IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 8 February 2023, accepted 10 March 2023, date of publication 13 March 2023, date of current version 16 March 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3256723

RESEARCH ARTICLE

Quantifying the Effects of Ground Truth Annotation Quality on Object Detection and Instance Segmentation Performance

CATHAOIR AGNEW, CIARÁN EISING, (Member, IEEE), PATRICK DENNY, (Member, IEEE), ANTHONY SCANLAN, PEPIJN VAN DE VEN, AND EOIN M. GRUA

Data-Driven Computer Engineering (D^2iCE) Group, Department of Electronic and Computer Engineering, University of Limerick, Limerick, V94 T9PX Ireland
CONFIRM Centre for Smart Manufacturing, University of Limerick, Limerick, V94 T9PX Ireland

Corresponding author: Cathaoir Agnew (cathaoir.agnew@ul.ie)

This work was supported by the Science Foundation Ireland (SFI) under Grant 16/RC/3918 (CONFIRM Centre).

ABSTRACT Fully-supervised object detection and instance segmentation models have accomplished notable results on large-scale computer vision benchmark datasets. However, fully-supervised machine learning algorithms' performances are immensely dependent on the quality of the training data. Preparing computer vision datasets for object detection and instance segmentation is a labor-intensive task requiring each instance in an image to be annotated. In practice, this often results in the quality of bounding box and polygon mask annotations being suboptimal. This paper quantifies empirically the ground truth annotation quality and COCO's mean average precision (mAP) performance by introducing two separate noise measures, uniform and radial, into the ground truth bounding box and polygon mask annotations for the COCO and Cityscapes datasets. Mask-RCNN models are trained on various levels of noise measures to investigate the performance of each level of noise. The results showed degradation of mAP as the level of both noise measures increased. For object detection and instance segmentation respectively, using the highest level of noise measure resulted in a mAP degradation of 0.185 & 0.208 for uniform noise with reductions of 0.118 & 0.064 for radial noise on the COCO dataset. As for the Cityscapes datasets, reductions of mAP performance of 0.147 & 0.142 for uniform noise and 0.101 & 0.033 for radial noise were recorded. Furthermore, a decrease in average precision is seen across all classes, with the exception of the class motorcycle. The reductions between classes vary, indicating the effects of annotation uncertainty are class-dependent.

INDEX TERMS Annotation uncertainty, computer vision, instance segmentation, object detection, supervised learning.

I. INTRODUCTION

Following AlexNet's success in ImageNet Large Scale Visual Recognition Competition (ILSVRC) in 2012 [1], a great deal of work has gone into refining deep neural network architectures for computer vision tasks. This has led to various convolutional neural network-based architectures developed for computer vision tasks, such as SegNet [2], Mask RCNN [3] and YOLO [4]. The advancements are not limited to just convolutional neural network-based architectures, as progress has been made with Vision Transformers Models [5] and Graph Convolutional Networks [6]. Sun et al. credited deep learning's recent success in computer vision tasks to three primary aspects [7]. To begin with, graphics processing units and parallel processing are becoming more widely available, allowing for the training of bigger models [7]. Following this, there have been technical improvements in network architecture design, parameter initialization, and training methodologies [7], [8]. Finally, the availability of vast and expanding datasets is increasing [7].

The associate editor coordinating the review of this manuscript and approving it for publication was Victor Sanchez.

We believe another key factor for deep learning's recent success is the availability and facility of deep learning frameworks such as TensorFlow [9], PyTorch [10] and Apache MXNet [11], which enabled deep learning to become more accessible to the broader research community. The recent advancements in deep learning methodologies for computer vision tasks have yielded momentous technologies in many domains such as intelligent transportation systems [12], [13], sports analytics [14], [15] and medical imaging [16], [17]. These advancements are not restricted to RGB imagery, with advances in infrared [18] and hyperspectral imagery [19].

Neural networks' performance for supervised computer vision tasks relies on the data they are trained on. This includes the annotations that are utilized as ground truth for supervised learning algorithms. Sun et al. found that performance on vision tasks improves logarithmically as the training dataset size increases [7]. Due to the large quantity of data that is regularly needed, the process of annotating datasets for computer vision-supervised learning tasks is time intensive. For example, it required approximately 60,000 worker hours to annotate the Common Objects in Context (COCO) dataset [20]. For object detection, bounding boxes must be manually annotated over the classes of interest for the entire dataset. Employing a crowd-sourcing method [21] that is optimized for bounding box annotation, each annotation in the ImageNet Visual Recognition dataset [22] took around 35s to annotate. For instance segmentation, a polygon mask must be outlined around each class of interest for the dataset. Polygon annotations are more accurate than bounding boxes but are also more laborious. This is reflected by the annotation time estimated to be 79.2s per polygon mask for the popular COCO dataset [20].

The importance of ground truth annotation quality has been acknowledged for computer vision tasks in the literature, with methods being developed attempting to rectify and account for noisy labels in computer vision tasks such as object detection [23], [24] and image classification [25], [26], [27], [28]. To the authors' knowledge, there is limited literature attempting to quantify the effects the ground truth bounding box and polygon mask annotation quality have on object detection and instance segmentation performance.

The main contribution of this paper is to quantify empirically the annotation quality levels and their effects on mAP [29] by introducing noise into both bounding boxes and polygon masks on a subset of the COCO dataset. To the authors' knowledge, this work is the first to investigate annotation uncertainty for three different aspects. To begin with, this work is the first to investigate annotation uncertainty for instance segmentation. Secondly, for object detection, this work introduces noise to the scale of pixel distance, allowing a finer scale of annotation uncertainty which may be more representative of annotation uncertainty seen in practice. Finally, the effects of annotation uncertainty on each individual class' average precision (AP) performance are investigated to provide further insight. Quantifying the relationship between annotation quality and performance will yield helpful insight into the trade-off between annotation quality and the time and cost associated with such annotation quality. This information in turn will allow for informed decision-making and enables the tailoring of annotations to the use case of the application.

The paper is structured as follows. In Section II an overview of related work is discussed. Then, in Section III, an explanation for how annotation uncertainty is modeled for this work is given. This is followed by a description of the experiment in Section IV and a presentation of experimental results in Section V. In Section VI, these results are analyzed and discussed. Lastly, in Section VII, the conclusions of this work are summarised.

II. RELATED WORK

Taran et al. used Cityscapes [31] fine and coarse annotated images to investigate the effects ground truth annotation quality has on semantic image segmentation performance of traffic conditions [30]. The authors explored two situations, firstly using the fine ground truth annotations for both training and inferencing; secondly training with the fine ground truth annotations but inferencing on the coarse ground truth annotations. PSPNet [32] was used for the semantic segmentation model and a subset of the Cityscapes dataset was used for the analysis, which included data from 3 cities and the following classes; road, car, pedestrian, traffic lights, and signs. Using mean intersection over union (IoU) as the metric of interest, the authors found the IoU values for coarse ground truth annotated images in general, were higher than those for fine ground truth annotated images. In light of the results of comparing fine and coarse ground truth annotations, the authors suggest that deep neural networks could be utilized to generate coarse ground truth annotated datasets, that can be modified and used to fine-tune the pre-trained models for the specific application.

A study by Mullen Jr et al. [33] compared annotation types and their effects on object detection performance on the Overhead Imagery Research Dataset (OIRDS) [34]. Three annotation types were considered for the analysis to detect cars from the OIRDS; polygon masks, bounding boxes, and target centroids. A modified version of the Overfeat [35] network architecture was used for the analysis. A Receiver Operating Characteristic (ROC) curve assessed at all pixel locations along with the area under the curve (AUC) was calculated for the 3 annotation types. The results showed polygon mask annotations scored marginally better AUC than the other two annotation types. The authors concluded when putting together a dataset for deep learning, comparing annotation types is a key step, as the cost of annotations and the advantages and disadvantages of each annotation type should be considered.

Xu et al. investigated training object detectors with noisy labels [23], including incorrect class labels and imprecise bounding boxes on both PASCAL VOC 2012 [36] and COCO

2017 [20]. Xu et al. proposed Meta-Refine-Net, a meta-learning-based approach to train more robust detectors from noisy labels. In this study, the authors generated imprecise bounding boxes by shifting the original annotations by factors of the bounding boxes' width and height. Category noise was also included by randomly sampling a chosen proportion of objects and modifying the class label to be incorrect. The results showed degradation in mAP for both incorrect class labels and imprecise bounding boxes for all ranges of noise used on both datasets.

Acuna et al. noted there is a substantial amount of label noise in relevant datasets for semantic border prediction [37]. The goal of semantic border prediction is to determine which pixels correspond to object boundaries. The authors presented a simple yet effective thinning layer and loss that can be utilized with boundary detectors, at the time of publishing, to reduce the label noise effects during training. The authors' experiments revealed an improvement of 18.61% in average precision on the CASENet [38] backbone network using the new thinning layer and loss, along with significant improvements in thinning semantic border labels over existing methods on the Semantic Boundaries Dataset [39] and Cityscapes dataset [31].

Rolnick et al. investigated label noise for image classification using deep learning [40]. The following datasets were used in the study; ImageNet [41], MNIST [42] and CIFAR [43]. The authors concluded with 3 key takeaways. Firstly, instead of just memorizing noise, deep neural networks can generalize after training on noisy data. Secondly, given a large enough training set, neural networks can handle a wide range of label noise levels. Lastly, larger batch sizes and downscaling the learning rate can offset the influence of noisy labels on effective batch size.

Whilst the presented literature answers a number of questions related to the influence of annotation quality, the performance degradation for varying levels of annotation uncertainty remains unknown. The objective of this study is to quantify empirically the annotation quality levels for bounding boxes and polygon masks and the effects it has on mAP. Whereas Taran et al. investigated the effects of ground truth annotation quality for semantic segmentation using Cityscapes [31] fine and coarse datasets, the disparity between the fine and coarse datasets does not yield insight into the various levels of annotation uncertainty that may arise in object detection and instance segmentation datasets. A direct comparison between results would also not be feasible due to the difference in annotation methodologies. For semantic segmentation, each individual pixel in an image must be annotated, however, for object detection and instance segmentation, only the objects of interest are annotated in an image. Mullen Jr et al. highlighted the need for exploring different annotation types and the associated costs along with each type, but the effects of annotation noise were not considered in their study. Xu et al. investigated noisy labels, using factors of the bounding boxes' width and height to introduce imprecise bounding boxes. However, in this research, the induced annotation noise used will be constant across object sizes rather than using factors of the original bounding boxes' width and height. It is the authors' belief that keeping the degradation across object sizes constant would yield a more comparable experiment across classes. This research also delves into the effect annotation uncertainty has on each individual class for the ranges of induced noise measures used in these experiments, which Xu et al. did not investigate. Acuna et al. and Rolnick et al. investigated noise in semantic border prediction and image classification respectively, but these results do not fully extend to object detection and instance segmentation due to the difference in annotation methodologies. Our work furthers the investigation of annotation quality and its effect on object detection performance and extends it to instance segmentation.

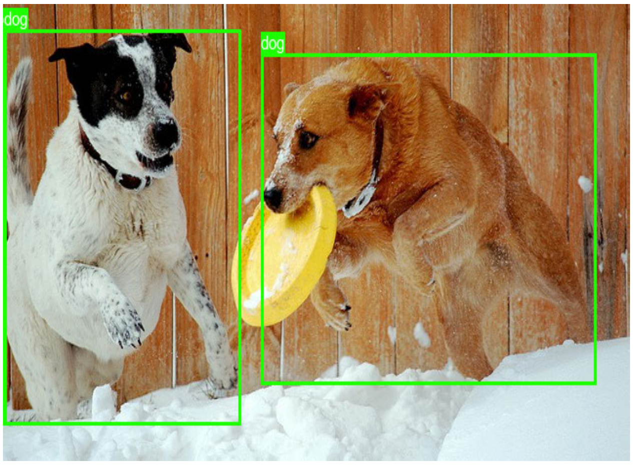

FIGURE 1. Example of bounding box annotation from COCO dataset [20].

III. MODELING ANNOTATION UNCERTAINTY

For supervised learning computer vision tasks, each image requires an associated annotation to be able to learn from. For object detection, bounding boxes must be manually annotated over the classes of interest for each image in the dataset. An example of a bounding box annotation for the class dog in the COCO dataset [20] can be seen in Fig. 1. For instance segmentation, a polygon mask must be outlined around each class of interest for each image in the dataset. An example of a polygon mask annotation for the class dog in the COCO dataset [20] can be seen in Fig. 2. A class label must also be given with each annotated object for object detection and instance segmentation. For this work, the focus of annotation uncertainty is on the polygons and bounding boxes. Class labels have not been tampered with. As such any effect of class label noise inherent in the COCO dataset would be consistent between all experiments.

Two methods for modelling annotation uncertainty were used for this research. Firstly, Shapely's polygon buffer method [44] was used to introduce an approximate uniform

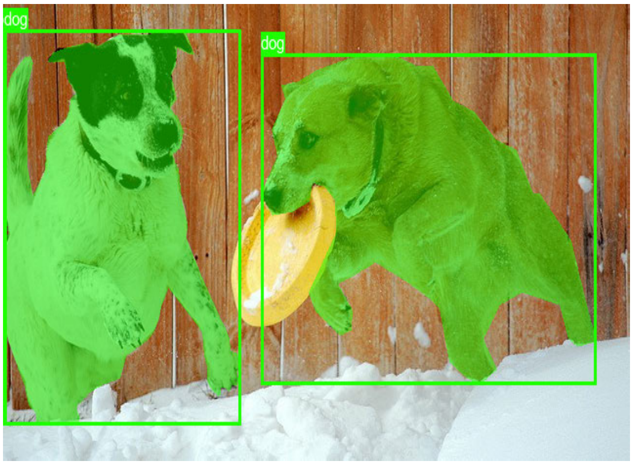

FIGURE 2. Example of bounding box & polygon mask annotation from COCO dataset [20].

noise by expanding the ground truth annotation outwards by an approximate euclidean pixel distance, as seen in Fig. 5. A uniform noise was introduced as a means to set a baseline for annotation uncertainty for these experiments. The pixel distance ranged from the integer values of 1 to 10 inclusive. COCO defines the bounding box annotation with x, y relating to the upper-left coordinates of the bounding box, the width defines the distance the object spans on the x-axis and finally the height defines the distance the object spans on the y-axis. The bounding boxes were updated to include the relevant uniform pixel distance noise according to (1). In Equation (1) width and height are represented by w and h, ϕ is the pixel distance noise used, and finally, x_u, y_u, w_u, h_u represent the new datapoints with uniform noise for the bounding box.

$$\begin{aligned} x_u &= x - \phi \\ y_u &= y - \phi \\ w_u &= w + 2(\phi) \\ h_u &= h + 2(\phi) \end{aligned} \tag{1}$$

Secondly, Gaussian radial noise was added to each vertex of the polygon mask to model annotation uncertainty. The Gaussian radial noise followed Algorithm 1 to introduce annotation uncertainty, with the standard deviation (σ) varying from the integer values of 1-5 inclusive to create 5 datasets of varying degrees of modelled annotation quality. The range for the allowable angles of θ was used to help push the polygon masks outwards. An example of Algorithm 1 performed on a single data point can be seen in Fig. 3. The bounding boxes were updated following (2), where σ is shared between the polygon masks and bounding boxes and x_r, y_r, w_r, h_r represent the new datapoints with radial noise.

$$\begin{aligned} x_r &= x - |\mathcal{N}(0, 1^2)| \\ y_r &= y - |\mathcal{N}(0, 1^2)| \\ w_r &= w + |\mathcal{N}(0, \sigma^2)| \\ h_r &= h + |\mathcal{N}(0, \sigma^2)| \end{aligned} \tag{2}$$

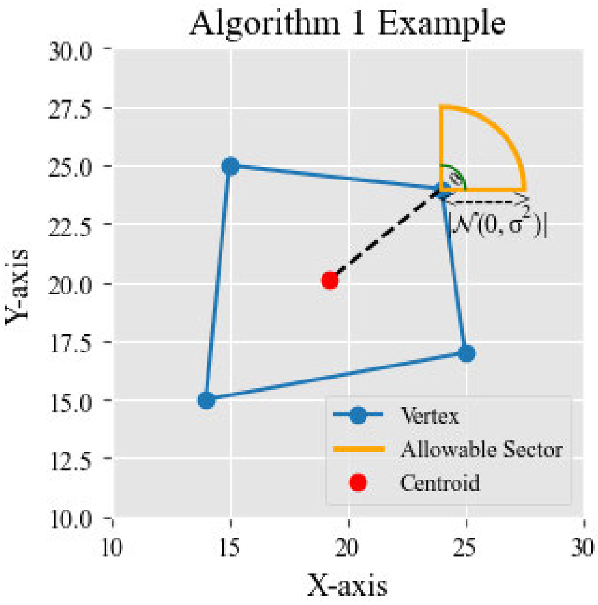

FIGURE 3. Example of Algorithm 1 on a datapoint.

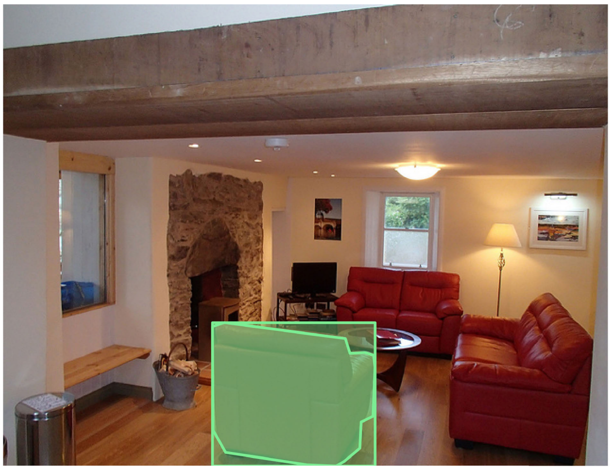

FIGURE 4. Ground truth annotation from COCO dataset [20].

In Fig. 4, the ground truth annotation for the chair is shown. In Fig. 5 the ground truth annotation using Shapely's buffer method with an approximate uniform buffer pixel distance of 5 is shown. And finally, in Fig. 6 the radial noise is introduced with a $\sigma = 5$. Yellow circles are used to highlight some differences in the annotations for Fig. 5 and Fig. 6 relative to the ground truth annotation in Fig. 4.

When investigating annotation quality for a sample of the COCO dataset, it was observed that annotation uncertainty was generally on the outer side of the object, relating to expanding out the annotation. To the authors' knowledge, the true distribution of annotation uncertainty for object detection and instance segmentation datasets is unknown. Taking this into consideration, uniform noise and radial noise were used to model annotation uncertainty. Lastly, reducing the annotations inwards was susceptible to self-intersecting polygons,

Algorithm 1 Algorithm for Adding Radial Noise to Polygon Mask

```
Input: Vertices for polygon mask, image dimensions
Output: Vertices with added radial noise for polygon mask
Calculate centroid of polygon mask; $x_{centroid}, y_{centroid}$
for $x_i, y_i$ in polygon mask vertices do
    Calculate the relevant quadrant for the current point relative to the centroid
    $xdiff_i := x_i - x_{centroid}$
    $ydiff_i := y_i - y_{centroid}$
    if ($xdiff_i \geq 0$ and $ydiff_i \geq 0$ ) then
        $\theta := |\mathcal{N}(45, 15^2)|$
    end if
    if ($xdiff_i \leq 0$ and $ydiff_i \geq 0$ ) then
        $\theta := |\mathcal{N}(135, 15^2)|$
    end if
    if ($xdiff_i \leq 0$ and $ydiff_i \leq 0$ ) then
        $\theta := |\mathcal{N}(225, 15^2)|$
    end if
    if ($xdiff_i \geq 0$ and $ydiff_i \leq 0$ ) then
        $\theta := |\mathcal{N}(315, 15^2)|$
    end if
    Calculate a random Gaussian value to move by $distance_r := |\mathcal{N}(0, \sigma^2)|$
    Update values of x and y to reflect the added radial noise
    $x_i := x_i + distance_r * \cos(\theta)$
    $y_i := y_i + distance_r * \sin(\theta)$
    Ensure new data points are within the image dimensions, 0, $max_{height}$ and 0,$max_{width}$
    if ($x_i > max_{width}$) then
        $x_i := max_{width}$
    end if
    if ($y_i > max_{height}$) then
        $y_i := max_{height}$
    end if
    if ($x_i \leq 0$) then
        $x_i := 0$
    end if
    if ($y_i \leq 0$) then
        $y_i := 0$
    end if
end for
```

which in turn can create numerous multipolygons for the single polygon annotation. On account of this, reducing the annotation inwards was out of scope for this work.

IV. EXPERIMENTAL DESIGN

A. DATASET

These experiments were conducted on a subset of the COCO 2017 dataset [20] and the Cityscapes dataset [31]. A subset of the COCO dataset and its classes were used as this allowed a more reasonable training time for the models. It was not the aim of the study to attain state-of-the-art performance.

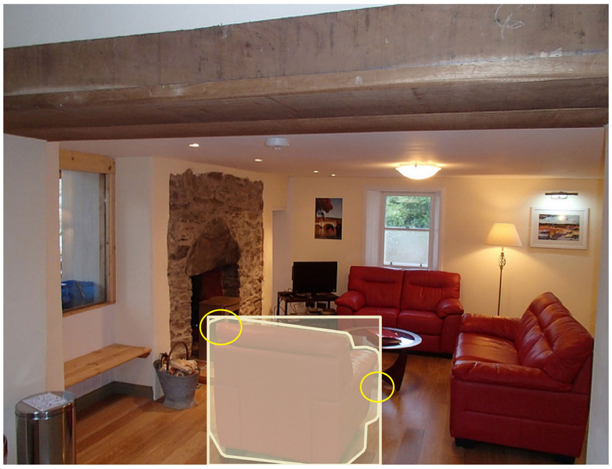

FIGURE 5. Shapely's buffer method (distance = 5) Annotation from COCO dataset [20].

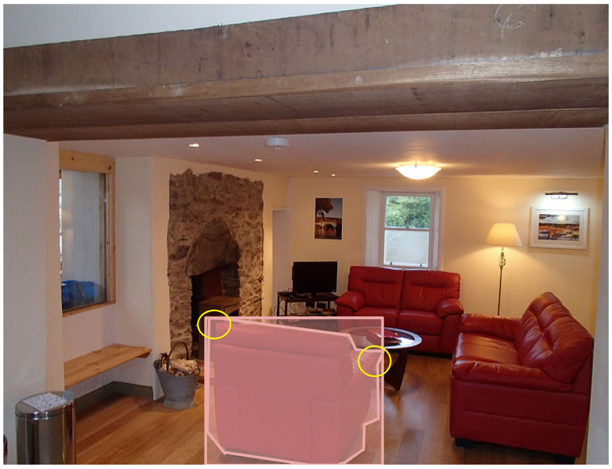

FIGURE 6. Radial noise method (σ = 5) annotation from COCO dataset [20].

The objective of this study is to investigate the relationship of the bounding box and polygon mask annotation quality on the metrics of interest and to quantify the change of metrics to each level of noise, relative to a baseline model trained with the original ground truth annotations.

FiftyOne [45] was used to download 25,000 images from COCO's original training dataset that contained 11 classes, one class per super category, which was randomly selected. The class person was omitted to avoid severe class imbalance. This is reflected in Table 1, where the class counts are shown for 25,000 images when including the class person in comparison to not including the class in the selection criteria for downloading the dataset. The randomly selected classes for the experiments were; bicycle, traffic light, dog, umbrella, skateboard, bottle, pizza, chair, tv, oven, and vase.

The 25,000 images were then split using an 80/20 train and validation split. The test set images were selected from COCO's original validation dataset that contained

TABLE 1. Class counts for COCO dataset classes selection.

Class	No. of Instances [%] when including Person	No. of Instances [%] when excluding Person
person	79026 [0.672]	-
pizza	1861 [0.015]	3594 [0.047]
oven	992 [0.008]	1991 [0.026]
dog	1749 [0.015]	3383 [0.044]
tv	1716 [0.015]	3416 [0.045]
bicycle	1986 [0.017]	4467 [0.058]
umbrella	3536 [0.030]	6772 [0.088]
skateboard	1779 [0.015]	3343 [0.044]
chair	11649 [0.099]	22905 [0.299]
vase	2012 [0.017]	3957 [0.052]
traffic light	4079 [0.034]	7985 [0.104]
bottle	7248 [0.062]	14612 [0.191]

the selected classes. This resulted in a test dataset size of 1,775 images. The resulting dataset breakdown was a train/validation/test split of 75%/19%/6% with 19,946 training images, 5,054 validation images, and a test set of 1,775 images.

For the Cityscapes dataset [31], the 5,000 images that are finely annotated were used for the experiments. The dataset was converted to be utilized for the task of object detection and instance segmentation. The Cityscapes benchmark considers 8 classes for the instance-level semantic labelling task, the classes are as follows; person, rider, car, truck, bus, train, motorcycle and bicycle. The original training dataset was split into an approximate 80/20 train and validation split which resulted in 2,400 images for training and 575 images for validation. The original validation set of 500 images is used as the out-of-sample test dataset.

A breakdown of each of the datasets can be seen in Table 10 and Table 11. The distribution for each class' object size is also given as a percentage under the columns *Small*, *Medium*, and *Large*. The COCO definitions [29] for small, medium, and large object sizes are used. Only single object annotations were considered for this work to minimize the complexity of the problem. Run length encoding (RLE) annotations which are used to annotate a crowd of objects were omitted from both datasets. RLE annotations are identified with COCO's iscrowd = 1 parameter, whereas single object annotations are identified with iscrowd = 0.

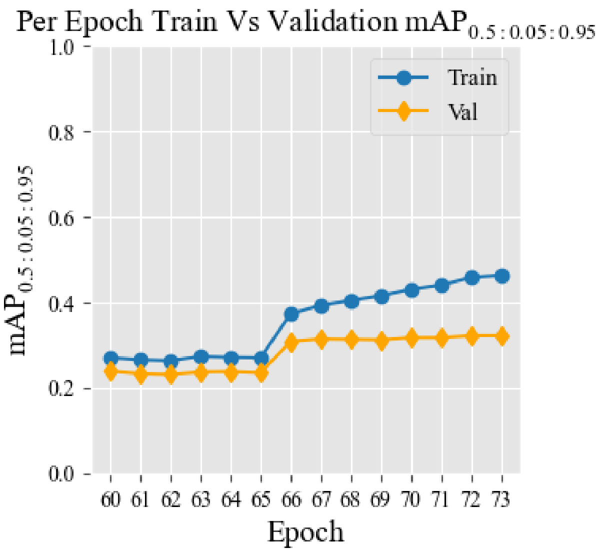

FIGURE 7. Evidence of overfitting when comparing the train & validation scores per epoch for the COCO dataset.

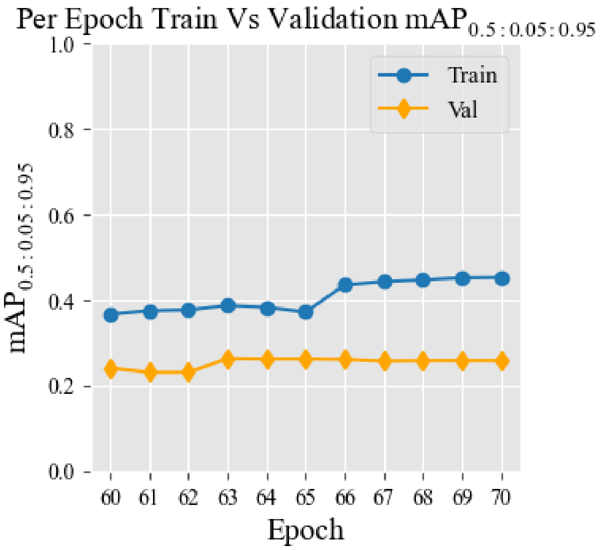

FIGURE 8. Evidence of overfitting when comparing the train & validation scores per Epoch for the cityscapes dataset.

B. TRAINING SETUP

The MMdetection framework [46] was used to train Mask-RCNN models with a ResNet-50 backbone for the experiments [3], [47]. One advantage of this model is its ability to output both object detection and instance segmentation results [3]. All experiments were conducted on a single workstation with an NVIDIA GeForce RTX 3060 GPU card with CUDA 11.6. For the experiments, the training and validation annotations contained the relevant induced noise. The test dataset remained with the original annotations and has not been tampered with. The Mask-RCNN models were trained from scratch for 73 epochs, which took approximately 120 hours per model to train for the COCO dataset and 12 hours to train for Cityscapes. A batch size of 2 images was utilized, due to hardware constraints, with a stochastic gradient descent (SGD) optimizer using a learning rate of 0.02, a momentum of 0.9, and a weight decay of 0.0001. A learning rate scheduler was utilized to drop the learning rate by a factor of 10 at training epoch numbers 65 and 71. Evidence of over-fitting was apparent after epoch 66 when training on the ground truth annotations, as seen in Fig. 7 for the COCO dataset. As for Cityscapes, evidence of over-fitting was apparent after epoch 65. The model weights from epoch 66 and 65 were used for inferencing on the respective test datasets.

C. METRICS

COCO's defined mean average precision [29] (mAP) is the primary metric of interest. When considering individual classes, average precision (AP) can be and is used in place of mAP. A breakdown of mAP and the individual classes' AP results will be reported to provide further insight into annotation quality and performance. COCO's definitions for mean average precision for small, medium, and large objects are denoted by mAP_s, mAP_m, and mAP_l, whereas the mean average precision requiring an IoU threshold of 0.5 and 0.75 are denoted by $mAP_{0.5}$ and $mAP_{0.75}$. $mAP_{0.50:0.05:0.95}$ denotes the COCO primary challenge metric [29].

Whereas comparing the reduction in mAP scores provides insight into how the individual components of mAP degraded, this would not yield an appropriate comparison between object sizes or classes. For example, if the initial score for $mAP_s = 0.1$, using the original annotations, the most mAP_s could degrade is its initial starting point. To put this into perspective, if $mAP_l = 0.5$ using the original annotations, and was to degrade by 0.15 when using noise-induced annotations, it would result in an $mAP_l = 0.35$. However, looking only at differences, mAP_s has degraded less, yet no small objects are being detected.

To provide further insight, linear regression models were fitted to the individual components of mAP scores for each level of noise, in an attempt to provide a standardized comparison between object classes and sizes, that relates directly to mAP. The linear regression models were fitted on a single variable, induced noise level, which in turn gives the interpretation of the β coefficient; for a one-unit increase in induced noise level, on average the mAP score will increase by β.

V. RESULTS

The results were obtained from the test set of 1,775 images of the COCO dataset and 500 test images for the Cityscapes dataset. The approximate uniform buffered pixel distances used in this experiment range from 1 to 10 inclusive, with the radial induced noise ranging from $\sigma = 1$ to 5. The ground truth annotations were also used to train a baseline model. In the figures to follow in this section, a noise level of 0 refers to the ground truth annotations. Mask R-CNN models were trained with both noise-induced annotations along with ground truth annotations. For both the COCO and Cityscapes datasets, 10 datasets, one for each level of approximate uniform pixel distance, were used to train the models. For radial noise, 5 datasets, one for each level of the standard deviation of radial noise were used to train the models. For the fitted linear regression models in this section, β refers to the coefficient of the pixel distance buffering size variable for the uniform noise models, whereas for the radial noise models β refers to the coefficient of the σ variable. A 95% confidence interval is given in square brackets for the constant and β coefficients. Due to the saturation of results in mAP_s after epoch 5 for the uniform models, two linear regression models were used for mAP_s. The first linear regression model was fit from pixel distance buffering size values 0 to 5 inclusive and the second using pixel distance buffering size values 6 to 10.

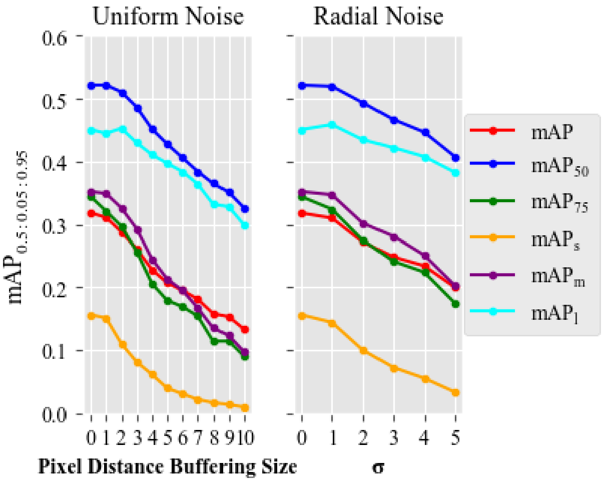

FIGURE 9. COCO dataset Object Detection mAP results.

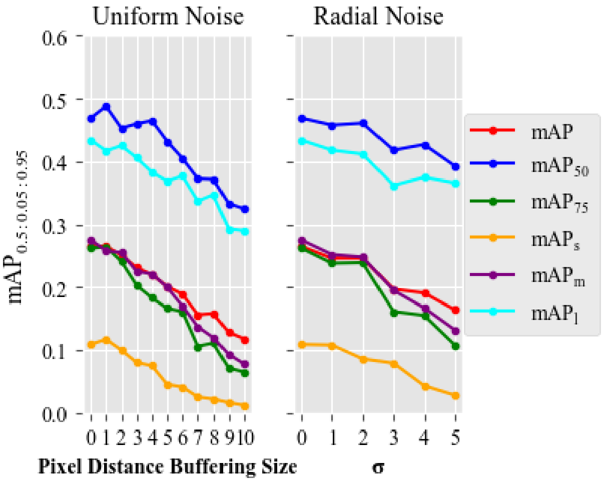

FIGURE 10. Cityscapes dataset object detection mAP results.

A. OBJECT DETECTION

The results of the experiments for object detection are outlined in this section. In Fig. 9 and Fig. 10 a plot of the individual components of mAP against pixel distance buffering size for the uniform noise and σ for radial noise is given for the COCO and Cityscapes datasets respectively. Linear regression models were used to model the relationship between induced noise measures and mAP. The results of the models are presented in Table 2 for the COCO dataset and Table 3 for the Cityscapes dataset. In Fig. 11 and Fig. 12 a plot of the per-class $AP_{0.50:0.05:0.95}$ against pixel distance buffering size for the uniform noise and σ for radial noise is given for the COCO and Cityscapes datasets respectively. The plots are separated by the majority object size for the class in the test dataset. This was utilized for ease

TABLE 2. COCO dataset linear regression model results for object detection.

Type	Adjusted R^2	Constant	β
Uniform Noise			
$mAP_{0.50:0.05:0.95}$	0.978	0.3185 [0.306, 0.331]	-0.0195 [-0.022, -0.017]
$mAP_{0.5}$	0.983	0.5392 [0.527, 0.551]	-0.0214 [-0.023, -0.019]
$mAP_{0.75}$	0.966	0.3355 [0.315, 0.356]	-0.0263 [-0.030, -0.023]
mAP_s	0.886	0.1406 [0.117, 0.164]	-0.0156 [-0.020, -0.012]
mAP_s_{0-5}	0.970	0.1628 [0.146, 0.179]	-0.0253 [-0.031, -0.020]
mAP_s_{6-10}	0.935	0.0578 [0.041, 0.075]	-0.0050 [-0.070, -0.003]
mAP_m	0.985	0.3656 [0.351, 0.380]	-0.0278 [-0.030, -0.025]
mAP_l	0.955	0.4699 [0.455, 0.484]	-0.0159 [-0.018, -0.013]
Radial Noise			
$mAP_{0.50:0.05:0.95}$	0.974	0.3238 [0.309, 0.339]	-0.0241 [-0.029, -0.019]
$mAP_{0.5}$	0.948	0.5346 [0.514, 0.555]	-0.0233 [-0.030, -0.017]
$mAP_{0.75}$	0.981	0.3480 [0.330, 0.366]	-0.0337 [-0.040, -0.028]
mAP_s	0.972	0.1583 [0.142, 0.175]	-0.0260 [-0.031, -0.021]
mAP_m	0.959	0.3646 [0.341, 0.388]	-0.0301 [-0.038, -0.022]
mAP_l	0.898	0.4625 [0.444, 0.481]	-0.0145 [-0.020, -0.008]

TABLE 3. Cityscapes dataset linear regression model results for object detection.

Type	Adjusted R^2	Constant	β
Uniform Noise			
$mAP_{0.50:0.05:0.95}$	0.980	0.4985 [0.475, 0.522]	-0.0165 [-0.020, -0.013]
$mAP_{0.5}$	0.899	0.5392 [0.527, 0.551]	-0.0214 [-0.023, -0.019]
$mAP_{0.75}$	0.974	0.2740 [0.259, 0.289]	-0.0215 [-0.024, -0.019]
mAP_s	0.946	0.1160 [0.104, 0.128]	-0.0115 [-0.013, -0.010]
mAP_s_{0-5}	0.856	0.1210 [0.101, 0.141]	-0.0133 [-0.020,-0.007]
mAP_s_{6-10}	0.867	0.0768 [0.043, 0.110]	-0.0067 [-0.011, -0.003]
mAP_m	0.977	0.2882 [0.275, 0.302]	-0.0207 [-0.023, -0.018]
mAP_l	0.924	0.4432 [0.426, 0.461]	-0.0145 [-0.017, -0.012]
Radial Noise			
$mAP_{0.50:0.05:0.95}$	0.924	0.2695 [0.248, 0.291]	-0.0205 [-0.028, -0.013]
$mAP_{0.5}$	0.822	0.4745 [0.449, 0.500]	-0.0147 [-0.023, -0.006]
$mAP_{0.75}$	0.906	0.2722 [0.235, 0.310]	-0.0313 [-0.044, -0.019]
mAP_s	0.922	0.1189 [0.100, 0.138]	-0.0173 [-0.024, -0.011]
mAP_m	0.950	0.2846 [0.259, 0.310]	-0.0292 [-0.038, -0.021]
mAP_l	0.781	0.4316 [0.403, 0.460]	-0.0148 [-0.024, -0.005]

of readability. Linear regression models were used to model the relationship between induced noise measures and per-class $AP_{0.50:0.05:0.95}$. The results of these models are presented in Table 4 and Table 5.

B. INSTANCE SEGMENTATION

The results of the experiments for instance segmentation are outlined in this section. In Fig. 13 and Fig. 14 a plot of the individual components of mAP against pixel distance buffering size for the uniform noise and σ for radial noise is given for the COCO and Cityscapes datasets respectively. Linear regression models were used to model the relationship between induced noise measures and mAP. The results of the models are presented in Table 6 for the COCO dataset and Table 7 for the Cityscapes dataset. In Fig. 15 and Fig. 16 a plot of the per-class $AP_{0.50:0.05:0.95}$ against pixel distance buffering size for the uniform noise and σ for radial noise is given for the COCO and Cityscapes datasets respectively. The plots are separated by the majority object size for the class in the test dataset. This was utilized for ease of readability. Linear regression models were used to model the relationship between induced noise measures and per-class $AP_{0.50:0.05:0.95}$. The results of these models are presented in Table 8 and Table 9.

TABLE 4. COCO dataset linear regression per-class model results for object detection.

Class	Adjusted R^2	Constant	β
Uniform Noise			
Large			
pizza	0.916	0.4442 [0.426, 0.462]	-0.0136 [-0.017, -0.010]
oven	0.928	0.2585 [0.248, 0.269]	-0.0089 [-0.011, -0.007]
dog	0.967	0.4383 [0.424, 0.453]	-0.0184 [-0.021, -0.016]
tv	0.969	0.4874 [0.471, 0.504]	-0.0214 [-0.025, -0.018]
Medium			
bicycle	0.972	0.2344 [0.223, 0.245]	-0.0149 [-0.017, -0.013]
umbrella	0.948	0.3006 [0.283, 0.319]	-0.0177 [-0.021, -0.014]
skateboard	0.975	0.3886 [0.368, 0.409]	-0.0299 [-0.034, -0.026]
chair	0.979	0.2165 [0.207, 0.226]	-0.0149 [-0.017, -0.013]
vase	0.921	0.2614 [0.234, 0.288]	-0.0215 [-0.027, -0.016]
Small			
traffic light	0.776	0.1936 [0.138, 0.249]	-0.0244 [-0.035, -0.014]
bottle	0.889	0.2861 [0.238, 0.334]	-0.0316 [-0.041, -0.022]
Radial Noise			
Large			
pizza	0.849	0.4417 [0.419, 0.464]	-0.0145 [-0.022, -0.007]
oven	0.743	0.2540 [0.230, 0.278]	-0.0113 [-0.019, -0.003]
dog	0.898	0.4290 [0.411, 0.447]	-0.0147 [-0.021, -0.009]
tv	0.865	0.4829 [0.457, 0.509]	-0.0179 [-0.027, -0.009]
Medium			
bicycle	0.902	0.2352 [0.214, 0.256]	-0.0171 [-0.024, -0.010]
umbrella	0.919	0.2970 [0.278, 0.316]	-0.0171 [-0.023, -0.011]
skateboard	0.904	0.4004 [0.356, 0.444]	-0.0364 [-0.051, -0.022]
chair	0.928	0.2188 [0.202, 0.236]	-0.0162 [-0.022, -0.011]
vase	0.941	0.2690 [0.242, 0.296]	-0.0416 [-0.038, -0.020]
Small			
traffic light	0.900	0.2221 [0.170, 0.274]	-0.0381 [-0.059, -0.025]
bottle	0.973	0.3123 [0.281, 0.343]	-0.0489 [-0.059, -0.039]

TABLE 5. Cityscapes dataset linear regression per-class model results for object detection.

Class	Adjusted R^2	Constant	β
Uniform Noise			
Large			
truck	0.595	0.2068 [0.182, 0.232]	-0.0073 [-0.012, -0.003]
bus	0.672	0.4075 [0.362, 0.453]	-0.0159 [-0.024, -0.008]
train	-0.021	0.1229 [0.087, 0.158]	-0.0024 [-0.008, 0.004]
Medium			
person	0.933	0.2944 [0.270, 0.319]	-0.0215 [-0.026, -0.017]
rider	0.939	0.3181 [0.292, 0.344]	-0.0241 [-0.029, -0.020]
car	0.992	0.5048 [0.493, 0.517]	-0.0327 [-0.035, -0.031]
motorcycle	0.879	0.1563 [0.141, 0.172]	-0.0101 [-0.013, -0.007]
bicycle	0.895	0.2090 [0.191, 0.227]	-0.0124 [-0.015, -0.009]
Radial Noise			
Large			
truck	-0.119	0.1692 [0.110, 0.228]	-0.0048 [-0.024, 0.015]
bus	0.823	0.3981 [0.355, 0.441]	-0.0254 [-0.040, -0.011]
train	0.100	0.0884 [0.026, 0.151]	0.0092 [-0.011, 0.030]
Medium			
person	0.933	0.3081 [0.278, 0.338]	-0.0301 [-0.040, -0.020]
rider	0.939	0.3181 [0.288, 0.349]	-0.0321 [-0.042, -0.022]
car	0.959	0.5119 [0.474, 0.549]	-0.0487 [-0.061, -0.036]
motorcycle	0.527	0.1443 [0.101, 0.188]	-0.0132 [-0.027, 0.001]
bicycle	0.960	0.2166 [0.203, 0.230]	-0.0181 [-0.023, -0.014]

VI. DISCUSSION

The results enable us to compare the mAP performance for object detection and instance segmentation for various ground truth annotation qualities. For object detection,

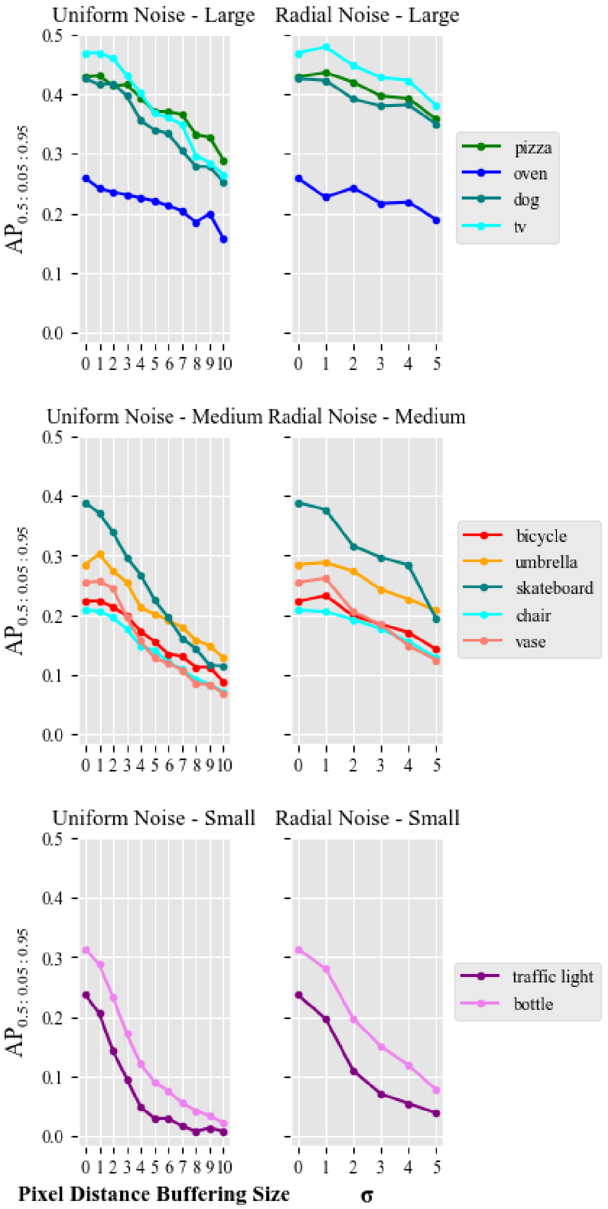

FIGURE 11. COCO dataset object detection AP per-class results.

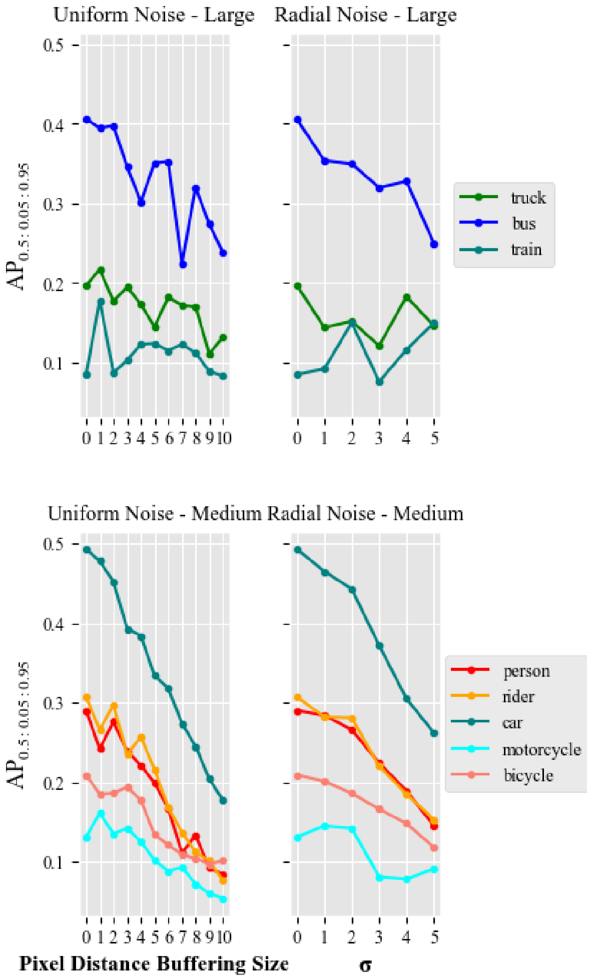

FIGURE 12. Cityscapes dataset object detection AP per-class results.

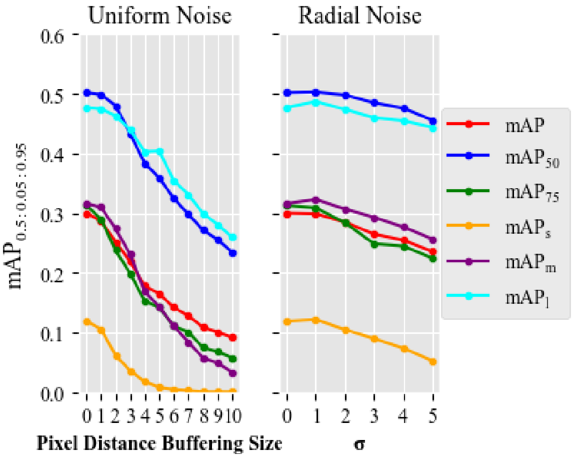

FIGURE 13. COCO dataset instance segmentation mAP results.

as seen in Fig. 9 and Fig. 10, when introducing uniform noise into the datasets, there was a reduction across all components of mAP. For the radially-induced noise, the degradation across the components of mAP is lesser in comparison to the uniform noise, albeit there is still degradation as annotation uncertainty increases. These results indicate there is a degradation in mAP performance when introducing annotation uncertainty into the annotations for object detection, for both noise types; uniform and radial. This reflects the need for accurate bounding boxes to be utilized as ground truth annotations for object detection.

Looking into the per-class scores, as seen in Fig. 11 and Fig. 12 along with the negative β coefficients in Table 4 and Table 5, there was a reduction for all of the classes used in the experiments for both induced noise types. However, the reductions between classes vary. This suggests that annotation quality and $AP_{0.50:0.05:0.95}$ performance are

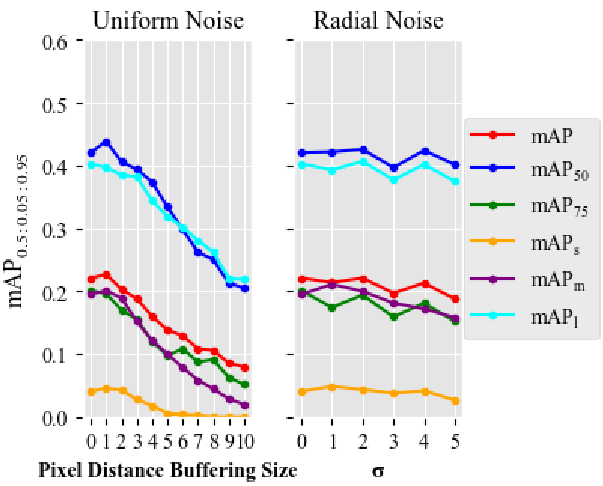

FIGURE 14. Cityscapes dataset instance segmentation mAP results.

TABLE 6. COCO dataset linear regression model results for instance segmentation.

Type	Adjusted R^2	Constant [95% CI]	Beta [95% CI]
Uniform Noise			
$mAP_{0.50:0.05:0.95}$	0.956	0.2899 [0.270, 0.310]	-0.0221 [-0.025, -0.019]
$mAP_{0.5}$	0.980	0.5157 [0.498, 0.534]	-0.0296 [-0.033, -0.027]
$mAP_{0.75}$	0.940	0.2903 [0.262, 0.318]	-0.0263 [-0.031, -0.022]
mAP_s	0.741	0.0905 [0.062, 0.119]	-0.0116 [-0.016, -0.007]
$mAP_{s_{0-5}}$	0.950	0.1185 [0.098, 0.139]	-0.0241 [-0.031, -0.017]
$mAP_{s_{6-10}}$	0.708	0.0102 [0.002, 0.018]	-0.0010 [-0.002, -0.001]
mAP_m	0.965	0.3187 [0.293, 0.344]	-0.0314 [-0.036, -0.027]
mAP_l	0.975	0.4997 [0.484, 0.516]	-0.0238 [-0.027, -0.021]
Radial Noise			
$mAP_{0.50:0.05:0.95}$	0.954	0.3070 [0.296, 0.318]	-0.0135 [-0.017 , -0.010]
$mAP_{0.5}$	0.861	0.5098 [0.496, 0.524]	-0.0093 [-0.014, -0.005]
$mAP_{0.75}$	0.944	0.3185 [0.301, 0.336]	-0.0191 [-0.025, -0.013]
mAP_s	0.927	0.1288 [0.114, 0.143]	-0.0140 [-0.019, -0.009]
mAP_m	0.886	0.3274 [0.310, 0.344]	-0.0128 [-0.018, -0.007]
mAP_l	0.816	0.4858 [0.472, 0.500]	-0.0079 [-0.012, -0.003]

TABLE 7. Cityscapes dataset linear regression model results for instance segmentation.

Type	Adjusted R^2	Constant [95% CI]	Beta [95% CI]
Uniform Noise			
$mAP_{0.50:0.05:0.95}$	0.973	0.2295 [0.218, 0.241]	-0.0160 [-0.018, -0.014]
$mAP_{0.5}$	0.964	0.4538 [0.433, 0.475]	-0.0254 [-0.029, -0.022]
$mAP_{0.75}$	0.943	0.1976 [0.182, 0.213]	-0.0152 [-0.018, -0.013]
mAP_s	0.836	0.0424 [0.033, 0.052]	-0.0052 [-0.007, -0.004]
$mAP_{s_{0-5}}$	0.806	0.0489 [0.035, 0.063]	-0.0078 [-0.012, -0.003]
$mAP_{s_{6-10}}$	0.733	0.0074 [0.001, 0.013]	-0.0008 [-0.002, -0.001]
mAP_m	0.971	0.2095 [0.195, 0.224]	-0.0203 [-0.023, -0.018]
mAP_l	0.971	0.4213 [0.406, 0.436]	-0.0204 [-0.023, -0.018]
Radial Noise			
$mAP_{0.50:0.05:0.95}$	0.470	0.2227 [0.203, 0.242]	-0.0055 [-0.012 , 0.001]
$mAP_{0.5}$	0.069	0.4238 [0.400, 0.448]	-0.0034 [-0.011, 0.005]
$mAP_{0.75}$	0.401	0.1953 [0.166, 0.225]	-0.0074 [-0.017, 0.002]
mAP_s	0.361	0.0461 [0.034, 0.058]	-0.0028 [-0.007, 0.001]
mAP_m	0.715	0.2091 [0.188, 0.230]	-0.0092 [-0.016, -0.002]
mAP_l	0.132	0.4030 [0.377, 0.429]	-0.0041 [-0.013, 0.004]

TABLE 8. COCO dataset linear regression per-class model results for instance segmentation.

Class	Adjusted R^2	Constant [95% CI]	β [95% CI]
Uniform Noise			
Large			
pizza	0.930	0.4419 [0.424, 0.460]	-0.0176 [-0.023, -0.013]
oven	0.881	0.2526 [0.235, 0.271]	-0.0131 [-0.018, -0.008]
dog	0.978	0.4406 [0.421, 0.460]	-0.0347 [-0.040, -0.029]
tv	0.934	0.5215 [0.496, 0.547]	-0.0254 [-0.032, -0.018]
Medium			
bicycle	0.970	0.1489 [0.138, 0.160]	-0.0170 [-0.020, -0.014]
umbrella	0.966	0.3824 [0.356, 0.409]	-0.0367 [-0.044, -0.029]
skateboard	0.972	0.2233 [0.202, 0.244]	-0.0331 [-0.039, -0.027]
chair	0.939	0.1442 [0.129, 0.160]	-0.0161 [-0.020, -0.012]
vase	0.978	0.2741 [0.257, 0.291]	-0.0304 [-0.035, -0.026]
Small			
traffic light	0.910	0.2228 [0.177, 0.269]	-0.0391 [-0.052, -0.026]
bottle	0.943	0.3027 [0.259, 0.346]	-0.0472 [-0.059, -0.035]
Radial Noise			
Large			
pizza	0.723	0.4394 [0.423, 0.456]	-0.0074 [-0.013, -0.002]
oven	0.676	0.2500 [0.226, 0.274]	-0.0095 [-0.017, -0.002]
dog	0.926	0.4294 [0.419, 0.439]	-0.0095 [-0.013, -0.006]
tv	0.384	0.5100 [0.491, 0.529]	-0.0045 [-0.011, 0.002]
Medium			
bicycle	0.942	0.1439 [0.138, 0.150]	-0.0065 [-0.009, -0.005]
umbrella	0.876	0.3845 [0.362, 0.407]	-0.0159 [-0.023, -0.009]
skateboard	0.812	0.2210 [0.202, 0.240]	-0.0106 [-0.017, -0.004]
chair	0.798	0.1429 [0.131, 0.155]	-0.0066 [-0.011, -0.003]
vase	0.922	0.2772 [0.259, 0.296]	-0.0172 [-0.023, -0.011]
Small			
traffic light	0.957	0.2502 [0.228, 0.272]	-0.0274 [-0.035, -0.020]
bottle	0.971	0.3259 [0.305, 0.346]	-0.0316 [-0.038, -0.025]

class-dependent for object detection. One potential factor for the observed class dependence for object detection is the size of the objects of interest. The classes traffic light and bottle had a majority of their instances in the size small category for the COCO dataset. Both these classes resulted in significant decreases in $AP_{0.50:0.05:0.95}$ for both induced noise types for object detection.

For instance segmentation, as seen in Fig. 13 and Fig. 14, when introducing uniform noise into the datasets, there was a reduction across all components of mAP. For radially-induced noise, the degradation across the components of mAP is far less severe in comparison to the uniform noise, however, there is still some degradation as annotation uncertainty increases. These results indicate there is a degradation in mAP performance when introducing annotation uncertainty into the annotations for instance segmentation, for both noise types; uniform and radial. This reflects the need for accurate polygon masks to be utilized as ground truth annotations for instance segmentation.

Looking into the per-class scores, as seen in Fig. 15 and Fig. 16 along with the β coefficients in Table 8 and Table 9, there was a reduction for most of the classes used in the experiments for both induced noise types. However, the reductions between classes vary. This suggests that annotation quality and $AP_{0.50:0.05:0.95}$ performance are class-dependent for instance segmentation. Again a potential factor for the observed class dependence is the size of the objects of interest. The classes traffic light and bottle had a majority of their instances in the size small category for the COCO dataset. Both these classes resulted in significant decreases in $AP_{0.50:0.05:0.95}$ for both induced noise types for instance segmentation.

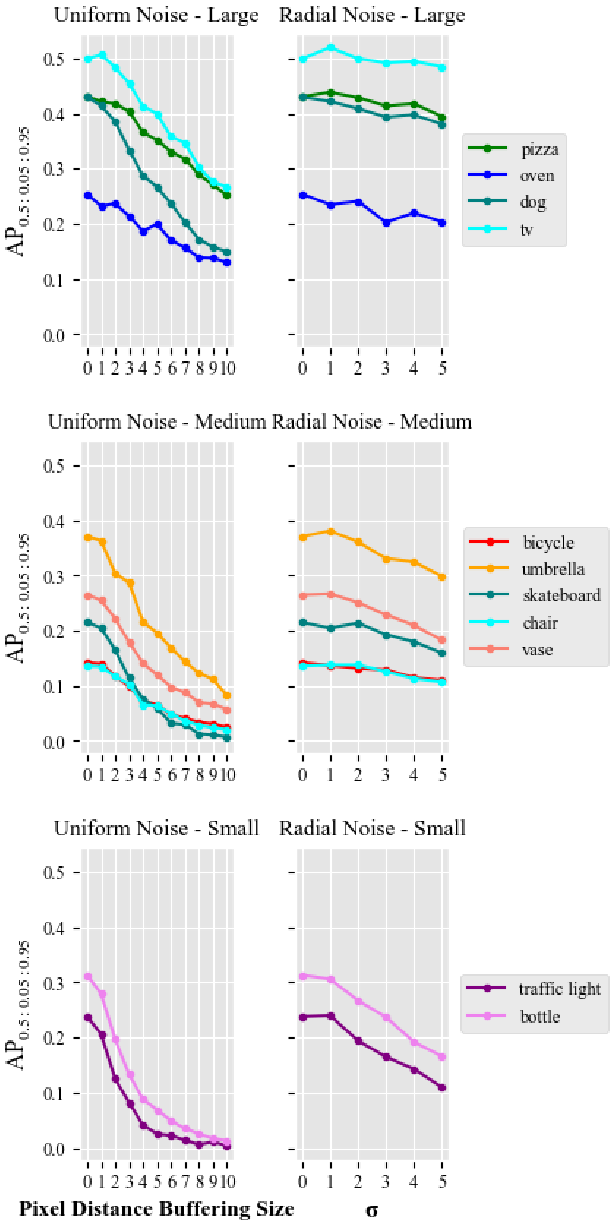

FIGURE 15. COCO dataset instance segmentation per-class mAP results.

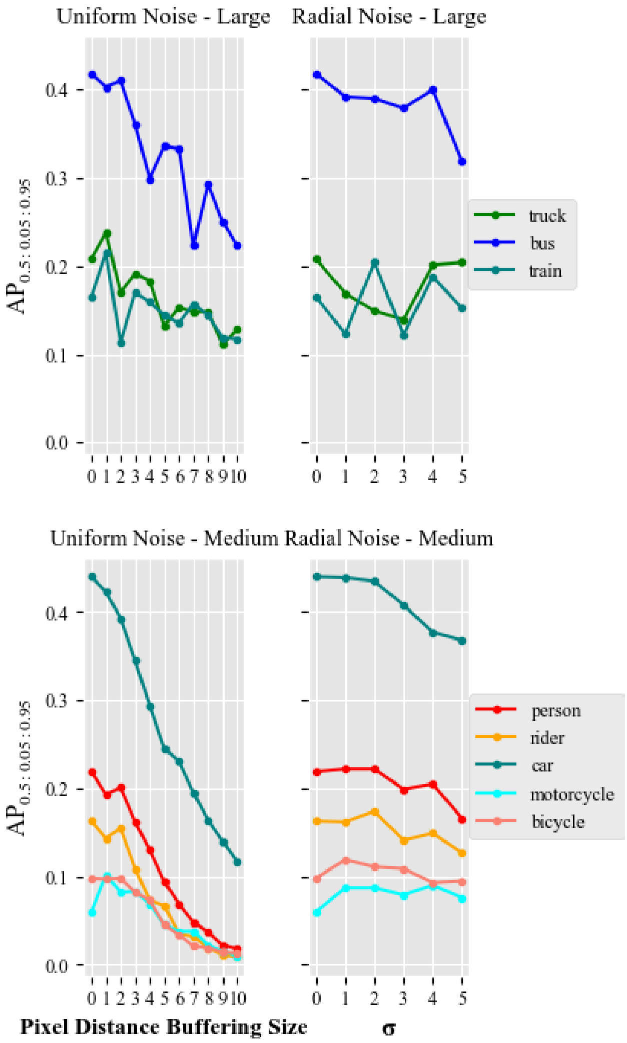

FIGURE 16. Cityscapes dataset instance segmentation per-class mAP results.

For the Cityscapes dataset, when looking into the per-class AP results for both object detection as seen in Fig. 12 and instance segmentation, as seen in Fig. 16, the variance is quite significant for the classes truck, bus, train and motorcycle. An explanation for this variance is the small sample size of the classes in the dataset, as seen in Table 11. As these classes are less than 1% of the number of instances in each of the train, validation and test datasets, this in turn would result in higher variances in the models. Small sample sizes can also skew the results due to the impact one instance can have on the overall percentage. For example, on the Cityscapes test dataset, getting one extra truck predicted correctly would result in an increase of 1.1% in comparison to the impact of one extra person predicted correctly, which results in an increase of 0.03%. Due to this class imbalance, the fitted linear regression models would struggle to account for this variance, which has resulted in lower adjusted R^2 values for the mAP components in comparison to the COCO counterparts. As the impact of each of the smaller class sizes would impact each of the mAP calculations, a reduction across the adjusted R^2 values is expected. With all of these factors in mind, it is important to note the results for the small class sizes should not be given great consideration.

Whereas for the COCO dataset, a strong linear relationship between both noise types and mAP for object detection and instance segmentation was observed. An explanation for this strong linear relationship in comparison to the Cityscapes results may be down to the more evenly distributed classes as seen in Table 10. An adjusted R^2 of 0.978 and 0.956 for object detection and instance segmentation were recorded respectively for the uniform noise, as seen in Table 4 and Table 8.

TABLE 9. Cityscapes dataset linear regression per-class model results for instance segmentation.

Class	Adjusted R^2	Constant [95% CI]	β [95% CI]
Uniform Noise			
Large			
truck	0.728	0.2141 [0.189, 0.239]	-0.0099 [-0.014, -0.006]
bus	0.821	0.4217 [0.383, 0.460]	-0.0198 [-0.026, -0.013]
train	0.282	0.1753 [0.143, 0.207]	-0.0053 [-0.011, 0.000]
Medium			
person	0.962	0.2207 [0.202, 0.240]	-0.0225 [-0.026, -0.019]
rider	0.928	0.1605 [0.140, 0.181]	-0.0173 [-0.021, -0.014]
car	0.984	0.4441 [0.426, 0.462]	-0.0345 [-0.038, -0.031]
motorcycle	0.778	0.0921 [0.074, 0.110]	-0.0082 [-0.011, -0.005]
bicycle	0.930	0.1065 [0.094, 0.119]	-0.0105 [-0.013, -0.008]
Radial Noise			
Large			
truck	-0.233	0.1736 [0.106, 0.241]	0.0019 [-0.020, 0.024]
bus	0.471	0.4174 [0.368, 0.467]	-0.0138 [-0.030, 0.003]
train	-0.243	0.1556 [0.080, 0.231]	0.0014 [-0.023, 0.026]
Medium			
person	0.631	0.2297 [0.203, 0.256]	-0.0097 [-0.018, -0.001]
rider	0.528	0.1707 [0.147, 0.194]	-0.0072 [-0.015, 0.001]
car	0.874	0.4521 [0.429, 0.475]	-0.0164 [-0.024, -0.009]
motorcycle	-0.059	0.0740 [0.051, 0.097]	0.0023 [-0.005, 0.010]
bicycle	0.050	0.1110 [0.091, 0.131]	-0.0027 [-0.009, 0.004]

For radial noise, the adjusted R^2 is 0.974 for object detection and 0.954 for instance segmentation. The β coefficient from the linear regression models yields insight into quantifying the performance degradation. For a one-unit increase in pixel distance buffer size, on average the $mAP_{0.50:0.05:0.95}$ will reduce by -0.0195[-0.022, -0.017] for object detection and -0.0221[-0.025, -0.019] for instance segmentation. For a one-unit increase in σ for radial noise, on average the $mAP_{0.50:0.05:0.95}$ will reduce by -0.0241[-0.029, -0.019] for object detection and -0.0135[-0.017, -0.010] for instance segmentation.

A reduction across mAP for object detection and instance segmentation when introducing annotation uncertainty is no surprise. As supervised-learning neural network performance relies on the quality of the annotations, a degradation in the annotation quality will be reflected in a reduction of the mAP. This work set out to investigate the relationship between annotation quality and mAP performance. For both types of induced noise used in this research, the noise from the training data has forced the model to include some noise around the objects of interest when inferencing with the model, thus reducing the IoU with the ground truth annotation on the test set, resulting in a reduction in mAP. This noise during inferencing was more apparent for the uniform noise models. However, the model predictions still allow for the identification and localization of the objects of interest.

A direct comparison between uniform and radially-induced noise may not be a fair comparison, due to the nature of the induced noise for both. The uniform noise significantly degrades each vertex for instance segmentation in comparison to the radial noise. The radial noise was normally distributed and centred around 0, meaning some vertices would only experience a marginal degradation. However, all things considered, the results show as the degradation of the annotation increases, a reduction in mAP performance is observed. This reflects the need for accurate annotations for supervised learning computer vision tasks.

Radial noise degradation for instance segmentation is lower than object detection. An explanation for this may be down to how the radial noise was implemented. As bounding boxes only require four normally distributed data points to introduce the noise; the first two update the x & y co-ordinates for the bounding box starting position, the third updates the width and the fourth and final point updates the height, there is a possibility, due to the nature of the normal distribution, for relatively high values being introduced. This in turn would significantly degrade the bounding box annotation.

The findings of this study have to be seen in light of some limitations. To the authors' knowledge, no prior work has modelled annotation uncertainty for object detection or instance segmentation datasets. In light of this information, the use of uniform noise and normally distributed radial noise was selected to model annotation uncertainty. This work allows us to quantify the degradation of mAP with respect to modelled annotation uncertainty to better understand the relationship between annotation quality and performance.

VII. CONCLUSION AND FUTURE WORK

In this paper, the relationship between object detection and instance segmentation annotation quality and mAP performance is studied. The observed results were attained by a Mask-RCNN model with a ResNet-50 backbone on a subset of the COCO 2017 challenge and Cityscapes datasets. The ground truth annotations for both bounding boxes and polygon masks had two separate types of noise introduced to the annotations; uniform and radial.

For object detection and instance segmentation, both types of induced noise negatively affected the mAP. When investigating the per-class $AP_{0.50:0.05:0.95}$ performance, there was a reduction seen in all classes but motorcycle used in the experiments, with the reductions between classes varying. This suggests that annotation quality and $AP_{0.50:0.05:0.95}$ performance is class-dependent. A strong linear relationship was observed between both noise types and mAP for the COCO dataset. An adjusted R^2 of 0.978 for uniform noise and 0.974 for radial noise was recorded for object detection, with instance segmentation recording an adjusted R^2 of 0.956 for uniform noise and 0.954 for radial noise when using $mAP_{0.50:0.05:0.95}$.

For radially-induced noise for instance segmentation, there is some robustness for $\sigma = 1$, as the degradation is less than 2% for all components of mAP. While the required accuracy of mask predictions for instance segmentation is application dependent, this work has quantified the degradation in mAP for varying annotation qualities to help inform any decisions on annotation labelling quality and the expected degradation.

This study has quantified empirically the performance between annotation quality and mAP when introducing two different noises to the ground truth annotations for a subset of the COCO 2017 and Cityscapes datasets. The reduction in mAP across both noise measures for object detection and instance segmentation reflects the need for accurate polygon and bounding boxes for fully supervised object detection and instance segmentation tasks.

Future research should further develop and confirm these initial findings by conducting experiments on more diverse computer vision datasets, such as other benchmark datasets used for object detection and instance segmentation with different model architectures to investigate if the results from these experiments generalize. Additionally, the use of transfer learning with noisy annotations should be investigated to determine if the results deviate from the current experiments, which were trained from scratch. Finally, combining the noise types used into a single dataset would be of interest, as this may better reflect the annotation uncertainty when multiple annotators are used to annotate a dataset.

APPENDIX A
COCO DATASET

TABLE 10. COCO dataset.

Class	No. of Instances [%]	Small	Medium	Large
Train				
pizza	2868 [0.04]	0.08	0.26	**0.66**
oven	1600 [0.03]	0.01	0.25	**0.74**
dog	2720 [0.04]	0.07	0.25	**0.69**
tv	2757 [0.05]	0.06	0.37	**0.57**
bicycle	3538 [0.06]	0.25	**0.43**	0.32
umbrella	5220 [0.09]	0.27	**0.39**	0.35
skateboard	2680 [0.04]	0.24	**0.49**	0.27
chair	18296 [0.30]	0.25	**0.45**	0.30
vase	3183 [0.05]	0.28	**0.40**	0.30
traffic light	6495 [0.11]	**0.73**	0.23	0.05
bottle	11533 [0.19]	**0.47**	0.43	0.10
Validation				
pizza	726 [0.05]	0.08	0.26	**0.66**
oven	391 [0.03]	0.01	0.25	**0.74**
dog	663 [0.04]	0.06	0.22	**0.71**
tv	659 [0.04]	0.04	0.39	**0.57**
bicycle	929 [0.06]	0.23	**0.45**	0.32
umbrella	1552 [0.10]	0.24	**0.44**	0.32
skateboard	663 [0.04]	0.22	**0.50**	0.27
chair	4609 [0.30]	0.22	**0.47**	0.31
vase	774 [0.05]	0.33	**0.37**	0.30
traffic light	1490 [0.10]	**0.73**	0.45	0.32
bottle	3079 [0.20]	**0.48**	0.42	0.09
Test				
pizza	284 [0.05]	0.11	0.29	**0.60**
oven	143 [0.03]	0.03	0.27	**0.70**
dog	218 [0.04]	0.04	0.22	**0.74**
tv	288 [0.05]	0.10	0.32	**0.57**
bicycle	314 [0.06]	0.26	**0.44**	0.31
umbrella	407 [0.07]	0.15	**0.43**	0.42
skateboard	179 [0.03]	0.19	**0.48**	0.33
chair	1771 [0.32]	0.23	**0.47**	0.30
vase	274 [0.05]	0.35	**0.41**	0.24
traffic light	634 [0.11]	**0.75**	0.21	0.04
bottle	1013 [0.18]	**0.48**	0.40	0.12

APPENDIX B
CITYSCAPES DATASET

TABLE 11. Cityscapes dataset.

Class	No. of Instances [%]	Small	Medium	Large
Train				
person	14176 [0.34]	0.04	**0.69**	0.27
rider	1388 [0.03]	0.02	**0.63**	0.35
car	21314 [0.52]	0.01	**0.56**	0.43
truck	366 [0.01]	0.003	0.45	**0.55**
bus	297 [0.01]	0.02	0.35	**0.63**
train	131 [0.003]	0.0	0.24	**0.76**
motorcycle	2680 [0.01]	0.03	**0.52**	0.45
bicycle	18296 [0.07]	0.02	**0.65**	0.34
Validation				
person	3322 [0.35]	0.04	**0.67**	0.29
rider	294 [0.03]	0.003	**0.6**	0.39
car	4908 [0.52]	0.01	**0.54**	0.45
truck	89 [0.01]	0.0	0.4	**0.6**
bus	56 [0.01]	0.0	0.21	**0.79**
train	27 [0.003]	0.0	0.3	**0.7**
motorcycle	126 [0.01]	0.03	**0.51**	0.46
bicycle	657 [0.07]	0.02	**0.65**	0.34
Test				
person	3315 [0.34]	0.03	**0.68**	0.29
rider	513 [0.05]	0.004	**0.64**	0.36
car	4537 [0.46]	0.02	**0.57**	0.41
truck	91 [0.01]	0.0	0.4	**0.6**
bus	96 [0.01]	0.0	0.25	**0.75**
train	23 [0.002]	0.0	0.13	**0.87**
motorcycle	144 [0.01]	0.01	**0.6**	0.39
bicycle	1136 [0.12]	0.01	**0.62**	0.37

REFERENCES

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 25, 2012, pp. 1106–1114.

[2] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A deep convolutional encoder–decoder architecture for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, Dec. 2017.

[3] K. He, G. Gkioxari, P. Dollar, and R. Girshick, "Mask R-CNN," in *Proc. ICCV*, 2017, pp. 2961–2969.

[4] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 779–788.

[5] S. Khan, M. Naseer, M. Hayat, S. W. Zamir, F. S. Khan, and M. Shah, "Transformers in vision: A survey," *ACM Comput. Surv.*, vol. 54, no. 10, pp. 1–41, 2022.

[6] P. Cao, Z. Zhu, Z. Wang, Y. Zhu, and Q. Niu, "Applications of graph convolutional networks in computer vision," *Neural Comput. Appl.*, vol. 34, no. 16, pp. 13387–13405, 2022.

[7] C. Sun, A. Shrivastava, S. Singh, and A. Gupta, "Revisiting unreasonable effectiveness of data in deep learning era," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 843–852.

[8] D. Karimi, H. Dou, S. K. Warfield, and A. Gholipour, "Deep learning with noisy labels: Exploring techniques and remedies in medical image analysis," *Med. Image Anal.*, vol. 65, Oct. 2020, Art. no. 101759.

[9] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, and S. Ghemawat, "TensorFlow: Large-scale machine learning on heterogeneous distributed systems," 2016, *arXiv:1603.04467*.

[10] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, and A. Desmaison, ''PyTorch: An imperative style, high-performance deep learning library,'' in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 32, 2019, pp. 1–12.

[11] T. Chen, M. Li, Y. Li, M. Lin, N. Wang, M. Wang, T. Xiao, B. Xu, C. Zhang, and Z. Zhang, ''MXNet: A flexible and efficient machine learning library for heterogeneous distributed systems,'' 2015, *arXiv:1512.01274*.

[12] T. Liang, H. Bao, W. Pan, and F. Pan, ''Traffic sign detection via improved sparse R-CNN for autonomous vehicles,'' *J. Adv. Transp.*, vol. 2022, pp. 1–16, Mar. 2022.

[13] C. Eising, J. Horgan, and S. Yogamani, ''Near-field perception for low-speed vehicle automation using surround-view fisheye cameras,'' *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 9, pp. 13976–13993, Sep. 2022.

[14] R. Zhang, L. Wu, Y. Yang, W. Wu, Y. Chen, and M. Xu, ''Multi-camera multi-player tracking with deep player identification in sports video,'' *Pattern Recognit.*, vol. 102, Jun. 2020, Art. no. 107260.

[15] G. Thomas, R. Gade, T. B. Moeslund, P. Carr, and A. Hilton, ''Computer vision for sports: Current applications and research topics,'' *Comput. Vis. Image Understand.*, vol. 159, pp. 3–18, Jun. 2017.

[16] A. Esteva, K. Chou, S. Yeung, N. Naik, A. Madani, A. Mottaghi, Y. Liu, E. Topol, J. Dean, and R. Socher, ''Deep learning-enabled medical computer vision,'' *Npj Digit. Med.*, vol. 4, no. 1, pp. 1–9, Jan. 2021.

[17] J. Gao, Y. Yang, P. Lin, and D. S. Park, ''Computer vision in healthcare applications,'' *J. Healthcare Eng.*, vol. 2018, Mar. 2018, Art. no. 5157020.

[18] R. Zhang, L. Xu, Z. Yu, Y. Shi, C. Mu, and M. Xu, ''Deep-IRTarget: An automatic target detector in infrared imagery using dual-domain feature extraction and allocation,'' *IEEE Trans. Multimedia*, vol. 24, pp. 1735–1749, 2022.

[19] X. Yang, Y. Ye, X. Li, R. Y. K. Lau, X. Zhang, and X. Huang, ''Hyperspectral image classification with deep learning models,'' *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 9, pp. 5408–5423, Sep. 2018.

[20] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollar, and C. L. Zitnick, ''Microsoft COCO: Common objects in context,'' in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 740–755.

[21] Y. Hu, Z. Ou, X. Xu, and M. Song, ''A crowdsourcing repeated annotations system for visual object detection,'' in *Proc. 3rd Int. Conf. Vis., Image Signal Process.*, Aug. 2019.

[22] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, and A. C. Berg, ''ImageNet large scale visual recognition challenge,'' *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.

[23] Y. Xu, L. Zhu, Y. Yang, and F. Wu, ''Training robust object detectors from noisy category labels and imprecise bounding boxes,'' *IEEE Trans. Image Process.*, vol. 30, pp. 5782–5792, 2021.

[24] H. Li, Z. Wu, C. Zhu, C. Xiong, R. Socher, and L. S. Davis, ''Learning from noisy anchors for one-stage object detection,'' in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10588–10597.

[25] Z. Zhang and M. Sabuncu, ''Generalized cross entropy loss for training deep neural networks with noisy labels,'' in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 31, 2018, pp. 1–11.

[26] L. Jiang, Z. Zhou, T. Leung, L.-J. Li, and L. Fei-Fei, ''MentorNet: Learning data-driven curriculum for very deep neural networks on corrupted labels,'' in *Proc. Int. Conf. Mach. Learn.*, Jul. 2018, pp. 2304–2313.

[27] M. Ren, W. Zeng, B. Yang, and R. Urtasun, ''Learning to reweight examples for robust deep learning,'' in *Proc. Int. Conf. Mach. Learn.*, Jul. 2018, pp. 4334–4343.

[28] J. Shu, Q. Xie, L. Yi, Q. Zhao, S. Zhou, Z. Xu, and D. Meng, ''Meta-weight-net: Learning an explicit mapping for sample weighting,'' in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 32, 2019, pp. 1–12.

[29] Microsoft COCO. (2021). *Detection Evaluation Metrics*. Accessed: Oct. 18, 2022. [Online]. Available: https://cocodataset.org/#detection-eval

[30] V. Taran, Y. Gordienko, A. Rokovyi, O. Alienin, and S. Stirenko, ''Impact of ground truth annotation quality on performance of semantic image segmentation of traffic conditions,'' in *Advances in Computer Science for Engineering and Education II*, Z. Hu, S. Petoukhov, I. Dychka, and M. He, Eds. Cham, Switzerland: Springer, 2020, pp. 183–193.

[31] M. Cordts, M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, ''The cityscapes dataset for semantic urban scene understanding,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 3213–3223.

[32] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, ''Pyramid scene parsing network,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2881–2890.

[33] J. F. Mullen, F. R. Tanner, and P. A. Sallee, ''Comparing the effects of annotation type on machine learning detection performance,'' in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2019.

[34] F. Tanner, B. Colder, C. Pullen, D. Heagy, M. Eppolito, V. Carlan, C. Oertel, and P. Sallee, ''Overhead imagery research data set-an annotated data library & tools to aid in the development of computer vision algorithms,'' in *Proc. IEEE Appl. Imag. Pattern Recognit. Workshop (AIPR)*, Oct. 2009, pp. 1–8.

[35] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, ''OverFeat: Integrated recognition, localization and detection using convolutional networks,'' 2013, *arXiv:1312.6229*.

[36] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, ''The Pascal visual object classes (VOC) challenge,'' *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, Sep. 2009.

[37] D. Acuna, A. Kar, and S. Fidler, ''Devil is in the edges: Learning semantic boundaries from noisy annotations,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 11075–11083.

[38] Z. Yu, C. Feng, M.-Y. Liu, and S. Ramalingam, ''CASENet: Deep category-aware semantic edge detection,'' in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Venice, Italy, Jul. 2017, pp. 5964–5973.

[39] B. Hariharan, P. Arbelaez, L. Bourdev, S. Maji, and J. Malik, ''Semantic contours from inverse detectors,'' in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 991–998.

[40] D. Rolnick, A. Veit, S. Belongie, and N. Shavit, ''Deep learning is robust to massive label noise,'' 2017, *arXiv:1705.10694*.

[41] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, ''ImageNet: A large-scale hierarchical image database,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.

[42] Y. LeCun. (1998). *The MNIST Database of Handwritten Digits*. [Online]. Available: http://yann.lecun.com/exdb/mnist/

[43] A. Krizhevsky, ''Learning multiple layers of features from tiny images,'' Univ. Toronto, Toronto, ON, Canada, Tech. Rep. TR-2009, 2009.

[44] S. Gillies. (2013). *The Shapely User Manual*. Accessed: Oct. 18, 2022. [Online]. Available: https://pypi.org/project/Shapely

[45] B. E. Moore and J. J. Corso. (2020). *Fiftyone*. [Online]. Available: https://github.com/voxel51/fiftyone

[46] K. Chen, J. Wang, J. Pang, Y. Cao, Y. Xiong, X. Li, S. Sun, W. Feng, Z. Liu, J. Xu, and Z. Zhang, ''MMDetection: Open MMLab detection toolbox and benchmark,'' 2019, *arXiv:1906.07155*.

[47] K. He, X. Zhang, S. Ren, and J. Sun, ''Deep residual learning for image recognition,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2016, pp. 770–778.

CATHAOIR AGNEW received the B.S. degree in financial mathematics and the M.S. degree in artificial intelligence and machine learning from the University of Limerick, Limerick, Ireland, in 2020 and 2021, respectively, where he is currently pursuing the Ph.D. degree in electronic and computer engineering. His research interests include artificial intelligence and computer vision.

CIARÁN EISING (Member, IEEE) received the B.E. degree in electronic and computer engineering and the Ph.D. degree from the National University of Ireland, Galway, in 2003 and 2010, respectively. From 2009 to 2020, he worked as a Computer Vision Team Lead and an Architect with Valeo Vision Systems, where he also held the title of Senior Expert. In 2016, he held the position of Adjunct Lecturer with the National University of Ireland, Galway. In 2020, he joined the University of Limerick, as a Lecturer of artificial intelligence and computer vision.

PATRICK DENNY (Member, IEEE) received the B.Sc. degree in experimental physics and mathematics from the National University of Ireland (NUI), Maynooth, Ireland, in 1993, and the M.Sc. degree in mathematics and the Ph.D. degree in physics from the University of Galway, Ireland, in 1994 and 2000, respectively. He was with GFZ Potsdam, Germany. From 1999 to 2001, he was an RF Engineer with AVM GmbH, Germany, developing the RF hardware for the first integrated GSM/ISDN/USB modem. After working in supercomputing with Compaq-HP, from 2001 to 2002, he joined Connaught Electronics Ltd. (later Valeo), Galway, Ireland, as a Team Leader of RF design. For more than 20 years, he worked as a Lead Engineer, developing novel RF and imaging systems and led the development of the first mass-production HDR automotive cameras for leading car companies, including Jaguar Land Rover, BMW, and Daimler. In 2010, he became an Adjunct Professor of engineering and informatics with the University of Galway and a Lecturer of artificial intelligence with the Department of Electronic and Computer Engineering, University of Limerick, Ireland, in 2022. He is a Co-Founder and a Committee Member of the IEEE P2020 Automotive Imaging Standards Group, the AutoSens Conference on Automotive Imaging, and the IS&T Electronic Imaging Autonomous Vehicles and Machines (AVM) Conference.

ANTHONY SCANLAN received the B.Sc. degree in experimental physics from the National University of Ireland, Galway, Galway, Ireland, in 1998, and the M.Eng. and Ph.D. degrees in electronic engineering from the University of Limerick, Limerick, Ireland, in 2001 and 2005, respectively. He is currently a Senior Research Fellow with the Department of Electronic and Computer Engineering, University of Limerick, and has been a principal investigator for several research projects in the areas of signal processing and data converter design. His current research interests include artificial intelligence, computer vision, and their industrial and environmental applications.

PEPIJN VAN DE VEN received the M.Sc. degree in electronic engineering from the Eindhoven University of Technology, The Netherlands, in 2000, and the Ph.D. degree in artificial intelligence for autonomous underwater vehicles from the University of Limerick (UL), in 2005. In 2018, he joined UL's teaching staff, as a Senior Lecturer in artificial intelligence. His research interests include artificial intelligence and machine learning, with a particular interest in medical applications.

EOIN M. GRUA was born in Cork, Ireland, in 1993. He received the B.S. degree in liberal arts and sciences from Amsterdam University College, Amsterdam, The Netherlands, in 2015, the M.S. degree in computer science from Swansea University, Swansea, Wales, in 2016, and the Ph.D. degree in computer science from Vrije Universiteit Amsterdam, Amsterdam, in 2021. In 2021, he was a Research Assistant with the University of Limerick, Limerick, Ireland, where he is currently a Postdoctoral Researcher with the Department of Electronic and Computer Engineering. His research interests include artificial intelligence, software engineering and architecture, and sustainability.