Robustness of Deep Learning Methods for Occluded Object Detection - A Study Introducing a Novel Occlusion Dataset

1st Ziling Wu
School of Computer Science
University of Nottingham
Nottingham, UK
psxzw11@nottingham.ac.uk

2nd Armaghan Moemeni
School of Computer Science
University of Nottingham
Nottingham, UK
pszam1@nottingham.ac.uk

3rd Simon Castle-Green
School of Computer Science
University of Nottingham
Nottingham, UK
pszsdc@nottingham.ac.uk

4th Praminda Caleb-Solly
School of Computer Science
University of Nottingham
Nottingham, UK
pszpc1@nottingham.ac.uk

***Abstract*—A large number of deep learning based object detection algorithms have been proposed and applied in a wide range of domains such as security, autonomous driving and robotics. In practical usage, objects being occluded are common, and can result in reduced accuracy and reliability. To increase the robustness of object detection algorithms under occlusion scenarios, it is necessary to consider the influence of different types of occlusion on the performance of object detection approaches. Our research revealed a gap in benchmarking datasets that could provide exemplars of occlusion that covered a range of occlusion scenarios. In this paper, we present a new benchmarking dataset that includes a range of exemplars providing coverage of different types of occlusion cases. This dataset is designed for object detection of everyday objects in indoor scenarios, and comprises occlusion in three orthogonal atomic factors, namely, the degree of occlusion, the location of occlusion, and classes of occluded object and those occluding other objects. Our dataset is balanced in terms of classes and degrees of occlusion, with a total of 5970 sample images. The effect of these three atomic factors has been investigated on some classic general object detectors. Using this benchmarking dataset, we also present results on the impact of the distribution of the training dataset, in terms of degree of occlusion, on the robustness of several typical object detection algorithms (e.g. Fast RCNN, Faster RCNN, and FCOS, etc). The benchmark is available at "https://drive.google.com/drive/folders/13VkLgbx6t0-vA3vRWrlvHcjra-8BS4aL?usp=sharing". This dataset is seen as a key contribution to research investigating the influence of occlusion on the performance of object detectors.**

I. Introduction

Object detection is a crucial task in computer vision, where the aim is to enable the computer to acquire an ability, similar to human vision, of recognising and locating objects of specific classes. Many deep learning based approaches have been proposed and widely applied to practical scenarios. With these algorithms, it is possible to locate and classify objects in multiple categories. However, as for most approaches, the performance of these algorithms inevitably deteriorates after being deployed in the practical usage (e.g. security [1], traffic [3], and robotics [2]), even if they have been well and fully trained. One common reason is that the distribution of training data and test data is different, i.e. the features and appearance of the instance in training datasets are different from that of test sets. The difference in features and appearance can be caused by clutter, illumination changes, and occlusion, which significantly decreases the robustness of object detectors.

Reduced accuracy due to occlusion is particularly common when it comes to the real-world deployment of object detection algorithms. Deep learning models are less likely to infer the existence of an occluded object with just visible parts of it [2]. In this paper, we describe occlusion as relating to three orthogonal atomic factors, namely, the extent or degree of the occlusion (e.g. half of an object is being occluded), the location of the occlusion (e.g. the left part of an object is being occluded), and the type of the occlusion (e.g. an object is being occluded by another object of the same or different class, such as a cup is being occluded by another cup or bottle). It is necessary to understand the influence of these atomic factors of occlusion on object detection to further improve the ability of models when dealing with occlusion scenarios.

Studying the effect of different distributions of the atomic factor "degree of occlusion" in the training set is also of interest. Most existing datasets consider the scale of each class (i.e. it is expected to contain a similar number of instances for each category to make the dataset balanced, or imbalanced situations are at least claimed and described), while none of them (to the best of our knowledge) takes the balance of occlusion into consideration. It is still unknown how the imbalance of occlusion impacts the performance of models.

Furthermore, the performance of different models dealing with occlusions is likely to vary due to the diverse mechanisms of object detection algorithms. Exploring and measuring typical object detection algorithms' robustness of detecting occluded objects can illustrate which methods are more appropriate for specific occlusion scenarios.

To study the aspects mentioned above, a benchmark which is specifically designed for representing different occlusion scenarios is indispensable. In this paper, we present NOODLES1 (aNnotated Occluded Objects Dataset for Evaluation and Learning Series 1) - a benchmark, which includes 5970 images, specifically designed with the representation of the three orthogonal atomic factors. It is balanced in both classes

and degrees of occlusion. Using this benchmark, the effect of these three atomic factors on the performance of some classic general object detectors (e.g. Fast RCNN [4], Faster RCNN [9], and FCOS [10], etc.) has been investigated. Furthermore, the impact of the distribution in terms of the degree of occlusion in the training set, on the robustness of the model to cope with occlusion, has also been explored.

(a) (b) (c) (d)

Fig. 1. Some examples of occlusion with different degrees. **a)** Single object without being occluded; **b)** Instances being slightly occluded by other objects. **c)** Occlusion of a medium degree. **d)** Examples of heavy occlusion.

In short, the contribution of this work is:
1) a benchmarking dataset has been created for investigating the impact of atomic factors on object detection algorithms.
2) investigating the influence of the distribution of datasets in terms of extent (or degree) of occlusion on the performance of the model under different occlusion scenarios.
3) exploring the challenge of occlusion of each atomic factor for typical object detection approaches.
4) identifying which object detection designs or mechanisms are appropriate for object detection under occlusion scenarios.

The paper is organized as follows. Section II introduces the previous work related to this paper. Section III illustrates the methods being used in this paper, such as the data collection method for the benchmark. Section IV indicates the design of each independent experiment and its corresponding results. Section V provides the analysis of the experiment results and gives reasonable suggestions on the future development of object detection techniques to further improve their ability to deal with occlusion scenarios. Section VI shows the conclusion of the paper and potential future work.

II. Related works

A. Deep Learning Based Object Detection Techniques

Object detection tasks can be divided into: regressing the location of a target, and classifying it into single or multiple categories. General CNN (convolutional neural networks) based object detection algorithms are amongst the most popular and can be classified into two types: 1) two-stage object detectors, which firstly propose many regions of interest (RoIs) representing bounding boxes which possibly cover an object in an image, and secondly utilize a classifier (e.g. Support Vector Machine and Fully Connected Layers) to assign a class to an RoI; 2) one-stage object detectors, which output bounding boxes and their corresponding classes simultaneously. As for two-stage object detection models, typical algorithms are R-CNN [5], Fast-RCNN [4], Faster-RCNN [9] and R-FCN [6]. In one-stage object detectors, representative models are YOLO [7], SSD [8], and FCOS [10]. Apart from CNN-based approaches, recently many algorithms based upon vision transformer have been proposed, such as DETR [11] and YOLOS [12]. In this paper, we focus on CNN-based models.

Existing deep learning models adopt different strategies with respect to regressing the size and coordinates of an object. One strategy is fine-tuning the size of an RoI (i.e. the model outputs the offset of an RoI so that the proposed bounding box maximises the intersection with the ground truth). Such a strategy has two approaches: 1) leveraging segmentation techniques (e.g. selective search [5]) to generate around 2000 RoIs for an image and adjusting the size of each RoI which is classified in a category instead of being considered as background [4], [5], [13]; 2) assigning multiple anchor boxes to each pixel of the feature map output by the neural network which judges whether an anchor box covers a possible target or not [8], [9] (object detectors based upon such methodology are named anchor-based models). Another approach is to regress the size and position of a target directly, which is anchor-free. Typically there are three different approaches to achieving this goal: 1) the top-left and bottom-right corners of a target are output and utilized to represent the location of it by the model [14]; 2) each pixel on a feature map output by a CNN model predicts the probability that it is the center of a target and estimates the height and width of it [3], [15]; 3) each pixel on a feature map projects the likelihood that it is located in the bounding box of an object and regresses the distance between the pixel and boundaries of the predicted bounding box [10].

In this paper, we present the results of experiments using some of the different approaches described above, measuring their ability to cope with the different occlusion scenarios contained in our benchmark. Showing which one is more or less appropriate for tackling occlusion will help to further direct the design of object detection models.

B. Data Collection

Data collection and labeling is not possible for occlusions of every instance in each category. As a result, most of the research has been conducted on synthetic datasets [2]. A commonly used method is to overlay random instances of an object on the image which has objects, that eventually causes occlusion [16]. Neural networks can then be trained to produce the occlusion mask, using the occluded regions' coordinates as labels. Another strategy is image augmentation, achieved by randomly selecting a region and adjusting its pixel value to zero [17]. This strategy enhances the robustness of models while handling occlusion. There are few datasets whose images are not synthetic and annotations are fully labeled manually. An example is KINS [18] which provides images of natural occlusion scenarios labeled by professional annotators. However, KINS is a large-scale dataset of outdoor scenarios. According to [2], there is no dataset comprised of real indoor scenes and well-labeled with annotations available.

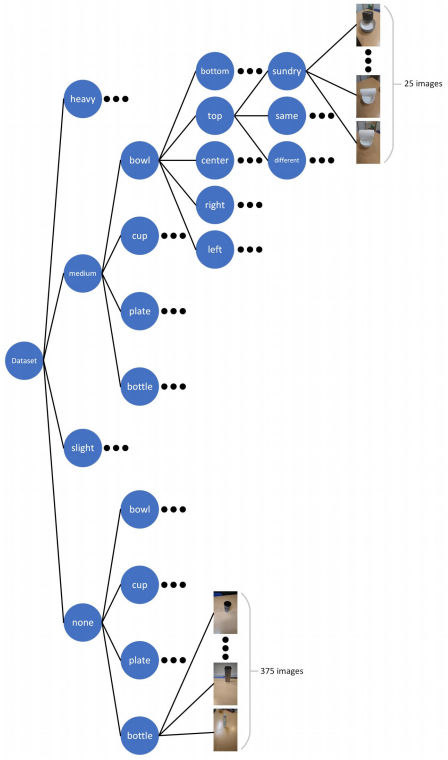

Fig. 2. The structure of the dataset.

C. Occlusion Detection

Occlusion detection refers to the recognition of whether an object is being obscured or not in an image. Qi et al. [18] proposed a Multi-Level Coding network which is designed to produce the visible and invisible parts of instances in an image. Specifically, an occlusion classification branch to predict the existence of occlusion is applied to boost the performance of the model. The model is trained on the KINS dataset. Similarly, the annotation of KINS also provides information on the existence and order of occlusion (e.g. the object which occludes other objects and the object being occluded), which can be leveraged to train a model predicting the occurrence of occlusion.

D. Techniques for Dealing with Occlusion

There are two main strategies to cope with partial occlusion when considering object detection. One is by the complement of occluded regions, and the other is by increasing the robustness of object detection techniques.

The complement of occluded regions may shrink the challenge of downstream tasks, e.g semantic segmentation and object detection. The general process can be divided into: 1) estimation of regions that are occluded; 2) repainting the occluded part of the object with RGB content. As, for the former, a model extended from Mask RCNN has been proposed by Follman et al. [19] to predict the mask of occluded regions. An unsupervised encoder-decoder network has been proposed by Zhen et al. [20] to estimate the area of occlusion. As for the latter, Zhan et al. [20] and Pathak et al. [21] utilized the encoder-decoder for the complement of the occluded regions' content. SeGAN [22] was proposed segmentation and generation of invisible areas.

A number of approaches have been proposed for increasing the robustness of object detectors. A data augmentation technique named 'Cutout' proposed by Devries and Taylor [23], fabricates artificial occlusion by setting the value of pixels to zero in the region randomly selected in an image. By applying such an approach, the model is less likely to overfit in case of occlusion. Adversarial Networks were used by Wang et al. [24] for feature mapping of normal objects to the features of deformed and occluded objects to augment the performance of Fast-RCNN [4] to detect occluded instances. Another strategy uses partial semantic information to refer only to the visible parts for object detection. In particular, Xiao et al. [25] proposed a comparison between features of visible parts and class patterns to segregate the irregular responses caused by the occlusion. In a series of works by [26]–[28], for partial regions, the generation of feature vectors with similar visual and semantic concepts was achieved through visual concepts [29].

Most of the previous work has been conducted to address occlusion in object detection, but to the best of our knowledge, no work has been done to understand the atomic factors of occlusions, or the impact these cause on an object detector's performance. In this paper, this gap is being addressed by starting to identify which factors (e.g. the location, degree, and appearance of occlusion) reduce the performance of a model, and in what way they impair the object detector's ability (proposing regions of interest, classification, and location), and to what extent they affect the robustness of the model.

III. Creation of the Benchmark

To investigate the influence of atomic factors of occlusion on object detection, a bespoke benchmark designed for this research is needed. In this section, the methods used to design, collect, and label the images in our benchmark dataset NOODLES1 is introduced.

A. Atomic Factors of Occlusion

In this paper, three atomic factors are considered: the degree of occlusion, the location of occlusion, and the type of occlusion.

1) Degree of Occlusion: The degree of occlusion refers to the ratio of the invisible part and visible part of the target object. To be more specific, there are 4 different degrees of occlusion in this paper, i.e. none (the object has no occlusion), slight (less than 20% of the object is occluded), medium (20% - 60% of the object is occluded), and heavy (over 60% of the object is occluded). Fig 1 shows some samples of the four different degrees selected randomly from our benchmark.

2) Location of Occlusion: The location of occlusion is defined as the position of the invisible part of an object being occluded. In this paper, it is divided into five different

parts: left, right, top, bottom, and center. Fig 3(a)(b)(c)(d)(e) illustrates some examples of these four invisible locations.

3) Types of Occlusion: The type of occlusion atomic factor refers to the class of the occluded object in relation to the class of the occluding object. In total, there are three types of occlusion considered in this paper: 1) different, which represents that an object is occluded by an object of a different class. For example, a cup occluded by a bottle; 2) same, which refers to an object being occluded by an object of the same class, e.g. a bowl occluded by another bowl; 3) sundry, meaning an object occluded by an object whose class is not included within the dataset. Four valid classes exist in our benchmark: bottle, cup, bowl, and plate. An example of sundry occlusion would be a plate occluded by a towel. Fig 3(f)(g)(h) gives some images of these three types of occlusion.

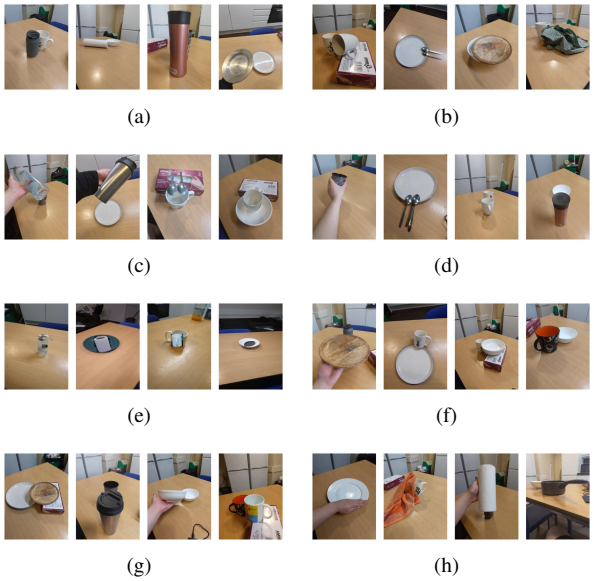

Fig. 3. Examples of occlusion at different locations: left **(a)**, right **(b)**, top **(c)**, bottom **(d)**, and center **(e)**. Examples of occlusion of three types: Objects occluded by instances of other classes **f)**, Instances occluded by objects from the same category **g)**, and Objects occluded by sundries **h)**.

B. Dataset Structure

The NOODLES1 benchmark was designed to contain four classes, namely: bottle, cup, bowl, and plate, since these are common objects in many practical scenarios and easy to obtain. There are five different instances of objects contained for each class. When it comes to sundries, five different kinds of items were chosen, i.e. tissue, towel, pot, hand, and spoon. We included two or three instances of each item.

In terms of the structure of the benchmark, for any occlusion scenario, the three atomic factors are always present, i.e. for an object being occluded, the occlusion is of a certain degree, location, and type of occlusion. It is necessary to make the benchmark balanced (e.g. the number of images of each atomic factor should be approximately same) so that the influence on the performance of the object detector caused by each

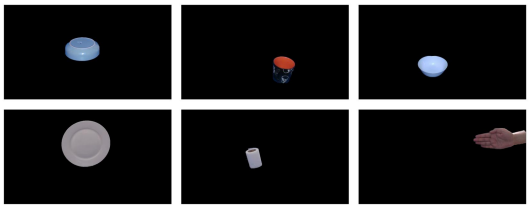

Fig. 4. Examples of masks which contain only the content of objects.

atomic factor is commensurable when the model is being trained with the images in the benchmark. To achieve this goal, a tree structure was utilized to organize the dataset. Fig 2 gives the information about that structure. The root node "Dataset" has four children related to the degree of occlusion. For each degree node, there exist four nodes representing each of the object classes. After that is the layer of locations, i.e. there are five location nodes under each class. Finally, each location node contains three types of occlusion nodes, which are considered as the leaf nodes of the tree. Each leaf node contains 25 images, i.e. five photos of each instance (there are five instances per class). The structure of the sub-tree whose root node is the degree "none" is different. Since there are no occlusions in this case, the layer of class nodes becomes the leaf nodes, under which 375 images are directly attached.

With the above structure, retrieving images of specific classes and atomic factors is easy, enabling research into the influence on the model from each atomic factor.

C. Dataset Balancing Scheme

Making the dataset balanced requires the number of images of each instance to be almost equal and each instance to be occluded by another instance of a specific occlusion type with the same probability (except for those of the degree "none"). Taking the class "cup" as an example, there are five instances of the cup, i.e. cup1, cup2, cup3, cup4, and cup5. As for the type of occlusion "same", cup1 can be occluded by any other cups with an even probability which equals 25%. In terms of the type of occlusion "different", cup1 can be occluded by any other instances of a different class (plate, bottle, and bowl) with an even probability equaling to $\frac{1}{15}$ ($\frac{1}{3} \times \frac{1}{5}$, where 3 refers to the number of the other classes and 5 represents the number of instances of one class). In addition, the number of times an instance is occluded should be similar. For example, cup1 should be occluded around 225 times in total. When it comes to the instance whose occlusion degree is "none", each instance is one of 75 photos taken from different angles and aspects, so that the number of images of each occlusion degree is the same, which equals 1500. In this paper, the dataset scheme is generated randomly based on the probability and number of instances mentioned above.

D. Data Collection

This dataset is designed to restrict the variety of other factors (as well as atomic factors) which may influence the

performance of object detection algorithms. There are two aspects constrained in this dataset: background and illumination.

Variation in background results in an uncertainty of data distribution, causing difficultly in judging whether detection failure is caused by occlusion or changes in the background. To avoid such confusion, firstly, the training set and test set are required to share a similar data distribution. Secondly, variation in image backgrounds should be limited. To achieve this, all images were taken in the same room whose layout, decoration, and furniture remained unchanged.

Changes in illumination levels can result in similar confusion, making it difficult to judge whether incorrect results are caused by occlusion or unstable illumination situations. Therefore, all images were taken using only artificial light sources to maintain consistency.

Around 1500 images were taken for each degree of occlusion. For the degree of occlusion categories, "slight", "medium", and "heavy", images were captured manually. Only two instances were allowed to be included in a single image, i.e. only one occlusion occurs per image, so that the number of occlusions for each combination of atomic factors is approximately same (i.e. the dataset is balanced). The content of each image (class of each instance, location of the occlusion, degree of the occlusion, and arrangement order of each instance) was set in strict accordance with the dataset scheme mentioned in section III-C. In terms of the degree "none", a video was taken for each instance, and 75 frames were chosen randomly and saved as the images of the instance. This approach significantly reduces the time of data collection for a single instance.

	none	slight	medium	heavy	total (each class)
plate	375	364	375	375	1489
bowl	375	370	378	376	1499
bottle	375	367	371	375	1488
cup	375	373	370	376	1494
total (each degree)	1500	1474	1494	1502	overall 5970

TABLE I
THE OVERVIEW OF THE BENCHMARK.

Note that the location of occlusion "center" (i.e. the invisible part of an instance situated at its center) is more difficult to obtain. Due to the principles of perspective, an instance can only be partially occluded by objects smaller than it, and it is possible to increase the degree of the occlusion by reducing the distance between the object occluding the target and the camera or by increasing the distance between the camera and the instance being occluded. However, when the distance between two objects is zero, it is impossible to further reduce the degree of occlusion by adjusting the distance between them and the camera. With two objects of similar sizes, it is difficult to obtain the image in which the occlusion degree is "slight" and "medium". To address this, pseudo data was generated. Masks containing only the content of objects, instead of background, were generated, as illustrated in Fig 4. Graph Cut [30] was utilized to make these masks.

The segmentation mask of each instance can be used to generate pseudo data by putting it in the center of an object of another image. To make sure that each image (those whose occlusion degrees are "slight", "medium", and "heavy" and the location of occlusion is center) only contains two objects, the pseudo data was generated by combining the image whose occlusion degree is "none" (i.e. only one object is in each image) with a segmentation mask (as detailed above). This task should be completed post-annotation, since the size and location of instances, whose occlusion degree is "none", is required in order to place the segmentation mask.

E. Labelling

Since NOODLES1 is designed for supervised learning techniques, it is necessary to generate labels for corresponding images. Each label is able to be represented by bounding boxes with classes. In this paper, the software LabelImg [31] is utilized to generate annotations for all the images. After manually labeling the instances, an ".xml" file is generated and saved in the local path by the software, in a style similar to PASCAL VOC [32].

F. Comparison with Existing Benchmarks

To highlight the novelty and contribution of the benchmark proposed in this paper, this section compares some existing representative datasets with ours. Table II shows the comparison between our dataset and seven other existing benchmarks, namely, OpenLORIS [33], KINS [18], CityPersons [34], PASCAL VOC2011 [32], CIFAR-100 [35], CUB-200 [36], and ARID [37]. Eleven dimensions are considered here, as shown below:

Context - indoor or outdoor based dataset.
Classes - number of classes covered by each dataset.
Scale - number of images in each dataset.
Illumination - diversity of illumination across image samples.
Background - diversity of backgrounds across image samples.
Occlusion Frequency - how often occlusion occurs.
Occlusion Labeled - presence of occlusion annotation.
Occlusion Decomposition - classification of images by atomic factors (degree, location, and type).
Balance - shows if the dataset is balanced in relation to the representation across the atomic factors.
Annotation Level - sophistication of annotation.
Fully Labeled - completeness of annotation.

The aim of the benchmark is to enable evaluation of the weakness and robustness of object detection algorithms under occlusion scenarios, not just overall performance. To enable research into robustness and the impact of occlusion (e.g. the influence of each atomic factor of occlusion), a benchmark should meet the following conditions:

1) illumination and background in images should be consistent, since it is difficult to judge whether a false negative or false positive is caused by the illumination/background difference between the training and test dataset or the invisibility resulting from occlusion. Stability in illumination and background enables evaluation without this interference.

2) occluded objects should be identified by annotations, enabling researchers to evaluate a model's performance in detecting occluded objects.

Dataset	Context	Classes	Scale	Illumination & Background	Occlusion Frequency	Occlusion Labeled	Occlusion Decomposition			Balance				Annotation Level	Fully Labeled
							degree	location	type	class	degree	location	type		
OpenLORIS	indoor	40	1106424	various	many	✗	explicitly	✗	✗	✓	✗	✗	✗	bounding boxes & masks	✗
KINS	outdoor	2	14991	various	many	✓	implicitly	implicitly	implicitly	✓	✗			bounding boxes & masks	✓
CityPersons	outdoor	30	5000	various	many	✓	implicitly	implicitly	✗	✗				bounding boxes & masks	✗
VOC2011	both	20	17125	various	many	✓	✗	✗	✗	✗	✗	✗	✗	bounding boxes & masks	✗
CIFAR-100	outdoor	100	60000	various	none	✗	✗	✗	✗	✓	✗	✗	✗	instances	✓
CUB-200	outdoor	200	6033	various	few	✗	✗	✗	✗	✗	✗	✗	✗	rough outlines	✓
ARID	indoor	51	6000	various	few	✓	✗	✗	✗					bounding boxes	✓
NOODLES1	**indoor**	**4**	**5970**	**similar**	**many**	✔	**explicitly**	**explicitly**	**explicitly**	✔	✔	✔	✔	**bounding boxes**	✔

TABLE II

THE COMPARISON BETWEEN NOODLES1 AND OTHER EXISTING DATASETS. AS FOR OCCLUSION DECOMPOSITION, "EXPLICITLY" REFERS TO THE DATASET INCLUDING LABELS ILLUSTRATING OCCLUSION'S DEGREE, LOCATION, AND TYPES. "IMPLICITLY" INDICATES THAT ALTHOUGH THERE ARE NO LABELS THAT GIVE INFORMATION ABOUT OCCLUSION'S ATOMIC FACTORS, IT IS POSSIBLE TO DEDUCE THESE FACTORS FROM THE GIVEN LABELS.

3) Annotation labels should indicate the occluded objects' corresponding atomic factors. The label should cover the degree, location, and type of the occluded part. As a result, the influence of each atomic factor on the model's training and test performance can be investigated.
4) The dataset should be balanced in terms of both class and atomic factors, by virtue of which the influence of sample imbalance on model performance can be eliminated.

Table II indicates that only our proposed benchmark, NOODLES1, fulfills all of the conditions introduced above, whereas other existing benchmarks meet only partial requirements. This can be utilized for research investigating the influence of each factor on the performance of object detectors. The overall composition of the benchmark is presented in Table I.

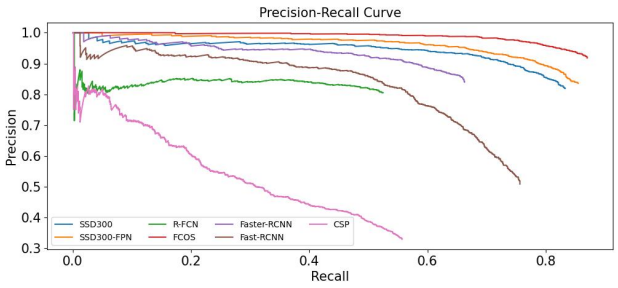

Fig. 5. The precision-recall curve of each algorithm.

IV. EXPERIMENT

In this section, various approaches were implemented using training and test data from NOODLES1, with the corresponding results presented and analysed. Representative models were chosen to conduct the experiment in this paper: a Selective Search [5], [40] based model, i.e. Fast-RCNN [4], two RPN based (anchor-based) two-stage frameworks (Faster-RCNN [9] and R-FCN [6]), two anchor-based one-stage approaches, namely, SSD300 [8] and SSD300-FPN [38], and two anchor-free one-stage algorithms (CSP [3] and FCOS [10]).

When it comes to training models, the Adam optimization method [39] was used to adjust the parameter of the model with a learning rate of 0.0001, mini-batch size of 16, and weight decay of 0.005. Transfer learning was applied to the backbone parameters with a learning rate of 0.000001. Images

RoI Proposal Mechanism	Model	Backbone	Recall	Precision	F1score	mAP
		Two-stage				
Selective Search based	Fast-RCNN	VGG16 [41]	0.75	0.53	0.62	0.65
Anchor-based	Faster-RCNN	VGG16	0.66	0.84	0.74	0.62
	R-FCN	VGG16	0.52	0.81	0.63	0.44
		One-stage				
	SSD300-FPN	VGG16	0.85	0.83	0.84	0.82
	SSD300	VGG16	0.83	0.82	0.82	0.79
Anchor-free	CSP	ResNet50 [42]	0.35	0.55	0.43	0.29
	FCOS	ResNet50	0.87	0.91	0.89	0.86

TABLE III

THE OVERALL PERFORMANCE OF MODELS TRAINED AND TESTED ON THE BENCHMARK BEING PROPOSED IN THIS PAPER.

are normalized and resized to 300×300 before being input into the model. All the models were trained for 50 epochs. During the measurement of precision, recall and mAP (mean Average Precision), an IoU (Intersection over Union) threshold of 0.5 was utilized to judge if a bounding box covers another one (i.e. a bounding box covers another one if the IoU between them is greater than 0.5). The training set and test set was split randomly with a ratio of 4:1 to maintain a similar distribution of background (the dividing scheme of training set and test set has been released with the images and annotations together).

A. Robustness of Object Detection Mechanisms

Table III illustrates the overall performance of each model. Additionally, Fig 5 gives information about the precision-recall curve of each approach. Compared with other models, FCOS shows the best performance in tackling occlusion scenarios, while CSP and R-FCN fail to detect most instances in the test dataset correctly. As for anchor-based models, the one-stage models perform better than two-stage models. The precision between two-stage and one-stage models is similar. However, when it comes to recall, one-stage models are significantly better than two stage-models. The possible reason is that RPN (Region Proposal Network [9]) is not able to propose enough RoIs to cover all the instances in an image of occlusion scenarios. By contrast, the precision of Fast-RCNN is lower than most of other models because selective search proposes too many RoIs, which results in many negative samples (RoIs which do not cover any instances) kept by the classifier. That is because the accuracy of the classifier is fixed, and more RoIs to be classified causes more RoIs being classified incorrectly.

Overall, Fig 5 indicates that the mechanism of FCOS is appropriate for coping with occlusion scenarios, while that of R-FCN and CSP is not fit for that. The reason for it is going to be further discussed in Section V.

B. Challenges of Atomic Factors

The challenge of dealing with the different atomic factors of occlusion has been considered in this paper. Each of the seven models, i.e. Fast-RCNN, Faster-RCNN, R-FCN, SSD300, SSD300-FPN, CSP, and FCOS, were trained with all of the images in the training set ensure a balance of classes and degrees of occlusions. All the models were trained for 50 epochs before being tested with the test set. When it comes to the evaluation, recall has been selected as the key performance indicator, since the challenge of occlusion can be represented by the percentage of occluded instances being detected correctly. Only the recall of occluded instances has been evaluated, while the instance without occlusion or occluding other objects has not been counted during the evaluation. That is because, when measuring the ability of the object detector to detect the occluded object, whether the occluded object is correctly detected is noteworthy, instead of those occluding others, and can be represented by recall.

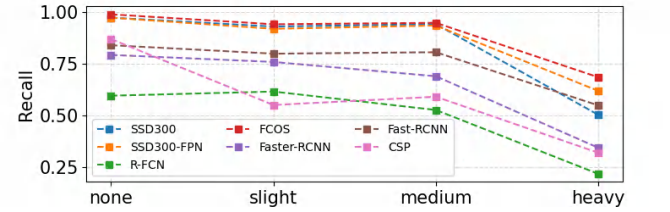

Fig. 6. The recall of instances of different occlusion degrees of each model.

1) Degree of Occlusion: After training the model, recall of each model on test instances of different degrees of occlusion was measured, as shown in Fig 6. If the curve of CSP and R-FCN is not considered (since they show a poor performance when dealing with occlusion scenarios, and so we are unable to infer anything concrete regarding occlusion scenarios), then a clear pattern of difficulty relating to the different degrees of occlusion can be seen in Fig 6. Scenarios without occlusion are easiest (since all the models show the best performance on instances with no occlusion). Then all the models' recall show a slight decrease in terms of "slight" occlusion. After that, unexpectedly, the recall of the majority of models does not further decrease as for "medium" occlusion, instead, they experience a minute increase (at least fluctuate or maintain this level). Finally, the recall of all models drops considerably when it comes to instances of "heavy" occlusion.

To summarise, the recall of CNN-based object detectors does not decrease linearly with the increase in the degree of occlusion, but initially decreases gradually and fluctuates, and then drops significantly when the occlusion is higher.

2) Locations of Occlusion: In this section, the recall of each approach on test instances whose invisible regions are situated in different locations has been evaluated, as shown in Fig 7. Unexpectedly, although CNN is well known for its rotation in-variance, CNN-based object detectors show various recalls when detecting objects being occluded in different orientations. Discarding CSP and R-FCN, all the anchor-based

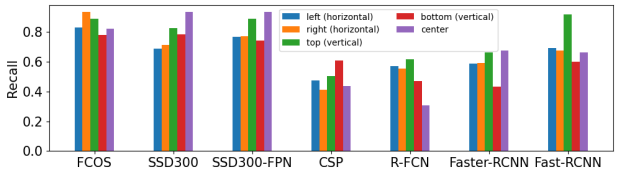

Fig. 7. The recall of instances with invisible parts in different locations.

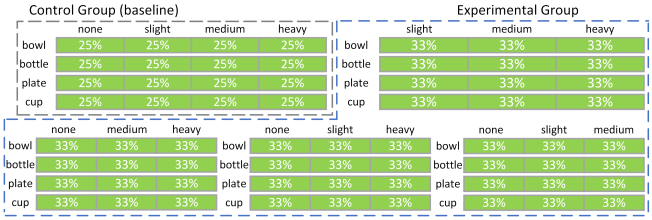

Fig. 8. The distribution of datasets which are leveraged to research into the importance of images of different occlusion degrees during training models.

models (i.e. SSD300, SSD300-FPN and Faster-RCNN) share the same pattern. The occlusion of "left" and "right" share a similar difficulty, and are harder than the occlusion of "top". In the vertical direction, occlusion of "top" is easier than that of "bottom". Finally, the occlusion of "center" is the easiest for all anchor-based frameworks.

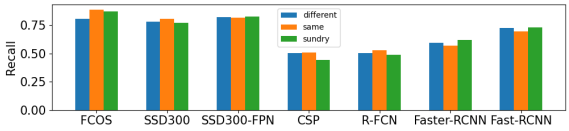

Fig. 9. The recall of instances with occlusion of different types.

3) Types of Occlusion: The recall of samples which are occluded by instances of different types has been evaluated after training the model, as shown in Fig 9. There is no fixed pattern shared by all the models. As a result, it can be speculated that the challenge of different occlusion types is the same for CNN-based object detectors.

C. Distribution's Occlusion Imbalance

In this paper, the influence of the dataset with different distributions has been investigated. Concretely, the impact of the proportion of images of different occlusion degrees was explored by investigating the influence of occlusion imbalance. For this experiment, it was necessary to make the training set balanced in terms of categories and to adjust the percentage of images of different degrees of occlusion. An example is illustrated in Fig 8. The number of images of each class is equal and fixed to make the training set balanced in the case of categories. As for the baseline (control group), the proportion of occlusion degree of each category is 25%. 1200 images have been selected randomly from the training set to train the model. Consequently, there are 300 images of each degree of occlusion in the baseline. When it comes to the experimental

group, only images comprising the three degrees of occlusion are kept, with 400 samples for each degree. The training set is balanced in terms of classes, i.e. the number of samples of each class is the same. Two models, namely, SSD300 and SSD300-FPN have been selected to be trained with the dataset of each distribution. Each model of SSD300 has been trained for 80 epochs, and those of SSD300-FPN have been trained for 50 epochs (The loss function of SSD300-FPN converges faster than that of SSD300 during training). Fig 10 shows the performance of them after being trained on the dataset of different distributions mentioned above.

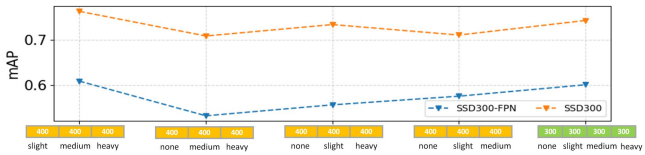

Fig. 10. The mAP of SSD300 and SSD300-FPN after they have been trained on the dataset of different distributions.

As shown in Fig 10, compared with the baseline which is balanced in terms of distribution, the absence of samples of "slight", "medium" and "heavy" occlusion results in the degradation of the performance of the model. Especially, the images of "slight" occlusion play the most important role in training the model, without which the performance of the model deteriorates most. By contrast, the model shows a better accuracy of detection than the baseline when it is trained with the dataset without occlusion. One possible reason is that the absence of "none" data leads to more samples of occlusion of different degrees, which significantly increases the robustness of the model when dealing with occlusion scenarios.

V. Discussion

In this section, the applicability of different object detection mechanisms to occlusion scenarios are analysed based on the experiment results presented in the previous section. Following this, suggestions for the construction of the dataset and the design of models are introduced which could lead to further improvement of the robustness of object detectors.

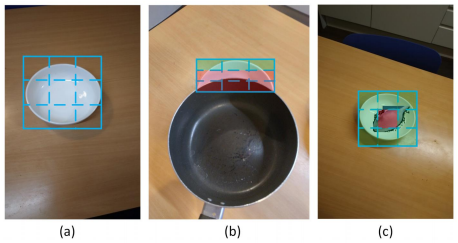

Fig. 11. Examples of success and failure of Position-sensitive RoI pooling.

A. Analysis of Mechanisms of Object Detection

In this section, the question of why the mechanism of CSP and R-FCN fail to maintain robustness in occlusion scenarios, and why FCOS works, are going to be answered.

1) Reasons for Failure of CSP and R-FCN: As shown in Table III, both CSP and R-FCN show poor performance on the test set of occlusion scenarios due to both being position-sensitive mechanisms. R-FCN utilizes Position-sensitive RoI pooling (PSRP) as the classifier, which considers all components of targets of a class, i.e. it outputs confidence for each component, and the final confidence of an RoI is the summation of the confidence of each component. For example, Fig 11 (a) indicates that PSRP outputs high confidence of class "bowl" if the instance is not occluded. However, when it comes to situations like Fig 11 (b) and (c), PSRP cannot give high confidence because only blocks of green colour are considered and contribute to the summation (confidence of the RoI), while the confidence of red blocks output by the model is very small leading to a low summation. Since many samples in the training set are of occlusion scenarios, it is likely to induce confusion in the model. For example, as per Fig 11, the left block is for detecting the left components of the bowl, during the training stage, sometimes it receives features of a left component of a bowl (as shown in Fig 11 (a)), while sometimes it receives features of a top-left component of a bowl (such as Fig 11 (b)). Such a phenomenon results in an error that a block cannot always receive features corresponding to its responsibility, which significantly disrupts training.

Fig. 12. The positive and negative samples are not fixed in different images when a CSP model is being trained.

Similarly, the reason for the failure of CSP is its position-sensitive nature. During training, it only considers the center point of the bounding box (annotations) and its eight neighbours as positive samples, even if they do not belong to the ground truth. By contrast, other pixels in ground truth are always treated as negative samples, even though they are part of the object. For instance, as shown in Fig 12 (a), only the center and its eight neighbors (red points) are considered as positive samples, and other pixels (e.g. green points in Fig 12 (a)) are negative samples. However, when it comes to the situation of Fig 12 (b), the red pixels, which are considered as negative samples in Fig 12 (a) (the green points), are seen as positive samples in Fig 12 (b), i.e. regions with similar visual appearance and features can be chosen as positive samples or negative samples in different circumstances. Such conflicts make the model fail in distinguishing between objects and backgrounds correctly. In Fig 12 (c), during training of a CSP model, red points belonging to sundry are improperly selected as positive samples, while green pixels, part of the target, are considered as negative samples. As a result, models

based on the detection of mechanisms of CSP show poor performance on object detection under occlusion scenarios, especially where the invisible part is centered.

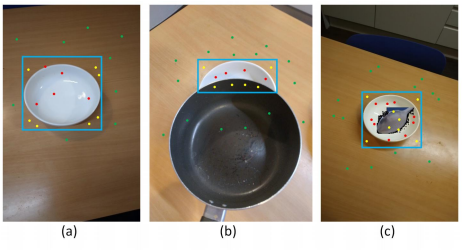

Fig. 13. Relationships between a pixel and the ground truth. Red points refer to pixels selected as positive samples, belonging to the ground truth. Yellow points are chosen as positive samples, not belonging to the target. Green points are pixels of negative samples, belonging to the background.

2) Causes of Success of FCOS: As shown in Table III, FCOS shows the best performance on the test set because the mechanisms of the positive sample selection strategy are appropriate for occlusion scenarios. As shown in Fig 13, there are three states of a pixel: 1) red pixels are part of an object and correctly selected as positive samples; 2) yellow pixels belong to the background but are improperly seen as positive samples; 3) green pixels are in the background and correctly treated as negative samples. These three states have different influences on training. Red points and green points are with positive roles, with the former enabling the model to classify different objects, and the latter distinguishing between foreground and background. By contrast, yellow points mislead training as they belong to the background, but the training process treats them as positive samples (foreground). Since the number of green points is much greater than that of yellow points, the model can still gain a robust ability to classify between foreground and background. Additionally, red points are always chosen as positive samples during training. Taking Fig 13 (a) and (b) as examples, in (a), the bowl is not occluded, and the red points at the top are considered as positive samples. In Fig 13 (b), points in the top of the bowl are still selected as positive samples. The problem for mechanisms of CSP mentioned previously never happens during the training of FCOS, which makes the pattern of features of positive samples always stable and easy to be recognized by CNN.

B. Model Development Suggestions for Occlusion Scenarios

1) Data Collection: As illustrated with Fig 10, when practitioners collect datasets to develop object detectors intended for deployment in occlusion scenarios, only making the dataset balanced in terms of classes is insufficient. If the scale of the dataset is limited, then it is suggested to reduce the proportion of instances without occlusion. It is not recommended to reduce the proportion of samples of slight occlusion in the dataset, as slight occlusion can provide both relatively complete features of an instance and information about occlusion. Fig 10 indicates that reducing the percentage of instances of slight occlusion is highly likely to cause models deteriorate in occlusion scenarios.

2) Model Design: Considering the design of CNN-based models for occlusion scenarios, position-sensitive mechanisms (e.g. PSRP and CSP) should not be selected as core modules of the RoIs proposal, and it is not appropriate for models which are highly likely to be deployed in scenarios of occlusion. By contrast, anchor-based RPN and mechanisms of FCOS (judging if a pixel is part of an object and proposing the distance between this pixel and four boundaries of the bounding box of the target) have been shown to give a robust performance on object detection tasks in occlusion scenarios, and should be considered for difficult occluded object detection.

VI. Conclusion

In this paper, occlusion scenarios in object detection have been systematically analyzed. In order to achieve this, a new benchmark dataset, NOODLES1, which includes samples of occlusion as per three atomic factors (degree, location, and type) with additional conditions i.e. similar illumination and background, has been created and described. NOODLES1 contains 5970 images. It is balanced in both categories and all the atomic factors of occlusion to enable the research into the impact of different factors. Using this dataset, the impact of different kinds of occlusions and their distribution are considered on the performance of CNN-based object detectors. The findings are as follows:

1) With increasing degrees of occlusion, the detection accuracy of the model does not decrease linearly. For slight and medium occlusions, the performance of the model degrades little compared with that of the scenarios without occlusion. When the degree of occlusion is considerably higher, the performance of the model decreases significantly;

2) When NOODLES1 is used to test model performance, top occlusion (invisible parts at the top of occluded objects) is simpler than horizontal occlusion (left or right of occluded object) for anchor-based models;

3) The type of occlusion (class of object occluding other instances) has no significant effect on detection accuracy;

4) The proportion of samples with different degrees of occlusion in the training set also affects the performance of the model. Reducing the proportion of samples without occlusion significantly improves the model performance. In addition, the lack of slightly occluded samples in the training data reduces the accuracy of the detector significantly;

5) Object detectors of position-sensitive mechanisms (e.g. PSRP and CSP) are not robust enough to cope with tasks of occlusion scenarios;

This research is not without drawbacks. First, the number of instances in each class is limited. Second, only CNN-based models have been investigated. Third, mechanisms for tackling occlusion of object detection (as introduced in Section II) have not been considered. As a result, future research will focus on:

1) extending the benchmark by increasing the number of classes and the number of instances in each class;

2) evaluating the performance of ViT-based [43] object detectors to further research into the impact of atomic factors of occlusion on those models;

3) trialling other approaches designed for dealing with occlusion scenarios, and testing their effectiveness against the NOODLES1 benchmark presented in this paper.

REFERENCES

[1] Wu, J., Nian C., Wenjie C., Huiheng W., Guotian W., "Automatic detection of hardhats worn by construction personnel: A deep learning approach and benchmark dataset," Automation in Construction 106: 102894. 2019.

[2] Saleh, K., Sándor S., and Zoltán V., "Occlusion handling in generic object detection: A review," IEEE 19th World Symp on Applied Machine Intelligence and Informatics (SAMI), pp. 000477-000484. IEEE, 2021.

[3] Liu, W., Shengcai L., Weiqiang R., Weidong H., Yinan Y., "High-level semantic feature detection: A new perspective for pedestrian detection," in Procs of IEEE/CVF conf on computer vision and pattern recognition, pp. 5187-5196. 2019.

[4] Girshick, R., "Fast r-cnn," in Procs of the IEEE Int Conf. on computer vision, pp. 1440-1448. 2015.

[5] Girshick, R., Donahue, J., Darrell, T., Malik, J."Rich feature hierarchies for accurate object detection and semantic segmentation," Procs of the IEEE conf on computer vision and pattern recognition, pp. 580-87. 2014.

[6] Dai, Jifeng, Yi Li, Kaiming He, and Jian Sun, "R-fcn: Object detection via region-based fully convolutional networks," Advances in neural information processing systems 29. 2016.

[7] Redmon, J., Divvala, S., Girshick, R., Farhadi, A. "You only look once: Unified, real-time object detection," in Procs of the IEEE conf on computer vision and pattern recognition, pp. 779-788. 2016.

[8] Liu, Wei, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C. Berg, "Ssd: Single shot multibox detector," in Computer Vision–ECCV 2016: 14th European Conf., Procs, Part I 14, pp. 21-37. Springer Int. Publishing, 2016.

[9] Ren, Shaoqing, Kaiming He, Ross Girshick, and Jian Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," Advances in neural information processing systems 28 (2015).

[10] Tian, Zhi, Chunhua Shen, Hao Chen, and Tong He, "Fcos: Fully convolutional one-stage object detection," in Proceedings of the IEEE/CVF international conference on computer vision, pp. 9627-9636. 2019.

[11] Carion, Nicolas, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko, "End-to-end object detection with transformers," in Computer Vision–ECCV 2020: 16th European Conf, Procs, Part I 16, pp. 213-229. Springer Int Publishing, 2020.

[12] Fang, Yuxin, Bencheng Liao, Xinggang Wang, Jiemin Fang, Jiyang Qi, Rui Wu, Jianwei Niu, and Wenyu Liu, "You only look at one sequence: Rethinking transformer in vision through object detection," Advances in Neural Information Processing Systems 34 (2021): 26183-26197.

[13] He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," IEEE transactions on pattern analysis and machine intelligence 37, no. 9, pp. 1904-1916. 2015.

[14] Law, H., Deng, J., "Cornernet: Detecting objects as paired keypoints," Procs of the European conf on computer vision, pp. 734-50. 2018.

[15] Zhou, Xingyi, Dequan Wang, and Philipp Krähenbühl, "Objects as points," arXiv preprint arXiv:1904.07850. 2019.

[16] Li, Ke, and Jitendra Malik, "Amodal instance segmentation," in Computer Vision–ECCV 2016: 14th European Conf, Procs, Part II 14, pp. 677-693. Springer International Publishing, 2016.

[17] DeVries, Terrance, and Graham W. Taylor, "Improved regularization of convolutional neural networks with cutout," arXiv preprint arXiv:1708.04552. 2017.

[18] Qi, Lu, Li Jiang, Shu Liu, Xiaoyong Shen, and Jiaya Jia, "Amodal instance segmentation with kins dataset," in Procs of the IEEE/CVF Conf on Computer Vision and Pattern Recognition, pp. 3014-3023. 2019.

[19] Follmann, P., König, R., Härtinger, P., Klostermann, M., Böttger, T., "Learning to see the invisible: End-to-end trainable amodal instance segmentation," in 2019 IEEE Winter Conf on Applications of Computer Vision (WACV), pp. 1328-1336. IEEE, 2019.

[20] Zhan, X, Pan, X., Dai, B., Liu, Z. Lin, D., and Loy, CC., "Self-supervised scene de-occlusion," in Procs of the IEEE/CVF Conf on Computer Vision and Pattern Recognition, pp. 3784-3792. 2020.

[21] Pathak, D., Krahenbuhl, P., Donahue, J., Darrell, T., and Efros, AA., "Context encoders: Feature learning by inpainting," in Procs of the IEEE conf on computer vision and pattern recognition, pp. 2536-2544. 2016.

[22] Ehsani, Kiana, Roozbeh Mottaghi, and Ali Farhadi, "Segan: Segmenting and generating the invisible," in Proceedings of the IEEE conf on computer vision and pattern recognition, pp. 6144-6153. 2018.

[23] DeVries, Terrance, and Graham W. Taylor, "Improved regularization of convolutional neural networks with cutout," arXiv preprint arXiv:1708.04552. 2017.

[24] Wang, X., Shrivastava, A and Gupta, A., "A-fast-rcnn: Hard positive generation via adversary for object detection," in Procs of the IEEE conf on computer vision and pattern recognition, pp. 2606-2615. 2017.

[25] Xiao, M.,, Kortylewski, A., Wu, R.,Qiao, S., Shen, W., Yuille, A., "Tdapnet: Prototype network with recurrent top-down attention for robust object classification under partial occlusion," preprint arXiv:1909.03879. 2019.

[26] Kortylewski, Adam, Qing Liu, Huiyu Wang, Zhishuai Zhang, and Alan Yuille, "Combining compositional models and deep networks for robust object classification under occlusion," in Procs of the IEEE/CVF winter conf on applications of computer vision, pp. 1333-1341. 2020.

[27] Wang, Angtian, Yihong Sun, Adam Kortylewski, and Alan L. Yuille, "Robust object detection under occlusion with context-aware compositional nets," in Procs of the IEEE/CVF Conf on Computer Vision and Pattern Recognition, pp. 12645-12654. 2020.

[28] Kortylewski, Adam, Ju He, Qing Liu, and Alan L. Yuille, "Compositional convolutional neural networks: A deep architecture with innate robustness to partial occlusion," in Procs of the IEEE/CVF Conf on Computer Vision and Pattern Recognition, pp. 8940-8949. 2020.

[29] Wang, Jianyu, Zhishuai Zhang, Cihang Xie, Vittal Premachandran, and Alan Yuille, "Unsupervised learning of object semantic parts from internal states of cnns by population encoding," arXiv preprint arXiv:1511.06855 (2015).

[30] Boykov, Y., and M-P. Jolly, "Interactive graph cuts for optimal boundary & region segmentation of objects in ND images," in Procs 8th IEEE int conf on computer vision. ICCV 2001, vol. 1, pp. 105-112. IEEE, 2001.

[31] labelImg. https://github.com/tzutalin/labelImg Accessed June 20, 2022

[32] Everingham, Mark, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman, "The pascal visual object classes (voc) challenge," Int journal of computer vision 88 (2009): 303-308.

[33] She, Qi, Fan Feng, Xinyue Hao, Qihan Yang, Chuanlin Lan, Vincenzo Lomonaco, Xuesong Shi et al, "Openloris-object: A robotic vision dataset and benchmark for lifelong deep learning," IEEE int conf on robotics and automation (ICRA), pp. 4767-4773. IEEE, 2020.

[34] Zhang, Shanshan, Rodrigo Benenson, and Bernt Schiele, "Citypersons: A diverse dataset for pedestrian detection," in Procs of the IEEE conf on computer vision and pattern recognition, pp. 3213-3221. 2017.

[35] Krizhevsky, A. and Hinton, G, "Learning multiple layers of features from tiny images,". 2009.

[36] Wah, Catherine, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie, "The caltech-ucsd birds-200-2011 dataset,". 2011.

[37] Loghmani, Mohammad Reza, Barbara Caputo, and Markus Vincze, "Recognizing objects in-the-wild: Where do we stand?," in 2018 IEEE International Conference on Robotics and Automation (ICRA), pp. 2170-2177. IEEE, 2018.

[38] Lin, Tsung-Yi, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie, "Feature pyramid networks for object detection," in Procs of the IEEE conference on computer vision and pattern recognition, pp. 2117-2125. 2017.

[39] Kingma, Diederik P., and Jimmy Ba, "Adam: A method for stochastic optimization," arXiv preprint arXiv:1412.6980. 2014.

[40] Felzenszwalb, PF., Huttenlocher, DP., "Efficient graph-based image segmentation," Int journal of computer vision 59, pp. 167-181. 2004.

[41] Simonyan, Karen, and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv preprint arXiv:1409.1556. 2014.

[42] He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in Procs of the IEEE conference on computer vision and pattern recognition, pp. 770-778. 2016.

[43] Dosovitskiy, Alexey, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani et al, "An image is worth 16x16 words: Transformers for image recognition at scale," arXiv preprint arXiv:2010.11929. 2020.