Object Detection in Hospital Facilities: A Comprehensive Dataset and Performance Evaluation

Da Hu[1], Shuai Li[2,*], Mengjun Wang[2]

[1] Department of Civil and Environmental Engineering, Kennesaw State University, Marietta, GA, 30060

[2] Department of Civil and Environmental Engineering, The University of Tennessee, Knoxville, TN 37996

* Corresponding Author, Shuai Li, email: sli48@utk.edu

Abstract

Detecting objects in hospital indoor environments is critical for scene understanding and can have various applications in healthcare. Deep learning (DL) algorithms have proven to be effective in object recognition from images or videos, but the availability of annotated datasets plays a crucial role in their successful application. However, there is a shortage of datasets for object detection in hospital settings, hindering the advancement of hospital indoor object detection algorithms. In this paper, we present the Hospital Indoor Object Detection (HIOD) dataset, consisting of 4,417 images covering 56 object categories. The HIOD dataset represents the frequently encountered objects in hospitals and comprises 51,809 annotated objects. The dataset is characterized by dense annotation, with an average of 11.7 objects and 6.8 object categories per image. An object detection benchmark was established using the HIOD dataset and eight state-of-the-art object detectors. The benchmark provides a comprehensive evaluation of the performance of the selected object detectors on a large and diverse set of images of objects commonly seen in hospital environments. The results of the benchmark can be used to compare and analyze the performance of different object detectors and identify their strengths and weaknesses for use in hospital environments. In the benchmark, one-stage detectors have shown superior performance compared to two-stage detectors of similar parameter sizes. In particular, YOLOv6-L was able to attain a mean average precision (mAP) of 51.7% while operating at a detection speed of 255 FPS. The benchmark and dataset can serve as a valuable resource for researchers and practitioners in the field of computer vision and robotics, helping to advance the development of more effective and efficient object detection algorithms for developing automated operations in hospitals such as robotic disinfection and patient assistance.

Keywords: object detection; hospital; deep learning; image dataset

1. Introduction

In the United States, more than 6,000 hospitals handle over 30 million hospital admissions per year [1]. According to data from the Bureau of Labor Statistics (BLS), 23% of hospitals are suffering from severe staffing shortages as of February 2022. In particular, the unfilled nursing positions alone will be more than 200,000 through 2029 [2], significantly affecting the daily operations in hospitals. The utilization of robots in healthcare services has gained significant attention as a potential solution to alleviate the labor shortage issues. By performing critical tasks such as disinfection, telepresence, and medical supply delivery, robots have the ability to increase efficiency and productivity in the hospital setting [3]. The global market for healthcare service robots was valued at $6.5 billion in 2021 and is projected to grow to $15.8 billion by 2028 [4]. The capability to detect objects in images or videos is crucial for healthcare service robots to comprehend the surrounding environments. This is particularly important for various healthcare applications, such as indoor navigation systems for patients with visual impairment or blindness. These systems must accurately identify objects in the environment to provide accurate and targeted guidance [5]. Additionally, for robotic disinfection, recognizing high-touch objects is crucial for a thorough and efficient cleaning process, as it allows for the identification of surfaces with a high risk of infection transmission [6,7]. However, current research in this area is limited in its ability to effectively detect and categorize a diverse range of indoor objects in hospital environments.

With the advancement of computing capability, deep learning-based (DL-based) algorithms have achieved superior performance in understanding the semantic meaning of an image. Many recent studies [8–10] have shown the capability of DL-based algorithms in locating and classifying object instances in images. A high-quality object detection dataset is essential for training accurate and robust models. However, there remains a gap in the availability of an image dataset that is comprehensive, diverse, and reflective of the task of identifying and locating objects within an indoor hospital environment. The absence of a comprehensive image dataset specifically designed for object detection in hospital environments is a result of two main challenges. Firstly, obtaining access to high-quality hospital indoor images is restricted, and images that capture medical equipment, such as ventilators and incubators, are not as easily accessible as other common objects, such as chairs, tables, and computers. Secondly, annotating these images to a level of accuracy that meets the standards for machine learning models can be a complex and demanding task, due to the cluttered nature of hospital environments, including intensive care units, operating rooms, and patient wards. The presence of numerous furniture and medical equipment in these spaces makes it difficult to accurately identify and localize objects in images, which is critical for the effective training and evaluation of object detection models.

To overcome the obstacles in creating a comprehensive dataset for object detection in hospitals, this paper presents a new and innovative solution. The construction of the hospital indoor object detection (HIOD) dataset underwent a four-step process including the selection of object categories, collection of images, selection of images, and image annotation. The result of these efforts is a dataset with 4,417 annotated images spanning 56 object categories and featuring 51,809 annotated object instances. To validate the effectiveness of the HIOD dataset, a benchmark was established using eight existing state-of-the-art object detectors. The results of the benchmark demonstrate that the network trained on the HIOD dataset can localize and classify objects in a hospital environment with satisfactory accuracy. The authors believe that the creation of the HIOD dataset and benchmark will serve as a valuable resource for researchers and practitioners in the development of computer vision-based applications in hospitals.

2. Literature review

2.1 Related studies on object detection dataset

Object detection datasets typically have a considerable number of images, and objects are annotated with bounding boxes in each image. In the past decade, deep learning algorithms have become de facto approaches to identifying objects from images or videos. The successful application of deep learning methods largely benefits from the availability of annotated image datasets. A number of popular datasets and benchmarks have been developed, driving the advancement of DL-based algorithms. For example, the Pascal Visual Object Classes (VOC) dataset is one of the well-known image datasets for object detection in the early development of DL-based algorithms. The first version of the dataset was published in 2007, which is known as VOC2007 [11]. The dataset consists of 5,000 images covering 20 object categories with a total of 12,000 annotated instances. The latest version of Pascal VOC was published in 2012 (known as VOC2012). This version increases the image size to 11,530 with 27,450 annotated object instances. Thereafter, Microsoft released Common Objects in Context (COCO) [12] in 2014 with a significant improvement, concerning the number of object categories, the number of images, and the number of annotated instances. Specifically, COCO contains 164,000 images covering 80 object categories with 897,000 object instances. OpenImages is another large image dataset, which was initially released in 2018. OpenImages has undergone a series of iterative updates, and the current version is named OpenImages-v6 [13]. OpenImages-v6 consists of 1,910,098 images that are distributed across 600 classes with 15,851,536 annotated object instances. In addition to these datasets, there are several other image-based object detection datasets, such as Objects365 [14]. These datasets consist of both indoor and outdoor environments and have been widely used as object detection benchmarks.

In the context of hospitals, there are very few image datasets specifically developed for object detection. In recent years, Bashiri et al. [15] proposed an object classification dataset named MCIndoor20000 using images collected from the Marshfield Clinic. The dataset contains a total of 2,055 images with only three object categories: doors, stairs, and hospital signs. More recently, Issmail et al. [16] created an image classification dataset (MYNursingHome) using a total of 37,500 images collected in several nursing homes. The dataset contains 25 object categories, such as cabinet, call bell, and television. However, MCIndoor20000 and MYNursingHome datasets are developed for image classification with an object class associated with each image. Object detection is more complex and computationally intensive because it requires not only recognizing the object, but also localizing it in the image. the limited object categories in these datasets, such as the absence of commonly seen objects like ventilator and door handle, limits their usefulness for real-world object detection tasks. To overcome these limitations, this study introduces a new dataset, which includes a total of 56 object categories and is specifically designed for object detection in hospital indoor environments.

2.2 Related studies on object detection algorithms

Object detection algorithms are developed to detect and identify objects (e.g., chair, table, and human) within an image. Object detection is a combination of image classification and object localization. Object detection algorithms can be characterized as traditional and DL-based algorithms. Fig. 1 shows the evolution of object detection algorithms. Before 2013, traditional object detectors were primarily used to detect objects in digital images. Since that, DL-based algorithms have dominated the research on object detection given their superior performance.

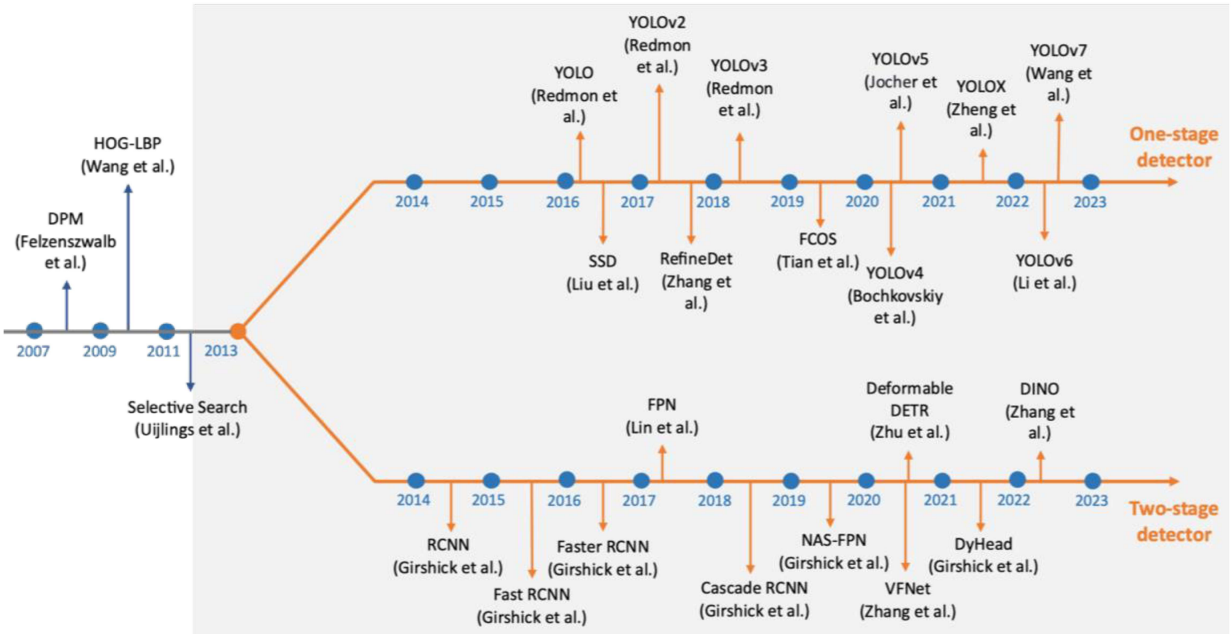

Fig. 1. Progression of object detection techniques. This includes traditional approaches [18–20] and Deep Learning (DL)-based algorithms. The DL-based algorithms can be further categorized into one-stage detectors [8,9,21–29] and two-stage detectors [30–39].

The traditional object detectors can be categorized into three steps: informative region selection, feature extraction, and classification. The informative region selection stage aims to identify candidate regions of objects using sliding windows. This process is computationally expensive and could generate a large number of candidate regions. The feature extraction stage focuses on extracting visual features for each candidate window for classification. The representative feature extraction methods include Scale-Invariant Feature Transform (SIFT), Histogram of Oriented Gradients (HOG), and Haar-like features. Finally, the feature classifier is trained to classify extracted features that are associated with the object in the region. Some well-known classifiers such as Support Vector Machine (SVM), AdaBoost, and Deformable Part-based Model (DPM) have been used and achieved fair performance on object classification. However, traditional approaches are computationally expensive and produce many redundant detections, which are inefficient and inaccurate in localizing objects. Furthermore, the performance is largely dependent on the selection of feature extraction methods that are unable to extract complex features in an image.

A large and growing body of literature has focused on the development of deep learning networks for the task of object detection. The DL-based approaches have the capability to automatically learn more complex image features compared to traditional approaches. The DL-based methods can be categorized into one-stage and two-stage object detection architectures. Specifically, the two-stage architecture separates the object localization task from the object classification task, which generates the region proposal first followed by the region classification. The Region-based Convolutional Neural Network (R-CNN) is the pioneering work for the two-stage detector [39]. R-CNN first generates 2000 region proposals based on the selective search algorithm. The 4096-dimensional features are then extracted for each region proposal using the deep learning network. Lastly, the features extracted are fed into a trained SVM classifier, and the bounding box regression is used to fine-tune the object position. As the pioneering work, R-CNN is very slow in region proposal generation, leading to slow inference speed. In addition, the feature extraction and SVM classifier are trained separately. To address these shortages, Fast R-CNN [38] and Faster R-CNN [37] were proposed in the following two years after the release of R-CNN. In recent years, the development of new algorithms such as Transformer network has significantly increased the performance of object detection. For example, with a Swin Transformer backbone, DETR with Improved denoising anchor boxes (DINO) achieved state-of-the-art performance on COCO dataset [30].

While these two-stage architectures achieve promising results, the inference speed is rather slow, which makes them unsuitable for real-time applications such as autonomous indoor robots. Compared to two-stage architecture, one-stage architecture directly predicts the bounding box over the images without the region proposal step which allows a faster inference speed. You Only Look Once (YOLO) is the most popular one-stage architecture, which provides state-of-the-art performance for real-time object detection. YOLO detector was introduced by Redmon et al. in 2015 [28], which is a unified and real-time detection approach. YOLO architecture is still under active development by other researchers, such as Jocher et al. [29] and Wang et al. [8]. The latest version is YOLOv7 with state-of-the-art real-time performance on COCO dataset. In addition, there are some other one-stage object detection algorithms, such as Single Shot MultiBox Detector (SSD), Precise Single-stage Detector (PSSD) [40], CornerNet, RetinaNet. The DL-based algorithms have achieved promising performance for object detection.

However, one of the main challenges with deep learning is that it often requires large amounts of data to train models effectively. To address the limited data availability, transfer learning has been demonstrated to be effective in a variety of tasks and across various domains [41]. Transfer learning is a technique where a model trained on one task is used as the starting point for a model on a second, related task. This approach has been widely studied in recent years and has been shown to be effective in a variety of settings, including computer vision, natural language processing, and speech recognition. Some examples of transfer learning include using a pre-trained model to classify images [6], fine-tuning a pre-trained model for a specific NLP task [42], and using a pre-trained model as a feature extractor for a different task [43]. Research in this area has shown that transfer learning can improve performance and reduce the amount of data and computation required for training a new model [41]. In this study, an algorithm benchmark will be established with the newly introduced hospital indoor object detection dataset. Pre-trained models on COCO dataset are used to fine-tune object detectors for better performance in the new dataset.

3. Dataset preparation

The preparation of the hospital indoor object detection (HIOD) dataset consists of four steps: object category selection, image collection, image selection, and image annotation (see Fig. 2). The details of each step are elaborated below.

Fig. 2. Flowchart of dataset preparation

3.1 Object category selection

In hospital settings, there is an abundance of furnishings and apparatus aimed at serving both patients and healthcare professionals. It is of utmost importance for a robot to be able to perceive and recognize these elements in order to comprehend the surrounding environment and make informed plans accordingly. The development of a diverse and comprehensive dataset is essential for enabling robots to accurately perceive their environment in hospital settings. To accomplish this, our dataset encompasses a broad range of objects commonly found in hospitals, such as ventilators, chairs, tables, and sofas, with a total of 56 object categories annotated. Furthermore, the dataset considers the importance of human context by annotating various types of individuals present in the hospital, including healthcare workers, patients, and visitors. If the type of person cannot be determined from the image, they are annotated as a separate human category.

3.2 Image collection

Image data were collected from online image search engines, including Google Images, Bing Images, Getty, and Shutterstock. Video data was also collected from online video-sharing websites like YouTube. The images and videos were obtained using a list of keywords related to hospital indoor environments, such as “intensive care unit,” “operating room,” “hospital consulting room,” “hospital tour,” “hospital waiting room,” and more. In total, 18,234 images were collected from the online image search engine in the context of hospitals. Furthermore, 59 videos were collected from video-sharing websites. The video data was initially transformed into separate frames, and a single frame was selected from a minimum of 30 consecutive frames to attain a dataset with more visual diversity. This process converts videos into 24,963 images. The collected images cover a variety of indoor environments within the hospitals. Fig. 3 shows some examples of hospital indoor scenes in the dataset.

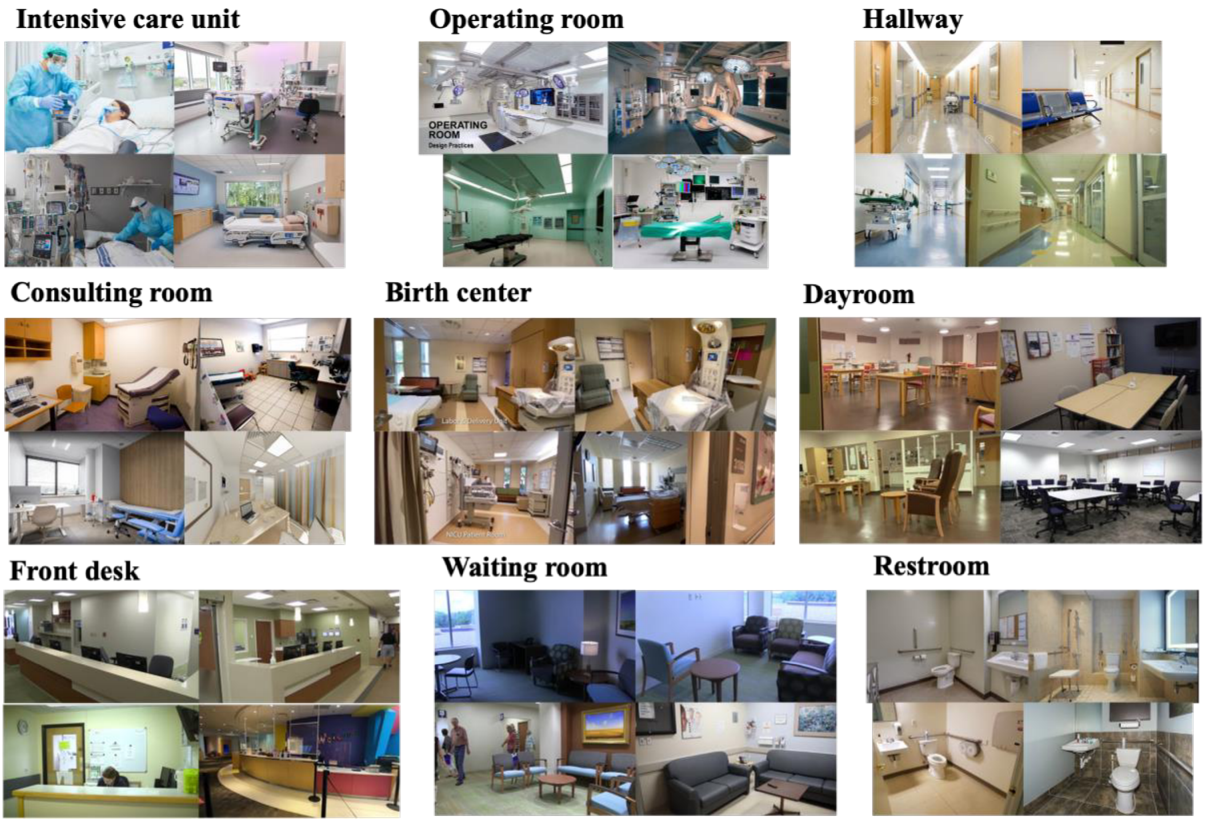

Fig. 3. Example indoor scenes in hospitals

3.3 Image selection

Following the image collection phase, a total of over 40,000 images were accumulated within the hospital environment. To maintain the quality of the dataset, duplicate images were removed, and a thorough cleaning process was performed on the collected images. The following steps outline the procedures involved in these processes.

Duplicates removal. As some images in the dataset were obtained from video data, it is inevitable that the dataset might contain images from similar scenes. These similar images, also referred to as near-duplicate and duplicate images, give rise to two issues for the dataset. First, the near-duplicates and duplicates introduce bias in the dataset, which could drive CNN to learn the pattern. Second, the generalizability of the trained network on unknown images could be compromised. Therefore. In the study, the near-duplicate and duplicate images are removed, in order to ensure the diversity of the dataset, and the generalizability of the network trained on the dataset.

The duplicate detection consists of two steps. Initially, image encodings are generated utilizing the perceptual hashing method as delineated in reference [44]. This technique is specifically designed to remain relatively invariant in response to minor discrepancies between images, such as

compression and brightness alterations, rendering it suitable for the task of duplicate elimination. The perceptual hashing method generates a 16-character hexadecimal string hash corresponding to each image within the dataset. Subsequently, the Hamming distance is computed between pairs of hashes. The Hamming distance, an integer ranging from 0 to 64, characterizes the similarity between two images, with a smaller value denoting a higher degree of resemblance. In the present study, the Hamming distance threshold is established at 12.

Image cleaning. The image cleaning process is necessary to remove low-quality images, such as blurry and low-resolution images. Superficially, low-resolution images in this study represent images with a shorter side smaller than 400, which are first removed from the dataset. The resolution criteria are set to ensure the performance of the deep learning network, especially on small-size objects like handles. After the initial filtering process, two students were recruited from the University of Tennessee, Knoxville to further clean the image dataset. This manual process primarily concentrated on eliminating images that were blurry or devoid of any objects in the scene. After completing the duplicate removal and image cleaning procedures, the resulting dataset comprised 4,417 images.

3.4 Image annotation

Image bounding box annotation was done by crowdsourcing the task to human labelers on the Scale AI platform. The crowdsourcing annotation can be summarized into three steps.

1) The bounding box annotation instruction for 56 object categories is created and posted on the Scale AI platform with a definition and example annotations. The instruction provides the objective of the task, objects to be labeled, and representative examples to the Scale AI human labeler. The labeler must review and understand the instruction to get familiar with the dataset and annotation task.
2) The second step aims to fine-tune the instruction to make it easy to understand for labelers. 30 images are selected as representative samples of the overall dataset, which covers a variety of indoor scenes in hospitals. These 30 images were published on the Scale AI platform and the labelers will annotate them and provide feedback on the instruction. The authors then review and audit the annotation. The overall calibration score will be given after finishing the audit, which is an indicator of the annotation quality. Per the feedback and calibration score, the instruction will be updated to resolve the confusion. This step takes several iterations to ensure the instruction readability and clarity.
3) After refining the instruction, training and evaluation tasks are created to ensure the quality of labels. Specifically, the 30 representative images are annotated with ground truth. 10 images are used for the training task and 20 images for the evaluation task. Labelers will complete

before attempting to label images in the production batch. These tasks make up the training course that all taskers must complete with a certain quality threshold in order to onboard onto your project. The evaluation task is used to track the quality of the human labelers after they start the annotation task. The labelers who cannot meet the quality threshold will be taken off the project.

The Scale AI annotations are served as our draft version of the dataset, which was audited by an auditing team, in order to ensure the quality of the dataset. The auditing team consists of six undergraduate and graduate students recruited from the University of Tennessee, Knoxville. The auditing team was given comprehensive training before they start to audit the dataset. The training will enable the team to get familiar with the task and objects to be annotated in the image. The team is divided into four inspectors and two examiners. The inspector is responsible to inspect all the annotated images in the draft version of the dataset. The inspector needs to correct falsely labeled images. The average inspecting speed is around 40 images per hour according to the report from the four inspectors. The image annotation quality is significantly improved after this process. The work of the examiner is to conduct a final examination of the dataset to ensure the quality of our dataset. The examiner requires to make the correction and refine the object bounding box in the image.

3.5 Data statistics

The annotated image dataset is named as Hospital Indoor Object Detection (HIOD) dataset. The HIOD consists of 4,417 images with 51,809 annotated object instances. Fig. 4 shows the number of objects and number of images for each type of indoor object in the HIOD dataset. The number of instances per object category presents a long-tailed distribution. In particular, Handle (5,951 objects), chair (4,777 objects), and dispenser (2,819 objects) are recognized as the most frequent indoor objects in HIOD. Whereas press-to-open button (53 objects) and Xray table (57 objects) have the least number of instances in HIOD. The number of images for each object category also exhibits a long-tailed distribution. Specifically, a total of 1,812 images contains chair object, however, only 48 images contain press-to-open button. The long-tailed distribution remains a challenging problem for the task of object detection, which could affect the object detectors' performance.

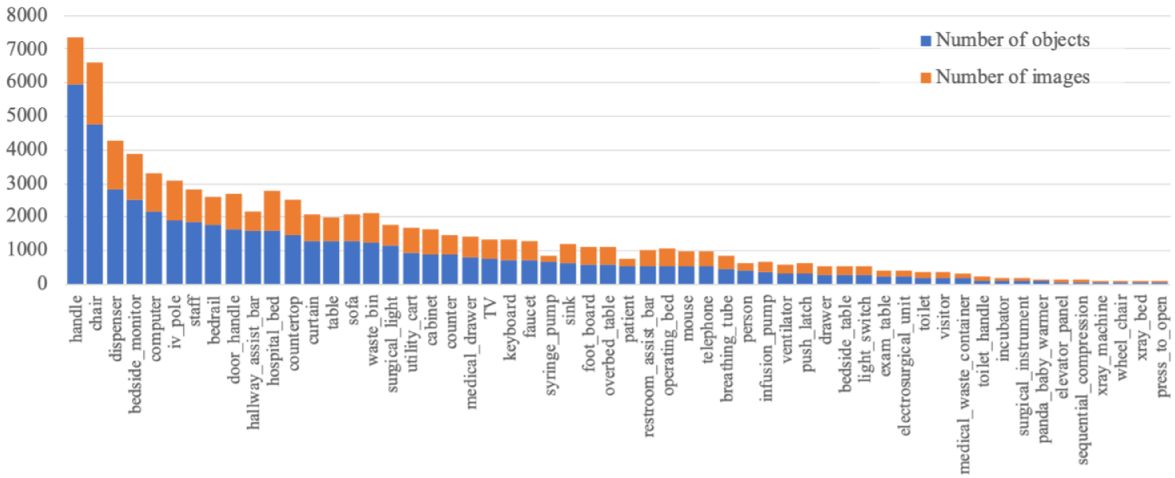

Fig. 4. Number of objects and number of images for each type of indoor object in the HIOD dataset.

Fig. 5 presents the distribution of bounding boxes over image size in the HIOD dataset. The statistic indicates that around 50% of objects occupy a pixel area representing less than 1% of the entire image. This is because the HIOD dataset contains many small objects, such as handles, dispensers, and door handles, which occupy only a small portion of an image. In addition, around 10% of objects occupy a pixel area greater than 10% of an image.

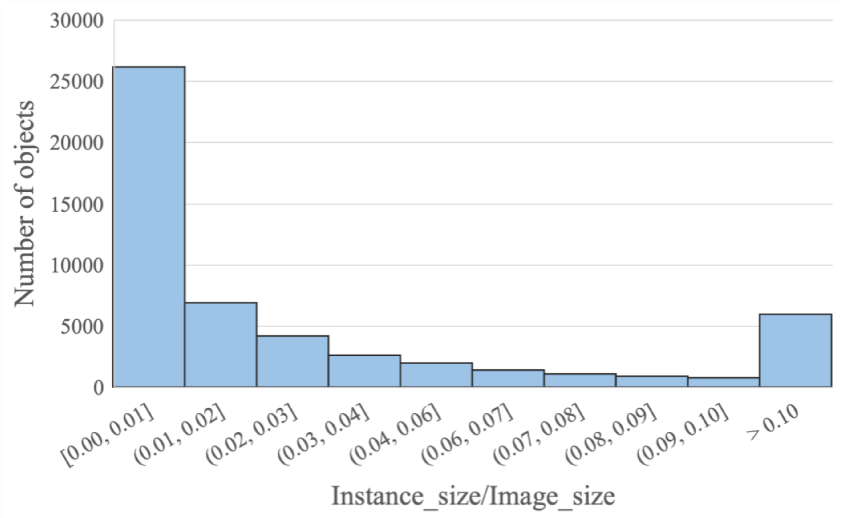

Fig. 5 Distribution of the bounding box size in the HIOD dataset

The number of objects and categories per image in the HIOD dataset is further analyzed and compared to other typical object detection benchmarks. Fig. 6 shows the statistics and comparison with COCO, VOC, and OpenImages. The HIOD is found to be denser and more diverse than COCO, VOC, and OpenImages with regard to the number of objects and categories on an image. Quantitatively, our HIOD dataset has an average and median of 11.7 and 10 objects on an image, respectively. In comparison, the average number of objects per image for COCO, VOC, and OpenImages are 7.3, 8.2, and 2.7, respectively. The median number of objects per image for

COCO, VOC, and OpenImages are 4, 2, and 4, respectively. On the other hand, our HIOD dataset contains an average and median of 6.8 and 6 object categories per image, respectively, which are both significantly greater than other benchmarks. The abundant instance density and broad category diversity in the HIOD dataset lay a solid foundation for building a robust object detector in hospital settings.

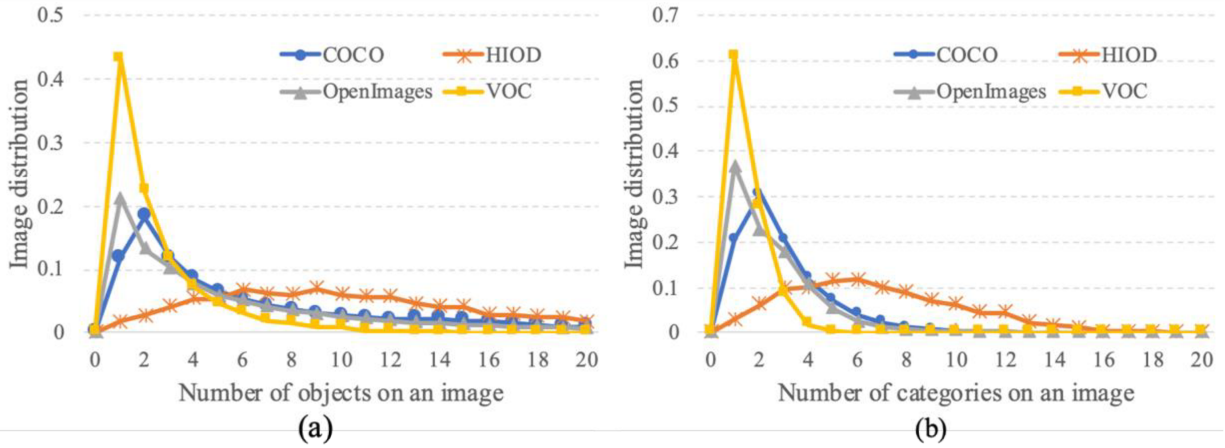

Fig. 6. Comparison of the number of objects and categories per image with COCO, VOC, and OpenImages. (a) number of objects; and (b) number of categories

4. Object detection algorithm

4.1 Algorithm selection

In this study, five one-stage and five two-stage object detection algorithms are selected and evaluated on the HIOD dataset (shown in Table 1). For one-stage algorithms, the YOLO series is selected including YOLOv5-L, YOLOX-L, YOLOv6-L, and YOLOv7. For two-stage algorithms, Faster R-CNN, Deformable DETR, VFNet, and DyHead are selected. Note that the performance of these algorithms is largely dependent on the selection of backbone. Generally, the larger the backbone, the better the performance. On the other hand, a large network could result in a slow inference speed. The selection of the backbone network is a trade-off between accuracy and speed. For a fair comparison between different detectors, the number of network parameters ranges from 33.6 to 65.3 million. All of the one-stage algorithms achieve a fast inference speed with an FPS over 94. One interesting point to mention is that state-of-the-art one-stage algorithms such as YOLOv6 and YOLOv7 outperform two-stage algorithms by a large margin with similar sizes of object detection networks. The inference speed is also much higher for one-stage algorithms compared to two-stage algorithms. In addition, the image size for the two-stage algorithms is greater than for one-stage algorithms. This is because the development of two-stage algorithms is mainly focused on prediction accuracy instead of speed. The performance of two-stage algorithms could be much better with a larger backbone network. For example, the mean average precision (mAP) of DyHead increases to

58.4% from 43.3%, if the backbone changes from ResNet50 to Swin-L [45]. However, the number of parameters increases to 210.4 million from 38.8 million, which is not applicable to be deployed in an embedded system for onboard detection.

Table 1 Object detection algorithms

Algorithm	Backbone	#Param. (M)	Image size	FPS	mAP[a]
One-stage[b]					
YOLOv5-L [29]	Modified CSPNet	46.5	640 × 640	113	49.0
YOLOX-L [21]	Modified CSPNet	54.2	640 × 640	94	49.7
YOLOv6-L [9]	CSPStackRep	58.5	640 × 640	98	52.5
YOLOv7 [8]	E-ELAN	36.9	640 × 640	110	51.2
Two-stage[c]					
Faster R-CNN [37]	ResNet50+FPN	41.4	1333 × 800	21[d]	40.3
Deformable DETR [33]	ResNet50	40.9	1333 × 800	19	46.8
VFNet [32]	ResNet50+FPN	33.6	1333 × 800	19	47.8
DyHead [31]	ATSS+ResNet50+FPN	38.8	1333 × 800	14[d]	43.3

[a] The reported mAP is evaluated on COCO2017 val; [b] The FPS and mAP for one-stage algorithms refer to [9]; [c] The mAP for two-stage algorithms refers to mmdetection benchmark [45]. [d] The FPS refers to mmdetection benchmark [45].

4.2 Implementation

The network is trained on a workstation running Ubuntu 16.04 with dual Intel Xeon Silver 4114 CPUs, 128 GB RAM, and an NVIDIA RTX A6000 GPU. To optimize time and resources, transfer learning techniques are employed for network training. Pretrained weights from the COCO dataset serve as the foundation for all networks. Table 2 outlines the image size, batch size, number of epochs, initial learning rate, and learning rate schedule. Default values are assigned to other hyperparameters. The HIOD dataset is randomly partitioned into a training set (70%), a validation set (10%), and a testing set (20%). The best performance achieved on the validation set is employed for evaluation on the testing set. The benchmark performance and subsequent analysis are grounded in the results obtained from the testing set.

Table 2 Training configurations and default values employed for additional hyperparameters

Algorithm	Image size	Batch size	#Epoch	Initial lr	lr schedule
One-stage					
YOLOv5-L	640 × 640	32	300	0.01	Cosine decay
YOLOX-L	640 × 640	32	300	0.005	Cosine decay
YOLOv6-L	640 × 640	32	300	0.01	Cosine decay
YOLOv7	640 × 640	32	300	0.01	Cosine decay
Two-stage					
Faster R-CNN	1333 × 800	24	24	0.02	[16,22]*
Deformable DETR	1333 × 800	6	50	0.0002	[40]*
VFNet	1333 × 800	16	24	0.01	[16,22]*
DyHead	1333 × 800	16	24	0.01	[16,22]*

* Learning rate decay by a factor of 10 at the specified epoch.

4.3 Evaluation metrics

The COCO Detection Challenge's mAP metric is viewed as the standard metric to evaluate the performance of object detection, which is used to evaluate the object detectors' performance on the HIOD dataset. The task of object detection consists of object localization and object classification. The intersection over union (IoU) is used to evaluate object localization by measuring the degree of overlap between ground-truth and predicted bounding boxes. If the IoU is greater than 0.5, the prediction can be considered valid. Each bounding box is associated with an object category and a confidence score. Detection results with confidence scores below a specified threshold are considered invalid and disregarded in the analysis. The detection is True Positive (TP) only if the bounding box is valid and the object category is correct. The detection is False Positive (FP) if either or both of the two conditions cannot meet. False Negative (FN) represents an object that is not detected by the detector.

The mAP metric stands for the mean average precision. The definition of average precision (AP) is the area under the precision-recall curve. The precision and recall are calculated in Eq (1) and Eq. (2). The precision and recall are very sensitive to the confidence threshold. In particular, a high confidence threshold could result in a high precision score but a low recall score. AP score is used to remove the dependency on selecting a specific confidence threshold.

$$\text{precision} = \frac{TP}{TP+FP} \tag{1}$$

$$\text{recall} = \frac{TP}{TP+FN} \tag{2}$$

AP is defined in Eq. (3), which summarizes the precision-recall curve to one scalar value. AP is calculated as the average precision over 101 recall levels from 0 to 1. P_r represents precision at recall level r.

$$AP = \frac{1}{101}\sum_{r\epsilon\{0,0.01,\ldots,1\}} P_r \quad (3)$$

IoU threshold is another important parameter, which has a significant impact on the AP score. A higher IoU threshold results in a smaller AP score. According to COCO detection competition, AP@[0.5:0.95] for each object category over 10 IoU levels from 0.5 to 0.95 is used to address this issue, and it is defined in Eq. (4) The mAP is the average over all the object categories.

$$AP@[0.5{:}\ 0.95] = \frac{1}{10}\sum_{r\epsilon\{0.5,0.55,\ldots,0.95\}} AP_{IoU} \quad (4)$$

In addition to mAP, other AP-variant metrics are also used to evaluate the performance of detectors. These metrics are summarized in Table 3.

Table 3 Evaluation metrics

Metric	Description
AP_{50}	AP at IoU =0.50
AP_{75}	AP at IoU =0.75
AP_{small}	mAP for small objects (area of object < 32^2 pixels)
AP_{medium}	mAP for medium objects (32^2 pixels < area of object < 96^2 pixels)
AP_{large}	mAP for large objects (96^2 pixels < area of object)

4.4 Benchmark

The performance of the eight algorithms on the testing dataset is detailed in Table 4. All algorithms achieve an AP_{50} score above 64%, underscoring the trained network's proficiency in detecting objects in hospitals using the HIOD dataset. Notably, the one-stage algorithms exhibit superior performance compared to their two-stage counterparts. In particular, all four one-stage algorithms manage to achieve an impressive AP_{75} score exceeding 50%. Among them, YOLOv6-L stands out with the best performance, achieving an mAP of 51.7%, while YOLOv7 follows closely behind at 50.6%. In contrast, among the two-stage algorithms, VFNet takes the lead with an mAP of 49.5%, trailed by Deformable DETR at 49.0%. The relatively lower performance of two-stage algorithms can be primarily ascribed to their smaller backbone size. Additionally, it is crucial to note that network performance is contingent upon object size. The AP score tends to improve as object size increases, which can be attributed to the fact that smaller objects generally possess a limited number of features and provide less information for the network to extract and learn from. Consequently, as the network encounters larger objects with more distinguishable features, its ability to detect and classify them accurately is enhanced, leading to better overall performance.

Table 4 Benchmark performance on the testing dataset

Algorithm	mAP	AP_{50}	AP_{75}	AP_{small}	AP_{medium}	AP_{large}
One-stage						
YOLOv5-L	0.473	0.696	0.501	0.193	0.402	0.520
YOLOX-L	0.484	0.708	0.520	0.178	0.407	0.554
YOLOv6-L	0.517	0.737	0.553	0.211	0.418	0.597
YOLOv7	0.506	0.738	0.545	0.201	0.432	0.546
Two-stage						
Faster R-CNN	0.403	0.646	0.438	0.141	0.327	0.466
Deformable DETR	0.490	0.741	0.529	0.209	0.406	0.565
VFNet	0.495	0.711	0.538	0.200	0.420	0.567
DyHead	0.430	0.645	0.468	0.145	0.345	0.507

The confusion matrices for YOLOv6-L and VFNet are shown in Fig. 7 In a confusion matrix, the diagonal elements represent the number of correctly classified samples for a particular class. The diagonal values in the confusion matrices for YOLOv6-L and VFNet indicate the recall, which is a metric that measures the fraction of positive instances that were correctly identified. The high recall values in the evaluation of the model indicate that it has a low rate of false negatives, which means that a significant proportion of positive instances have been accurately identified. The confusion matrix reflects a considerable performance discrepancy between the different object categories. For instance, the "Xray bed" and "Xray machine" categories obtain a recall of 100% using the YOLOv6-L network, while the "surgical instrument" category only achieves a recall of 37.5%. On the other hand, the VFNet network achieved the highest recall score of 86.7% for the "restroom assistant bar" category, and the lowest recall score was 27.8% for the " surgical instrument" category. To showcase the performance of the YOLOv6-L network, some detection results are presented in Figure 8.

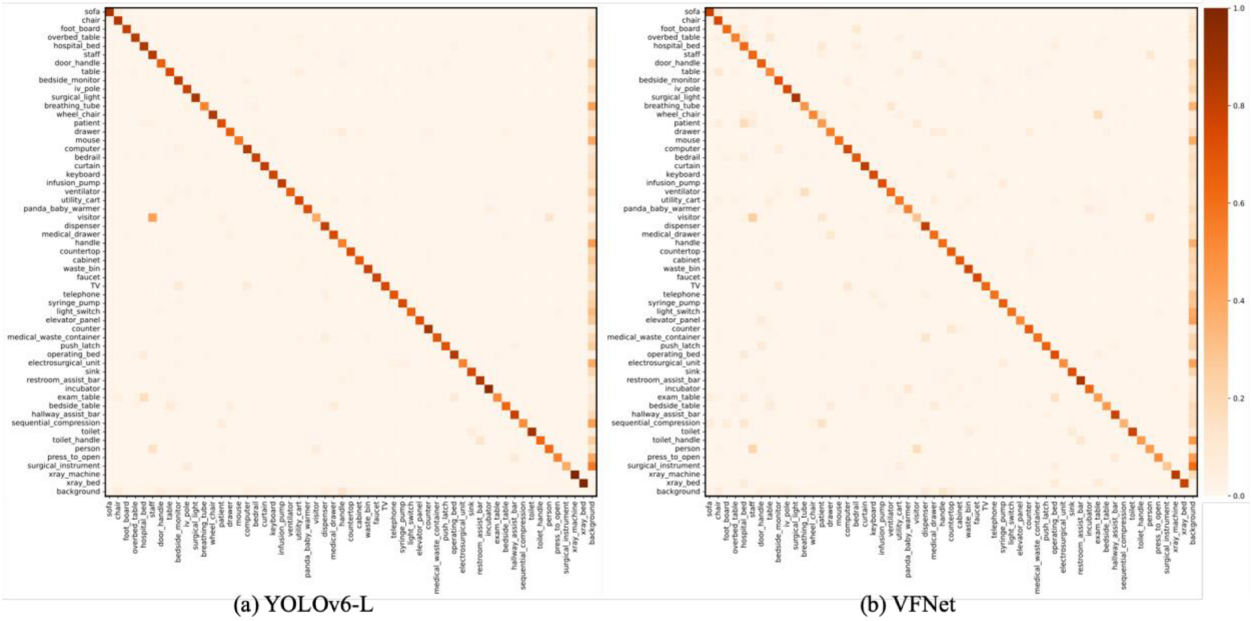

Fig. 7. Confusion matrix for YOLOv6-L and VFNet. Diagonal value represents recall for each object category

Fig. 8 Sample detection results using YOLOv6-L, displaying bounding boxes with associated category labels and confidence levels

The image resolution and network size have significant impacts on the performance of object detectors regarding both accuracy and speed. Given the superior performance of YOLOv6-L, the algorithm is selected to investigate the effect of image resolution and network size. Table 5 shows the performance of the detectors over different image resolutions from 416×416 to 1280×1280. The results indicate an increasing trend of detection accuracy over increasing image resolutions. This is because the HIOD dataset contains a lot of small objects such as handles and doorknob. A larger image resolution could be beneficial for the detection of these small objects [46]. As indicated, the AP_{small} has a larger performance increase with increasing image resolutions compared to AP_{medium} and AP_{large}. Overall, the network's performance experiences a substantial improvement when the image size increases from 416x416 to 640x640, resulting in a 4.1% rise in mAP. However, the enhancement becomes less pronounced as the image size progresses from 640x640 to 832x832, with only a 0.5% increment in mAP. Note that when the image size is further increased from 832x832 to 1280x1280, there is a slight 0.2% decrease in mAP. On the other hand, the object detector's inference speed decreases as the image resolution increases, as evaluated on NVIDIA RTX A6000 GPU. Since the speed is not stable with a batch size of 1, the batch size is set to 32 for speed evaluation. The testing results indicate that the inference speed decreases with increasing image resolutions. In specific, the inference speed reaches 561.1 FPS with an image size of 416×416, which is reduced to 65.9 FPS with an image size of 1280×1280. Therefore, the selection of image size in real-world applications is a tradeoff between speed and accuracy.

Table 5 Effect of image size on YOLOv6-L network

Image size	FPS (bs=32)	mAP	AP_{50}	AP_{75}	AP_{small}	AP_{medium}	AP_{large}
416×416	561.8	0.476	0.689	0.506	0.130	0.381	0.572
640×640	255.1	0.517	0.737	0.553	0.211	0.418	0.597
832×832	147.7	0.522	0.757	0.559	0.209	0.441	0.598
1280×1280	65.9	0.520	0.756	0.553	0.219	0.448	0.590

The size and backbone of the network are also influential factors affecting detectors' performance. Table 6 presents the performance comparison of other networks in the family of YOLOv6 and VFNet with different backbones. The performance of the YOLOv6 network is regulated by two factors: a depth-multiple and a width-multiple. The results show that the model's performance improves with an increase in the number of parameters. In particular, the YOLOv6-L model exhibits a 5.3% improvement in mean average precision (mAP) compared to the YOLOv6-S model. While YOLOv6 produces better performance, the number of parameters is more than three times that of YOLOv6-S. For VFNet, mAP has an improvement of 2.8% with the ResNeXt101-64x4d+FPN backbone compared to the network with the ResNet50+FPN backbone. The large network typically requires more storage and computation cost which hinders its deployment to mobile platforms such as

475 robots. The small network can achieve a greater detection rate, so as to be integrated into an
476 embedded system for real-time detection.

477 Table 6 Effect of backbone on network performance

Network	#Params	FPS	mAP	AP_{50}	AP_{75}	AP_{small}	AP_{medium}	AP_{large}
YOLOv6-S	17.2	763.4*	0.464	0.680	0.499	0.140	0.372	0.557
YOLOv6-M	34.3	375.9*	0.504	0.734	0.538	0.187	0.414	0.590
YOLOv6-L	58.5	255.1*	0.517	0.737	0.553	0.211	0.418	0.597
VFNet (ResNet50+FPN)	33.6	14.1	0.495	0.711	0.538	0.200	0.420	0.567
VFNet (ResNet101+FPN)	53.7	11.7	0.503	0.726	0.545	0.197	0.425	0.576
VFNet (ResNeXt101-64x4d+FPN)	98.2	5.8	0.523	0.751	0.568	0.226	0.459	0.586

478 * The batch size is set to 32 for speed testing.

479 ## 5. Discussion

480 ### *5.1 Dataset contributions*

481 In this study, the HIOD dataset was introduced as a first-of-its-kind resource for detecting objects in
482 hospital environments. With a total of 4,417 images and 51,809 annotated objects across 56 object
483 categories, the HIOD dataset has the potential to support the development of innovative computer
484 vision-based applications within the healthcare sector. The dataset represents a significant step
485 forward in the field as it provides a comprehensive and diverse representation of the objects
486 commonly seen in hospital environments, making it a valuable resource for researchers and
487 practitioners alike. The HIOD dataset has been designed to tackle the limitations of previous
488 datasets, such as the lack of diversity and limited object categories, thus providing a much-needed
489 resource for advancing the state-of-the-art in hospital object detection. Table 7 provides a
490 comparison between the HIOD dataset and other available datasets within similar settings. This
491 comparison highlights the unique features and characteristics of the HIOD dataset and sheds light
492 on how it stands out from other existing datasets.

493 Table 7 Comparison with existing datasets within similar settings

Dataset	Scenario	# of images	# of Categories	Task
MCIndoor20000	Marshfield Clinic	2,055	3	Image classification
MYNursingHome	Nursing home	1,950	25	Image classification
Ours	Hospital facilities	4,417	56	Object detection

494

495 The HIOD dataset has several key features and benefits that make it significant for the field of
496 computer vision and object detection. These significant features of the HIOD dataset are discussed
497 in detail below.

Uniqueness. While there are many datasets for indoor object detection, very few, if any, datasets are developed specifically for hospitals. The hospital environment is unique from other indoor environments, because it contains a variety of medical equipment, such as ventilators and iv poles. Consequently, to facilitate the creation of applications in hospital settings, it is imperative that a dataset incorporates various pieces of medical equipment. However, common object detection datasets have not geared towards annotating these equipment, making them unsuitable for applications, such as robotic disinfection. The HIOD is a first-of-its-kind object detection dataset, providing tremendous opportunities for researchers and practitioners in developing new applications in hospital indoor environments.

Image diversity. The HIOD dataset is constructed by collecting a diversity of image and video data from online sources. In specific, a total of 59 videos from different hospitals were collected. The total length of these videos is 7hrs, and 24,963 individual frames are extracted. The images were also collected from a variety of online image search engines, including Google Images, Bing Images, Getty, and Shutterstock. In combination, a total of 43,197 images were collected in hospital indoor environments. After carefully cleaning the images that cannot meet requirements, the resulting HIOD dataset contains a total of 4,417 images, covering a diversity of hospital environments.

Object category. The HIOD dataset contains 56 object categories, which cover the most commonly seen objects and equipment found in hospitals. The HIOD dataset stands out in terms of object instance annotation density and diversity compared to other popular datasets such as COCO, OpenImages, and VOC. On average, each image in the HIOD dataset contains more objects and a greater number of object categories than images in these other datasets. This dense annotation makes it ideal for training and evaluating object detection algorithms. Additionally, the HIOD dataset provides a unique advantage by separately annotating hospital staff, patients, and visitors. This information could be extremely useful for developing context-aware human assistance applications that require an understanding of the hospital environment and the people within it.

Precise annotation: The image annotation process was outsourced to the Scale AI platform, which has established rigorous protocols to guarantee the annotation's accuracy. To ensure the annotation's quality, the Scale AI platform follows a series of meticulous steps during the annotation process. After the completion of the annotation process, the annotated dataset was carefully reviewed and evaluated by the research team through extensive training. The research team's comprehensive training on the annotated dataset served as an additional quality check, ensuring the dataset was of high accuracy and ready for use in further research and analysis.

5.2 Benchmark performance

A hospital object detection benchmark was created based on the HIOD dataset. The benchmark provides a baseline performance for future object detection algorithm development in hospitals. The benchmark results show that all detectors achieve an mAP greater than 40% and an AP_{50} greater than 64% on the HIOD dataset. This indicates that the detector trained on the training dataset can detect objects with satisfactory accuracy. One-stage algorithms achieved better performance in both detection accuracy and speed compared to two-stage algorithms. For example, YOLOv6-L achieved an mAP of 51.7% with an inference speed of 255 FPS. For two-stage algorithms, the best mAP is 49.5% with a much smaller inference speed of 14 FPS. Therefore, it is highly recommended to select one-stage algorithms for computer vision-based application development in hospitals for a balance between accuracy and speed. The HIOD dataset demonstrates a marginally lower benchmark performance in comparison to the COCO dataset. For instance, YOLOv6-L and YOLOv7 exhibit a slightly lower mAP of 51.7% and 50.6% on the HIOD dataset in contrast to their mAP of 52.5% and 51.2% on the COCO dataset. This underscores the effectiveness of the HIOD dataset, considering the complex and challenging nature of hospital indoor environments. The presence of small objects, such as handles and doorknobs, in the dataset presents a challenge for the object detector. Detecting small objects remains a significant challenge for deep learning networks due to their limited features. To improve object detectors' performance on the HIOD dataset, it would be promising to investigate some techniques for small object detection, such as increasing the model's input resolution, augmenting data, and auto-learning model anchors.

5.3 Hyperparameters evolution

Hyperparameters are parameters that are set before the training process begins and these parameters have a significant impact on the accuracy and efficiency of the model. The optimal values of these parameters can vary depending on the specific application and dataset. In this study, YOLOv7 is selected for hyperparameter optimization, which contains a total of 30 parameters related to learning rate, loss function, data augmentation, etc. In this study, genetic algorithm (GA) is used for YOLOv7 hyperparameter optimization, which is a form of evolutionary computing. The GA algorithm uses the principle of evolution to optimize hyperparameters by recombining and mutating the genes in each generation to produce a new population of better-performing candidate solutions. The fitness of each candidate solution is evaluated based on the performance of the model on a validation set, and the best-performing hyperparameters are selected for use in the final model. The fitness function is a weighted combination of mAP contributing 90% of the weight and AP_{50} contributing the remaining 10%. The mutation-based genetic operator is used with a probability of 0.9 and a variance of 0.04. The number of epochs and iteration are set to 150 and 200, respectively. Fig. 9 shows example plots for some optimized hyperparameters. Using the evolved parameters,

YOLOv7 attains a mean average precision (mAP) of 52.0% and 50.9% on the validation and testing datasets, respectively. This represents a slight improvement of 0.6% and 0.3%, compared to the performance achieved using the original hyperparameters. Although the increase in mAP is modest, it underscores the significance of hyperparameter tuning in optimizing the performance of object detectors on HIOD dataset.

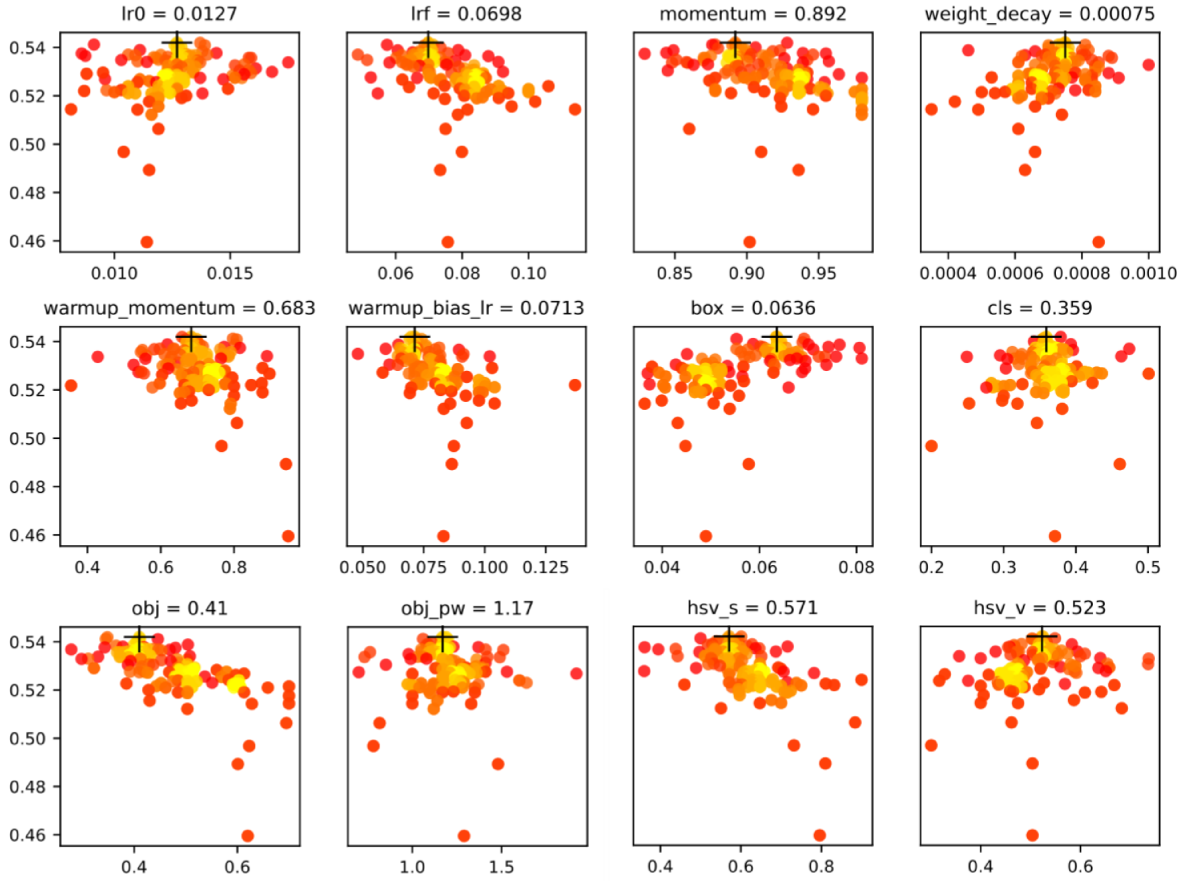

Fig. 9 Optimized YOLOv7 hyperparameters utilized for network training

5.4 Effect of transfer learning

In the benchmark, the transfer learning technique is used to improve performance and reduce the amount of data and computation to learn from scratch. Specifically, pre-trained weights on large COCO dataset are used to initialize the weights of a model for object detection in healthcare facilities. The effect of transfer learning on the object detector's performance on HIOD dataset is evaluated in this section. One-stage algorithms are selected for performance comparison with and without pre-trained models, and the results are shown in Table 8. The results of this comparison indicate that the performance of object detectors on HIOD dataset is significantly improved using pre-trained models. Specifically, the utilization of pre-trained weights in object detectors results in a noticeable mAP improvement of performance by a range of 5.1% to 9.1% on the HIOD dataset. The

significant performance improvement, as indicated by the results of the comparison, highlights the effectiveness of transfer learning for improving object detection performance, and suggests that using pre-trained models is a valuable technique for improving the performance of object detectors on the HIOD dataset.

Table 8 Comparative performance of one-stage object detectors with and without the application of transfer learning

Algorithm	Transfer learning			Scratch		
	mAP	AP_{50}	AP_{75}	mAP	AP_{50}	AP_{75}
YOLOv5-L	0.473	0.696	0.501	0.414	0.647	0.438
YOLOX-L	0.484	0.708	0.520	0.393	0.638	0.426
YOLOv6-L	0.517	0.737	0.553	0.439	0.656	0.470
YOLOv7	0.506	0.738	0.545	0.455	0.698	0.487

6. Conclusions and future research directions

The current research aimed to develop an image dataset for object detection in hospital indoor environments, recognizing the importance of understanding and improving the functionality of such environments. To fulfill this aim, a new dataset was created, named HIOD, by collecting images from various hospital indoor environments. The proposed dataset is comprised of 4,417 images and 51,809 annotated objects, displaying a significant diversity in scale and appearance. The dataset also includes annotations for 56 object categories, which cover the most frequently seen objects in hospitals. One of the primary goals of the HIOD dataset is to provide a benchmark for future algorithms development. To achieve this, the dataset was tested using eight state-of-the-art object detectors, and the results were evaluated to establish a benchmark for comparison. The benchmark results indicate that detectors trained on the HIOD dataset are capable of detecting objects from an image, demonstrating the effectiveness and utility of the dataset. In summary, the HIOD dataset and accompanying benchmark provide a valuable resource for researchers and practitioners interested in object detection in hospital indoor environments. The creation of the HIOD dataset presents new opportunities for exploring and understanding the complexities of these environments and for improving the functionality of hospitals. The diversity and size of the dataset ensure that future algorithms will be able to generalize well and be applicable to a wide range of indoor environments in hospitals.

This study has several limitations that must be acknowledged. Firstly, the HIOD dataset only consists of 4,417 images, which is significantly smaller compared to other datasets such as the COCO dataset which has 164,000 images. The size and diversity of the dataset play a vital role in determining the validity and generalizability of object detection algorithms. Therefore, it is imperative

to expand the HIOD dataset by collecting more images from various hospital indoor environments to improve its size and diversity. In addition to the limitations mentioned above, the HIOD dataset also has some object categories missing. Although the dataset annotated 56 object categories that are frequently seen in hospitals, some objects and equipment such as walkers, stethoscopes, and endoscopes may not have been captured in the dataset due to their infrequent appearance. To address this limitation, a natural next step would be to expand the dataset by including more categories of objects and collecting more images in the future. This would ensure that the dataset is comprehensive and captures a wider range of objects and equipment commonly seen in hospital indoor environments. Finally, the construction of the HIOD dataset is focused on detecting objects in images using bounding boxes. Hence, the images are annotated using bounding boxes instead of pixel-wise level annotations. This approach provides coarse detection of objects with a bounding box, but it is limited in terms of accuracy. Pixel-wise annotations are crucial for achieving a higher level of accuracy in object detection and enabling object segmentation at the pixel level. In light of these limitations, it is a promising area for future work to annotate the HIOD dataset at the pixel level to advance the accuracy of object detection algorithms. This will require additional effort and resources, but the results will be well worth it, leading to improved performance in object detection in hospital indoor environments.

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.

Acknowledgments

This research was funded by the US National Science Foundation (NSF) via Grant numbers: 2026719, 1952140, and 2038967. This research also received support from the Science Alliance at the University of Tennessee Knoxville (UTK) via the Joint Directed Research and Development Program. The authors gratefully acknowledge support from NSF and UTK. Any opinions, findings, recommendations, and conclusions in this paper are those of the authors and do not necessarily reflect the views of NSF and UTK.

Data availability

The dataset constructed in this study can be downloaded from the following link: https://github.com/Wangmmstar/Hospital_Scene_Data

References

[1] American Hospital Association, Fast Facts on U.S. Hospitals, 2022. https://www.aha.org/statistics/fast-facts-us-hospitals (accessed September 28, 2022).

[2] American Association of Critical-Care Nurses, Hear Us Out Campaign Reports Nurses' COVID-19 Reality, (2021). https://www.aacn.org/newsroom/hear-us-out-campaign-reports-nurses-covid-19-reality (accessed September 28, 2022).
[3] J. Holland, L. Kingston, C. McCarthy, E. Armstrong, P. O'Dwyer, F. Merz, M. McConnell, Service Robots in the Healthcare Sector, Robotics. 10 (2021) 47. https://doi.org/10.3390/robotics10010047.
[4] VynZ Research, Healthcare Service Robots Market, (2022). https://www.vynzresearch.com/healthcare/healthcare-service-robots-market (accessed September 30, 2022).
[5] F.S. Bashiri, E. LaRose, J.C. Badger, R.M. D'Souza, Z. Yu, P. Peissig, Object Detection to Assist Visually Impaired People: A Deep Neural Network Adventure, in: International Symposium on Visual Computing, Springer, 2018: pp. 500–510. https://doi.org/10.1007/978-3-030-03801-4_44.
[6] D. Hu, S. Li, Recognizing object surface materials to adapt robotic disinfection in infrastructure facilities, Computer-Aided Civil and Infrastructure Engineering. (2022). https://doi.org/10.1111/mice.12811.
[7] D. Hu, H. Zhong, S. Li, J. Tan, Q. He, Segmenting areas of potential contamination for adaptive robotic disinfection in built environments, Build Environ. 184 (2020) 107226. https://doi.org/10.1016/j.buildenv.2020.107226.
[8] C.-Y. Wang, A. Bochkovskiy, H.-Y.M. Liao, YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors, ArXiv Preprint ArXiv:2207.02696. (2022). http://arxiv.org/abs/2207.02696.
[9] C. Li, L. Li, H. Jiang, K. Weng, Y. Geng, L. Li, Z. Ke, Q. Li, M. Cheng, W. Nie, Y. Li, B. Zhang, Y. Liang, L. Zhou, X. Xu, X. Chu, X. Wei, X. Wei, YOLOv6: A Single-Stage Object Detection Framework for Industrial Applications, ArXiv Preprint ArXiv:2209.02976. (2022). http://arxiv.org/abs/2209.02976.
[10] D. Hu, S. Li, J. Du, J. Cai, Automating Building Damage Reconnaissance to Optimize Drone Mission Planning for Disaster Response, Journal of Computing in Civil Engineering. 37 (2023) 04023006. https://doi.org/10.1061/(ASCE)CP.1943-5487.0001061.
[11] M. Everingham, L. van Gool, C.K.I. Williams, J. Winn, A. Zisserman, The Pascal Visual Object Classes (VOC) Challenge, Int J Comput Vis. 88 (2010) 303–338. https://doi.org/10.1007/s11263-009-0275-4.
[12] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C.L. Zitnick, Microsoft COCO: Common Objects in Context, in: European Conference on Computer Vision, Springer, 2014: pp. 740–755. https://doi.org/10.1007/978-3-319-10602-1_48.

[13] A. Kuznetsova, H. Rom, N. Alldrin, J. Uijlings, I. Krasin, J. Pont-Tuset, S. Kamali, S. Popov, M. Malloci, A. Kolesnikov, T. Duerig, V. Ferrari, The Open Images Dataset V4, Int J Comput Vis. 128 (2020) 1956–1981. https://doi.org/10.1007/s11263-020-01316-z.

[14] S. Shao, Z. Li, T. Zhang, C. Peng, G. Yu, X. Zhang, J. Li, J. Sun, Objects365: A large-scale, high-quality dataset for object detection, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019: pp. 8430–8439.

[15] F.S. Bashiri, E. LaRose, P. Peissig, A.P. Tafti, MCIndoor20000: A fully-labeled image dataset to advance indoor objects detection, Data Brief. 17 (2018) 71–75. https://doi.org/10.1016/j.dib.2017.12.047.

[16] A. Ismail, S.A. Ahmad, A. Che Soh, M.K. Hassan, H.H. Harith, MYNursingHome: A fully-labelled image dataset for indoor object classification, Data Brief. 32 (2020) 106268. https://doi.org/10.1016/j.dib.2020.106268.

[17] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, Commun ACM. 60 (2017) 84–90. https://doi.org/10.1145/3065386.

[18] J.R.R. Uijlings, K.E.A. van de Sande, T. Gevers, A.W.M. Smeulders, Selective Search for Object Recognition, Int J Comput Vis. 104 (2013) 154–171. https://doi.org/10.1007/s11263-013-0620-5.

[19] X. Wang, T.X. Han, S. Yan, An HOG-LBP human detector with partial occlusion handling, in: 2009 IEEE 12th International Conference on Computer Vision, IEEE, 2009: pp. 32–39.

[20] P. Felzenszwalb, D. McAllester, D. Ramanan, A discriminatively trained, multiscale, deformable part model, in: 2008 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2008: pp. 1–8. https://doi.org/10.1109/CVPR.2008.4587597.

[21] Z. Ge, S. Liu, F. Wang, Z. Li, J. Sun, YOLOX: Exceeding YOLO Series in 2021, ArXiv Preprint ArXiv:2107.08430. (2021). http://arxiv.org/abs/2107.08430.

[22] A. Bochkovskiy, C.-Y. Wang, H.-Y.M. Liao, YOLOv4: Optimal Speed and Accuracy of Object Detection, ArXiv Preprint ArXiv:2004.10934. (2020). http://arxiv.org/abs/2004.10934.

[23] Z. Tian, C. Shen, H. Chen, T. He, FCOS: Fully Convolutional One-Stage Object Detection, in: 2019 IEEE/CVF International Conference on Computer Vision (ICCV), IEEE, 2019: pp. 9626–9635. https://doi.org/10.1109/ICCV.2019.00972.

[24] S. Zhang, L. Wen, X. Bian, Z. Lei, S.Z. Li, Single-Shot Refinement Neural Network for Object Detection, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, IEEE, 2018: pp. 4203–4212. https://doi.org/10.1109/CVPR.2018.00442.

[25] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, A.C. Berg, SSD: Single Shot MultiBox Detector, in: European Conference on Computer Vision, Springer, 2016: pp. 21–37. https://doi.org/10.1007/978-3-319-46448-0_2.

[26] J. Redmon, A. Farhadi, YOLO9000: better, faster, stronger, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017: pp. 7263–7271. https://doi.org/10.1109/CVPR.2017.690.

[27] J. Redmon, A. Farhadi, YOLOv3: An Incremental Improvement, ArXiv Preprint ArXiv:1804.02767. (2018). http://arxiv.org/abs/1804.02767.

[28] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You Only Look Once: Unified, Real-Time Object Detection, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2016: pp. 779–788. https://doi.org/10.1109/CVPR.2016.91.

[29] G. Jocher, A. Stoken, A. Chaurasia, N. Jirka Borovec, TaoXie, Y. Kwon, K. Michael, C. Liu, J. Fang, A. V, L. Tkianai, YxNONG, P. Skalski, A. Hogan, J. Nadar, L.M. Imyhxy, ultralytics/yolov5: v6.0 - YOLOv5n “Nano” models, Roboflow integration, TensorFlow export, OpenCV DNN support (v6.0), Zenodo. (2021). https://doi.org/10.5281/zenodo.5563715.

[30] H. Zhang, F. Li, S. Liu, L. Zhang, H. Su, J. Zhu, L.M. Ni, H.-Y. Shum, DINO: DETR with Improved DeNoising Anchor Boxes for End-to-End Object Detection, ArXiv Preprint ArXiv:2203.03605. (2022). http://arxiv.org/abs/2203.03605.

[31] X. Dai, Y. Chen, B. Xiao, D. Chen, M. Liu, L. Yuan, L. Zhang, Dynamic Head: Unifying Object Detection Heads with Attentions, in: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2021: pp. 7369–7378. https://doi.org/10.1109/CVPR46437.2021.00729.

[32] H. Zhang, Y. Wang, F. Dayoub, N. Sunderhauf, VarifocalNet: An IoU-aware Dense Object Detector, in: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2021: pp. 8510–8519. https://doi.org/10.1109/CVPR46437.2021.00841.

[33] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, J. Dai, Deformable DETR: Deformable Transformers for End-to-End Object Detection, ArXiv Preprint ArXiv:2010.04159. (2020). http://arxiv.org/abs/2010.04159.

[34] G. Ghiasi, T.-Y. Lin, Q. v. Le, NAS-FPN: Learning Scalable Feature Pyramid Architecture for Object Detection, in: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2019: pp. 7029–7038. https://doi.org/10.1109/CVPR.2019.00720.

[35] Z. Cai, N. Vasconcelos, Cascade R-CNN: Delving Into High Quality Object Detection, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, IEEE, 2018: pp. 6154–6162. https://doi.org/10.1109/CVPR.2018.00644.

[36] T.-Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, S. Belongie, Feature Pyramid Networks for Object Detection, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2017: pp. 936–944. https://doi.org/10.1109/CVPR.2017.106.

[37] S. Ren, K. He, R. Girshick, J. Sun, Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks, IEEE Trans Pattern Anal Mach Intell. 39 (2017) 1137–1149. https://doi.org/10.1109/TPAMI.2016.2577031.

[38] R. Girshick, Fast R-CNN, in: 2015 IEEE International Conference on Computer Vision (ICCV), IEEE, 2015: pp. 1440–1448. https://doi.org/10.1109/ICCV.2015.169.

[39] R. Girshick, J. Donahue, T. Darrell, J. Malik, Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation, in: 2014 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2014: pp. 580–587. https://doi.org/10.1109/CVPR.2014.81.

[40] A. Chandio, G. Gui, T. Kumar, I. Ullah, R. Ranjbarzadeh, A.M. Roy, A. Hussain, Y. Shen, Precise Single-stage Detector, ArXiv Preprint ArXiv:2210.04252. (2022). http://arxiv.org/abs/2210.04252.

[41] F. Zhuang, Z. Qi, K. Duan, D. Xi, Y. Zhu, H. Zhu, H. Xiong, Q. He, A Comprehensive Survey on Transfer Learning, Proceedings of the IEEE. 109 (2021) 43–76. https://doi.org/10.1109/JPROC.2020.3004555.

[42] X. Qiu, T. Sun, Y. Xu, Y. Shao, N. Dai, X. Huang, Pre-trained models for natural language processing: A survey, Sci China Technol Sci. 63 (2020) 1872–1897. https://doi.org/10.1007/s11431-020-1647-3.

[43] J. Yosinski, J. Clune, Y. Bengio, H. Lipson, How transferable are features in deep neural networks?, Adv Neural Inf Process Syst. 27 (2014).

[44] A. Buldas, A. Kroonmaa, R. Laanoja, Keyless Signatures' Infrastructure: How to Build Global Distributed Hash-Trees, in: Nordic Conference on Secure IT Systems, Springer, 2013: pp. 313–320. https://doi.org/10.1007/978-3-642-41488-6_21.

[45] K. Chen, J. Wang, J. Pang, Y. Cao, Y. Xiong, X. Li, S. Sun, W. Feng, Z. Liu, J. Xu, Z. Zhang, D. Cheng, C. Zhu, T. Cheng, Q. Zhao, B. Li, X. Lu, R. Zhu, Y. Wu, J. Dai, J. Wang, J. Shi, W. Ouyang, C.C. Loy, D. Lin, MMDetection: Open MMLab Detection Toolbox and Benchmark, ArXiv Preprint ArXiv:1906.07155. (2019). http://arxiv.org/abs/1906.07155.

[46] Z. Liu, G. Gao, L. Sun, Z. Fang, HRDNet: High-Resolution Detection Network for Small Objects, in: 2021 IEEE International Conference on Multimedia and Expo (ICME), IEEE, 2021: pp. 1–6. https://doi.org/10.1109/ICME51207.2021.9428241.