

Why Object Detectors Fail: Investigating the Influence of the Dataset

Dimity Miller^{1,2}, Georgia Goode¹, Callum Bennie¹, Peyman Moghadam² and Raja Jurdak¹

¹ Queensland University of Technology, ² Robotics and Autonomous Systems, Data61, CSIRO

Abstract

A false negative in object detection describes an object that was not correctly localised and classified by a detector. In prior work, we introduced five ‘false negative mechanisms’ that identify the specific component inside the detector architecture that failed to detect the object. Using these mechanisms, we explore how different computer vision datasets and their inherent characteristics can influence object detector failures. Specifically, we investigate the false negative mechanisms of Faster R-CNN and RetinaNet across five computer vision datasets, namely Microsoft COCO, Pascal VOC, ExDark, ObjectNet, and COD10K. Our results show that object size and class influence the false negative mechanisms of object detectors. We also show that comparing the false negative mechanisms of a single object class across different datasets can highlight potentially unknown biases in datasets.

1. Introduction

Object detection is the task of identifying target objects in an image, describing objects in terms of *where* they are and *what* they are. It has become a significant topic of interest within the computer vision community, with regular advances in model architecture and training paradigms producing detectors that perform increasingly well on benchmark datasets such as COCO [11]. However, object detectors are still prone to failures, particularly when tested on challenging data, e.g. poor lighting [23], atypical object viewpoints [23], object occlusions [26], and unknown objects [14, 16, 17].

One specific type of failure in object detection is a false negative. False negatives occur when an object detector does not accurately detect an object that is present – either by silently failing to detect the object’s presence entirely, or by miscalculating the object’s location or classification. We

Dimity Miller acknowledges continued support from CSIRO’s Machine Learning and Artificial Intelligence Future Science Platform (MLAI FSP). Georgia Goode and Callum Bennie acknowledge support from the QUT Vacation and Research Experience Scheme. Contact: d24.miller@qut.edu.au

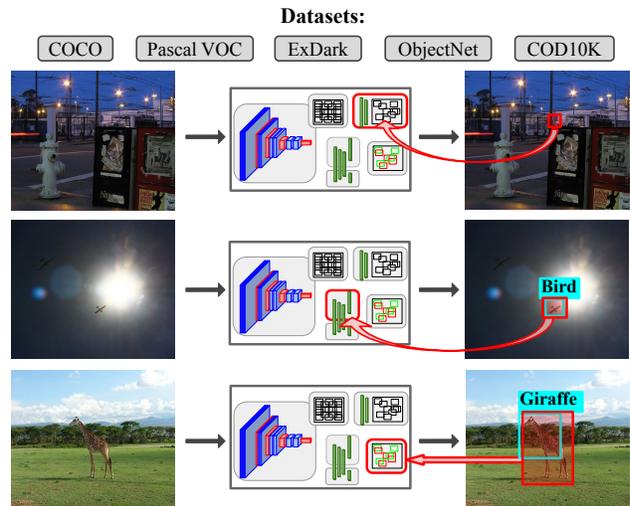


Figure 1. False negatives occur when an object detector fails to accurately describe an object’s location or classification. We investigate the phenomena of false negatives across five computer vision datasets, comparing the false negative mechanisms elicited by each dataset. To identify the false negative mechanism, we look inside an object detector to find the component of the architecture that failed to detect the object – for example, the detector’s Region Proposal Network failing to propose the object as a region (top image), the detector’s classifier confusing the object for an incorrect class (middle image), or a poorly localised detection suppressing a correctly localised detection during the detector’s NMS process (bottom image).

show some examples of this in Figure 1. In prior work [15], we introduced a set of five ‘false negative mechanisms’. Contrary to previous work that described false negatives based only on the output of an object detector [2] – treating the detector itself as a black box – we instead examined *inside* the architecture of object detectors for the source of failure. Using this approach, we proposed an algorithm for identifying the specific component of the detector architecture that failed and why.

This paper investigates how different datasets influence the false negative mechanisms produced by an object detector. Focusing on two-stage and one-stage anchor-box object detector architectures [10, 22], we quantify detector false

negative mechanisms across five computer vision datasets – COCO [11], Pascal VOC [5], ExDark [13], ObjectNet [1], and COD10k [6]. We specifically analyse the changes in distribution of false negative mechanisms across the different datasets, as well as how dataset properties such as object classes and object size influence the false negative mechanisms elicited.

2. Background and Related Work

Quantifying Object Detection Performance On a dataset of images, the primary metric used to summarise object detection performance is mean Average Precision (mAP) [5, 11]. mAP is the mean of the Average Precision (AP) for each target object class, where AP is the area under a precision-recall curve. The precision-recall curves calculates the number of ‘true positive’ detections, ‘false positive’ detections, and ‘false negative’ objects for each target class. A detection is a true positive if it localises and correctly classifies an object that has not already been detected by a previous detection. Localisation is measured by the Intersection-over-Union (IoU) between the object bounding box and the detection predicted box, and must be greater than a specified threshold – typically an IoU of at least 0.5 is required [5, 11]. If the detection does not meet this threshold with any undetected objects of its predicted class, it is a false positive. A false negative describes an object that was not associated with a detection during the evaluation – *i.e.* there were no detections with the correct class label and an IoU greater than 0.5.

Categorising Object Detection Failures As established above, the primary metric assessing object detection performance is mAP. With the mAP metric, failures in object detection can only be coarsely described as false positives or false negatives. To further understand *why* a detection is considered a failure, Hoiem *et al.* [8] introduced a categorisation for false positive detections.

More recently, Bolya *et al.* [2] built upon this categorisation to include categories describing false negative objects, also improving the generalisability of the categorisation to different datasets. Their work introduces TIDE, which describes six categories for failures in object detectors – classification errors (Cls), localisation errors (Loc), classification and localisation errors (Cls+Loc), duplicate errors, background errors, and missed ground-truth errors (Missed GT). These failure categories can represent both false positives and false negatives – in fact, a false negative failure can be a Cls, Loc, Cls+Loc, or Missed GT error. TIDE [2] categorises false negatives by asking the question – why didn’t the predicted detections capture the false negative object? This places the emphasis on the output of the detector, and treats the detector as a black box.

In contrast, the false negative mechanisms introduced in our prior work [15] – proposal process mechanism, regressor mechanism, interclass classification mechanism, background classification mechanism, and classifier calibration mechanism – focus on the detection process occurring *inside* the object detector. We describe these false negative mechanisms, and how to quantify them, in Section 3. Compared to TIDE, this approach explicitly identifies the specific component of the detector that failed.

Detecting False Negatives in Object Detection In the last few years, a number of works have emerged introducing techniques that identify when an object detector fails to detect an object [18, 19, 25]. These works focused on exploiting a variety of signals for detecting false negatives – temporal or stereo camera detection inconsistencies [19], learning underlying biases in the objects that produce false negatives [25], and relying on hand-crafted indicators of false negatives in a detector’s feature maps [18]. However, all works examined false negatives from the perspective of autonomous driving, only examining datasets relevant for this context [18, 19, 25].

3. Object Detector False Negative Mechanisms

In this section, we detail the five false negative mechanisms introduced in our prior work, along with how they can be identified [15]. We refer the reader to the prior work [15] for additional explanation. The introduced false negative mechanisms focus on the role of different architecture components as observed from existing two-stage and one-stage anchor-box detectors, including Faster R-CNN [22], Cascade R-CNN [4], SSD [12], YOLOv2 [20], YOLOv3 [21], and RetinaNet [10].

Proposal Process False Negative Mechanism: After extracting feature maps from an image, the detector identifies image regions that may contain objects. This is handled differently between two-stage and one-stage anchor-box architectures. Two-stage architectures rely on fully convolutional networks, known as Region Proposal Networks (RPN) [4, 22], to propose a set of image regions that may contain an object. In this step, two-stage detectors define a dense grid of ‘anchors’ over the computed feature maps and predict the ‘objectness’ scores and regressed anchor boundaries for each anchor. The anchors with the top-k highest objectness scores, and their regressed bounding boxes, are then output from the RPN as the initial set of object proposals. Similar to two-stage detectors, one-stage anchor-box architectures segment feature maps into a grid of anchor boxes [10, 12, 20, 21]. However, rather than using an RPN to filter these anchor boxes into region proposals, they instead treat *every* anchor box as a region proposal.

A false negative can be produced if no object proposals localise a given target object. We refer to this as a ‘proposal process’ false negative mechanism. Without a proposal for the object’s image region, later components in the detector will not receive the features necessary to accurately classify or localise the object. We identify this mechanism by computing the IoU between the target object bounding box and all object proposals, where an IoU of at least θ_{loc} is required for an object to be considered localised. In a one-stage anchor-box architecture, this is far less likely to occur than in a two-stage architecture. For example, given a 640x480 image, a two-stage Faster R-CNN produces 1000 RPN object proposals [22], whereas the one-stage RetinaNet produces 163206 anchor boxes [10].

Regressor False Negative Mechanism: To localise objects in the image, the object proposals computed in the proposal process are extracted from the feature maps and passed into the bounding box regressor. The bounding box regressor is typically a small Fully Convolutional Network [4, 10, 12, 20–22], and predicts changes, or offsets, to the initially proposed bounding box from the prior step.

Assuming an object proposal localises an object, the regressor should predict offsets that improve this localisation. However, the regressor may produce offsets that reduce the overlap of the bounding box with a target object. We refer to this as a ‘regressor’ false negative mechanism. We identify this mechanism when there are no regressed bounding boxes with an IoU of at least θ_{loc} with the target object box, despite the presence of a proposal box that had localised the object.

Interclass Classification and Background Classification False Negative Mechanisms: Similar to the regressor, the object classifier is typically a small Fully Convolutional Network that intakes the features of each object proposal [4, 10, 12, 20–22]. The classifier determines which class an object proposal belongs to – distinguishing between the different target object classes and the background class. For every object proposal, the classifier outputs a confidence score for each target class and the background class, where high confidence scores indicate the class the proposal belongs to. For all detectors, a minimum confidence score is specified as a parameter of the detector, θ_{cls} . All proposals with a target class score above this minimum score threshold form a detection, which contains the target class label, associated confidence score, and the refined bounding box from the regressor.

While an object proposal may have localised an object, the classifier can produce a false negative if it does not assign a classification score above θ_{cls} to the correct target object class. In some cases, the classifier can confuse an object for another incorrect target class. We refer to this as an ‘in-

terclass classification’ false negative mechanism. Given the classification scores of boxes that have localised the target object, we identify this mechanism when the classifier assigned any incorrect target classes a confidence score above θ_{cls} . In other cases, the classifier can fail by misclassifying all proposals of the object as belonging to the background of the image, i.e. the ‘background’ class. Assuming an interclass classification mechanism has not been identified, we identify this ‘background classification’ false negative mechanism when all localised boxes of the object are assigned a background class confidence score above θ_{cls} .

Classifier Calibration False Negative Mechanism: After the detector regressor and classifier refine the object proposals into detections, there is often multiple overlapping detections that jointly detect a single object with the same predicted class label. Non-Maximum Suppression (NMS) is an algorithm commonly used by object detectors to suppress duplicate detections of a single object [4, 10, 12, 20–22]. In a class-wise manner, NMS removes low confidence detections that highly overlap with higher confidence detections. The process relies on the best localised bounding boxes having the highest classification confidence scores. If this assumption is not satisfied, a false negative can be introduced when a correctly localised detection is suppressed by a detection that has not localised the object, i.e. IoU less than θ_{loc} . We refer to this as a ‘classifier calibration’ false negative mechanism, as it is ultimately the fault of the calibration of the classifier’s confidence scores – detections that better localise an object should be coupled with higher confidence scores from the classifier. Prior to NMS, we identify this false negative mechanism when there is a regressed bounding box that localises the target object with an IoU greater than θ_{loc} (correctly localised), that also has a confidence score for the correct target class above θ_{cls} (correctly classified), but it is not produced as an output detection.

4. Experimental Setup

Following the approach of our prior work [15], we investigate the false negative mechanisms of two object detectors: Faster R-CNN [22], and RetinaNet [10]. For both detectors, we use the public implementation available via `detectron2` [24] and a ResNet50 [7] with Feature Pyramid Network [9] backbone. For all experiments, detectors were trained on COCO [11]. False negative objects are identified when there is no detection with a minimum IoU θ_{loc} of 0.5 and a minimum confidence score θ_{cls} of 0.3 for the correct object class.



Figure 2. Example images from the five datasets tested.

4.1. Datasets

To investigate the role of datasets on detector false negative mechanisms, we test five computer vision datasets: Microsoft COCO [11], Pascal VOC [5], ExDark [13], COD10k [6], and ObjectNet [1]. We detail each dataset and the testing protocol below. We also show example images from each dataset in Figure 2.

Microsoft COCO [11]: is the predominant benchmark dataset for evaluating object detectors. We evaluate with the ‘val2017’ split of the dataset, which contains 4,952 images featuring objects from 80 different object classes.

Pascal VOC dataset [5]: was the prevailing object detection benchmark prior to COCO. Pascal VOC has 20 object classes, all of which overlap with the COCO dataset classes – plane, bicycle, bird, boat, bottle, bus, car, cat, chair, cow, dining table, dog, horse, motorbike, person, potted plant, sheep, sofa, train and tv monitor. We evaluate with all 17,125 images from the 2012 release of the dataset.

Exclusively Dark (ExDark) [13]: is a detection dataset containing 7,363 images from low-light indoor and outdoor environments. It includes 12 object classes – bicycle, boat, bottle, bus, car, cat, chair, cup, dog, motorbike, person, and table – which all are present in the COCO class list.

Camouflaged Object Detection (COD10k) [6]: is an object detection dataset designed specifically to benchmark detection of camouflaged objects. It contains a total of 10,000 images with 78 object classes across aquatic, flying, amphibian and terrestrial categories. We test all images with the 11 object classes overlapping with the COCO class list – cat, dog, sheep, giraffe, human, bird, frogmouth, heron, mockingbird, owl, and duck – which makes a total of 932 images.

ObjectNet [1]: is a large real-world dataset for object recognition where object backgrounds, rotations, and viewpoints are random. The entire dataset contains 50,000 images and 313 object classes. We test with images from

Table 1. False negative rates on each dataset.

	# Objects	False Negative Rate (#)	
		Faster R-CNN	RetinaNet
COCO [11]	36335	28.8% (10463)	32.7% (11872)
VOC [5]	40138	11.5% (4621)	12.1% (4874)
ExDark [13]	23710	28.5% (6749)	30.3% (7181)
ObjectNet [1]	2777	52.7% (1464)	45.4% (1260)
COD10K [6]	1124	37.5% (421)	38.5% (433)

17 classes that overlap with the COCO class list – bicycle, bench, umbrella, tie, baseball bat, baseball glove, tennis racket, banana, orange, chair, laptop, remote, keyboard, cell phone, microwave, book, and vase – which makes a total of 2,777 images. While the original dataset only included an object label for each image, [3] released bounding box annotations for the labelled object in each image, and we use these in our evaluation.

5. Results

In this section, we investigate how different datasets induce different false negative mechanisms from an object detector, specifically observing how the class and size of the dataset’s objects influence the failures of the detector.

5.1. Overall Dataset False Negative Mechanisms

First, we compare the overall distribution of false negative mechanisms for each dataset with Faster R-CNN and RetinaNet in Figure 3. We also show the false negative rate for each dataset in Table 1.

Consistently across the datasets, most false negatives are due to the background classification mechanism (with the exception of ObjectNet, see below) and negligible false negatives are due to the regressor mechanism. Interestingly, Pascal VOC and ExDark show very similar distributions of

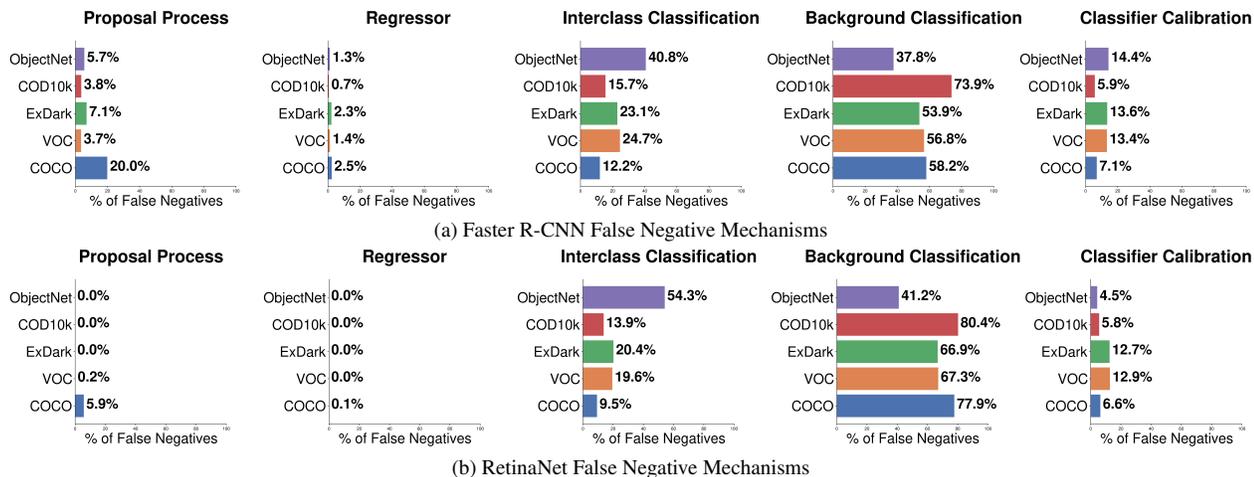


Figure 3. The influence of the datasets on the false negative mechanisms produced by Faster R-CNN (top) and RetinaNet (bottom).

false negative mechanisms. While ExDark features more challenging illumination conditions than Pascal VOC, this appears to instead have only induced an overall greater false negative rate. We hypothesise that the similar mechanism distributions of these datasets is due to the high overlap between their object classes.

There are also a number of noticeable discrepancies across the datasets. Compared with the other datasets, COCO elicits a greater number of proposal process false negatives – in the following section, we link this to COCO’s greater prevalence of small objects. For COD10k, there is a greater proportion of background classification false negatives compared to other datasets. This is a reflection of the ‘camouflage’ nature of the dataset, where objects blend into the textures of the image background and pose a challenge for the detector’s classifier.

Of all datasets, ObjectNet is the most dissimilar in the distribution of false negative mechanisms. ObjectNet is traditionally an object recognition dataset rather than object detection dataset [1] – it features a single target object very prominently in the image (reducing the likelihood of background classification), while the unusual placements and rotations of objects may lead to increased interclass classification mechanisms.

5.2. Object Size and False Negative Mechanisms

In Figure 4, we compare the influence of the object size on the false negative mechanism elicited from an object detector. For both detector architectures and across nearly all datasets, proposal process and regressor false negative mechanisms are characterised by small object sizes – this is intuitive, as small objects in an image may be more difficult for a Region Proposal Network to distinguish from the image background or may not be represented by the default detector anchor boxes. This also explains the greater

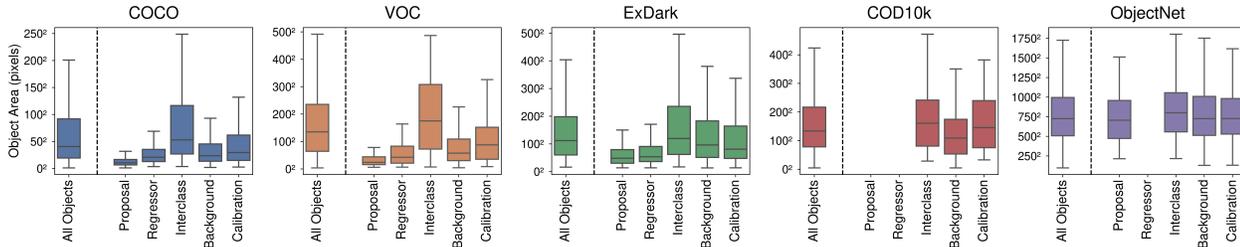
number of proposal process false negatives for the detectors on the COCO dataset, as when COCO was introduced, a key distinguishing factor was an increase in small object instances [11].

Interestingly, all datasets show a relationship between larger objects and the interclass classification mechanism. We hypothesise that larger objects (and thus larger bounding boxes) are more likely to overlap with other objects in the image – for example, a table annotation often overlaps with chairs, bowls, cups, etc. – which may result in image regions with features for conflicting object classes, thus leading to an interclass classification false negative.

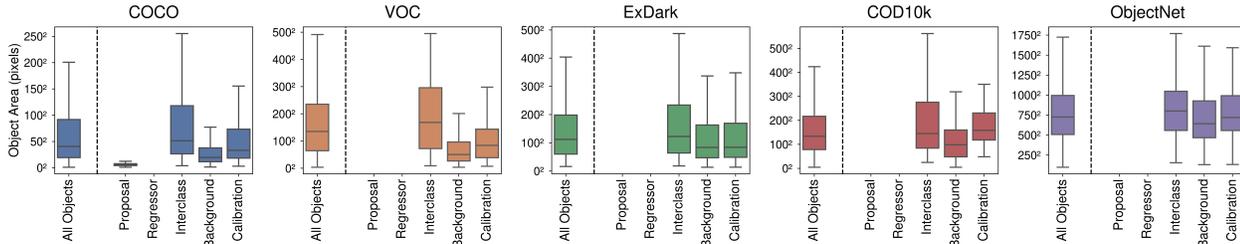
5.3. Object Class and False Negative Mechanisms

In Figure 5, we visualise the false negative mechanisms elicited by each class in the COCO dataset, where differences in colour indicate differences in the class-specific false negative mechanism distributions. We also indicate and sort classes from left to right based on number of false negatives – it’s worth noting that results calculated for classes with only a handful of false negatives (to the right of the figure) may not be statistically significant.

While most object classes follow the overall trend of COCO’s false negative mechanisms, there is evidence of some class-specific false negative behaviour from the detectors. For example, while most Faster R-CNN false negatives on COCO are due to background classification mechanisms, for class ‘bench’, the majority of false negatives are due to the proposal process mechanism, and for class ‘sheep’, most false negatives are due to the interclass classification mechanism. Contrary to the majority of other results we present, these class-specific trends do not necessarily generalise across the detector architectures – for RetinaNet, false negatives from class ‘sheep’ have mostly background class mechanisms, and few interclass classification mechanisms.

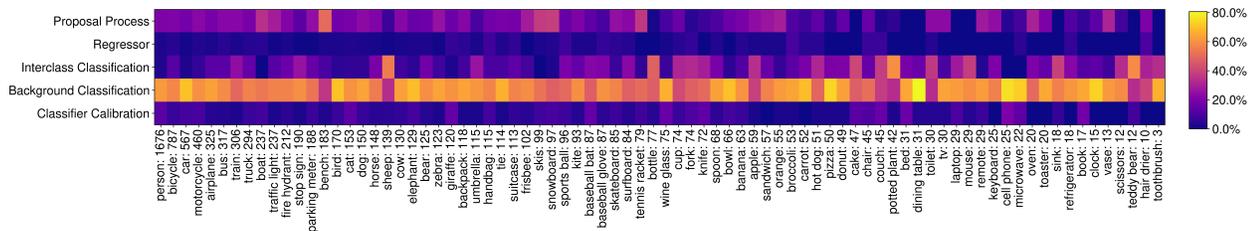


(a) Boxplots of the distribution of object pixel area for each Faster R-CNN false negative mechanism.

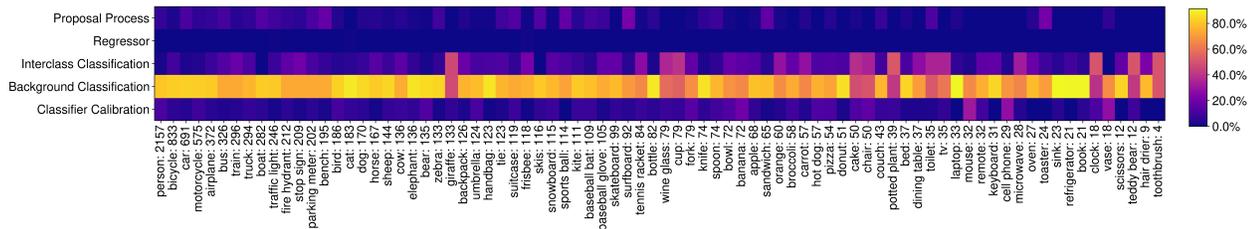


(b) Boxplots of the distribution of object pixel area for each RetinaNet false negative mechanism.

Figure 4. The influence of false negative object size on the false negative mechanisms produced by Faster R-CNN (top) and RetinaNet (bottom). For each dataset, we also show the total distribution of object sizes. Note that outliers are not shown in the boxplots.



(a) Faster R-CNN class-specific false negative mechanisms on the COCO dataset.



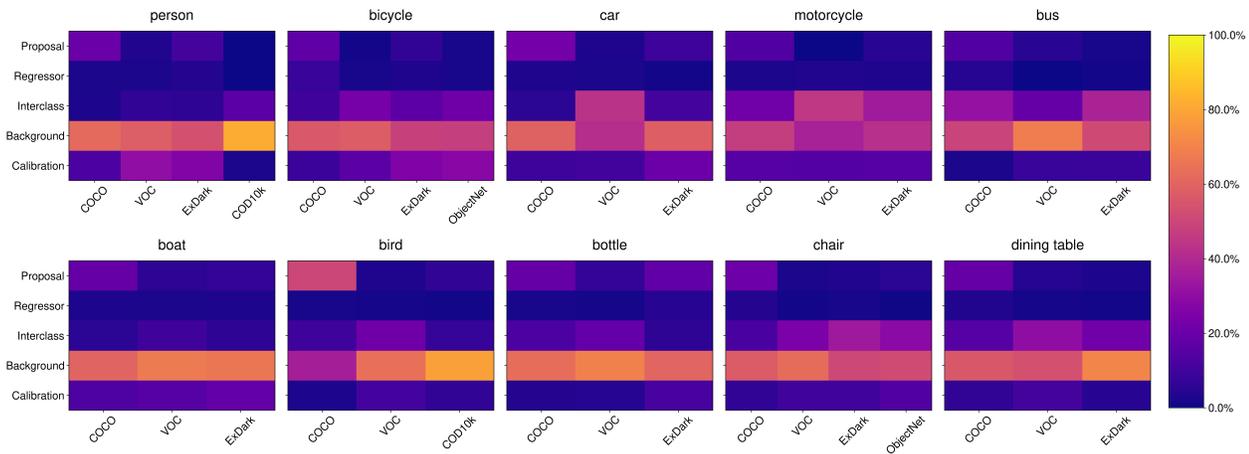
(b) RetinaNet class-specific false negative mechanisms on the COCO dataset.

Figure 5. The influence of the object class on the false negative mechanisms produced by Faster R-CNN (top) and RetinaNet (bottom) when tested on COCO. For each class, we show the distribution of false negatives across the five mechanisms, as well as the total number of false negatives reported.

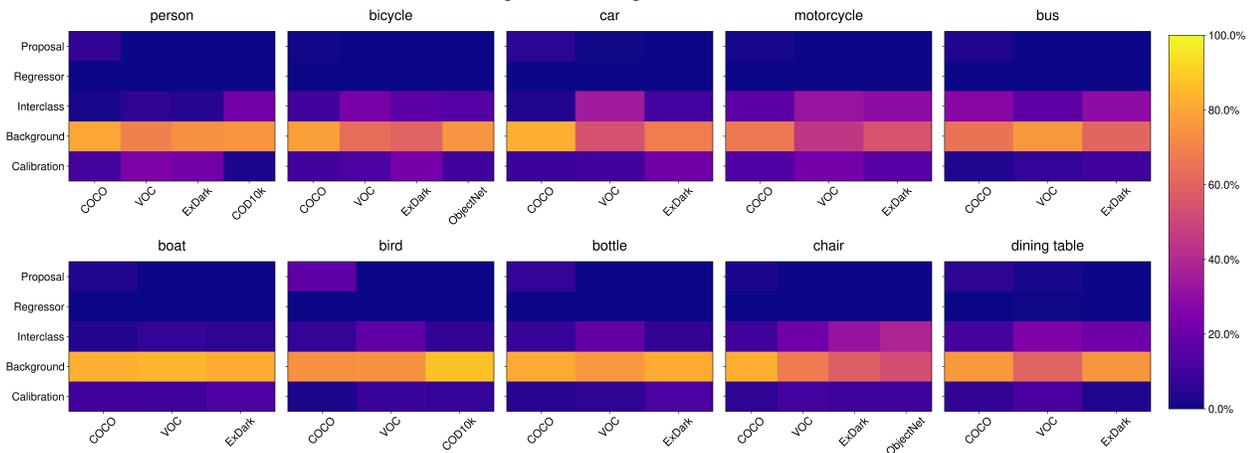
5.4. Class-specific False Negative Mechanisms Across Datasets

Given the observed class-specific distributions of false negative mechanisms, we then compare these class-specific distributions across the datasets for shared object classes in Figure 6. We find that some object classes consistently induce a specific false negative mechanism distribution, regardless of the dataset – e.g. class ‘boat’, ‘bottle’ and ‘chair’ in Faster R-CNN, and class ‘bird’, ‘boat’, ‘bottle’, and ‘per-

son’ in RetinaNet. However, other classes show dataset-dependent differences to the false negative mechanism distributions. In some cases, this is consistent with the properties of the dataset – e.g. for Faster R-CNN, class ‘person’ in COD10k has a greater number of background classification mechanisms, as expected given the camouflage nature of the dataset. However, other differences in class distributions are not so clearly linked to dataset properties – e.g. for both detectors, the Pascal VOC dataset induces a greater



(a) Faster R-CNN class-specific false negative mechanisms across the datasets.



(b) RetinaNet class-specific false negative mechanisms across the datasets.

Figure 6. The influence of the dataset on the class-specific false negative mechanisms produced by Faster R-CNN (top) and RetinaNet (bottom). For each class and dataset (with minimum 40 false negatives per class and dataset), we show the distribution of false negatives across the five mechanisms.

number of background classification mechanisms for class ‘bus’ than COCO or ExDark. This may indicate the presence of other dataset biases that are not clearly identified or easy to observe from the data.

6. Conclusion

In this paper, we investigated how the properties and characteristics of different computer vision datasets influence the false negatives of an object detector. Our results highlighted a number of insights into the relationship between data properties and false negatives from a detector – namely the relationship between object size and false negative mechanisms, and the presence of class-specific trends in the false negative mechanisms produced by a detector.

The overarching goal of this work is to enable future research into object detectors that are robust to false negatives. Given our observations that the testing data properties in-

fluence why detectors fail, we conclude that future development of object detectors should be tightly coupled with an understanding of the testing data characteristics. Additionally, we showed that comparing the false negative mechanisms for a single class can reveal discrepancies across datasets. Some of these discrepancies may be linked to the known and reported characteristics of the dataset (e.g. a camouflaged dataset eliciting more background classification false negatives), however others are not immediately intuitive or explainable. In future, this could be used as a tool for further investigating the presence of unknown bias in datasets.

References

- [1] Andrei Barbu, David Mayo, Julian Alverio, William Luo, Christopher Wang, Dan Gutfreund, Josh Tenenbaum, and Boris Katz. Objectnet: A large-scale bias-controlled dataset

- for pushing the limits of object recognition models. *Advances in neural information processing systems*, 32, 2019. 2, 4, 5
- [2] Daniel Bolya, Sean Foley, James Hays, and Judy Hoffman. Tide: A general toolbox for identifying object detection errors. In *European Conference on Computer Vision*, pages 558–573. Springer, 2020. 1, 2
- [3] Ali Borji. Objectnet dataset: Reanalysis and correction. *arXiv preprint arXiv:2004.02042*, 2020. 4
- [4] Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: Diving into high quality object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6154–6162, 2018. 2, 3
- [5] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010. 2, 4
- [6] Deng-Ping Fan, Ge-Peng Ji, Guolei Sun, Ming-Ming Cheng, Jianbing Shen, and Ling Shao. Camouflaged object detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2, 4
- [7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 3
- [8] Derek Hoiem, Yodsawalai Chodpathumwan, and Qieyun Dai. Diagnosing error in object detectors. In *European conference on computer vision*, pages 340–353. Springer, 2012. 2
- [9] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017. 3
- [10] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017. 1, 2, 3
- [11] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*. Springer, 2014. 1, 2, 3, 4, 5
- [12] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016. 2, 3
- [13] Yuen Peng Loh and Chee Seng Chan. Getting to know low-light images with the exclusively dark dataset. *Computer Vision and Image Understanding*, 178:30–42, 2019. 2, 4
- [14] Dimity Miller, Feras Dayoub, Michael Milford, and Niko Sünderhauf. Evaluating merging strategies for sampling-based uncertainty techniques in object detection. In *International Conference on Robotics and Automation*, pages 2348–2354. IEEE, 2019. 1
- [15] Dimity Miller, Peyman Moghadam, Mark Cox, Matt Wildie, and Raja Jurdak. What’s in the Black Box? The False Negative Mechanisms Inside Object Detectors. *arXiv preprint arXiv:2203.07662*, 2022. 1, 2, 3
- [16] Dimity Miller, Lachlan Nicholson, Feras Dayoub, and Niko Sünderhauf. Dropout sampling for robust object detection in open-set conditions. In *IEEE International Conference on Robotics and Automation*, pages 1–7, 2018. 1
- [17] Dimity Miller, Niko Sünderhauf, Michael Milford, and Feras Dayoub. Uncertainty for identifying open-set errors in visual object detection. *IEEE Robotics and Automation Letters*, 7(1):215–222, 2021. 1
- [18] Quazi Marufur Rahman, Niko Sünderhauf, and Feras Dayoub. Did you miss the sign? a false negative alarm system for traffic sign detectors. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3748–3753. IEEE, 2019. 2
- [19] Manikandasriram Srinivasan Ramanagopal, Cyrus Anderson, Ram Vasudevan, and Matthew Johnson-Roberson. Failing to learn: Autonomously identifying perception failures for self-driving cars. *IEEE Robotics and Automation Letters*, 3(4):3860–3867, 2018. 2
- [20] Joseph Redmon and Ali Farhadi. Yolo9000: better, faster, stronger. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7263–7271, 2017. 2, 3
- [21] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018. 2, 3
- [22] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *arXiv preprint arXiv:1506.01497*, 2015. 1, 2, 3
- [23] Niko Sünderhauf, Feras Dayoub, David Hall, John Skinner, Haoyang Zhang, Gustavo Carneiro, and Peter Corke. A probabilistic challenge for object detection. *Nature Machine Intelligence*, 1(9):443–443, 2019. 1
- [24] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. <https://github.com/facebookresearch/detectron2>, 2019. 3
- [25] Qinghua Yang, Hui Chen, Zhe Chen, and Junzhe Su. Introspective false negative prediction for black-box object detectors in autonomous driving. *Sensors*, 21(8):2819, 2021. 2
- [26] Shifeng Zhang, Yiliang Xie, Jun Wan, Hansheng Xia, Stan Z Li, and Guodong Guo. Widerperson: A diverse dataset for dense pedestrian detection in the wild. *IEEE Transactions on Multimedia*, 22(2):380–393, 2019. 1