DiL: An Explainable and Practical Metric for Abnormal Uncertainty in Object Detection

Amit Giloni[1*], Omer Hofman[2*], Ikuya Morikawa[3], Toshiya Shimizu[3], Yuval Elovici[1], Asaf Shabtai[1]
[1]Ben-Gurion University of the Negev
[2]Fujitsu Research of Europe
[3]Fujitsu Limited
[*]Equal contribution

Abstract

Although object detection models are widely used, their predictive performance has been shown to deteriorate when faced with abnormal scenes. Such abnormalities can occur naturally (by partially occluded or out-of-distribution objects) or deliberately (in the case of an adversarial attack). Existing uncertainty quantification methods, such as object detection evaluation metrics and label-uncertainty quantification techniques, do not consider the abnormalities' effect on the model's internal decision-making process. Furthermore, practical methods that consider the effects of abnormalities (such as abnormality detection and mitigation) are designed to deal with one type of abnormality. We present distinctive localization (DiL), an unsupervised, practical and explainable metric that quantitatively interprets any type of abnormality and can be leveraged for preventive purposes. By utilizing XAI techniques (saliency maps), DiL maps the objectness of a given scene and captures the model's inner uncertainty regarding the identified (and missed) objects. DiL was evaluated across nine use cases, including partially occluded and out-of-distribution objects, as well as adversarial patches, in both physical and digital spaces, on benchmark datasets, and our newly E-PO dataset (generated with DALL-E 2). Our results show that DiL: i) successfully interprets and quantifies an abnormality's effect on the model's decision-making process, regardless of the abnormality type; and ii) can be leveraged to detect and mitigate this effect.

1. Introduction

Object detection (OD) models play a vital role in computer vision and are widely used in numerous industries, including autonomous driving [19], retail [15], and security [18]. Although widely used, the performance of OD models has been shown to deteriorate when they are faced with an abnormal scene [29], i.e., a scene that contains conditions unfamiliar to the OD model. Such abnormalities can stem from a *natural* event (such as partially occluded [40] or out-of-distribution (OOD) objects [8]) or a *deliberate* event caused by an adversary that changes the OD model's predictions (such as an adversarial patch attack [16]).

To fairly measure the effect of these abnormalities, it is important to consider all abnormality types and their effect on both the model's predictions and the model's decision-making process (MDM). Furthermore, since the effect of abnormalities appears in new samples at inference time, a practical solution would require solely access to the model and the input sample itself. The existing solutions for measuring the effect of these abnormalities on an OD model include predictive performance measures [34] and uncertainty quantification methods [10, 13, 23, 44]. None of these methods upheld all of the above conditions. Furthermore, practical methods (e.g., abnormality detection [8, 47] and mitigation methods [8, 20, 21]) that do uphold these conditions address only one abnormality type and do not provide a quantitative measure.

We present distinctive localization (DiL), an unsupervised, practical and explainable metric that quantitatively interprets an abnormality's effect on the MDM process, regardless of the type of abnormality. Given a new scene, DiL uses explainable AI (XAI) techniques (saliency maps) to map the objectness level based on the model's internal components and examines any incompatibility with the model's output (illustrated in Figure 1). This results in a DiL score for every scene and allows DiL to interpret the model's inner uncertainty regarding its identified (and missed) objects. The DiL score can also be used for preventive purposes: *i)* a DiL threshold can be set to detect the occurrence of an abnormality, i.e., abnormality detection; and *ii)* the DiL score can be used to dynamically adjust the decision threshold for a given scene, i.e., mitigating the abnormality's effect.

We empirically evaluated DiL using two clean and seven

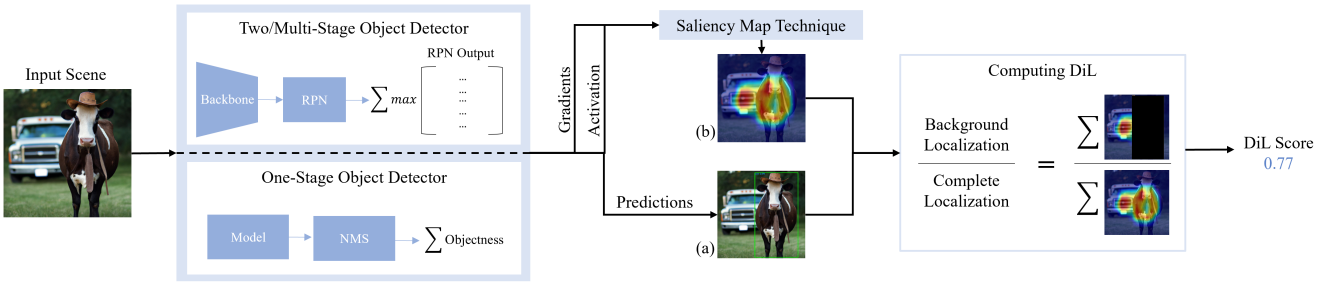

Figure 1. DiL pipeline and components. The input scene (an E-PO dataset image) is used to get (a) the model's predictions and the XAI technique's input to produce (b) a saliency map. Then, (a) and (b) are used to calculate the DiL score.

abnormality use cases in both the physical and digital space; the use cases varied in terms of the type of abnormality - partially occluded, OOD, and adversarially attacked objects. Eight OD algorithms were used in our evaluation, including one-stage (e.g., YOLOF [5]), two-stage (e.g., Faster R-CNN [38]), and multi-stage detectors (e.g., Cascade RPN [42]). In the digital evaluation, the Microsoft Common Object in Context (COCO) [28] and Occluded-PASCAL 3D+ [43] datasets were used, along with a subset of the ImageNet [7] dataset, in the clean, partial occlusion, and OOD use cases respectively, and our new E-PO dataset (created with DALL-E 2). In the physical evaluation, the Superstore [15] dataset was used, and our new PO-Superstore and OOD-Superstore datasets. For the digital and physical adversarial use cases, we crafted four adversarial patches and placed them both on images and real objects. The results of our evaluation demonstrate that DiL: *i)* successfully interprets and estimates an abnormality's effect on the MDM process, regardless of the type of abnormality; and *ii)* can be successfully leveraged to detect and mitigate the abnormality's effect on the model's outputs.

DiL's novelty lies in the unique integration, use, and adaptation of techniques from the XAI and OD fields resulting in a practical and effective uncertainty metric. The main contributions of this paper are as follows:

- To the best of our knowledge, DiL is the first metric capable of quantitatively interpreting the inner uncertainty of OD models under abnormal scenarios.
- To the best of our knowledge, DiL is the first metric that addresses all types of abnormalities, i.e., partially occluded and OOD objects, and adversarial patch attacks.
- DiL is a model-agnostic, unsupervised, and practical metric. We demonstrate that DiL can be used for abnormality analysis with one-, two- and multi-OD algorithms, does not necessitate the ground truth label, and solely requires access to the model and the input scene.
- The resources developed in this research (the code and the five new datasets) will be publicly available upon publication and can be used by the research community to further investigate abnormalities.

2. Background and Related Work

Abnormal scenes can be categorized based on their cause: a *natural* event and a *deliberate* event. Examples of the former include scenes containing: *i)* an object that is partially occluded by a another object [22, 39, 51]; or *ii)* an object from a class that was not part of the model's training set, i.e., an OOD object [8]. Both scenarios lead to the omission of a bounding box, a critical error in applications such as autonomous vehicles [8]. In contrast, abnormal scenes caused by a *deliberate* event are initiated by an adversary that aims to hide an object from the model, often via a physically placed adversarial patch attack [3].

The effect of abnormalities on an OD model is traditionally measured using existing supervised performance metrics, such as mean average precision (mAP) and its variants [33, 45], intersection over union (IoU) [27], and probability-based detection quality (PDQ) (which is suitable only for probabilistic OD) [12]. However, these metrics do not consider the abnormalities' effect on the MDM process [34]. Additionally, they presume the availability of ground truth - an impractical assumption at inference time. Newer uncertainty quantification techniques [10, 23, 44], such as Bayesian estimation [13], Monte Carlo dropout [9] and ensemble [24] approaches, provide insights on the model's label-uncertainty levels. However, these methods inherently assume that all objects are detected, which limits their applicability in abnormal scenarios that result in a not-detected object. Additionally, some of these methods assume access to the training set, which is not practical.

Existing methods that address the effect of abnormalities include practical methods that aim to detect or mitigate the abnormalities' effect, such as the effect of partial occlusion [35, 43, 50], OOD [8, 14], and adversarial patches [6, 15, 17, 20, 47, 49]. However, those methods: *1)* deal with one type of abnormality; *2)* do not quantify the inner effect of the abnormality; and *3)* lack explainability.

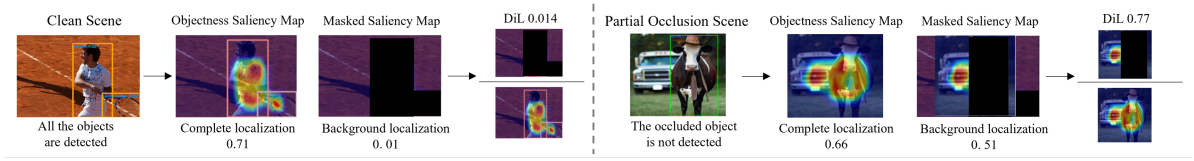

Figure 2. An illustration of computing the DiL score for a clean scene (left images) and partially occluded scene (right images).

DiL is the first practical and explainable metric to quantify the internal effect of all types of abnormalities without relying on non-practical assumptions. A table summarizing the related works is in the supplementary material.

To successfully interpret the abnormality's effect on the MDM process, DiL utilizes XAI techniques to produce a saliency-map-based interpretation of the model's decisions [25]. A saliency map is a vector matching the input scene's dimensions where elements correlating with more contributive pixels have higher values [1]. During the saliency map's creation, the XAI techniques rely on the model's activations [32], gradients [4,41], or both [11]. The quality of a saliency map is evaluated based on its fulfillment of the localization objective [25,26]. A well-localized explanation discriminates between the object and its background, showing higher saliency map values for object-related pixels. DiL is inspired by the localization objective and utilizes it for a different goal. Instead of using it to evaluate the quality of the explanation itself (the saliency map), DiL uses it to explain and evaluate the model's behavior.

3. The Method

DiL was designed based on the assumption that the MDM process will be unstable when faced with an abnormal scene. This instability will be reflected in an inconsistent presence (and absence) of an object throughout the entire OD pipeline. To capture this inconsistency for a given scene, DiL measures the relation between two stages: *i)* the model's mid-stage 'perception' (the objectness saliency map); and *ii)* the final-stage 'perception' (the predicted bounding boxes).

In this section, we introduce DiL's components, their computation process, and the final calculation of the DiL score (illustrated in Figures 1 and 2). DiL reflects the difference between the *background localization* and the *complete localization*, both of which are derived from the objectness saliency map. The *complete localization* (*CL*) implies the existence of objects in the entire scene, whereas the *background localization* (*BL*) implies the existence of objects in the areas of a scene that fall outside of any predicted bounding box. The DiL score reflects the compatibility between *BL* and *CL* values, i.e., greater compatibility implies greater abnormality (i.e., a higher DiL score) and vice versa.

The notation used is as follows: Let M be an OD model and $\mathbf{x}$ be an input scene. Let $b_i \in M(\mathbf{x})$ be the i'th bounding box in M's output for scene $\mathbf{x}$. Let M_o be M's internal component that outputs M's objectness. Let $M_A(\mathbf{x})$ and $M_\nabla(\mathbf{x})$ be M's activation and gradients respectively for processing $\mathbf{x}$ from M's first layer until M_o. Let S be a saliency map technique.

3.1. Distinctive Localization (DiL)

To compute the DiL score for a given input scene $\mathbf{x}$, one should first obtain: *i)* $M(\mathbf{x})$ predictions for input scene $\mathbf{x}$, i.e., the identified bounding boxes $b_i \in M(\mathbf{x})$ (denoted as 'a' in Figure 1); and *ii)* a saliency map of M_o's objectness. Both are used to compute the CL and BL values that correspond specifically to input scene $\mathbf{x}$. The process of obtaining the former is performed identically for all model types by querying model M. To obtain the latter (the saliency map of M_o's objectness), the internal component M_o needs to be selected.

Since different OD algorithms compute objectness differently, M_o is selected according to the type of OD algorithm used by M. In most one-stage OD algorithms (such as YOLO), objectness is computed simultaneously with other bounding box attributes, such as the object class and the associated class confidence. Then the bounding boxes are refined by the non-maximum suppression (NMS) function into a list of the final predicted bounding boxes. The objectness of the bounding boxes produced by the NMS is more concise than other objectness indications; therefore, in one-stage models, the last layer before the NMS function is selected as M_o. In one-stage OD algorithms that do not explicitly produce objectness scores, such as SSD and keypoint-based detectors (e.g., CornerNet and CenterNet) M_o can be set as the layer that produces the class logits rather than an objectness score. These logits can effectively serve as a proxy for objectness. In contrast, in two- and multi-stage OD algorithms (such as Faster R-CNN and Cascade RPN) the objectness is computed by the region proposal network (RPN) component. Then the RPN's output (i.e., the objectness) is processed by additional components (such as NMS), which vary depending on the algorithm used. Therefore, in two- and multi-stage models, M_o is defined as the final layer of the RPN component (e.g., before the NMS).

After M_o has been selected, the predefined saliency map technique S is used to compute the saliency map of M_o's

objectness according to S. When S is an activation-based technique, S receives $M_A(\mathbf{x})$ (the activations of M until M_o) as input, regardless of the type of OD algorithm used by M. However, when S is a gradient-based technique, S receives $M_\nabla(\mathbf{x})$ (the gradients of M until M_o) as input, which is calculated by backpropagating from M_o's objectness to $\mathbf{x}$. Note that an aggregation technique compresses the objectness scores into a single value to apply backpropagation. This aggregation's calculation depends on the OD algorithm type used by M. When M is a one-stage model, the objectness scores produced by M_o are summed into a single value. This aggregation and the calculation of $M_\nabla(\mathbf{x})$ are detailed in Equation 1:

$$M_\nabla(\mathbf{x}) = \nabla_{\mathbf{x}} \left(\sum_{o_i \in M_o(\mathbf{x})} o_i \right) \tag{1}$$

where $M_o(\mathbf{x})$ are the outputs of M_o obtained when M is queried with $\mathbf{x}$. When M is a two- or multi-stage model, M_o outputs several vectors, each of which indicates a region of interest that contains several bounding box candidates. To aggregate M_o outputs into a single scalar value, the most indicative bounding box of each region is selected (the one that has the highest objectness score), and then the regions' objectness scores are summed. This aggregation and the calculation of $M_\nabla(\mathbf{x})$ are presented in Equation 2:

$$M_\nabla(\mathbf{x}) = \nabla_{\mathbf{x}} \left(\sum_{r_i \in M_o(\mathbf{x})} max\left(bb \in r_i\right) \right) \tag{2}$$

where $M_o(\mathbf{x})$ are the outputs of M_o obtained when M is queried with $\mathbf{x}$ and $bb \in r_i$ is a bounding box in region r_i. The use of summing as the aggregation technique ensures that no region of interest is overlooked, which results in gradients that are more indicative for S. The resulting saliency map of M_o's objectness is the vector $SM_{\mathbf{x}}$, which has the same dimensions as $\mathbf{x}$ and reflects the objectness level of every pixel. $SM_{\mathbf{x}}$ can be visualized as a map in which higher values appear in warmer colors ('b' in Figure 1).

After obtaining M's predictions ($b_i \in M(\mathbf{x})$) and the objectness saliency map $SM_{\mathbf{x}}$, the values of DiL's components can be calculated. The DiL score is the division of the *BL* by the *CL* of input scene $\mathbf{x}$. The latter (*CL* value) is calculated by summing all values in $SM_{\mathbf{x}}$, indicating the existence of objects in the entire scene. The calculation of *CL value* is presented in Equation 3:

$$CL(\mathbf{x}) = \sum_{sm_i \in SM_{\mathbf{x}}} sm_i \tag{3}$$

where $CL(\mathbf{x})$ is the *CL* value of input scene $\mathbf{x}$. The *BL* value is computed by summing the $SM_{\mathbf{x}}$ values that relate to pixels that were predicted as 'background' by M, i.e., pixels that are not bounded by any predicted bounding box. The *BL value* calculation is presented in Equation 4:

$$BL(\mathbf{x}) = \sum_{sm_i \in SM_{\mathbf{x}}} \begin{cases} sm_i & sm_i \in BG \\ 0 & sm_i \notin BG \end{cases} \tag{4}$$

where $BL(\mathbf{x})$ is the *BL* value of input scene $\mathbf{x}$ and BG are pixels that are not bounded by any predicted bounding box $b_i \in M(\mathbf{x})$, i.e., background pixels. The calculation of the final DiL score is presented in Equation 5:

$$DiL(\mathbf{x}) = \frac{Background_L(\mathbf{x})}{Complete_L(\mathbf{x})} \tag{5}$$

This calculation reflects the relation between the *BL* value and the *CL* value. Since the *BL* is a subset of the *CL*, the *BL* value is invariably less than the *CL* value; thus, the DiL score ranges from $(0, 1)$. When the *BL* value is far from the *CL* value (i.e., a low DiL score), there are more indications for objects inside the predicted bounding boxes than in the background. Therefore, model M's perception of $\mathbf{x}$ did not change during M's pipeline, i.e., there is no change between the perception gained by M_o and the model's final prediction. However, a close *BL* and *CL* values (i.e., a high DiL score) indicate that the MDM process was irregular. This irregularity, indicated by a high DiL score, arises when the values of $sm_i \in BG$ are close to the values of $sm_i \notin BG$, reflecting a shift between the initial perception by M_o and the final model's prediction. Furthermore, DiL's behavior reflects the localization objective fulfillment (Section 2), i.e., better localization in the objectness saliency map will result in a lower DiL score and vice versa.

3.2. Practical Applications of DiL

3.2.1 DiL for Model Evaluation

The DiL metric can serve as an evaluation measure that reflects a model's nature and robustness. When using traditional performance metrics to evaluate a model (such as mAP), the assessment is performed for the model's final prediction, ignoring the MDM process. By computing the average DiL score for abnormal scenes, DiL can serve as an evaluation metric. It can differentiate between two models exhibiting the same predictive behavior; e.g., one might have a better MDM process, which indicates that the model is more robust to abnormalities. The results of DiL for model evaluation can be seen at the beginning of Section 5.

3.2.2 DiL for Detection

DiL can be leveraged for detecting abnormal scenes, i.e., partially occluded and OOD objects, or an adversarial patch. Such detection can be done by setting a DiL threshold that distinguishes clean from abnormal scenes. A scene

with a DiL score above the DiL threshold will be detected as an abnormal scene. This DiL threshold can be set empirically or based on the context of use [15]. The results of DiL for detection can be seen in Figure 5 within Section 5.

3.2.3 DiL for Robustness

DiL can be leveraged to improve the model's robustness when faced with abnormal scenes. Abnormal scenes can cause a reduction in the model's confidence in the predicted object. Therefore, DiL can be incorporated into the model's inference process by changing the fixed decision threshold (FDT) to a dynamic decision threshold (DDT). The DDT will be lower than the FDT when the DiL score is higher and vice versa, i.e., when the model indicates an inconsistency in the MDM process (a high DiL score), the DDT will be lower. The DDT is calculated by subtracting the product of the DiL score and a degradation factor from the FDT. The degradation factor ensures that the decision threshold remains within a reasonable range to avoid performance degradation in clean scenes. The DDT calculation is presented in Equation 6:

$$DDT_{\mathbf{x}} = FDT - (DiL(\mathbf{x}) \cdot \alpha) \tag{6}$$

where $DDT_{\mathbf{x}}$ is the adjusted decision threshold for an input scene $\mathbf{x}$ and α is the degradation factor. The results of DiL for robustness can be seen in Table 3 within Section 5.

4. Evaluation

4.1. Evaluation Use Cases

Table 1 presents the nine use cases used to evaluate DiL and their corresponding datasets, the evaluated abnormality, and the evaluation space. The use cases were defined to include both physical and digital evaluations of all evaluated abnormalities. The digital evaluations (use cases 1-5) were performed on models trained on the COCO dataset. The physical evaluations (use cases 6-9) were performed on models trained on the Superstore dataset.

4.2. Datasets

The evaluation was performed using these datasets:

Microsoft Common Object in Context (COCO) 2017 [28] – an OD benchmark dataset containing 80 object classes and over 120K labeled images.

Superstore [15] – an OD in retail dataset containing 2,200 labeled images of 20 superstore items (classes) presented from a smart shopping cart perspective.

OccludedPASCAL 3D+ [43] – an OD dataset that simulates partial occlusion by digitally placing objects cropped from the COCO dataset on top of objects from the OccludedPASCAL3D+ dataset [48].

ImageNet [7] – a small subset of the ImageNet dataset that includes 100 images from 27 classes serves as OOD for the models trained on the COCO dataset.

Additionally, we created the following datasets:

E-PO – a realistic OD dataset containing 100 images of occluded objects related to the classes in the COCO dataset. This dataset was generated with the assistance of DALL-E 2 [36] and contains real-looking images with natural-looking partial occlusions.

Adv-COCO – an OD dataset containing 100 images from the COCO dataset that were attacked. We digitally placed an adversarial patch, crafted based on the DPatch attack [30], on objects in the images, deliberately causing them to be "hidden" from the model. Due to the patches' low transferability, we created two versions of the Adv-COCO dataset using a different adversarial patch. Each patch was crafted to mislead different models in each version - either one- or two/multi-stage OD models.

PO-Superstore – an OD dataset with 100 images of occluded objects related to the classes in the Superstore dataset. The superstore items were physically placed to occlude one another simulating real-world partial occlusions.

OOD-Superstore Dataset – an OD dataset containing 100 images of superstore items that do not appear in the Superstore dataset. This dataset can be considered a real-world OOD scenario in the retail domain.

Adv-Superstore Dataset – an OD dataset containing 100 images of attacked objects related to the classes in the Superstore dataset. This dataset was created in the same manner as the Adv-COCO dataset with the exception of placing the adversarial patches physically on the items.

4.3. Experimental Settings

We used two sets of models according to the evaluated use case. In both sets, the models had identical parameters which were set as follows. Each set includes the following OD models. One-stage models: *1)* YOLOv3 [37] with a Darknet-53 backbone; *2)* YOLOv5X with a CSPDarknet53 backbone and a YOLOv3 head; and *3)* YOLOF [5] with a ResNet50 backbone and a dilated encoder that serves as an FPN. The decision thresholds were set as 0.5 and 0.7 in the one-stage models trained on the COCO and Superstore datasets respectively. Two-stage models: 1) Faster R-CNN [38] with a ResNet50 backbone and FPN architecture; 2) Grid R-CNN [31] with a ResNet50 backbone and grid guided localization mechanism; and 3) Double Head R-CNN [46] with a ResNet50 backbone and two-head structure output. Multi-stage models: 1) Cascade R-CNN [2] with a ResNet50 backbone and a sequence of detectors trained with increasing IoU thresholds; and 2) Cascade RPN [42] with a ResNet50 backbone and multi-stage anchoring refinement. The decision thresholds were set as 0.8 and 0.7 in the two-stage and multi-stage models trained on the COCO and Superstore datasets respectively. The ad-

No.	Use case name	Model training set	Evaluation space	Abnormality type	Abnormality sub-type	Evaluation set
1	COCO clean	COCO train	Digital	None	None	COCO validation
2	COCO unrealistic PO			Natural	Partial occlusion	OccludedPASCAL 3D+
3	COCO realistic PO					E-PO Dataset
4	COCO OOD				OOD	Subset of ImageNet
5	COCO Adv.			Deliberate	Adversarial	Adv-COCO
6	Superstore clean	Superstore train	Physical	None	None	Superstore validation
7	Superstore PO			Natural	Partial occlusion	PO-Superstore
8	Superstore OOD				OOD	OOD-Superstore
9	Superstore Adv.			Deliberate	Adversarial	Adv-Superstore

Table 1. Use cases examined.

Model Type	Metric	Use case								
		[1] Clean	[2] Unrealistic PO	[3] Realistic PO	[4] OOD	[5] Adv.	[6] Clean	[7] PO	[8] OOD	[9] Adv.
One-stage	Misclassification Rate	–	0.89 ↓	0.87 ↓	0.63 ↓	0.85 ↓	–	0.97 ↓	0.7 ↓	1.00 ↓
	CL	0.011 ↑	0.01 ↓	0.0047 ↓	0.0033 ↓	0.016 ↓	0.017 ↑	0.0105 ↓	0.019 ↓	0.02 ↓
	BL	0.0024 ↓	0.0046 ↑	0.0025 ↑	0.003 ↑	0.008 ↑	0.0004 ↓	0.008 ↑	0.0106 ↑	0.006 ↑
	DiL	**0.216** ↓	**0.475** ↑	**0.53** ↑	**0.93** ↑	**0.519** ↑	**0.03** ↓	**0.413** ↑	**0.494** ↑	**0.342** ↑
Two-stage	Misclassification Rate	–	0.93 ↓	0.93 ↓	0.83 ↓	0.85 ↓	–	0.77 ↓	0.9 ↓	0.72 ↓
	CL	0.114 ↑	0.155 ↓	0.149 ↓	0.089 ↓	0.068 ↓	0.039 ↑	0.0223 ↓	0.061 ↓	0.014 ↓
	BL	0.011 ↓	0.091 ↑	0.065 ↑	0.08 ↑	0.0318 ↑	0.004 ↓	0.009 ↑	0.046 ↑	0.002 ↑
	DiL	**0.122** ↓	**0.59** ↑	**0.449** ↑	**0.908** ↑	**0.617** ↑	**0.11** ↓	**0.396** ↑	**0.756** ↑	**0.793** ↑
Multi-stage	Misclassification Rate	–	0.92 ↓	0.84 ↓	0.84 ↓	0.76 ↓	–	0.93 ↓	0.53 ↓	0.75 ↓
	CL	0.114 ↑	0.1549 ↓	0.1493 ↓	0.087 ↓	0.066 ↓	0.0395 ↑	0.024 ↓	0.06 ↓	0.009 ↓
	BL	0.014 ↓	0.0856 ↑	0.064 ↑	0.0779 ↑	0.0269 ↑	0.00525 ↓	0.05 ↑	0.055 ↑	0.036 ↑
	DiL	**0.125** ↓	**0.556** ↑	**0.4515** ↑	**0.895** ↑	**0.535** ↑	**0.13** ↓	**0.39** ↑	**0.94** ↑	**0.78** ↑
All types	DiL mean	**0.154** ↓	**0.54** ↑	**0.497** ↑	**0.911** ↑	**0.557** ↑	**0.09** ↓	**0.39** ↑	**0.73** ↑	**0.63** ↑
All types	mAP	0.359 ↑	0.23* ↓	0.253 ↓	0.0 ↓	0.212 ↓	0.9 ↑	0.5 ↓	0.5 ↓	0.01 ↓

Table 2. Mean DiL scores for all types of OD models in the digital COCO (1-5) and physical Superstore use cases (6-9).

versarial patches used in use cases 5 and 9 were crafted with the DPatch attack [30]. More details on the crafting process are provided in the supplementary material.

The Grad-CAM [41] XAI technique was used to evaluate DiL as a metric and as an abnormality detector. Grad-CAM++ [4] was employed to assess DiL as a robustness solution. For more details on the selection of saliency map techniques and related experiments see supplementary material. In the evaluation of DiL as a robustness solution, we empirically set the degradation factor to 0.2.

5. Experimental Results

Detailed experimental implementations, comprehensive results, and DiL's runtime analysis are available in the supplementary materials. The following section provides a summary of the key findings. Table 2 shows the results obtained by the one-, two-, and multi-stage OD models for all nine use cases (Section 4.1). Each row lists the mean misclassification rate (fraction of incorrectly classified scenes), *CL*, *BL*, DiL scores, and mean average precision (mAP) as a representative of the existing methods for the evaluated models. The arrow in each cell indicates whether a high or low value is the optimal value.

The misclassification rate results presented in the table show that the OD models struggle to detect objects across all types of abnormalities, underscoring their potential threat to OD model performance. Moreover, these results indicate that multi-stage models are more resilient to PO and adversarial patches, while one-stage models are more resilient to OOD scenes. Additionally, the results show that overall, DiL succeeds in reflecting the difference in the MDM process when faced with clean and abnormal scenes, i.e., the DiL scores obtained in the clean scenes are lower than the scores obtained in the abnormal scenes. The DiL scores for the first and sixth use cases (clean use cases) range in $[0.122, 0.216]$ and $[0.03, 0.13]$ respectively. In contrast, the DiL scores for use cases 2-5 and 7-9 (abnormal use cases) range in $[0.449, 0.93]$ and $[0.342, 0.94]$ respectively. Moreover, the results show that DiL aligns with the supervised mAP metric, distinguishing between clean and abnormal scenes in an unsupervised manner without reliance on ground truth data. In addition, the results show that the DiL scores are more stable compared to the mAP score with respect to the relations between different abnormalities (see supplementary material).

Furthermore, we can see that neither the *CL* nor the *BL* metric can be used independently as an abnormality indicator. For example, the *CL* value of the one-stage models in the sixth use case (a clean use case) is 0.017, and in the eighth use case (an abnormal use case) the *CL* value is 0.019, i.e., very close values like these would not enable differentiation between clean and abnormal scenes.

Moreover, the results indicate that the type of abnormality influences the value of the DiL score. In use cases 2-5 (the COCO use cases), the mean DiL score for the PO use cases is 0.5185, which is the lowest mean score obtained

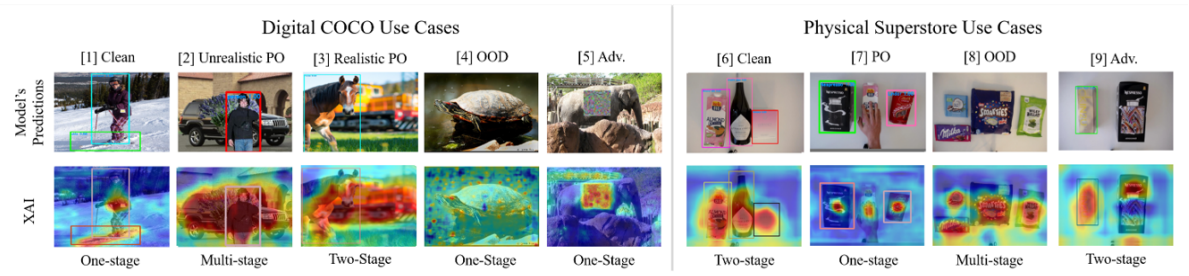

Figure 3. Qualitative assessment of DiL (the model's predictions and the corresponding saliency maps).

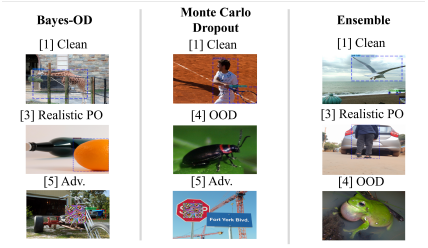

Figure 4. The uncertainty techniques outputs when faced with an abnormality. None of the abnormalities was detected.

in all the use cases; the mean DiL score for the adversarial use case is 0.557; and the mean DiL score for the OOD use case is 0.911, which is the highest obtained in all the use cases. The same phenomena can be seen in the Superstore use cases (7-9). This can be explained by the different levels of alienation between the examined abnormality and the distribution of normal scenes. In the PO and adversarial use cases, the non-occluded/attacked objects and backgrounds presented were taken from the normal scenes' distribution (i.e., low alienation), whereas that is not the case in the OOD use cases (i.e., high alienation). Furthermore, in the PO use cases, the objects that were used to cover the occluded object were from the normal scenes' distribution (i.e., low alienation), whereas, in the adversarial use cases, the adversarial patches used were not (i.e., high alienation). The results also show that DiL scores for realistic PO use-case are lower compared to unrealistic PO use-case, likely due to their greater deviation from natural scene distributions (see supplementary material).

Figure 3 shows a qualitative evaluation of DiL- examples of model predictions along with their corresponding saliency maps for each use case. The saliency maps visually explain the abnormality's effect on the MDM process. The examples illustrate the different models' inner perceptions of different types of abnormalities. In the PO and OOD use cases, the saliency map accurately locates the partially occluded/OOD object, whereas the model's prediction did not, resulting in a high DiL score. In adversarial use cases, the patch disrupts the model's inner perception by causing one-stage models to fixate on the patch and diverting attention in two/multi-stage models. This results in unusual saliency maps focused only on the patch area, leading to a high DiL score. Figure 4 shows output examples across various use cases obtained from three label-uncertainty techniques: Bayes-OD [13], Monte-Carlo Dropout [9], and Model's Ensemble [24]. These examples reveal that while existing techniques quantify uncertainty for detected objects, they disregard the missed ones. As these techniques are designed to assess label uncertainty, they are applied at the prediction process's final stages, such as the ROI pooling layer in Faster-RCNN. However, the abnormalities impact the outputs of earlier stages of the prediction process, causing the model to fail to 'propose' an object for further processing, rendering these techniques ineffective. For more information on the uncertainty techniques implementations, along with a quantitative evaluation, see supplementary material.

Figure 5 presents the mean detection results for all evaluated models using DiL as a detector (Section 3.2). The upper plots present the true positive rate (TPR) and false negative rate (FNR) for all evaluated abnormalities when using different DiL percentile cutoffs (PCs) as the detection threshold. A scene with a DiL score above the PC would be detected as an abnormality. The lower plots display the true negative rate (TNR) and false positive rate (FPR) for clean use cases using varying DiL PCs. Figure 5 shows that overall, DiL effectively differentiates between clean and abnormal scenes. In the TPR and FNR plots, we can see that a lower PC threshold increases TPR and decreases FNR, i.e., a lower PC threshold improves the detection of abnormalities. However, in the TNR and FPR plots, we see the opposite phenomenon - a lower PC threshold decreases the ability to identify normal scenes. Notably, DiL's most effective detection is observed in OOD use cases, consistent with the high DiL scores for OOD cases shown in Table 2.

Table 3 presents the results obtained when DiL was used as a robustness method, showing average DDT results for all model types across all nine use cases (detailed results in the supplementary material). Each cell presents the recall, precision, F1 score, and TPR improvement for each model when using DDT. Since an abnormality's effect on

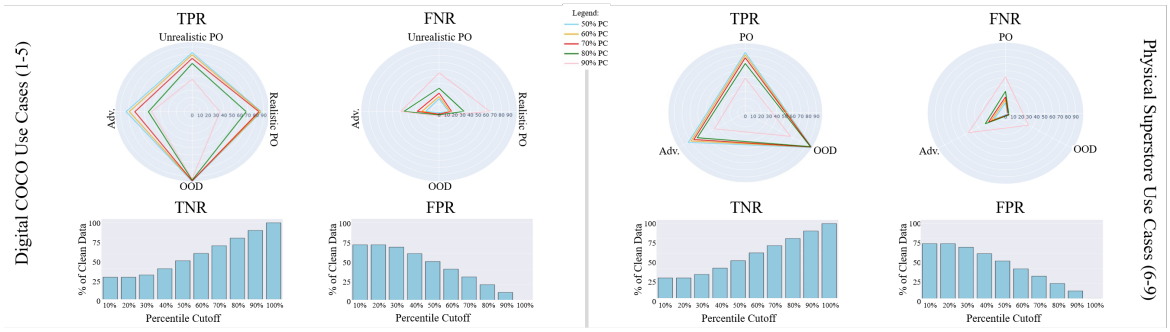

Figure 5. DiL's performance as an abnormality detector. The TPR and FNR (spider plots) and the TNR and FPR (bar plots).

Metric	Use case and abnormality								
	[1] Clean	[2] Unrealistic PO	[3] Realistic PO	[4] OOD	[5] Adv.	[6] Clean	[7] PO	[8] OOD	[9] Adv.
Recall	0.671 (**+6.6%**)	0.712 (**+39.6%**)	0.615 (**+30%**)	0.36 (**+50%**)	0.571 (**+17.6%**)	0.908 (**0.3%**)	0.622 (**+6.6%**)	0.43 (**+260%**)	0.67 (**+9.6%**)
Precision	0.943 (**+0.0%**)	0.919 (**+3.4%**)	0.946 (**+1.6%**)	0.515 (**+3.2%**)	0.918 (**+1.3%**)	0.987 (**+0.007%**)	0.993 (**+0.16%**)	0.95 (**+0.6%**)	0.958 (**+0.8%**)
F1 Score	0.782 (**+0.0%**)	0.787 (**+15%**)	0.737 (**+10%**)	0.419 (**+4.1%**)	0.703 (**+6.8%**)	0.946 (**+0.5%**)	0.748 (**+7.3%**)	0.554 (**+5.1%**)	0.785 (**+5.7%**)
TPR improvement	7.6%	38%	25%	14.6%	12.3%	4%	20%	13.3%	9.6%

Table 3. Original and DDT performance for all OD model types and use cases.

the model's output is to "hide" an object, we expect to see the highest improvement in the recall metric (the mean percentage of objects localized by the model). The results presented in Table 3 show DDT generally mitigates the abnormalities' effect without harming and even improving the model's performance (F1 score). In COCO abnormality use cases (2-5), the model recall improvement ranged from 15% to 68%; TPR improvement from 3% to 52%; and F1 score improvement in $[4.1\%, 15\%]$. Similarly, in Superstore abnormality use cases (7-9), recall improvement ranged from 2% to 260%; TPR improvement from 5% to 31%, and F1 score improvement in $[0.5\%, 7.3\%]$.

6. Discussion

The effectiveness of DiL in uncertainty quantification relies on selecting a particularly informative layer within the object detection model: the layer responsible for generating the objectness score. This layer captures the model's earliest assessments of object presence and influences initial bounding box predictions. To support our layer choice, we conducted a qualitative experiment using saliency map visualizations across various layers (see Figure 6 (A)) showing that the selected layer provides a clear understanding of the MDM process in different scenarios.

While DiL has been consistent across many scenarios, its performance may vary when dealing with extremely small objects. Since DiL relies on a saliency map technique, the quality of the DiL scores depends on the sensitivity of the saliency map technique used. In our experiments, we benchmarked numerous saliency map techniques and empirically showed that the Grad-CAM technique had the most balanced sensitivity. However, for scenarios requiring greater sensitivity, such as detecting small objects, Grad-CAM++ is recommended due to its use of second-order gradients, which yield more detailed saliency maps (see Figure 6 (B)). Further details and robustness analysis of the objectness saliency map are in the supplementary material.

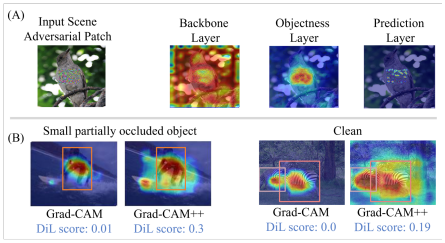

Figure 6. (A) Saliency map visualization across layers. (B) example saliency maps for small and regular objects.

7. Conclusions and Future Work

In this paper, we presented distinctive localization (DiL), an unsupervised explainable metric that captures the inner uncertainty of an OD model when faced with abnormal scenes. In our experiments on various types of abnormalities, we showed that, in contrast to existing methods, DiL requires practical assumptions and can: *i)* reflect the model's inner uncertainty, both quantitatively and visually; *ii)* explain the DiL score produced; and *iii)* be leveraged for preventive actions. Future work may include DiL evaluation on other algorithms (e.g., transformers), tasks (e.g., object segmentation), and attacks (adversarial perturbations and backdoors). Additional future work may include enhancing the abnormality mitigation using DiL by setting a smart degradation factor. This can be done by analyzing the evidence for missing objects in the objectness saliency map.

References

[1] Amina Adadi and Mohammed Berrada. Peeking inside the black-box: a survey on explainable artificial intelligence (xai). *IEEE access*, 6:52138–52160, 2018. 3

[2] Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: high quality object detection and instance segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 43(5):1483–1498, 2019. 5

[3] Anirban Chakraborty, Manaar Alam, Vishal Dey, Anupam Chattopadhyay, and Debdeep Mukhopadhyay. A survey on adversarial attacks and defences. *CAAI Transactions on Intelligence Technology*, 6(1):25–45, 2021. 2

[4] Aditya Chattopadhay, Anirban Sarkar, Prantik Howlader, and Vineeth N Balasubramanian. Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. In *2018 IEEE winter conference on applications of computer vision (WACV)*, pages 839–847. IEEE, 2018. 3, 6

[5] Qiang Chen, Yingming Wang, Tong Yang, Xiangyu Zhang, Jian Cheng, and Jian Sun. You only look one-level feature. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2021. 2, 5

[6] Ping-Han Chiang, Chi-Shen Chan, and Shan-Hung Wu. Adversarial pixel masking: A defense against physical attacks for pre-trained object detectors. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 1856–1865, 2021. 2

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 2, 5

[8] Xuefeng Du, Xin Wang, Gabriel Gozum, and Yixuan Li. Unknown-aware object detection: Learning what you don't know from videos in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13678–13688, 2022. 1, 2

[9] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pages 1050–1059. PMLR, 2016. 2, 7

[10] Stefano Gasperini, Jan Haug, Mohammad-Ali Nikouei Mahani, Alvaro Marcos-Ramiro, Nassir Navab, Benjamin Busam, and Federico Tombari. Certainnet: Sampling-free uncertainty estimation for object detection. *IEEE Robotics and Automation Letters*, 7(2):698–705, 2021. 1, 2

[11] Jacob Gildenblat and contributors. Pytorch library for cam methods. https://github.com/jacobgil/pytorch-grad-cam, 2021. 3

[12] David Hall, Feras Dayoub, John Skinner, Haoyang Zhang, Dimity Miller, Peter Corke, Gustavo Carneiro, Anelia Angelova, and Niko Sunderhauf. Probabilistic object detection: Definition and evaluation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1031–1040, 2020. 2

[13] Ali Harakeh, Michael Smart, and Steven L Waslander. Bayesod: A bayesian approach for uncertainty estimation in deep object detectors. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 87–93. IEEE, 2020. 1, 2, 7

[14] Vahid Hashemi, Jan Křetínskỳ, Sabine Rieder, and Jessica Schmidt. Runtime monitoring for out-of-distribution detection in object detection neural networks. In *International Symposium on Formal Methods*, pages 622–634. Springer, 2023. 2

[15] Omer Hofman, Amit Giloni, Yarin Hayun, Ikuya Morikawa, Toshiya Shimizu, Yuval Elovici, and Asaf Shabtai. X-detect: Explainable adversarial patch detection for object detectors in retail. *arXiv preprint arXiv:2306.08422*, 2023. 1, 2, 5

[16] Yu-Chih-Tuan Hu, Bo-Han Kung, Daniel Stanley Tan, Jun-Cheng Chen, Kai-Lung Hua, and Wen-Huang Cheng. Naturalistic physical adversarial patch for object detectors. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7848–7857, 2021. 1

[17] Nan Ji, YanFei Feng, Haidong Xie, Xueshuang Xiang, and Naijin Liu. Adversarial yolo: Defense human detection patch attacks via detecting adversarial patches. *arXiv preprint arXiv:2103.08860*, 2021. 2

[18] SivaNagiReddy Kalli, T Suresh, A Prasanth, T Muthumanickam, and K Mohanram. An effective motion object detection using adaptive background modeling mechanism in video surveillance system. *Journal of Intelligent & Fuzzy Systems*, (Preprint):1–13, 2021. 1

[19] Esraa Khatab, Ahmed Onsy, Martin Varley, and Ahmed Abouelfarag. Vulnerable objects detection for autonomous driving: A review. *Integration*, 78:36–48, 2021. 1

[20] Taeheon Kim, Youngjoon Yu, and Yong Man Ro. Defending physical adversarial attack on object detection via adversarial patch-feature energy. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 1905–1913, 2022. 1, 2

[21] Adam Kortylewski, Qing Liu, Angtian Wang, Yihong Sun, and Alan Yuille. Compositional convolutional neural networks: A robust and interpretable model for object recognition under occlusion. *International Journal of Computer Vision*, 129:736–760, 2021. 1

[22] Adam Kortylewski, Qing Liu, Huiyu Wang, Zhishuai Zhang, and Alan Yuille. Combining compositional models and deep networks for robust object classification under occlusion. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 1333–1341, 2020. 2

[23] Florian Kraus and Klaus Dietmayer. Uncertainty estimation in one-stage object detection. In *2019 ieee intelligent transportation systems conference (itsc)*, pages 53–60. IEEE, 2019. 1, 2

[24] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in neural information processing systems*, 30, 2017. 2, 7

[25] Xuhong Li, Haoyi Xiong, Xingjian Li, Xuanyu Wu, Xiao Zhang, Ji Liu, Jiang Bian, and Dejing Dou. Interpretable deep learning: Interpretation, interpretability, trustworthiness, and beyond. *Knowledge and Information Systems*, 64(12):3197–3234, 2022. 3

[26] Xiao-Hui Li, Yuhan Shi, Haoyang Li, Wei Bai, Caleb Chen Cao, and Lei Chen. An experimental study of quantitative evaluations on saliency methods. In *Proceedings of the 27th ACM sigkdd conference on knowledge discovery & data mining*, pages 3200–3208, 2021. 3

[27] Tsung-Yi Lin. Y, dollár p, girshick r, he k, hariharan b, belongie s. feature pyramid networks for object detection. In *Proceedings-30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, volume 2017, pages 936–944, 2017. 2

[28] 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*, pages 740–755. Springer, 2014. 2, 5

[29] Li Liu, Wanli Ouyang, Xiaogang Wang, Paul Fieguth, Jie Chen, Xinwang Liu, and Matti Pietikäinen. Deep learning for generic object detection: A survey. *International journal of computer vision*, 128:261–318, 2020. 1

[30] Xin Liu, Huanrui Yang, Ziwei Liu, Linghao Song, Hai Li, and Yiran Chen. Dpatch: An adversarial patch attack on object detectors. *arXiv preprint arXiv:1806.02299*, 2018. 5, 6

[31] Xin Lu, Buyu Li, Yuxin Yue, Quanquan Li, and Junjie Yan. Grid r-cnn. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 5

[32] Mohammed Bany Muhammad and Mohammed Yeasin. Eigen-cam: Class activation map using principal components. In *2020 international joint conference on neural networks (IJCNN)*, pages 1–7. IEEE, 2020. 3

[33] Kemal Oksuz, Baris Can Cam, Emre Akbas, and Sinan Kalkan. Localization recall precision (lrp): A new performance metric for object detection. In *Proceedings of the European conference on computer vision (ECCV)*, pages 504–519, 2018. 2

[34] Rafael Padilla, Sergio L Netto, and Eduardo AB Da Silva. A survey on performance metrics for object-detection algorithms. In *2020 international conference on systems, signals and image processing (IWSSIP)*, pages 237–242. IEEE, 2020. 1, 2

[35] Lu Qi, Li Jiang, Shu Liu, Xiaoyong Shen, and Jiaya Jia. Amodal instance segmentation with kins dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3014–3023, 2019. 2

[36] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents, 2022. *URL https://arxiv.org/abs/2204.06125*, 7, 2022. 5

[37] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018. 5

[38] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015. 2, 5

[39] Jiageng Ruan, Hanghang Cui, Yuhan Huang, Tongyang Li, Changcheng Wu, and Kaixuan Zhang. A review of occluded objects detection in real complex scenarios for autonomous driving. *Green Energy and Intelligent Transportation*, page 100092, 2023. 2

[40] Kaziwa Saleh, Sándor Szénási, and Zoltán Vámossy. Occlusion handling in generic object detection: A review. In *2021 IEEE 19th World Symposium on Applied Machine Intelligence and Informatics (SAMI)*, pages 000477–000484. IEEE, 2021. 1

[41] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017. 3, 6

[42] Thang Vu, Hyunjun Jang, Trung X Pham, and Chang Yoo. Cascade rpn: Delving into high-quality region proposal network with adaptive convolution. *Advances in neural information processing systems*, 32, 2019. 2, 5

[43] Angtian Wang, Yihong Sun, Adam Kortylewski, and Alan L Yuille. Robust object detection under occlusion with context-aware compositionalnets. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12645–12654, 2020. 2, 5

[44] Zining Wang, Di Feng, Yiyang Zhou, Lars Rosenbaum, Fabian Timm, Klaus Dietmayer, Masayoshi Tomizuka, and Wei Zhan. Inferring spatial uncertainty in object detection. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5792–5799. IEEE, 2020. 1, 2

[45] Everingham M Van Gool L Williams. Ck winn j zisserman a the pascal visual object classes (voc) challenge. *Int. J. Comput. Vis*, 88(2):303, 2010. 2

[46] Yue Wu, Yinpeng Chen, Lu Yuan, Zicheng Liu, Lijuan Wang, Hongzhi Li, and Yun Fu. Rethinking classification and localization for object detection. 2019. 5

[47] Chong Xiang and Prateek Mittal. Detectorguard: Provably securing object detectors against localized patch hiding attacks. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, pages 3177–3196, 2021. 1, 2

[48] Yu Xiang, Roozbeh Mottaghi, and Silvio Savarese. Beyond pascal: A benchmark for 3d object detection in the wild. In *IEEE winter conference on applications of computer vision*, pages 75–82. IEEE, 2014. 5

[49] Ke Xu, Yao Xiao, Zhaoheng Zheng, Kaijie Cai, and Ram Nevatia. Patchzero: Defending against adversarial patch attacks by detecting and zeroing the patch. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 4632–4641, 2023. 2

[50] Xiaohang Zhan, Xingang Pan, Bo Dai, Ziwei Liu, Dahua Lin, and Chen Change Loy. Self-supervised scene deocclusion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3784–3792, 2020. 2

[51] Hongru Zhu, Peng Tang, Jeongho Park, Soojin Park, and Alan Yuille. Robustness of object recognition under extreme occlusion in humans and computational models. *arXiv preprint arXiv:1905.04598*, 2019. 2