Point process and CNN for small object detection in satellite images

Jules Mabon, Mathias Ortner, Josiane Zerubia

Point process and CNN for small object detection in satellite images

Jules Mabon [a], Mathias Ortner[b], and Josiane Zerubia [a]

[a]Inria, Université Côte d'Azur, Sophia-Antipolis, France
[b]Airbus Defense and Space, Toulouse, France

ABSTRACT

In this article we present a combination of marked point processes with convolutional neural networks applied to remote sensing. While point processes allow modeling interactions between objects via priors, classical methods rely on contrast measures that become unreliable as objects of interest and context become more diverse. We propose learning likelihood measures using convolutional neural networks to make these measures more versatile and resilient. We apply our method to the detection of vehicles in satellite images.

Keywords: Small object detection, Remote sensing, Marked point process, Convolutional neural network

1. INTRODUCTION

Detecting small objects in satellite images is a challenging task: objects of interest appear a few pixels wide because of the low spatial resolution, limiting the available visual information. Moreover, objects are often densely scattered, which can make the instance separation difficult.[1]

Multiple methods based on Convolutional Neural Networks (CNN) such as Faster R-CNN,[2] YOLO,[3] or RetinaNet[4] detect objects in "natural" images containing large objects and limited interactions between those. The size and limited geometrical parametrization of detection boxes, along with limits on the number of detections make these approaches less reliable in the context of remote sensing. These approaches have been adapted to remote sensing image challenges but still rely on rather high resolutions (around 15 cm/pixel) such as in Refs. 5–14.

On the other hand, approaches based on stochastic geometry offer to jointly solve the detection and selection of objects. Such models allow incorporating interaction models (priors), while extracting vectorized information (rather than raster masks). Stochastic geometry approaches classically make use of contrast measures[15–21] as the likelihood of the model. Properties inherent to our application such as objects and context visual diversity, partial occlusions and lighting conditions make the simple contrast measure inoperable. While one could devise more complex and reliable measures fit to each situation, this would prove tedious and computationally expensive.

We thus propose to combine the stochastic geometry approach – that incorporates priors – with likelihood measures built from CNN. Our main contributions are as follows :

- stating the detection task as an energy minimization problem, allowing to introduce priors on configurations
- introducing likelihood measures based on simple CNN models to replace contrast measures
- comparing quantitatively and qualitatively results from our model against another CNN based model[6] for small vehicle detection task on several datasets.

Further author information:
J. Mabon: E-mail: jules.mabon@inria.fr

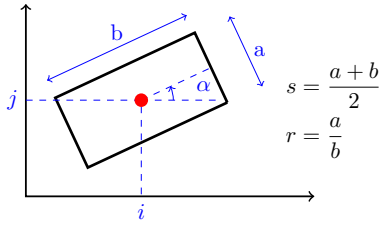


Figure 1. Rectangle parametrization

2. POINT PROCESS DEFINITION

Let $S \in \mathbb{R}^2$ the image space; A configuration of points Y is a finite unordered set of elements of $S \times M$, with M the marks space. In this paper, an element of the set – corresponding to an object – $y \in Y$is comprised of coordinates i, j, and three marks that describe a rectangle (see Fig. 1): size s, ratio r and angle α. We denote the set of all possible configurations (*ie.* the union of all possible n points configurations) as $\mathcal{Y} = \bigcup_{n=0}^{\infty}(S \times M)^n$.

A configuration of points is modeled as the realization of a non-uniform Marked Point Process (MPP); defined through a density h relative to the uniform Poisson process.[22]

The model of selection and interaction of points is given by an energy U, via a non-normalized Gibbs density:

$$h(Y) \propto \exp\left(-U(Y)\right) \tag{1}$$

Given an observed image X, the best fitting configuration of points $\widehat{Y}$ is inferred as $\underset{Y \in \mathcal{Y}}{\operatorname{argmin}}\, U(X, Y)$. At the training stage, we thus need to devise an energy $U(X, Y)$ such as its minimum is reached at – or closest to – the ground truth configuration Y^+ associated to image X.

3. ENERGY MODEL

The energy for an image X and a configuration Y is as follows:

$$U(X,Y) = \sum_{y \in Y} \omega_p U_p(X,y) + \mathbb{1}_{U_p(X,y)<0}\left(\omega_m U_m(X,y) + \omega_s U_s(y) + \omega_o U_o(y, \mathcal{N}_y) + \omega_a U_a(y, \mathcal{N}_y)\right) \tag{2}$$

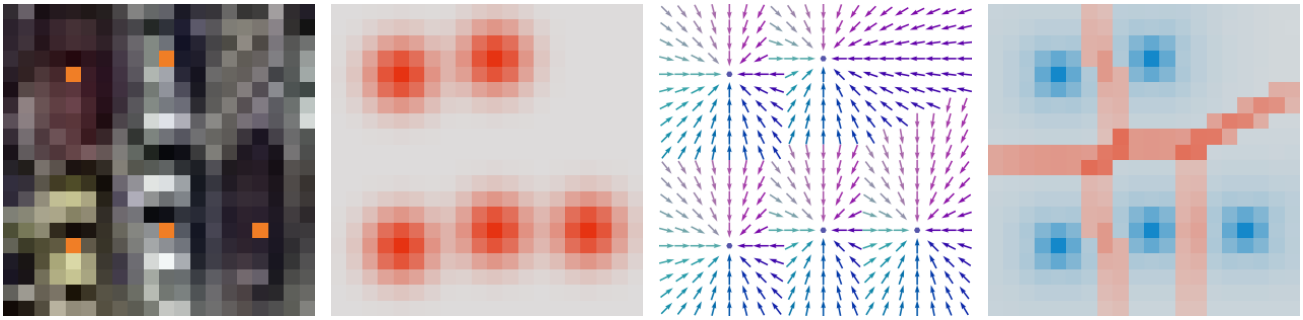
where the position and marks energies U_p and U_m measure the likelihood of object y according to the image X. While the size U_s, overlap U_o and alignment U_a energies serve as regularization priors on the points and their interaction with their neighborhoods $\mathcal{N}_y y = \{\tilde{y} \in Y \mid \tilde{y} \neq y, d(\tilde{y}, y) < d_{\max}\}$ (with d the Euclidean distance and $d_{\max}$ a maximum interaction distance).

The scalars $\omega_p, \ldots \omega_a$ weight the relative importance of each energy term for the total energy. Lastly, $\mathbb{1}_{U_p(X,y)<0}$ is the indicator function valued 1 when $U_p(X, y) < 0$ or 0 else, so that marks and priors energies are not contributing when the proposed object y is not likely to be at a good position*.

3.1 CNN for likelihood estimation

Classical MPP approaches use measures of contrast between the inside and the outside of each object.[15–17] These measures perform well on clearly contrasted images, but the varied visual aspect of objects and background within our application scope make them less reliable. We thus use CNNs to infer energy maps and build more versatile measures.

*This avoids negative energy priors to override the positive value of the point positioning energy (*ie.* low probability)



(a) (b) (c) (d)

Figure 2. For an image, with ground truth center overlaid in orange, (a); the corresponding heatmap (b), vector map (c) and divergence map (d). In (b) and (d): red> 0, blue< 0.

3.1.1 Position energy term

We train a Unet[23†] to infer an energy-map, so that for each pixel of the image we measure the likelihood of it being the center of an object. While this approach is similar to keypoint detection,[24] the objects of interest being closely packed, a straightforward heatmap inference (*ie.* inferring a map of centers dilated by a Gaussian filter, see Fig. 2b) leads to inferred objects/blobs connectivity thus making instances separation more difficult. Moreover, reducing the variance on the heatmap contradicts the uncertainty on center labeling while increasing the labeling imbalance over the whole image.

To tackle that issue we proceed as follows: for an image X of size $(h, w, 3)$[‡] we first infer a map of 2D vectors $\widehat{V}$ of size $(h, w, 2)$ pointing towards the closest object center (Fig. 2c)[§]. Centers of objects can then be identified by computing the divergence of $\widehat{V}$, as pixels with negative divergence (Fig. 2d).

Thus, for $y \in Y$ the position energy map $\widehat{A}_p$ is derived as such:

$$\widehat{A}_p = -2\left(\sigma\left(a.\mathrm{div}(\widehat{V}) + b\right) - t_p\right) + 1 \tag{3}$$

where a, b are two scalars learned along with the Unet model (see Sec. 4.1), and $t_p \in [0, 1]$ a detection threshold. The position energy of a point $y \in Y$is then sampled from map $\widehat{A}_p$ at position y_i, y_j:

$$U_p(X, y) = \widehat{A}_p[y_i, y_j] \tag{4}$$

3.1.2 Mark energy term

As for the position, we compute energy values for a mark of y (here s, r or α) given its position by sampling from an energy map inferred by a Unet.

For each mark $k \in \{s, r, \alpha\}$ its values are discretized into N_m value intervals. From the Unet we extract, given an image X of size $(h, w, 3)$, a tensor $\widehat{A}_k$ of size (h, w, N_m) so that each element over the last axis of the tensor corresponds to a value interval. The Unet is trained so that $\mathrm{Softmax}(\widehat{A}_k[y_i, y_j])$ corresponds to the likelihood distribution over each value interval at position y_i, y_j for mark k; *ie.* $\mathrm{Softmax}(\widehat{A}_k[y_i, y_j])[\mathrm{index}(y_m)]$ is trained to estimate $P(y_m|X, y_i, y_j)$, with $\mathrm{index}(y_m)$ the integer index corresponding to the value interval of y_m.

The mark energy is then derived as follows:

$$U_m(X, y) = \sum_{k \in \{s,r,\alpha\}} -\mathrm{Softmax}(\widehat{A}_k[y_i, y_j])[\mathrm{index}(y_m)] \tag{5}$$

[†]We implement the Unet as in Ref. 23 with added dropout and limiting the depth (number of pooling operations) from 4 to 3

[‡]Images with 3 color channels

[§]This was inspired from the approach in Ref. 25

This approach is akin to a classification task; while we could use a regression model to obtain the most likely mark value, this does not inform on the uncertainty or possible multiple modes of the likelihood density over values.

3.2 Priors on configurations

We define a few priors in order to regularize the inferred configurations

Size prior Enforces object area limits (between $a_{\min}$ and $a_{\max}$), *eg.* avoiding zero area objects:[21]

$$U_s(y) = \max\{a_{\min} - \text{area}(y), \text{area}(y) - a_{\max}, 0\} \tag{6}$$

Superposition prior Penalizes the overlapping of objects:[16]

$$U_o(\mathcal{N}_y y, y) = \max_{\tilde{y} \in \mathcal{N}_y y} \left\{ \frac{\text{area}(\tilde{y} \cap y)}{\min\{\text{area}(\tilde{y}), \text{area}(y)\}} \right\} \tag{7}$$

Alignment prior Rewards objects that align to their neighbors:[21]

$$U_a(\mathcal{N}_y y, y) = \min_{\tilde{y} \in \mathcal{N}_y y} \{-|\cos(|y_\alpha - \tilde{y}_\alpha|)|\} \tag{8}$$

4. TRAINING AND INFERENCE

Our model operates as follows:

- Training Unet models to infer position and mark energy maps, and adjusting of energy weights $\omega_p, \ldots, \omega_a$ (see Sec. 4.1).
- At inference, given an image X, inferring $\widehat{A}_p$ and $\widehat{A}_k$, $k \in \{s, r, \alpha\}$ once.
- Simulating the point process to infer $\widehat{Y}$ as the minimum of $U(X, \cdot)$ (see Sec. 4.2).

4.1 Training the position and mark CNNs

4.1.1 Position energy

The position energy model is trained to infer the position energy map by performing gradient descent to minimize the following loss function:

$$L_p(X, Y^+) = \text{MSE}(\widehat{V}, V^+) + \text{BCE}(B^+, \sigma(a.\text{div}(\widehat{V}) + b)) \tag{9}$$

with Y^+ the ground truth associated to image X, V^+ the vector map built from the ground truth, and B^+ a binary map of centers dilated by a Gaussian filter ($\sigma = 0.6$). MSE and BCE stand respectively for the Mean Squared Error and the Binary Cross Entropy.

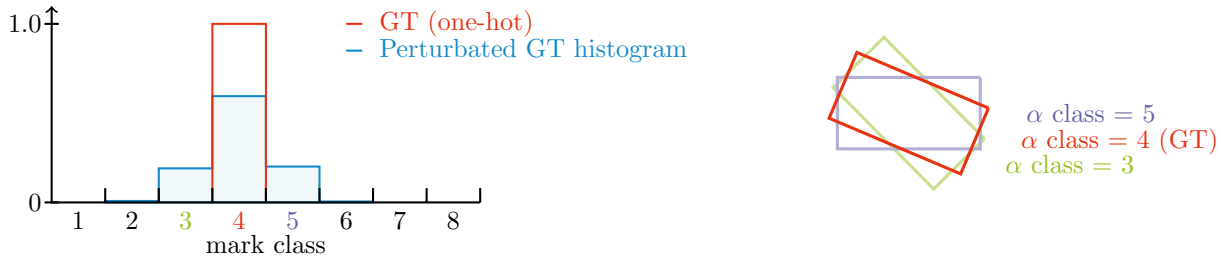


Figure 3. Toy example of ground truth mark perturbation with $N_m = 8$ and a perturbation of variance $\sigma = 0.6$. Left: in red the ground truth one-hot encoding, in blue the histogram of values taken by the perturbated ground truth. Right: some class values illustrated using the angle α as the example mark.

4.1.2 Mark energy

For each mark $k \in \{s, r, \alpha\}$ we minimize the loss :

$$L_k(X, Y^+) = \frac{1}{|P^+|} \sum_{p \in P^+} \text{CE}\left(\text{Softmax}(\widehat{A}_k[p]), \text{Softmax}(A_k^+[p])\right) \tag{10}$$

With P^+ denoting the set of pixels that belong to any object of interest. This way the loss is not computed on pixels that do not correspond to any object, *ie.* where the marks (size, ratio or angle) are not defined. The 1D vector of size N_m, $A_k^+[p]$ corresponds to the one-hot encoding of the ground truth mark at pixel p for mark k.

For large values of N_m (that allow more resolution over the mark discretization), the positive training labels in A_k^+ can become sparse. To alleviate this problem we perform a random perturbation on the ground truth at training by offsetting the ground truth class using a normal law. The benefit is twofold: first this models the inherent uncertainty on the ground truth labels (which gets more prominent as N_m gets bigger). Secondly it allows learning a proximity between classes, so that for example, class prediction errors equal to one are less penalized on average than errors superior to 1. We illustrate this procedure on a simplified example in Fig. 3 with a low N_m.

4.2 Inference: point process simulation

Once the energy model is defined, we simulate the point process to find the configuration that minimizes the total energy:

$$\widehat{Y} = \underset{Y \in \mathcal{Y}}{\text{argmin}} U(X, Y) \tag{11}$$

We simulate the MPP of density $\exp(-U(X, \cdot)/T)$, with temperature T decreasing geometrically to perform simulated annealing[26] and collapse the density to a single mode corresponding to a minimum. We use a Reversible Jump Monte Carlo Markov Chain[27] (RJMCMC). The RJMCMC extends the Metropolis Hastings algorithm, allowing jump between state-spaces of different dimensions. Convergence is ensured given we use at least a uniform birth-and-death kernel in $S \times M$.

Moreover, we add non-uniform birth-and-death and translation/rotation/scaling kernels that propose points using the densities derived from the potential $U_p(X, \cdot) + U_m(X, \cdot)$.[16] These densities are easy to sample as they only depend on pre-computed energy maps. The kernel reversibility (necessary condition to convergence) is proven in Ref. 16.

5. EXPERIMENTAL RESULTS

5.1 Data

DOTA50 We aim at performing detection in images from satellites such as Pléiades[¶] or CO3D[‖]. Thus, we restrict the spatial resolution to 50 cm/pixel. We use the DOTA dataset.[28] As it is from various sources and

[¶]Pléiades Constellation [pleiades.cnes.fr]

[‖]Constellation Optique 3D [intelligence-airbusds.com]

varying resolution, we select all images with resolution equal or higher to the target 50 cm/pixel resolution and perform subsampling to reach the desired resolution. The labels of the DOTA dataset consist of oriented rectangles. Moreover, as we only focus on land vehicles detection we only keep the *small-vehicle* and *large-vehicle* classes.

COWC50 and Airbus data We also test our method on the COWC[29] dataset (subsampled), and some data provided by Airbus Defense and Space. Since the ground truth is insufficient (COWC has only centers labeled) or unavailable (in the case of the Airbus data), we use the model trained on the DOTA dataset when testing on the COWC50 and Airbus images. This data allows to test the generalization power of models as we do not train our models on the COWC50 and Airbus data.

5.2 Parameters

The following weights are used for the energy model, selected by trial and error on the training data: $\omega_p = 0.4, \omega_m = 0.1, \omega_s = 0.1, \omega_o = 0.3, \omega_a = 0.025$. We set $N_m = 32$ as a compromise between the size of energy maps $\widehat{A}_k$ ($k \in \{s, r, \alpha\}$), and the mark distribution resolution.

5.3 Qualitative and quantitative evaluation

We train our model on two thirds of the DOTA50 data and compute metrics[**] on the remaining data. The IOU threshold is set to 0.25 and 0.5 successively. For BBA-Vectors[6] the detection threshold is $s = \text{argmax}_{s\in[0,1]}\ \text{F1}(s)$[††].

Results on sample images are shown in Figure 4. While both BBA-Vectors and our model perform good detection of vehicles in DOTA50, it is to note that our model outputs object detections that are more coherent with their neighborhood: we notice far fewer misaligned detections than in BBA-Vectors This is explained by the priors added within the energy model of our method.

Table 1. Metrics for oriented bounding boxes detection on DOTA. Pr@t: precision for IOU threshold t, Rc@t: recall

Method	F1@0.25	Pr@0.25	Rc@0.25	F1@0.5	Pr@0.5	Rc@0.5
BBA-Vec.[6]	**0.68**	0.63	0.74	0.48	0.43	0.53
BBA-Vec. (50% train data)	0.58	0.55	0.62	0.23	0.22	0.25
BBA-Vec.(25% train data)	0.52	0.51	0.54	**0.13**	0.12	0.15
MPP+CNN	**0.66**	0.56	0.79	0.42	0.32	0.64
MPP+CNN (50% train data)	0.57	0.46	0.75	0.31	0.22	0.52
MPP+CNN (25% train data)	0.55	0.48	0.64	**0.34**	0.26	0.49

Metrics listed in Table 1 show that our model achieves an F1 score similar to BBA-Vectors when training on the whole available training dataset.

As being able to train a model on fewer data is of practical interest – high quality labeled data in remote sensing can be hard and costly to procure – we also compare the performance when training with fewer data: we train both our and BBA-Vectors methods on 100%, 50% and 25% of the training set and compute metrics over the same constant testing set. Both models have their performance decrease as the amount of data shrinks.

It is to note that, at low IOU thresholds, for a positive match between detection and ground truth to be counted, it is enough to have a decent positioning of the detection with a rough shape fitness (*ie.* approximate size, ratio and angle). On the contrary, with higher threshold, the fine matching of shape of the inferred object with the ground truth becomes more important for the matching (as ground truth and inference need to have more overlap).

[**]Since the point process simulation yields a single configuration with a score over the whole set of points rather than independent scores per point (as classical CNN methods do) the mean average precision can not be computed for the MPP method.

[††]With F1(s) the F-Score for detection threshold s

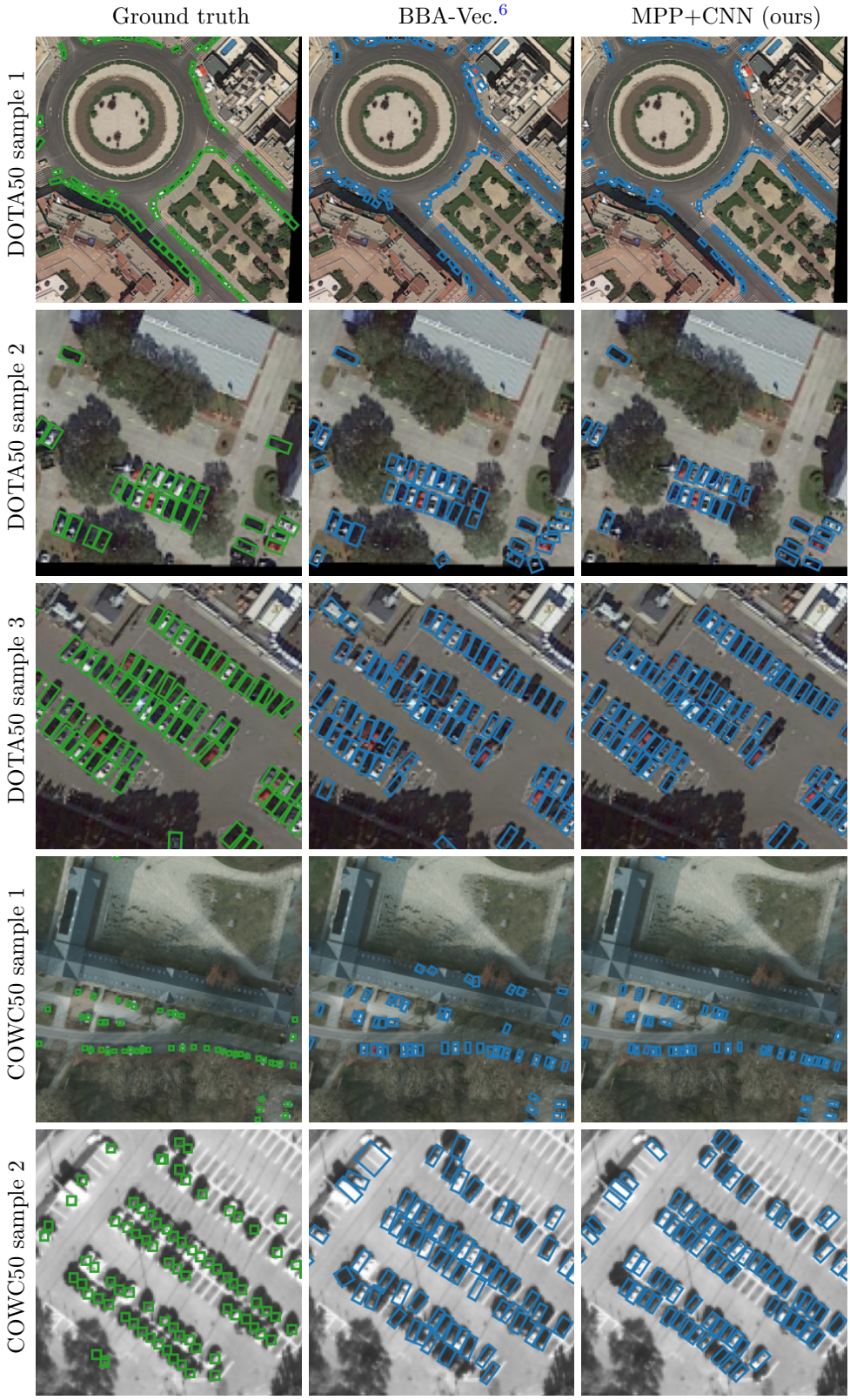


Figure 4. Detection results on samples of the testing sets for DOTA and COWC (GT on COWC is being shown as squares since the labeling is only center positions)

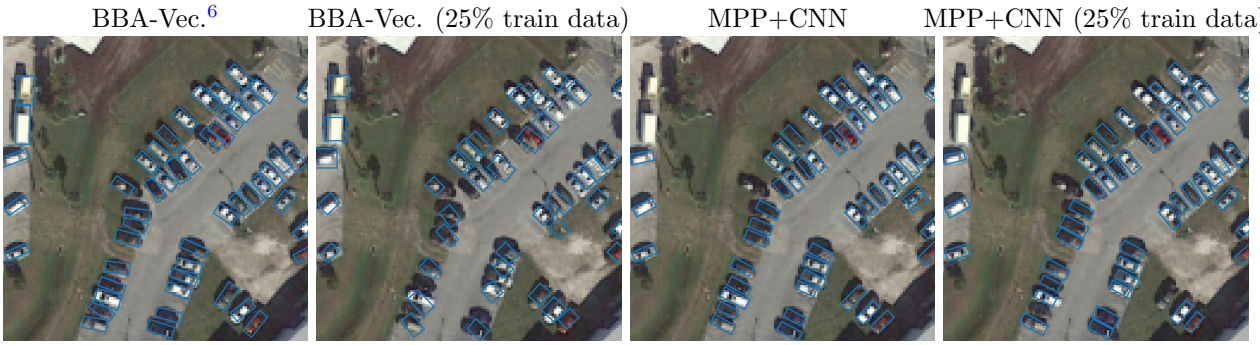


Figure 5. Example of decreasing shape accuracy on BBA-Vec. when reducing training data.

Figure 6. Detection results on samples of the data provided by Airbus, aerial photos subsampled to 50 cm/px, no ground truth available. [copyright Airbus Defense and Space]

We notice in Table 1 a sharp decrease in performance for BBA-Vectors with fewer training data with the higher 0.5 IOU threshold. Fig. 5 confirms that this is due to a loss in inferred shape precision ; BBA-Vectors on 25% training data shows bad angle, size and ratio estimation, while maintaining a decent localization of objects in the figure. On the contrary, our model does not suffer from this loss of precision with fewer training data. This could be explained by regularization performed via the priors and the lower number of training parameters for our method ($\sim 2M$ for our model, $\sim 70M$ for BBA-Vectors) that would help to avoid overfitting.

Lastly we show inference results on some data provided by Airbus in Fig. 6. The image samples on the first line show an interesting example where priors allow compensating for lack of visual information caused by the shadow cast by the building.

6. CONCLUSION AND FUTURE WORKS

We propose a method based on marked point process combined with convolutional neural networks. The point process framework allows accounting for priors on interactions, while the CNN bring versatility and adaptability to complex visual settings. Not only our method offers metrics similar to current CNN-based approaches, it also increases regularity within inferred configurations of points. Furthermore, it requires few trainable parameters compared to typical CNN methods for object detection. Finally, we are working on applying our method to image sequences, where our model will make use of object dynamic priors as well as spatial interaction priors.

APPENDIX : IMPLEMENTATION

The code was implemented with Python, using PyTorch for the neural network models. It is available at https://github.com/Ayana-Inria/MPP_CNN_RS_object_detection. The source data for DOTA and COWC is publicly available, and the code used to transform it to our specifications is provided in the above-mentioned repository.

ACKNOWLEDGMENTS

Thanks to BPI France (LiChiE contract) for funding this research work, and to the OPAL infrastructure from Université Côte d'Azur for providing computational resources and support.

REFERENCES

[1] LaLonde, R., Zhang, D., and Shah, M., "ClusterNet: Detecting small objects in large scenes by exploiting spatio-temporal information," in [*IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*], (2018).

[2] Ren, S., He, K., Girshick, R., and Sun, J., "Faster R-CNN: Towards real-time object detection with region proposal networks," in [*Advances in Neural Information Processing Systems*], Cortes, C., Lawrence, N., Lee, D., Sugiyama, M., and Garnett, R., eds., **28**, Curran Associates, Inc. (2015).

[3] Redmon, J., Divvala, S., Girshick, R., and Farhadi, A., "You Only Look Once: Unified, Real-Time Object Detection," in [*2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*], 779–788, IEEE (2016).

[4] Lin, T.-Y., Goyal, P., Girshick, R., He, K., and Dollar, P., "Focal loss for dense object detection," in [*IEEE International Conference on Computer Vision (ICCV)*], (2017).

[5] Hou, L., Lu, K., and Xue, J., "Refined one-stage oriented object detection method for remote sensing images," *IEEE Transactions on Image Processing* , 1–1 (2022).

[6] Yi, J., Wu, P., Liu, B., Huang, Q., Qu, H., and Metaxas, D., "Oriented object detection in aerial images with box boundary-aware vectors," in [*IEEE/CVF Winter Conference on Applications of Computer Vision*], 2150–2159 (2021).

[7] Wu, Z.-Z., Wang, X.-F., Zou, L., Xu, L.-X., Li, X.-L., and Weise, T., "Hierarchical object detection for very high-resolution satellite images," *Applied Soft Computing* **113** (2021).

[8] Li, X., Men, F., Lv, S., Jiang, X., Pan, M., Ma, Q., and Yu, H., "Vehicle Detection in Very-High-Resolution Remote Sensing Images Based on an Anchor-Free Detection Model with a More Precise Foveal Area," *ISPRS International Journal of Geo-Information* **10**(8), 549 (2021).

[9] Yang, Y., Tang, X., Cheung, Y.-M., Zhang, X., Liu, F., Ma, J., and Jiao, L., "AR^{2}Det: An Accurate and Real-Time Rotational One-Stage Ship Detector in Remote Sensing Images," *IEEE Transactions on Geoscience and Remote Sensing* , 1–14 (2021).

[10] Sun, Y., Ran, J., Yang, F., Gao, C., Kurozumi, T., Kimata, H., and Ye, Z., "Oriented Object Detection For Remote Sensing Images Based On Weakly Supervised Learning," in [*2021 IEEE International Conference on Multimedia Expo Workshops (ICMEW)*], 1–6 (2021).

[11] Zhao, T., Liu, N., Celik, T., and Li, H.-C., "An Arbitrary-Oriented Object Detector Based on Variant Gaussian Label in Remote Sensing Images," *IEEE Geoscience and Remote Sensing Letters* , 1–5 (2021).

[12] Chen, X., Ma, L., and Du, Q., "Oriented Object Detection by Searching Corner Points in Remote Sensing Imagery," *IEEE Geoscience and Remote Sensing Letters* , 1–5 (2021).

[13] Liu, Z., Hu, J., Weng, L., and Yang, Y., "Rotated region based CNN for ship detection," in [*IEEE International Conference on Image Processing (ICIP)*], 900–904 (2017).

[14] Maggiori, E., Tarabalka, Y., Charpiat, G., and Alliez, P., "Convolutional Neural Networks for Large-Scale Remote-Sensing Image Classification," *IEEE Transactions on Geoscience and Remote Sensing* **55**(2), 645–657 (2017).

[15] Craciun, P., Ortner, M., and Zerubia, J., "Joint detection and tracking of moving objects using spatio-temporal marked point processes," in [*IEEE Winter Conference on Applications of Computer Vision*], 177–184 (2015).

[16] Lacoste, C., Descombes, X., and Zerubia, J., "Point processes for unsupervised line network extraction in remote sensing," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **27**(10), 1568–1579 (2005).

[17] Descombes, X., "Multiple objects detection in biological images using a marked point process framework," *Methods* **115**, 2–8 (2017).

[18] Kulikova, M. S., Jermyn, I. H., Descombes, X., Zhizhina, E., and Zerubia, J., "Extraction of arbitrarily-shaped objects using stochastic multiple birth-and-death dynamics and active contours," in [*Computational Imaging VIII*], **7533**, 58–64, SPIE (2010).

[19] Pham, T. T., Hamid Rezatofighi, S., Reid, I., and Chin, T.-J., "Efficient Point Process Inference for Large-Scale Object Detection," in [*IEEE Conference on Computer Vision and Pattern Recognition*], 2837–2845 (2016).

[20] Zhou, J., Proisy, C., Descombes, X., le Maire, G., Nouvellon, Y., Stape, J.-L., Viennois, G., Zerubia, J., and Couteron, P., "Mapping local density of young Eucalyptus plantations by individual tree detection in high spatial resolution satellite images," *Forest Ecology and Management* **301**, 129–141 (2013).

[21] Ortner, M., Descombes, X., and Zerubia, J., "A Marked Point Process of Rectangles and Segments for Automatic Analysis of Digital Elevation Models," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **30**(1), 105–119 (2008).

[22] Lieshout, M.-C. V., [*Markov Point Processes and Their Applications*], Imperial College Press (2000).

[23] Ronneberger, O., Fischer, P., and Brox, T., "U-Net: Convolutional Networks for Biomedical Image Segmentation," in [*Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*], Navab, N., Hornegger, J., Wells, W. M., and Frangi, A. F., eds., *Lecture Notes in Computer Science*, 234–241 (2015).

[24] Cao, Z., Simon, T., Wei, S.-E., and Sheikh, Y., "Realtime Multi-Person 2D Pose Estimation Using Part Affinity Fields," in [*IEEE Conference on Computer Vision and Pattern Recognition*], 7291–7299 (2017).

[25] Neven, D., Brabandere, B. D., Proesmans, M., and Van Gool, L., "Instance Segmentation by Jointly Optimizing Spatial Embeddings and Clustering Bandwidth," in [*IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*], 8829–8837, IEEE (2019).

[26] Guilmeau, T., Chouzenoux, E., and Elvira, V., "Simulated annealing: A review and a new scheme," in [*IEEE Statistical Signal Processing Workshop (SSP)*], 101–105 (2021).

[27] Green, P. J., "Reversible jump Markov chain Monte Carlo computation and Bayesian model determination," *Biometrika* **82**(4), 711–732 (1995).

[28] Xia, G.-S., Bai, X., Ding, J., Zhu, Z., Belongie, S., Luo, J., Datcu, M., Pelillo, M., and Zhang, L., "Dota: A large-scale dataset for object detection in aerial images," in [*IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*], (2018).

[29] Mundhenk, T. N., Konjevod, G., Sakla, W. A., and Boakye, K., "A Large Contextual Dataset for Classification, Detection and Counting of Cars with Deep Learning," in [*Computer Vision – ECCV 2016*], Leibe, B., Matas, J., Sebe, N., and Welling, M., eds., *Lecture Notes in Computer Science*, 785–800 (2016).