Submission and Formatting Instructions for International Conference on Machine Learning (ICML 2026)

Anonymous Authors[1]

Abstract

Object detection systems are essential in safety-critical applications, but they are vulnerable to object disappearance (OD) threat, in which valid objects become undetected under small input perturbations, creating serious risks. This paper addresses the problem of verifying the robustness of YOLO (You Only Look Once) networks against OD by proposing a three-step probabilistic verification framework: (1) estimating output ranges under a distribution of input perturbations, (2) formally verifying the Non-Maximum Suppression (NMS) process within these ranges, and (3) iteratively refining the results to reduce over-approximation. The framework scales to practical YOLO models. Both theoretical analysis and experimental results demonstrate that our method achieves comparable probabilistic guarantees and provides tighter Intersection-over-Union (IoU) lower bounds while requiring significantly fewer samples than existing methods.

1. Introduction

Object detection (Zhao et al., 2019; Zou et al., 2023) is a fundamental computer vision task that combines object localization and classification. Neural network architectures, including YOLO (You Only Look Once) (Redmon, 2016; Redmon & Farhadi, 2017; Farhadi & Redmon, 2018; Bochkovskiy et al., 2020a), Fast R-CNN (Girshick, 2015), and SSD (Liu et al., 2016; Li et al., 2017), have achieved significant progress in both accuracy and computational efficiency, enabling their widespread deployment in real-world applications. Despite these advances, neural network-based detection systems remain vulnerable to minute, often imperceptible, input perturbations (Im Choi & Tian, 2022; Lin et al., 2025; Goodfellow et al., 2015; Madry et al., 2018; Dong et al., 2018; Carlini & Wagner, 2017). Of particular concern is the *object disappearance (OD) problem*, in which minor input perturbations suppress the detection of valid objects. Such perturbations pose substantial risks in safety-critical domains, potentially leading to catastrophic consequences due to detection failures. Consequently, verifying the safety of object detection systems is crucial for their reliable deployment.

To measure network robustness, verification methods are commonly employed. For a given network F, an input $\boldsymbol{x}$, and a property function ϕ, verification methods can be grouped into three categories:

Formal Verification. The goal is to find the maximum perturbation radius ε such that $\phi(\mathrm{F}(\boldsymbol{x}')) = \phi(\mathrm{F}(\boldsymbol{x}))$ for all $\boldsymbol{x}' \in \mathcal{B}_p(\boldsymbol{x}, \varepsilon)$, where $\mathcal{B}_p(\boldsymbol{x}, \varepsilon) = \{\boldsymbol{x}' : \|\boldsymbol{x}' - \boldsymbol{x}\|_p \leq \varepsilon\}$ is the p-norm ball of radius ε centered at $\boldsymbol{x}$. Alternatively, for a fixed ε, one can verify whether the property holds for all $\boldsymbol{x}' \in \mathcal{B}_p(\boldsymbol{x}, \varepsilon)$. However, formal verification is NP-complete (Katz et al., 2017), making it infeasible for large-scale networks. Even state-of-the-art tools (Zhang et al., 2022b;a) face challenges in handling networks with millions of neurons (Brix et al., 2023; 2024).

Probabilistic Verification. Given a radius ε and a tolerance α, the goal is to verify whether $\mathrm{P}_{\boldsymbol{x}' \sim \mathcal{D}}(\phi(\mathrm{F}(\boldsymbol{x}')) = \phi(\mathrm{F}(\boldsymbol{x}))) \geq 1 - \alpha$, where $\mathcal{D}$ is a distribution over $\mathcal{B}_p(\boldsymbol{x}, \varepsilon)$. Although this approach leverages probabilistic guarantees to reduce verification time and memory, its reliance on processing internal network nodes prevents it from scaling to larger network architectures. Representative works include (Weng et al., 2019; Boetius et al., 2025).

PAC Verification. Given ε, α, and β, the goal is to verify whether $\mathrm{P}_{\boldsymbol{x}' \sim \mathcal{D}}(\phi(\mathrm{F}(\boldsymbol{x}')) = \phi(\mathrm{F}(\boldsymbol{x}))) \geq 1 - \alpha$ holds with confidence at least $1 - \beta$. PAC methods rely on sampling and do not require access to internal network nodes, which allows them to scale further to larger models and datasets. Representative works include (Tran et al., 2023; Park et al., 2020; Li et al., 2022; Blohm et al., 2025).

Verifying object detection networks with these methods, however, presents additional challenges beyond the large parameter scales:

(1) **Post-Processing Stage**: Critical post-processing steps, such as Non-Maximum Suppression (NMS) (Neubeck &

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

Van Gool, 2006), generally fall outside the scope of current formal verification methods (Cohen et al., 2024; Elboher et al., 2024);
(2) **Large Input-Output Spaces**: The dimensionality of the detection inputs and outputs even renders PAC-based methods (Li et al., 2022; Blohm et al., 2025; Haussler & Welzl, 1987) computationally infeasible.

Due to these limitations, even recent verification methods specifically designed for object detection (Cohen et al., 2024; Elboher et al., 2024) are restricted to simplified models or do not account for complex operations such as NMS. To address this gap, we propose a PAC-based **O**bject **D**etection **P**robabilistic **V**erification (ODPV) framework for YOLO networks under OD threats. To our knowledge, **this is the first framework that effectively verifies the robustness of the original object detection networks at a practical scale**. Although PAC verification cannot provide deterministic guarantees, it currently offers the most practical means to validate YOLO in a reasonable time.

Our methodology includes three main components: (1) estimating output ranges under input perturbations, (2) formally verifying NMS within the estimated output space, and (3) iteratively refining verification results. We implement our approach and evaluate it on standard benchmarks. Our main contributions are as follows.

(1) We formally define the PAC verification problem of the OD threat in object detection and propose a novel verification approach to address it.
(2) We implement a complete verification process that includes the NMS step, which has been under-explored in previous work, and provide probabilistic guarantees for each step.
(3) We conduct experiments on widely used networks and datasets to evaluate our proposed method. We demonstrate that our method requires fewer samples to achieve comparable probabilistic guarantees and tighter certified Intersection-over-Union (IoU) bounds.

In summary, we are the first to address the challenges of verifying large-scale detection networks and to provide an efficient probabilistic verification method.

Remark 1.1. We emphasize an important distinction: Our work differs from randomized smoothing in the type of guarantee it provides (Cohen et al., 2019; Yang et al., 2020). Randomized smoothing establishes robustness for modified, "smoothed" classifiers, not the original detector. In contrast, we leave the network unchanged and provide statistical guarantees for the original model.

2. Related Work

Object detection. Early detectors relied on hand-crafted features such as HOG (Dalal & Triggs, 2005) and sliding windows (Viola & Jones, 2001), but lacked adaptability. CNN-based approaches transformed feature extraction; R-CNN variants (Girshick et al., 2014; Ren, 2015) combined region proposals with deep learning methods. More recent approaches such as YOLO (Redmon, 2016; Redmon & Farhadi, 2017; Farhadi & Redmon, 2018; Bochkovskiy et al., 2020b) and SSD (Liu et al., 2016; 2017) achieved real-time detection in complex scenarios.

Verification techniques for Neural Networks. Formal verification determines whether a property holds under given input constraints. State-of-the-art tools (Katz et al., 2017; 2019; Zhang et al., 2022a; 2018) employ Branch-and-Bound, combining relaxations (Singh et al., 2019; Bak, 2021), bound propagation (Wang et al., 2018b; Weng et al., 2018; Wang et al., 2018a; Gowal et al., 2019), and constraint solving (Khedr et al., 2021; Ehlers, 2017; Henriksen & Lomuscio, 2020; Kouvaros & Lomuscio, 2021). However, for large networks such as YOLO (with $640 \times 480 \times 3$ inputs), even basic bound propagation may require more than 5000 GB of memory, rendering formal verification infeasible in practice. To address scalability, probabilistic verification estimates the likelihood of property satisfaction. Sampling-based methods (Webb et al., 2019; Cardelli et al., 2019; Mangal et al., 2019; Anderson & Sojoudi, 2023) provide probabilistic estimates, but may miss rare cases, thereby creating gaps between analysis and actual robustness. DeepPAC (Li et al., 2022) approximates local network behavior with linear equations and high-confidence error bounds, but it requires prohibitively large sample sizes for models such as YOLO. Techniques like median smoothing (Chiang et al., 2020) certify robustness for a modified, "smoothed" detector, whereas our approach directly verifies the original network.

Verification of Object Detection. Current efforts mainly focus on small or simplified detectors. (Cohen et al., 2024) propagate bounds to certify IoU, while (Elboher et al., 2024) encode IoU into networks for existing verifiers. Both approaches ignore the NMS step and fail to scale to real-world detectors. Comprehensive verification of complete detection pipelines remains an open problem.

3. Preliminaries

This section outlines the key stages of YOLO object detection, as shown in Fig. 1- 3 with an image from the COCO validation dataset (Lin et al., 2014) and defines the threat of OD.

3.1. Key Stages of YOLO Object Detection

Bounding Box Prediction (First Stage). The YOLO network $\mathrm{F} : \mathbb{R}^{d_0} \to \mathbb{R}^{d_L}$ processes an input $\boldsymbol{x}$ (with dimension d_0) to generate an output $\boldsymbol{y} = \mathrm{F}(\boldsymbol{x})$ (with dimension d_L).

Figure 1. (First Stage) The network tries to find all boxes that may contain objects. A subset of these boxes is shown here.

Figure 2. (Second Stage) Final output boxes selected by NMS include the corresponding label and its confidence score.

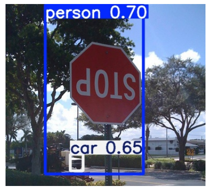

Figure 3. Under imperceptible input perturbations, YOLO can no longer recognize these objects.

The output $\boldsymbol{y}$ can be reformulated as a set of bounding boxes $\{box_i\}_{i=1}^{n_{\boldsymbol{x}}}$, where $n_{\boldsymbol{x}}$ is a constant determined by the fixed input dimension. Each bounding box box_i is represented as $(x_i, y_i, w_i, h_i, c_i, p_{i_1}, p_{i_2}, \ldots, p_{i_n})$. Here, (x_i, y_i) denotes the box's center coordinates, (w_i, h_i) its width and height, c_i its confidence score, and p_{i_j} the probability of the object belonging to class j (for $j \in [n]$, where n is the total number of classes). The class of box_i is assigned as $\mathrm{Class}(box_i) = \arg\max_{j\in[n]} p_{i_j}$. These boxes collectively identify possible object locations in the input image, as Figure 1 illustrates.

Non-Maximum Suppression (Second Stage). Let $\boldsymbol{y} = \mathrm{F}(\boldsymbol{x})$ be the output tensor from the first stage. The second stage processes $\boldsymbol{y}$ by using an operator N to select a subset of bounding boxes $\{box_{i_j}\}_{i_j\in[n_{\boldsymbol{x}}]} \subseteq \boldsymbol{y} = \{box_i\}_{i=1}^{n_{\boldsymbol{x}}}$, forming the final YOLO output (Figure 2). The standard operator N is NMS (Neubeck & Van Gool, 2006) in YOLO, which uses $\boldsymbol{y}$ and predefined thresholds $\eta, \iota \in (0, 1)$ to select the final output. For simplicity, we denote this as $\mathrm{N}(\boldsymbol{y})$, as η and ι are fixed, so we omit them. NMS selects boxes based on the following three rules:
(n1): If $i_j \in [n_{\boldsymbol{x}}]$ and $box_{i_j} \in \mathrm{N}(\boldsymbol{y})$, then it must satisfy $c_{i_j} \geq \iota$;
(n2): If $i_j \in [n_{\boldsymbol{x}}]$ satisfies $box_{i_j} \notin \mathrm{N}(\boldsymbol{y})$ and $c_{i_j} \geq \iota$, then there must exist a $box_{i_k} \in \mathrm{N}(\boldsymbol{y})$ such that $\mathrm{Class}(box_{i_j}) = \mathrm{Class}(box_{i_k})$ and $c_{i_j} \leq c_{i_k}$, $\mathrm{IoU}(box_{i_j}, box_{i_k}) \geq \eta$;
(n3): If $i_j, i_k \in [n_{\boldsymbol{x}}]$ such that $box_{i_j}, box_{i_k} \in \mathrm{N}(\boldsymbol{y})$ and $\mathrm{Class}(box_{i_j}) = \mathrm{Class}(box_{i_k})$, then it must satisfy $\mathrm{IoU}(box_{i_j}, box_{i_k}) < \eta$.

The $\mathrm{IoU}(box_1, box_2) = \frac{\mathrm{Area}(box_1 \cap box_2)}{\mathrm{Area}(box_1 \cup box_2)}$ measures overlap between two boxes, where $\mathrm{Area}(box_1 \cap box_2)$ and $\mathrm{Area}(box_1 \cup box_2)$ denote the IoU areas. The NMS-selected subset is unique and we focus on its properties, as implementation details are beyond our scope.

3.2. Object Disappearance Threat on Object Detection

An object detection model successfully detects an object O in the image $\boldsymbol{x}$ if there exists at least one $box_i \in \mathrm{N}(\mathrm{F}(\boldsymbol{x}))$ satisfying: $\mathrm{Class}(box_i) = \mathrm{Class}(box_{\mathrm{gt}})$ and $\mathrm{IoU}(box_i, box_{\mathrm{gt}}) \geq \tau$, where τ is a predefined IoU threshold and box_{gt} is O's ground truth bounding box. We define the OD threat as follows:

OD Threat Definition. Given ground truth box box_{gt}, perturbation radius ε, IoU threshold τ, and class $\mathrm{Class}(box_{\mathrm{gt}})$, OD occurs if there exists a perturbation $\boldsymbol{\delta}$ with $\|\boldsymbol{\delta}\|_p \leq \varepsilon$ such that

$$\max_{box_i \in \mathrm{N}(\mathrm{F}(\boldsymbol{x}+\boldsymbol{\delta}))} \Big[\mathrm{IoU}(box_i, box_{\mathrm{gt}}) \cdot \mathbb{I}\left(\mathrm{Class}(box_i) = \mathrm{Class}(box_{\mathrm{gt}})\right) \Big] < \tau.$$

where $\mathbb{I}(\cdot)$ denotes an indicator function (returns 1 if true, 0 otherwise).

Impact Statement

Authors are **required** to include a statement of the potential broader impact of their work, including its ethical aspects and future societal consequences. This statement should be in an unnumbered section at the end of the paper (co-located with Acknowledgements – the two may appear in either order, but both must be before References), and does not count toward the paper page limit. In many cases, where the ethical impacts and expected societal implications are those that are well established when advancing the field of Machine Learning, substantial discussion is not required, and a simple statement such as the following will suffice:

"This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here."

The above statement can be used verbatim in such cases, but we encourage authors to think about whether there is content which does warrant further discussion, as this statement will be apparent if the paper is later flagged for ethics review.

References

Anderson, B. G. and Sojoudi, S. Data-driven certification of neural networks with random input noise. *IEEE Transactions on Control of Network Systems*, 10(1):249–260, 2023. doi: 10.1109/TCNS.2022.3199148.

Bak, S. nnenum: Verification of ReLU neural networks with optimized abstraction refinement. In *Proceedings of the 13th International Symposium on NASA Formal Methods (NFM)*, pp. 19–36. Springer, 2021.

Blohm, P., Indri, P., Gärtner, T., and MALHOTRA, S. Probably approximately global robustness certification. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=UKHlXpiFMy.

Bochkovskiy, A., Wang, C., and Liao, H. M. Yolov4: Optimal speed and accuracy of object detection. *CoRR*, abs/2004.10934, 2020a. URL https://arxiv.org/abs/2004.10934.

Bochkovskiy, A., Wang, C.-Y., and Liao, H.-Y. M. Yolov4: Optimal speed and accuracy of object detection, 2020b.

Boetius, D., Leue, S., and Sutter, T. Solving probabilistic verification problems of neural networks using branch and bound. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=suZ1pNdKrV.

Brix, C., Bak, S., Liu, C., and Johnson, T. T. The fourth international verification of neural networks competition (vnn-comp 2023): Summary and results, 2023.

Brix, C., Bak, S., Johnson, T. T., and Wu, H. The fifth international verification of neural networks competition (vnn-comp 2024): Summary and results, 2024. URL https://arxiv.org/abs/2412.19985.

Cardelli, L., Kwiatkowska, M., Laurenti, L., and Patane, A. Robustness guarantees for bayesian inference with gaussian processes. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pp. 7759–7768, 2019.

Carlini, N. and Wagner, D. A. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy, SP 2017, San Jose, CA, USA, May 22-26, 2017*, pp. 39–57. IEEE Computer Society, 2017. doi: 10.1109/SP.2017.49. URL https://doi.org/10.1109/SP.2017.49.

Chiang, P.-y., Curry, M., Abdelkader, A., Kumar, A., Dickerson, J., and Goldstein, T. Detection as regression: Certified object detection with median smoothing. *Advances in Neural Information Processing Systems*, 33:1275–1286, 2020.

Cohen, J., Rosenfeld, E., and Kolter, Z. Certified adversarial robustness via randomized smoothing. In *international conference on machine learning*, pp. 1310–1320. PMLR, 2019.

Cohen, N., Ducoffe, M., Boumazouza, R., Gabreau, C., Pagetti, C., Pucel, X., and Galametz, A. Verification for object detection–ibp iou. *arXiv preprint arXiv:2403.08788*, 2024.

Dalal, N. and Triggs, B. Histograms of oriented gradients for human detection. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pp. 886–893 vol. 1, 2005. doi: 10.1109/CVPR.2005.177.

Dong, Y., Liao, F., Pang, T., Su, H., Zhu, J., Hu, X., and Li, J. Boosting adversarial attacks with momentum. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pp. 9185–9193. Computer Vision Foundation / IEEE Computer Society, 2018. doi: 10.1109/CVPR.2018.00957.

Ehlers, R. Formal verification of piece-wise linear feed-forward neural networks. In *Automated Technology for Verification and Analysis: 15th International Symposium, ATVA 2017, Pune, India, October 3–6, 2017, Proceedings 15*, pp. 269–286. Springer, 2017.

Elboher, Y. Y., Raviv, A., Weiss, Y. L., Cohen, O., Assa, R., Katz, G., and Kugler, H. Formal verification of deep neural networks for object detection, 2024. URL https://arxiv.org/abs/2407.01295.

Farhadi, A. and Redmon, J. Yolov3: An incremental improvement. In *Computer vision and pattern recognition*, volume 1804, pp. 1–6. Springer Berlin/Heidelberg, Germany, 2018.

Girshick, R. Fast r-cnn. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 1440–1448, 2015. doi: 10.1109/ICCV.2015.169.

Girshick, R., Donahue, J., Darrell, T., and Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 580–587, 2014.

Goodfellow, I. J., Shlens, J., and Szegedy, C. Explaining and harnessing adversarial examples. In Bengio, Y. and LeCun, Y. (eds.), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL http://arxiv.org/abs/1412.6572.

Gowal, S., Dvijotham, K. D., Stanforth, R., Bunel, R., Qin, C., Uesato, J., Arandjelovic, R., Mann, T., and Kohli, P. Scalable verified training for provably robust image classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4842–4851, 2019.

Haussler, D. and Welzl, E. ε-nets and simplex range queries. *Discrete & Computational Geometry*, 2:127–151, 1987. URL https://api.semanticscholar.org/CorpusID:27638326.

Henriksen, P. and Lomuscio, A. Efficient neural network verification via adaptive refinement and adversarial search. In *ECAI 2020*, pp. 2513–2520. IOS Press, 2020.

Im Choi, J. and Tian, Q. Adversarial attack and defense of yolo detectors in autonomous driving scenarios. In *2022 IEEE Intelligent Vehicles Symposium (IV)*, pp. 1011–1017. IEEE, 2022.

Katz, G., Barrett, C., Dill, D. L., Julian, K., and Kochenderfer, M. J. Reluplex: An efficient smt solver for verifying deep neural networks. In *Proceedings of the 29th International Conference on Computer Aided Verification (CAV)*, pp. 97–117. Springer, 2017.

Katz, G., Huang, D. A., Ibeling, D., Julian, K., Lazarus, C., Lim, R., Shah, P., Thakoor, S., Wu, H., Zeljic, A., Dill, D. L., Kochenderfer, M. J., and Barrett, C. W. The marabou framework for verification and analysis of deep neural networks. In Dillig, I. and Tasiran, S. (eds.), *Computer Aided Verification - 31st International Conference, CAV 2019, New York City, NY, USA, July 15-18, 2019, Proceedings, Part I*, volume 11561 of *Lecture Notes in Computer Science*, pp. 443–452. Springer, 2019. doi: 10.1007/978-3-030-25540-4_26. URL https://doi.org/10.1007/978-3-030-25540-4_26.

Khedr, H., Ferlez, J., and Shoukry, Y. Peregrinn: Penalized-relaxation greedy neural network verifier. In *Computer Aided Verification: 33rd International Conference, CAV 2021, Virtual Event, July 20–23, 2021, Proceedings, Part I 33*, pp. 287–300. Springer, 2021.

Kouvaros, P. and Lomuscio, A. Towards scalable complete verification of ReLU neural networks via dependency-based branching. In *IJCAI*, pp. 2643–2650, 2021.

Langley, P. Crafting papers on machine learning. In Langley, P. (ed.), *Proceedings of the 17th International Conference on Machine Learning (ICML 2000)*, pp. 1207–1216, Stanford, CA, 2000. Morgan Kaufmann.

Li, R., Yang, P., Huang, C.-C., Sun, Y., Xue, B., and Zhang, L. Towards practical robustness analysis for dnns based on pac-model learning. In *Proceedings of the 44th International Conference on Software Engineering*, pp. 2189–2201, 2022.

Li, Z., Yang, L., and Zhou, F. Fssd: feature fusion single shot multibox detector. *arXiv preprint arXiv:1712.00960*, 2017.

Lin, T., Yu, L., Jin, G., Li, R., Wu, P., and Zhang, L. Out-of-bounding-box triggers: A stealthy approach to cheat object detectors. In Leonardis, A., Ricci, E., Roth, S., Russakovsky, O., Sattler, T., and Varol, G. (eds.), *Computer Vision – ECCV 2024*, pp. 269–287, Cham, 2025. Springer Nature Switzerland. ISBN 978-3-031-72848-8.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pp. 740–755. Springer, 2014.

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

Liu, Y., Chen, X., Liu, C., and Song, D. Delving into transferable adversarial examples and black-box attacks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. URL https://openreview.net/forum?id=Sys6GJqxl.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. Towards deep learning models resistant to adversarial attacks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. URL https://openreview.net/forum?id=rJzIBfZAb.

Mangal, R., Nori, A. V., and Orso, A. Robustness of neural networks: a probabilistic and practical approach. In *Proceedings of the 41st International Conference on Software Engineering: New Ideas and Emerging Results*, ICSE-NIER '19, pp. 93–96. IEEE Press, 2019. doi: 10.1109/ICSE-NIER.2019.00032. URL https://doi.org/10.1109/ICSE-NIER.2019.00032.

Neubeck, A. and Van Gool, L. Efficient non-maximum suppression. In *18th international conference on pattern recognition (ICPR'06)*, volume 3, pp. 850–855. IEEE, 2006.

Park, S., Bastani, O., Matni, N., and Lee, I. PAC confidence sets for deep neural networks via calibrated prediction. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020. URL https://openreview.net/forum?id=BJxVI04YvB.

Redmon, J. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016.

Redmon, J. and Farhadi, A. Yolo9000: better, faster, stronger. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7263–7271, 2017.

Ren, S. Faster r-cnn: Towards real-time object detection with region proposal networks. *arXiv preprint arXiv:1506.01497*, 2015.

Singh, G., Gehr, T., Püschel, M., and Vechev, M. An abstract domain for certifying neural networks. *Proceedings of the ACM on Programming Languages*, 3(POPL):1–30, 2019.

Tran, H.-D., Choi, S., Okamoto, H., Hoxha, B., Fainekos, G., and Prokhorov, D. Quantitative verification for neural networks using probstars. In *Proceedings of the 26th ACM International Conference on Hybrid Systems: Computation and Control*, HSCC '23, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9798400700330. doi: 10.1145/3575870.3587112. URL https://doi.org/10.1145/3575870.3587112.

Viola, P. and Jones, M. Rapid object detection using a boosted cascade of simple features. In *Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001*, volume 1, pp. I–I. Ieee, 2001.

Wang, S., Pei, K., Whitehouse, J., Yang, J., and Jana, S. Efficient formal safety analysis of neural networks. *Advances in neural information processing systems*, 31, 2018a.

Wang, S., Pei, K., Whitehouse, J., Yang, J., and Jana, S. Formal security analysis of neural networks using symbolic intervals. In *27th USENIX Security Symposium (USENIX Security 18)*, pp. 1599–1614, 2018b.

Webb, S., Rainforth, T., Teh, Y. W., and Kumar, M. P. Statistical verification of neural networks. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=S1xcx3C5FX.

Weng, L., Zhang, H., Chen, H., Song, Z., Hsieh, C.-J., Daniel, L., Boning, D., and Dhillon, I. Towards fast computation of certified robustness for ReLu networks. In *International Conference on Machine Learning*, pp. 5276–5285. PMLR, 2018.

Weng, L., Chen, P.-Y., Nguyen, L., Squillante, M., Boopathy, A., Oseledets, I., and Daniel, L. Proven: Verifying robustness of neural networks with a probabilistic approach. In *International Conference on Machine Learning*, pp. 6727–6736. PMLR, 2019.

Yang, G., Duan, T., Hu, J. E., Salman, H., Razenshteyn, I., and Li, J. Randomized smoothing of all shapes and sizes. In *International conference on machine learning*, pp. 10693–10705. PMLR, 2020.

Zhang, H., Weng, T.-W., Chen, P.-Y., Hsieh, C.-J., and Daniel, L. Efficient neural network robustness certification with general activation functions. *Advances in neural information processing systems*, 31, 2018.

Zhang, H., Wang, S., Xu, K., Li, L., Li, B., Jana, S., Hsieh, C., and Kolter, J. Z. General cutting planes for bound-propagation-based neural network verification. In *NeurIPS*, 2022a.

Zhang, H., Wang, S., Xu, K., Wang, Y., Jana, S., Hsieh, C.-J., and Kolter, Z. A branch and bound framework for stronger adversarial attacks of ReLU networks. In *International Conference on Machine Learning*, pp. 26591–26604. PMLR, 2022b.

Zhao, Z.-Q., Zheng, P., Xu, S.-t., and Wu, X. Object detection with deep learning: A review. *IEEE transactions on neural networks and learning systems*, 30(11):3212–3232, 2019.

Zou, Z., Chen, K., Shi, Z., Guo, Y., and Ye, J. Object detection in 20 years: A survey. *Proceedings of the IEEE*, 111(3):257–276, 2023.

A. You *can* have an appendix here.

You can have as much text here as you want. The main body must be at most 8 pages long. For the final version, one more page can be added. If you want, you can use an appendix like this one.

The `\onecolumn` command above can be kept in place if you prefer a one-column appendix, or can be removed if you prefer a two-column appendix. Apart from this possible change, the style (font size, spacing, margins, page numbering, etc.) should be kept the same as the main body.