IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 8 January 2024, accepted 15 February 2024, date of publication 26 February 2024, date of current version 1 March 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3369890

RESEARCH ARTICLE

Fast Explanation Using Shapley Value for Object Detection

MICHIHIRO KUROKI, (Member, IEEE), AND TOSHIHIKO YAMASAKI, (Member, IEEE)
Graduate School of Information Science and Technology, The University of Tokyo, Tokyo 113-8656, Japan

Corresponding author: Michihiro Kuroki (kuroki@cvm.t.u-tokyo.ac.jp)

The work of Toshihiko Yamasaki was supported in part by the JSPS KAKENHI under Grant JP22H03640 and in part by the Institute for AI and Beyond of The University of Tokyo.

ABSTRACT In explainable artificial intelligence (XAI) for object detection, saliency maps are employed to highlight important regions for a learned model's prediction. However, a trade-off exists wherein the higher the accuracy of explanation results, the higher the computational cost, posing a challenge for practical applications. Therefore, this study proposes a novel XAI method for object detection to address this challenge. In recent years, research on XAI that satisfies desirable properties for explanatory validity by introducing the Shapley value has been widely conducted. However, a common drawback across these approaches is the high computational cost, which has hindered broad implementation. Our proposed method utilizes an explainer model that learns to estimate the Shapley value and provides a reliable explanation for object detection in a real-time inference. This framework can be applied to various object detectors in a model-agnostic manner. Through quantitative evaluation, we experimentally demonstrate that our method achieves the fastest explanation while delivering superior performance compared with other existing methods.

INDEX TERMS Explainable artificial intelligence, object recognition, shapley value.

I. INTRODUCTION

Explaining the underlying reasons behind AI's decisions is challenging because of its inherent black-box nature. This challenge becomes pronounced in safety-critical domains, such as autonomous driving and medicine, where addressing this issue becomes crucial to ensure the secure utilization and social acceptance of AI-equipped systems. In recent years, explainable AI (XAI) has gained attention as a promising approach to elucidate the rationale behind AI's decisions, with extensive research in the field of computer vision, encompassing tasks such as image classification and object detection. The most widely used method in these tasks is visualizing the inference rationale through a saliency map. This map visualizes feature attributions, which indicate the importance of each pixel for a model's prediction, in the form of a heatmap. While various methods have been proposed, most have focused on the basic task of image classification, with relatively few designed for the more intricate task of object detection. Despite the substantial demand for object detection in various domains, XAI for object detection has not yet reached a level of practical applicability owing to some challenges. Therefore, this study tackles these challenges to advance research toward the practical application of XAI for object detection.

Fig. 1 presents examples of saliency maps targeting object detection results. Notably, determining the best method is challenging owing to substantial differences in explanations generated by different methods. Concerning this issue, some studies [1], [2], [3] emphasize the importance of explanatory validity by noting that certain methods provide explanations that do not align with the model's predictions. To ensure explanatory validity, methods introducing the Shapley value [4] have recently gained attention. The Shapley value originates from a cooperative game theory and offers a method that can achieve a fair and justified distribution of rewards. In addition, the Shapley value has the advantage of satisfying properties desirable for ensuring the validity of explanations. While these methods can yield more accurate explanations, they entail a high computational cost for estimating the Shapley value. Consequently, a significant trade-off arises in generating saliency maps between achieving

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

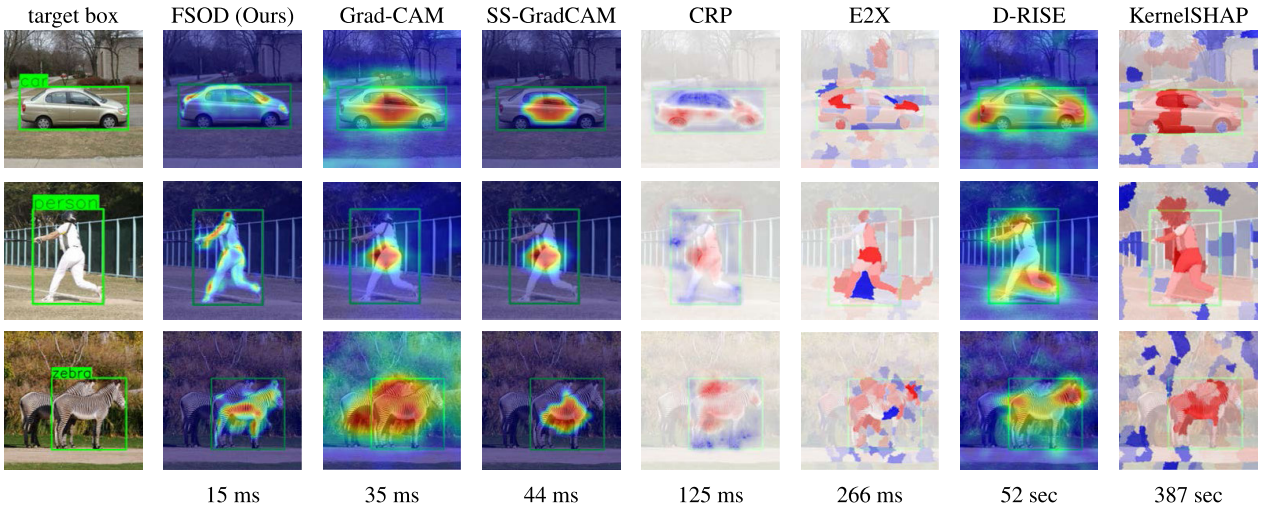

FIGURE 1. Comparison results of saliency maps generated by the existing methods. Many methods represent the values of feature attributions in the form of a heatmap; some methods indicate positive values in red and negative values in blue, following the official implementations. The average processing times on a single Tesla V100 GPU are presented beneath their respective saliency maps.

high explanatory accuracy and reducing computational costs, and this trade-off applies to other XAI methods as well. Fig. 1 also includes the average processing time in generating those explanations. It indicates that a fast method, such as SS-Grad-CAM [5], which utilizes information directly from within the AI model to generate an explanation result, highlights regions that humans find challenging to interpret. Conversely, a method that accurately captures the target objects, such as D-RISE [6], which obtains an explanation result from changes in the model's outputs derived from multiple input samplings with added perturbations, involves a substantial increase in computational cost. This increase incurs a significant challenge in the practical application of XAI for object detection because there are multiple objects per image for explanation. Thus, reducing the computational cost per object while maintaining highly reliable explanations is crucial in practical applications.

To address this trade-off, FastSHAP [7] trains an explainer model to estimate the Shapley value and achieve real-time and highly accurate explanations during the inference phase. This approach has succeeded in image classification tasks but not yet in object detection tasks. To extend its application to object detection, several challenges stemming from the differences between classification and detection tasks need to be addressed. In this study, we propose a novel XAI for object detection named FSOD (Fast explanation using Shapley value for Object Detection), capable of providing reliable explanations in real time. It demonstrates the effectiveness of an explainer model in object detection tasks. The technical contributions of this study are as follows.

- We introduced a novel XAI for object detection that quickly generates explanations based on the Shapley value, overcoming the challenges of extending an explainer model to object detection tasks.
- We applied our method to various object detectors to demonstrate that our framework is adaptable to a wide range of object detectors.
- We validated our method using benchmark datasets through various evaluation metrics, demonstrating superior performance to other methods while running fastest among them.

The remainder of this paper is organized as follows. First, we summarize related studies and provide a derivation of our method. We then experimentally demonstrate our method's performance and validity. Finally, concluding remarks are presented.

II. RELATED WORKS

A. VISUALIZATION OF EXPLANATION

In XAIs that employ a saliency map as an explanation output, methods are primarily designed for image classification and object detection tasks. In XAI for image classification, several methods have been developed to compute feature attributions, indicating the importance of each pixel in a prediction. They are mainly classified into three categories: back-propagation-based, activation-map-based, and perturbation-based methods. Back-propagation-based methods [2], [8], [9] compute feature attributions by back-propagating a classification probability score, referred to as *relevance*, through the network. For instance, Layer-wise Relevance Propagation (LRP) [10] calculates pixel-wise feature attributions by propagating relevances from the output layer back to the input layer. Contrastive Relevance Propagation (CRP) [11] extends LRP and highlights the relevances originating from the true class object by contrasting it with those originating from other classes. Activation-map-based methods [12], [13], [14] leverage weighted sums of feature maps in the last convolutional neural network (CNN) layer in the AI model to generate explanations. For instance, Grad-CAM [15] calculates weighted sums of the feature maps by utilizing the gradients in the neural network as the weights. Perturbation-based methods [16], [17], [18] examine the change of output scores resulting from perturbations added to input samples. For instance, RISE [16] calculates the

weighted sums of the random samplings of binary masks using their corresponding output classification scores as weights to generate a saliency map.

In XAI for object detection, numerous methods extend the existing methods designed for image classification by considering both localization and classification of a target object detection. As an application of back-propagation-based methods, Karasmanoglou et al. [19] applied CRP to the YOLO detectors [20], [21], which are widely utilized owing to their high processing speed, to explain detection results. In this method, the calculation of the relevances is restricted to regions near the target bounding box, aligning with the target class label. Similarly, E2X [22] utilizes feature attributions derived from the backpropagation scores. In E2X, the input image is divided into superpixels, and the average attributions within them are used to mitigate pixel-wise calculation noises. Activation-map-based methods, such as Spatially-Sensitive Grad-CAM [5], enhance Grad-CAM by integrating a spatial map that weights feature attributions based on proximity to the target object. Perturbation-based methods, such as D-RISE [6], extend RISE by adopting the detection similarity as an output score to consider both prediction and localization. By observing the output score changes due to input perturbations, feature attributions can be calculated. Fig. 1 shows the comparison of these methods, with the detection results from the small YOLOv5 model (YOLOv5s)[1] set as an explanation target. Back-propagation-based [19] and activation-map-based methods [5] offer faster processing speed owing to their ability to leverage internal information within the object detector. However, these saliency maps may fail to correctly capture the target object. Conversely, perturbation-based methods [6] provide a more interpretable saliency map at the expense of increased input sampling time. In generating explanation results, there exists a trade-off between processing speed and the reliability of the output. Striking a balance between these factors becomes crucial for the practical application.

B. SHAPLEY VALUE

The Shapley value was originally developed as a method for fair reward distribution in cooperative game theory. It satisfies several desirable properties, known as *axioms* [2], [23], [24]. The axioms include properties, such as *Dummy*, which asserts that feature attributions with no contribution to the prediction are zero, and *Efficiency*, stipulating that the sum of feature attributions equals the prediction score. Owing to its ability to satisfy these properties, the Shapley value has been actively applied to various XAIs [23], [24], [25], [26] for image classification. Nevertheless, its major drawback is the high computational complexity. If N represents all features for a model's input, the Shapley value of a certain feature $i \in N$ can be described as a feature attribution ϕ_i as follows,

$$\phi_i = \sum_{S \subseteq N \setminus i} \frac{|S|! * (d - |S| - 1)!}{d!} (v(S \cup i) - v(S)). \quad (1)$$

Here, S indicates a subset of features excluding i. d $(= |N|)$ denotes the number of input dimensions for the model's input. $v(S)$ refers to a value function of the model using S as an input. The attribution of i can be obtained by calculating the average of the marginal contribution, defined as the change in output resulting from adding i to the subset. Because it requires considering all patterns of S, the computational cost of Eq. 1 becomes $\mathcal{O}(2^d)$, requiring an enormous amount of processing time. To compute these values efficiently, an approach beyond direct computation is necessary.

III. PROPOSED METHOD

A. MOTIVATION

Despite the numerous methods integrating the Shapley value into XAI, they often encounter computational complexity challenges. Consequently, several approximations of the Shapley value have emerged to address this issue. One widely recognized approximation is KernelSHAP [24], which transforms the Shapley value computation to the determination of the weights of a linear model. Nevertheless, KernelSHAP still faces the challenge of requiring extensive sampling each time to deduce the weights for explaining each instance. FastSHAP [7] has improved upon KernelSHAP by facilitating the Shapley value estimation across multiple instances with a single training cycle for the explainer model. This approach achieves a more favorable balance between processing speed and explanation accuracy than other Shapley value-based methods. However, the effectiveness of an explainer model has mainly been noted in simpler contexts where the AI model (1) outputs only class predictions as observed in image classification, (2) deals with small images (e.g., up to 224 × 224) as encountered in ImageNet [27], and (3) handles one prediction corresponding to one image. As mentioned earlier, our focus is XAI for response-time-critical object detection. Object detection tasks involve more complex scenarios, prohibiting the straightforward application of FastSHAP. This study outlines the challenges of extending an explainer model to object detection and presents strategies to overcome these challenges.

B. TECHNICAL BACKGROUND

We provide a detailed explanation of KernelSHAP [24] and FastSHAP [7], which are promising approximations of the Shapley value. Let $\boldsymbol{x}$ be an input image consisting of d pixels, and c be a classification label. Here, $\mathbf{s} \in \{0, 1\}^d$ is used to denote subsets of the pixel indices $\{1, \ldots, d\}$.

KernelSHAP approximates the computation of Eq. 1 using the weights of a linear model $\Phi_{\boldsymbol{x},c} \in \mathbb{R}^d$ and a value function $v_{\boldsymbol{x},c}(\mathbf{s}) : \mathbb{R}^d \mapsto \mathbb{R}$ for a given pair of $(\boldsymbol{x}, c)$ as follows.

$$\mathcal{L}(v_{\boldsymbol{x},c}, \Phi_{\boldsymbol{x},c}, p) = \sum_{\mathbf{s}} \left\{ v_{\boldsymbol{x},c}(\mathbf{s}) - v_{\boldsymbol{x},c}(\mathbf{0}) - \mathbf{s}^{\mathrm{T}} \Phi_{\boldsymbol{x},c} \right\}^2 p(\mathbf{s}), \quad (2)$$

[1] Glenn Jocher and contributors. Yolov5, Accessed 2022. https://github.com/ultralytics/yolov5

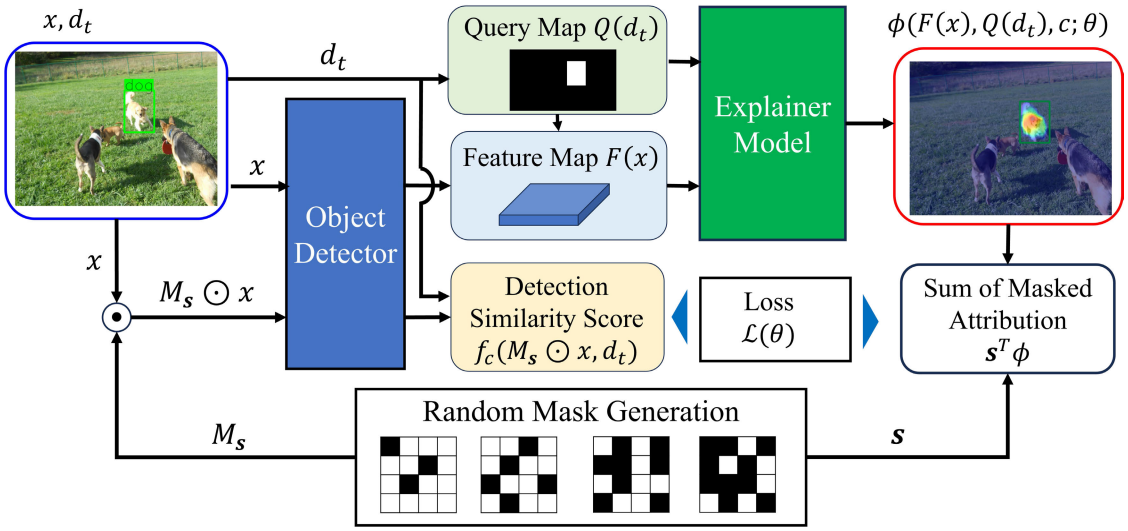

FIGURE 2. Overview of our proposed method, FSOD. A framework is presented for learning the parameter function $\phi(F(x), Q(d_t), c; \theta)$ of the explainer model, taking an input image x and a target object detection d_t as inputs.

$$p(\mathbf{s}) \propto \frac{d-1}{\binom{d}{\mathbf{1}^{\mathrm{T}}\mathbf{s}} \cdot \mathbf{1}^{\mathrm{T}}\mathbf{s} \cdot (d - \mathbf{1}^{\mathrm{T}}\mathbf{s})}, \tag{3}$$

$$\phi(v_{\boldsymbol{x},c}) = \underset{\Phi_{\boldsymbol{x},c}}{\operatorname{argmin}}\, \mathcal{L}(v_{\boldsymbol{x},c}, \Phi_{\boldsymbol{x},c}, p). \tag{4}$$

Here, $\phi(v_{\boldsymbol{x},c}) = \{\phi_1, \ldots, \phi_i, \ldots, \phi_d\}$ indicates the approximated Shapley value for each pixel. $\mathbf{1}$ and $\mathbf{0}$ represent all-ones and all-zeros d-dimensions vectors. KernelSHAP minimizes the loss function in Eq. 2 by sampling $\mathbf{s}$ based on the probability from Eq. 3, training the weights $\Phi_{\boldsymbol{x},c}$. Then, the KernelSHAP estimates the Shapley value $\phi(v_{\boldsymbol{x},c})$. Adopting the value function described in Eq. 6 facilitates the application of KernelSHAP to object detection. To mitigate calculation noise in explanation results, Hogan et al. [28] divided the image into superpixels, applying Eq. 2 to each to determine superpixel-wise feature attributions. Fig. 1 shows the saliency maps obtained using KernelSHAP with 5000 input samplings. Notably, the explanation results depend on the superpixel segmentation, and it faces high computational complexity owing to the extensive sampling required for learning the linear model's weights. Moreover, a significant challenge arises as it requires the learning of weights $\Phi_{\boldsymbol{x},c}$ for each pair of (x, c) every time, leading to a considerable amount of time. FastSHAP [7] addressed this issue by leveraging an explainer model that is trainable in a single process.

FastSHAP utilizes an explainer model inspired by the Shapley value's weighted least squares property [24]. The Shapley value can be obtained by training the explainer model using the following weighted least square loss function:

$$\mathcal{L}(\theta) = \mathop{\mathbb{E}}_{p(\boldsymbol{x})} \mathop{\mathbb{E}}_{\mathrm{Unif}(c)} \mathop{\mathbb{E}}_{p(\mathbf{s})} \Big[\left(v_{\boldsymbol{x},c}(\mathbf{s}) - v_{\boldsymbol{x},c}(\mathbf{0}) - \mathbf{s}^{\mathrm{T}} \phi(\boldsymbol{x}, c; \theta) \right)^2 \Big]. \tag{5}$$

Here, $\mathrm{Unif}(c)$ and $p(\boldsymbol{x})$ represent a uniform distribution over classes and the distribution of $\boldsymbol{x}$ within a training dataset, respectively. $\phi(\boldsymbol{x}, c; \theta) : \mathbb{R}^d \mapsto \mathbb{R}^d$ indicates a learned parametric function of an explainer model to estimate the Shapley value. The training is performed by sampling $\mathbf{s}$ according to the distribution of $p(\mathbf{s})$ for a variety of images of $\boldsymbol{x}$. After minimizing the loss function of Eq. 5 and optimizing the parameter θ^* in the neural network of the explainer model, the output components of $\phi(\boldsymbol{x}, c; \theta^*)$ approximates the Shapley value for each pixel. Once trained, the explainer model can be applied to various pairs of $(\boldsymbol{x}, c)$ without further training. However, this approach has only been applied to image classification tasks and remains unexplored for object detection tasks. In the following section, we discuss the challenges and their solutions for extending the explainer model to object detection.

C. EXTENTION TO OBJECT DETECTION

The overall framework of our method, called FSOD (Fast explanation using Shapley value for Object Detection), is shown in Fig.2. This framework introduces three novel approaches. We provide detailed explanations for each approach, along with their corresponding challenges.

1) VALUE FUNCTION FOR OBJECT DETECTION

In the case of image classification tasks, it was sufficient to define the classification prediction score of the model as the value function. However, in object detection tasks, the value function should consider both classification and localization of a target object. To this end, we define the value function $v_{\boldsymbol{x},c}(\mathbf{s})$ as follows:

$$v_{\boldsymbol{x},c}(\mathbf{s}) = f_c(\boldsymbol{M}_{\mathbf{s}} \odot \boldsymbol{x}, \boldsymbol{d}_t), \tag{6}$$

$$f_c(\boldsymbol{M}_{\mathbf{s}} \odot \boldsymbol{x}, \boldsymbol{d}_t) = \max_{\boldsymbol{d}_j \in \mathcal{D}(\boldsymbol{M}_{\mathbf{s}} \odot \boldsymbol{x})} \mathrm{IoU}(\boldsymbol{d}_j, \boldsymbol{d}_t) \cdot o_j \cdot p_j^c. \tag{7}$$

Here, $\odot$ is element-wise multiplication, and $\boldsymbol{M}_{\mathbf{s}}$ indicates a random binary mask corresponding to pixel indices of $\mathbf{s}$. Consequently, $\boldsymbol{M}_{\mathbf{s}} \odot \boldsymbol{x}$ produces a masked image, retaining pixels at the indices of $\mathbf{s}$ from $\boldsymbol{x}$ and masking the others.

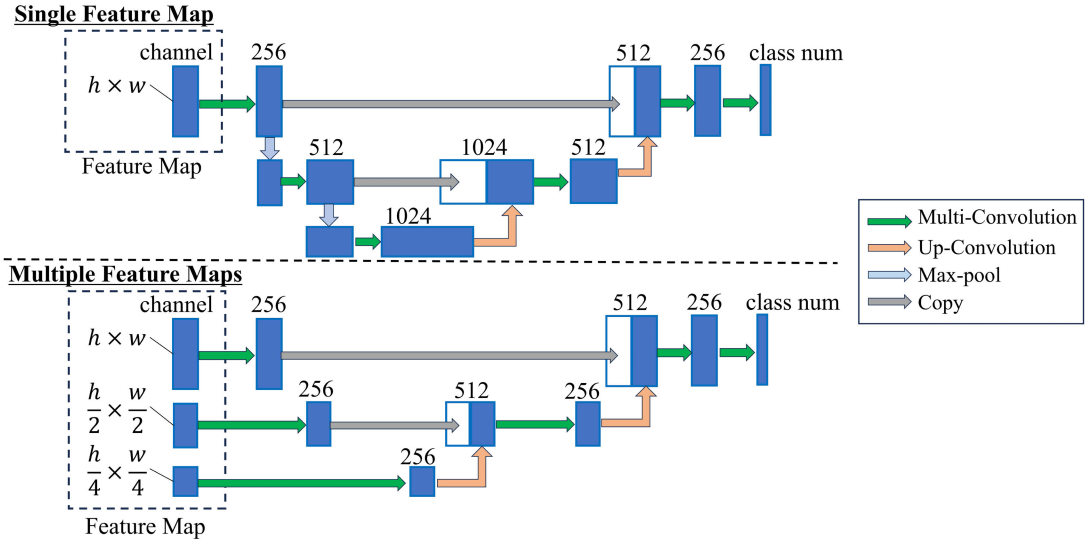

FIGURE 3. Schematic image of the neural network in the explainer model. The architecture is illustrated for the case with a single feature map (top) and multiple feature maps (bottom).

$f_c : \boldsymbol{M}_\mathbf{s} \odot \boldsymbol{x} \mapsto \mathbb{R}$ indicates the detection similarity function. $\mathcal{D}$ denotes the function of an object detector and $\boldsymbol{d}_j$ indicates the vector representation of its detection results. $\boldsymbol{d}_t$ is a vector representation of the target object detection to be explained, including the bounding box and classification information. The localization similarity between $\boldsymbol{d}_t$ and $\boldsymbol{d}_j$ is denoted by the Intersection over Union (IoU), which measures the degree of the overlap in the two bounding boxes. The term o_j represents the objectness score of $\boldsymbol{d}_j$, while p_j^c indicates its classification score for the class label c. Consequently, this score function f_c assigns a detection score reflecting the highest localization and classification similarity among the detection results from a masked image.

2) UTILIZATION OF FEATURE MAPS AS INPUT

The image size of datasets for object detection, such as COCO [29] and VOC [30] is approximately 600×600 pixels or larger. This is larger than that of datasets for image classification, such as CIFAR-10 (32×32 pixels) [31] and ImageNet (224×224 pixels) [27] because object detection tasks require an image to contain multiple objects. The larger image size complicates the explainer model's task of learning image features, potentially leading to feature representations that diverge from those identified by the object detector. To address this issue, we incorporate a function for generating feature maps $F(\boldsymbol{x})$, which represents the output of the backbone network in the object detector and is a prevalent component in many CNN-based object detectors. By condensing spatial information from the image, these feature maps enable a more synchronized learning process between the explainer model and the object detector. Some object detectors employ a feature pyramid network (FPN) [32] to capture feature maps across multiple scales. In that case, the explainer model is designed to utilize feature maps from all scaling layers, as shown in Fig. 3. The operations such as convolution described in Fig. 3 are implemented to be equivalent to those in UNet [33].

3) OBJECT-SPECIFIC EXPLANATIONS

Contrary to image classification XAI methods, which typically produce a single explanation for an entire image, object detection XAI methods require generating unique explanations for each detected object. Therefore, it is essential to provide the explainer model with information specifying the objects to be explained. To facilitate this, our approach incorporates a query map that provides spatial context for the target object under explanation. This query map, denoted as $Q(\boldsymbol{d}_t)$ and based on the target object detection $\boldsymbol{d}_t$, assigns a value of 1 to pixels within the target object's bounding box, forming a binary map. This map is merged with the explainer model's input $F(\boldsymbol{x})$ along the channel direction and resized to align with the feature map size across all scaling layers.

D. EXPLAINER MODEL FOR OBJECT DETECTION

The overall framework of our method and our explainer model are illustrated in Figs. 2 and 3, respectively. The explainer model's architecture is inspired by UNet [33]. When our approaches are integrated, the loss function for the training of our explainer model can be rewritten as follows:

$$\mathcal{L}(\theta) = \mathop{\mathbb{E}}_{p(\boldsymbol{x})} \mathop{\mathbb{E}}_{\mathrm{Unif}(c)} \mathop{\mathbb{E}}_{p(\mathbf{s})} \Big[\Big\{ f_c(\boldsymbol{M}_\mathbf{s} \odot \boldsymbol{x}, \boldsymbol{d}_t) - \mathbf{s}^{\mathrm{T}} \phi(F(\boldsymbol{x}), Q(\boldsymbol{d}_t), c; \theta) \Big\}^2 \Big]. \tag{8}$$

The steps of training an explainer model are as follows.

1) Obtain an image $\boldsymbol{x}$ and compute the feature map $F(\boldsymbol{x})$ from the object detector, and then, estimate an explanation map $\phi(F(\boldsymbol{x}), Q(\boldsymbol{d}_t), c; \theta)$.
2) Sample $\mathbf{s}$ randomly based on the probability distribution $p(\mathbf{s})$ and generate the corresponding mask $\boldsymbol{M}_\mathbf{s}$.

TABLE 1. Comparison results in quantitative evaluation. It presents the average results of the Energy-based Pointing Game (EPG), Visualization Explanation Accuracy (VEA), Insertion (Ins.), and Deletion (Del.), as well as the average frames per second (FPS) on the target dataset. Values in parentheses denote standard errors. Considering a 95% confidence interval, the candidate with the best candidate is highlighted in bold, and the second-best candidate is emphasized with an underline. The asterisk * denotes a statistically significant difference ($p < 0.05$) between those values.

	COCO [29]					VOC [30]				
	EPG(↑)	VEA(↑)	Ins.(↑)	Del.(↓)	FPS (↑)	EPG(↑)	VEA(↑)	Ins.(↑)	Del.(↓)	FPS (↑)
Grad-CAM	0.154 (0,008)	0.114 (0.006)	0.515 (0.010)	0.147 (0.008)	30.75 (0.25)	0.165 (0.014)	0.131 (0.011)	0.393 (0.014)	0.133 (0.010)	27.70 (0.38)
SS-GradCAM	0.729 (0.013)	0.349 (0.008)	0.467 (0.010)	0.110 (0.006)	21.65 (0.17)	**0.590** (0.022)	**0.264** (0.011)	0.372 (0.016)	0.070 (0.006)	25.90 (0.33)
CRP	0.321 (0.014)	0.055 (0.004)	0.186 (0.008)	0.327 (0.012)	7.79 (0.10)	0.197 (0.017)	0.058 (0.007)	0.129 (0.009)	0.277 (0.016)	7.78 (0.14)
E2X	0.332 (0.011)	0.096 (0.004)	0.358 (0.013)	0.097 (0.007)	4.83 (0.06)	0.286 (0.018)	0.103 (0.008)	0.254 (0.015)	0.088 (0.008)	4.62 (0.06)
D-RISE	0.163 (0.006)	0.133 (0.006)	**0.688*** (0.009)	**0.047*** (0.003)	0.02 (0.00)	0.160 (0.012)	0.141 (0.010)	**0.565*** (0.014)	**0.039*** (0.003)	0.02 (0.00)
KernelSHAP	0.261 (0.008)	0.122 (0.005)	0.582 (0.011)	0.062 (0.003)	0.01 (0.00)	0.236 (0.013)	0.128 (0.009)	0.471 (0.015)	0.054 (0.004)	0.01 (0.00)
FSOD (Ours)	**0.791*** (0.008)	**0.501*** (0.007)	0.614 (0.010)	0.056 (0.003)	**61.85*** (0.35)	**0.571** (0.017)	**0.261** (0.013)	0.519 (0.014)	0.045 (0.003)	**67.37*** (0.69)

To reduce computational noise and refine the sampling patterns, the image is divided into superpixels ($h \times w$ pixels) during the sampling of $\mathbf{s}$.

3) Calculate the detection similarity $f_c(\boldsymbol{M}_\mathbf{s} \odot \boldsymbol{x}, \boldsymbol{d}_t)$ using a masked image and determine the loss function of $\mathcal{L}(\theta)$ of Eq. 8.
4) Update the explainer model's parameter θ based on feedback from the loss function and repeat these steps until the learning converges.

IV. EXPERIMENTS AND RESULTS

To assess our proposed method's performance, we conducted a quantitative evaluation, focusing on its explanatory accuracy and processing speed in comparison to existing methods. The existing methods, introduced in the experiment depicted in Fig. 1, were chosen from various categories of feature attribution calculation for comparison. For evaluation, we used the validation split of COCO [29] and VOC [30] datasets, commonly utilized in the benchmark evaluations of object detection tasks. Explanation targets were based on object detections obtained from YOLOv5s, specifically focusing on true positive detections. The results are derived from a random selection of 10% of the images in the datasets. To optimize the object detector's performance, the image size was standardized at 640 × 640 pixels. Additionally, the parameters and training conditions of the explainer model used in the evaluation were set as follows.

Model parameters. The feature maps, derived from YOLOv5s' backbone network and used as input for the explainer model, consist of three layers with dimensions 80 × 80, 40 × 40, and 20 × 20. Each of these layers has 255 channels. The number of output classes for the COCO dataset is 80 and that for the VOC dataset is 20.

Training conditions. During training, we used the Adam optimization algorithm [34] with an initial learning rate set to 10^{-4}. The training spanned 100 epochs, incorporating early stopping to halt training upon signs of learning saturation.

A. EVALUATION METRICS

Given the absence of established optimal evaluation metrics for assessing explanation accuracy, this study employs metrics commonly used in existing research.

1) ENERGY-BASED POINTING GAME (EPG)

EPG [13] quantifies how precisely feature attributions highlight the target object. This metric computes the ratio of the sum of the feature attributions within the target object to the sum of all feature attributions. We used an object segmentation mask for the ground truth to identify the target object regions instead of using a bounding box. This approach helps distinguish cases where some methods exhibit strong responses to non-target objects within a bounding box containing multiple objects.

2) VISUAL EXPLANATION ACCURACY (VEA)

VEA [35] measures the degree of overlap between the high-importance area and the target object by calculating IoU between a ground truth object segmentation mask and a saliency map thresholded at various levels. The IoU is plotted with IoU on the vertical axis and thresholds on the horizontal axis, using the area under the curve (AUC) as the evaluation metric.

3) DELETION AND INSERTION

Deletion and Insertion metrics [36] quantify the impact of important pixels on the model's prediction. For the Deletion metric, pixels with higher attributions are progressively removed from the input image, and the decrease in the model's output score is assessed. The Insertion metric, on the other hand, involves adding pixels to the baseline image in the same order and assessing the increase in the model's output score. In both metrics, the variation of the score is plotted as a function of the number of added or removed pixels, using the AUC as the metric for evaluation.

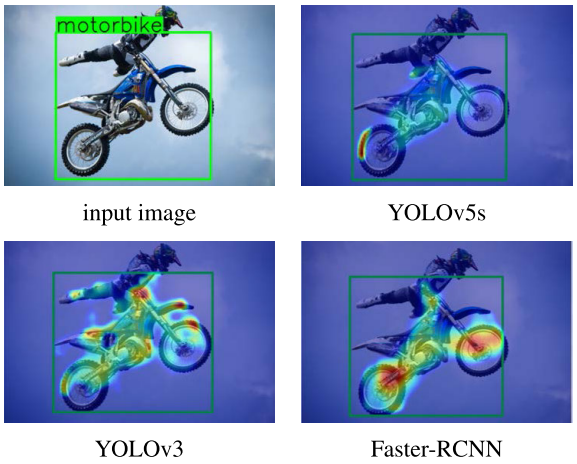

FIGURE 4. Comparison results of saliency maps for object detections obtained by various object detectors.

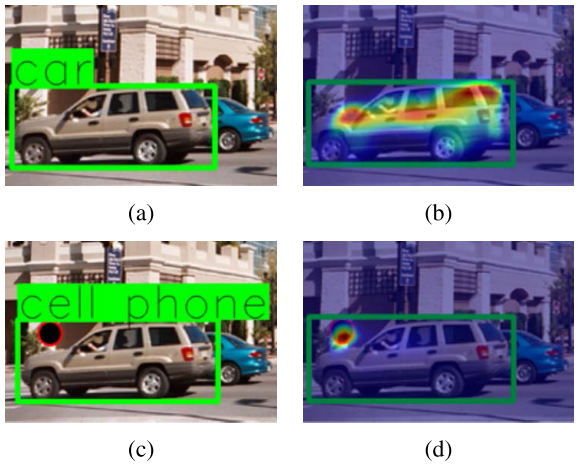

FIGURE 5. Car detection result (a) and its saliency map (b). The object detector is trained to detect a car mislabelled as a cell phone when a black circle is drawn at the top left of the bounding box, as shown in (c). Its saliency map (d) accurately highlights the circle as a clue for detection.

B. EVALUATION RESULTS

The evaluation results of each metric and the processing speed for generating a saliency map are presented in Table 1. The values represent the average results for each explanation target, with parenthetical values indicating standard errors. Using these values and considering a 95% confidence interval, we determine the best possible method. The results highlight the best candidate method in bold, while the second-best candidate is underscored. Deletion and Insertion metrics yield higher scores when the impact of important regions on the detection results is substantial. While these metrics do not consider the flow of feature attributions to other objects, as observed in CRP in Fig. 1, VEA and EPG can distinguish whether the feature attributions target the specified object, and our method yielded the most superior results with these methods. Because no established consensus exists on which evaluation metric is the most crucial, identifying the best-performing method remains a challenge. While no method excels in all metrics, our proposed method consistently achieves either the top or second-best results across all indicators. Notably, it is the fastest among the methods compared. These results demonstrate that our method is superior to other methods in terms of offering both high explanatory accuracy and rapid processing speed.

C. SANITY CHECK

We conducted a sanity check to confirm that our method functions as intended. First, we demonstrated that our method can be applied in a model-agnostic manner to other CNN-based detectors. CNN-based detectors can be categorized into two types: one-stage detectors, such as YOLOv5 and YOLOv3 [21], which concurrently process the object classification and localization, and two-stage detectors, such as Faster-RCNN [37], which first generate region proposals, and subsequently perform classification within these areas. Fig. 4 shows the results when the method is applied to Faster-RCNN [37] and YOLOv3 [21]. Because both detectors exhibit higher detection accuracy compared to YOLOv5s, their resultant saliency maps are more distinct. Additionally, we verified that the saliency maps accurately capture the object detection cues. We introduced a bias into YOLOv5s during training: specifically, if a black circle appears in the upper-left region of a car object, it is classified as a cell phone. Fig. 5 shows that the saliency map appropriately emphasizes the black circle as the basis for its decision. Lastly, to demonstrate the effectiveness of our method, an ablation study was conducted. The transition from image classification to object detection involves the use of feature maps as inputs and the introduction of query maps to specify the target object. The impact of the presence of these two components on explanatory accuracy is examined. The results are presented in Table 2, using the same metrics as in Table 1. Given that each element enhances explanatory accuracy, the combination of these components validates the effectiveness of our proposed method.

TABLE 2. Ablation study based on the input of feature maps and the utilization of query maps as conditions.

Feature map	Query map	EPG(↑)	VEA(↑)	Ins.(↑)	Del.(↓)
✓		0.374	0.135	0.488	0.080
	✓	0.736	0.384	0.518	0.079
✓	✓	**0.791**	**0.501**	**0.614**	**0.056**

V. CONCLUSION

Achieving a balance between processing speed and high explanatory accuracy in XAI for image classification and object detection tasks poses a substantial challenge for practical application. While the existing method tackles this challenge in image classification by rapidly estimating Shapley values with an explanation model, extending this approach to object detection involves several issues. In this study, we explicitly identified and addressed these challenges and presented a novel method for object detection that employs an explanation model. Our qualitative evaluation demonstrated that our method is model-agnostic and effectively captures the cues leading to object detection. Additionally, quantitative evaluation demonstrated that our method is the fastest among existing methods and maintains

a high level of explanatory accuracy. Achieving a balanced approach between explanatory accuracy and computational cost is expected to broaden the applicability of XAI, bringing it closer to practical application.

REFERENCES

[1] J. Adebayo, J. Gilmer, M. Muelly, I. Goodfellow, M. Hardt, and B. Kim, "Sanity checks for saliency maps," in *Proc. NeurIPS*, 2018, pp. 9525–9536.
[2] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic attribution for deep networks," in *Proc. ICML*, 2017, pp. 3319–3328.
[3] A. Khakzar, P. Khorsandi, R. Nobahari, and N. Navab, "Do explanations explain? Model knows best," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 10234–10243.
[4] L. S. Shapley, "A value for n-person games," in *Contributions to Theory Games*, vol. 2, 1953, pp. 307–317.
[5] T. Yamauchi and M. Ishikawa, "Spatial sensitive GRAD-CAM: Visual explanations for object detection by incorporating spatial sensitivity," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2022, pp. 256–260.
[6] V. Petsiuk, R. Jain, V. Manjunatha, V. I. Morariu, A. Mehra, V. Ordonez, and K. Saenko, "Black-box explanation of object detectors via saliency maps," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 11438–11447.
[7] N. Jethani, M. Sudarshan, I. C. Covert, S.-I. Lee, and R. Ranganath, "FastSHAP: Real-time Shapley value estimation," in *Proc. ICLR*, 2021, pp. 1–23.
[8] D. Smilkov, N. Thorat, B. Kim, F. Viégas, and M. Wattenberg, "SmoothGrad: Removing noise by adding noise," 2017, *arXiv:1706.03825*.
[9] J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller, "Striving for simplicity: The all convolutional net," 2014, *arXiv:1412.6806*.
[10] S. Bach, A. Binder, G. Montavon, F. Klauschen, K.-R. Müller, and W. Samek, "On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation," *PLoS ONE*, vol. 10, no. 7, 2015, Art. no. e0130140.
[11] J. Gu, Y. Yang, and V. Tresp, "Understanding individual decisions of CNNs via contrastive backpropagation," in *Proc. ACCV*, 2019, pp. 119–134.
[12] A. Chattopadhay, A. Sarkar, P. Howlader, and V. N. Balasubramanian, "Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks," in *Proc. WACV*, 2018, pp. 839–847.
[13] H. Wang, Z. Wang, M. Du, F. Yang, Z. Zhang, S. Ding, P. Mardziel, and X. Hu, "Score-CAM: Score-weighted visual explanations for convolutional neural networks," in *Proc. CVPRW*, 2020, pp. 24–25.
[14] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proc. CVPR*, 2016, pp. 2921–2929.
[15] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 618–626.
[16] V. Petsiuk, A. Das, and K. Saenko, "RISE: Randomized input sampling for explanation of black-box models," in *Proc. BMVC*, 2018, pp. 1–17.
[17] R. Fong, M. Patrick, and A. Vedaldi, "Understanding deep networks via extremal perturbations and smooth masks," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 2950–2958.
[18] M. T. Ribeiro, S. Singh, and C. Guestrin, "Why should i trust you?': Explaining the predictions of any classifier," in *Proc. SIGKDD*, 2016, pp. 1135–1144.
[19] A. Karasmanoglou, M. Antonakakis, and M. Zervakis, "Heatmap-based explanation of YOLOv5 object detection with layer-wise relevance propagation," in *Proc. IEEE Int. Conf. Imag. Syst. Techn. (IST)*, Jun. 2022, pp. 1–6.
[20] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.
[21] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*.
[22] D. Gudovskiy, A. Hodgkinson, T. Yamaguchi, Y. Ishii, and S. Tsukizawa, "Explain to fix: A framework to interpret and correct DNN object detector predictions," 2018, *arXiv:1811.08011*.
[23] M. Sundararajan and A. Najmi, "The many Shapley values for model explanation," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 9269–9278.
[24] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Proc. NeurIPS*, 2017, pp. 4765–4774.
[25] S. M. Lundberg, G. G. Erion, and S.-I. Lee, "Consistent individualized feature attribution for tree ensembles," 2018, *arXiv:1802.03888*.
[26] Q. Zheng, Z. Wang, J. Zhou, and J. Lu, "Shap-CAM: Visual explanations for convolutional neural networks based on Shapley value," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2022, pp. 459–474.
[27] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.
[28] M. Hogan, N. Aouf, P. Spencer, and J. Almond, "Explainable object detection for uncrewed aerial vehicles using KernelSHAP," in *Proc. IEEE Int. Conf. Auto. Robot Syst. Competitions (ICARSC)*, Apr. 2022, pp. 136–141.
[29] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *Proc. CVPR*, 2014, pp. 740–755.
[30] M. Everingham, S. M. A. Eslami, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal visual object classes challenge: A retrospective," *Int. J. Comput. Vis.*, vol. 111, no. 1, pp. 98–136, Jan. 2015.
[31] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," Master's thesis, Dept. Comput. Science, Univ. Toronto, Toronto, ON, Canada, 2009.
[32] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 936–944.
[33] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, vol. 9351, 2015, pp. 234–241.
[34] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.
[35] J. Oramas, K. Wang, and T. Tuytelaars, "Visual explanation by interpretation: Improving visual feedback capabilities of deep neural networks," in *Proc. ICLR*, 2019, pp. 1–10.
[36] R. C. Fong and A. Vedaldi, "Interpretable explanations of black boxes by meaningful perturbation," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 3449–3457.
[37] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. NeurIPS*, 2015, pp. 91–99.

MICHIHIRO KUROKI (Member, IEEE) received the B.E. degree in electrical engineering from Keio University, Japan, in 2013, and the M.E. degree from The University of Tokyo, Japan, in 2015, where he is currently pursuing the Ph.D. degree. Since 2015, he has been an Autonomous Driving Development Engineer with a company. His main research interests include explainable AI for computer vision fields, such as object detection in images and point clouds.

TOSHIHIKO YAMASAKI (Member, IEEE) received the B.S., M.S., and Ph.D. degrees from The University of Tokyo. From February 2011 to February 2013, he was a JSPS Fellow for Research Abroad and a Visiting Scientist with Cornell University. He is currently a Professor with the Department of Information and Communication Engineering, Graduate School of Information Science and Technology, The University of Tokyo. His current research interests include attractiveness computing based on multimedia big data analysis and fundamental problems in computer vision and multimedia. He is a member of ACM, AAAI, IEICE, IPSJ, and ITE.