., 2014. ZenRobotics Recycler – Robotic Sorting using Machine Learning. Helsinki, Finland.

Maltoudoglou, L., Paisios, A., Lenc, L., Martinek, J., Kral, P., Papadopoulos, H., 2022. Well-calibrated confidence measures for multi-label text classification with a large number of labels. Pattern Recogn. 122.

Mao, W.L., Chen, W.C., Wang, C.T., Lin, Y.H., 2021. Recycling waste classification using optimized convolutional neural network. Recour. Conserv. Recycl. 164.

Na, S., Heo, S., Han, S., Shin, Y., Lee, M., 2022. Development of an artificial intelligence model to recognise construction waste by applying image data augmentation and transfer learning. Buildings 12.

Oikonomopoulou, K., Savva, P., Ioannou, S., Nicolaides, D., Petrou, M.F., 2020. Production of Recycled Aggregate Concrete Using Construction and Demolition Waste, RILEM Spring Convention and Conference.

Oikonomopoulou, K., Ioannou, S., Savva, P., Spanou, M., Nicolaides, D., Petrou, M.F., 2022b. Effect of mechanically treated recycled aggregates on the long term mechanical properties and durability of concrete. Materials 15.

Papadopoulos, H., 2008. Inductive conformal prediction: theory and application to neural networks. Tools in Artificial Intelligence chap. 18, 315–330.

Redmon, J., Farhadi, A., 2018. YOLOv3: An Incremental Improvement. arXiv: 1804.02767.

Redmon, J., Divvala, S., Girshick, R., Farhadi, A., 2015. You only look once: unified. Real-Time Object Detec. arXiv:1506.02640.

Redmon, J., Farhadi, A., 2016. YOLO9000: Better, Faster. Stronger. arXiv:1612.08242.

Ren, S., He, K., Girshick, R., Sun, J., 2016. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. arXiv:1506.01497.

Robert, P., Giannopoulou, I., Savva, P., Sakkas, K., Petrou, M.F., Nicolaides, D., 2023. New eco-friendly inorganic polymeric materials for the passive fire protection of structures. To be Included in the Proceedings of TMS 2023 152th Annual Meeting, San Diego, California, USA.

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.C., 2019. MobileNetV2: Inverted Residuals and Linear Bottlenecks. arXiv:1801.04381.

Sarc, R., Curtis, A., Kandlbauer, L., Khodier, K., Lorber, K.E., Pomberger, R., 2019. Digitalisation and intelligent robotics in value chain of circular economy oriented waste management – a review. Waste Manag. 95.

Savva, P., Ioannou, S., Oikonomopoulou, K., Nicolaides, D., Petrou, M.F., 2021. A mechanical treatment method for recycled aggregates and its effect on recycled aggregate-based concrete. Materials 14.

Simonyan, K., Zisserman, A., 2014. Very deep convolutional networks for large-scale image recognition. arXiv:1409.1556.

Tan, M., Pang, R., Le, Q.V., 2020. EfficientDet: Scalable and Efficient Object Detection. arXiv:1911.09070.

Tian, Z., Shen, C., Chen, H., He, T., 2020. FCOS: A Simple and Strong Anchor-free Oject Detector. arXiv:2006.09214v3.

Valanides, M., Robert, P., Giannopoulou, I., Oikonomopoulou, K., Savva, P., Nicolaides, D., 2023. Sustainable Materials for Energy Improvement and Fire Protection of Buildings. In: To be Included in the Proceedings of ICOCE 2023, 7th International Conference on Civil Engineering, Singapore.

Wang, C., Bochkovskiy, A., Liao, H.M., 2022. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. arXiv:2207.02696.

Wang, Z., Li, H., Zhang, X., 2019. Construction waste recycling robot for nails and screws: computer vision technology and neural network approach. Autom. Constr. 97, 220–228.

Wang, Z., Li, H., Yang, X., 2020. Vision-based robotic system for on-site construction and demolition waste sorting and recycling. J. Build. Eng. 322.

Wu, Y., Kirillov, A., Mass, F., Lo, W.A., Girshick, R., 2019. Detectron2.

Xiao, W., Yang, J., Fang, H., Zhuang, J., Ku, Y., Zhang, X., 2020. Development of an automatic sorting robot for construction and demolition waste. Clean Techn. Environ. Policy 22, 1829–1841.

Yang, M., Thung, G., 2016. Classification of trash for recyclability status. CS229 Project Report.

Yu, B., Wang, J., Li, J., Lu, W., LI, C.Z., Xu, X, 2020. Quantifying the potential of recycling demolition waste generated from urban renewal: a case study in Shenzhen, China. J. Clean. Product. 247.

Zhang, X., Zhou, X., Lin, M., Sun, J., 2018. ShuffleNet: An extremely efficient convolutional neural network for mobile devices. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6848–6856.