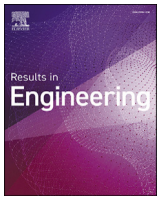
Results in Engineering 24 (2024) 103498

Contents lists available at ScienceDirect

Results in Engineering

journal homepage: www.sciencedirect.com/journal/results-in-engineering



Research paper

On the black-box explainability of object detection models for safe and trustworthy industrial applications

Alain Andres [a,b,*], Aitor Martinez-Seras [a], Ibai Laña [a,b], Javier Del Ser [a,c]

[a] TECNALIA, Basque Research and Technology Alliance (BRTA), Mikeletegi Pasealekua 2, Donostia-San Sebastian, 20009, Spain
[b] University of Deusto, 20012, Donostia-San Sebastián, Spain
[c] University of the Basque Country (UPV/EHU), Bilbao, 48013, Spain

ARTICLE INFO

Keywords:
Explainable Artificial Intelligence
Safe Artificial Intelligence
Trustworthy Artificial Intelligence
Object detection
Single-stage object detection
Industrial robotics

ABSTRACT

In the realm of human-machine interaction, artificial intelligence has become a powerful tool for accelerating data modeling tasks. Object detection methods have achieved outstanding results and are widely used in critical domains like autonomous driving and video surveillance. However, their adoption in high-risk applications, where errors may cause severe consequences, remains limited. Explainable Artificial Intelligence methods aim to address this issue, but many existing techniques are model-specific and designed for classification tasks, making them less effective for object detection and difficult for non-specialists to interpret. In this work we focus on *model-agnostic* explainability methods for object detection models and propose D-MFPP, an extension of the Morphological Fragmental Perturbation Pyramid (MFPP) technique based on segmentation-based masks to generate explanations. Additionally, we introduce D-Deletion, a novel metric combining faithfulness and localization, adapted specifically to meet the unique demands of object detectors. We evaluate these methods on real-world industrial and robotic datasets, examining the influence of parameters such as the number of masks, model size, and image resolution on the quality of explanations. Our experiments use single-stage object detection models applied to two safety-critical robotic environments: i) a shared human-robot workspace where safety is of paramount importance, and ii) an assembly area of battery kits, where safety is critical due to the potential for damage among high-risk components. Our findings evince that D-Deletion effectively gauges the performance of explanations when multiple elements of the same class appear in a scene, while D-MFPP provides a promising alternative to D-RISE when fewer masks are used.

1. Introduction

In recent years, Artificial Intelligence (AI) has emerged as a transformative force across various domains, especially in human-machine interaction, where it has enabled significant advancements in data-driven decision-making processes. Among these advances, object detection has become a key component, finding application in critical areas such as autonomous driving, security surveillance, industrial automation, and robotics [45,22]. State-of-the-art object detection models, including Faster-RCNN [28], DETR [6], and the YOLO series [37], have demonstrated impressive performance in identifying and localizing objects within images. Despite their success, the adoption of these models in highly sensitive environments remains limited, particularly in domains where errors could result in serious consequences such as injury, equipment damage, or operational failures. One of the primary reasons for this hesitancy is the black-box nature of object detectors implemented as Deep Learning models, which to date amount to the majority of proposals in the literature. The internal activations of these are not inherently interpretable, making it challenging for end-users to trust the predictions issued by object detectors, especially in high-risk environments operating in open-world environments such as autonomous vehicles and industrial robotics.

In this context, the field of Explainable AI (XAI) [5] aims to enhance the interpretability of AI systems by their *audience* and ultimately, to enhance the user's trust in the output of AI-based systems. Leaving aside the category of transparent AI models (which are inherently interpretable and do not require any explanations for a user to understand how they work), explainability methods in XAI can be broadly categorized into *white-box* and *black-box* approaches. *White-box XAI methods* require access to the internal workings of the model, such as weights,

* Corresponding author at: TECNALIA, Basque Research and Technology Alliance (BRTA), Mikeletegi Pasealekua 2, Donostia-San Sebastian, 20009, Spain.
E-mail address: alain.andres@tecnalia.com (Alain Andres).

https://doi.org/10.1016/j.rineng.2024.103498
Received 24 October 2024; Received in revised form 14 November 2024; Accepted 21 November 2024
Available online 26 November 2024

activations, or gradients (e.g., Grad-CAM [34]). While these methods can provide powerful insights, they are often limited by their dependence on specific model architectures, making them difficult to generalize across different models and less accessible to users unfamiliar with AI research/tools. In contrast, *black-box XAI methods* treat the model as an opaque entity, providing explanations based solely on the model's input-output behavior without requiring any access to its internal components. However, most black-box XAI methods are designed for classification tasks rather than for object detection [29,19,25,3].

While classification models produce a single label per image, object detection models must identify and localize multiple objects within an image. Therefore, they need to explain not only the class prediction for each detected object – *what* they detect– but also the spatial reasoning behind the bounding boxes that define the object's location – *where* the object is positioned within the image. Balancing these dual aspects complicates the explanation process and requires more sophisticated techniques than those used for classification tasks.

In this paper, we address the gap in XAI methods for object detection by focusing on *model-agnostic*, *black-box* XAI techniques. We propose and evaluate novel black-box XAI methods and XAI metrics that are specifically tailored for object detection models, without requiring access to internal model details. Our proposed methods are generalizable to object detection frameworks beyond those utilized in our experiments. Specifically, the contributions of this work can be summarized as follows:

- We formally define a quantitative evaluation metric, **D-Deletion**, which extends the existing Deletion metric [4,25] proposed for classification tasks. This metric is adapted to handle the unique challenges of object detection, including localization (as seen in Fig. 4), which is of utmost importance when multiple instances of the same object appear in the same scene.
- By using the similarity score of D-RISE [26], we analyze multiple mask generation methods' performance and introduce **D-MFPP**, an extension of MFPP [42] originally developed for classification tasks. D-MFPP utilizes segmentation-based mask generation to improve explanations for object detection models.
- We analyze the impact of key parameters, such as image dimensions and the model sizes within the YOLOv8 architecture utilized in our experiments, which can significantly influence the quality of the resulting explanations.
- Last but not least, we facilitate the broader adoption of the developed techniques for object detection in real-world use cases by releasing the code publicly in a repository: https://github.com/aklein1995/drise_dmfpp_ddeletion.

The remainder of this paper is structured as follows: in Section 2, we first review literature related to XAI for object detection. In Section 3, we provide the necessary background on object detection and XAI to familiarize the reader with the key concepts used in the definitions of D-RISE and Deletion. Next, Section 4 presents the experimental setup, including datasets, object detection training configuration, employed XAI methods, and evaluation metrics. In this section we also introduce our proposed D-MFPP method and D-Deletion metric. We discuss our results in Section 5. Finally, Section 6 concludes the paper with a summary of our key findings and directions for future research.

2. Related work

Before proceeding with the materials and novel methods introduced in this work, we first pause briefly at XAI methods, focusing on those used for object detection tasks and put to practice in industrial applications:

XAI methods. As stated in the introduction, XAI offers insights into the procedure followed by an AI-based system to elicit their outputs, enabling end-users to understand and eventually trust the decisions output by the AI-based system grounded on objective data [3]. To date, the majority of XAI methods are designed for models learned to address classification tasks. For instance, CAM-based methods like GradCAM [34], GradCAM++ [7] and Integrated Gradients [36] quantify and attribute the pixel-wise importance of a given input according to the gradients with respect a target class. Moreover, making use of backpropagation, LRP [20] calculates the contribution that a neuron has with neurons in consecutive layers to get relevance scores. In contrast, perturbation-based techniques work by occluding certain parts of the input and analyzing its impact in the predictions. Within this type of techniques, LIME [29], approximates a NN with an interpretable model; SHAP [19] assigns importance values to each input feature based on Shapley values; RISE [25] generates saliency maps by probing the model with randomly masked versions of the input image; and MFPP [42] generates masks by dividing the input image into multi-scale superpixels. Nonetheless, none of them have been explicitly extended for object detection tasks –with the exception of RISE, which has been adapted for this purpose– although techniques like SHAP can also be utilized for regression problems.

XAI methods for object detection models. In recent times, a scarcity of XAI approaches has been proposed to support the interpretability of complex object detection models. SODEx [33] is a method capable of explaining any object detection algorithm using classification explainers, demonstrating how LIME can be integrated within YOLOv4, a variant of the YOLO family of single-stage object detectors. Similarly, D-RISE [26] extends RISE's mask generation technique by introducing a new similarity score that assesses both the localization and classification aspects of object detection models. More recently, D-CLOSE [38] enhances D-RISE by producing less noisy explanations. Along with other methodological improvements, D-CLOSE uses multiple levels of segmentation in the mask generation phase. Other approaches focusing on hierarchical masking have been proposed. Concretely, GSM-NH [41] evaluates the saliency maps at multiple levels based on the information of previous less fine-grained saliency maps, whereas BODEM [21] further extends this idea but focuses on an extreme black-box scenario where only object coordinates are available.

XAI methods for industrial applications. Although XAI is increasingly important in industrial settings to ensure safety, reliability and compliance, the adoption of XAI for object detection methods in industrial use cases has been limited to date [17,15,9]. The vast majority of the works focus either on image classification, like [8] that utilizes Grad-CAM to interpret vibration signal images in the classification of bearing faults; time-series data, e.g. [35] that presents the implementation and explanations of a remaining life estimator model; or tabular data, as in [31] where SHAP is used to interpret and study the influence of soil and climate features on crop recommendations. Regarding XAI and object detection for industrial applications, we can find a few exemplary studies that expose the shortage of real-world use cases currently noted in this technological crossroads. In [23], various object detection models are evaluated for their effectiveness in detecting weld characteristics in radiography images, with an emphasis on explainability and deployment on edge devices to assist workers. In the same sense, [32] provides a comprehensive review and analysis of various XAI techniques applied to object detection tasks in computerized tomography imaging for medical purposes. Finally, [14] demonstrates how to integrate Grad-CAM into the YOLO architecture and performs experiments in both public and private datasets of vehicle front collision and rear-view cameras.

3. Background

We now proceed by elaborating on key concepts needed to properly understand the details of the proposed D-MFPP technique and the D-Deletion metric that lie at the core of this work. Concretely, we provide fundamentals for object detection models (Section 3.1) and XAI, with a focus on model-agnostic black-box methods to explain the predictions of object detection models (Section 3.2).

3.1. Object detectors

Object detectors are crucial components in computer vision tasks, capable of identifying and localizing objects within an image. They can be broadly categorized into single-stage and two-stage detectors.

Single-stage detectors. They directly predict bounding boxes and class probabilities from input images in a single pass. Popular single-stage detectors, such as YOLO [37], SSD [18] and RetinaNet [30], treat object detection as a simple regression problem, straight from image pixels to bounding box coordinates and class probabilities. To this end, they produce a dense grid of bounding box proposals and class probabilities in one step. Specifically, YOLO [37] divides the input image into a grid and predicts bounding boxes and class probabilities for each grid cell. Although this efficiency is beneficial for real-time applications, it often comes at the cost of accuracy when compared to two-stage detectors

Two-stage detectors. These models, among which Faster R-CNN [28] can be considered to be the most representative one, follow a more complex approach that divides the detection process into two stages. In the first stage, a Region Proposal Network (RPN) generates a set of candidate object proposals (bounding boxes) from the input image. In the second stage, these proposals are refined and classified into different object categories by a second network. This second stage typically involves a more complex network, such as a convolutional neural network (CNN), which performs classification and further refinement of the bounding box coordinates. This two-step process boosts accuracy by allowing for a more refined feature analysis, though it also slows down processing, making two-stage detectors less suited for applications that require high-speed performance.

Most detector networks, including Faster R-CNN and YOLO, produce a large number of bounding box proposals which are subsequently refined using confidence thresholding and Non-Maximum Suppression (NMS) to produce a set of finally detected objects in the image. Each bounding box proposal d_i can be defined as follows:

$$\mathbf{d}_i = \left[\mathbf{L}_i, O_i, \mathbf{P}_i\right] = \left[(x^i_1, y^i_1, x^i_2, y^i_2), O_i, (p^i_1, \dots, p^i_C)\right], \tag{1}$$

where $\mathbf{L}_i$ defines the bounding box corners (x^i_1, y^i_1) and (x^i_2, y^i_2); $O_i \in [0, 1]$ refers to the probability that bounding box L_i contains an object of any class; and $\mathbf{P}_i$ is a vector of probabilities $(p^i_1, \dots, p^i_C)$ representing the probability that region $\mathbf{L}_i$ belongs to each of C classes. Unlike traditional classifiers, which assign a single class label to an entire image, object detectors must handle both classification and localization simultaneously. This dual task, predicting the class and precise location of each object, increases the complexity of making these models interpretable.

3.2. Explainable Artificial Intelligence (XAI)

Despite the great performance exhibited by object detectors in manifold applications, their adoption in risk-sensitive scenarios is often hindered by a lack of trust and transparency by the user making decisions based on the detections issued by these models. As introduced previously, research on XAI produce techniques and methods that make the behavior and predictions of AI models understandable to humans without sacrificing performance [11]. To this end, multiple XAI techniques have been proposed, which can be classified into four broad categories [3]:

- *Scoop-based techniques* focus on the extent of the explanation, providing either local explanations for specific predictions or global explanations for the overall model behavior.
- *Complexity-based methods* consider the complexity of the model, with simpler, interpretable models offering intrinsic interpretability and more complex models requiring post-hoc explanations.
- *Model-based approaches* distinguish between XAI methods that are specific to particular types of models, and those that are model-agnostic, capable of being applied to any model disregarding the specifics of their internals.
- *Methodology-based techniques* are categorized by their methodological approach, such as backpropagation-based methods that trace input influences, or perturbation-based methods that alter inputs to observe changes in the output of the model.

Given that object detectors are typically complex neural networks, they fall under the *complexity-based* category, thereby requiring post-hoc explainability methods to explain their decisions. Among the various *methodology-based* techniques, *attribution methods* are commonly used to estimate the relevance of each pixel in an image for the detection task. Attribution methods are particularly important for object detection, where both localization and classification need to be explained.

Traditional attribution methods have been primarily developed for image classifiers [1], which produce a single categorical output, making them less suited for object detectors. Object detectors, unlike classifiers, generate multiple detection vectors that encode not only class probabilities, but also localization information and additional metrics, such as objectness scores (see Section 3.1). Furthermore, techniques like NMS and confidence threshold filtering, which are used to refine bounding box proposals, add complexities that require a deeper understanding of the model's internal workings, complicating the use of certain XAI methods, such as gradient-based approaches. Therefore, we focus on *model-agnostic black-box XAI* approaches, which are designed to be architecture-independent, and do not depend at all on the specifics of the model under target.

Among model-agnostic XAI methods, *perturbation-based* approaches are commonly used due to their simplicity and effectiveness in revealing which parts of the input are most influential for the model's predictions. Perturbation-based techniques offer a direct way to assess how changes to the input image affect the model's output. By systematically altering or masking parts of the input image (using masks to generate perturbed samples), these methods allow inferring the importance of different regions based on the model's input-output behavior.

The typical pipeline for perturbation-based XAI methods can be divided into three stages: *(1) Data Preparation, (2) Model Assessment, and (3) Importance Computation.* In the Data Preparation stage, masks are generated and applied to the image to create perturbed samples. The Model Assessment stage involves passing these perturbed images through the model to observe the changes in output. Finally, in the Importance Computation stage, the importance of each pixel is calculated by comparing the model's outputs for the original and perturbed images. While the Model Assessment stage remains consistent across methods, with each perturbed image passed through the model, the Importance Computation varies depending on the XAI approach used. This can range from simple techniques like retraining a model (e.g., LIME) to more complex approaches. Since the effectiveness of these methods largely depends on how the perturbed images are generated, three mask generation algorithms are next described (Fig. 1):

- *Sliding Window*: This method, which is similar to the Occlusion technique proposed in [43], systematically moves a window of fixed size across the image and sets the region within the window to a constant value (e.g., zero) to occlude that part of the image. By iteratively sliding the window across the entire image, we can assess the impact of each occluded region on the model's output. The method requires specifying the window size, which determines the area of the image being occluded at each step, and the stride, which sets how much the window moves between iterations.
- *RISE*: Randomized Input Sampling for Explanation (RISE) [25] involves sampling N binary masks of size $h \times w$, which are smaller than the original image size $H \times W$. Each element in the mask is independently set to 1 with probability p and to 0 with the remaining probability $1 - p$. These masks are then upsampled to size $(h+1) \cdot C_H \times (w+1) \cdot C_W$ using bilinear interpolation, where

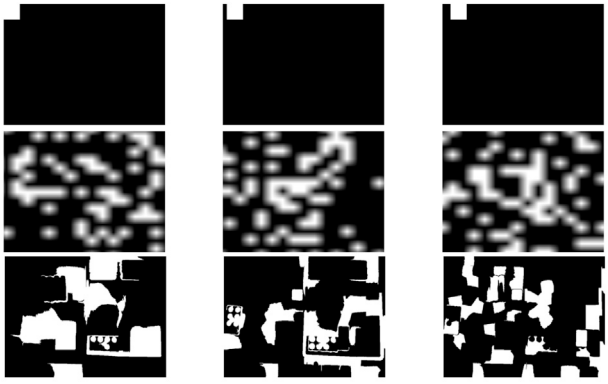


Fig. 1. Example of three masks generated using Sliding Window (top), RISE (middle), and MFPP (bottom). MFPP masks are dependent on the image at the input of the model. In this case, we consider a sample from the battery assembly dataset detailed in Section 4.

$C_H \times C_W = \lfloor H/h \rfloor \times \lfloor W/w \rfloor$. The upsampled masks are cropped to the original image size $H \times W$ with uniformly random offsets ranging from $(0,0)$ to (C_H, C_W). This method creates a diverse set of masks that cover different parts of the image, allowing for a comprehensive evaluation of the importance of various regions.

- *MFPP*: The so-called Morphological Fragmental Perturbation Pyramid (MFPP) [42] method divides the input image into multi-scale fragments and perturbs them randomly. In this sense, it is similar to RISE, but instead of perturbing elements of the generated masks with dimension $h \times w$, MFPP defines regions according to segmentations at different scales. Depending on the number of defined fragments, the regions would be more fine-grained yet more time-consuming. The segments are dependent on each image, requiring the creation of new masks for every image.

4. Materials and methods

This section describes the industrial robotics use cases in what refers to the datasets (Section 4.1), object detection model (Section 4.2), XAI methods (Section 4.3) and the explanation quality metrics (Section 4.4) considered in our work. The novel XAI technique and quality metrics proposed in this manuscript are also described in Section 4.3.

4.1. Industrial robotics datasets under consideration

The datasets used in this manuscript have been collected during the course of the ULTIMATE project, https://ultimate-project.eu/, which features two distinct real robotics use cases [16]. The first dataset, from PIAP https://piap.lukasiewicz.gov.pl/, involves a collaborative workspace where a human and a robotic arm work together. The second dataset, provided by Robotnik https://robotnik.eu/, focuses on a battery assembly area, where a robotic arm assembles components for a battery kit.[1]

Dataset 1: Human-Robot Dataset. This dataset consists of 96 images captured from three different cameras, as exemplified in Fig. 2, with 32 images taken from each camera. The dataset includes two object classes: `human` and `gripper`. Importantly, each image in this dataset contains only a single object of each class, meaning a maximum of one human and one `gripper` per image. To ensure a diverse and representative sample, we applied feature extraction using ResNet [12] to obtain embeddings for the entire dataset. The dimensionality of these embeddings was reduced using Principal Component Analysis (PCA), followed by K-means clustering (with $k = 8$ clusters). From each cluster, four images were randomly selected, resulting in a final subset. The data were split into three sets: 72 images for training (75%), 6 for validation (6.25%), and 18 for testing (18.75%). To maintain consistency, we applied the same partitioning to the data from each camera. This resulted in 24 images for training, 2 for validation, and 6 for testing from each camera.

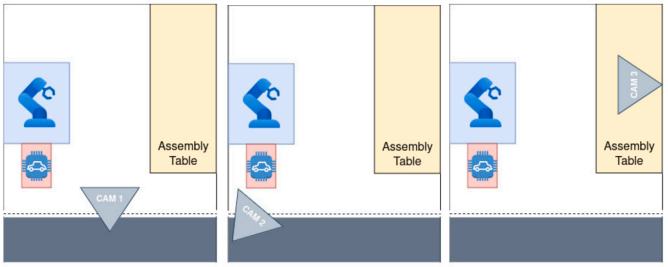


Fig. 2. Dataset 1 (Human-Robot collaboration): Data are captured from cameras located in 3 different positions. All the images belonging to this dataset contain the faces blur to preserver anonymity.

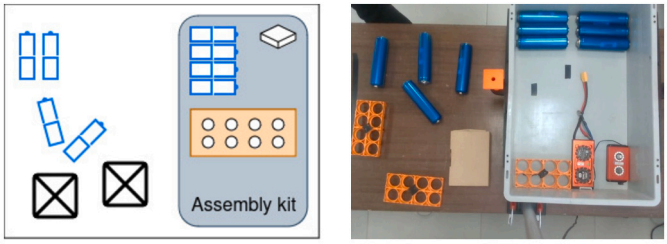


Fig. 3. Dataset 2 (Battery Assembly kit): The setup where a robotic arm would assemble the kit based a bird-eye view of the table where all component are expected to be; (left) a theoretical setup; (right) an actual sample.

Dataset 2: Battery Assembly Dataset. This dataset consists of 7 images, all captured from a bird's-eye (top-down) view, showing a robotic arm assembling a battery kit, as shown in Fig. 3. The dataset includes five distinct object types: `individual battery`, `bms_a`, `bms_b`, `battery holder`, and `unknown object`. In contrast to the Human-Robot Dataset, each image in the Battery Assembly Dataset may contain multiple objects of the same class, such as several individual batteries in a single scene.

It is worth noting that XAI techniques can be applied to any type of data. When applied to training data, they help reveal what the model has learned to focus on during training. When applied to test data, they provide insight into how well the model generalizes to new, unseen examples. For the Human-Robot Dataset, XAI explanations were applied exclusively to the test images, allowing us to assess the model's behavior on unseen data. However, for the Battery Assembly Dataset, given the limited number of images (only 7), XAI explanations were applied to the entire dataset.

4.2. Object detection model: YOLOv8

Among the possible object detector models, we selected one of the state-of-the-art options, YOLOv8, due to its numerous advancements over previous versions and its robust performance in object detection tasks [37]. YOLOv8 [27] integrates a novel combination of Feature Pyramid Network (FPN) and Path Aggregation Network (PAN) architectures, enhancing its ability to detect objects at various scales and resolutions.

[1] While the datasets contain a relatively small number of images, this data shortage is typically encountered in real-world industrial scenarios subject to data availability constraints. Nevertheless, in the use cases under considerations the contextual and scene variability is minimal, yielding short-tailed distributions of the objects to be detected. Therefore, the small datasets described in the paper sufficiently capture the relevant features for the specific object detection tasks addressed by the models.

The FPN gradually reduces the spatial resolution of the input image while increasing feature channels, facilitating multi-scale object detection. The PAN architecture further aggregates features from different levels through skip connections, improving the detection of objects with diverse sizes and shapes. Additionally, YOLOv8 introduces an anchor-free detection mechanism that directly predicts the center of an object (instead of the offset from a known anchor box), reducing the number of box proposals and speeding-up the post-processing. Furthermore, it was trained with larger and more diverse datasets including the popular COCO dataset, improving its performance across a wider range of images.

YOLOv8 was developed and released by Ultralytics, and although the model and its weights are open-source, most users are expected to utilize the Ultralytics framework for its enhanced usability. However, unlike previous YOLO releases where the probability for each class per predicted box was accessible, in YOLOv8, the Ultralytics API outputs only the probability for the class with the highest confidence in each box.[2] Consequently, by default, YOLOv8 outputs:

$$\mathbf{d}_i = \left[\mathbf{L}_i, O_i, C_i\right] = \left[(x_1^i, y_1^i, x_2^i, y_2^i), O_i, C_i\right], \tag{2}$$

where $\mathbf{L}_i = (x_1^i, y_1^i, x_2^i, y_2^i)$ represents the coordinates of the bounding box, O_i denotes the objectness score, and C_i corresponds to the predicted class label for the object within the bounding box, which differs with respect to the outputs shown in Expression (1).

4.3. Explainability methods

We evaluate four popular methods for generating visual explanations of black-box models: LIME, RISE, D-RISE, and D-MFPP. The first two methods, LIME and RISE,[3] were originally developed for image classifiers but can be adapted to object detectors. However, they primarily focus on explaining classification aspects and are not capable of addressing localization characteristics. In contrast, D-RISE is one of the first XAI methods specifically designed for object detectors, providing explanations that encompass both classification and localization. Additionally, we extend the existing MFPP method (originally tailored for classifiers) into a version suitable for object detection, which we refer to as D-MFPP. In what follows we briefly describe them, flowing into a description of the proposed D-MFPP approach:

- **LIME** was originally designed to explain the predictions of any classifier by approximating it locally with an interpretable model. To explain the prediction for an input image I, LIME fits an interpretable model g (e.g., a linear model) to approximate the behavior of the black-box model f locally around I. The similarity between the original image and the perturbed samples is measured using a kernel function $\pi_I(z)$. When image explanations are targeted, LIME groups contiguous pixels into superpixels based on similar features they represent. This approach allows LIME to measure the importance of regions in the image rather than individual pixels, making the explanations more interpretable.
- As introduced in the previous section, **RISE** [25] was originally designed for deep neural networks that take images as input and output a class probability (e.g., a classifier like ResNet-50). It generates saliency maps that indicate the importance of each pixel by applying randomly generated binary masks M_i to the input image I and observing the changes in the model's output $f(I \odot M_i)$. In RISE, N binary masks $M_i \in \{0,1\}^{h \times w}$ are generated (as explained in Section 3.2). These masks are then applied to the input image I to generate masked images $I_i' = I \odot M_i$, where $\odot$ denotes element-wise multiplication. The model is evaluated on each masked image I_i' to obtain the outputs $f(I \odot M_i)$. The importance score for each pixel (x, y) is then calculated as the weighted sum of the outputs:

$$S_{I,f}(x,y) = \frac{1}{N} \sum_{i=1}^{N} f(I \odot M_i) \cdot M_i(x,y) \tag{3}$$

where the weights $M_i(x, y)$ represent the value of mask i at pixel (x, y). The intuition behind RISE is that $f(I \odot M_i)$ would be high when pixels preserved by mask M_i are important. Although this is true when having infinite diverse masks, in practice RISE calculates each pixel's importance empirically by Monte Carlo sampling. Therefore, RISE largely depends on the number of masks (N) and how they are generated (i.e., is sensitive to the selected probability p and resolution s).

4.3.1. D-RISE and proposed D-MFPP approach

Unlike the other two approaches originally designed for classifiers that measure solely classification aspects, **D-RISE** (Detector Randomized Input Sampling for Explanation) [26] was designed to explain both the classification and localization of a detection. In this sense, D-RISE extends RISE by producing saliency maps specifically for object detectors. As previously seen in Section 3.1, the output given by an object detector differs from the probability vector given by a classifier, obtaining localization information L_i, an objectness score O_i and the probability of classifying each bounding box to any of the considered classes P_i. As a consequence, Expression (3) used by RISE is replaced in D-RISE with a new similarity score, given by:

$$S_{I,f}(\mathbf{d}_t, \mathbf{d}_j) = s_L(\mathbf{d}_t, \mathbf{d}_j) \cdot s_P(\mathbf{d}_t, \mathbf{d}_j) \cdot s_O(\mathbf{d}_t, \mathbf{d}_j), \tag{4}$$

where $s_L = IoU(\mathbf{L}_t, \mathbf{L}_j)$, $s_P = \mathbf{P}_t \cdot \mathbf{P}_j / (||\mathbf{P}_t|| \cdot ||\mathbf{P}_j||)$, and $s_O = O_j$. In this formulation, s_L represents the spatial proximity of the bounding boxes encoded by the target detection $\mathbf{d}_t$ and the proposal $\mathbf{d}_j$, measured using the Intersection over Union (IoU); the term s_P evaluates the similarity between the class probabilities of the target detection and the proposal using cosine similarity; and s_O incorporates the objectness score of the proposal O_j. It is important to note that for a detection target $\mathbf{d}_t$ there would potentially be more than one detection proposals $\mathbf{d}_j$. Therefore, we would have multiple $S_{I,f}(\mathbf{d}_t, \mathbf{d}_j)$. As explained in D-RISE, the explanations consider only the detection with maximal score for each mask:

$$S_{I,f}\left(\mathbf{d}_t, f(M_i \odot I)\right) = \max_{\mathbf{d}_j \in f(M_i \odot I)} S_{I,f}(\mathbf{d}_t, \mathbf{d}_j). \tag{5}$$

Given the YOLOv8 outputs explained in Section 4.2, which do not provide the class probability vector P_i without modifying its architecture (an approach we want to avoid within the scope of this paper), we must adapt the similarity score to only consider s_L and s_O. Consequently, the modified similarity score can be expressed as:

$$S_{I,f}(\mathbf{d}_t, \mathbf{d}_j) = s_L(\mathbf{d}_t, \mathbf{d}_j) \cdot s_O(\mathbf{d}_t, \mathbf{d}_j) = IoU(\mathbf{L}_t, \mathbf{L}_j) \cdot O_j. \tag{6}$$

This adjustment allows still utilizing D-RISE effectively for generating saliency maps with the default YOLOv8 model, focusing on the spatial and objectness aspects of detections, while maintaining the integrity of the model's original architecture.

Similarly, we can adopt this similarity score but apply it with a different mask generation process. The MFPP method introduced in Section 3.2, originally designed for classification tasks, can be extended by applying Equation (6), resulting in *D-MFPP*. To the best of our knowledge, no previous work has proposed this variant of MFPP for object detection tasks.

[2] https://github.com/ultralytics/ultralytics/issues/2863% https://github.com/ultralytics/ultralytics/issues/4908.

[3] These XAI methods have been chosen due to their perturbation-based nature, which aligns closely with the methodology followed by the XAI methods D-RISE and D-MFPP proposed in this work. Both D-RISE and D-MFPP generate explanations through perturbations.

4.4. Metrics

Evaluating the performance of attribution-based explainability methods for image data involves assessing how well the generated relevance heatmaps highlight important regions of the input image that contribute to the model's decision. Generally, according to [13], explanation quality metrics can be grouped into six categories based on their logical similarity: faithfulness, robustness, localization, complexity, randomization, and axiomatic metrics. In this study, we focus on two of these categories that are particularly relevant to object detection: localization (Section 4.4.1) and faithfulness (Section 4.4.2).

4.4.1. Localization

Localization metrics evaluate whether the explainable evidence is centered around a region of interest (RoI) defined by a bounding box, segmentation mask, or a cell within a grid. These metrics aim to verify if the saliency maps correctly highlight the areas in the image that contain the object of interest. Among them, our experiments will consider:

- *Pointing Game* (PG), which is a human evaluation metric introduced in [44]. If the highest saliency point lies inside the human-annotated bounding box of an object, it is counted as a hit. The PG accuracy is given by:

$$\text{PG} = \frac{\#Hits}{\#Hits + \#Misses}, \tag{7}$$

  which is averaged over all categories in the dataset.
- *Energy-based Pointing Game* (EBPG) [39], which measures the proportion of activations within the given bounding box relative to the whole activation in the image. It assesses how much of the model's activation energy is concentrated within the predefined region of interest. Formally:

$$\text{EBPG} = \frac{\sum_{(x,y)\in \text{bbox}} S_{I,f}(x,y)}{\sum_{(x,y)\in \text{bbox}} S_{I,f}(x,y) + \sum_{(x,y)\notin \text{bbox}} S_{I,f}(x,y)}, \tag{8}$$

  where $S_{I,f}(x,y)$ represent the saliency score at pixel (x,y), $\sum_{(x,y)\in \text{bbox}} S_{I,f}(x,y)$ represents the sum of activation values within the bounding box, and $\sum_{(x,y)\notin \text{bbox}} S_{I,f}(x,y)$ represents the sum of activation values outside the bounding box.

4.4.2. Faithfulness

Metrics accounting for faithfulness quantify to what extent explanations follow the predictive behavior of the model, asserting that more important features play a larger role in model outcomes. These metrics focus on understanding the causal relationship between input features and the model's output by systematically altering the features and observing the changes in predictions. Among them:

- *Deletion*: Inspired by the work by [4], the Deletion metric was proposed in RISE [25]. This metric measures a decrease in the probability of the predicted class as more and more important pixels are removed, where the importance is obtained from the saliency map. A sharp drop, and thus a low Area Under the probability Curve (AUC, as a function of the fraction of removed pixels), indicates a good explanation. Given the importance score for each pixel calculated by any XAI method, $S_{I,f}$, we can formulate the Deletion metric as:

$$\text{Deletion(I,S,c)} = \text{AUC}\left(\left\{Pr(f\left(I \odot M_k\right) = c)\right\}_{k=1}^{K}\right), \tag{9}$$

  where I is the original image, M_k represent a mask with the k-th most important pixels removed sorted by $S_{I,f}$, $Pr(f\left(I \odot M_k\right) = c)$ represents the probability of model f predicting that the bounding box belongs to class c, and AUC$(\cdot)$ computes the area under the curve for the K predictions.
- *Minimum Subset*: It follows the same logic as *Deletion*, but instead of determining the AUC, it considers the required number of pixels that make the prediction to change [10]. Given the importance score for each pixel ($S_{I,f}$), *Min-Subset* is defined as the smallest subset of pixels that needs to be removed to change the model's prediction. Mathematically:

$$\text{Min-Subset}(I,S,c) = \min\left\{k \in \{1,2\ldots,K\} : f\left(I \odot M_k\right) \neq f(I)\right\}, \tag{10}$$

  where $f(I \odot M_k)$ represents the class label assigned by the model f after passing the image I with the top k most important pixels removed, and $f(I)$ is the class label predicted for the original image.

4.4.3. Proposed D-deletion and D-minimal subset metrics

Originally, Deletion was designed for classifiers. However, with object detectors, multiple detections in a single image can occur. Although D-RISE stated the necessity to adapt this metric for object detectors [26], no formal definition can be found in the literature. Therefore, considering the importance of this issue in real use cases, we formally re-define Equation (9) in two manners:

1. *Deletion.* Measures the explanation given the target class label C_t (regardless if there is more than one element for a class) and iteratively removes the top k most important pixels:

$$\text{Deletion}(I,S,C_t) = AUC\left(\left\{\max_{\mathbf{d}_j^k}\left[O_j^k \cdot \mathbb{I}\{C_j^k = C_t\}\right]\right\}_{k=1}^{K}\right). \tag{11}$$

   The model $f(\cdot)$ takes as input the masked image $I \odot M_k$ and outputs a set of bounding box proposals $\mathbf{d}_j^k = [\mathbf{L}_j^k, O_j^k, C_j^k]$. The indicator function $\mathbb{I}\{C_j^k = C_t\}$ equals 1 if the predicted class C_j^k matches the target class C_t, and 0 otherwise. The term $\max_{\mathbf{d}_j^k}\left[O_j^k \cdot \mathbb{I}\{C_j^k = C_t\}\right]$ selects the maximum objectness score O_j^k for the bounding boxes where the predicted class matches the target class. The AUC is then computed over the set of prediction scores for the K steps, where at each step the most important pixels are progressively removed.
2. *D-Deletion.* While the standard Deletion metric evaluates the impact of pixel removal on a class prediction, it lacks the ability to account for spatial localization, which is essential in object detection tasks where multiple instances of the same class can appear. D-Deletion addresses this limitation by focusing on a specific target bounding box $\mathbf{d}_t$, considering both the class information, C_t, and IoU between the target and other detected proposals, $\mathbf{d}_j^k$. This ensures that the metric not only measures faithfulness but also takes localization into account, providing more precise explanations in situations where different objects of the same class coexist. Mathematically is expressed as:

$$\text{D-Deletion}(I,S,C_t) = AUC\left(\left\{\max_{\mathbf{d}_j^k}\left[O_j^k \cdot \mathbb{I}\{C_j^k = C_t\} \cdot \mathbb{I}\{IoU(\mathbf{d}_t,\mathbf{d}_j^k) > \gamma\}\right]\right\}_{k=1}^{K}\right) \tag{12}$$

   where γ is a threshold. As a consequence, when multiple elements of the same class are in an image, *D-Deletion* will only consider those proposals $\mathbf{d}_j^k$ predicted by the model that have a predefined IoU with the target bounding box $\mathbf{d}_t$.

The difference between Deletion and D-Deletion is illustrated in Fig. 4. This figure highlights how D-Deletion distinguishes between different objects of the same class by incorporating localization information, leading to more refined and accurate explanations (↓ AUC in the Figure's last row) when multiple objects of the same class are detected in an image. For the sake of clarity, we provide the pseudocode of Deletion in Algorithm 1, where the main difference with respect to D-Deletion are lines **10** to **12**.

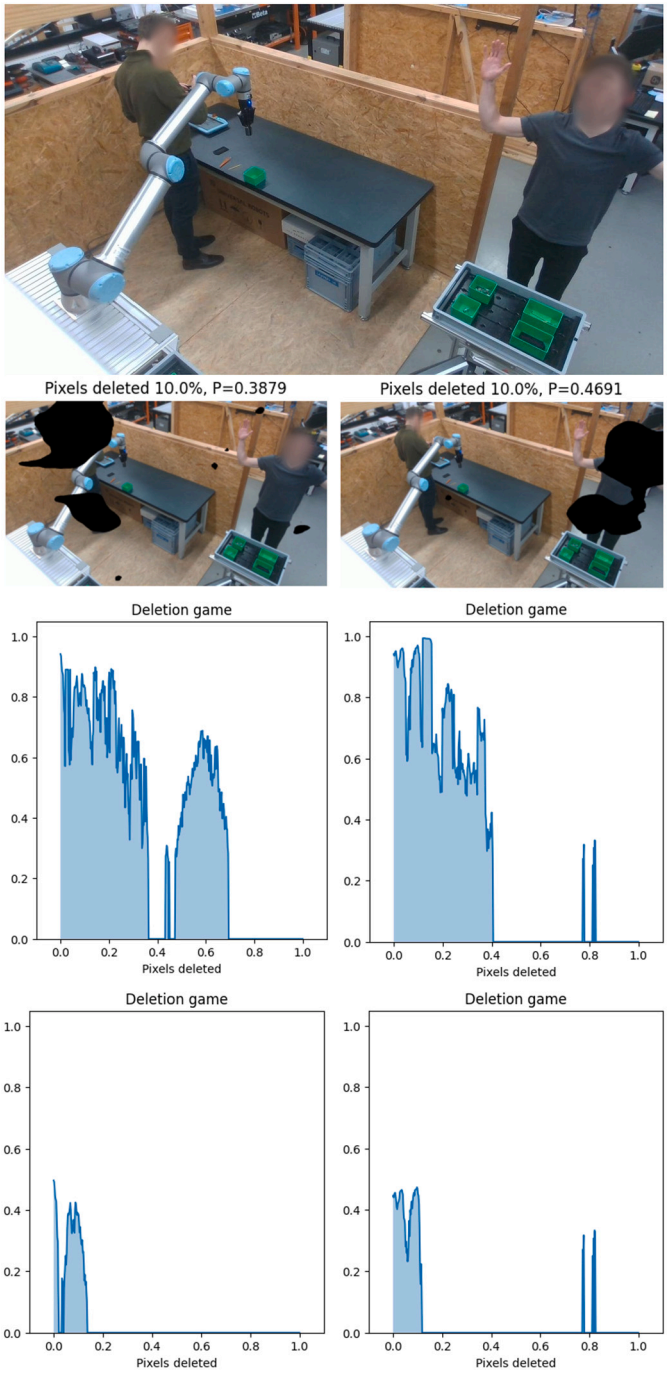


Fig. 4. Illustration of a collaborative workspace featuring two humans and a robotic arm. The first row shows the original image. The second row displays the image with the 10% most important pixels removed for each human, as identified by an XAI method. In the third row, the Deletion metric curve, which only considers class type, shows a high probability score even when the primary human is largely occluded by the other person. The fourth row presents the D-Deletion metric curve, which incorporates a localization component, providing a more accurate measure of explanation importance by considering the positions of entities within the image. A lower area under the curve indicates a better explanation.

Lastly, akin to how *Min-Subset* is related to *Deletion*, *D-Min-Subset* is associated with *D-Deletion*. Consequently, *D-Min-Subset* considers both the class type and the IoU to determine the number of pixels required to make the prediction to change:

$$\text{D-Min-Subset}(I, S, C_t) =$$

Algorithm 1 Deletion Metric's Pseudocode for Object Detector.

Require: Image I, saliency map $S_{I,f}$, number of steps K, target detection $\mathbf{d}_t$, target class C_t

Initialize $S \leftarrow []$
for $k = 1$ to K **do**
  $M_k \leftarrow S_{I,f}$ removing the top k most important pixels
  Apply mask M_k to image I
  Forward pass through the model f and obtain the bounding box proposals $\mathbf{d}_j = [\mathbf{L}_j, O_j, C_j] = f(I \odot M_k)$
  Initialize list of proposals: `proposals` $\leftarrow []$
  for each bounding box $\mathbf{d}_j$ predicted by the model f **do**
    if $C_j = C_t$ **then**
      `proposals.append`(O_j)
      % *For D-Deletion*
      if $IoU(\mathbf{d}_t, \mathbf{d}_j) > \gamma$ **then**
        `proposals.append`(O_j)
      end if
    else
      `proposals.append`(0)
    end if
  end for
  Insert the maximum score within the deletion buffer: $S \leftarrow S \cup \max(\texttt{proposals})$
end for
Calculate the Deletion metric as the area under the curve: $D = \text{AUC}(S)$
return Deletion score D

$$\min \left\{ k \in \{1, 2 \ldots, K\} : C_j^k \neq C_t \text{ or } IoU(\mathbf{d}_t, \mathbf{d}_j^k) < \gamma \right\}, \tag{13}$$

where C_j^k represents the predicted class label for detection j when passing the masked image $I \odot M_k$ through the model f, with $\mathbf{d}_j^k = f(I \odot M_k)$ being the set of detections after removing the top k most important pixels. In this context, *D-Min-Subset* depends on two conditions: (1) the class probability labels C_j^k for the predicted bounding box $\mathbf{d}_j^k$ must no longer match the target class C_t, or (2) the IoU between the target bounding box $\mathbf{d}_t$ and the predicted bounding box $\mathbf{d}_j^k$ falls below the threshold γ. The minimum k is identified as the step where either of these conditions is first met.

5. Experiments and results

Contrarily to most studies in the XAI literature that primarily focus on benchmark datasets, our research work focuses on assessing the explainability of object detectors in real-world industrial data. In this context, to evaluate the effectiveness of explanations, we formulate four key research questions to answer them with empirical evidence:

- **RQ1:** Which XAI method provides the most reliable and insightful explanations for object detection models?
- **RQ2:** Does the D-Deletion metric enhance the trustworthiness of XAI outputs when multiple objects of the same class are present in the image?
- **RQ3:** How does the mask generation process influence the quality of explanations, particularly when using similarity scores for object detection? How does D-MFPP behave?
- **RQ4:** Do different image dimensions impact the explanations generated by XAI methods? Do models of varying sizes (large, medium, small, nano) focus on different regions of the image in their explanations?

Next, we outline the hyperparameters used across our experiments to ensure consistency in training and evaluation. For both datasets, models were trained using the YOLOv8 architecture for a total of 100 epochs. The image size (imgsize) was set to the largest dimension of the input image (e.g., $720 \times 1280 \rightarrow 1280$), and data augmentation techniques such as random horizontal flipping and color jitter were applied. For consistency, the default Ultralytics settings were used wherever applicable. In

Table 1
Quantitative metrics of LIME, RISE and D-RISE for the Human-Robot dataset. The table presents the performance of each XAI technique in terms of classification (Deletion, D-Deletion, Min-Subset, D-Min-Subset) and localization metrics (PG and EBPG), with scores reported for each class (`Human`, `Gripper`) and the overall average. Lower values are better for metrics marked with ↓, while higher values are better for those marked with ↑. **Bold values** indicate the best average scores across all objects, highlighting the best-performing XAI method for each metric. Values highlighted in gray represent the best scores for each object category (`Human` or `Gripper`) and should be interpreted vertically.

XAI Method	`Object`	Deletion (↓)	D-Deletion (↓)	Min-Subset (↓)	D-Min-Subset (↓)	PG (↑)	EBPG (↑)
LIME	`Human`	0.0759	0.0632	4.2703	4.2703	1.0000	34.984
	`Gripper`	0.4688	0.0324	1.3859	1.3859	1.0000	2.2270
	Average	0.2723	0.0478	**2.8281**	**2.8281**	**1.0000**	**18.6060**
RISE	`Human`	0.1827	0.1241	9.0108	9.0108	0.7500	19.0824
	`Gripper`	0.2637	0.0060	0.6355	0.63	1.0000	1.0542
	Average	0.2232	0.0651	4.8232	4.8232	0.875	10.0683
D-RISE	`Human`	0.1255	0.0818	5.7335	5.7335	0.8750	20.4061
	`Gripper`	0.2777	0.0059	0.6091	0.6091	1.0000	1.0815
	Average	**0.2016**	**0.0438**	3.1713	3.1713	0.9375	10.7438

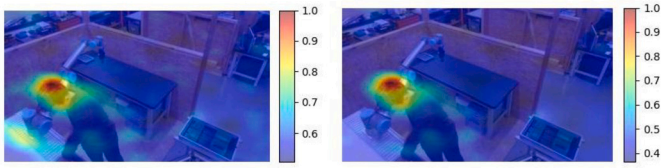


Fig. 5. Heatmaps obtained by applying RISE (left) and D-RISE (right) for the detection of a human in the Human-Robot Dataset.

the case of LIME, we use the baseline implementation of [29], where we adopt the SLIC segmentation algorithm [2] (with 100 segments) and generated 1000 samples to assess the quality of the produced explanations. For RISE and D-RISE, we employed 5000 masks with a probability of 0.25 and a resolution of 16×16 to produce the saliency maps. Lastly, for all object detection predictions, a confidence threshold of 0.7 was set to determine the validity of each detection.

In what follows we present and discuss on the results obtained to answer each of the RQ formulated above:

RQ1: Comparison between XAI methods

In the Human-Robot dataset, the comparison between LIME, RISE, and D-RISE, as shown in Table 1, reveals distinct strengths across different metrics (Section 4.4). LIME performs good in terms of localization, with higher PG and EBPG scores (100% and 18.60%, respectively) compared to D-RISE (93.75% and 10.74%). This indicates that LIME generates more localized saliency maps, focusing closely on the bounding boxes of detected objects. However, this superior performance is partly due to the size of the object being analyzed. LIME's superpixel generation is better suited for larger objects (e.g., `human`), as larger regions of the image can be grouped effectively into meaningful segments, leading to higher localization scores. This advantage also applies to classification, where larger objects allow LIME to better preserve relevant features for detection. Conversely, for smaller objects (e.g., `gripper`), LIME struggles when compared to the other methods, as reflected by its worse performance metrics in those cases.

In contrast, RISE and D-RISE are less sensitive to object size, making them more robust across different object scales, which is evident in their better performance on smaller objects like the `gripper`. They achieve Deletion scores of 0.2636 and 0.2777, respectively, compared to LIME's 0.4688. When considering the overall performance across classes, RISE, with a Deletion score of 0.2232 and D-Deletion of 0.0651, shows improvement over LIME in classification-related tasks but still lags behind D-RISE, which achieves the lowest Deletion (0.2016) and D-Deletion (0.0438) scores. Although D-RISE offers the best balance between classification and localization, the difference between RISE and D-RISE is

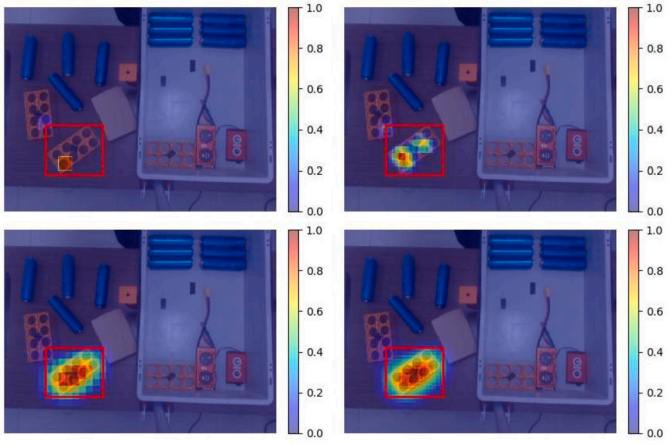


Fig. 6. Explanations of a scene using different stride and window size configurations when using D-Sliding Window (combination of mask generation explained in Section 3.2 and Equation (6)). (Left) Stride of 16; (Right) Stride of 8; (Top) Window size of 32; (Bottom) Window size of 64.

minimal in this dataset, where each image contains only a single object per class. As a result, as shown in Fig. 5, their heatmaps are very similar to each other, both highlighting the human head. However, D-RISE eliminates less relevant areas more effectively.

In the results obtained over the Battery Assembly dataset (Table 2), a similar pattern can be noticed. LIME excels at localization with an average EBPG of 16.03%, while RISE and D-RISE perform better in retaining key classification features. Since this dataset includes multiple objects of the same class (e.g., multiple batteries), both LIME and RISE, which are not designed to handle multiple detections of the same class, expose severe limitations. RISE, with a D-Deletion score of 0.1474, preserves key features better than LIME, but is outperformed by D-RISE, which achieves a score of 0.0344. D-RISE also shows the highest PG score (97%), performing significantly better than LIME (76.85%) and RISE (66.95%).

Overall, when dealing with datasets containing only one object per class, the differences between LIME, RISE, and D-RISE are relatively small in quantitative terms. However, when multiple objects of the same class appear in a given input image, D-RISE clearly dominates over the rest of techniques. As illustrated in Fig. 7, D-RISE generates coherent heatmaps for each detected object in the Battery Assembly dataset, whereas LIME and RISE provide a global saliency map for the entire class. By combining the individual saliency maps from D-RISE, a more accurate and object-specific explanation can be produced. This also highlights the limitations of LIME and RISE when applied to mul-

Table 2
Quantitative metrics of LIME, RISE and D-RISE for the Battery Assembly dataset. The table presents the performance of each XAI technique in terms of classification (Deletion, D-Deletion, Min-Subset, D-Min-Subset) and localization metrics (PG and EBPG), with scores reported for each object (`indiv batt`, `bms a`, `bms b`, `unknown object`, `batt holder`) and the overall average. Lower values are better for metrics marked with ↓, while higher values are better for those marked with ↑. **Bold values** indicate the best average scores across all objects, highlighting the best-performing XAI method for each metric. Gray-highlighted values represent the best scores for each object category (`indiv batt`, `bms a`, `bms b`, `unknown object` or `batt holder`) and should be interpreted vertically.

XAI Method	Object	Deletion (↓)	D-Deletion (↓)	Min-Subset (↓)	D-Min-Subset (↓)	PG (↑)	EBPG (↑)
LIME	`indiv batt`	0.7549	0.2806	44.3412	14.6756	0.3188	2.5347
	`bms a`	0.0184	0.0181	0.8784	0.8784	1.0000	30.7573
	`bms b`	0.0125	0.0125	0.8321	0.8321	1.0000	16.5440
	`unknown object`	0.0624	0.0245	2.0342	2.0342	1.0000	20.1071
	`batt holder`	0.1498	0.0849	6.8115	4.3766	0.5238	10.2087
	Average	**0.1996**	0.0841	**10.9795**	4.5594	0.7685	**16.0304**
RISE	`indiv batt`	0.7659	0.4359	81.2344	32.0482	0.0144	1.2558
	`bms a`	0.0190	0.0190	1.9417	1.9417	1.0000	3.0409
	`bms b`	0.0217	0.0217	2.1266	2.1266	1.0000	2.4561
	`unknown object`	0.5333	0.2008	3.8372	3.8372	1.0000	5.5780
	`batt holder`	0.0902	0.0595	7.9519	5.5632	0.3333	3.3681
	Average	0.2860	0.1474	19.4184	9.1034	0.6695	3.1398
D-RISE	`indiv batt`	0.6214	0.0311	35.7678	2.7725	1.0000	2.0546
	`bms a`	0.0181	0.0181	1.9880	1.9880	1.0000	3.2955
	`bms b`	0.0128	0.0116	1.3407	1.3407	1.0000	2.8876
	`unknown object`	0.0879	0.0485	3.7448	4.2071	0.8571	7.6831
	`batt holder`	0.4839	0.0626	21.7291	5.1009	1.0000	4.9635
	Average	0.2448	**0.0344**	12.9141	**3.0819**	**0.9714**	4.1768

tiple objects, as their global saliency maps do not differentiate between individual instances.

RQ2: D-deletion metric for scenes with multiple objects of the same class

As a secondary observation in the experiments of RQ1, the D-Deletion metric is specifically designed to overcome the limitations of traditional deletion metrics, particularly when multiple objects of the same class are present in an image.

In the Battery Assembly dataset, where several instances of the same class (e.g., `indiv batt`) appear, D-Deletion demonstrates clear advantages. By inspecting Table 2, RISE, while performing reasonably well with an average Deletion score of 0.2860, it still obtains a relatively high D-Deletion score of 0.1474, suggesting that it struggles to differentiate between the contributions of individual objects. In contrast, D-RISE, which obtains an average Deletion score of 0.2448, outperforms RISE with a D-Deletion score of 0.03444. This highlights D-RISE's ability to isolate and preserve key features for each object, providing more trustworthy, object-specific explanations rather than broad, class-level insights.

The Min-Subset and D-Min-Subset metrics, which measure the minimal proportion of pixels needed to disrupt a detection, reinforce these findings. In the Human-Robot dataset, Table 1, where only one object per class appears, the differences between Deletion and D-Deletion scores are minor, and the Min- Subset and D-Min-Subset values are close to each other. However, in the Battery Assembly dataset, where the differences between Deletion and D-Deletion are more substantial and multiple objects of the same class co-occur in the same image, the Min-Subset (12.9141) and D-Min-Subset (3.0819) values also diverge significantly.

RQ3: Influence of the mask generation strategy

When comparing XAI approaches for object detection tasks configured with different mask generation techniques, the results in Tables 3 and 4 initially suggest that D-Sliding Window performs the best in almost all metrics. However, as noted in the captions, this is only in cases where explanations were provided. For the Human-Robot dataset (Table 3), regardless of the window size and stride, D-Sliding Window failed to provide explanations for larger objects, such as `humans`, and only provided meaningful explanations for smaller objects like `gripper` instances. Similarly, in the Battery Assembly dataset (Table 4), D-Sliding Window struggled with large objects when using a smaller window size (w = 32), which led to higher scores in classification metrics. Even with

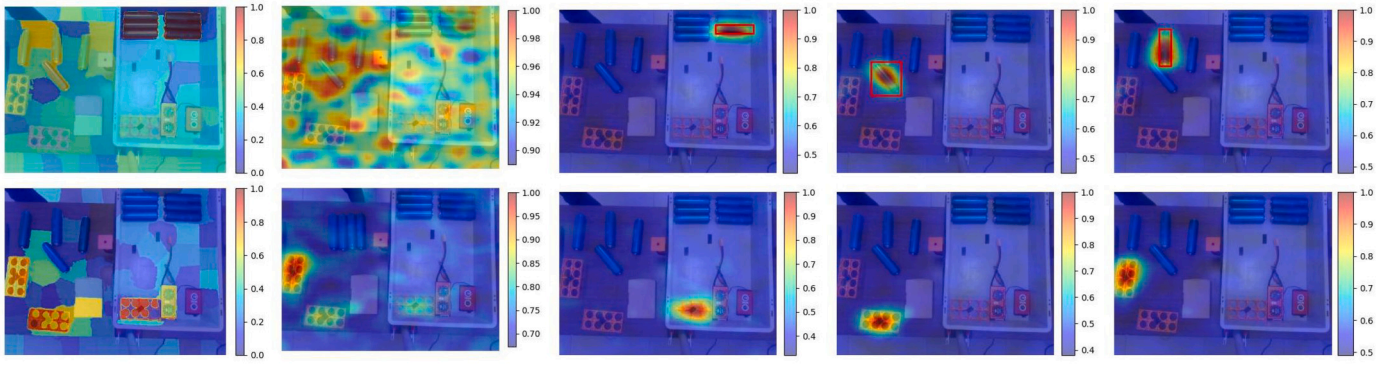


Fig. 7. Heatmaps generated in a scene of the Battery Assembly dataset for two target classes: `individual battery` (top row) and `battery holder` (bottom row). The first and second columns shows the saliency maps obtained using LIME and RISE, independent of the number of objects of the same class in the image. Columns 3,4 and 5 display heatmaps generated using D-RISE for three different individual elements of the same class.

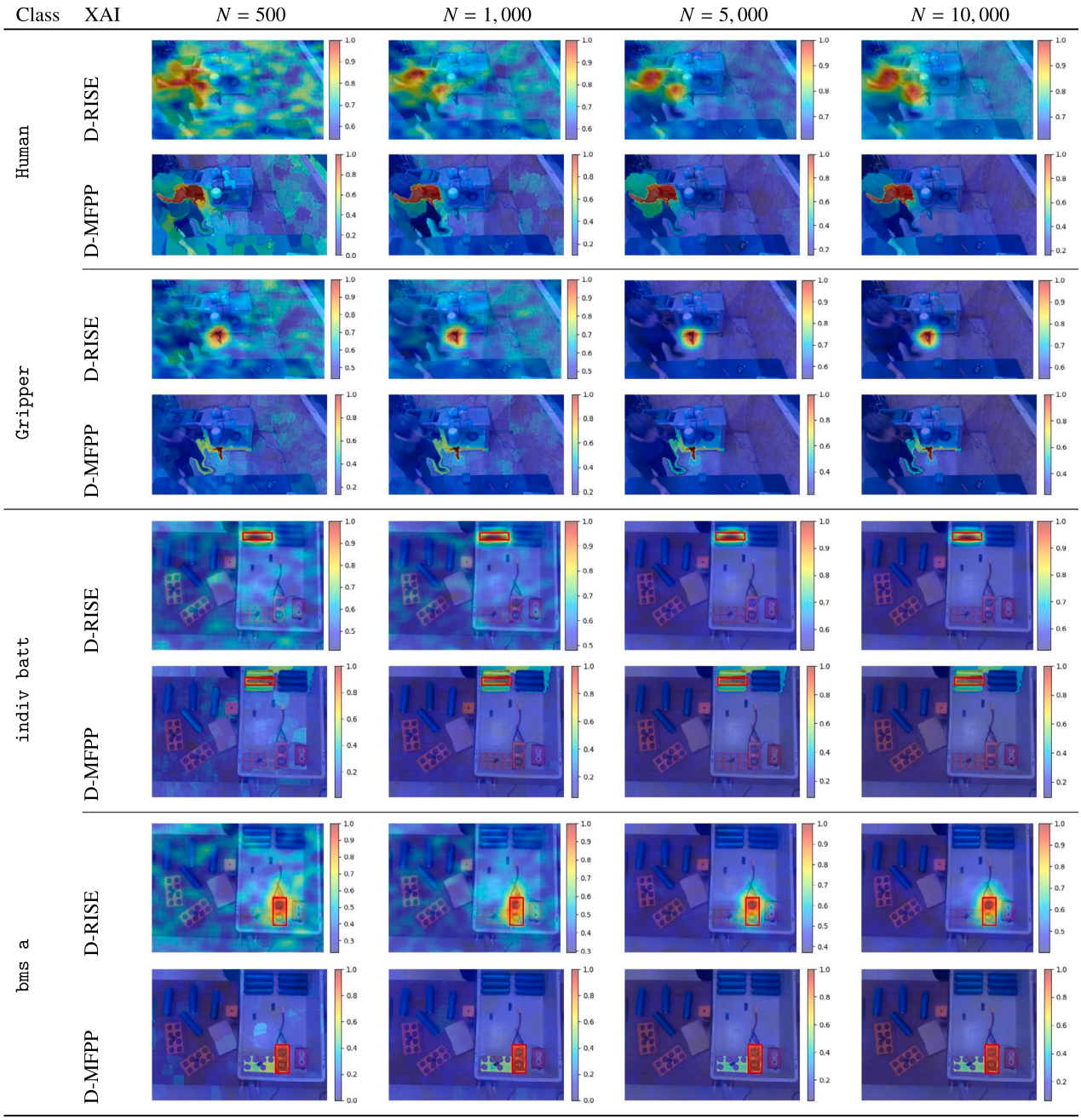


Fig. 8. Heatmaps obtained with D-RISE and D-MFPP using varying numbers of masks (500, 1000, 5000, 10000). Saliency maps are presented for two scenes, one from each dataset, demonstrating the evolution of heatmaps for a single element as the number of masks increases, highlighting the effect on explanation quality and focus.

an increased window size (w = 64), some objects were still not detected, making it difficult to fairly compare with D-RISE and D-MFPP, both of which provided explanations for all detection proposals.

In contrast, D-RISE consistently performs best across all detection proposals when explanations are provided, particularly in terms of the D-Deletion metric, with D-MFPP as a close second. D-MFPP manages to better distinguish important regions by de-emphasizing less relevant areas, as reflected in the PG and EBPG localization scores. Yet, due to the superpixel segmentation approach used by D-MFPP, some relevant features are grouped with less important ones (see Fig. 8), causing a slight drop in classification metrics. As a result, D-MFPP trades off classification precision for improved localization and focus. D-Sliding Window, as previously noted, performs well on smaller objects but struggles to generate reliable explanations for larger objects. This limitation can be mitigated by using larger window sizes, which occlude larger portions of the image, thereby increasing the likelihood of capturing explanations for bigger objects, while smaller strides help produce more finely detailed explanations, as seen in Fig. 6.

Influence of the number of masks. This secondary analysis within RQ3 confirms that the number of masks in use has a clear impact on the performance of the XAI method. Generally, more masks result in better outcomes. Interestingly, D-MFPP initially outperforms D-RISE in some metrics when using fewer masks (500 or 1,000), achieving a perfect PG score of 100% in the Human-Robot dataset and a D-Deletion score of 0.0505 with just 500 masks. However, as the number of masks increases, D-RISE slightly surpasses D-MFPP in classification-related metrics. This trend can be also noted in Fig. 8, where (1) increasing the number of

Table 3
Quantitative results for saliency maps generated with various masks techniques (D-Sliding Window, D-RISE, and D-MFPP) in the Human-Robot dataset. Only the average score across all classes is presented. Results are interpreted vertically within each column (each metric represents a separate column). The best mask configuration for each XAI method and for each metric is highlighted in gray, while the overall best-performing XAI method with its optimal mask configuration is shown in bold.

XAI Method	Num Masks	Deletion (↓)	D-Deletion (↓)	Min-Subset (↓)	D-Min-Subset (↓)	PG (↑)	EBPG (↑)
D-Sliding Window (*)	$w = 32,\ s = 8$	0.2323	0.2323	24.1966	24.1966	**1.0000**	97.7128
	$w = 32,\ s = 16$	0.3009	0.3009	34.3990	34.3990	**1.0000**	96.5487
	$w = 64,\ s = 8$	0.2046	0.2002	20.7980	20.7980	**1.0000**	83.9141
	$w = 64,\ s = 16$	0.2039	0.1995	20.7362	20.7362	**1.0000**	**83.5582**
D-RISE	500	0.2645	0.0702	4.3520	4.3520	0.9062	10.7117
	1000	0.2530	0.0519	4.2019	4.2019	0.9375	10.8382
	5000	**0.2016**	0.0438	3.1713	3.1713	0.9375	10.7438
	10000	0.2246	**0.0389**	3.2496	3.2496	0.9375	10.7094
D-MFPP	500	0.4137	0.0505	**2.9608**	**2.9519**	**1.0000**	15.9700
	1000	0.4227	0.0678	5.8648	5.8648	**1.0000**	15.1166
	5000	0.3944	0.0469	3.8146	3.8058	**1.0000**	16.0809
	10000	0.3893	0.0471	3.6734	3.6734	**1.0000**	16.2579

*: D-Sliding Window scores have been computed excluding the cases where no explanations are produced for images with large-sized object instances.

Table 4
Quantitative results for saliency maps generated with various masks techniques (D-Sliding Window, D-RISE, and D-MFPP) in the Battery Assembly dataset. Results are interpreted vertically within each column (each metric represents a separate column). The best mask configuration for each XAI method and for each metric is highlighted in gray, while the overall best-performing XAI method with its optimal mask configuration is shown in bold.

XAI Method	Num Masks	Deletion (↓)	D-Deletion (↓)	Min-Subset (↓)	D-Min-Subset (↓)	PG (↑)	EBPG (↑)
D-Sliding Window (*)	$w = 32,\ s = 8$	0.5198	0.3258	53.9463	31.2219	0.9855	90.4736
	$w = 32,\ s = 16$	0.4700	0.3046	45.9925	24.7660	0.9788	84.5302
	$w = 64,\ s = 8$	0.2578	**0.0245**	26.4318	**2.4062**	0.9855	55.8151
	$w = 64,\ s = 16$	**0.2570**	0.0246	26.3537	2.4772	0.9884	**54.8283**
D-RISE	500	0.4307	0.0783	29.7843	4.9001	**0.9971**	3.8152
	1000	0.3837	0.0701	26.4242	3.3443	**0.9971**	3.8154
	5000	0.3645	0.0431	25.2766	3.1374	**0.9971**	3.8015
	10000	0.3707	0.0342	**23.3216**	2.9789	**0.9971**	3.7883
D-MFPP	500	0.3327	0.0708	30.7555	5.6594	0.9942	6.2320
	1000	0.3375	0.0630	30.0871	4.8843	**0.9971**	6.1938
	5000	0.3042	0.0427	27.8461	3.4328	**0.9971**	6.1772
	10000	0.3100	0.0409	29.6047	3.8185	**0.9971**	6.1680

*: D-Sliding Window scores have been computed excluding the cases where no explanations are produced for images with large-sized object instances.

masks improves the quality of the saliency maps, and (2) D-MFPP provides cleaner maps with a lower number of masks.

RQ4: Image dimension variation and YOLOv8 complexity

We further analyze the importance of image dimensions in the quality of explanations. For this purpose, we interpolate the image dimensions and train a new object detector for those image dimensions. As reported in Table 5, larger image dimensions (720×1280) yield better results in terms of faithfulness metrics, degrading their value when decreasing the image resolution to 360×640 and 180×320. Nonetheless, PG and EBPG remain relatively stable across resolutions, suggesting that localization is less affected by image size than classification accuracy.

Additionally, Table 5 shows the effect of applying D-RISE with masks generated at varying resolutions. While no clear trend emerges in the classification metrics (with some instances of Deletion increasing while D-Deletion decreases), the localization metrics generally improve with lower mask resolutions. This result is expected, as lower-resolution masks produce less fine-grained explanations but tend to highlight a larger area around the target object. This pattern is clearly visualized in Fig. 9.

Lastly, we investigate whether the use of different levels of parametric complexity of the YOLOv8 object detector affect the expected outcomes. When examining the results in Table 6, no clear insights can be drawn, as the best results were reported by the Medium and Nano models. Nonetheless, it is important to highlight the performance of the Nano model, which despite requiring half the inference time and an order of magnitude less memory for deployment, achieves competitive results with respect to the rest of counterparts in the table.

6. Conclusion

This work has explored the performance of various XAI methods for object detection models, focusing on perturbation-based black-box explanation techniques like LIME, RISE, D-RISE applied to YOLOv8 in real-world object detection tasks in industrial setups. Our contribution is twofold: i) we have introduced D-MFPP, a method tailored for object detection that generates masks based on multi-level superpixels; and ii) we have proposed D-Deletion, a new quantitative metric that accounts for both class probability and localization when evaluating explanations.

Our results have demonstrated that D-RISE consistently outperforms LIME and RISE, particularly with the D-Deletion metric, which improves trustworthiness by delivering more focused, object-specific explanations. The mask generation process has been proven to play a key role in the quality of explanations, with more masks producing better explanations. The proposed D-MFPP technique has shown competitive performance with fewer masks, proving to be an efficient option in resource-constrained deployments. We have also found out that larger image dimensions improve classification metrics, while localization metrics remained stable across resolutions, suggesting that image

Table 5
Explanations comparison for three image dimensions (720×1280, 360×640, and 180×320) and varying resolutions (16×16, 8×8, and 4×4) in the Human-Robot dataset.

Image dimensions	Resolution	Deletion (↓)	D-Deletion (↓)	Min-Subset (↓)	D-Min-Subset (↓)	PG (↑)	EBPG (↑)
720×1280	16×16	0.2016	0.0438	3.1713	3.1713	0.9375	10.7438
	8×8	0.0851	0.0512	4.0050	4.0050	1.0000	11.8188
	4×4	0.0541	0.0504	4.0684	4.0684	0.9166	12.4406
360×640	16×16	0.4331	0.1697	10.6146	9.8644	0.7647	9.8634
	8×8	0.1889	0.1003	6.1750	6.1750	0.8823	11.0002
	4×4	0.1053	0.1013	8.2321	8.2321	0.9166	12.4447
180×320	16×16	0.3668	0.0756	7.2829	7.2829	0.8957	10.5611
	8×8	0.2290	0.1206	10.1774	10.1774	1.000	11.6374
	4×4	0.1531	0.1232	11.6246	11.6246	0.9545	12.0230

size mainly affects classification. No clear patterns have emerged regarding model complexity; nevertheless, the *nano* YOLOv8 model, with its low computational and memory demands, showed promise for real-time industrial applications.

Limitations of the study. While this study provides valuable insights into the explainability of object detectors in real-world scenarios, several limitations must be acknowledged:

- *Limited data*: The real-world datasets considered in the study, particularly the Battery Assembly use case, are limited in size (only 7 data instances). This may restrict the generalization of our findings. Future studies with larger datasets embracing the methodology herein followed could provide more robust conclusions.
- *Generalization of the object detector*: Although the methods presented here are model-agnostic, the study primarily focuses on YOLOv8 (i.e., one stage detector). Extending this work to consider other object detector architectures, such as Faster R-CNN (i.e., two stage detector) or Detection Transformers (DETR, RT-DETR) could ease a broader understanding of XAI methods across different neural architectures for this modeling task.
- *Computation time*: The computation time required to generate explanations can be significant, especially when using perturbation-based methods. Approaches that reduce the number of masks used, or optimize mask generation, could help mitigate this limitation and improve efficiency.

Future Research. We plan to explore hierarchical masking approaches [41,40,21] and feature fusion techniques [38] that further refine pixel saliency across multiple levels, reducing masking efforts and enhancing the quality of the saliency maps. Moreover, we will explore techniques to reduce the computation time while maintaining performance and explanation quality [24], which can be crucial in real-world critical applications wherein explanations have to be furnished fastly and presented to the human for its supervision and validation.

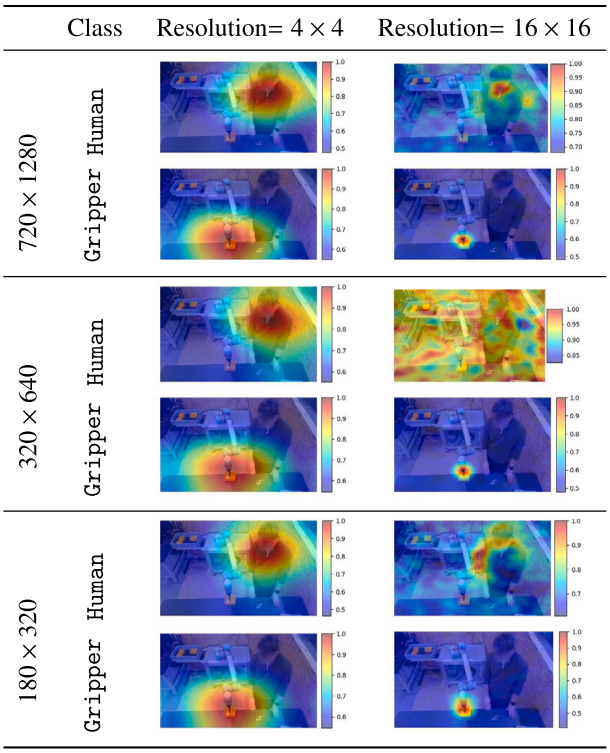


Fig. 9. Heatmaps generated with D-RISE ($p = 0.25$ and resolutions of 4×4 and 16×16) on a scene from the Human-Robot scenario. The effect of varying input image dimensions can be observed.

CRediT authorship contribution statement

Alain Andres: Writing – original draft, Software, Resources, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Aitor Martinez-Seras:** Writing – original draft, Validation. **Ibai Laña:** Writing – review & editing, Supervision, Project administration. **Javier Del Ser:** Writing – review & editing, Validation, Supervision, Funding acquisition, Conceptualization.

Declaration of Generative AI and AI-assisted technologies in the writing process

During the preparation of this work the author(s) used OpenAI's ChatGPT4 in order to enhance the clarity of the writing, improve the readability of the manuscript, and check for possible English language mistakes. After using this tool/service, the authors reviewed and edited

Table 6
Impact of YOLOv8 model complexity (nano, small, medium, large) on the explanations for the same experimental setup.

Model size	Deletion (↓)	D-Deletion (↓)	Min-Subset (↓)	D-Min-Subset (↓)	PG (↑)	EBPG (↑)
Large	0.1735	0.1569	7.1964	7.1964	0.8333	10.0317
Medium	0.2016	0.0438	3.1713	3.1713	0.9375	10.7438
Small	0.3120	0.2294	5.3750	4.3060	0.7352	9.6248
Nano	0.1434	0.0930	4.1177	4.1177	0.9117	10.2061

the content as needed, and take full responsibility for the content of the publication.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Alain Andres reports financial support was provided by European Commission. Ibai Lana reports financial support was provided by European Commission. Javier Del Ser reports financial support was provided by European Commission. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

A. Andres, I. Laña and J. Del Ser receive support from the ULTIMATE project (ref. 101070162) funded by the European Commission under the HORIZON-CL4-DIS program (HORIZON-CL4-2021-HUMAN-01). J. Del Ser and I. Laña also acknowledge funding from the Basque Government (MATHMODE, IT1456-22).

Data availability

The authors do not have permission to share data.

References

[1] K. Abhishek, D. Kamath, Attribution-based XAI methods in computer vision: a review, arXiv preprint, arXiv:2211.14736, 2022.

[2] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, S. Süsstrunk, SLIC superpixels, https://infoscience.epfl.ch/handle/20.500.14299/50758, 2010.

[3] S. Ali, T. Abuhmed, S. El-Sappagh, K. Muhammad, J.M. Alonso-Moral, R. Confalonieri, R. Guidotti, J. Del Ser, N. Díaz-Rodríguez, F. Herrera, Explainable Artificial Intelligence (XAI): what we know and what is left to attain Trustworthy Artificial Intelligence, Inf. Fusion 99 (2023) 101805, https://doi.org/10.1016/j.inffus.2023.101805, https://www.sciencedirect.com/science/article/pii/S1566253523001148.

[4] S. Bach, A. Binder, G. Montavon, F. Klauschen, K.R. Müller, W. Samek, On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation, PLoS ONE 10 (2015) e0130140, https://doi.org/10.1371/journal.pone.0130140, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4498753/.

[5] A. Barredo Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. Garcia, S. Gil-Lopez, D. Molina, R. Benjamins, R. Chatila, F. Herrera, Explainable Artificial Intelligence (XAI): concepts, taxonomies, opportunities and challenges toward responsible AI, Inf. Fusion 58 (2020) 82–115, https://doi.org/10.1016/j.inffus.2019.12.012, https://www.sciencedirect.com/science/article/pii/S1566253519308103.

[6] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, S. Zagoruyko, End-to-end object detection with transformers, in: A. Vedaldi, H. Bischof, T. Brox, J.M. Frahm (Eds.), Computer Vision – ECCV 2020, in: Lecture Notes in Computer Science., vol. 12346, Springer International Publishing, Cham, 2020, pp. 213–229, https://link.springer.com/10.1007/978-3-030-58452-8_13.

[7] A. Chattopadhay, A. Sarkar, P. Howlader, V.N. Balasubramanian, Grad-CAM++: generalized gradient-based visual explanations for deep convolutional networks, in: 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), 2018, pp. 839–847, https://ieeexplore.ieee.org/abstract/document/8354201.

[8] H.Y. Chen, C.H. Lee, Vibration signals analysis by Explainable Artificial Intelligence (XAI) approach: application on bearing faults diagnosis, IEEE Access 8 (2020) 134246–134256, https://doi.org/10.1109/ACCESS.2020.3006491, https://ieeexplore.ieee.org/document/9131692/?arnumber=9131692, conference Name: IEEE Access.

[9] T.C.T. Chen, Explainable Artificial Intelligence (XAI), in: Manufacturing: Methodology, Tools, and Applications, in: SpringerBriefs in Applied Sciences and Technology, Springer International Publishing, Cham, 2023, https://link.springer.com/10.1007/978-3-031-27961-4.

[10] A. Gevaert, A.J. Rousseau, T. Becker, D. Valkenborg, T. De Bie, Y. Saeys, Evaluating feature attribution methods in the image domain, Mach. Learn. 113 (2024) 6019–6064, https://doi.org/10.1007/s10994-024-06550-x, arXiv:2202.12270 [cs], http://arxiv.org/abs/2202.12270.

[11] D. Gunning, M. Stefik, J. Choi, T. Miller, S. Stumpf, G.Z. Yang, XAI—explainable artificial intelligence, Sci. Robot. 4 (2019) eaay7120, https://doi.org/10.1126/scirobotics.aay7120, https://www.science.org/doi/10.1126/scirobotics.aay7120, publisher: American Association for the Advancement of Science.

[12] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, arXiv:1512.03385 [cs], http://arxiv.org/abs/1512.03385, 2015.

[13] A. Hedström, L. Weber, D. Bareeva, D. Krakowczyk, F. Motzkus, W. Samek, S. Lapuschkin, Quantus: an explainable AI toolkit for responsible evaluation of neural network explanations and beyond, J. Mach. Learn. Res. 24 (2023) 1–11.

[14] A. Kirchknopf, D. Slijepcevic, I. Wunderlich, M. Breiter, J. Traxler, M. Zeppelzauer, Explaining YOLO: leveraging grad-CAM to explain object detections, https://doi.org/10.3217/978-3-85125-869-1-13, http://arxiv.org/abs/2211.12108, arXiv:2211.12108 [cs], 2022.

[15] A. Kotriwala, B. Klöpper, M. Dix, G. Gopalakrishnan, D. Ziobro, A. Potschka, XAI for operations in the process industry-applications, theses, and research directions, in: AAAI Spring Symposium: Combining Machine Learning with Knowledge Engineering, 2021, pp. 1–12.

[16] R. Kozik, D. Puchalski, A. Pawlicka, S. Buś, J. Główka, K. Chandramouli, M. Tiemann, M. Pawlicki, R. Renk, M. Choraś, ULTIMATE project toolkit for robotic AI-based data analysis and visualization, in: N.T. Nguyen, R. Chbeir, Y. Manolopoulos, H. Fujita, T.P. Hong, L.M. Nguyen, K. Wojtkiewicz (Eds.), Intelligent Information and Database Systems, Springer Nature, Singapore, 2024, pp. 44–55.

[17] T.T.H. Le, A.T. Prihatno, Y.E. Oktian, H. Kang, H. Kim, Exploring Local Explanation of Practical Industrial AI Applications: A Systematic Literature Review, Applied Sciences, vol. 13, 2023, p. 5809, https://www.mdpi.com/2076-3417/13/9/5809, number: 9 Publisher: Multidisciplinary Digital Publishing Institute.

[18] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.Y. Fu, A.C. Berg, Ssd: single shot multibox detector, in: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14, Springer, 2016, pp. 21–37.

[19] S.M. Lundberg, S.I. Lee, A unified approach to interpreting model predictions, in: Neural Information Processing System, Long Beach, USA, 2017.

[20] G. Montavon, A. Binder, S. Lapuschkin, W. Samek, K.R. Müller, Layer-Wise relevance propagation: an overview, in: W. Samek, G. Montavon, A. Vedaldi, L.K. Hansen, K.R. Müller (Eds.), Explainable AI: Interpreting, Explaining and Visualizing Deep Learning, Springer International Publishing, Cham, 2019, pp. 193–209.

[21] M. Moradi, K. Yan, D. Colwell, M. Samwald, R. Asgari, Model-agnostic explainable artificial intelligence for object detection in image data, Eng. Appl. Artif. Intell. 137 (2024) 109183, https://doi.org/10.1016/j.engappai.2024.109183, https://www.sciencedirect.com/science/article/pii/S0952197624013411.

[22] K. Muhammad, A. Ullah, J. Lloret, J. Del Ser, V.H.C. de Albuquerque, Deep learning for safe autonomous driving: current challenges and future directions, IEEE Trans. Intell. Transp. Syst. 22 (2020) 4316–4336.

[23] S. Naddaf-Sh, M.M. Naddaf-Sh, H. Zargarzadeh, M. Dalton, S. Ramezani, G. Elpers, V.S. Baburao, A.R. Kashani, Real-time explainable multiclass object detection for quality assessment in 2-dimensional radiography images, Complexity 2022 (2022) 4637939, https://doi.org/10.1155/2022/4637939, https://onlinelibrary.wiley.com/doi/abs/10.1155/2022/4637939, https://onlinelibrary.wiley.com/doi/pdf/10.1155/2022/4637939.

[24] K. Nguyen, H. Nguyen, K. Nguyen, B. Truong, T. Phan, H. Cao, Efficient and concise explanations for object detection with Gaussian-class activation mapping explainer, in: Canadian Conference on Artificial Intelligence, Ontario, 2024.

[25] V. Petsiuk, RISE: randomized input sampling for explanation of black-box models, in: Proceedings of the British Machine Vision Conference (BMVC), 2018.

[26] V. Petsiuk, R. Jain, V. Manjunatha, V.I. Morariu, A. Mehra, V. Ordonez, K. Saenko, Black-box explanation of object detectors via saliency maps, http://arxiv.org/abs/2006.03204, arXiv:2006.03204 [cs], 2021.

[27] D. Reis, J. Kupec, J. Hong, A. Daoudi, Real-time flying object detection with YOLOv8, http://arxiv.org/abs/2305.09972, arXiv:2305.09972 [cs], 2024.

[28] S. Ren, K. He, R. Girshick, J. Sun, Faster R-CNN: towards real-time object detection with region proposal networks, IEEE Trans. Pattern Anal. Mach. Intell. 39 (2017) 1137–1149, https://doi.org/10.1109/TPAMI.2016.2577031, http://ieeexplore.ieee.org/document/7485869/.

[29] M.T. Ribeiro, S. Singh, C. Guestrin, "Why should I trust you?": explaining the predictions of any classifier, in: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, San Francisco California USA, 2016, pp. 1135–1144, https://dl.acm.org/doi/10.1145/2939672.2939778.

[30] T.Y. Ross, G. Dollár, Focal loss for dense object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 2980–2988.

[31] M. Ryo, Explainable artificial intelligence and interpretable machine learning for agricultural data analysis, Artif. Intell. Agricult. 6 (2022) 257–265, https://doi.org/10.1016/j.aiia.2022.11.003, https://www.sciencedirect.com/science/article/pii/S2589721722000216.

[32] K. Sahatova, K. Balabaeva, An overview and comparison of XAI methods for object detection in computer tomography, Proc. Comput. Sci. 212 (2022) 209–219, https://doi.org/10.1016/j.procs.2022.11.005, https://www.sciencedirect.com/science/article/pii/S1877050922016969.

[33] J.H. Sejr, P. Schneider-Kamp, N. Ayoub, Surrogate Object Detection Explainer (SODEx) with YOLOv4 and LIME, Mach. Learn. Knowl. Extr. 3 (2021) 662–671, https://doi.org/10.3390/make3030033, https://www.mdpi.com/2504-4990/3/3/33.

[34] R.R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, D. Batra, Grad-CAM: visual explanations from deep networks via gradient-based localization, in: 2017 IEEE International Conference on Computer Vision (ICCV), 2017, pp. 618–626, iSSN: 2380-7504, https://ieeexplore.ieee.org/document/8237336.

[35] O. Serradilla, E. Zugasti, C. Cernuda, A. Aranburu, J.R. De Okariz, U. Zurutuza, Interpreting remaining useful life estimations combining Explainable Artificial Intelligence and domain knowledge in industrial machinery, in: 2020 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), IEEE, Glasgow, United Kingdom, 2020, pp. 1–8, https://ieeexplore.ieee.org/document/9177537/.

[36] M. Sundararajan, A. Taly, Q. Yan, Axiomatic attribution for deep networks, in: Proceedings of the 34th International Conference on Machine Learning, PMLR, 2017, pp. 3319–3328, iSSN: 2640-3498, https://proceedings.mlr.press/v70/sundararajan17a.html.

[37] J. Terven, D.M. Córdova-Esparza, J.A. Romero-González, A comprehensive review of YOLO architectures in computer vision: from YOLOv1 to YOLOv8 and YOLO-NAS, Mach. Learn. Knowl. Extr. 5 (2023) 1680–1716, https://doi.org/10.3390/make5040083, https://www.mdpi.com/2504-4990/5/4/83, number: 4 Publisher: Multidisciplinary Digital Publishing Institute.

[38] V.B. Truong, T.T.H. Nguyen, V.T.K. Nguyen, Q.K. Nguyen, Q.H. Cao, Towards better explanations for object detection, in: Proceedings of the 15th Asian Conference on Machine Learning, PMLR, 2024, pp. 1385–1400.

[39] H. Wang, Z. Wang, M. Du, F. Yang, Z. Zhang, S. Ding, P. Mardziel, X. Hu, Score-CAM: score-weighted visual explanations for convolutional neural networks, in: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE, Seattle, WA, USA, 2020, pp. 111–119, https://ieeexplore.ieee.org/document/9150840/.

[40] Y. Yan, T. Jiang, X. Li, L. Sun, J. Zhu, J. Lin, Model-agnostic progressive saliency map generation for object detector, Image Vis. Comput. 145 (2024), https://doi.org/10.1016/j.imavis.2024.104988.

[41] Y. Yan, X. Li, Y. Zhan, L. Sun, J. Zhu, GSM-HM: generation of saliency maps for black-box object detection model based on hierarchical masking, IEEE Access 10 (2022) 98268–98277, https://doi.org/10.1109/ACCESS.2022.3206379, https://ieeexplore.ieee.org/document/9888073/.

[42] Q. Yang, X. Zhu, J.K. Fwu, Y. Ye, G. You, Y. Zhu, MFPP: morphological fragmental perturbation pyramid for black-box model explanations, http://arxiv.org/abs/2006.02659, arXiv:2006.02659 [cs], 2020.

[43] M.D. Zeiler, R. Fergus, Visualizing and understanding convolutional networks, in: D. Fleet, T. Pajdla, B. Schiele, T. Tuytelaars (Eds.), Computer Vision – ECCV 2014, in: Lecture Notes in Computer Science, vol. 8689, Springer International Publishing, Cham, 2014, pp. 818–833, http://link.springer.com/10.1007/978-3-319-10590-1_53.

[44] J. Zhang, S.A. Bargal, Z. Lin, J. Brandt, X. Shen, S. Sclaroff, Top-down neural attention by excitation backprop, Int. J. Comput. Vis. 126 (2018) 1084–1102, https://doi.org/10.1007/s11263-017-1059-x.

[45] Z. Zou, K. Chen, Z. Shi, Y. Guo, J. Ye, Object detection in 20 years: a survey, Proc. IEEE 111 (2023) 257–276.