GEOAI FOR MARINE ECOSYSTEM MONITORING: A COMPLETE WORKFLOW TO GENERATE MAPS FROM AI MODEL PREDICTIONS

J. Talpaert Daudon[1]*, M. Contini[2], I. Urbina-Barreto[3], B. Elliott[4], F. Guilhaumon[3], A. Joly[5], S. Bonhommeau[2], J. Barde[6],

[1] UMR Marbec, IRD, La Réunion, France - justine.daudon@ird.fr
[2] Ifremer DOI, La Réunion, France - (matteo.contini, sylvain.bonhommeau)@ifremer.fr
[3] UMR Entropie (Future Maore Reefs), IRD, La Réunion, France - (isabel.urbina-barreto, francois.guilhaumon)@ird.fr
[4] Duke University, Duke Marine Lab, Beaufort, NC, USA - brianna.elliott@duke.edu
[5] INRIA Zenith, Montpellier, France - alexis.joly@inria.fr
[6] UMR Marbec, IRD, France - julien.barde@ird.fr

KEY WORDS: GeoAI, computer vision, deep learning, marine ecology, object detection, segmentation, geospatial, photogrammetry

ABSTRACT:

Mapping and monitoring marine ecosystems imply several challenges for data collection and processing: water depth, restricted access to locations, instrumentation costs or weather constraints for sampling, among others. Nowadays, Artificial Intelligence (AI) and Geographic Information System (GIS) open source software can be combined in new kinds of workflows, to annotate and predict objects directly on georeferenced raster data (e.g. orthomosaics). Here, we describe and share the code of a generic method to train a deep learning model with spatial annotations and use it to directly generate model predictions as spatial features. This workflow has been tested and validated in three use cases related to marine ecosystem monitoring at different geographic scales: (i) segmentation of corals on orthomosaics made of underwater images to automate coral reef habitats mapping, (ii) detection and classification of fishing vessels on remote sensing satellite imagery to estimate a proxy of fishing effort (iii) segmentation of marine species and habitats on underwater images with a simple geolocation. Models have been successfully trained and the models predictions are displayed with maps in the three use cases.

1. INTRODUCTION

The world's oceans are concurrently affected by anthropogenic activities and climate change impacts (Lyu et al., 2021, Hoegh-Guldberg et al., 2017). Mapping and monitoring marine ecosystems are key to improve understanding of ecosystems globally, minimize these impacts, and guide ecosystem conservation and restoration. (Westoby et al., 2020, Anthony et al., 2017). However, monitoring marine ecosystems imply several challenges for data collection and processing: water depth, restricted access to locations, instrumentation costs and weather constraints for sampling. Nowadays, artificial intelligence (AI) and Geographic Information System (GIS) open source software can be combined in new kinds of workflows, to generate, among others, marine habitat maps from deep learning models predictions. While it has been suggested that at least 80% of all data are geographic in nature (VoPham et al., 2018), AI is a relevant and powerful tool to assist (by automated labeling) the ecological analyses associated to temporal and spatial ecosystem surveys (Hopkinson et al., 2020, Pavoni et al., 2022, Yuval et al., 2021). Nevertheless, one of the major issues for geoAI consists in tailoring usual AI workflow to better deal with spatial data formats used to manage both vector annotation and large georeferenced raster images (e.g. orthomosaics, drone or satellite images). A critical goal consists in enabling computer vision models to be trained directly with spatial annotations (Touya et al., 2019, Courtial et al., 2022), as well as delivering model predictions through spatial data formats to automate the production of marine maps from raster data. Moreover, another goal is addressing large raster constraints (whose size exceeds the GPU cache memory) in terms of machine resources for training

---
* Corresponding author

deep learning models. In this paper, we describe and share the code of a generic method used to annotate and predict objects within georeferenced images. This has been achieved by setting up a workflow which relies on the following process steps: (i) spatial annotation of raster images by editing vector data directly within a GIS, (ii) splitting large raster images (orthomosaics, satellite images) into tiles to fit available machine resources, and while keeping raster (images) and vector (annotation) quality unchanged, (iii) training of deep learning models (CNN) thanks to the transfer learning strategy (iii) model predictions delivered in spatial vector formats. Here, we demonstrate that open source tools can be used to develop a workflow capable of automating map production using deep learning models trained on georeferenced images and spatial and non-spatial (pixel-value) annotations. Also, we test the impact of three data processing and model training strategies on model accuracies : (i) overlapping tiles or non-overlapping tiles, (ii) sizes of raster tiles (500x500 and 1500x1500 pixels), (iii) two pre-trained models.

The whole framework relies on `Python` libraries for both geospatial processing and AI and is shared on GitHub and has been assigned a DOI on Zenodo, along with sample data. Moreover, a QGIS plugin is available to facilitate the use of pre-trained deep learning models to automate the production of maps from raster data (e.g. underwater orthomosaics or satellite images).

2. MATERIAL AND METHODS

2.1 Input data

The current workflow is meant to process different types of geospatial data : raster data (e.g. orthomosaics or remote sens-

Datasets	Deepmosaics	Gillnet	Seatizen
Dataset type	orthomosaic	satellite imagery	simple georeferenced images
Annotation type	vector polygons	pixel-value bounding boxes	pixel-value instance segmentation polylines
Annotation tool	Qgis	Biigle	Biigle
Computer vision task	Instance segmentation	Object detection	Instance segmentation
Categories	49	3	41
Annotated samples	3	833	1200
Size (px)	21392 - 32097 x 14879 - 30990	8192 x 5452	3648 x 2736

Table 1. Description of the datasets used to implement the presented workflow

ing satellite imagery) or underwater images associated with a unique spatial coordinate. The tool is built to support rasters with as much channels as they have (RGB, multispectral...). AI annotation process is based on training supervised deep learning models with manually annotated images. The workflow has been designed to handle both geospatial (vector annotations) and non-geospatial annotation (pixel-values annotations) formats.

This workflow has been implemented in three use cases related to marine ecology and based on different types of computer vision tasks, images and annotations (Table 1). The choice of the annotation tool is left to the annotator and does not generate any constraint for the data processing because these formats are then converted into a reference format : the COCO format (Lin et al., 2015) commonly used to train computer vision models.

2.1.1 Deepmosaics Orthomosaics of three coral reef sites (about 6m depth, total surface area around 620 m²) on Mayotte lagoon were performed. A consistent underwater photogrammetry protocol by structure from motion has been used to collect the images using SCUBA (upper than 70% overlapping between images). Ecological analyses were conducted using QGIS (QGIS Development Team, 2009), (version 3.24.1) to describe reef-building corals, colonies are manually delineated as polygons (by drawing edges of colonies) considering an individual as a colony growing independently from its neighbor. Each colony was classified by genus and species.

The resulting dataset has 49 classes and 27 were underrepresented. Indeed, these classes had less than 50 occurrences per class while 119 was the mean number of occurrences per class in the full dataset. Annotations related to these species were removed from the dataset for training and testing. Despite this choice, the dataset is still very unbalanced : the most represented class had 867 occurrences and the less represented class had 54 occurrences.

2.1.2 Gillnet The overall goal of this work is to better document and describe tuna drift gillnet fleets in the northern Indian Ocean using satellite imagery, including the number of gillnet vessels and their characteristics (e.g. vessel length, presence or absence of gear on board). This work focuses on the Pakistani tuna drift gillnet fleet as a case study, given the ongoing and dedicated monitoring of the gillnet fleet by WWF Pakistan, which has provided a dearth of information to supplement our satellite analysis (Kiszka et al., 2021).

Following consultation with WWF Pakistan, polygons within three major fishing harbors were selected in Pakistan as the areas of interest (AOI) to collect satellite imagery: Karachi, Gwadar, and Pishukan. Two sources of satellite imagery are being used for this analysis: freely and widely-available Google Earth Pro and WorldView-3. The WorlView-3 portion is currently ongoing, so we only focus on Google Earth Pro in this document. We reviewed all publicly available satellite imagery

Figure 1. Example of *Seatizen* platforms

from Google Earth Pro from January 2021 to December 2022 that was available at 700-750 feet digital elevation through January 25, 2023. We used Google Earth Pro's "save image" feature and downloaded imagery at the highest resolution available.

To annotate the images, we selected BIIGLE 2.0 as our image annotation software (Langenkämper et al., 2017). We categorized bounding boxes with three categories for image annotation: *yes*, *maybe*, and *no*. A vessel labeled as *yes* indicated that the analyst detected it to be a gillnet vessel; *maybe* referred to vessels that had the shape and other defining features of a tuna drift gillnet but could not definitively be categorized as a gillnet vessel due to image quality or similarity to other gear (e.g. trawls); and *no* referred to vessels which were definitely not gillnet vessels, such as water supply vessels.

Although the *Gillnets* dataset contains only 3 classes, the dataset is very unbalanced. For the training, the dataset has been rebalanced between classes by deleting images which contain only the most represented class (e.g. *no*) resulting in an even lower number of images (262 samples after rebalancing against 564 in the initial training set).

2.1.3 Seatizen This project proposes a new approach to monitor underwater species. The methodology explores the possible use of data collected from citizens practicing water sports (kitesurfing, paddle, snorkeling, etc.) in order to build a participatory science project: Seatizen. Data acquisition was carried out using instrumented marine platforms. These platforms can be divided into two groups, citizen platforms and scientific platforms. The first type of platforms (paddles, kitesurf and masks) are designed for being used by citizens practicing marine sports. These platforms are equipped with a camera and, most of the time, with a differential GPS module allowing the acquisition of georeferenced images with centimeter accuracy (Figure 1 on the left). The second type of platform is an autonomous board, developed by Ifremer as part of the IOT (Indian Ocean sea Turtles) project, equipped with a GPS module and a GoPro (Figure 1 on the right).

This dataset currently contains images collected in the Indian Ocean and, at its latest update, consists of images sampled from Reunion island, Mauritius, Europa island, Aldabra, Saint-Brandon and Mayotte. Images can be georeferenced, some are annotated or not. Annotations were made by indicating the

presence or absence of 41 classes including corals, fauna associated with corals, algae, marine plants and classes introduced specifically to describe a participatory science problem. Images were taken between 2015 and 2023 and present different quality and resolution and among these images 1200 were annotated using the instance segmentation technique by an expert in marine biology.

This dataset is stronlgy unbalanced between classes. Classes that had less than 20 occurrences per class were removed for the model training and testing processes.

2.2 Tile georeferenced imagery

Depending on machine resources (such as GPU cache storage) available to user, the process to prepare the data for model trainings can be adapted.
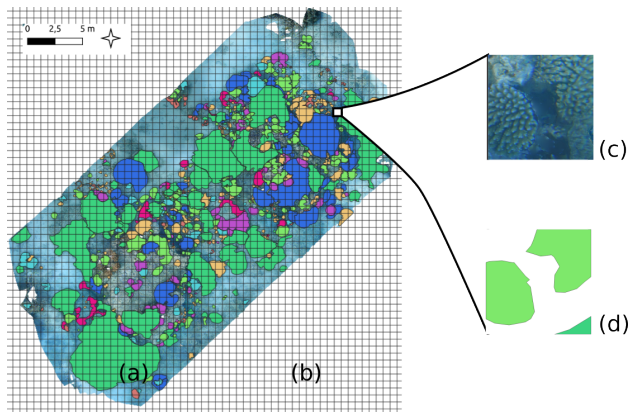
Figure 2. Tiling process on annotated orthomosaics (a). Tiles are cut according to a regular grid (b) and produces georeferenced tiles (c) with their matching annotations (d).

Figure 2 presents a method to handle large raster data by slicing them into tiles. Indeed, the orthomosaic exposed is large (1.6 GB) and cannot be supported as is in GPU memory for training a deep learning model. The size of the tiles (defined in pixels or in meters) is defined by the user according to his constraints (available machine resources, size of the annotated objects or sampled surface). Both wide geospatial raster data and related vector annotation data are split into a large number of raster tiles (for instance, 500 x 500 pixels) along with smaller vector files sharing the exact same boundaries as the raster tiles (converted in GeoJSON files). This tiling process can then be used either in the training phase of the model or in inference. The workflow also offers the possibility to use different tile cutting strategies. The tiles can be cut according to a regular cutting grid (without overlaps), or by allowing the superposition of some tiles. Figure 3) introduces the two tiling strategy tested. On this example, a regular grid produces 9 tiles without overlapping whereas splitting with 25% overlap between tiles produces 16 tiles for the same tiled area. The degree of overlap can be set by the user. The superposition of tiles leads, in some cases, to the presence of the same object on several tiles and carries out a data-augmentation process that can be relevant depending on the model training or inference strategies.

2.3 Train a deep learning model

The workflow presented here can be used with both spatial and non-spatial annotations and supports different annotation
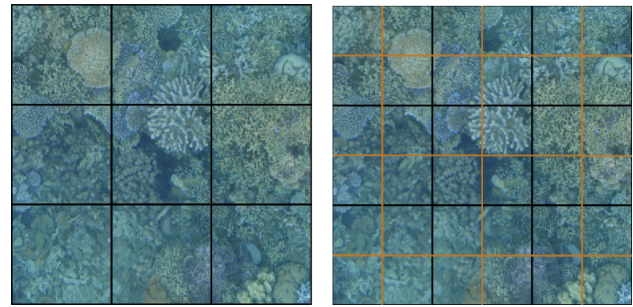

Figure 3. Tiling strategies for georeferenced rasters : a regular grid (left) and an overlapping grid (right).

formats. The annotations and raster formats generated by the workflow when slicing large datasets are fully compatible with the dataloading processes available in `Pytorch` (Paszke et al., 2019) or `Detectron2` (Wu et al., 2019a). Indeed, the annotations are converted to the COCO format (Figure 4) widely used to train deep learning models. Thus, the geospatial masks or bounding boxes are converted into pixel values on the image thanks to an inverse affine transformation.

Models are trained using the transfer learning technique (Weiss et al., 2016). This method reuses the weights of the pre-trained models on large datasets (ImageNet in our case) and trains the last layers of these models specifically on the available data.

The models used as a backbone were chosen from the `Detectron2` models' benchmark (Wu et al., 2019b). Models adapted to each computer vision task (instance segmentation or object detection) were chosen according to their average accuracy ranking. The best were chosen for a test on our data. Models were trained with 1 GPU (Quadro RTX 4000), 32 cpus and 64GB of RAM, using 80% of the dataset. 10% of the dataset were kept for the validation step and 10% for the testing step. Models were evaluated using the COCO-style evaluation (Lin et al., 2015). The average precision (AP) and average precision at 50% of intersection over union (AP50) will be the metrics used to evaluate the models trained.
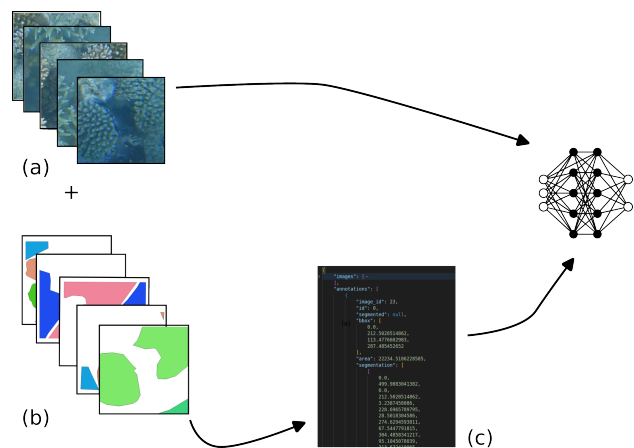

Figure 4. Labeled raster tiles (.tif format) (a) and vector tiles of annotation (.geojson format) (b). COCO file containing annotations (c) to train models.

2.4 Spatialized model predictions

Our workflow uses the affine transformation to convert the annotation results of the model into geometric shapes holding spa-

tial coordinates that match the predicted image. The model predictions are merged according to their class to form closed polygons or bounding boxes.

The annotations provided by the trained deep learning model are produced in the same Coordinate Reference System (CRS) as the input image. The annotations can be exported in a geospatial format such as geojson, shapefile or geopackage in order to be displayed on maps with GIS software.

2.5 Implementation

The workflow has been built with open source `Python` packages. Main geospatial packages used are : `geopandas` (Jordahl et al., 2022), `rasterio` (Gillies et al., 2015) and `solaris` (CosmiQ, 2020). `detectron2` (Wu et al., 2019a) is the package that provided pre-trained computer vision models. Predictions of the computer vision models were centralized using the `Fiftyone` package (Moore and Corso, 2020). The whole workflow to process rasters and simple georeferenced images, train and evaluate deep learning models, spatialize model's predictions is available under a `pip` package (Talpaert Daudon, 2023a) and a DOI has been assigned along with tutorials (Talpaert Daudon, 2023b) and sample data. Moreover, a QGIS plugin (Talpaert Daudon, 2023c) performs inference with the models trained on the three datasets (Section 2.1). It is possible to select a geographic area and produce spatial occurrences of objects or marine species there.

3. EXPERIMENTS AND RESULTS

3.1 Orthomosaics

Three deep learning models have been trained on this dataset using different data processing strategies. A Mask-RCNN (X101-FPN pre-trained model) was chosen as a backbone for the transfer learning strategy. First, we trained models on tiles of 500 pixels by 500 pixels and a regular grid cut. Second, we trained a model with tiles of the same size but cut according to a grid with a 25% overlap between adjacent tiles. Finally, we repeated these two operations with tiles of 1500 pixels x 1500 pixels (Table 2).

Model	AP (%)	AP50 (%)
Tiles 500 px no overlap	16,2	26,4
Tiles 500 px overlap 25%	17,6	26,7
Tiles 1500 px overlap 25%	18,7	31,7

Table 2. Models metrics for the three tiling strategies explored.

These models have been then applied to a manually unannotated orthomosaic and the results of these models were spatialized (Figure 5).

3.2 Satellite imagery

Models have been trained on tiles (8000 pixels per 5000 pixels) split with a regular grid. Two pre-trained models were implemented in the training process : a X101 pre-trained on ImageNet and a R50 pre-trained on PASCAL VOC object detection. The X101 achieved 69.9% and the R50 achieved 71% of AP50 after training on a balanced datasets including 261 samples. Trained models were used to detect gillnet fishing vessels (Figure 6).
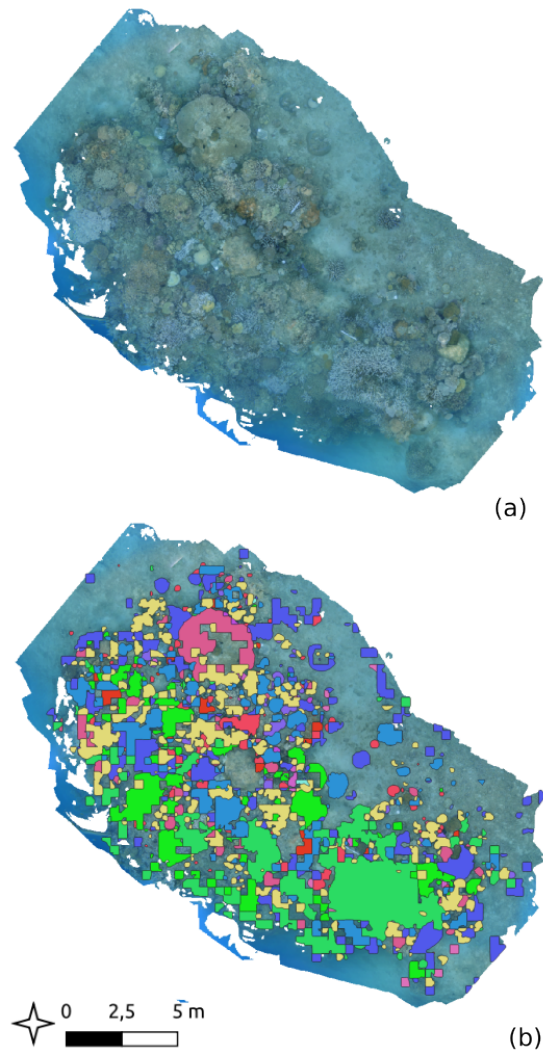

(a)

(b)

Figure 5. A non-annotated orthomosaic (a) and the same orthomosaic where coral species are detected and classified (b) by a deep learning model trained on 500 by 500px tiles split with a regular grid (model called 'Tiles 500 px no overlap').
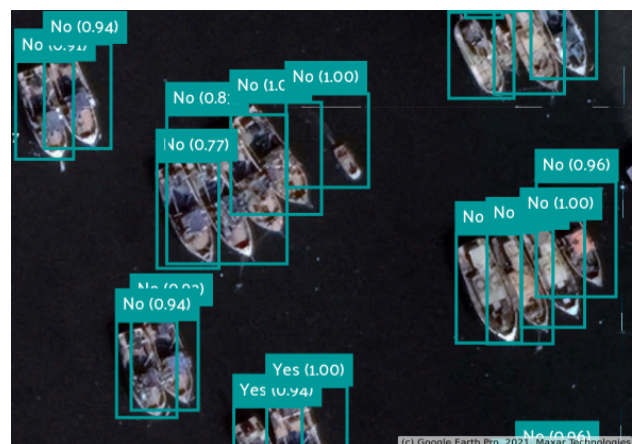

Figure 6. Predictions with model's confidence from an X101 model on a satellite image.

3.3 Georeferenced images

We fine tuned a MaskRCNN-R50 achieving 12% of AP and 22% of AP50. Predictions were matched with the geolocation belonging to the images predicted (Figure 7) and maps could be created to monitor marine corals in the area (Figure 8).

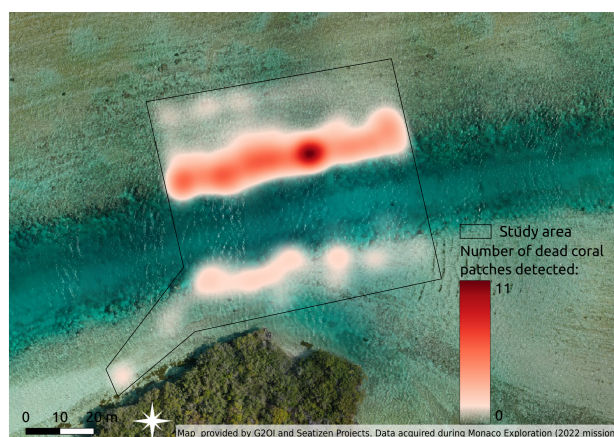Figure 7. Model's prediction of a dead coral patch on a georeferenced image.

Figure 8. Model 's predictions of dead coral patches on georeferenced images within a study area.

4. DISCUSSION AND PERSPECTIVES

Extending deep learning frameworks to geospatial data has already been implemented in deep learning model training processes(Soliman and Terstriep, 2019, Stewart et al., 2022, CosmiQ, 2020). While some tools provides an end-to-end process to train and predict on rasters (Beilschmidt et al., 2023, GeoAlert, 2021) there is a need of free and open source software to automatically produce maps. Here, we described a package that combines new pre-trained computer vision models from `detectron2` (Wu et al., 2019b) package and achieves automatic maps production.

4.1 Input data types

We have widened the spectrum of input data by offering not only deep learning on satellite images (CosmiQ, 2020, Stewart et al., 2022, Soliman and Terstriep, 2019) but also on orthomosaics covering small areas or underwater images associated with a single GPS point.

Furthermore, the need for multiple disciplines to be involved in the geoAI process has already been identified (VoPham et al.,

2018). Indeed, the expertise of cross-disciplinary skills from the application field (e.g. marine ecology), data science and data engineering is necessary to establish best practices for how to deal with the complexity of geospatial and environmental data. The ubiquity of the input data assimilated by this workflow and the implementation into three research projects makes this code robust for different user profiles, whatever the data they have.

We have worked on various data sources but our workflow is only suitable for 2D images. However, photogrammetry is a new tool for precisely measuring key parameters for monitoring coral reefs (Urbina-Barreto et al., 2022) and generates 3D models that this workflow cannot handle so far. Future work could be done to spatialize the predictions on 3D models using digital elevation model (DEM) or directly training a deep learning model on the mesh (Hopkinson et al., 2020, Pierce et al., 2021).

4.2 Annotation tools

The workflow was tested on manually annotated input data using geospatial tools (QGIS) or non-spatial annotation tools commonly used in computer vision. The annotations coming from spatialized or non-spatialized annotation tools did not generate any obstacles in the training of the models. On the one hand, annotation of rasters via QGIS allows to benefit from functionalities adapted to spatial data such as the native support of data in geotiff format and associated metadata by using vector data formats to manage annotations. On the other hand, computer vision annotation tools (Biigle, CVAT (CVAT.ai Corporation, 2022)) provide functionalities (Sager et al., 2021) that speed up object segmentation by implementing pre-trained models (like segment anything (Kirillov et al., 2023)) and allows the creation of collaborative annotation projects with different users. Both products of annotations tools are handled in this workflow. This makes the use of annotations versatile and non-restrictive for users both familiar and unfamiliar with geospatial software.

4.3 Tiling strategies

Grid functionalities for tiling were already explored (Soliman and Terstriep, 2019, CosmiQ, 2020) but the effects of the tiling strategies were not documented yet. In our study, different tiling strategies were tested on orthomosaics. The trained model on 1500 pixels by 1500 pixels tiles cut with a 25% overlap between tiles resulted in better evaluation metrics than the trained model on 500 pixels by 500 pixels tiles also sliced with overlap. This metrics improvement could be explained by the fact that smaller tiles do not allow to encompass patches of corals while larger tiles allowed the identification of shape characteristic (e.g. texture and arrangement of coral species) (Figure 9). This is crucial information for the detection and classification of objects. Thus, tiles size must be chosen as an input variable while processing data. It must be defined not only according to the available machine resources but also according to the size of the objects to be classified. These findings suggest that the tiling method for geospatial data probably impacts the performance of deep learning models. We recommend further studies to take into account annotated objects size on the rasters to ensure the integrity of the annotations after tiling the dataset.

In addition, we compared the effect of two tile-cutting strategies on the performance of the model (tiling according to a regular grid and an overlapping grid) (Table 2). In our use case, the use
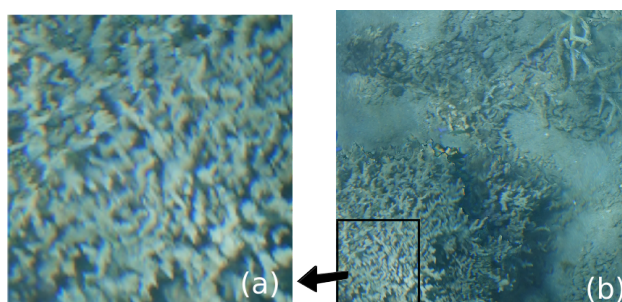
Figure 9. (a) A 500x500 pixel tile and (b) a 1500x1500 pixel tile showing that a single object is more fragmented when cut with small ones than with larger ones.

of a slicing strategy with 25% overlap between tiles had little impact on model metrics. To further explore this parameter it would be appropriate to test larger overlaps (e.g. 50% and 75%) to confirm the hypothesis that tiles overlap helps to create a beneficial data-augmentation process for object recognition (Yang et al., 2022).

4.4 Models performance

In our use case related to satellite images, we tried two pre-trained models during the transfer learning process. The R50 performed slightly better on our test dataset compared to the X101 and this is in line with the benchmark performed by `detectron2` (Wu et al., 2019b).

For the three use cases, model metrics are currently weak compared to similar computer vision tasks in the terrestrial domain and in other deep learning tasks (Rottensteiner et al., 2012, Stewart et al., 2022). This is mainly due to the fact that the training datasets are too small and, in the meanwhile, contain a large number of unbalanced classes.

The lack of training data is a limitation for models performance. However, we identified two ways to address the issue. Trained models can already predict unlabeled rasters. Admittedly, these predictions are not accurate, but they constitute a basis on which the annotator can rely to speed up the annotation process (e.g. a human-in-the-loop process) (Wu et al., 2022). Next, tasks related datasets (Ionescu et al., 2022) can help to strengthen datasets presented (Section 2.1) by providing more annotations.

A major limitation in the valuation of spatial data by the workflow is the lack of consideration of spatial information in the model training process. This information is all the more relevant when the objects to be predicted (e.g. boats or living species) present polymorphism linked to the geographical context (McLean and Stuart-Fox, 2014, Munday et al., 2003). Implementation of spatial information as a variable by deep learning models constitutes a perspective for improving this work (Yang and Tang, 2021, Janowicz et al., 2020).

5. CONCLUSION

We introduce a GeoAI tool, fully built with open source packages that enables to train and test deep learning models on different types of geospatial data. It automatically generates maps using deep learning models.The technical workflow which manages spatialized predictions has been implemented in

three use cases related to marine ecosystems and fishing monitoring. It has been validated and already provides results proving that AI-assisted mapping can value different types of marine images.

Optimizing the model scores is the next step in the development of this tool. This optimization will be done by increasing the number of samples available for training the models. Beyond the optimization of model scores, one of the major perspectives of this work is to improve and ease AI-assisted mapping, as well as to include spatial information as input variables into a multi-channel deep learning model to make the most from spatial imagery.

REFERENCES

Anthony, K., Bay, L. K., Costanza, R., Firn, J., Gunn, J., Harrison, P., Heyward, A., Lundgren, P., Mead, D., Moore, T., Mumby, P. J., van Oppen, M. J. H., Robertson, J., Runge, M. C., Suggett, D. J., Schaffelke, B., Wachenfeld, D., Walshe, T., 2017. New interventions are needed to save coral reefs. *Nature Ecology & Evolution*, 1(10), 1420–1422. https://doi.org/10.1038/s41559-017-0313-5.

Beilschmidt, C., Drönner, J., Mattig, M., Schweitzer, P., Seeger, B., 2023. Geo Engine: Workflow-backed Geo Data Portals. B. König-Ries, S. Scherzinger, W. Lehner, G. Vossen (eds), *BTW 2023*, Gesellschaft für Informatik e.V.

CosmiQ, 2020. solaris. github.com/CosmiQ/solaris/tree/0.5.0 (18 March 2023).

Courtial, A., Touya, G., Zhang, X., 2022. Representing Vector Geographic Information As a Tensor for Deep Learning Based Map Generalisation. *AGILE: GIScience Series*, 3, 1–8. https://agile-giss.copernicus.org/articles/3/32/2022/. Publisher: Copernicus GmbH.

CVAT.ai Corporation, 2022. Computer Vision Annotation Tool (CVAT). github.com/opencv/cvat (23 November 2022).

GeoAlert, L., 2021. Mapflow ai. mapflow.ai/ (07 April 2023).

Gillies, S., Perry, M., Erickson, T., 2015. rasterio. github.com/rasterio/rasterio (02 April 2023).

228

Hoegh-Guldberg, O., Poloczanska, E. S., Skirving, W., Dove, S., 2017. Coral Reef Ecosystems under Climate Change and Ocean Acidification. *Frontiers in Marine Science*, 4. https://www.frontiersin.org/articles/10.3389/fmars.2017.00158.

Hopkinson, B. M., King, A. C., Owen, D. P., Johnson-Roberson, M., Long, M. H., Bhandarkar, S. M., 2020. Automated classification of three-dimensional reconstructions of coral reefs using convolutional neural networks. *PLOS ONE*, 15(3), 1–20. https://doi.org/10.1371/journal.pone.0230671. Publisher: Public Library of Science.

Ionescu, B., Müller, H., Peteri, R., Rückert, J., Ben Abacha, A., García Seco de Herrera, A., Friedrich, C., Bloch, L., Singh, J., Idrissi-Yaghir, A., Schäfer, H., Kozlovski, S., Dicente Cid, Y., Popescu, A., Schindler, H., Chamberlain, J., Campbell, J., Clark, A., Brüngel, R., 2022. Overview of the ImageCLEF 2022: Multimedia Retrieval in Medical, Social Media and Nature Applications. *Lecture Notes in Computer Science*, 541–564. Journal Abbreviation: Lecture Notes in Computer Science.

Janowicz, K., Gao, S., McKenzie, G., Hu, Y., Bhaduri, B., 2020. GeoAI: spatially explicit artificial intelligence techniques for geographic knowledge discovery and beyond. *International Journal of Geographical Information Science*, 34(4), 625–636. https://doi.org/10.1080/13658816.2019.1684500. Publisher: Taylor & Francis.

Jordahl, K., Bossche, J. V. d., Fleischmann, M., McBride, J., Wasserman, J., Richards, M., Badaracco, A. G., Snow, A. D., Gerard, J., Tratner, J., Perry, M., Ward, B., Farmer, C., Hjelle, G. A., Taves, M., Hoeven, E. t., Cochran, M., rraymondgh, Gillies, S., Caria, G., Culbertson, L., Bartos, M., Eubank, N., Bell, R., sangarshanan, Flavin, J., Rey, S., maxalbert, Bilogur, A., Ren, C., 2022. geopandas/geopandas: v0.12.2. 10.5281/zenodo.7422493. 10.5281/zenodo.7422493.

Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A. C., Lo, W.-Y., Dollár, P., Girshick, R., 2023. Segment Anything. github.com/facebookresearch/segment-anything (18 April 2023).

Kiszka, J. J., Moazzam, M., Boussarie, G., Shahid, U., Khan, B., Nawaz, R., 2021. Setting the net lower: A potential low-cost mitigation method to reduce cetacean bycatch in drift gillnet fisheries. *Aquatic Conservation: Marine and Freshwater Ecosystems*, 31(11), 3111–3119. https://onlinelibrary.wiley.com/doi/abs/10.1002/aqc.3706. _eprint: onlinelibrary.wiley.com/doi/pdf/10.1002/aqc.3706.

Langenkämper, D., Zurowietz, M., Schoening, T., Nattkemper, T. W., 2017. BIIGLE 2.0 - Browsing and Annotating Large Marine Image Collections. *Frontiers in Marine Science*, 4. https://www.frontiersin.org/articles/10.3389/fmars.2017.00083.

Lin, T.-Y., Maire, M., Belongie, S., Bourdev, L., Girshick, R., Hays, J., Perona, P., Ramanan, D., Zitnick, C. L., Dollár, P., 2015. Microsoft COCO: Common Objects in Context. _eprint: 1405.0312.

Lyu, K., Zhang, X., Church, J. A., 2021. Projected ocean warming constrained by the ocean observational record. *Nature Climate Change*, 11(10), 834–839. https://doi.org/10.1038/s41558-021-01151-1.

McLean, C. A., Stuart-Fox, D., 2014. Geographic variation in animal colour polymorphisms and its role in speciation. *Biological Reviews*, 89(4), 860–873. onlinelibrary.wiley.com/doi/abs/10.1111/brv.12083. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/brv.12083.

Moore, B. E., Corso, J. J., 2020. FiftyOne. voxel51.com/fiftyone/ (16 April 2023).

Munday, P. L., Eyre, P. J., Jones, G. P., 2003. Ecological mechanisms for coexistence of colour polymorphism in a coral-reef fish: an experimental evaluation. *Oecologia*, 137(4), 519–526. https://doi.org/10.1007/s00442-003-1356-7.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S., 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. Curran Associates, Inc. pytorch.org (30 March 2023).

Pavoni, G., Corsini, M., Ponchio, F., Muntoni, A., Edwards, C., Pedersen, N., Sandin, S., Cignoni, P., 2022. TagLab: AI-assisted annotation for the fast and accurate semantic segmentation of coral reef orthoimages. *Journal of Field Robotics*, 39(3), 246–262. onlinelibrary.wiley.com/doi/abs/10.1002/rob.22049. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/rob.22049.

Pierce, J., Butler, M. J., Rzhanov, Y., Lowell, K., Dijkstra, J. A., 2021. Classifying 3-D Models of Coral Reefs Using Structure-From-Motion and Multi-View Semantic Segmentation. *Frontiers in Marine Science*, 8. https://www.frontiersin.org/articles/10.3389/fmars.2021.706674.

QGIS Development Team, 2009. QGIS Geographic Information System. Open Source Geospatial Foundation. qgis.osgeo.org (06 April 2023).

Rottensteiner, F., Sohn, G., Jung, J., Gerke, M., Baillard, C., Bénitez, S., Breitkopf, U., 2012. The ISPRS benchmark on urban object classification and 3D building reconstruction. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, I-3.

Sager, C., Janiesch, C., Zschech, P., 2021. A survey of image labelling for computer vision applications. *Journal of Business Analytics*, 4(2), 91–110. https://doi.org/10.1080/2573234X.2021.1908861. Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/2573234X.2021.1908861.

Soliman, A., Terstriep, J., 2019. Keras Spatial: Extending deep learning frameworks for preprocessing and on-the-fly augmentation of geospatial data. 69–76. 10.1145/3356471.3365240.

Stewart, A. J., Robinson, C., Corley, I. A., Ortiz, A., Ferres, J. M. L., Banerjee, A., 2022. TorchGeo: Deep Learning With Geospatial Data. _eprint: 2111.08872.

Talpaert Daudon, J., 2023a. artus. 10.5281/zenodo.7852855.

Talpaert Daudon, J., 2023b. artus-examples. 10.5281/zenodo.7852866.

Talpaert Daudon, J., 2023c. artus-qgis. 10.5281/zenodo.7852873.

Touya, G., Zhang, X., Lokhat, I., 2019. Is deep learning the new agent for map generalization? *International Journal of Cartography*, 5(2-3), 142–157. https://doi.org/10.1080/23729333.2019.1613071. Publisher: Taylor & Francis.

Urbina-Barreto, I., Elise, S., Guilhaumon, F., Bruggemann, J. H., Pinel, R., Kulbicki, M., Vigliola, L., Mou-Tham, G., Mahamadaly, V., Facon, M., Bureau, S., Peignon, C., Dutrieux, E., Garnier, R., Penin, L., Adjeroud, M., 2022. Underwater photogrammetry reveals new links between coral reefscape traits and fishes that ensure key functions. *Ecosphere*, 13(2), e3934. esa-journals.onlinelibrary.wiley.com/doi/abs/10.1002/ecs2.3934.

VoPham, T., Hart, J. E., Laden, F., Chiang, Y.-Y., 2018. Emerging trends in geospatial artificial intelligence (geoAI): potential applications for environmental epidemiology. *Environmental Health*, 17(1), 40. https://doi.org/10.1186/s12940-018-0386-x.

Weiss, K., Khoshgoftaar, T. M., Wang, D., 2016. A survey of transfer learning. *Journal of Big Data*, 3(1), 9. https://doi.org/10.1186/s40537-016-0043-6.

Westoby, R., Becken, S., Laria, A. P., 2020. Perspectives on the human dimensions of coral restoration. *Regional Environmental Change*, 20(4), 109. https://doi.org/10.1007/s10113-020-01694-7.

Wu, X., Xiao, L., Sun, Y., Zhang, J., Ma, T., He, L., 2022. A survey of human-in-the-loop for machine learning. *Future Generation Computer Systems*, 135, 364–381. https://doi.org/10.1016%2Fj.future.2022.05.014. Publisher : Elsevier BV.

Wu, Y., Kirillov, A., Massa, F., Lo, W.-Y., Girshick, R., 2019a. Detectron2. github.com/facebookresearch/detectron2 (15 April 2023).

Wu, Y., Kirillov, A., Massa, F., Lo, W.-Y., Girshick, R., 2019b. Detectron2 Model Zoo and Baselines. github.com/facebookresearch/detectron2/blob/main/MODEL_ZOO.md (15 April 2023).

Yang, N., Tang, H., 2021. Semantic Segmentation of Satellite Images: A Deep Learning Approach Integrated with Geospatial Hash Codes. *Remote Sensing*, 13(14). https://www.mdpi.com/2072-4292/13/14/2723.

Yang, S., Xiao, W., Zhang, M., Guo, S., Zhao, J., Shen, F., 2022. Image Data Augmentation for Deep Learning: A Survey. _eprint: 2204.08610.

Yuval, M., Alonso, I., Eyal, G., Tchernov, D., Loya, Y., Murillo, A. C., Treibitz, T., 2021. Repeatable Semantic Reef-Mapping through Photogrammetry and Label-Augmentation. *Remote Sensing*, 13(4). https://www.mdpi.com/2072-4292/13/4/659.