MULTI-DATE WIND TURBINE DETECTION ON OPTICAL SATELLITE IMAGES

N. Mandroux,*   S. Drouyer,   R. Grompone von Gioi

Centre Borelli, ENS Paris-Saclay, France

KEY WORDS: remote sensing, wind turbine detection, a contrario method, NFA, multidate

ABSTRACT:

Satellite imaging is shifting from the photo-interpreter era to one of automatic monitoring. Indeed, the vast amount of data provided by the recent constellations of satellites, performing recurrent observation of every point on the globe, can only be handled by automatic methods; controlling false detections is thus crucial. The low costs of those satellites often imply lower resolution; the fusion of multi-date images can compensate to some extent the low resolution. Given their future role in the energetic transition and their spread over countries or continents, monitoring wind turbines is a natural candidate for such studies. This work details an algorithm for automatic, multi-date wind turbine detection on low resolution optical satellite images. The method is based on the *a contrario* statistical approach to provide a control of false detections and exploits the geometry of wind turbines' shadows and hubs.

1. INTRODUCTION

A wind turbine is a device that converts the wind's kinetic energy into electricity. It is composed of a vertical tower at the top of which stands the hub with its three rotating blades, see Figure 1. The American Wind Energy Association (AWEA, 2019) identifies wind as America's top renewable, no-emissions energy source. Therefore, wind turbines may have a key role to play in the energetic transition coming within the next decades. Being able to detect automatically their locations and thus their number and installed capacity could prove useful in managing the electrical network and planning wind power plant projects.

As the objective is to monitor the energy production of countries, huge areas must be acquired and analyzed. As a result, using aerial, drone or most high resolution commercial satellite images is too expensive. Images used must have an adequate resolution for detecting wind turbines – their hubs' length are around 10 meters – but still be inexpensive enough so that the cost of monitoring doesn't overwhelm the value of the generated data. The Sentinel-2 constellation, launched by the European Spatial Agency (ESA) in 2015 and 2017, provides free optical images of the whole world with a revisit time of 5 days and a 10 meters resolution. Images produced by Sentinel-2 seem therefore to be adapted to our objective.

Object detection in remote sensing is often dealt with neural networks. For this study though, we prefer the more probabilistic *a contrario* approach (Desolneux et al., 2007) for two reasons. Firstly, we do not have access to a large database of correctly annotated wind turbine images. Secondly we can naturally build an *a contrario* method using the shadow of the wind turbine and the brightness of its hub.

This paper presents a new application of the *a contrario* framework to wind turbine detection in remote sensing, using images with resolution lower than what can be found in previous works. The article is structured as follows. In Section 2, we explore the state of the art of wind turbine detection in remote sensing. We then describe our proposed method in Section 3. Finally, in
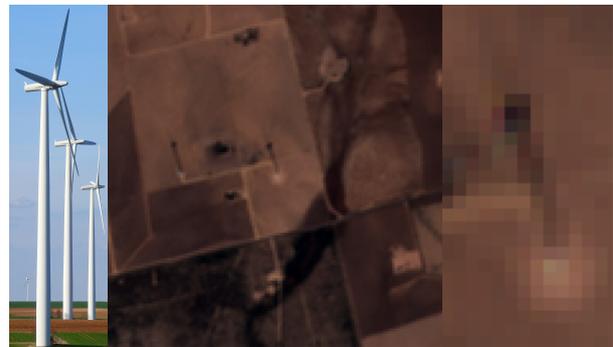

Figure 1. Pictures of wind turbines. Left, seen from the ground. If we ignore the blades, we can see its T-shape: we call tower the vertical bar and hub the top horizontal bar. Middle, scene of a wind farm acquired with Sentinel-2. Right, detail of middle picture on the central wind turbine. We can spot it thanks to its dark shadow and its bright hub. This example is one of the most visible case we can get with Sentinel-2 images.

Section 4 we quantitatively evaluate our method on a dataset we created and discuss the results.

2. STATE OF THE ART

Remote sensing detection has boomed over the past years (Cheng and Han, 2016), with the launch of many satellite constellations capable of acquiring data at high or very high resolution. The objects of detection are variable: trees, buildings, roads, airports, ships, etc. Wind turbine detection still remains quite untouched despite the usefulness of knowing their locations for meteorological (Christiansen and Hasager, 2005), environmental (Chang et al., 2016), radar analysis (Nepal et al., 2015), or electricity management reasons (Han et al., 2018, Supper & Supper, 2019, Abedini et al., 2019).

The majority of state-of-the-art methods use neural networks (Han et al., 2018, Supper & Supper, 2019, Abedini et al., 2019, Manso-Callejo et al., 2020, Zhang et al., 2020). Other non deep learning based methods have also been developed
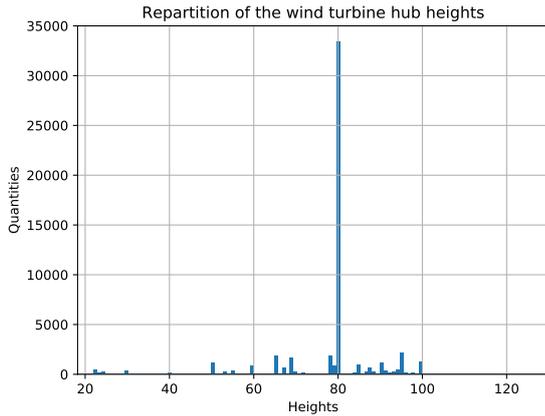

Figure 2. Height histogram of US wind turbines in meters.

such as the algorithm developped by Chen et al. (Chen et al., 2018). These methods produce accurate results, but require fully-annotated database of relatively high resolution images (at least 2 m/px), which are rarely publicly available and expensive to get or to create (Manso-Callejo et al., 2020); especially for neural networks when one needs thousands of examples.

A recent work (Mandroux et al., 2021) focuses on Sentinel-2 images, using an *a contrario* method. We based our research on this method and improve it by the addition of temporal coherence along multi-date images.

3. PROPOSED METHOD

The structure of a wind turbine is T-shaped, with a vertical tower at the top of which stands the hub, where the rotating blades connect (see Figure 1). The height of a majority of towers is around 80 meters (Hoen et al., 2018) (see Figure 2). For heat reasons they are all painted in white (see Figure 1). As we can see in Figure 1, a wind turbine's footprint is composed of two parts: the dark shadow induced by the tower, and the bright hub. Since this kind of footprint is shared by every wind turbine in the world, we can use this knowledge coupled with the known positions of the satellite and the sun and a hypothesis on the height of the tower to build a detector. This detector is the fusion of a shadow detector and a hub detector. To use most of that knowledge and quantify the degree of certainty of our detections, we choose the *a contrario* framework (Desolneux et al., 2007). To avoid blurring effects due to possibly rotating blades acquired by the push-broom scanning, only one spectral band is used: the blue B02 one. Given the sun's (resp. satellite's) altitude θ and azimuth ϕ, and the tower's height h, the coordinates (x, y) at the end of the shadow (resp. hub) are:

$$(x,y) = \left(-\frac{h \sin(\phi)}{|\tan(\theta)|}, -\frac{h \cos(\phi)}{|\tan(\theta)|} \right). \quad (1)$$

3.1 *A contrario* framework

The *a contrario* approach is inspired on visual perception (Desolneux et al., 2007). Human vision is attracted by patterns which differ from their background. If the background can be probabilistically modeled, one can question this model by observing the structure of some of the pixels present in the image, and then quantify how far it diverges from the model. If it diverges sufficiently, it means there is a structure to be detected. This is stated by the non-accidentalness principle (Witkin and Tenenbaum, 1983, Lowe, 1985) which informally says that an observed structure is meaningful only when the relation between its parts is too regular to be the result of an accidental arrangements of independent parts. This leads to a statistical framework used to set detection thresholds automatically in order to control the number of false detections.

Following the *a contrario* methodology (Desolneux et al., 2007), we define the *Number of False Alarms* (NFA) of an event e with an observed measurement $k(e)$ as

$$\mathbf{NFA}(e) = N_{tests} \cdot \mathbf{P}\Big[K_{H_0}(e) \geq k(e)\Big], \quad (2)$$

where the right hand term is the probability of obtaining, in the background model H_0, a value $K_{H_0}(e)$ larger or equal to the observed one $k(e)$; N_{tests} is the total number of tests performed. The smaller the NFA, the more unlikely the event e is to be observed by chance in the background model H_0; thus, the more meaningful. The *a contrario* approach prescribes accepting as valid detections the candidates with NFA $< \epsilon$ for a predefined value ϵ. It can be shown (Desolneux et al., 2007) that under H_0, the expected number of tests with NFA $< \epsilon$ is bounded by ϵ. As a result, ϵ gives an *a priori* estimate of the mean number of false detections under H_0.

3.2 Pixel values comparisons

Let U be an image. Consider each of its pixels u_i to be a continuous random scalar variable, and suppose these variables are iid. This is the background random model H_0 for our *a contrario* setting. Under these assumptions,

$$\mathbf{P}(u_{i_1} \leq u_{i_2}) = \mathbf{P}(u_{i_1} \geq u_{i_2}) = \frac{1}{2},$$

for $i_1 \neq i_2$. More generally,

$$\mathbf{P}\left(\bigcap_{k=1}^{l} (u_{i_0} \leq u_{i_k}) \right) = \mathbf{P}\left(\bigcap_{k=1}^{l} (u_{i_0} \geq u_{i_k}) \right) = \frac{1}{2^l}. \quad (3)$$

Let $X_{i,j} = \mathbb{1}_{u_i \leq u_j}$, where $\mathbb{1}$ is the indicator function, be a random variable; thus $X_{i,j} = 1$ if $u_i \leq u_j$ and $X_{i,j} = 0$ otherwise. Symmetrically, let $Y_{i,j} = \mathbb{1}_{u_i \geq u_j}$. Denoting B the Bernoulli distribution, we know, from Eq. (3) that $X_{i,j} \sim \mathrm{B}(\frac{1}{2})$ and $Y_{i,j} \sim \mathrm{B}(\frac{1}{2})$.

Let $J = \{j_1, j_2, \ldots, j_l\}$ with j_k all different. We define the random variable

$$X_{i,J} = \prod_{k=1}^{l} X_{i,j_k}. \quad (4)$$

By independence, $X_{i,J} \sim \mathrm{B}(\frac{1}{2^l})$. Symmetrically, we can define $Y_{i,J}$; $Y_{i,J} \sim \mathrm{B}(\frac{1}{2^l})$.

Finally, let $X = \{X_{i_1,J_1}, \ldots, X_{i_n,J_n}\}$, with $i_1 \neq \ldots \neq i_n$, $\#J_1 = \cdots = \#J_n = l$, and $\forall 1 \leq i, j \leq n, J_i \cap J_j = \emptyset$. We define $S_X = \sum_{s=1}^{n} X_{i_s,J_s}$. By the iid hypothesis,

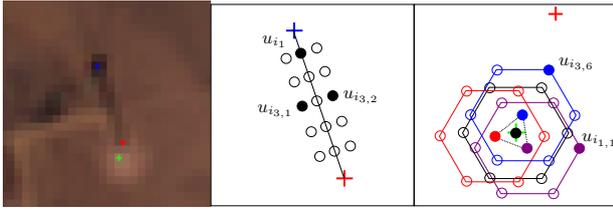$$S_X \sim \mathrm{Bin}\left(n, \frac{1}{2^l} \right), \quad (5)$$

Figure 3. Left, a wind turbine view from a Sentinel-2 image, with its hub shadow (blue cross), pylone foot (red cross) and hub (green cross). Middle, zoomed in, the sampling along the pylone shadow where $n_s = 5$. Right, zoomed in, the sampling around the hub where $n_h = 4$.

where Bin is the binomial distribution. What we have stated allows us to easily compute the probability for a group of specific pixels to have larger values than neighbours. Intuitively, we will be surprised to find a group X of n random variables such that S_X is large (i.e., large relative to the what would be expected under H_0).

3.3 Shadow detection

3.3.1 Monodate For each pixel $u_i \in U$, we want to compute if it is a plausible candidate for the beginning of the wind turbine's shadow. Given the satellite's metadata and the knowledge of the sun's position, one can compute the footprint of the shadow, and sample it. For each of these samples $u_{i_k}, 1 \le k \le n_s$, we consider the two neighbours perpendicular to the shadow's direction: $J_{i_k} = \{u_{i_{k,1}}, u_{i_{k,2}}\}$, see Figure 3, middle. We consider

$$s_i^{shadow} = \sum_{k=1}^{n_s} X_{i_k,i_{k,1}} \times X_{i_k,i_{k,2}}, \tag{6}$$

where s_i^{shadow} takes a value between 1 and n_s, the larger the value the better the agreement with a shadow being observed at pixel u_i. Under the random assuptions H_0, and using Eq. 5, s_i^{shadow} becomes the random variable $S_i^{shadow} \sim \text{Bin}(n_s, \frac{1}{4})$.

3.3.2 Multidate Let $T = (U_1, \ldots, U_n)$ be n images of the exact same scene at different dates. For each of these images, one can compute the shadow score of a given pixel i: $s_{i,U_q}^{shadow}, 1 \le q \le n$. We consider the multidate shadow score of pixel i:

$$s_{i,T}^{shadow} = \sum_{q=1}^{n} s_{i,U_q}^{shadow}. \tag{7}$$

Under the random model H_0, $s_{i,T}^{shadow}$ follows the random variable $S_T^{shadow} \sim \text{Bin}(n_T, \frac{1}{4})$, where $n_T = \sum_{q=1}^{n} n_{s_q}$ and n_{s_q} is the number of shadow samples in the image U_q.

3.4 Hub detection

3.4.1 Monodate For each pixel $u_i \in U$, we want to compute if it is a plausible candidate for the bottom of the turbine's tower. Given the satellite's metadata, one can compute the position of the hub (Figure 3 green cross) and sample it. Notice that the apparent hub is a bit shifted relative to the pylon foot because the satellite is not necessary at the wind turbine zenith; the more the wind turbine is far from the nadir point of view, the more it appears slanted. For each of these samples $u_{i_k}, 1 \le k \le n_h$, we consider the six neighbours which are the

vertices of a hexagon centered in u_{i_k}: $J_{i_k} = \{u_{i_{k,1}}, \ldots, u_{i_{k,6}}\}$, see Figure 3, right. We consider

$$s_i^{hub} = \sum_{k=1}^{n_h} \prod_{l=1}^{6} Y_{i_k,i_{k,l}}, \tag{8}$$

where s_i^{hub} takes a value between 1 and n_h, the larger the value the better the agreement with a hub being observed at pixel u_i. Under the random assuptions H_0, and using Eq. 5, s_i^{hub} becomes the random variable $S_i^{hub} \sim \text{Bin}(n_h, \frac{1}{2^6})$. n_h is set to 7 for the experiments. Therefore the hub is sampled by 7 points: the central one and six others just around it.

3.4.2 Multidate Let $T = (U_1, \ldots, U_n)$ be n images of the exact same scene at different dates. For each of these images, one can compute the hub score of a given pixel i: $s_{i,U_q}^{hub}, 1 \le q \le n$. We consider the multidate hub score of pixel i:

$$s_{i,T}^{hub} = \sum_{q=1}^{n} s_{i,U_q}^{hub} \tag{9}$$

Under the random assumptions H_0, $s_{i,T}^{hub}$ follows the random variable $S_T^{hub} \sim \text{Bin}(n \times n_h, \frac{1}{2^6})$, where n_h is the number of hub samples in a given image.

3.5 Tightening tests

We have presented the theoretical framework of the method. Yet, we could dramatically improve the performances by tightening the tests. Rather than wanting for a shadow-sampled pixel to be just darker than its two neighbors, we ask it to be darker than its two neighbors minus a threshold t^s. Similarly, for a hub-sampled pixel, we want it to be brighter than its six neighbors plus a threshold t^h. These thresholds have been fixed empirically for this article, further research could focus on a more substantial definition.

We replace $X_{i,j}$ by $\widehat{X}_{i,j}^t = \mathbb{1}(u_i \le u_j - t)$ and $Y_{i,j}$ by $\widehat{Y}_{i,j}^t = \mathbb{1}(u_i \ge u_j + t)$. With these new notations, we can define the new hub and shadow scores:

$$\widehat{s}_i^{hub} = \sum_{k=1}^{n_h} \prod_{l=1}^{6} \widehat{Y}_{i_k,i_{k,l}}^{t^h}, \tag{10}$$

$$\widehat{s}_i^{shadow} = \sum_{k=1}^{n_s} \widehat{X}_{i_k,i_{k,1}}^{t^s} \times \widehat{X}_{i_k,i_{k,2}}^{t^s}. \tag{11}$$

3.6 Shadow and hub aggregation

Once $\widehat{s}_{i,T}^{shadow}$ and $\widehat{s}_{i,T}^{hub}$ are computed, we need to decide how to use these two pieces of information. Since a wind turbine footprint is composed of a shadow *and* a hub, we want both structures to be detected.

Since we know the distributions of S_T^{shadow} (resp. S_T^{hub}), we can compute the probability of false alarms for a shadow (resp. hub) detection: $p_i^{shadow} = \mathbf{P}(S_T^{shadow} \ge \widehat{s}_{i,T}^{shadow})$ (resp. hub). Finally, the number of false alarms for shadow (resp. hub) detection is $\mathbf{NFA}_i^s = N_{tests} \times p_i^{shadow}$ (resp. hub), which corresponds to the expected number of false alarms under H_0 when N_{tests} trials are made – here, the number of pixels in the image.

Finally, we impose both NFAs to be low enough, to avoid false alarms. Given a threshold $\epsilon \in \mathbb{R}$, we compute $\mathbf{NFA}_i$ and deduce the final detection map by thresholding:

$$\mathbf{NFA}_i = \begin{cases} \mathbf{NFA}_i^s \times \mathbf{NFA}_i^h & \text{if } (\mathbf{NFA}_i^s \le \epsilon) \text{ and } (\mathbf{NFA}_i^h \le \epsilon) \\ N_{tests} & \text{otherwise.} \end{cases}$$

(12)

$$\mathbf{DETEC}_i = \begin{cases} 1 & \text{if } \mathbf{NFA}_i \le \epsilon \\ 0 & \text{otherwise.} \end{cases}$$

(13)

4. DATA, EXPERIMENTS AND RESULTS

4.1 First experiment and performance quantification

We selected a Sentinel-2 scene of 1000×1000 pixels, downloaded 4 dates of the same scene: in March, September, November and December. We avoided summer dates to have the most visible shadows. We also avoided cloudy dates: there is no cloud in the used images. On the selected scene are 40 wind turbines, and various landscapes: town, roads, relief, crops, paths, water, trees. We applied both algorithms to it. As said in Section 3, only the B02 channel was used. We set the threshold $\epsilon = 1$ and applied our multidate algorithm on the time series; we got a shadow detection map, a hub detection map, and a global detection map. Since the original image is 1000×1000 pixels, we show partial panels of it, see Figure 4. The first panel shows the only false positive we got.

We can state four remarks about this experiment. First, except for the example shown, there is no other false positive detected. Having one false positive is in agreement with our choice of threshold $\epsilon = 1$ for the $\mathbf{NFA}$. It means that we expect to get 1 false positive, and it is the case here. Secondly, wind turbines are well detected, thanks to the double classifier shadow-hub: around 80% of them are spotted. Thirdly, the good detecting is mainly due to the shadow detector, which gets very few false positives. We can see a plum-shaped detection pattern, which tends to be corrected by the hub detector. Fourthly, the hub detector is very loose: it detects all kind of structures: roads and relief as we can see on Figure 4, but also buildings and local maxima. To quantify it: 1.6% of the pixels in the image are detected by the hub detector. It could be decent if we were not working on an image containing millions of pixels: here, this 1.6% corresponds to $16,000$ detections. As a result, given the small number of positives, it is an enormous amount of false positives detected by the hub detector.

We compare the performance of the proposed approach to the monodate one (Mandroux et al., 2021). For both algorithms, detection maps have been computed and ROC curves plotted, see Figure 5. We can see that the multidate method clearly outperforms the monodate one, its ROC curve being close to perfection. Yet the monodate ROC curve we computed here is quite better than in the original article (Mandroux et al., 2021), so we should assess that the selected image is easier than the database used in the previous article. Nevertheless, in absolute terms the level of performance obtained by the multidate method seems close to perfect, and suggests that we got what we wanted: a good detector of wind turbines. That would be true if we were not looking at a vast amount of pixel candidates in which only a few are objects of interest; since it is the case, the ROC curve does not give a correct vision of the absolute performances.

Indeed, it is a good way to show that the multidate method is better than the monodate one, since we only had this measure
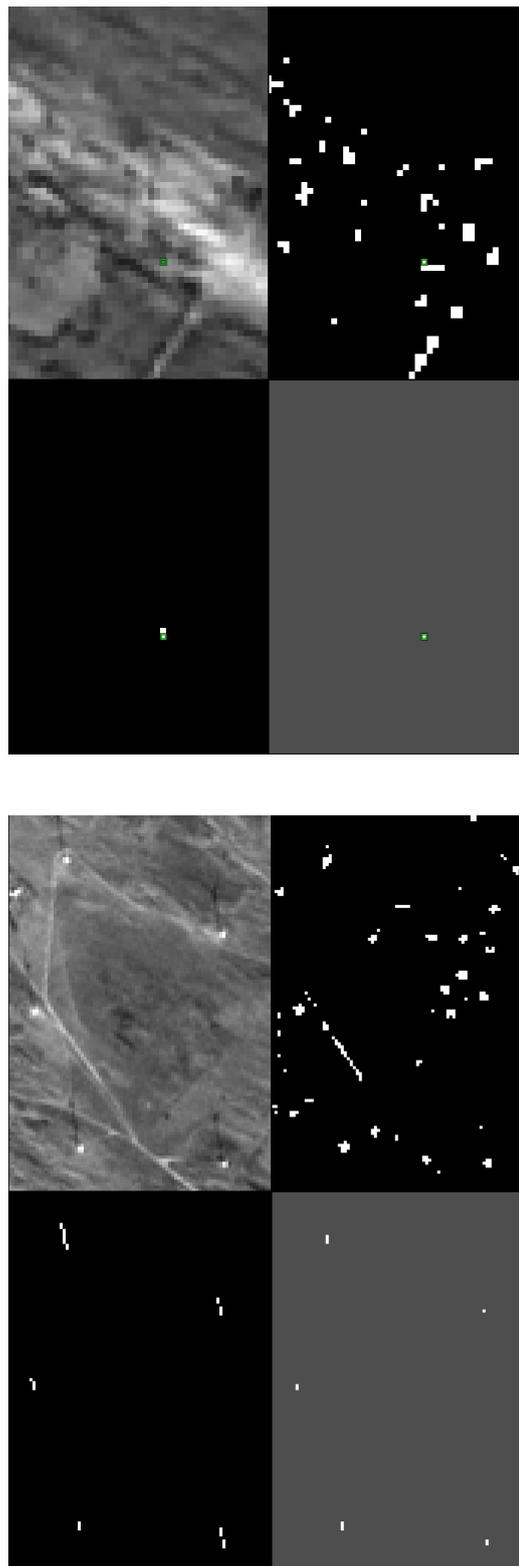

Figure 4. Two panels of detections of our algorithm, $\epsilon = 1$. The first panel shows a false positive highlighted by a green square, the second panel shows some true positives. On both panels, the first image is the blue channel of the Sentinel-2 scene in November, the top right is the hub detection map, the bottom left one is the shadow detection map, the last one is the detection map. On the detection maps, white pixels are detected, black ones are not.
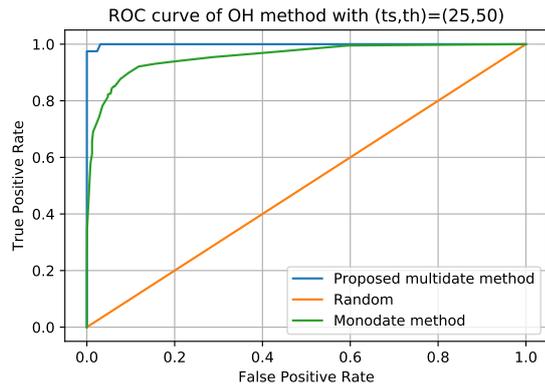
Figure 5. Comparison in performances between the multidate and monodate method. In blue, the ROC curve of the multidate method. In green, the ROC curve of the monodate one. Each point is obtained by thresholding the NFA map with a different value ϵ. In orange, the performance of a pure random classifier.
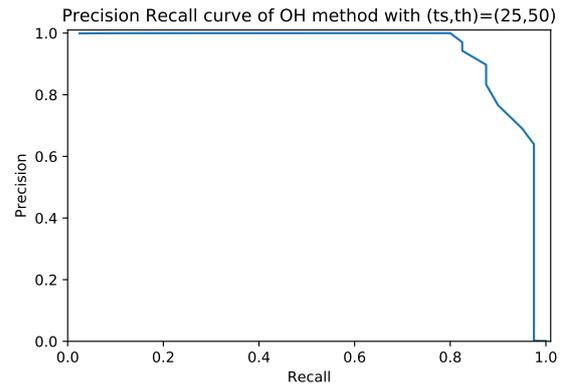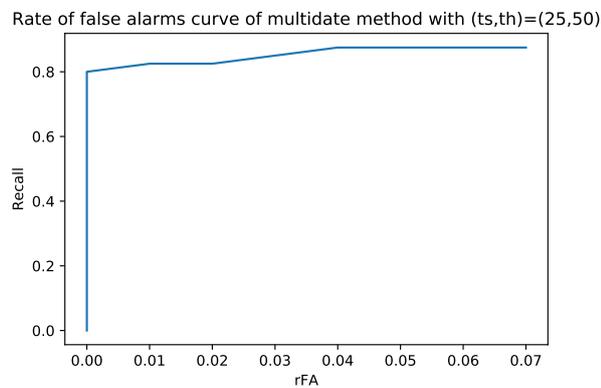

Figure 6. rFA of the multidate method.

of performance for the monodate method, but we shall not be too enthusiastic at the sight of this nearly perpendicular curve. Hence, we should better look at another metric which we call the rate of false alarms. It is the number of false positives per square kilometer. It allows us to compute an estimate of the total number of false alarms we would get if we tried the algorithm on a way bigger area like a whole country or a continent, and thus determines more specifically our performances, see Figure 6.

It shows that to detect more than 80% of the wind turbines, we get 0.01 false positives by square kilometers. This is a poor result if our aim is to apply the algorithm on a whole country which size can exceed the million of square kilometers: we would get an order of magnitude of $1,000,000 \times 0.01 = 10,000$ false alarms, which is a lot to check manually.

Another way of quantifying the performances is the Precision-Recall curve, see Figure 7. This figure shows good performances: up to around 75% of the wind turbines can be found without any false positive. The associated mean Average Precision is 0.939. To give a rough idea of comparison, deep learning methods (Zhang et al., 2020) can give an Average Precision between 0.9 and 0.98 using images with much better resolution: between 0.5 and 2 meters, whereas Sentinel-2 optical ones have a resolution of 10 meters.

Figure 7. Precision-Recall graph of the multidate method. Each point is obtained by varying ϵ, the **NFA** threshold.

4.2 Large scale experiment

We tried the algorithm on 100 square kilometers, which enabled us to work on a fully annotated image and produce the quantitative performances. In absolute terms, it is not a small area: it is practically the size of the city of Paris in France. But it stays small compared with the size of a country. So we tried the same algorithm on a $100 \times 100 = 10,000$ square kilometers area, a size close to the State of Connecticut in the USA. The chosen area contains the previous 100-square-kilometers one, which is annotated. The rest is not: we suppose it empty of wind turbines, and we look manually at the detected pixels to check whether a wind turbine was effectively detected. There again, we avoid clouds and only use images free of it; this can be done automatically using cloud detectors such as Fmask (Qiu et al., 2019).

For computer memory reasons, we do not apply the algorithm to the whole $10,000$ by $10,000$ image, but we divide it into 1000 by 1000 images. This affects the choice of epsilon: we will need it stricter than 1. We try 3 values of **NFA** threshold: $\epsilon = 1$, 0.1 and 0.01. For a given series of 4 1000-by-1000 images, the computation time is 14s; for the whole scene, it is around 23 minutes. It was executed on a PC with an Intel® Xeon(R) E-2176M CPU @ 2.70GHz × 12 and 16 GB memory.

For $\epsilon = 1$, the loosest threshold, we detect a bit too much. We expect to detect on average $\epsilon = 1$ false positives on every 1000 by 1000 image, and we get something closer to 3 or 4 false positives on average, which is worse but not very different from what we expected. The false positives come from, sorted in descending frequency: crops, buildings, roads, relief, and also randomness. Figure 8 shows some zoomed failures. On the bright side, we detect three other wind turbine fields, containing about 30 wind turbines each. Approximately 60% of them are detected, which is less than anticipated with our study on the first experiment. It confirms what we suspected before: the first experiment had very visible wind turbines and was a favorable case of study. Figure 9 shows some zoomed good detections.

When ϵ decreases, the threshold tightens, thus the number of false alarms and true positives. With $\epsilon = 0.1$, we expect to get on average 0.1 false positives by 1000 by 1000 image. We get a total of 26 false positives, which corresponds to 0.26 false positive by 1000 by 1000 image. There again, it is more than expected, but still close to. The number of correct detections falls too: only 45% of wind turbines are detected. With $\epsilon = 0.01$, we
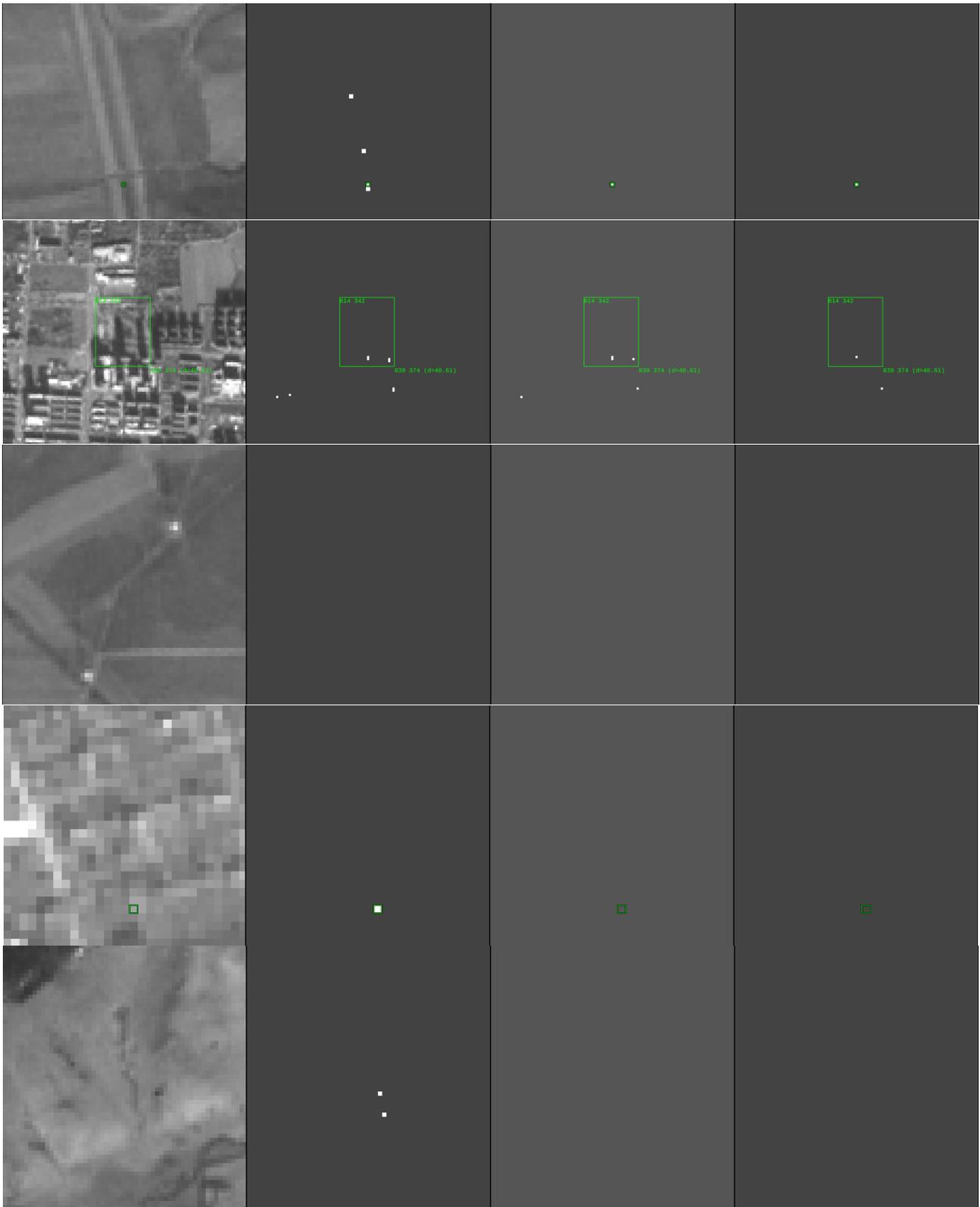
Figure 8. Bad results of multidate detection. The first column is the Sentinel-2 scene, the second one is the final detection map with $\epsilon = 1$, the third one is the detection map with $\epsilon = 0.1$, the final one is the detection map with $\epsilon = 0.01$. Each row shows a different example. On the detection maps, white pixels are detections, black ones are not. On some of the rows, a rectangular green box is drawn to ease the matching between the columns.
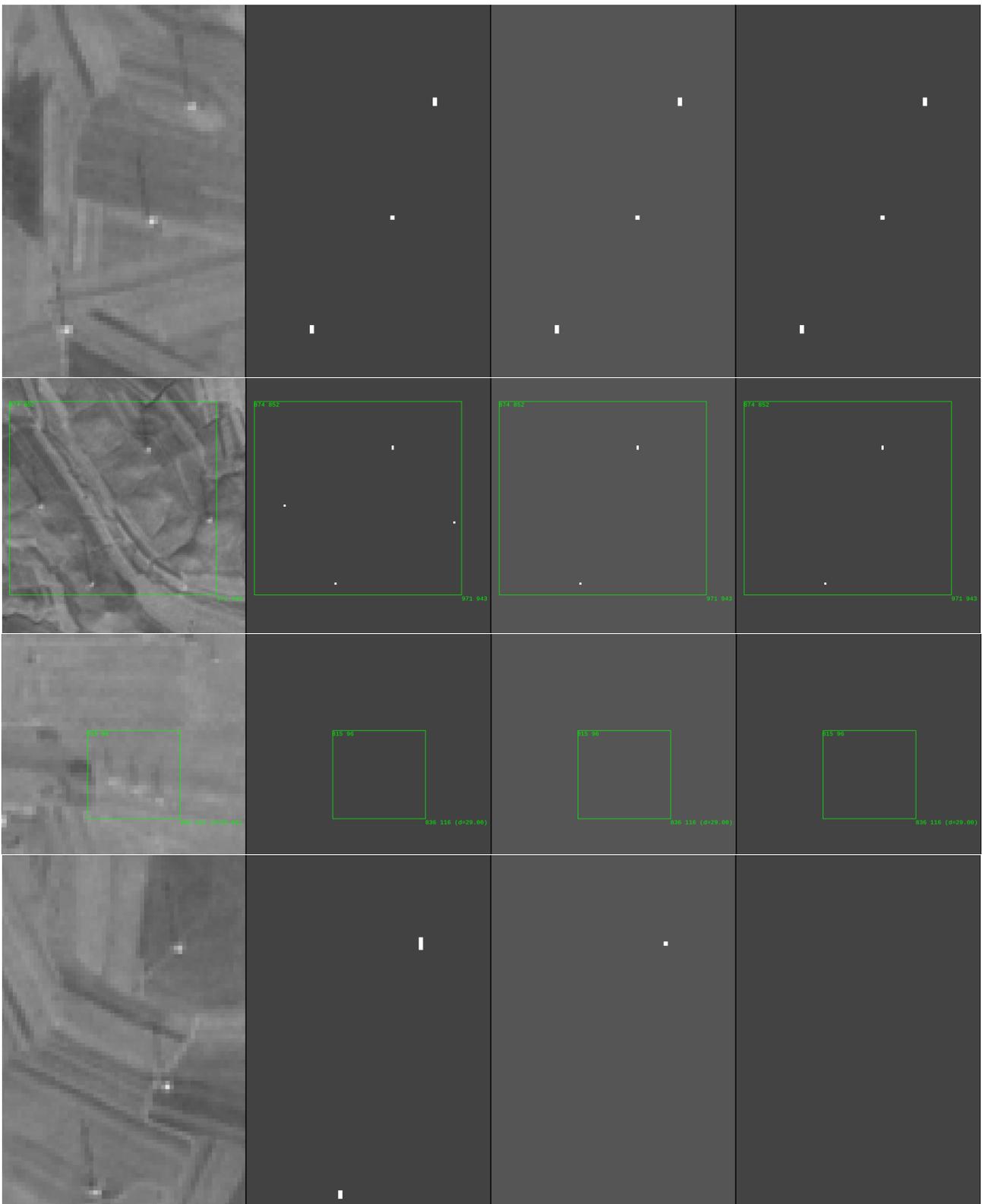
Figure 9. Good results of multidate detection. The first column is the Sentinel-2 scene, the second one is the final detection map with $\epsilon = 1$, the third one is the detection map with $\epsilon = 0.1$, the final one is the detection map with $\epsilon = 0.01$. Each row shows a different example. On the detection maps, white pixels are detections, black ones are not. On some of the rows, a rectangular green box is drawn to ease the matching between the columns.

should only get 1 false positive in total, but we get 15. Those are no mere coincidences, and come from difficult structures on the ground. The fraction of true positives decreases to 40%.

Looking at Figure 8, on its first and second rows, dark lines coherent with shadow's directions according to the sun azimuth are clearly visible; the shadow detector activates rightly. We could nonetheless argue about the width of the line: it is too large compared to a wind turbine's, so there might be some refinements to do there. For the hub, it is quite different: on the crop structure of the first line, we cannot really see any local maximum, and the detection here seems abusive. There is substantial improvement to be made on these cases. On the second row it is less blameworthy as we can see a local maximum at the base of the would-be wind turbine. In fact, if what we see there are indeed tall white buildings projecting their shadows on the floor, then we shall not be surprised to detect them. The third row shows an example of false negative where the shadow detector fails because of the weak contrast between shadow and neighbors. The fourth and fifth rows show examples of detections due to randomness: no visible structure, the Sentinel-2 looks very noisy, the detectors' thresholds are barely passed. As ϵ decreases, these random detections disappear.

On Figure 9, the first two rows show newly detected wind turbines. Thanks to the automatic detection, we were able to find these wind turbines and their associated fields, which would not have been easily found otherwise. The wind turbines on the first row are very visible; on the second one, some of them are more difficult. For example the shadow of the one close to the left border is polluted by a ground structure which might be a road. As ϵ decreases, the most polluted of them are not detected anymore. The third row shows a convincing true negative. We can see three dark lines coherent with the shadow's directions, associated with a local bright maxima. They are not detected thanks to our hypothesis on the height of the wind turbines: their shadows are not long enough. The fourth row shows some weak true positives, which barely pass the detectors' tests and become false negatives when ϵ decreases.

5. CONCLUSION

An algorithm for wind turbine detection in Sentinel-2 images was described which leverages the information on multi-date images to compensate for the low resolution, improving the state of the art. The method is based on the *a contrario* statistical framework to control false detections, resulting in a low false detection rate. Yet, when aiming at automatic detection on continent-wide extensions, the false detection rate may still not be good enough. Future work will address this issue by improving the hub detection step, by coupling optical and SAR data and by adding a mechanism for selecting dates with good images. The latter improvement shall also tackle the major difficulty of cloudy images. Another hint is first detecting wind turbine clusters and then refining the detection by the identification of individual wind turbines on higher-resolution images; high-resolution (and expensive) images would only be required for very limited zones.

REFERENCES

Abedini, F., Bahaghighat, M., S'hoyan, M., 2019. Wind turbine tower detection using feature descriptors and deep learning. *Facta Universitatis, Series: Electronics and Energetics*, 33(1), 133–153.

AWEA, 2019. Wind powers america amended annual report 2019. Technical report, American Wind Energy Association.

Chang, R., Zhu, R., Guo, P., 2016. A Case Study of Land-Surface-Temperature Impact from Large-Scale Deployment of Wind Farms in China from Guazhou. *Remote. Sens.*, 8, 790.

Chen, J., Yue, A., Wang, C., Huang, Q., Chen, J., Meng, Y., He, D., 2018. Wind turbine extraction from high spatial resolution remote sensing images based on saliency detection. *Journal of Applied Remote Sensing*, 12, 016041.

Cheng, G., Han, J., 2016. A survey on object detection in optical remote sensing images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 117, 11 - 28.

Christiansen, M., Hasager, C., 2005. Wake effects of large offshore wind farms identified from satellite SAR. *Remote Sensing of Environment*, 98(2-3), 251–268.

Desolneux, A., Moisan, L., Morel, J.-M., 2007. *From Gestalt Theory to Image Analysis: A Probabilistic Approach*. 1st edn, Springer Publishing Company, Incorporated.

Han, M., Wang, H., Wang, G., Liu, Y., 2018. Targets Mask U-Net For Wind Turbines Detection In Remote Sensing Images. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-3, 475–480.

Hoen, B. et al., 2018. U.S. geological survey, american wind energy association, and lawrence berkeley national laboratory data release. https://doi.org/10.5066/F7TX3DN0. Version 3.3 of January 14, 2021.

Lowe, D., 1985. *Perceptual Organization and Visual Recognition*. Kluwer Academic Publishers. ISBN 978-1-4613-2551-2.

Mandroux, N., Dagobert, T., Drouyer, S., Grompone von Gioi, R., 2021. Wind turbine detection on Sentinel-2 images. *2021 IEEE International Geoscience and Remote Sensing Symposium*.

Manso-Callejo, M.-, Cira, C.-I., Alcarria, R., Arranz Justel, J.-J., 2020. Optimizing the Recognition and Feature Extraction of Wind Turbines through Hybrid Semantic Segmentation Architectures. *Remote Sensing*, 12(22).

Nepal, R., Cai, J., Yan, Z., 2015. Micro-Doppler radar signature identification within wind turbine clutter based on short-CPI airborne radar observations. *IET Radar, Sonar & Navigation*, 9(9), 1268–1275.

Qiu, S., Zhu, Z., He, B., 2019. Fmask 4.0: Improved cloud and cloud shadow detection in Landsats 4–8 and Sentinel-2 imagery. *Remote Sensing of Environment*, 231(May), 111205.

Supper & Supper, 2019. Deep Learning Utilizing Satellite Imagery Feature Detection to create a Wind Turbine Register. https://supperundsupper.com/wp-content/uploads/2019/10/Deep-Learning-Utilizing-Satellite-Imagery.pdf.

Witkin, A. P., Tenenbaum, J. M., 1983. On the role of structure in vision. J. Beck, B. Hope, A. Rosenfeld (eds), *Human and Machine Vision*, Academic Press, 481–543.

Zhang, X., Han, L., Han, L., Zhu, L., 2020. How Well Do Deep Learning-Based Methods for Land Cover Classification and Object Detection Perform on High Resolution Remote Sensing Imagery? *Remote Sensing*, 12(3), 417.