Deep-NFA: A deep a contrario framework for tiny object detection

Alina Ciocarlan, Sylvie Le Hégarat-Mascle, Sidonie Lefebvre, Arnaud Woiselle

Pattern Recognition 150 (2024) 110312

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Deep-NFA: A deep *a contrario* framework for tiny object detection

Alina Ciocarlan [a,b,*], Sylvie Le Hégarat-Mascle [b], Sidonie Lefebvre [a], Arnaud Woiselle [c]

[a] DOTA & LMA2S, ONERA, Paris-Saclay University, F-91123 Palaiseau, France
[b] SATIE, Paris-Saclay University, 91405 Orsay, France
[c] Safran Electronics & Defense, F-91344 Massy, France

ARTICLE INFO

Keywords:
A contrario reasoning
One-class semantic segmentation
Deep learning
Convolutional neural networks
Small target detection

ABSTRACT

The detection of tiny objects is a challenging task in computer vision. Conventional object detection methods have difficulties in finding the balance between high detection rate and low false alarm rate. In the literature, some methods have addressed this issue by enhancing the feature map responses for small objects, but without guaranteeing robustness with respect to the number of false alarms induced by background elements. To tackle this problem, we introduce an *a contrario* decision criterion into the learning process to take into account the *unexpectedness* of tiny objects. This statistic criterion enhances the feature map responses while controlling the number of false alarms (NFA) and can be integrated as an add-on into any semantic segmentation neural network. Our add-on NFA module not only allows us to obtain competitive results for small target, road crack and ship detection tasks respectively, but also leads to more robust and interpretable results.

1. Introduction

The detection of tiny objects, defined as having at least one small (1–5 pixels) dimension, is a great challenge in computer vision. This issue is of a key interest in many real-world applications, e.g. in the defense and security fields for surveillance or in the medical field for early and accurate diagnosis. The rise of deep learning methods has led to impressive progress in object detection in the past decades, mostly thanks to their ability to extract non-linear features well adapted to the downstream task. However, most state-of-the-art (SOTA) object detection methods such as YOLO [1] perform poorly on very small objects. Indeed, neural networks (NN) based on bounding box regression extract features at deeper levels and the lack of data reconstruction erases small structures, which are essential for the detection of tiny objects. Semantic segmentation methods are then preferred, although their performance remains limited. This is firstly due to the nature of the objects: their surface area is made of only few pixels, and they do not present a specific structure. Secondly, tiny objects are often partially hidden in complex and highly textured backgrounds, leading to many false alarms. Some examples are shown on Fig. 1. Moreover, dealing with small object detection results in learning from highly class-imbalanced datasets. Thus, in a semantic segmentation scheme, tiny object features cannot be learned easily due to the small number of samples in comparison with the background class.

Among the few approaches that have been proposed to improve tiny object detection, some of them focus on augmenting or oversampling the dataset [2], while others focus on improving small object feature-enhancement. Among the latter, we can cite the Feature Pyramid Networks (FPN) and their variants [3], whose multi-scale approach is beneficial for the detection of objects of various sizes. Other works include attention mechanisms to learn long-range dependencies [4], or use super-resolution to enhance feature responses of small objects [5]. However, these methods do not take advantage of the *unexpectedness* of small objects with respect to the background, as one could do in an anomaly detection approach with, for example, one-class classifiers [6], that discriminate small objects as *unexpected* patterns with respect to the background. Such a criterion can efficiently reduce the number of false alarms induced by the background and thus can allow for a better balance between precision and detection rate.

We therefore propose a new deep learning paradigm for detecting tiny objects by taking into account their unexpectedness. It relies on *a contrario* reasoning, introduced by [7]. These methods allow us to automatically derive a decision criterion by modeling the background using a naive model and detecting structures or objects as too structured to appear 'by chance' under the naive model. They draw inspiration from theories of perception, in particular the Gestalt theory [8]. The latter are based on the Helmholtz principle, which states that a large deviation from a random pattern is probably due to the presence of a structure. Our motivation of using such methods is that they model the background, for which we have a lot of samples, rather than the objects to be detected. It thus circumvents the problem of class imbalance

* Corresponding author at: DOTA & LMA2S, ONERA, Paris-Saclay University, F-91123 Palaiseau, France.
E-mail address: alina.ciocarlan@onera.fr (A. Ciocarlan).

https://doi.org/10.1016/j.patcog.2024.110312
Received 16 May 2023; Received in revised form 24 September 2023; Accepted 31 January 2024
Available online 2 February 2024

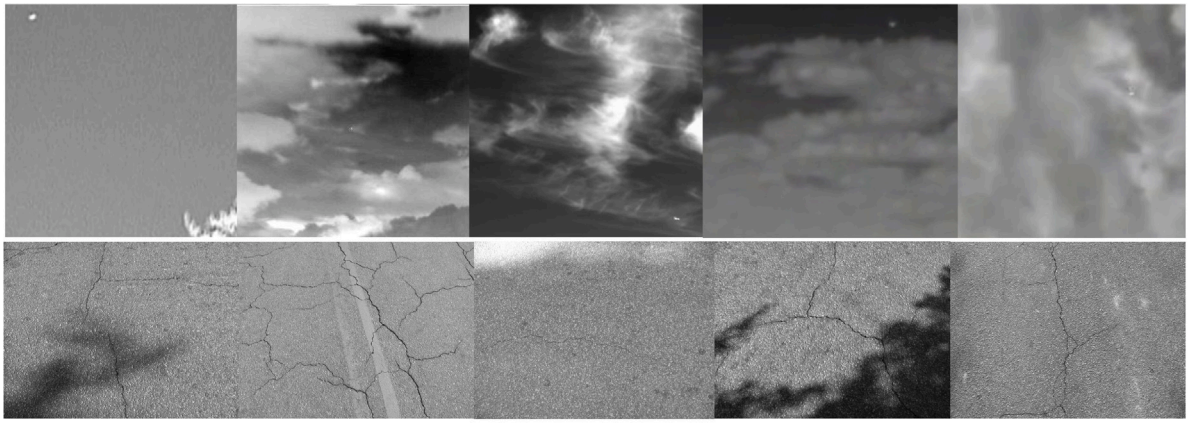

Fig. 1. Example of tiny objects. The first line shows small targets on a sky background. Note the challenging conditions: very small targets, low contrast, cloud-induced textures. The second line shows road cracks, which have different thicknesses, and are sometimes blended with the textured roads or shadows.

by focusing on the background class and performing detection by rejection of its distribution hypothesis. Such a modeling appears even more appropriate as tiny objects often contain very few geometric features, unlike larger objects for which the literature is very extensive. Moreover, the *a contrario* formulation aims at minimizing the Number of False Alarms (NFA, defined in Section 2.1.1), thus allowing for a better control of the precision.

In the literature, the *a contrario* decision is applied either on natural images directly or after extracting features from the image by traditional image processing methods. This filtering step can be replaced by Neural Networks (NN). Indeed, when looking at feature maps of a NN trained for detecting objects, the objects to detect stand out against a background made of noise. [9] applied, as a post-processing step, *a contrario* detection on feature maps obtained by a NN. However, doing so appears suboptimal since the feature map statistical distribution may not match the naive assumption made on the background when applying *a contrario* decision. We therefore propose to guide the NN training by including the *a contrario* criterion in the training loop through our NFA module. The latter guides the network to extract features in a way that the object features will be likely to contradict the naive hypothesis made on the background. This induces interesting properties: (1) the results are more interpretable; (2) the threshold choice allows for a more intuitive control of the number of false alarms (NFA). Our NFA module can be integrated into any segmentation NN, and can even take advantage of multi-scale information if the backbone allows it. We can summarize our main contributions as follows:

1. We propose a new module specifically designed for tiny object detection that takes into account the unexpectedness of an object thanks to an *a contrario* decision criterion.
2. We demonstrate the effectiveness of our method for infrared small target detection, which achieves state-of-the-art performance while also leading to more interpretable results.
3. We extend our experiments to other backbones and applications, namely road crack detection and ship detection, in order to show the generalization ability of the proposed method.

2. Theoretical background

2.1. A contrario decision criterion

2.1.1. General a contrario framework

A contrario reasoning consists in rejecting a naive model characterizing a destructured background by using an interpretable detection threshold. The latter controls 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. Many *a contrario* formulations have been proposed in the literature, relying on different naive models. An important distinction is the kind of considered images, namely either binary or gray-level images. In the first case, the most widely used naive model is the uniform spatial distribution of the "true" pixels in the image lattice, leading to binomial distribution for the number of "true" pixels falling within any given parametric shape [7,10]. In the second case, the most widely used naive model is the Gaussian distribution of the pixel gray-level values, leading to chi-square distribution for the sum of the squared errors [9,11]. We base our approach on this second formulation since we will deal with gray-scale feature maps.

Assuming the naive model for the background is a centered Gaussian distribution (hypothesis H_0) with parameters p_{H_0}, the following function is defined, for each tested pixel $i \in \mathbb{N}$ with value x_i:

$$f(x_i, N_{test}, p_{H_0}) = N_{test} \times \mathbb{P}_{H_0}(\|X\|_2^2 \geq \|x_i\|_2^2), \tag{1}$$

where X is a sequence of variables that are assumed to follow H_0, $\mathbb{P}_{H_0}$ the associated probability and N_{test} is the so-called "number of tests" that corresponds to the total number of analyzed observations x.

As stated by [8], f defines a Number of False Alarms (NFA) provided that, $\forall \epsilon > 0$, and for $X_i \sim H_0$, it is ϵ-meaningful, i.e. the following condition is verified:

$$\mathbb{E}[\#\{i, f(X_i, N_{test}, p_{H_0}) \leq \epsilon\}] \leq \epsilon, \tag{2}$$

where the symbol $\mathbb{E}[.]$ stands for the mathematical expectation and $\#\{.\}$ for the cardinality of a set. This property guarantees that, on average, raising a detection every time f is lower than ϵ should lead to at most ϵ false alarms. Thus, such a function allows for the control of the number of false alarms. [12] showed that defining N_{test} as the total amount of tested elements allows f to verify the condition (2). Thereafter, we define N_{test} as the number of pixels composing the image and we call NFA the tested value $f(x_i, N_{test}, p_{H_0})$.

2.1.2. Multi-channel formulation

In [9], the authors adapted the previous single channel formulation to multi-channel input by considering each channel independently. The obtained NFA maps are then merged together by taking the union of detections. In this study, we rather reformulate the previous approach in terms of a multivariate normal distribution, as suggested by [11]. By considering a centered input X_i with K channels, we can rewrite Eq. (1) using the Gamma and upper incomplete Gamma functions (denoted $\Gamma(.)$ and $\Gamma(.,.)$ respectively):

$$\text{NFA}(x_i, N_{test}, K, \Sigma) = \frac{N_{test}}{\Gamma(K/2)} \Gamma\left(\frac{K}{2}, \frac{1}{2}\|\Sigma^{-1/2} x_i\|_2^2\right), \tag{3}$$

where Σ represents the covariance matrix of the centered variable X_i. Three assumptions about the feature noise can then be considered: (1)

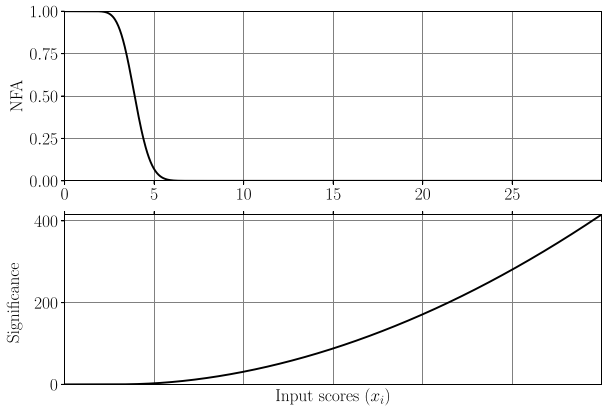

Fig. 2. NFA and *significance* values for a centered unit variance Gaussian variable, with $N_{test} = 1$ for simplicity.

Elliptical distribution with dependent channels: in this case, Σ is a dense positive-definite matrix; (2) Elliptical distribution with independent channels, which leads to $\Sigma = \lambda\Delta$ where Δ is a diagonal matrix with $|\Delta| = 1$ and $\lambda \in \mathbb{R}$; (3) Spherical distribution, leading to $\Sigma = \lambda I_d$ where I_d is the identity matrix. In this particular case, no direction or channel is privileged in the decision process. The impact of these different hypotheses on the training is assessed in Section 4.3.4.

2.1.3. NFA and significance

According to Eq. (1), the NFA values range from 0 to N_{test}. However, for large values of x_i (i.e., when there is a significant response in the feature map), the NFA values tend towards very small values, often lower than 10^{-200}. In order to increase the readability of those values, some authors (e.g. [10]) have proposed to rather consider the *significance* $S(x_i, N_{test}, K, \Sigma)$ defined using a logarithmic scaling:

$$S(x_i, N_{test}, K, \Sigma) = -\ln\left(\mathrm{NFA}(x_i, N_{test}, K, \Sigma)\right). \tag{4}$$

Then, the *significance* associated to Eq. (3) is:

$$S(x_i, N_{test}, K, \Sigma) = -\ln\left(\frac{N_{test}}{\Gamma(K/2)}\Gamma\left(\frac{K}{2}, \frac{1}{2}\|\Sigma^{-1/2}x_i\|_2^2\right)\right). \tag{5}$$

NFA values and their corresponding *significance* are represented on Fig. 2. Note that due to rounding problems to 0, we use the approximation of the $\Gamma(a, x)$ function for $x \to +\infty$ (in practice, for $x > 40$) given in [13]:

$$\Gamma(a, x) \approx x^{a-1}e^{-x}\left(1 + \frac{a-1}{x} + \frac{(a-1)(a-2)}{x^2}\right). \tag{6}$$

2.2. Attention mechanisms

Despite the efficiency of CNNs in extracting meaningful information from an image, the translation invariance induced by convolutions seems to impair the overall understanding of the scene. Attention mechanisms partly circumvent this limitation by imitating the human perception and by dynamically weighting features depending on their relevance to a given final task. Several types of attention mechanisms have been proposed. In our work, we focus on the use of channel and spatial attention mechanisms.

2.2.1. Channel-based attention

Channel-based attention allows us to select the relevant channels in a set of feature maps. This concept was firstly presented in [14], where the authors introduce a squeeze-and-excitation block made of two steps. The first one, called the squeeze step, consists in a reduction in dimensionality while keeping global spatial information. Then, an excitation module allows for learning channel-wise relationships, which gives rise to an attention vector that indicates the weights to apply to the different channels. Several variants have been proposed to overcome SE block shortcomings. For example, [15] propose the Efficient Channel Attention (ECA) block, where they reduce the complexity of the fully-connected layers used in the excitation step by replacing them with a 1D convolution. In the following, we focus on this solution.

2.2.2. Spatial attention

Unlike channel attention, spatial attention is intended to indicate the regions of the image where most attention is needed. This is achieved by modeling long-range dependencies between the different regions of an input. Such attention can be useful for detecting objects that are tiny in one dimension and significantly large in the other (e.g., cracks). Several strategies have been proposed, including training a subnetwork to identify the important regions [16], or increasing the receptive field of CNNs. The methods based on the latest strategy use self-attention mechanisms, which were introduced in computer vision tasks by [17]. They lead to impressive results compared to the performance achieved so far using CNNs, especially when it comes to the use of Vision Transformers (ViT) for various visual tasks [18]. However, this process is computationally very expensive, and it also requires a lot of training data. In addition, for small object detection, the spatial dependencies are mainly local. In this work, we rather consider the use of local self-attention layers, and more specifically the stand-alone self-attention layers proposed by [19].

3. Deep *a contrario* framework

In this section, we present our method for integrating an *a contrario* decision criterion into the training loop of a one-class semantic segmentation NN. Two key ingredients are needed for such module: an *a contrario* block, called NFA block, that backpropagates the gradients, and a specific activation function that allows the NN to learn from the obtained *significance* scores.

3.1. NFA blocks

3.1.1. Basic NFA block

We propose a basic NFA block that transforms multi-channel feature maps into a one-channel score map representing the *significance* defined by Eq. (5). This block is described by Fig. 4a. Two convolution blocks (i.e., 2D convolution with kernel 3×3 followed by batch normalization and ReLU activation) are applied on the input features in order to extract some relevant features for computing the NFA. The *significance* scores are then computed using Eq. (5), where N_{test} is equal to the total number of tested pixels for a given image (i.e., the size of the image). This equation is derivable, allowing for the backpropagation step in the NN.

Our NFA block can replace the segmentation head of any one-class segmentation NN. Its integration on a U-shaped NN is presented in Fig. 3. Note that the assumptions made on the background do not need to be verified. Indeed, in *a contrario* reasoning, the hypothesis made on the background distribution only needs to be contradicted in the presence of any structure of object of interest. It means that the background distribution modeling can be only approximate, as long as the objects of interest fall outside this distribution. Moreover, introducing the *a contrario* criterion into the supervised training loop will guide the network to extract features in a way that object features will be likely to contradict the naive hypothesis (here Gaussian distribution) made on the background.

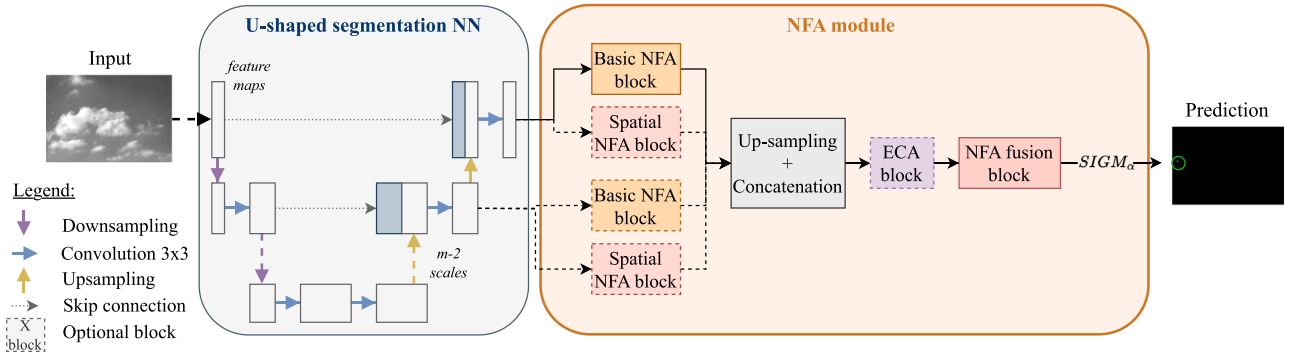

Fig. 3. Diagram showing the integration of our NFA module into a U-shaped segmentation NN. Optional blocks are drawn in dotted lines. Details for ECA block can be found in the original paper [15].

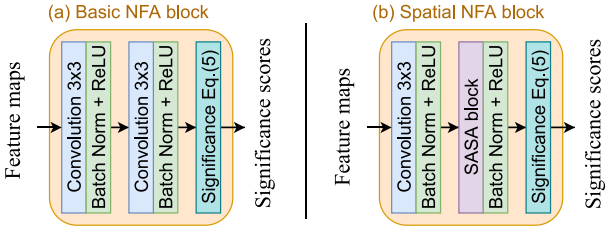

Fig. 4. Diagram of (a) the basic NFA block and (b) the spatial NFA block. The details of the stand-alone self-attention (SASA) block can be found in [19].

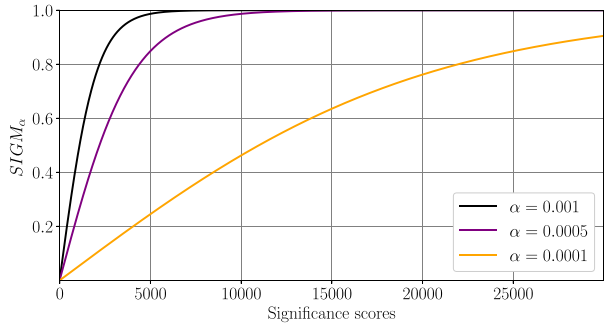

Fig. 5. Variations of SIGM_α function defined in Eq. (8), with different values of α. For simplicity, we choose $N_{test} = 1$.

3.1.2. Multi-scale fusion of significance maps

Many popular segmentation networks rely on encoder–decoder models introduced in [20,21]. The advantage of using U-shaped NN is that we can easily extract low-level semantic feature maps and use the large-scale spatial information they contain for detecting objects of different sizes. Although the highest-level feature maps are the most relevant for segmenting tiny objects, we will see that the feature maps from deeper scales are also useful. These contain rich spatial information and enable the NN to detect targets of different sizes (and therefore not be specific to a single target size), and also to better discriminate targets from potential background false alarms. To do so, we integrate our basic NFA block at each intermediate scale of any U-shaped NN, as illustrated in Fig. 3. Considering a NN with m scales, we perform the detection at each scale and thus obtain m *significance* score maps. Note that considering $m > 1$ scales increases the number of tests, which thus becomes $N_{test} = h \times w \times (1 + \frac{1}{2^2} + \cdots + \frac{1}{2^{2(m-1)}})$, where $h \times w$ is the number of pixels composing the image. In order to merge the detections performed at all scales, the low-level *significance* score maps are upsampled to match the NN input size $h \times w$ using bilinear interpolation. All *significance* maps $S_1, \ldots, S_m$ are then merged together through the NFA fusion block by taking the union of all detections. This leads to the final *significance* score map S_{final}, defined for each pixel i as follows:

$$S_{final}(i) = \max\{S_1(i), \ldots, S_m(i)\}. \tag{7}$$

However, with such a multi-scaling strategy, the detections from the lower and higher resolution scales have the same weight in the final *significance* score map, which may increase the false alarm rate for applications where coarse scales are less relevant. We thus propose to dynamically weight the impact of the different scales by learning weighting coefficients using a channel attention module. The integration of an ECA block [15] before merging the *significance* maps is illustrated on Fig. 3.

3.1.3. Spatial NFA block

The basic NFA block defined in Section 3.1.1 is designed to improve the detection of tiny objects that do not present a specific geometric structure (e.g., point-shaped objects). However, for objects that are small only in one dimensionality and significantly large in other dimension(s) (e.g., cracks), spatial information is a discriminating feature. Indeed, in the case of crack detection, several pixels forming a continuous line are more likely to belong to a crack than a few isolated pixels. It is therefore necessary to extend the NN receptive field in order to better take into account the information from more distant pixels. To improve performance on such objects, we design a second version of our NFA block that includes spatial attention mechanisms. Fig. 4b shows this block, where the second convolution layer is replaced by a stand-alone self-attention (SASA) layer [19]. As shown in Fig. 3, if we add a spatial NFA block, it is done in addition to a basic NFA block.

3.2. NFA-friendly activation function

The NFA block output is a *significance* score map whose distribution of scores is not only asymmetric between positive and negative values, but also has a much wider dynamic than conventional NN output. Indeed, as explained in Section 2.1.3, the background values are expected to be pushed towards $-\ln(N_{test}) \leq 0$, while the object values are expected to be spread over the interval $(-\ln(N_{test}), +\infty)$. Consequently, the conventional symmetric activation functions, such as the sigmoid function, are not suitable. This has been confirmed by the experiments presented in Section 4.3.4. Therefore we rather design the following activation function:

$$\text{SIGM}_\alpha(x, N_{test}) = \frac{2}{1 + e^{-\alpha(x + \ln(N_{test}))}} - 1, \tag{8}$$

where $\alpha \in \mathbb{R}$ is a parameter that allows us to control the slope of the sigmoid. We represent the variations of SIGM_α function on Fig. 5 for different values of α. This activation function strongly penalizes the background values while compressing progressively the object values, thus respecting the dynamics induced by the computation of the *significance*. Note that the higher the value of the parameter α, the more

the dynamic of the *significance* scores will be non linearly compressed. The sensitivity of the NN training to this parameter is studied in Section 4.3.4. This activation function is applied after having combined all the *significance* maps obtained from the NFA blocks computed at different scales, as shown on Fig. 3. The final output scores, therefore, range between 0 and 1, which allows the user to apply any cost function that is suitable for one-class segmentation tasks.

Nevertheless, substituting the conventional segmentation head in a segmentation NN by the NFA module makes the threshold usually used to binarize the segmentation map (namely, 0.5) no longer suitable, as background values are constrained to 0 after applying the function SIGM_α and even low output values can be significant. Thus we have to derive the new detection threshold. One argument often given in favor of *a contrario* approaches is the interpretation of the NFA and thus the more or less direct choice of the threshold (nevertheless application dependent). In our case, the segmentation threshold t is linked to the ϵ threshold defined in Eq. (2) through Eq. (8):

$$t = \text{SIGM}_\alpha(-\ln(\epsilon), N_{test}), \tag{9}$$

where ϵ represents the average number of false alarms on background images, at pixel level, that we can tolerate for our application. In the literature, ϵ is chosen in the interval $[10^{-200}, 1]$, leading to a thin threshold interval for the value t, namely $[10^{-3}, 0.12]$. We will discuss a more refined choice of this theoretical threshold for each considered application, depending on our tolerance for false alarms.

In the following of the paper, we consider different backbones and evaluate the benefits of our NFA module on three applications, namely small target detection, road crack detection and ship detection. Note, however, that the former has been studied more extensively, and the latter two have been considered mainly to test the possible generalization of the NFA module to other applications (and backbones).

4. Application to small target detection

We first evaluate the contribution of our NFA module in the case of infrared small target detection. This application constitutes an ideal framework for the detection of tiny objects: the targets have a surface area of only a few pixels, are not very contrasted compared to the background and do not present a specific structure. Most proposed methods to tackle this problem use semantic segmentation NN [22] rather than off-the-shelf detection NN [1]. SOTA NN for small target detection rely on U-shaped architectures and include spatial attention mechanisms [23–25].

4.1. Assessed methods

We propose to integrate our NFA module into one of the U-shaped SOTA backbones. We select the recent DNANet [23], which has shown impressive performance on widely used small target detection datasets. DNANet is composed of two parts: a dense-nested U-shaped backbone (DNIM), which allows for the feature extraction step, and a feature pyramid fusion module (FPFM), which allows for a multi-scale fusion of intermediate outputs from the backbone. We substitute the FPFM block with our NFA module, and we evaluate its contribution with respect to the backbone DNIM (ResNet-18 version) and DNANet. We also extend our experiments to the use of a classical backbone, namely ResUNet [26], to show the generalization of our method to another backbone that is not specifically designed for small target detection.

For our NFA module, we set the α parameter in Eq. (8) to 0.0005, as it has shown to lead to the best results in Section 4.3.4. To guide the selection of the binarization threshold, let us remind that the considered application handles detections at object level (the first parameter of interest is the number of detected targets and then their localization, speed etc.). Thus, the impact of one-pixel false alarm is completely different whether it is isolated or connected to a detection, since it will or not affect the number of detected targets. The strong constraint to absolutely avoid such errors implies a very low tolerance for false alarms at pixel level. In the literature, a low NFA in the case of the *a contrario* approach lies around $\epsilon \approx 10^{-200}$, leading to a binarization threshold $t \approx 0.1$. This value has been confirmed on a validation set and we kept it unchanged for every experiment of this application. For the baselines, the detection threshold is set to 0.5 as suggested in the original paper. All networks are trained from scratch[1] on Nvidia RTX6000 GPU for 1000 epochs using the Soft-IoU loss function [27]. The latter is optimized by Adagrad optimizer with the Cosine Annealing scheduler, using the same parameters as in [23]. The learning rate is set to 0.05 for DNIM and DNANet as suggested in the original paper. For DNIM+NFA, we found that decreasing the learning rate allows for better convergence; we thus set it to 0.03.

4.2. Dataset and evaluation metrics

We conduct our experiments on two datasets. We first consider NUAA-SIRST dataset [28], which is one of the few infrared small target datasets publicly released and widely used in the literature. This dataset contains 427 images, and most targets follow the definition of a small target proposed by Society of Photo-Optical Instrumentation (SPIE), that is objects having a total spatial extent of less than 80 pixels (9×9) [29]. We also consider a recently published dataset for small target detection, namely IRSTD-1k [24]. This dataset is larger (1000 images) and contains more challenging scenes, with different kinds of small objects (e.g., aircrafts, animals). It also contains some very large objects, which fall outside the scope of our method (designed for tiny object detection, cf. Section 6). We therefore remove images that contain targets having a spatial extent larger than 90 pixels (this represents 15% of the dataset), and refer to the filtered dataset as "IRSTD-850". We discuss the behavior of our method on larger objects in Section 6.

Both datasets are split into training, validation and test sets using a ratio of 60 : 20 : 20. We use the same pre-processing steps as those proposed in [23]. For the evaluation, we mainly focus on object-level metrics as suggested by [23]. From the predicted binary segmentation map, targets are individually labeled using a 8-connectivity connected component module. A detected object is counted as a true positive (TP) if it has an Intersection over Union (IoU) of at least 5% with the ground truth. This low-constrained condition is due to the fact that a small shift in the number of predicted pixels leads to a large deviation in the IoU. We then compute the Precision (Prec.), Recall (Rec.) and F1 score (F1) at object-scale. We also consider the area under the object-level Precision–Recall curve, namely Average Precision (AP), which allows us to free from the detection threshold, and the number of false alarms (still at object-level) per image (FA/image).

In the tables, the presented results have been averaged over five distinct training sessions and they are given in the form $\mu \pm \sigma$, where μ is the mean and σ the standard deviation.

4.3. Results

4.3.1. NFA module improves the precision

Results obtained with SOTA backbones - Table 1 shows the performance for the three compared methods that are based on DNIM backbone, on NUAA-SIRST and IRSTD-850 datasets. On both datasets, DNIM+NFA leads to a significant improvement of the baseline DNIM in both AP and F1. For example, the F1 score is increased by 1.8% on NUAA-SIRST dataset and by 2.3% on IRSTD-850. More specifically, since the NFA layer controls the number of false alarms, the precision appears significantly improved, while keeping the number of correctly

[1] We used the official implementation of DNANet https://github.com/YeRen123455/Infrared-Small-Target-Detection.

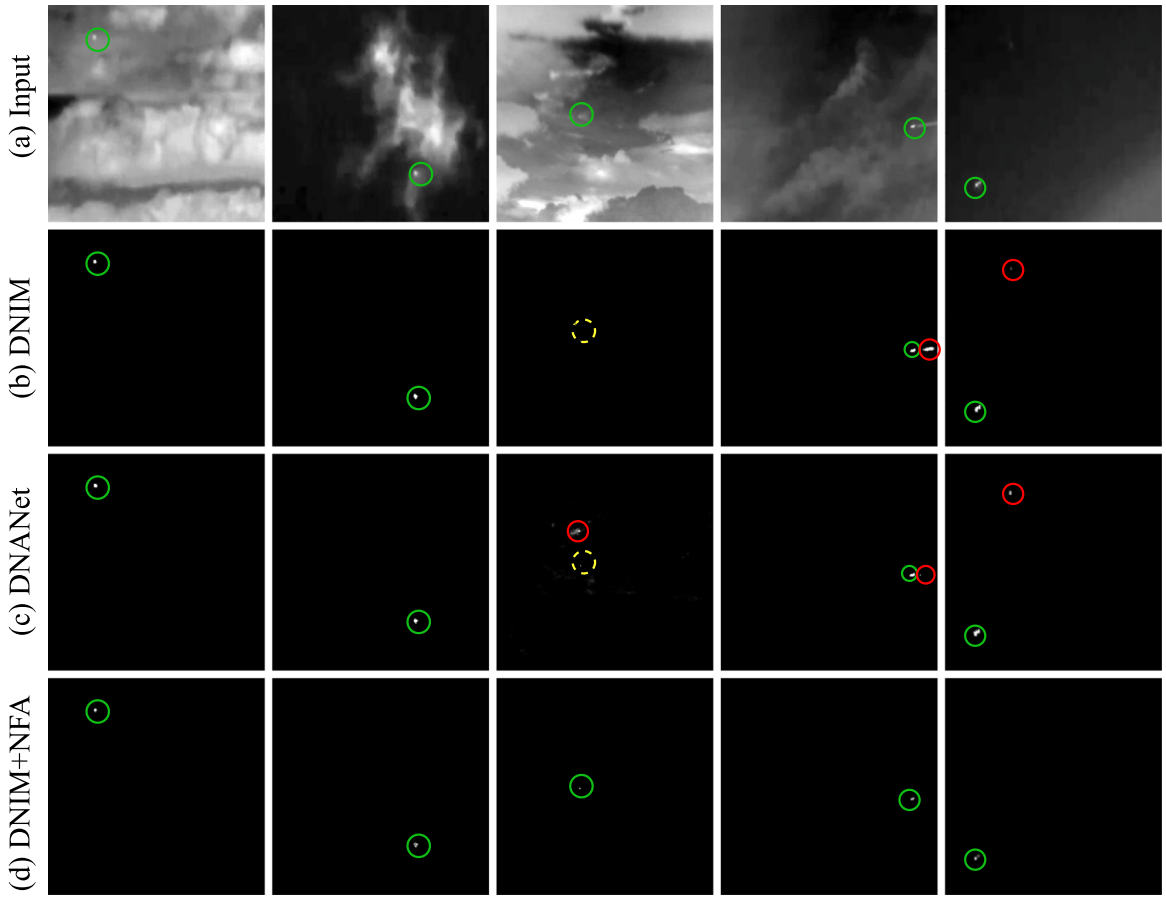

Fig. 6. Qualitative results obtained with different detection methods (columns (b) to (d)) on NUAA-SIRST dataset. Good detections, false positives and missed detections are circled in green, red and dotted yellow lines respectively.

Table 1
Object-level F1 (%), AP (%), Prec. (%), Rec. (%), and FA/image achieved by the compared methods on NUAA-SIRST and IRSTD-850. Best results are in bold and second best results are underlined.

Method	F1	AP	Prec.	Rec.	FA/image
NUAA-SIRST dataset					
DNIM	$95.8^{\pm 1.3}$	$96.2^{\pm 1.3}$	$94.6^{\pm 2.4}$	$97.1^{\pm 0.}$	$0.06^{\pm 0.03}$
DNIM + FPFM (DNANet)	$\underline{97.1}^{\pm 0.4}$	$\mathbf{98.4}^{\pm 0.9}$	$\underline{96.9}^{\pm 1.2}$	$\underline{97.3}^{\pm 0.4}$	$\underline{0.04}^{\pm 0.02}$
DNIM + NFA (Ours)	$\mathbf{97.6}^{\pm 0.3}$	$\mathbf{98.4}^{\pm 0.6}$	$\mathbf{97.9}^{\pm 0.1}$	$\mathbf{97.4}^{\pm 0.6}$	$\mathbf{0.02}^{\pm 0.00}$
IRSTD-850 dataset					
DNIM	$89.0^{\pm 1.4}$	$89.9^{\pm 1.6}$	$87.6^{\pm 2.5}$	$90.5^{\pm 0.8}$	$0.20^{\pm 0.05}$
DNIM + FPFM (DNANet)	$\mathbf{91.4}^{\pm 1.4}$	$\underline{92.4}^{\pm 1.9}$	$\underline{91.8}^{\pm 2.4}$	$\mathbf{91.1}^{\pm 0.9}$	$\underline{0.13}^{\pm 0.04}$
DNIM + NFA (Ours)	$\underline{91.3}^{\pm 0.7}$	$\mathbf{94.2}^{\pm 0.2}$	$\mathbf{92.1}^{\pm 0.3}$	$\underline{90.6}^{\pm 1.5}$	$\mathbf{0.12}^{\pm 0.00}$

detected targets (recall criterion) at the same level. This improvement in precision is all the more impressive on the challenging IRSTD-850 dataset (+4.3% in AP). Note that the addition of the NFA module in DNIM greatly improves the stability of the training, as evidenced by the decrease in the standard deviation of the results. DNIM+NFA is also very competitive with SOTA method DNANet. Indeed, on NUAA-SIRST dataset, the F1 score is better in average (+0.5%), and it can be noticed that the number of false alarms per image has been divided by 2, while having a better recall. The standard deviation is also reduced. On IRSTD-850 dataset, although the F1 scores are equivalent for both methods, the AP is significantly improved by DNIM+NFA (+1.8%). This confirms the benefit of our NFA module, especially on the control of the false alarm rate even in scenes with complex backgrounds. Furthermore, as far as computation costs are concerned, the NFA layer adds less than 0.1 million training parameters to the initial model, which is negligible with respect to the benefits deriving therefrom.

Table 2
Comparison of ResUNet and ResUNet + NFA on small target detection. Metrics are computed at object-level and averaged over three runs.

Method	NUAA-SIRST		IRSTD-850	
	F1	AP	F1	AP
ResUNet	$93.2^{\pm 0.9}$	$90.3^{\pm 2.4}$	$85.3^{\pm 1.2}$	$87.4^{\pm 1.2}$
ResUNet + NFA	$\mathbf{95.4}^{\pm 1.3}$	$\mathbf{96.1}^{\pm 1.9}$	$\mathbf{87.7}^{\pm 2.7}$	$\mathbf{96.0}^{\pm 0.8}$

Fig. 6 illustrates some predictions (output score maps before threshold) on challenging scenes, where the contribution of the NFA module can clearly be seen. For example, the target of the third column is particularly small and blurred in the background, which does not affect the performance of the NFA module, unlike the other methods. Moreover, the baseline methods mistakenly detect the aircraft contrail (fourth column). The NFA module not only allows for better detection of small and tiny objects in particularly difficult scenes, but also provides robustness with respect to challenging environments.

Generalization to conventional backbones - We extend the experiments conducted previously to another conventional NN, namely ResUNet. We evaluate the benefits of our NFA module for a NN that is not specifically designed for small target detection. Results are presented in Table 2, and it can be seen that the NFA module greatly improves global performance. Indeed, on NUAA-SIRST dataset, the F1 score is improved by 2%, and the AP by 6%, which is mainly explained by an increase in average precision as observed for DNIM+NFA. This improvement in precision is even more striking when considering IRSTD-850 dataset (+8.6% in AP). This confirms that adding an NFA module on a segmentation network (being SOTA or not) improves the precision and thus the performance. Furthermore, the results obtained

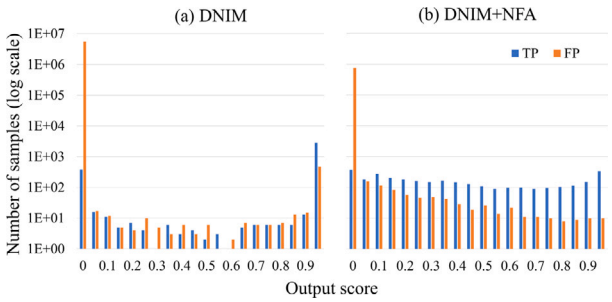

Fig. 7. Output scores histograms for (a) DNIM and (b) DNIM+NFA.

Table 3
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
DNIM	$72.8^{\pm 23.7}$	$68.0^{\pm 31.3}$	$87.0^{\pm 2.5}$	$82.6^{\pm 2.7}$
DNIM + NFA	$\mathbf{87.7}^{\pm 2.9}$	$\mathbf{86.3}^{\pm 3.9}$	$\mathbf{90.9}^{\pm 2.7}$	$\mathbf{93.1}^{\pm 2.0}$

Table 4
Transfer learning from SIRST to IRSTD-850.

Method	F1	AP
DNIM	83.4	83.6
DNIM + NFA	**84.9**	**91.2**

when adding the NFA module on a conventional segmentation NN are only few percents lower than what can be obtained by a SOTA backbone specifically designed for small target detection. For example, the difference in F1 score between ResUNet+NFA and DNIM is only of 0.4% on NUAA-SIRST. This shows that although the careful design of the feature extractor is essential to improve performance, the choice of decision criterion is also very important, especially in the case of small/tiny object detection.

4.3.2. Overconfidence, did you say?

Most recent neural networks tend to be overconfident as outlined in [30]. The pixel-level histogram of output scores shown on Fig. 7a illustrates this phenomenon for DNIM network, where all pixel values on the final score map are either very close to 0 or to 1. The impact of NFA layer can clearly be seen on the corresponding histogram in Fig. 7b: TP are uniformly spread all over the confidence scores and the number of false positives (FP) decreases monotonically as the score level increases. Fig. 8 illustrate the relationships between accuracy and output scores (interpreted as confidence values). When comparing Fig. 8a and b, we see that the achieved scores are more informative since the accuracy versus score function is globally increasing. Besides, we notice that the NFA module prevents the network from being overconfident. To better calibrate the DNIM+NFA outputs, we can play *a posteriori* with the parameter α from Eq. (8). The value of α to get a calibrated network can be found by solving $\text{SIGM}_{\alpha}(10^{-200}, N_{test}) = 0.5$ (since for a calibrated output optimal segmentation threshold should be equal to 0.5), so that $\alpha = 0.003$. Fig. 8c illustrates the results after calibration using $\alpha = 0.003$: adding the NFA module to a segmentation NN allows us to obtain a nearly calibrated network without the need of complex methods. The output scores are also relevant, which is a step towards AI interpretability.

4.3.3. Robustness analysis

In this subsection, we assess the robustness of our method compared to the baseline DNIM in two scenarios: weak training conditions and generalization to new or noisy data.

Few-shot learning - In many real world applications, data collection and annotation requires expertise, which is very expensive and time consuming. Having a method that leads to good performance even with little training data is essential in such real world applications. In the subsection, we evaluate the robustness of our method in few-shot settings, by training the NN on 15 and 25 images from NUAA-SIRST dataset (representing respectively about 5% and 10% of the training set used in Section 4.3.1). DNIM and DNIM+NFA are trained on three different non-overlapping sets of data in both 15-shot and 25-shot settings, and the averaged results are given in Table 3. It can be seen that our method performs significantly better in a frugal setting than the baseline. Indeed, both AP and F1 metrics are increased by more than 15% when adding the NFA module to DNIM in a 15-shot training. Moreover, the AP is decreased by only 5.3% for DNIM+NFA (compared to 13.6% for the baseline) when dividing by 10 the number of training samples. The robustness of the NFA module towards frugal setting is explained by the *a contrario* paradigm introduced in the training loop: we force the NN to model the background elements (rather than the targets themselves), for which we have sufficient samples even in a few-shot setting.

Generalization to noisy and new data - One essential property of strong detectors is their ability to correctly generalize to unseen data. To this end, we first evaluate the robustness of DNIM+NFA towards noisy data during the inference. We consider two types of noise: additive and multiplicative Gaussian noises, with different variances (namely 0.01, 0.05 and 0.10). For the additive Gaussian noise the mean is set to 0 while for the multiplicative one it is set to 1. As we can see from Fig. 9, although F1 score and AP decrease with the increase in variance for both methods, DNIM+NFA still achieves the best performance by a large margin, for both considered types of noise. It is also significantly robust towards false alarms (AP criterion) compared to DNIM, especially in the case of additive Gaussian noise (Fig. 9(a)).

We finally evaluate the methods on new scenes by transferring the knowledge learned on NUAA-SIRST dataset to IRSTD-850 dataset, without fine-tuning. The results in Table 4 confirm the generalization ability of our method on new challenging scenes. Compared to the baseline, the F1 score is increased by 1.5%, and the AP by 7.6%. This robustness to new or noisy data is explained by the use of a naive model, which can only be approximate (provided that it contradicts the detections).

Table 5
Ablation study performed on NUAA-SIRST. We evaluated (object-level metrics) the different forms of the covariance matrix Σ and compared the benefits of multi-scaling (MS), adding a regularization term (Reg.) and using channel attention (ECA) in our NFA module.

Σ (Eq. (3))	MS	ECA	Reg.	F1	AP
$\Sigma = \lambda I_d$			✓	$96.0^{\pm 0.9}$	$97.6^{\pm 0.9}$
Dense Σ			✓	$95.1^{\pm 1.3}$	$96.3^{\pm 1.0}$
$\Sigma = \lambda\Delta$			✓	$96.9^{\pm 0.5}$	$\mathbf{98.6}^{\pm 1.0}$
$\Sigma = \lambda\Delta$	✓			$\underline{97.2}^{\pm 0.6}$	$95.6^{\pm 2.3}$
$\Sigma = \lambda\Delta$	✓		✓	$\underline{97.2}^{\pm 0.6}$	$97.9^{\pm 0.9}$
$\Sigma = \lambda\Delta$	✓	✓	✓	$\mathbf{97.6}^{\pm 0.3}$	$\underline{98.4}^{\pm 0.6}$

4.3.4. Ablation study

Tables 5, 6 and 7 present the ablation and sensitivity studies performed on small target detection on NUAA-SIRST dataset. The conclusions are summarized in the five following points.

(a) Assumptions made on the covariance distribution - In Section 2.1.2, we present three different forms for the covariance distribution Σ in Eq. (3), corresponding to three different assumptions about feature noise: spherical distribution, elliptical distribution with independent channels or components, elliptical distribution with dependent channels. The first two lines of Table 5 show that assuming a spherical distribution assumption leads to worse results, as does the channel-dependence assumption. To explain this, one has to remind that in deep learning, in order to disentangle causal factors, a series

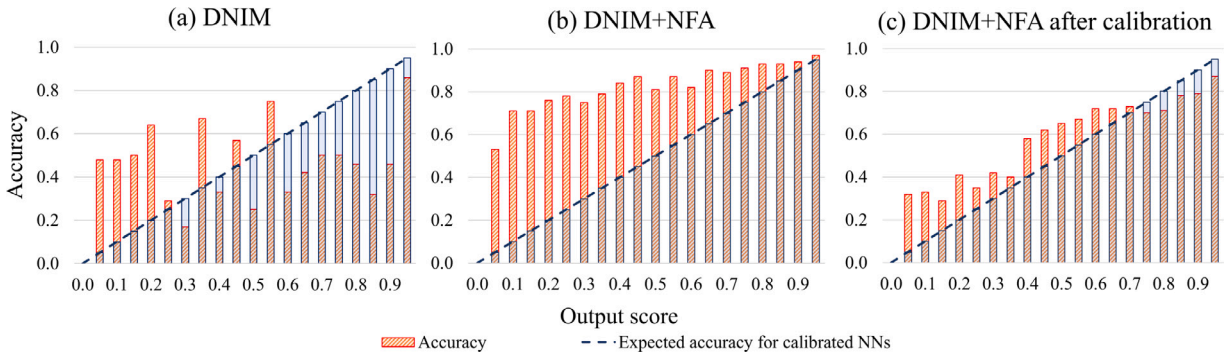

Fig. 8. Variations in accuracy as a function of output scores for (a) DNIM, (b) DNIM+NFA with $\alpha = 0.0005$, and for (c) DNIM+NFA after calibration using $\alpha = 0.003$.

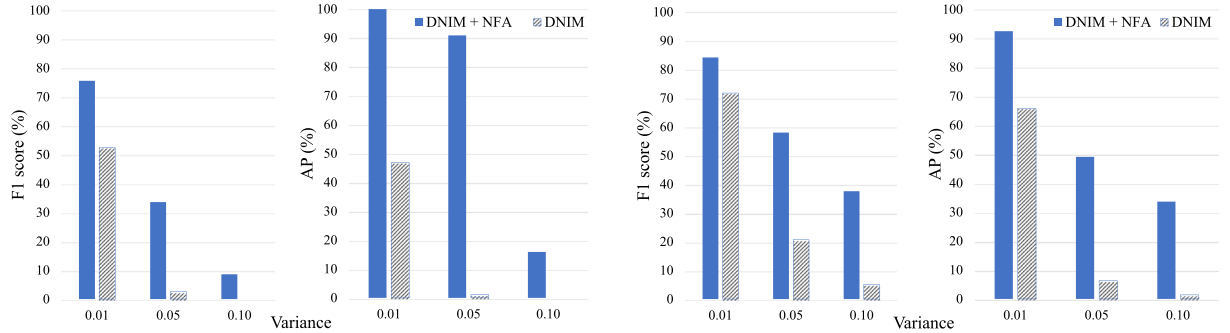

Fig. 9. Sensitivity of DNIM and DNIM+NFA towards noisy images from NUAA-SIRST during inference.

of filters are applied to extract the relevant characteristics. Each filter extracts a particular feature represented by a channel in the next feature maps. Depending on the downstream task, some features will be more or less relevant. The relevant information is therefore not equally distributed over all the channels of the feature maps. Besides, estimating the full covariance matrix in high dimensionality may be numerically unstable, while the correlation between extracted features remains low. As a result, the independent elliptical distribution appears as the more relevant hypothesis.

(b) Adding a regularization term prevents from object fragmentation - According to Table 5, gradient regularization, which is defined as the L2 norm of the gradient of the image in both vertical and horizontal directions, improves AP criterion. Indeed, it allows us to force low value difference between neighboring pixels, and thus to avoid object fragmentation when increasing the segmentation threshold. The achieved results are then more robust to this threshold, which increases the AP.

(c) Importance of multiscaling - Adding information from low-level features helps the network in detecting objects of various size. This is the case for the baseline DNIM, whose multiscale version is DNANet, and it has been confirmed when adding NFA layers to the 5 scales of DNIM and considering the F1 criterion in Table 5. However, F1 increase is at the cost of a decrease of the AP criterion since introducing low-scale features may bring out more false positives for lower thresholds. We also performed an ablation study on the number of scales used in the NFA fusion block. We varied the parameter m in Eq. (7) from 2 to 5, 5 being the maximum number of scales in DNIM. The results presented in Table 6 show the importance of considering all the 5 scales, even for detecting small targets. Indeed, the F1 score increases gradually from 96.0% to 97.6% as more scales are added, and the AP reaches its maximum when considering 5 scales. The standard deviation also decreases when adding more scales, meaning that the NN gains in stability.

Table 6
Ablation study on the number of scales m in Eq. (7).

m (Eq. (7))	F1	AP
2	$96.0^{\pm 0.8}$	$96.7^{\pm 2.0}$
3	$96.5^{\pm 0.8}$	$98.3^{\pm 0.7}$
4	$97.6^{\pm 0.3}$	$97.8^{\pm 1.0}$
5	$97.6^{\pm 0.3}$	$98.4^{\pm 0.6}$

Table 7
Sensitivity study made on the activation function. Metrics are given at object-level.

Activation function	F1	AP
Sigmoid	$90.5^{\pm 6.3}$	$93.5^{\pm 1.2}$
$\text{SIGM}_{\alpha=0.0001}$	$96.4^{\pm 1.4}$	$95.1^{\pm 0.1}$
$\text{SIGM}_{\alpha=0.0005}$	$\mathbf{97.2}^{\pm 0.6}$	$\mathbf{97.9}^{\pm 0.9}$
$\text{SIGM}_{\alpha=0.001}$	$96.7^{\pm 0.2}$	$94.8^{\pm 1.0}$

(d) Channel attention highlights the importance of high-level scales for small target detection - To tackle previous issue, we introduced a channel attention layer before merging the different scales, that is, ECA block. Table 5 clearly shows the superiority of the NFA module when adding this step. It noticeably improves the average precision as well as the F1 score, by reducing the object false alarm rate. Looking at the multiplying factors computed by this channel attention layer, we observe that, for small target detection, the high-level features are of primary importance: their weight is about 0.99 when the weight of lower-level feature maps is about 20 times less, though they still contribute to the decision.

(e) Appropriate activation function - To confirm that conventional symmetric activation functions such as the sigmoid function are not suitable for the *significance* values, Table 7 shows the result obtained considering the sigmoid activation function, which indeed severely degrades the F1 score and AP of our method. The results are

Table 8
Comparison of ResUNet and ResUNet + NFA on crack and ship detection. Metrics are computed at pixel-level for crack detection, and at object-level for ship detection.

Method	CrackTree		S2SHIPS	
	F1	AP	F1	AP
ResUNet	$85.6^{\pm 0.4}$	$85.2^{\pm 0.2}$	$23.7^{\pm 2.0}$	$52.3^{\pm 6.4}$
ResUNet + NFA	$\mathbf{87.2}^{\pm 0.0}$	$\mathbf{96.7}^{\pm 0.2}$	$\mathbf{35.3}^{\pm 1.5}$	$\mathbf{62.3}^{\pm 7.9}$

also less stable across different weight initializations, as shown by the large standard deviation values (more than 6% in F1 score).

Now, for the proposed activation SIGM_α, as discussed in Section 3.2, the choice of α has an impact on the range of optimal thresholds for score map binarization. We tested three values of α, which moves the upper bound for thresholds from 0.02 to 0.3. According to Table 7, $\alpha = 0.0005$ leads to the best performance, and we recommend to use this value.

5. Extension to other applications

We have shown in previous section that the NFA module can improve the performance of a segmentation NN specifically designed for small target detection. This allowed us to obtain state-of-the-art results on an application that represents an ideal framework for small object detection. Now, we propose to expand the boundaries of previous framework. For this purpose, we integrate our method in a classical semantic segmentation backbone and we apply it to two other applications, namely road crack detection and ship detection from remote sensing data. Both applications deal with small object detection in a frugal setting, and they are challenging for several reasons. In the case of road crack detection, the difficulty lies in the fact that (i) the cracks are very thin and their pixels are very few with respect to the background class, and (ii) the textured background and road artifacts can lead to numerous false alarms. Some generic deep learning approaches have been tested on this application, and are mainly based on classical segmentation NN [31–33]. Ship detection from low resolution satellite imagery is even more challenging because of (i) the large number of boats in the same area and their varying sizes (e.g., pleasure boats, cargo ships), (ii) the moored or tiny ships that are very hard to distinguish from decks or water wings, and (iii) the low resolution of satellite data, which requires to use subpixel information. Most efficient methods for ship detection rely on data fusion (e.g., using SAR data, or the informations provided by automatic identification systems (AIS) [34]). Few deep learning methods for detecting ships from optical data have been proposed, and these detectors mainly rely on classical segmentation NN [35].

5.1. Assessed methods

We take as a baseline classical segmentation backbone, namely a Unet with a ResNet encoder (ResUNet, [26]). Note that, for crack detection, geometric information is crucial since the cracks exhibit a specific shape. Therefore, we take the opportunity given by crack detection to evaluate the contribution of the spatial NFA block in the NFA module. Based on the ablation study conducted in Section 4.3.4, we use the multi-scale NFA module with $\Sigma = \lambda\Delta$ (Eq. (3)) and set the parameter α in Eq. (8) to 0.0005. For both crack and ship detection, the theoretical threshold can be defined as follows. In the case of an image without cracks, one false alarm at a pixel level will not be significant for the application: indeed, a crack is defined by several hundred pixels. The same reasoning can be applied for boat detection as in the considered dataset there are many ships, including cargo ships that have a spatial extent of several hundred pixels. It is therefore reasonable to tolerate one pixel false alarm per image, which makes the false alarm expectation $\epsilon = 1$, leading to a binarization threshold $t \approx 0.001$. For a fair comparison with the baseline, whose optimal threshold no longer seems to be 0.5, we choose the threshold for the baseline based on the validation dataset. Both methods are trained for 700 epochs using the same loss and optimizer as in Section 4.2. ResUNet is trained with a learning rate of 0.01, and we lower the learning rate for ResUNet+NFA to 0.005.

5.2. Datasets and evaluation metrics

Crack detection - We train and evaluate all methods on Crack Tree dataset from [36]. It is composed of 206 real pavement images, and it includes various types of cracks. Because very few data is available, the algorithms are trained using 120 images only. This frugal setting adds some challenge to the application. Finally, 36 images are used for the validation step, and 50 for testing. All methods are evaluated using pixel-level metrics, namely F1 score and average precision. However, as stated in [36], the annotations do not accurately report crack thickness. Therefore, like in the original paper, we adopt a tolerance margin of 2 pixels in crack localization.

Ship detection - We consider the dataset S2SHIPS [35], which is composed of 16 multispectral images from Sentinel 2 satellite sensor, of size 1783×938 pixels. Four images are kept for test, and the others are used for the train and validation datasets. From each image we extract 18 patches of size 256×256, which makes a total amount of 216 patches for the training and validation sets. We use the following six spectral channels as in [35]: B2 (B), B3 (G), B4 (R), B8 (NIR), B11 and B12 (SWIR). Note that in this application, training conditions are particularly difficult: there is very little training data, much of which does not include ships. The assessed methods are evaluated using F1 score and average precision computed at object-level.

5.3. Results

5.3.1. NFA module leads to better performance

Table 8 shows the performance of the evaluated methods on Crack Tree and S2SHIPS datasets. It is clear that the NFA module contributes in improving the baseline. Indeed, in the case of crack detection, the F1 score is increased by 1.4% when including the NFA module in the baseline. More precisely, we observe a very significant improvement in the average precision (more than 10%), which confirms the ability of the NFA module to control the number of false alarms at a pixel level. We notice that the recall is also improved. Fig. 10 shows some results obtained by the different methods on four different crack examples. The NFA module appears more robust to the presence of shadows or textures on the road, since less false alarms are observed.

The same conclusions can be drawn for ship detection: both F1 score and AP are increased by at least 10%. However, despite a significant improvement of the baseline thanks to the NFA module, the performance remains weak: this is explained by the challenging conditions described in Section 5.2 and illustrated on the first row of Fig. 11. Indeed, the presence of decks, coastlines, or even ship wakes leads to several false alarms. Nonetheless, even though both algorithms struggle to detect the most tiny ships, ResUNet+NFA considerably increases their detection as it can be seen on the first image. These experiments on two different applications confirm once more the robustness towards challenging conditions brought by our NFA module.

5.3.2. Contribution of attention mechanisms

We have evaluated the contribution of the different attention mechanisms, spatial attention and channel one separately, on crack detection. The results are detailed in Table 9 and our conclusions are summarized in the following points.

(a) All scales are equally important for crack detection - In crack detection application, unlike in small target one where the decision process mainly relies on the high-level feature map (cf. Section 4.3.4), the deeper level feature maps almost equally contribute to the prediction (multiplying factors all around 0.6). Indeed, the low

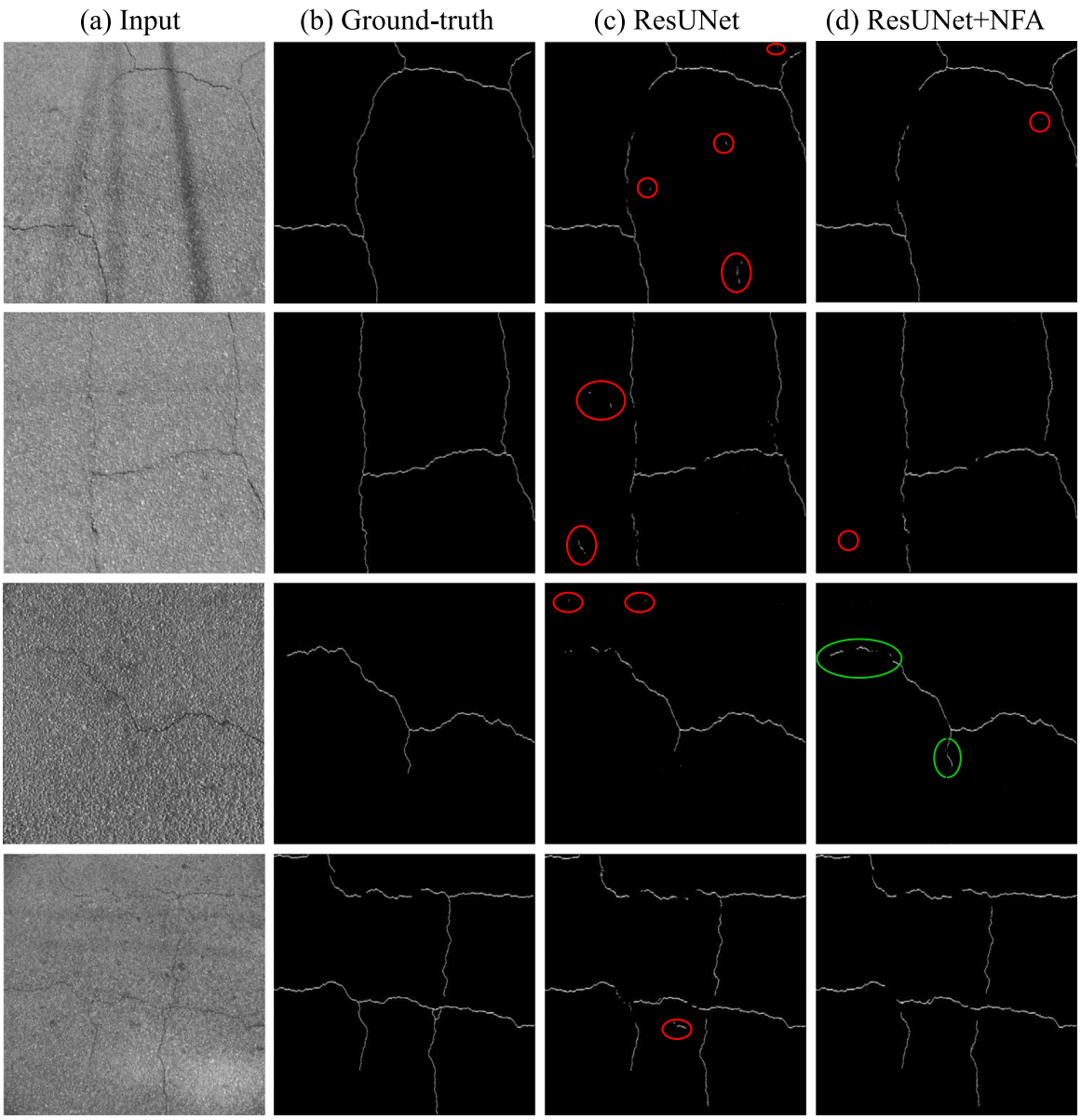

Fig. 10. Qualitative results obtained with different detection methods on Crack Tree dataset. False positives are circled in red, and reconstruction improvements are circled in green.

Table 9
Ablation study performed on Crack Tree dataset (pixel-level metrics).

ECA	SASA	F1	AP
		$86.4^{\pm 0.1}$	$95.8^{\pm 0.2}$
✓		$\underline{87.0}^{\pm 0.3}$	$96.4^{\pm 0.1}$
	✓	$\underline{87.0}^{\pm 0.3}$	$\mathbf{96.8}^{\pm 0.2}$
✓	✓	$\mathbf{87.2}^{\pm 0.0}$	$\underline{96.7}^{\pm 0.2}$

resolution feature maps contain some useful information to describe large objects, while the high-level feature maps are meant for capturing the smaller details as outlined in [3].

(b) Spatial attention has a very significant impact on performance - As expected, the spatial attention block (SASA block) helps detecting precisely large objects. Indeed, thanks to spatial attention, the average precision is considerably improved: the shape of the cracks is estimated in an accurate way while eliminating some false positives.

Finally, combining both spatial and channel attention leads to even better and more stable results.

6. Discussion

Sections 4 and 5 show experimentally the benefits of our NFA module for tiny object detection. It significantly improves the performance of a conventional segmentation backbone such as ResUNet in three challenging applications, namely small target detection in infrared images, road crack detection in usual RGB images, and detection of ships from multispectral satellite images. Furthermore, we have shown that, when our NFA module is added on top of a backbone specifically designed for small target detection, such as DNIM, it is very competitive with the SOTA NN for small target detection DNANet, while being more interpretable. This improvement is due to the introduction of the *a contrario* paradigm in the training loop. The NN is forced to learn an approximate background model rather than the objects to be detected, by computing a number of false alarms (NFA). This gives new properties to the NN that includes such *a contrario* criterion: (i) the control of the number of false alarms, which translates into a clear improvement of the AP criterion, and (ii) the ability to learn from few samples of the object to be detected. The latter property increases the

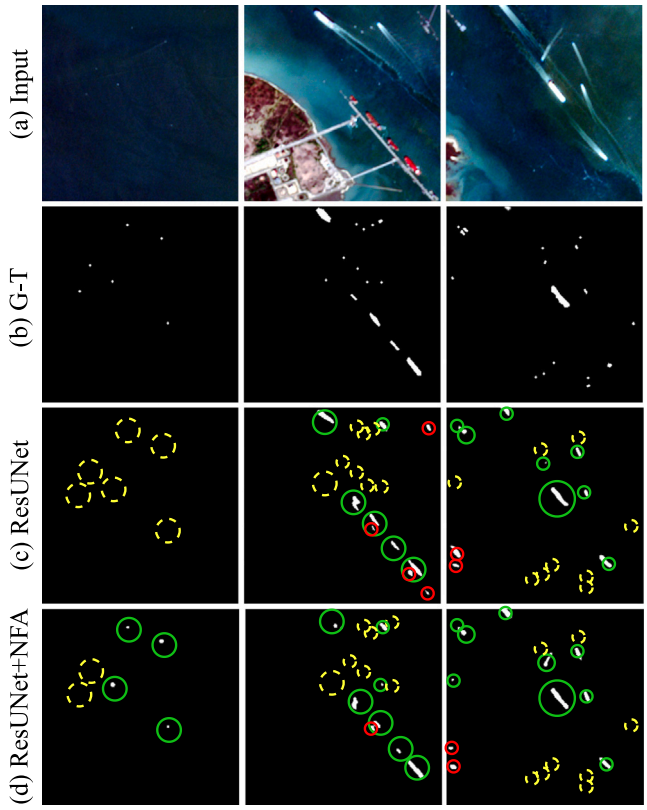

Fig. 11. Qualitative results obtained with ResUNet and ResUNet+NFA on S2SHIPS dataset. True positives, false positives and missed detections are circled in green, red and dotted yellow lines, respectively.

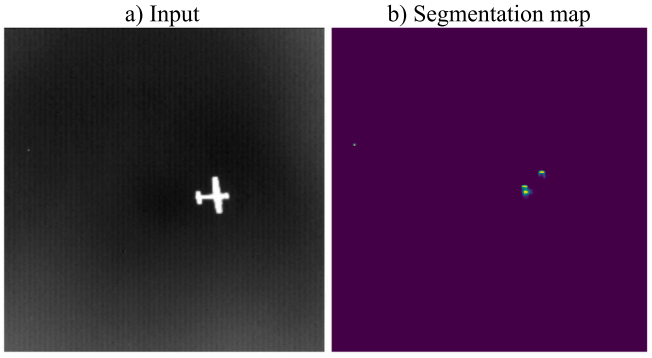

Fig. 12. Behavior of DNIM+NFA on large objects.

robustness of the NN to frugal learning and helps to better generalize to unseen data, as shown experimentally in Section 4.3.3.

The use of our method can be extended to other small object detection tasks such as the early medical diagnosis, early forest fires detection, or the detection of buildings in rural areas. Nevertheless, our method was specifically designed for the detection of very small objects (with respect to the number of image pixels), and we cannot expect good performance on large object detection. Indeed, as the proportion of the objects in the image increases, not only does the NN struggle to separate the distributions of the background from those of the objects, but also the unexpectedness feature of the targets becomes less pregnant. One consequence will be a fragmented detection of large objects, as illustrated in Fig. 12. Only the 'hottest' points of the aircraft will show up on the segmentation map. Therefore, there will be three detections for a single object, which will artificially increase the number of false alarms. A possible solution (left for future work) would be to be able to link these detections as belonging to the same object, e.g. according to a priori of distance or shape, or by coupling our approach with bounding box proposals.

7. Conclusion

In this paper, we propose an add-on NFA module to improve tiny object detection through semantic segmentation NN. This module introduces *a contrario* reasoning into the training of the NN, which therefore performs the detection of small objects by rejecting the background hypothesis. In addition to enhancing the feature map responses of small objects as proposed in the literature, our module allows for the control of the number of false alarms by exploiting the *unexpectedness* of tiny objects. We have experimentally demonstrated the competitiveness of our method compared to state of the art networks in the case of small target detection. More specifically, our method increases the precision of a baseline, while maintaining a high detection rate. It also provides more robustness to frugal training and it leads to a better generalization to unseen data. The results are also more interpretable, which is essential in many real-world applications. We have also shown that our approach generalizes well to other challenging tiny object detection tasks such as ship detection in satellite imagery, and road crack detection. Our NFA module also provides interpretable results, which is essential in many real-world applications.

An interesting perspective would be to apply our NFA module on object detection NN (e.g., YOLO or Faster R-CNN) for object-level detections in order to control the NFA at the object level, although this implies designing an object-level NFA (as in [8]). More generally, we believe that exploiting statistical criteria based on the *unexpectedness* of small objects, as with the *a contrario* approach we propose, can significantly improve the quality and interpretability of the results in the context of tiny object detection.

CRediT authorship contribution statement

Alina Ciocarlan: Methodology, Software, Validation, Writing – original draft. **Sylvie Le Hégarat-Mascle:** Conceptualization, Supervision, Writing – Review and Editing. **Sidonie Lefebvre:** Supervision, Writing – Review and Editing. **Arnaud Woiselle:** Supervision, Writing – Review and Editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

References

[1] Joseph Redmon, Santosh Kumar Divvala, Ross B. Girshick, Ali Farhadi, You only look once: Unified, real-time object detection, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016, IEEE Computer Society, 2016, pp. 779–788, http://dx.doi.org/10.1109/CVPR.2016.91.

[2] Fatih Cagatay Akyon, Sinan Onur Altinuc, Alptekin Temizel, Slicing aided hyper inference and fine-tuning for small object detection, in: 2022 IEEE International Conference on Image Processing (ICIP), 2022, pp. 966–970, http://dx.doi.org/10.1109/ICIP46576.2022.9897990.

[3] Tsung-Yi Lin, Piotr Dollár, Ross B. Girshick, Kaiming He, Bharath Hariharan, Serge J. Belongie, Feature pyramid networks for object detection, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017, IEEE Computer Society, 2017, pp. 936–944, http://dx.doi.org/10.1109/CVPR.2017.106.

[4] Xin Wu, Danfeng Hong, Jocelyn Chanussot, UIU-net: U-net in u-net for infrared small object detection, IEEE Trans. Image Process. 32 (2022) 364–376.
[5] Chunfang Deng, Mengmeng Wang, Liang Liu, Yong Liu, Yunliang Jiang, Extended feature pyramid network for small object detection, IEEE Trans. Multimed. 24 (2021) 1968–1979.
[6] Lukas Ruff, Nico Görnitz, Lucas Deecke, Shoaib Ahmed Siddiqui, Robert A. Vandermeulen, Alexander Binder, Emmanuel Müller, Marius Kloft, Deep one-class classification, in: Proceedings of the 35th International Conference on Machine Learning, ICML 2018, StockholmsmäSsan, Stockholm, Sweden, July 10-15, 2018, in: Proceedings of Machine Learning Research, vol. 80, PMLR, 2018, pp. 4390–4399.
[7] Agnès Desolneux, Lionel Moisan, J.-M. Morel, A grouping principle and four applications, IEEE Trans. Pattern Anal. Mach. Intell. 25 (4) (2003) 508–513.
[8] Agnes Desolneux, Lionel Moisan, Jean-Michel Morel, From Gestalt Theory to Image Analysis: A Probabilistic Approach, vol. 34, Springer Science & Business Media, 2007.
[9] Thibaud Ehret, Axel Davy, Mauricio Delbracio, Jean-Michel Morel, How to reduce anomaly detection in images to anomaly detection in noise, Image Process. Line 9 (2019) 391–412.
[10] Sylvie Le Hégarat-Mascle, Emanuel Aldea, Jennifer Vandoni, Efficient evaluation of the number of false alarm criterion, EURASIP J. Image Video Process. 2019 (2019) 35.
[11] Vincent Vidal, Matthieu Limbert, Tugdual Ceillier, 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] Bénédicte Grosjean, Lionel Moisan, A-contrario detectability of spots in textured backgrounds, J. Math. Imaging Vision 33 (3) (2009) 313–337.
[13] Milton Abramowitz, Handbook of Mathematical Functions, with Formulas, Graphs, and Mathematical Tables, Dover Publications, Inc., USA, ISBN: 0486612724, 1974, p. 263.
[14] Jie Hu, Li Shen, Gang Sun, Squeeze-and-excitation networks, in: 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018, IEEE Computer Society, 2018, pp. 7132–7141, http://dx.doi.org/10.1109/CVPR.2018.00745.
[15] Qilong Wang, Banggu Wu, Pengfei Zhu, Peihua Li, Wangmeng Zuo, Qinghua Hu, ECA-Net: Efficient channel attention for deep convolutional neural networks, in: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020, IEEE, 2020, pp. 11531–11539, http://dx.doi.org/10.1109/CVPR42600.2020.01155.
[16] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, Yichen Wei, Deformable convolutional networks, in: IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017, IEEE Computer Society, 2017, pp. 764–773, http://dx.doi.org/10.1109/ICCV.2017.89.
[17] Xiaolong Wang, Ross B. Girshick, Abhinav Gupta, Kaiming He, Non-local neural networks, in: 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018, IEEE Computer Society, 2018, pp. 7794–7803, http://dx.doi.org/10.1109/CVPR.2018.00813.
[18] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin, Attention is all you need, in: Advances in Neural Information Processing Systems, December 4-9, 2017, Long Beach, CA, USA, 2017, pp. 5998–6008.
[19] Niki Parmar, Prajit Ramachandran, Ashish Vaswani, Irwan Bello, Anselm Levskaya, Jon Shlens, Stand-alone self-attention in vision models, in: Advances in Neural Information Processing Systems, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada, 2019, pp. 68–80.
[20] Karen Simonyan, Andrew Zisserman, Very deep convolutional networks for large-scale image recognition, in: 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings, 2015.
[21] Olaf Ronneberger, Philipp Fischer, Thomas Brox, U-net: Convolutional networks for biomedical image segmentation, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2015, pp. 234–241.
[22] Renke Kou, Chunping Wang, Zhenming Peng, Zhihe Zhao, Yaohong Chen, Jinhui Han, Fuyu Huang, Ying Yu, Qiang Fu, Infrared small target segmentation networks: A survey, Pattern Recognit. 143 (2023) 109788.
[23] Boyang Li, Chao Xiao, Longguang Wang, Yingqian Wang, Zaiping Lin, Miao Li, Wei An, Yulan Guo, Dense nested attention network for infrared small target detection, IEEE Trans. Image Process. 32 (2022) 1745–1758.
[24] Mingjin Zhang, Rui Zhang, Yuxiang Yang, Haichen Bai, Jing Zhang, Jie Guo, ISNet: Shape matters for infrared small target detection, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 877–886.
[25] Mingjin Zhang, Rui Zhang, Jing Zhang, Jie Guo, Yunsong Li, Xinbo Gao, Dim2Clear network for infrared small target detection, IEEE Trans. Geosci. Remote Sens. 61 (2023) 1–14.
[26] Zhengxin Zhang, Qingjie Liu, Yunhong Wang, Road extraction by deep residual u-net, IEEE Geosci. Remote Sens. Lett. 15 (5) (2018) 749–753.
[27] Md Atiqur Rahman, Yang Wang, Optimizing intersection-over-union in deep neural networks for image segmentation, in: International Symposium on Visual Computing, Springer, 2016, pp. 234–244.
[28] Yimian Dai, Yiquan Wu, Fei Zhou, 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.
[29] Wei Zhang, Mingyu Cong, Liping Wang, Algorithms for optical weak small targets detection and tracking: review, in: International Conference on Neural Networks and Signal Processing, 2003. Proceedings of the 2003, Vol. 1, 2003, pp. 643–647, http://dx.doi.org/10.1109/ICNNSP.2003.1279357, Vol.1.
[30] Chuan Guo, Geoff Pleiss, Yu Sun, Kilian Q. Weinberger, On calibration of modern neural networks, in: Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017, in: Proceedings of Machine Learning Research, vol. 70, PMLR, 2017, pp. 1321–1330.
[31] Jacob König, Mark David Jenkins, Peter Barrie, Mike Mannion, Gordon Morison, A convolutional neural network for pavement surface crack segmentation using residual connections and attention gating, in: 2019 IEEE International Conference on Image Processing, ICIP, IEEE, 2019, pp. 1460–1464.
[32] Haifeng Li, Jianping Zong, Jingjing Nie, Zhilong Wu, Hongyang Han, Pavement crack detection algorithm based on densely connected and deeply supervised network, IEEE Access 9 (2021) 11835–11842.
[33] Rodrigo Rill-García, Eva Dokládalová, Petr Dokládal, Pixel-accurate road crack detection in presence of inaccurate annotations, Neurocomputing 480 (2022) 1–13, http://dx.doi.org/10.1016/j.neucom.2022.01.051.
[34] Zhenjie Liu, Jianming Xu, Jun Li, Antonio Plaza, Shaoquan Zhang, Lizhe Wang, Moving ship optimal association for maritime surveillance: Fusing AIS and sentinel-2 data, IEEE Trans. Geosci. Remote Sens. 60 (2022) 1–18.
[35] Alina Ciocarlan, Andrei Stoian, Ship Detection in Sentinel 2 Multi-Spectral Images with Self-Supervised Learning, Remote Sens. (ISSN: 2072-4292) 13 (21) (2021) 4255, http://dx.doi.org/10.3390/rs13214255.
[36] Qin Zou, Yu Cao, Qingquan Li, Qingzhou Mao, Song Wang, CrackTree: Automatic crack detection from pavement images, Pattern Recognit. Lett. 33 (3) (2012) 227–238.

Alina Ciocarlan received the Engineer degree from IMT Atlantique (France) in 2021 and the master degree from University of Rennes I in the same year. She is currently working towards a Ph.D degree in image processing in Paris-Saclay University.

Sylvie Le Hegarat-Mascle received the Ph.D. degree from Telecom ParisTech in 1996. She is currently a full Professor at Paris-Saclay University (France). Her research interests focus on statistical pattern recognition (gestalt and structure detection), image analysis and data fusion. Application domains include remote sensing and scene analysis and understanding.

Sidonie Lefebvre received the Ph.D. degree from Ecole Centrale Paris in 2006. She is currently working as a Research Scientist in statistics at ONERA Palaiseau (France), in the Optics and Associated Techniques Department. Her research interests are centered on the design and modeling of computer experiments, uncertainty quantification and sensitivity analysis, and target detection.

Arnaud Woiselle graduated from the engineering school Ecole Centrale Marseille in 2007, obtained his master's degree from University Aix-Marseille in image processing the same year, and his PhD in applied mathematics for image processing in 2010 from University Paris Diderot. He works at Safran as research engineer in image and video processing on super-resolution, contrast enhancement, sensor calibration for both visible and thermal infrared, using optimization and deep learning techniques.