A Contrario Paradigm for Yolo-Based Infrared Small Target Detection

Alina Ciocarlan, Sylvie Le Hegarat-Mascle, Sidonie Lefebvre, Arnaud Woiselle, Clara Barbanson

A CONTRARIO PARADIGM FOR YOLO-BASED INFRARED SMALL TARGET DETECTION

Alina Ciocarlan[1,2], Sylvie Le Hegarat-Mascle[2], Sidonie Lefebvre[1], Arnaud Woiselle[3], Clara Barbanson[3]

[1]DOTA and LMA2S, ONERA, Université Paris-Saclay, F-91123 Palaiseau, France
[2]SATIE Université Paris-Saclay, 91405 Orsay, France
[3]Safran Electronics & Defense F-91344 Massy, France

ABSTRACT

Detecting small to tiny targets in infrared images is a challenging task in computer vision, especially when it comes to differentiating these targets from noisy or textured backgrounds. Traditional object detection methods such as YOLO struggle to detect tiny objects compared to segmentation neural networks, resulting in weaker performance when detecting small targets. To reduce the number of false alarms while maintaining a high detection rate, we introduce an *a contrario* decision criterion into the training of a YOLO detector. The latter takes advantage of the *unexpectedness* of small targets to discriminate them from complex backgrounds. Adding this statistical criterion to a YOLOv7-tiny bridges the performance gap between state-of-the-art segmentation methods for infrared small target detection and object detection networks. It also significantly increases the robustness of YOLO towards few-shot settings.

Index Terms— small target detection, *a contrario* reasoning, YOLO, few-shot detection

1. INTRODUCTION

Detecting small objects in infrared (IR) images accurately is essential in various applications, including medical or security fields. Infrared small target detection (IRSTD) is a great challenge in computer vision, where the difficulties are mainly raised by (i) the size of the targets (area below 20 pixels), (ii) the complex and highly textured backgrounds, leading to many false alarms, and (iii) the learning conditions, namely learning from small, little diversified and highly class-imbalanced datasets, since the number of target class pixels is very small in comparison with the background class one. The rise of deep learning methods has led to impressive advances in object detection in the past decades, mostly thanks to their ability to learn from a huge amount of annotated data to extract non-linear features well adapted to the final task. For IRSTD, semantic segmentation neural networks (NN) are the most widely used [1]. These include ACM [2], LSPM [3] and one of the recent state-of-the-art (SOTA) method, namely DNANet [4], which consists of several nested UNets and a multiscale fusion module that enable the segmentation of small objects with variable sizes. However, a major issue of relying on segmentation NN for object detection is that object fragmentation can occur when tuning the threshold used to binarize the segmentation map. This can lead to many undesired false alarms and distort counting metrics. Object detection algorithms like Faster-RCNN [5] or YOLO [6] reduce this risk by explicitly localizing the objects thanks to the bounding box regression. However, they often have difficulty in detecting tiny objects. Very few studies have focused on adapting such detectors for IRSTD [7], and no rigorous comparison was made with SOTA IRSTD methods.

In this paper, we propose a novel YOLO detection head, called OL-NFA (for Object-Level Number of False Alarms), that is specifically designed for small object detection. This module integrates an *a contrario* decision criterion that guides the feature extraction so that *unexpected* objects stand out against the background and are detected. It is used to re-estimate the objectness scores computed by a YOLO backbone, and has been carefully implemented to allow the back-propagation during training. One advantage of using *a contrario* paradigm is that it focuses on modeling the background, for which we have a lot of samples, rather than the objects themselves. In this way, the problems of class imbalance and little training data are bypassed by carrying out the detection by rejecting the hypothesis of the background distribution. Our main contributions are as follows:

1. We design a novel YOLO detection head that integrates *a contrario* criterion for estimating the objectness scores. By focusing on modeling the background rather than the object itself, we relax the constraint of having lots of training samples.

2. We compare both SOTA segmentation NN and object detection methods on a famous IRSTD benchmark and show that adding OL-NFA to a YOLOv7-tiny backbone bridges the performance gap between object detectors and SOTA segmentation NN for IRSTD.

3. On top of that, our method improves YOLOv7-tiny performance by a large margin (39.2% AP for 15-shot) in few-shot settings, demonstrating the robustness of the *a contrario* paradigm in weak training conditions.

2. RELATED WORK

2.1. *A contrario* reasoning

A contrario decision methods allow to automatically derive a decision criterion with regards to a hypothesis test. They draw inspiration from theories of perception, in particular the Gestalt theory [8]. They consist in rejecting a naive model characterizing a destructured background by using an interpretable detection threshold. The latter allows us to control the Number of False Alarms (NFA), often defined as the product between the total number of tested *objects* and the tail distribution of the law followed by the chosen naive model. An NFA value can then be associated to any given *object* since the computed tail value depends on the object features. Several *a contrario* formulations have been proposed in the literature. They depend on whether we consider grey level or binary images. In the first case, the most commonly used naive model is the Gaussian distribution of the pixel grey-level values [9, 10, 11]. The latter has been integrated into a deep learning framework by [12], and has shown great performance for small target segmentation. In the second case, the most widely used naive model is the uniform spatial distribution of "true" pixels in the image grid. This leads to a binomial distribution of parameter p for the number of "true" pixels κ falling within any given parametric shape of area ν [13, 14]:

$$\text{NFA}\left(\kappa, \nu, p\right) = \eta \sum_{i=\kappa}^{\nu} \binom{\nu}{i} p^i \left(1-p\right)^{\nu-i}, \qquad (1)$$

where η is the number of tested objects. Based on Eq. (1), a subset of pixels likely to represent an object is all the more significant as it contains many points spatially close compared to the image overall density. Our work focuses on integrating this naive model into the training loop of an object detector to guide the feature extraction, which was not considered in previous studies. Unlike [12], whose naive model is suitable for pixel-level classification (i.e. segmentation), we consider a different model that directly applies at object level and is thus more relevant for NN with bounding box proposals.

2.2. Object detection methods

Object detection is the task of detecting objects of interest within an image and identifying their locations with bounding boxes. Several types of deep learning approaches have been proposed for such a task [15, 6]. YOLO framework is the most widely used one as it leads to great performance in various applications, with low execution time. It is a single-stage algorithm that uses a single convolutional neural network to predict together bounding box coordinates, objectness and classification scores. Concretely, it divides the image into a grid and predicts the probability (denoted as the objectness score) for any given grid cell to contain an object and the bounding box coordinates of the object if it exists. One issue with the early versions of YOLO is that they struggle in detecting small objects. Indeed, if the object to detect is too small, it may only occupy a small portion of a grid cell, making it difficult for YOLO to detect it accurately. To address this issue, YOLOv3 [16] introduced a feature pyramid network (FPN) that combines the features detected at multiple scales. Some of the latest versions of YOLO, such as YOLOR [17] or YOLOv7 [18], lead to competitive detection performance on several famous computer vision benchmarks, while also improving the execution speed. Tiny versions of YOLO with less convolutional layers have also been proposed.

3. METHOD

3.1. Overall architecture

We propose a novel YOLO detection head, called OL-NFA for object-level NFA detection head, that integrates an *a contrario* criterion to detect objects with features that *unexpectedly* deviate from the background distribution. Our OL-NFA will compute objectness score based on NFA criterion, Eq. (1), applied to feature maps derived by the network.

The overall architecture of our approach is illustrated in Fig. 1. The infrared input images first go through a YOLO backbone that extracts feature maps at different scales. Then, the three lower-level features are combined together through the neck, which gives us the final feature maps F_i used to perform the detection at three levels: $i \in \{1, 2, 3\}$. To achieve the detection, the bounding box coordinates are first predicted through a dense layer. We then introduce our OL-NFA module to re-estimate the objectness score for each bounding box using NFA criterion. To do so, we extract η regions of interest (ROI), denoted as f_{roi}, using ROI Align from Faster R-CNN [15], and we compute a *significance* score for each ROI through the significance layer described in Section 3.2. Finally, these scores are ranged in $[0, 1]$ via the function f_{act} defined in in Section 3.2, which allows us to apply the Binary Cross Entropy loss used in YOLO.

3.2. Differentiable integration of the *a contrario* criterion

Our significance layer in Fig. 1 integrates the *a contrario* criterion given in Eq. (1). However, since this equation is (i) designed for binary images rather than greyscale feature maps, and (ii) not differentiable, several approximations were made in order to allow its integration into the YOLO training loop. The first difficulty raised by Eq. (1) is to count the number of "true" pixels κ in $f_{roi} \in \mathbb{R}^2$. Thresholding f_{roi} to binarize it would break the back-propagation loop. Thus, we propose instead to consider real number membership coefficients (in the spirit of fuzzy clustering or classification), which boils down to handling, for each pixel, a coefficient indicating the degree to which it belongs to the set containing pixels with a

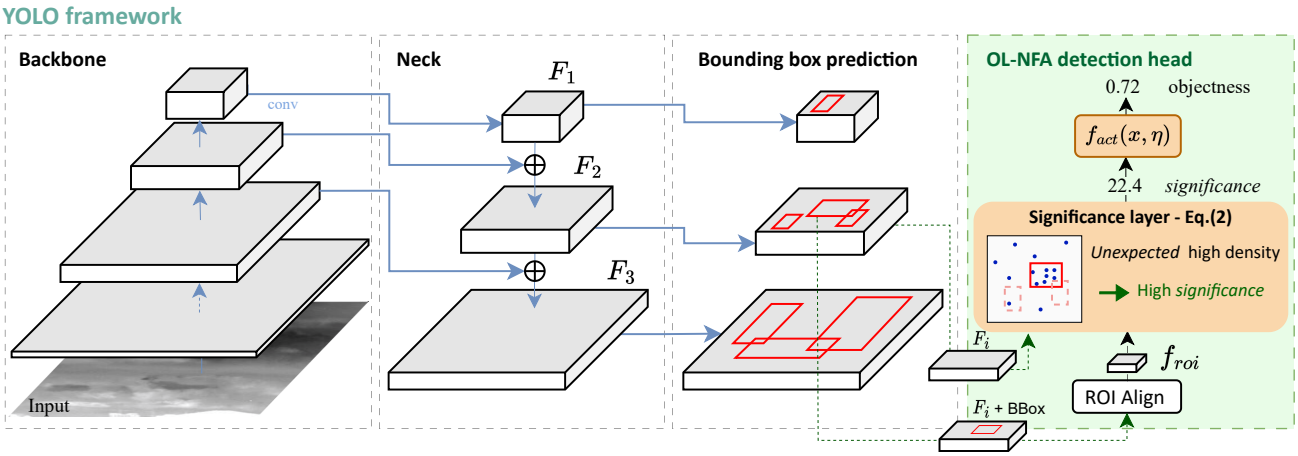

Fig. 1. Integration of our OL-NFA detection head into a YOLO framework. This module can be added on top of any YOLO.

value of 1 in the binary case. For this purpose, we apply the sigmoid function σ on the pixel values, which allows us to approximate the number of pixels contained in f_{roi} for estimating the local density, by the sum of these fuzzy belonging coefficients. The same approximation is made to compute the total number of points in F_i for estimating the parameter p (representing the global density of F_i) of the binomial law in Eq. (1). The second issue is that the NFA function is discontinuous, non differentiable and, as we deal with objects having a small area ν, it only takes very few distinct values. These elements make it difficult to integrate Eq. (1) "as is" into the training loop, with a working back-propagation. We therefore define the *significance* $S(\kappa, \nu, p) = -\ln(\text{NFA}(\kappa, \nu, p))$ and use the Hoeffding approximation when $\frac{\kappa}{\nu} > p$, leading to

$$S(\kappa, \nu, p) \approx \nu \left[\frac{\kappa}{\nu} \ln \left(\frac{\frac{\kappa}{\nu}}{p} \right) + \left(1 - \frac{\kappa}{\nu} \right) \ln \left(\frac{1 - \frac{\kappa}{\nu}}{1 - p} \right) \right] - \ln \eta. \tag{2}$$

This allows us to expand the domain of the function $S(\kappa, \nu, p)$ to $\mathbb{R}^3$, and to output more intermediate values. In the case of $\frac{\kappa}{\nu} \leq p$, we simply assign $(\kappa, \nu, p) = -\ln \eta$ as it corresponds to obvious background values. Finally, since the *significance* values range from $[-\ln(N_{test}), +\infty)$, where large values correspond to possible targets, to obtain objectness scores that range in $[0, 1]$, we apply an asymmetric activation function $f_{act}(x, \eta) = 2\sigma(x + \ln \eta) - 1$, with $x \in \mathbb{R}$ and $\eta \in \mathbb{N}^*$.

4. EXPERIMENTS

4.1. Dataset and metrics

We evaluate our method on the NUAA-SIRST dataset [2], which is one of the few infrared small target datasets publicly available and widely used in the literature. It is composed of 427 infrared images, with wavelengths ranging from 950 to 1200 nm. Targets from NUAA-SIRST have a spatial extent that vary from $2 - 3$ pixels to more than 100 pixels for the largest targets, which makes this dataset suitable to evaluate our method on a wide range of target sizes. Targets are drowned into challenging scenes such as textured clouds, as shown on the first row of Fig. 2. We split the dataset into training, validation and test sets using a ratio of $60 : 20 : 20$. We also evaluate the benefits of our method in a few-shot setting, by training the NN on 15 and 25 images only. Regarding quantitative evaluation, we focus on conventional detection metrics: the F1-score (F1) and the Average Precision (AP, area under Precision-Recall curve). We also rely on the Precision (Prec.) and the Recall (Rec.) to understand the achieved values of F1-score. In the tables, the presented results have been averaged over three distinct training sessions, and the standard deviation is given for F1 et AP in superscript.

4.2. Settings

We add our OL-NFA detection head on top of YOLOv7-tiny, as this baseline has shown to lead to good performance on NUAA-SIRST dataset compared to other YOLO backbones. We compare it to several baselines[1]: 1) segmentation networks specifically designed for IRSTD, namely ACM [2], LSPM [3] and DNANet [4]; 2) YOLO baselines such as YOLOv3 [16], YOLOR [17], YOLOv7 and YOLOv7-tiny [18]. For the IRSTD segmentation NN, we use the training settings recommended in the original papers. All object detection NN are trained from scratch on Nvidia RTX6000 GPU for 600 epochs, with Adam optimizer [19], a batch size equal to 16 and a learning rate equal to 0.001. The same settings are used for the few-shot training.

[1]For YOLO baselines, we used the official PyTorch implementation of YOLO WongKinYiu/yolov7. For IRSTD baselines we used the implementation given by kourenke/Review-Infrared-small-target-segmentation-networks

4.3. *A contrario* reasoning improves YOLO-based IRSTD

Table 1. Object-level metrics (F1, AP, Prec., Rec.) achieved by the compared methods on NUAA-SIRST. Best results are in bold and second best results are underlined. The inference time (frames per second, FPS) is also given.

Method	F1	AP	Prec.	Rec.	FPS
Segmentation networks for IRSTD					
ACM [2]	$95.4^{\pm 1.7}$	$95.2^{\pm 3.8}$	95.1	95.8	251
LSPM [3]	$85.0^{\pm 2.9}$	$90.2^{\pm 0.8}$	86.6	83.5	125
DNANet [4]	$\underline{96.9}^{\pm 0.5}$	$\underline{98.1}^{\pm 1.2}$	96.6	**97.2**	33
Object detection methods					
YOLOv3 [16]	$96.1^{\pm 0.3}$	$97.5^{\pm 0.1}$	96.9	95.4	144
YOLOR [17]	$95.7^{\pm 2.2}$	$96.7^{\pm 1.1}$	96.5	94.9	136
YOLOv7 [18]	$96.5^{\pm 1.2}$	$97.6^{\pm 0.7}$	97.2	95.9	147
YOLOv7-tiny	$96.5^{\pm 0.6}$	$97.8^{\pm 0.4}$	96.9	$\underline{96.2}$	256
Ours	$\mathbf{97.2}^{\pm 0.6}$	$\mathbf{98.2}^{\pm 0.2}$	**98.6**	95.9	208

Table 1 shows the performance achieved by each of the compared methods on NUAA-SIRST. We can see that substituting conventional YOLO detection head with our OL-NFA not only improves YOLO for tiny object detection, but also bridges the performance gap observed between SOTA IRSTD segmentation NN and conventional object detection NN. Specifically, our method achieves a F1 score higher by 0.7% than the best YOLO baseline. The AP criterion is also increased by 0.4%. Moreover, our method performs slightly better in terms of F1 and AP than DNANet, which is SOTA method for IRSTD. The inference time for our method is also much lower than for DNANet, thus allowing for real-time object detection. The high performance of our OL-NFA module is mostly due to a higher precision with a limited loss in recall, which is explained by the NFA property of controlling the number of false alarms. Indeed, adding an *a contrario* decision criterion helps in enhancing small object features and thus discriminating them from complex backgrounds. This can be seen in Fig. 2, where the best YOLO baseline leads to several false alarms for inputs 3 and 4, while our method provides correct detections without any false alarm.

4.4. OL-NFA brings robustness towards few-shot settings

Table 2. Results achieved in 15 and 25-shot settings on NUAA-SIRST. Best results are in bold.

Method	15-shots		25-shots	
	F1	AP	F1	AP
YOLOv7-tiny	$50.7^{\pm 7.0}$	$51.3^{\pm 7.0}$	$68.0^{\pm 6.6}$	$69.6^{\pm 8.4}$
Ours	$\mathbf{85.0}^{\pm 5.0}$	$\mathbf{90.5}^{\pm 5.2}$	$\mathbf{89.7}^{\pm 4.2}$	$\mathbf{93.4}^{\pm 2.0}$

One important motivation of integrating *a contrario* reasoning into a NN is that the network learns to discriminate small targets by learning a representation of background elements rather than the targets themselves. It should thus provide robustness to the NN towards weak training conditions. To confirm our intuition, we quantitatively evaluate the benefit of the proposed approach in a few-shot setting on NUAA-SIRST dataset. For this purpose, we trained the networks on 15 and 25 images. For each few-shot setting, we train the detectors on three distinct folds, with no overlap between them. The results obtained on the test set defined in Section 4.1 are averaged over these three folds and computed means are given in Table 2. It can be seen that our method performs significantly better in a frugal setting than the baseline. Indeed, in those cases, both F1 score and Average Precision are increased by at least 20%. We thus conclude that adding an object-level NFA to the baseline significantly improves its robustness towards frugal setting: the F1 score is decreased by only 15% when dividing by more than 10 the number of training samples and the AP is maintained above 90%.

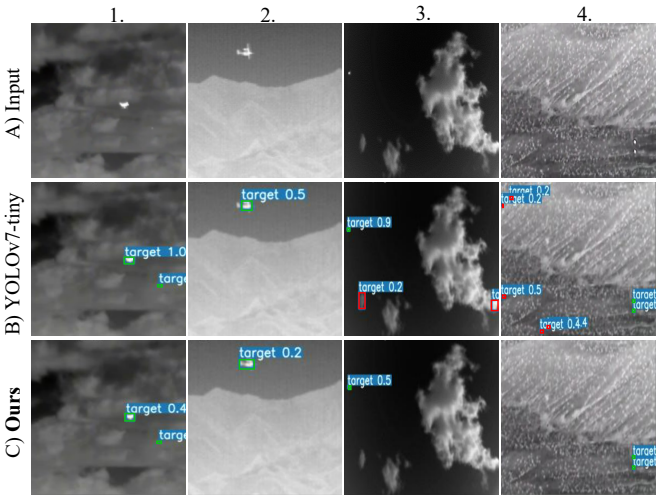

Fig. 2. Qualitative results obtained with YOLOv7-tiny and our method on NUAA-SIRST dataset. Good detections and false positives are framed in green and red, respectively.

5. CONCLUSION

In this paper, we propose a novel YOLO detection head named OL-NFA that integrates an *a contrario* decision criterion into the training loop of a YOLO network. It forces the network to model the background distribution rather than the objects to detect. Extensive experiments have shown that our method not only significantly improves YOLO performance for small object detection in frugal and few-shot settings, but also performs on par with SOTA segmentation networks for small target detection. This promising performance encourages to consider further research into using *a contrario* paradigm for tiny object detection.

6. REFERENCES

[1] Renke Kou, Chunping Wang, Zhenming Peng, Zhihe Zhao, Yaohong Chen, Jinhui Han, Fuyu Huang, Ying Yu, and Qiang Fu, "Infrared small target segmentation networks: A survey," *Pattern Recognition*, vol. 143, pp. 109788, 2023.

[2] Yimian Dai, Yiquan Wu, Fei Zhou, and Kobus Barnard, "Asymmetric contextual modulation for infrared small target detection," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2021, pp. 950–959.

[3] Lian Huang, Shaosheng Dai, Tao Huang, Xiangkang Huang, and Haining Wang, "Infrared small target segmentation with multiscale feature representation," *Infrared Physics & Technology*, vol. 116, pp. 103755, 2021.

[4] Boyang Li, Chao Xiao, Longguang Wang, Yingqian Wang, Zaiping Lin, Miao Li, Wei An, and Yulan Guo, "Dense nested attention network for infrared small target detection," *IEEE Transactions on Image Processing*, vol. 32, pp. 1745–1758, 2022.

[5] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *Advances in neural information processing systems*, vol. 28, pp. 91–99, 2015.

[6] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.

[7] Xingang Mou, Shuai Lei, and Xiao Zhou, "Yolo-fr: A yolov5 infrared small target detection algorithm based on feature reassembly sampling method," *Sensors*, vol. 23, no. 5, pp. 2710, 2023.

[8] Agnes Desolneux, Lionel Moisan, and Jean-Michel Morel, *From gestalt theory to image analysis: a probabilistic approach*, vol. 34, Springer Science & Business Media, 2007.

[9] Amandine Robin, Lionel Moisan, and Sylvie Le Hégarat-Mascle, "An a-contrario approach for subpixel change detection in satellite imagery," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 11, pp. 1977–1993, 2010.

[10] Thibaud Ehret, Axel Davy, Mauricio Delbracio, and Jean-Michel Morel, "How to reduce anomaly detection in images to anomaly detection in noise," *Image Processing On Line*, vol. 9, pp. 391–412, 2019.

[11] Vincent Vidal, Matthieu Limbert, Tugdual Ceillier, and Lionel Moisan, "Aggregated primary detectors for generic change detection in satellite images," in *IGARSS 2019-2019 IEEE International Geoscience and Remote Sensing Symposium*. IEEE, 2019, pp. 59–62.

[12] Alina Ciocarlan, Sylvie Le Hegarat-Mascle, Sidonie Lefebvre, and Arnaud Woiselle, "Deep-nfa: a deep *a contrario* framework for small object detection," *arXiv preprint arXiv:2303.01363*, 2023.

[13] Agnès Desolneux, Lionel Moisan, and J-M Morel, "A grouping principle and four applications," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 4, pp. 508–513, 2003.

[14] Sylvie Le Hégarat-Mascle, Emanuel Aldea, and Jennifer Vandoni, "Efficient evaluation of the number of false alarm criterion," *EURASIP J. Image Video Process.*, vol. 2019, pp. 35, 2019.

[15] Ross Girshick, "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440–1448.

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

[17] Chien-Yao Wang, I-Hau Yeh, and Hong-Yuan Mark Liao, "You only learn one representation: Unified network for multiple tasks," *arXiv preprint arXiv:2105.04206*, 2021.

[18] Chien-Yao Wang, Alexey Bochkovskiy, and Hong-Yuan Mark Liao, "Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," *arXiv preprint arXiv:2207.02696*, 2022.

[19] Diederik P Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.