Enhancing the Fairness and Performance of Edge Cameras with Explainable AI

Truong Thanh Hung Nguyen[*†‡], Vo Thanh Khang Nguyen[*‡], Quoc Hung Cao[*‡],
Van Binh Truong[‡], Quoc Khanh Nguyen[‡], Hung Cao[†]
[†]Analytics Everywhere Lab, University of New Brunswick, Canada
[‡]Quy Nhon AI, FPT Software, Vietnam
Email: {hungntt, khangnvt1, hungcq3, binhtv8, khanhnq33}@fpt.com, hcao3@unb.ca

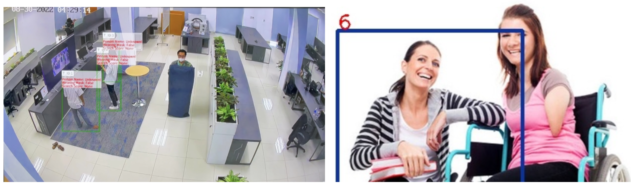
***Abstract*—The escalating integration of Artificial Intelligence (AI), particularly in human detection models on camera systems at the Edge, has led to the proliferation of highly accurate, yet complex AI architectures. These complexities pose substantial challenges in interpreting predictions and debugging. This research introduces a diagnostic framework using Explainable AI (XAI) for model debugging, involving expert-led problem identification and solution development based on diagnostic outcomes. We validate this framework through experiments on the Bytetrack model and its real-world application in an office camera system at the Edge network. Our findings highlight the training dataset as the primary source of model bias, proposing a solution through model augmentation. This framework aids in pinpointing model biases, a crucial step toward establishing fair, transparent, and unbiased models, thereby bolstering trust and confidence.**

***Index Terms*—Explainable AI, Edge Camera, Human Detection**

I. Introduction

Human detection via security cameras, a critical AI task, involves deploying an AI model for various alerts, including fall detection and intrusion warnings. YOLO, with its variant YOLOX, serves as a leading model for human detection, with Bytetrack, a YOLOX-based model, excelling in multi-object tracking by associating all detection boxes [1]–[3]. However, our experiments reveal Bytetrack's susceptibility to abnormal human detection cases, such as obscured bodies (Fig. 1a) and physically disabled individuals (Fig. 1b). The black-box nature of these models complicates bug identification, necessitating advanced debugging and improvement techniques [4]. While XAI has been used to debug models in tabular data and text data [5], [6], its application in image data, particularly human detection, remains limited.

This paper, therefore, introduces an XAI-supported debugging framework for human detection models on security cameras. The framework involves domain experts for problem identification and solution suggestion based on diagnostic outcomes, with potential applicability to object detection and classification problems.



(a) (b)

Fig. 1. (a) A security camera on the ceiling of an office can detect ordinary people (green boxes), but not people who cover their bodies with a cloth. (b) The Bytetrack model cannot detect the disabled woman but still detect the other, who is not disabled.

II. Related Work

A. Human Detection

Human detection, the process of identifying human presence in images, videos, or security camera footage, has been addressed through various techniques. The advent of Deep Learning (DL) introduced innovative models, notably Faster R-CNN [7] and YOLO [1], which effectively mitigated challenges related to object size, varying illumination conditions, and real-time computational constraints. Building on the impressive object detection results of YOLOX [2], Bytetrack [3] was developed to focus on human detection, utilizing YOLOX for detection and Byte for post-processing.

B. Explainable AI

The integration of AI into real-life applications has spurred the development of numerous XAI methods, broadly categorized into perturbation-based, backpropagation-based, and example-based approaches.

Perturbation-based methods, independent of model architecture, generate perturbed input images by masking pixels or superpixel regions, followed by prediction analysis to determine the influence of each pixel or superpixel on the model's prediction. Despite their broad applicability, their computational complexity can be prohibitive. Notable methods include LIME [8], D-RISE [9], D-CLOSE [10].

Backpropagation-based methods access the model architecture to derive and analyze information for explanations. Prominent methods include Class Activation Mapping (CAM) [11], GradCAM [12], SeCAM [13], and ScoreCAM [14].

[*]Corresponding author

Example-based methods, such as Influence Function [15] and ExMatchina [16], provide explanations using examples from the training dataset, analyzing the positive and negative impacts on input image prediction.

Applying XAI to object detection is more challenging due to model complexity compared to classification models. However, several XAI methods, including D-RISE [9], D-CLOSE [10], SODEx [17], and G-CAME [18], have been adapted from classification methods for object detection models.

C. Debugging Model Framework with XAI

While numerous studies have employed a variety of XAI methods [19], [20], most merely address the question, *"Why does the model make this prediction?"* A subsequent question, *"How can we enhance model performance based on the explanation?"* [21], necessitates a strategy for applying XAI to improve the AI system. The DARPA framework, used in the military sector, addresses this question by enhancing user trust in model decision-making [22]. However, to date, no study has proposed a framework for debugging human detection models. Therefore, this paper presents a debugging framework for such models, utilizing XAI as a diagnostic tool to identify problems and enhance model fairness and performance.

III. Methodology

We introduce a systematic debugging model framework, depicted in Fig. 2, consisting of seven interlinked stages, each dependent on the preceding stage's outcomes. In instances where stages allow for multiple methods or assumptions, we provide guidelines for appropriate strategy selection. Within this framework, XAI serves as a tool assisting specialists in pinpointing the model's root problem and facilitating the proposal of solutions to enhance model performance.

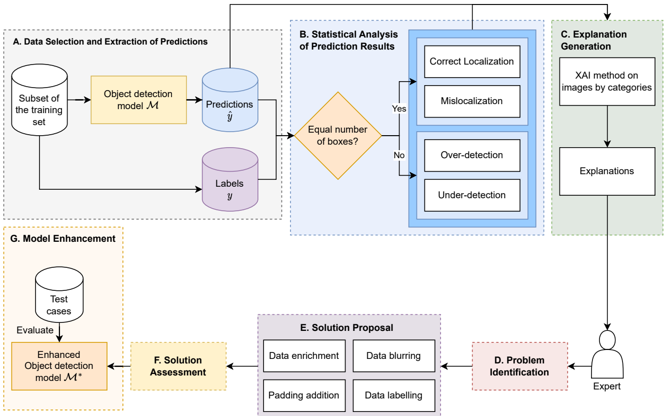

A. Data Selection and Extraction of Predictions
Subset of the training set
Object detection model $\mathcal{M}$
Predictions $\hat{y}$
Labels y
B. Statistical Analysis of Prediction Results
Equal number of boxes?
Yes
No
Correct Localization
Mislocalization
Over-detection
Under-detection
C. Explanation Generation
XAI method on images by categories
Explanations
Expert
D. Problem Identification
E. Solution Proposal
Data enrichment
Data blurring
Padding addition
Data labelling
F. Solution Assessment
G. Model Enhancement
Test cases
Evaluate
Enhanced Object detection model $\mathcal{M}^*$

Fig. 2. The Debugging Framework for Human Detection Models

A. Data Selection and Extraction of Predictions

Our proposed framework begins with the selection of a training dataset subset to enhance the model, enabling model verification and addressing potential training set issues. It is important to note that public datasets like CrowdHuman [23], part of Bytetrack's training data, are susceptible to data poisoning [24], compromising data integrity and model performance.

Efficient error detection in the model or training data, with minimal resource expenditure, is achieved through random testing [25], [26]. This technique involves random data subset selection for testing, identifying major errors and inconsistencies without exhaustive dataset testing.

Adhering to the principle that smaller sample sizes can yield reliable results, we employ a heuristic from statistical sampling to determine an appropriate sample size relative to the total population. The maximum sample size should not exceed 10% of the total dataset or a limit of 1000, ensuring a representative, robust, and efficient sample [27], [28].

Upon data subset selection, it is input into the model for prediction generation. These predictions are analyzed, compared with ground truth, and evaluated for model performance, offering insights into model accuracy, precision, reliability, and potential improvement areas.

B. Statistical Analysis of Prediction Results

Upon obtaining the model's predictions, they are systematically categorized based on comparison with ground-truth data. This classification is problem-specific and adjudicated by field experts. In the context of this study, focusing on human detection, data is divided into four categories.

The initial dataset split depends on whether the model's predicted box count matches the ground truth. If the model detects fewer people, the image is labeled "Under-detection"; if more, it is "Over-detection."

When the model's box count aligns with the ground truth, detection quality is further assessed. Each model-detected box is compared with its corresponding ground truth box using Intersection over Union (IoU) values. Images with all box pairs having IoU ≥ 0.5 are labeled "Correct Localization"; otherwise, they are "Mislocalization".

This step classifies the selected dataset into groups based on prediction outcomes. Three categories—"Under-detection," "Over-detection," and "Mislocalization"—indicate areas for model performance improvement. The subsequent step involves a detailed error source analysis, forming the basis for strategies to enhance model accuracy in correct people count detection within an image.

C. Explanation Generation

In this step, we apply XAI methods to get the explanation for each category of images. Since D-RISE [9], a novel method for object detection can be used for many different types of models because it does not require access to the model's architecture and D-RISE allows us to get an explanation for the ground truth box, which helps us compare it with the box detected by the model, we employ D-RISE for human detection models. The explanations can help experts diagnose the cause of the wrong prediction in the next phase.

D. Problem Identification

Based on the XAI results obtained in the previous phase, experts will be engaged in analyzing each specific category divided by the statistical analysis (Sec. III-B). The XAI results provide which regions are being focused by the model on the input image. Experts scrutinize these regions to determine their significance and detect potential biases. A comparison of these regions among images within the same category is conducted to reveal their commonalities, which then be checked against the remaining categories for any shared traits. Furthermore, we compare the XAI outcomes across different models to aid in the identification and diagnosis of potential issues.

E. Solution Proposal

Solution proposal is a crucial phase in determining how to elevate the model performance. After determining the problem, the expert analyzes and checks the dataset and model to find out the possible causes, which can come from the distribution of data, labels, bias, or even model architecture. [6] also suggested some other potential errors, such as natural artifacts, limited training subsets, incorrect label injection, and out-of-distribution tests. Sequentially, several approaches, including adjusting model parameters, improving training data, and augmenting the training process, can be applied.

F. Solution Assessment

Rather than implementing all possible solutions, we shall assess the feasibility of proposed solutions on a small dataset initially. We evaluate the advantages and disadvantages of each solution, drawing from prior case studies to assess their relevance to the present problem. The infeasible solutions can be identified and eliminated, thereby allowing for the selection of the most suitable solution.

G. Model Enhancement

Finally, employing the efficacious solution from the previous step, the model is fine-tuned to produce an improved version that fixes the problem identified in Sec. III-D. Then, we evaluate the model's improvement through a comparative analysis of the model's performance before and after fine-tuning, which can be conducted by comparing the model's predictive statistics with the selected images in the initial phase. Furthermore, additional testing may be carried out using anomalous instances that the original model failed to predict, with the aim of ascertaining the efficacy of the model's improvements in addressing the identified problem.

IV. Experiment

In our experiment, we clarify each step according to the process as described in Fig. 2. We utilize the Bytetrack model pre-trained on MOT17 [29], Cityperson [30], ETHZ [31], and CrowdHuman [23] dataset for our experiment.

A. Data Selection and Extraction of Predictions

The training dataset comprises four distinct public datasets, as previously delineated. Among these, MOT17, Cityperson, and ETHZ consist of image frames extracted from videos, while CrowdHuman, a benchmark dataset for evaluating object detectors in crowd scenarios, comprises publicly sourced images offering a diverse range of contextual backgrounds [23], [29]–[31]. Notably, the images within CrowdHuman are independent and not restricted to extraction from the same video.

Given these dataset characteristics, we opt to employ CrowdHuman in our experiment, partitioned into 15000, 4370, and 5000 images for training, validation, and testing, respectively. The training and validation sets collectively contain 470K human instances, each annotated with a head bounding box, human visible-region bounding box, and human full-body bounding box. We randomly select 1000 images from the CrowdHuman training dataset as a subset for model prediction extraction, as detailed in Sec. III-A.

B. Statistical Analysis of Prediction Results

In this phase, the predicted boxes from the previous step are compared to the ground truth. We statisticize the model's prediction result as in Table I, in which the "Under-detection" accounts for the highest percentage among these four cases, with 85.5%.

TABLE I
THE CATEGORIES OF 1000 IMAGES IN THE SUBSET.

Case	Number of images
Under-detection	855
Over-detection	17
Correct Localization	108
Mislocalization	20

C. Explanation Generation

The Bytetrack model consists of two components: the YOLOX model for individual detection and the Byte stage for processing detected boxes. Specifically, YOLOX plays a crucial role in box detection as subsequent processing is dependent on this stage. The Byte processing stage aims to retain low prediction score boxes that may be obscured by other objects [3]. Consequently, we employ D-RISE to extract explanations for YOLOX, using the final box coordinates predicted by the Bytetrack model [32]. Furthermore, we apply D-RISE to the YOLOX weight with Bytetrack's output boxes to discern differences between Bytetrack and YOLOX, pre-trained on COCO 2017, as depicted in Fig. 3 [2], [33].

D. Problem Identification

As demonstrated by the XAI explanations in Fig. 3, the Bytetrack model primarily detects the human body, revealing its inability to detect partially visible individuals, where only their heads are visible. This limitation is further explored through experiments using images of wheelchair-bound individuals with obscured body parts, leading to the model's

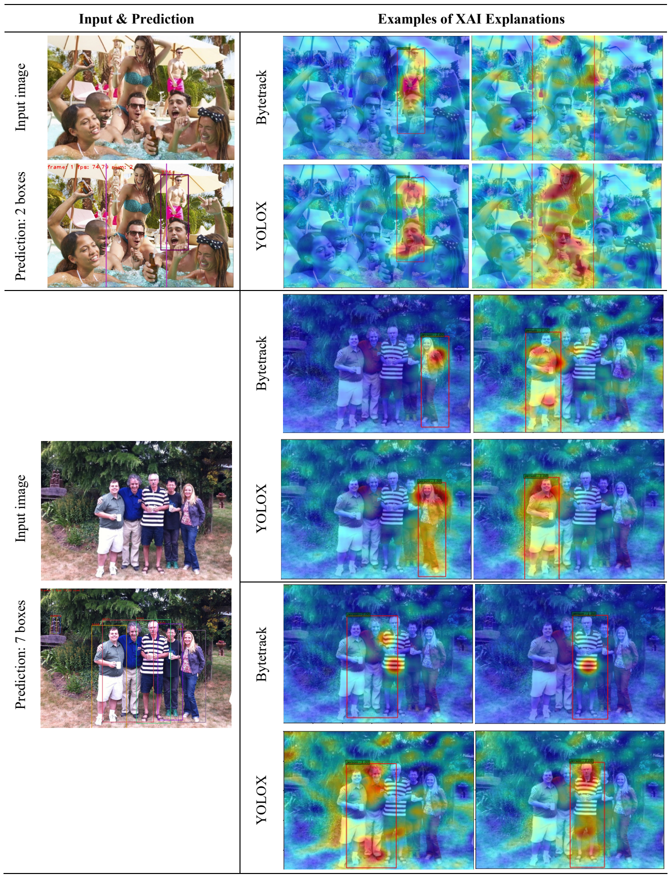


Fig. 3. Examples of XAI Explanations with Bytetrack and YOLOX model. In which, each image in the second column is the XAI Explanations for a corresponding box.

failure to detect a person, as shown in Fig. 1b. Similar failures occur when individuals are obscured by objects, as depicted in Fig. 1a. Thus, the model's incapacity to identify physically concealed humans is identified as a problem warranting further investigation and solutions.

E. Solution Proposal

Based on the problem identified, we propose the following assumptions:

- Dataset: The average image contains 23 individuals. Given the high object count per image, the head region appears smaller than the body region, potentially inducing a body bias. Additionally, we hypothesize about labeling, where ground truth box coordinates are outside the image, as exemplified in the first row of Fig. 3 and Table II.
- Model: Bytetrack attempts to resolve occluded object issues [3]. For images containing only the head, Bytetrack seeks a location containing the body.

Based on these assumptions, we propose potential solutions:

- Data enrichment: Incorporate additional training images where a significant portion of the body region is obscured, such as portrait photos or images of people in classrooms or at work.

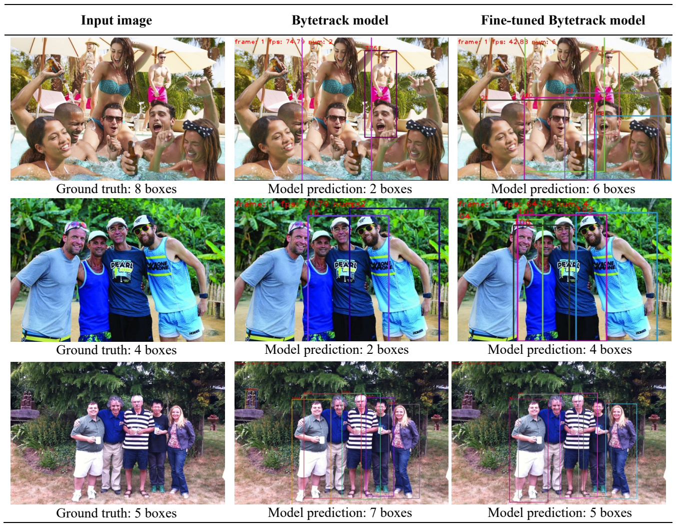

Input image
Bytetrack model
Fine-tuned Bytetrack model
Ground truth: 8 boxes
Model prediction: 2 boxes
Model prediction: 6 boxes
Ground truth: 4 boxes
Model prediction: 2 boxes
Model prediction: 4 boxes
Ground truth: 5 boxes
Model prediction: 7 boxes
Model prediction: 5 boxes

Fig. 4. Predictions of the Bytetrack model before and after fine-tuning.

TABLE II
GROUND TRUTH BOXES' COORDINATE OF THE INPUT IMAGE IN THE FIRST ROW OF FIG. 3, WHERE 7/8 BOXES ARE OUTSIDE THE IMAGE.

Left	Top	Right	Bottom	Outside image
-50	35	531	131	×
-12	87	451	1325	×
308	292	635	1228	×
499	171	988	1201	×
618	370	1034	1243	×
608	61	758	444	
318	-14	673	745	×
303	-3	444	437	×

- Data blurring: Blur the human body in the image based on XAI results, allowing the model to focus more on the head area [34].
- Padding: Add padding to the image during preprocessing so that box coordinates are always within the image.
- Relabeling: Adjust by clipping the outside box coordinates to ensure they are solely within the image.

F. Solution Assessment

We conduct a comprehensive analysis to identify and implement the most suitable solution to the problem. Each solution is evaluated as follows:

- Data enrichment: Upon dataset review, our analysis indicates that additional data incorporation would not yield significant improvements due to the presence of partially obscured images in the current dataset.
- Data blurring: While blurring has proven effective in image classification problems [34], its application to the human detection problem, where the model predicts only one class (human), is not deemed appropriate.
- Padding: Padding is added to the sides of "Under-detection" images to examine if the model can detect humans not bound by the input image frame. While this approach yields improvements in some cases (Fig. 5), the

model still fails to detect individuals obscured by objects (Fig. 1).

- Relabeling: Given the dataset's box coordinates are outside the image, differing from the COCO dataset, and considering the divergent learned features of the models (Fig. 3), relabeling emerges as a potential and effective solution.

Following this analysis, relabeling is identified as the most efficient solution.

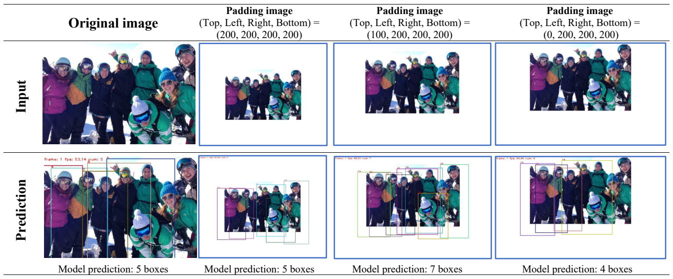

Original image
Padding image (Top, Left, Right, Bottom) = (200, 200, 200, 200)
Padding image (Top, Left, Right, Bottom) = (100, 200, 200, 200)
Padding image (Top, Left, Right, Bottom) = (0, 200, 200, 200)
Input
Prediction
Model prediction: 5 boxes
Model prediction: 5 boxes
Model prediction: 7 boxes
Model prediction: 4 boxes

Fig. 5. Example of padding result. (Top, Left, Right, Bottom) = (100, 200, 200, 200) signifies padding of 100, 200, 200, and 200 pixels respectively on the top, left, right, and bottom.

G. Model and Dataset Enhancement

The CrowdHuman dataset is reannotated by constraining bounding box coordinates within the image dimensions, as delineated by the following equations:

$$x^{'}_{\text{top, left}} = \mathtt{max}(0, x_{\text{top, left}}) \tag{1}$$

$$y^{'}_{\text{top, left}} = \mathtt{max}(0, y_{\text{top, left}}) \tag{2}$$

$$x^{'}_{\text{bottom, right}} = \mathtt{min}(w, x_{\text{bottom, right}}) \tag{3}$$

$$y^{'}_{\text{bottom, right}} = \mathtt{min}(h, y_{\text{bottom, right}}) \tag{4}$$

Here, w, h represents the image's width and height, respectively. The coordinates $(x^{'}_{\text{top, left}}, y^{'}_{\text{top, left}})$ and $(x^{'}_{\text{bottom, right}}, y^{'}_{\text{bottom, right}})$ denote the adjusted top-left and bottom-right points, respectively.

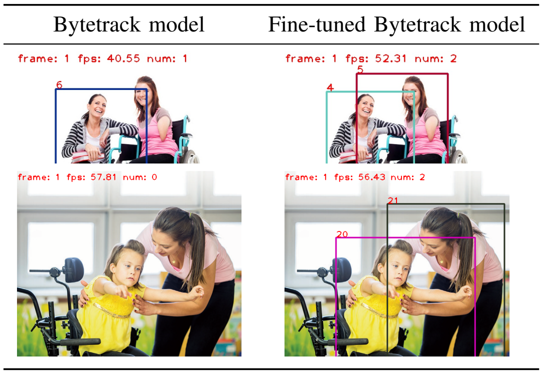

Bytetrack model
Fine-tuned Bytetrack model
frame: 1 fps: 40.55 num: 1
6
frame: 1 fps: 52.31 num: 2
5
4
frame: 1 fps: 57.81 num: 0
frame: 1 fps: 56.43 num: 2
21
20

Fig. 6. Model's prediction on physically disabled person images. After fine-tuning, the model performs better than the original pre-trained model.

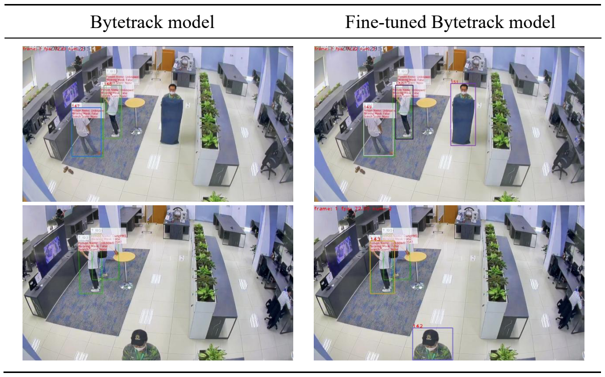


Fig. 7. Model's prediction on a security camera. The fine-tuned model performs better than the original pre-trained model detecting covered people.

Subsequent model refinement occurs over 10 epochs, with performance enhancement evaluated in three scenarios:

- Selected training dataset: A subset of 1000 images is tested post-refinement, with quantitative and qualitative comparisons made to the pre-refinement model in Table III and Fig. 4. Notably, the refined model improves correct localization on 855 "Under-detection" images by 21 instances.
- Physically disabled individuals: The refined model exhibits enhanced detection capabilities on images of physically disabled individuals, as depicted in Fig. 6.
- Partially obscured individuals in security footage: The model's performance in real-world scenarios, such as office surveillance footage where individuals may be partially obscured, is tested and shown to improve post-refinement, as illustrated in Fig. 7.

TABLE III
STATISTICAL RESULT PRE-TRAINED MODEL VERSUS FINE-TUNED MODEL. THE ARROW ↑/↓ INDICATES THE HIGHER/LOWER VALUE, THE BETTER. THE BOLD INDICATES THE BETTER RESULT.

Case	Pre-trained model	Fine-tuned model
Under-detection (↓)	855	**834**
Over-detection (↓)	17	**13**
Correct Localization (↑)	108	**133**
Mislocalization (↓)	20	20

V. CONCLUSION AND FUTURE WORK

This paper proposes a debugging model framework with XAI for the human detection problem. Based on the XAI explanations, the experts can identify problems and propose solutions to improve the model and the dataset. In our experiment, the problem leading to the unfairness and underperformance of the Bytetrack model, where it cannot detect the person is partially obscured, is in data labeling, showing that the label can make the model biased. Our framework can be extended to other object detection problems that require focused consideration of specific classes. Moving forward, we plan to generalize further and extend this methodology to address a broader range of problems effectively.

References

[1] J. Redmon, S. K. Divvala, R. B. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," *CoRR*, vol. abs/1506.02640, 2015. [Online]. Available: http://arxiv.org/abs/1506.02640

[2] Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun, "YOLOX: exceeding YOLO series in 2021," *CoRR*, vol. abs/2107.08430, 2021. [Online]. Available: https://arxiv.org/abs/2107.08430

[3] Y. Zhang, P. Sun, Y. Jiang, D. Yu, Z. Yuan, P. Luo, W. Liu, and X. Wang, "Bytetrack: Multi-object tracking by associating every detection box," *CoRR*, vol. abs/2110.06864, 2021. [Online]. Available: https://arxiv.org/abs/2110.06864

[4] P. Rasouli and I. C. Yu, "Explainable debugger for black-box machine learning models," in *2021 International Joint Conference on Neural Networks (IJCNN)*, 2021, pp. 1–10.

[5] R. Yousefzadeh and D. P. O'Leary, "Auditing and debugging deep learning models via decision boundaries: Individual-level and group-level analysis," *CoRR*, vol. abs/2001.00682, 2020. [Online]. Available: http://arxiv.org/abs/2001.00682

[6] P. Lertvittayakumjorn and F. Toni, "Explanation-based human debugging of NLP models: A survey," *CoRR*, vol. abs/2104.15135, 2021. [Online]. Available: https://arxiv.org/abs/2104.15135

[7] S. Ren, K. He, R. B. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," *CoRR*, vol. abs/1506.01497, 2015. [Online]. Available: http://arxiv.org/abs/1506.01497

[8] M. T. Ribeiro, S. Singh, and C. Guestrin, ""why should I trust you?": Explaining the predictions of any classifier," *CoRR*, vol. abs/1602.04938, 2016. [Online]. Available: http://arxiv.org/abs/1602.04938

[9] 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," *CoRR*, vol. abs/2006.03204, 2020. [Online]. Available: https://arxiv.org/abs/2006.03204

[10] V. B. Truong, T. T. H. Nguyen, V. T. K. Nguyen, Q. K. Nguyen, and Q. H. Cao, "Towards better explanations for object detection," *arXiv preprint arXiv:2306.02744*, 2023.

[11] B. Zhou, A. Khosla, À. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," *CoRR*, vol. abs/1512.04150, 2015. [Online]. Available: http://arxiv.org/abs/1512.04150

[12] R. R. Selvaraju, A. Das, R. Vedantam, M. Cogswell, D. Parikh, and D. Batra, "Grad-cam: Why did you say that? visual explanations from deep networks via gradient-based localization," *CoRR*, vol. abs/1610.02391, 2016. [Online]. Available: http://arxiv.org/abs/1610.02391

[13] P. Nguyen, H. CAO, K. NGUYEN, H. NGUYEN, and T. YAIRI, "Secam: Tightly accelerate the image explanation via region-based segmentation," *IEICE Transactions on Information and Systems*, vol. E105.D, pp. 1401–1417, 08 2022.

[14] H. Wang, M. Du, F. Yang, and Z. Zhang, "Score-cam: Improved visual explanations via score-weighted class activation mapping," *CoRR*, vol. abs/1910.01279, 2019. [Online]. Available: http://arxiv.org/abs/1910.01279

[15] P. W. Koh and P. Liang, "Understanding black-box predictions via influence functions," 2017. [Online]. Available: https://arxiv.org/abs/1703.04730

[16] J. V. Jeyakumar, J. Noor, Y.-H. Cheng, L. Garcia, and M. Srivastava, "How can i explain this to you? an empirical study of deep neural network explanation methods," *Advances in Neural Information Processing Systems*, vol. 33, 2020.

[17] J. H. Sejr, P. Schneider-Kamp, and N. Ayoub, "Surrogate object detection explainer (sodex) with yolov4 and lime," *Mach. Learn. Knowl. Extr.*, vol. 3, pp. 662–671, 2021.

[18] Q. K. Nguyen, T. T. H. Nguyen, V. T. K. Nguyen, V. B. Truong, and Q. H. Cao, "G-came: Gaussian-class activation mapping explainer for object detectors," *arXiv preprint arXiv:2306.03400*, 2023.

[19] Q. Ye, J. Xia, and G. Yang, "Explainable AI for COVID-19 CT classifiers: An initial comparison study," *CoRR*, vol. abs/2104.14506, 2021. [Online]. Available: https://arxiv.org/abs/2104.14506

[20] T. T. H. Nguyen, V. B. Truong, V. T. K. Nguyen, Q. H. Cao, and Q. K. Nguyen, "Towards trust of explainable ai in thyroid nodule diagnosis," *arXiv preprint arXiv:2303.04731*, 2023.

[21] L. Weber, S. Lapuschkin, A. Binder, and W. Samek, "Beyond explaining: Opportunities and challenges of xai-based model improvement," *Information Fusion*, 2022.

[22] D. Gunning, E. Vorm, J. Y. Wang, and M. Turek, "Darpa's explainable ai (xai) program: A retrospective," *Applied AI Letters*, vol. 2, no. 4, p. e61, 2021. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/ail2.61

[23] S. Shao, Z. Zhao, B. Li, T. Xiao, G. Yu, X. Zhang, and J. Sun, "Crowdhuman: A benchmark for detecting human in a crowd," *CoRR*, vol. abs/1805.00123, 2018. [Online]. Available: http://arxiv.org/abs/1805.00123

[24] R. S. S. Kumar, M. Nyström, J. Lambert, A. Marshall, M. Goertzel, A. Comissoneru, M. Swann, and S. Xia, "Adversarial machine learning - industry perspectives," *CoRR*, vol. abs/2002.05646, 2020. [Online]. Available: https://arxiv.org/abs/2002.05646

[25] J. Mayer and C. Schneckenburger, "An empirical analysis and comparison of random testing techniques," in *Proceedings of the 2006 ACM/IEEE international symposium on Empirical software engineering*, 2006, pp. 105–114.

[26] B. P. Miller, G. Cooksey, and F. Moore, "An empirical study of the robustness of macos applications using random testing," in *Proceedings of the 1st international workshop on Random testing*, 2006, pp. 46–54.

[27] R. M. Conroy, "The rcsi sample size handbook," 2016.

[28] C. R. W. VanVoorhis and B. L. Morgan, "Understanding power and rules of thumb for determining sample sizes," 2007.

[29] A. Milan, L. Leal-Taixé, I. D. Reid, S. Roth, and K. Schindler, "MOT16: A benchmark for multi-object tracking," *CoRR*, vol. abs/1603.00831, 2016. [Online]. Available: http://arxiv.org/abs/1603.00831

[30] S. Zhang, R. Benenson, and B. Schiele, "Citypersons: A diverse dataset for pedestrian detection," *CoRR*, vol. abs/1702.05693, 2017. [Online]. Available: http://arxiv.org/abs/1702.05693

[31] A. Ess, B. Leibe, K. Schindler, and L. Van Gool, "A mobile vision system for robust multi-person tracking," in *2008 IEEE Conference on Computer Vision and Pattern Recognition*, 2008, pp. 1–8.

[32] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 11 443–11 452.

[33] T. Lin, M. Maire, S. J. Belongie, L. D. Bourdev, R. B. Girshick, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: common objects in context," *CoRR*, vol. abs/1405.0312, 2014. [Online]. Available: http://arxiv.org/abs/1405.0312

[34] Q. Zhang, L. Rao, and Y. Yang, "Group-cam: Group score-weighted visual explanations for deep convolutional networks," *CoRR*, vol. abs/2103.13859, 2021. [Online]. Available: https://arxiv.org/abs/2103.13859