



HAL
open science

Using the IPOL Journal for Online Reproducible Research in Remote Sensing

Miguel Colom, Tristan Dagobert, Carlo de Franchis, Rafael Grompone Von Gioi, Charles Hessel, Jean-Michel Morel

► **To cite this version:**

Miguel Colom, Tristan Dagobert, Carlo de Franchis, Rafael Grompone Von Gioi, Charles Hessel, et al.. Using the IPOL Journal for Online Reproducible Research in Remote Sensing. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 2020, 13, pp.6384-6390. 10.1109/JSTARS.2020.3032100 . hal-03135023

HAL Id: hal-03135023

<https://hal.science/hal-03135023>

Submitted on 5 Mar 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Using the IPOL Journal for Online Reproducible Research in Remote Sensing

Miguel Colom , Tristan Dagobert, Carlo de Franchis,
Rafael Grompone von Gioi, Charles Hessel, and Jean-Michel Morel

Abstract—Reproducible research is needed to ensure that scientific results in the literature are reliable, unbiased, and verifiable by others. The journal *Image Processing On Line (IPOL)* publishes reproducible articles since 2010. This means publishing an algorithm by a literary description, a pseudocode, its source code, a series of test examples, an online facility allowing to test the code on this data and other data submitted by the user, and finally an experimental archive. In this work, we discuss how to publish and review reproducible research in the specific discipline of remote sensing. We put a special emphasis on the construction and proper documentation of public datasets. We show case studies of remote sensing articles publicly available in IPOL, which demonstrate the feasibility of reproducible research in this area. The methods and their application are explained, along with details on how the datasets were built and made available for evaluation, comparison, and scoring to eventually help establish a reliable state-of-the-art of the discipline. Finally, we give specific recommendations for authors and editors willing to publish reproducible research in remote sensing.

Index Terms—Algorithm comparison, datasets, remote sensing, reproducible research, image processing on line (IPOL).

I. INTRODUCTION

THE *credibility crisis* in scientific research was warned by Donoho [1]. Paraphrasing Claerbout, they defined [2] what a reproducible article is as follows.

An article about computational science in a scientific publication is not the scholarship itself, it is merely advertising of the scholarship. The actual scholarship is the complete software development environment and the complete set of instructions, which generated the figures.

In [3], to analyze the extent of the reproducibility crisis in remote sensing, the authors conducted a survey on the state of reproducibility in SAR imaging. To the question *Have*

you ever encountered difficulties in reproducing results from other researchers?, some 53% answer “yes, sometimes”, 21% “yes, often”, and more surprisingly 17% answer “No, I never tried”.

Many initiatives such as journals devoted to fully-reproducible research, platforms, and services have started trying to address this problem. *Image Processing On Line (IPOL)*¹ is a journal, which publishes reproducible research on general image processing and lately expanded to other more specific applications, including remote sensing. Founded with the support of an advanced grant of the European Research Council to develop the mathematical theory and algorithms in the image processing field, its main goal was to demonstrate the ability to mount complex image processing chains for image blind restoration, automatic 3-D reconstruction from several photographs of a scene, and automatic analysis and detection of features in images. The journal also encourages the more informal exchange of executable algorithms by hosting *workshops*: temporary submissions of online algorithms, which do not need to follow the publication rules, but allow researchers and partners to directly experiment and eventually evolve into a final paper. IPOL follows the definition of *reproducible research* given by Buckheit *et al.* [4], where the output of the research work is not just a text but also the software and data. In IPOL, this is considered as a whole and text + source code + data is published under the same digital object identifier (DOI). Since the *credibility crisis* in research was acknowledged, several initiatives have flourished [5], as for example the OpenAIRE² program operated by CERN, and Zenodo [6] (a general-purpose and open-access repository under openAIRE), as well as infrastructure and services that can be used for reproducible research such as the Galaxy platform [7] in the field of genomic research, or IEEE’s Code Ocean service,³ among others. Frery *et al.* [8] proposed a *badging* system to identify works that comply with minimum requirements of reproducibility and replicability in remote sensing research. The remote sensing reproducible research (RSRR) badge would be based on about 20 criteria concerning web page, code, and data. It ends up requiring actions from the authors that are similar to those requested in IPOL.

In this work, based on the experience of the last ten years developing the IPOL journal, we give recommendations to

Manuscript received September 29, 2020; revised October 13, 2020; accepted October 14, 2020. Date of publication October 19, 2020; date of current version November 5, 2020. This work was supported in part by the Office of Naval research under Grant N00014-17-1-2552 and Grant N00014-20-S-B001, in part by the DGA Astrid Project filmer la Terre ANR-17-ASTR-0013-01, in part by the MENRT, in part by the Fondation Mathématique Jacques Hadamard, and in part by Kayrros SAS. (Corresponding author: Miguel Colom.)

The authors are with the Université Paris-Saclay, Cole Normale suprieure Paris-Saclay, Centre National de la Recherche Scientifique, F-91190 Gif-sur-Yvette, France (e-mail: miguel.colom-barco@ens-paris-saclay.fr; tristan.dagobert@dga.defense.gouv.fr; carlo.de-franchis@ens-paris-saclay.fr; rafael.grompone@ens-paris-saclay.fr; charles.hessel@ens-paris-saclay.fr; jean-michel.morel@ens-paris-saclay.fr).

Digital Object Identifier 10.1109/JSTARS.2020.3032100

¹[Online]. Available: <https://www.ipol.im/>

²[Online]. Available: <https://www.openaire.eu/>

³[Online]. Available: <https://codeocean.com/>

authors and editors willing to publish reproducible research in remote sensing, and show real case studies demonstrating that reproducible research in remote sensing is certainly feasible (and absolutely necessary).

The plan of the article follows. Section II discusses how to write and review reproducible articles in remote sensing, and gives some details on how this is conducted in IPOL. Section III addresses the problem of public datasets, their availability, their quality, and how to reuse them to establish a reliable state-of-the-art by proper evaluation and scoring of methods. Section IV shows five case studies of reproducible remote sensing published in IPOL, giving details on the application of the methods and on how the datasets were built. Section V contains recommendations for editors and authors willing to publish reproducible research in general and in remote sensing specifically, based on the experience of IPOL's editors. Section VI concludes this article.

II. HOW TO WRITE AND REVIEW A REPRODUCIBLE RESEARCH IN REMOTE SENSING

Reviewing a reproducible research article in remote sensing requires additional steps with respect to a classic publication in order to ensure the reproducibility of the results, repeatability of the experiments, availability of the data and, in general, the overall quality of the published works.

A fundamental step, often forgotten in nonreproducible publications, is to check that the source code and the algorithmic descriptions (pseudocodes) in the article match accurately. Ideally, one should be able to reimplement the method exactly from the descriptions in the article. However, both the source code and the pseudocodes in the article need to be provided, because the source code shows precisely, *which operations* are performed, while the pseudocodes are much shorter and are the code's specification, that also allows rewriting in another language. They also help the reader focus on the *significant* parts of the code. The article itself puts everything in its scientific context.

Establishing a true state-of-the-art in remote sensing methods requires reliable datasets. By *reliable*, we mean that the process that built them is detailed and communicated to the community (say, as a publication in a journal). The dataset should not be biased and might contain real data as well as synthetic. Examples of biased datasets would be cloud-free Sentinel-2 images or SMOS data without radio-frequency interference (RFI) [9]; indeed, about 60% of SMOS data are affected by RFIs and a similar proportion of Sentinel-2 images are cloudy [10]. The availability of proper datasets allows for a fair comparison and ranking of algorithms. Thus, the editor and reviewers of a reproducible article in remote sensing should request the use of public and well-documented datasets. Section III discusses in detail this topic.

Finally, the editors should encourage that the authors to show not only positive results, but also cases of failure to better understand the limitations of the method. Here, IPOL's setup comes to great advantage, as it allows referees to test the paper's code on their own data.

These are minimal recommendations for reviewing reproducible articles. In Section II-A, we describe how the review process is conducted in the IPOL journal.

A. IPOL Approach

IPOL was born as a fully reproducible research journal [11], [12]. After more than ten years of experience, we have implemented standard procedures to ensure reproducible publications beyond the minimal requirements mentioned before [13].

Each IPOL article needs to follow the principles of reproducibility (the possibility that other researchers can obtain the same results given that source code and data is public and comes along a detailed scientific article) and repeatability (the fact that the experiments can be redone as many times as needed, with publicly available evaluation datasets). Indeed, all source code in IPOL is open-source, data are publicly available to the scientific community (with details on how datasets were built), and the text (the article) is under public documentation license. Repeatability is ensured by providing public demonstrations that allow users to freely test the algorithms with the proposed datasets or their own data. A public archive of experiments is also available. The access is open-source and there is no author or user fee. The IPOL demo editors take care of building the demos of the authors, who are only requested to give their source code, the compilation instructions, examples of data they would like their demo to offer, and a description of the demo.

IPOL has style guidelines for authors about the writing of the article⁴ as well as for the software.⁵ At first the software guidelines were a list of highly-technical mandatory requirements, but soon it became clear that only experts in software engineering were able to follow them; we realized that they were preventing authors to submit their work. Thus, we adopted the practice of establishing a tight contact between editors and authors to reach the required source code quality. This approach has proven successful and indicated that enforcing the adoption of high-quality coding standards is not automatic and is in fact an *educational* process.

A major problem when reviewing research in computational sciences is the requirement of finding reviewers and editors who are both competent in analyzing source code and are experts in the corresponding scientific field. The lack of these multidisciplinary profiles makes hard the in-depth review of reproducible articles. In IPOL, this problem has been mitigated by choosing at least two reviewers, one of them an expert in software engineering and the other an expert in the scientific field. Note that we require that both reviewers (as well as the editor) are skilled in both. The most important task of the code reviewer is certainly to check that the source code matches accurately the algorithmic descriptions given in the article. With this review system, the typical review time from submission to the edition of an online preprint is two weeks. The time from preprint to publication has high variance, as many authors often work on improvements. It is on average four months.

Once the article, code, and data have been published, all of them become available as a whole under the same DOI. If any

⁴[Online]. Available: https://tools.ipol.im/wiki/ref/author_manual/

⁵[Online]. Available: https://tools.ipol.im/wiki/ref/software_guidelines/

bugs are detected after publication,⁶ the authors are requested to submit a revised version of their code, which will be checked by the IPOL's Editorial Board (in case of a bug correction, there is no need to start another peer-review evaluation), and added to the history, which is available on the article's page.⁷ Note that even if corrections are performed after publication, the original sources are always kept unaltered and under the same DOI. Here, we refer to bugs or technical changes, but keeping the same algorithm. In case the method changes, also would the pseudocode descriptions and the implementation, and thus it would require indeed a new submission given that it would correspond to a different algorithm. Hence, if the improvements are significant and the changes major, authors are encouraged to submit them as a different publication.

Each IPOL article comes along with an online demo allowing to quickly test the algorithms on sample data or user provided data. Also, each demo keeps an archive of experiments performed with uploaded data. This has proven very useful to have a large corpus of use cases and to uncover the interests of the community. Indeed, demos with large archives mean a large interest (some articles have archives of up to 59 000 distinct experiments).

IPOL encourages authors to restrict as much as possible the code dependencies on external libraries. On the one hand, this ensures that the main code is self-contained and that its functionality is not black-boxed inside external components. On the other hand, it helps long-term durability of the code. There is no guarantee that a published code can be compiled in the long term, given that it depends on external libraries that can change the signature of the functions, as well as the compilation rules. In the case of Python code, the IPOL demo system automatically takes care of creating virtual environments [14] and it is mandatory that authors declare in a `requirements.txt` file the list of dependencies and their exact versions. This has proven useful for long-term maintenance. Other approaches such as lightweight virtualization are also useful [15].

III. COMMON DATASETS FOR EVALUATION AND COMPARISON

Establishing the true state-of-the-art in remote sensing research requires a fair comparison of algorithms. This implies reliable datasets for evaluation purposes, and raises the question of what *reliable* datasets are, and how to build them. Depending on the application, the dataset could include *ground-truth* data, i.e., a reference to compare with. For example, let us consider a denoising algorithm for satellite images. If the noise model is known (e.g., Poisson noise with known parameter), one could evaluate the denoising algorithms by adding synthetic noise to a set of noise-free ground-truth images used as reference. How to build these noise-free images? They could be obtained by averaging several snapshots of the same location and conditions (i.e., including same incidence angle, season, time, atmospheric conditions, calibration of the sensor, no changes on the ground).

⁶The IPOL demo system performs weekly compilation tests of all published codes. Also, it emails the Editorial Board in case of an abnormal termination of any of the published codes.

⁷See for example <https://www.ipol.im/pub/art/2019/277/>

It is important that the datasets are publicly available and that the details on how they were built are published. This helps in assessing any possible limitation in the data or in the ground-truth annotation. In the case of deep-learning applications, the size of the dataset is also a requirement. We can cite two actual examples of publication of datasets in IPOL, such as high precision 3-D datasets [16], and human physiological data [17].

In the case of IPOL, we have standardized the concept of *templates* in the demo system. They are in fact datasets of objects than can be used by different demos. So far there are 23 different datasets that can be added to any IPOL demo, including `denoising` (noise-free images), `microtextures`, `standardTestImages` (standard test images used in the image processing literature), `JPEGquality` (images under different JPEG quality factors, for compression analysis), `forgery` (image forgery detection), `deblurring`, `statokinesigrams` (physiological data), `kodakImageSuite` (standard dataset), and `CloudDetection_Sentinel2_L1C_RGB` and `CloudDetection_TimeSeries_Sentinel2_L2A` (Sentinel-2 datasets for cloud detection).

This enables different methods to be evaluated with the same data, and brings up the possibility to create accessory workshops that compare several algorithms with the same standard input data for benchmarking.

IV. CASE STUDIES: REPRODUCIBLE METHODS IN REMOTE SENSING PUBLISHED ON IPOL

This section presents case studies of actual IPOL demos on remote sensing, discussing the actual method and application, as well as explaining how the corresponding datasets were built.

A. Temporal Repetition Detector for Time Series of Spectrally Limited Satellite Imagers

The article [18] along with its demo⁸ addresses the problem of detecting visually stable areas in time series of satellite images. It can be used to detect cloud-free areas. It focuses on sensors with few spectral bands and high revisit frequency and exploits the redundancy of information acquired during these revisits. It detects stable areas with an unsupervised multitemporal ground matching based algorithm able to learn the “normal” aspect of the ground from long enough time series of registered images. Experiments have been carried out on 18 PlanetScope image times series of various locations. These time series come with hand-made ground truth labels that are published with the article. The demo allows to run the algorithm and to change its parameters. Users can upload their own input data or use the `Sat_Spectrally_Limited_TS` dataset available in IPOL for this and other demos. Fig. 1 shows this demo.

B. Cloud Detection by Disparity Phase Analysis for Pushbroom Satellite Imagers

The article [19] and its demo⁹ address the problem of detecting clouds in a single satellite image. It focuses on satellites

⁸[Online]. Available: <https://www.ipol.im/pub/art/2020/245/>

⁹[Online]. Available: <https://www.ipol.im/pub/pre/271/>



Fig. 1. Demo of the method temporal repetition detector for Time Series of Spectrally Limited Satellite Imagers showing the parameter controls and part of the standard dataset `Sat_Spectrally_Limited_TS`.

equipped with pushbroom sensors. The method exploits the parallax effect caused by pushbroom sensors, which induces an observable shift between color bands. The magnitude of the shift is different between cloudy and noncloudy areas. The algorithm has several steps such as rank transform of the input image, interbands optical flow estimation, and feature-based discrimination of cloudy and cloud free areas. Experiments have been carried out on 9854 patches of size 366×366 pixels extracted from the Sentinel-2 dataset of Hollstein *et al.*, [10]. The online demo allows to evaluate the algorithm on a subset of these patches, and to vary the parameters of the method. In order to exhibit the influence of the different parameters in the algorithm pipeline, both intermediate and final detection maps are displayed after the computation.

The complete dataset of 9854 Sentinel-2 image patches with their ground truth are available for download on the article page. The demo uses a subset of 9 images as the `Clouds_Pushbroom`, which can be reused by other IPOL demos and workshops.

C. Workshop on Relative Radiometric Normalization of Multisatellite Image Time Series

This workshop¹⁰ demonstrates the relative radiometric normalization of long image time series, described in [20]. This algorithm allows to merge time series from different satellites, thus allowing to increase the revisit time under comparable conditions. This method is illustrated by the fusion of time series of Landsat-8 and Sentinel-2 images. Hence, the workshop includes several time series made of Sentinel-2 images at processing levels L1C and L2A, as well as Landsat-8 images. The nature of this problem does not allow to have a ground truth. Hence, the comparison with other methods is only possible through the use of a fixed dataset. We provide in the workshop three satellite image time series.

¹⁰[Online]. Available: <https://ipolcore.ipol.im/demo/clientApp/demo.html?id=77777000114>

Sentinel-2 blue, green, and red bands have a pixel size of 10 m, while Landsat-8 panchromatic band is sampled at 15 m and its blue, green, and red bands are sampled at 30 m. To create a time series with images from both satellites, the pixel sizes must be made uniform. We, thus, use the panchromatic band to pan-sharpen Landsat-8 color bands to 15 m/pixel and then upsample them to 10 m/pixel by interpolation. GDAL¹¹ pansharpening tool is used for the first step. The time series is then registered with subpixel accuracy using the phase correlation method [21], as described in the following Section IV-D. The resampling of the images for both the change of pixel size and registration are performed by spline interpolation [22]. The dataset name in IPOL is `RelativeRadiometricNormalization`.

D. Workshop: Image Time Series Registration

The workshop¹² on time series registration demonstrates the alignment of long satellite image time series. This demo takes as input satellite image time series (optical or radar), and outputs the registered time series, where all interimages motion has been removed. This is a prerequisite for practically all time series analysis algorithms.

This method has three main parts. The first one is the pairwise estimation of translations, using phase correlation [21]. Shifts are computed for all possible pairs of images. The second one is a verification of the coherence of these shifts, and the computation of a single, final shift for each input image. Indeed, this tool is used to register time series that may contain images that cannot be registered: this is generally the case if they are fully covered by clouds. The misaligned images are removed from the time series. The verification is made using an alignment graph; nodes that are not connected are discarded. The third and final step is interpolation, performed using b-splines [22].

The dataset in this workshop (`ImageTimeSeriesRegistration`), thus, contains several misaligned time series from Sentinel-2 and Landsat-8. Their misalignment is due to native inaccuracies in the satellite's position and attitude used for geo-referencing the images.

E. Workshop: Comparison of Cloud Detectors

This workshop¹³ compares four cloud detectors, two of which are described in Sections IV-A and IV-B. The two others are articles in preparation. For each of them, several versions are available.

The proposed algorithms are of two types: disparity-based and time-series-based. To compare their results, we need inputs that satisfy the requirements of both types of methods, namely image time series (for repetition-based detectors), whose images have several bands acquired with time delays (for the disparity-based detectors). The proposed inputs therefore are time series of RGB images. We provide images from several satellites: Planet, Sentinel-2 (correction level L1C and L2A), and Landsat-8. These datasets in IPOL are

¹¹[Online]. Available: <https://gdal.org/index.html>

¹²[Online]. Available: <https://ipolcore.ipol.im/demo/clientApp/demo.html?id=77777000117>

¹³[Online]. Available: <https://ipolcore.ipol.im/demo/clientApp/demo.html?id=77777000116>

CloudDetection_Sentinel2_L1C_RGB, Cloud-Detection_TimeSeries_Sentinel2_L2A, Cloud-Detection_Registered_RGB_TS_Landsat8, and CloudDetection_Registered_RGB_TS_Planet.

We created one dataset by satellite and correction level. All provided image time series are registered by phase correlation, as described in Section IV-D.

The demos presented here use common datasets with images or groups of images whose size allows to execute the algorithms in a reasonable time (typically, 30 s). The IPOL demo system allows demos to use interactive controls over the images,¹⁴ thus allowing to choose particular points or regions over a reduced view of a large satellite image, if needed.

V. GOOD PRACTICES FOR REPRODUCIBLE RESEARCH IN REMOTE SENSING

The experience of IPOL publishing computational reproducible research (first focused on classic image processing, and then expanded to other fields and data types) enabled our editors to identify good practices and make recommendations.

We shall list the most important here.

- 1) *Reusable and publicly available datasets*: As discussed in Section. III, the use of datasets that can be reused by different demos allows to compare methods and helps establishing the true state-of-the-art.
- 2) *Reliable datasets*: Making available the datasets is not enough. A detailed description of how a dataset was built is needed. Ideally, an actual publication should come along with the data as well as any code that was used to build the dataset.
- 3) *Accurate metadata*: Every image provided in a dataset should include all the metadata needed to retrieve the original files (distributed by a satellite operator) from which the image comes from. For example, every Sentinel-2 image should be provided with the identifier of the corresponding product distributed in SAFE format by Copernicus Open Access Hub and band number. Similarly, PlanetScope images should be provided with their `item_id` (product identifier) and `asset_type` (processing level) as listed in Planet Explorer and Planet API catalogs. If an image provided in a dataset is only a subset of the original image, which is common in remote sensing research as remote sensing images are often very large, the exact coordinates of the subset should be included in the metadata, in order to allow reproducing the dataset image from the original image.
- 4) *Standard file formats*: Standard image file formats commonly used in remote sensing, such as GeoTIFF, should be preferred when possible. GeoTIFF allows to store remote sensing metadata, and in particular georeferencing information, within the image file. In the case of Sentinel-2 L1C, for example, using GeoTIFF ensures that the image geographic bounds are stored with the image, hence, the subset coordinates mentioned in the previous point are available.

- 5) *Rich datasets*: Remote sensing image datasets should contain images acquired on a variety of locations, at different dates spanning all seasons and atmospheric conditions. Datasets built for machine learning applications are usually made of small image crops extracted from a few satellite acquisitions. Particular attention should be made when preparing such datasets to ensure that they contain image crops coming from different satellite acquisitions.
- 6) *Avoid third party dependencies*: The published codes are likely to break if the signatures of functions in third party libraries change. Thus, authors should limit external dependencies and avoid that important parts of their methods are hidden as black boxes inside external libraries. A README file included in the published code should specify, which libraries and versions are required.
- 7) *Use virtual environments in Python*: In the case of Python, virtual environments are a solution to the versioning problem mentioned in the previous point. They allow to declare, which libraries are needed to run the program and set their exact versions. In IPOL, this is a mandatory requirement.
- 8) *Machine-independent sources*: Authors should try their best not to write code that only runs in a particular architecture, or hardcoding capabilities instead of letting the compiler determine if they are present in the underlying architecture. Typical pitfalls: assuming that all architectures have AVX instructions or any vendor-specific capabilities.¹⁵ In case, this is unavoidable (for example, the program runs GPU code with specific libraries), this needs to be clearly explained in the README.
- 9) *Archival and easy referencing*: Authors are encouraged to submit their source code to an archival platform. In the case of IPOL, the code is transparently submitted to Software Heritage.¹⁶
- 10) *Online demo*: This helps readers to get a quick idea of the method and use it without technical knowledge. The main benefit for the authors is receiving citations of their work given that the experimentation is easier and the method convincing.
- 11) *Show failures*: The article should not only show successful experiments, but also failure cases and bad results. This helps to understand the limitations of the proposed method and to establish to which particular application it is better suited.

VI. CONCLUSION

IPOL started in 2009 as a fully reproducible research journal, with a special emphasis on detailed mathematical descriptions of the published methods. Each publication comes with the text of the article (as in a classic journal), but including as well under the same DOI the source code and the data of the methods, as a whole. Endowed with on online demos and growing large experimental archives, it has helped to establish

¹⁴See for example <https://ipolcore.ipol.im/demo/clientApp/demo.html?id=286>

¹⁵See for example <https://github.com/tensorflow/tensorflow/issues/29788>

¹⁶<https://www.softwareheritage.org/2020/06/11/ipol-and-swh/?lang=es>

the true state-of-the-art in image processing on several basic problems such as image comparison or edge detection. IPOL has naturally expanded to remote sensing. In Section IV, we gave four emblematic examples of articles or workshops on this topic: they address the first basic problem faced by any researcher in the analysis of time series of satellite images: registration, cloud detection, and radiometry equalization. Without these three bases, no analysis and detection algorithm is possible.

The IPOL demo system (a distributed system of microservices) has made it possible to run hundreds of concurrent demos along several load-balanced servers. Moreover, a large engineering effort has been made to allow editors create and modify online demos easily, without requiring much technical knowledge. The fact that a minimal effort is required to create new demos and a common tool for the editors (the control panel) have been proven a good solution for the initial bottleneck faced when creating new demos [23].

Following the guidelines and principles described in this article, we expect the publication of several papers currently in preparation for IPOL, on the following topics covering the three main types of earth observation satellites: anomaly detection in hyperspectral imagery (Sentinel 5), methane lume detection, phase unwrapping of SAR Sentinel1 images, traffic monitoring in Planetscope images, change detection, and classification in times series of ellipse (SMOS-HR).

REFERENCES

- [1] D. L. Donoho, "An invitation to reproducible computational research," *Biostatistics*, vol. 11, no. 3, pp. 385–388, Jul. 2010.
- [2] J. B. Buckheit and D. L. Donoho, *Wavelab and Reproducible Research*. Berlin, Germany: Springer, 1995.
- [3] T. Balz and F. Rocca, "Reproducibility and replicability in SAR remote sensing," *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 3834–3843, Jun. 30, 2020, doi: [10.1109/JSTARS.2020.3005912](https://doi.org/10.1109/JSTARS.2020.3005912).
- [4] S. Fomel and J. F. Claerbout, "Guest editors' introduction: Reproducible research," *Comput. Sci. Eng.*, vol. 11, no. 1, pp. 5–7, Jan. 2009. [Online]. Available: <http://ieeexplore.ieee.org/document/4720217>
- [5] M. Colom, B. Kerautret, and A. Krhenbhl, "An overview of platforms for reproducible research and augmented publications," in *Reproducible Research in Pattern Recognition*, B. Kerautret, M. Colom, D. Lopresti, P. Monasse, and H. Talbot, Eds. Berlin, Germany: Springer, 2019, pp. 25–39.
- [6] P. Manghi, N. Manola, W. Horstmann, and D. Peters, "An infrastructure for managing EC funded research output-The OpenAIRE project," *Grey J., Int. J. Grey Literature*, vol. 6, no. 1, pp. 31–39, 2010.
- [7] B. Giardine *et al.*, "Galaxy: A platform for interactive large-scale genome analysis," *Genome Res.*, vol. 15, no. 10, pp. 1451–1455, Oct. 2005.
- [8] A. C. Frery, L. Gomez, and A. C. Medeiros, "A badging system for reproducibility and replicability in remote sensing research," *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 4988–4995, Aug. 25, 2020, doi: [10.1109/JSTARS.2020.3019418](https://doi.org/10.1109/JSTARS.2020.3019418).
- [9] N. Skou, S. Misra, J. Balling, S. Kristensen, and S. Sobjaerg, "L-Band RFI as experienced during airborne campaigns in preparation for SMOS," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 3, pp. 1398–1407, Mar. 2010.
- [10] A. Hollstein, K. Segl, L. Guanter, M. Brell, and M. Enesco, "Ready-to-Use methods for the detection of clouds, cirrus, snow, shadow, water and clear sky pixels in Sentinel-2 MSI Images," *Remote Sens.*, vol. 8, no. 8, p. 666, Aug. 2016. [Online]. Available: <http://www.mdpi.com/2072-4292/8/8/666>
- [11] M. Colom, B. Kerautret, N. Limare, P. Monasse, and J.-M. Morel, "IPOL: A new journal for fully reproducible research; analysis of four years development," in *Proc. IEEE 7th Int. New Technolo., Mobility*, 2015, pp. 1–5.
- [12] N. Limare and J.-M. Morel, "The IPOL initiative: Publishing and testing algorithms on line for reproducible research in image processing," *Procedia Comput. Sci.*, vol. 4, pp. 716–725, 2011.
- [13] M. Arvalo, C. Escobar, P. Monasse, N. Monzn, and M. Colom, "The IPOL demo system: A scalable architecture of microservices for reproducible research," in *Reproducible Research in Pattern Recognition* (series Lecture Notes in Computer Science), B. Kerautret, M. Colom, and P. Monasse, Eds., vol. 10214. Cham, Switzerland: Springer, 2017, pp. 3–16. [Online]. Available: http://link.springer.com/10.1007/978-3-319-56414-2_1
- [14] M. Valiev, B. Vasilescu, and J. Herbsleb, "Ecosystem-level determinants of sustained activity in open-source projects: A case study of the PyPI ecosystem," in *Proc. 26th ACM Joint Softw. Eng.*, 2018, pp. 644–655. [Online]. Available: <http://dl.acm.org/citation.cfm?doi=3236024.3236062>
- [15] C. Boettiger, "An introduction to Docker for reproducible research," *ACM SIGOPS Operating Syst. Rev.*, vol. 49, no. 1, pp. 71–79, Jan. 2015. [Online]. Available: <https://dl.acm.org/doi/10.1145/2723872.2723882>
- [16] J. Digne, N. Audfray, C. Lartigue, C. Mehdi Souzani, and J.-M. Morel, "Farman Institute 3D point sets - high precision 3D data sets," *Image Process.*, vol. 1, pp. 281–291, Sep. 2011. [Online]. Available: http://www.ipol.im/pub/art/2011/dalmm_ps/?utm_source=doi
- [17] C. Truong *et al.*, "A data set for the study of human locomotion with inertial measurements units," *Image Process.*, vol. 9, pp. 381–390, Nov. 2019. [Online]. Available: http://www.ipol.im/pub/art/2019/265/?utm_source=doi
- [18] T. Dagobert, R. Grompone von Gioi, J.-M. Morel, and C. de Franchis, "Temporal repetition detector for time series of spectrally limited satellite imagers," *Image Process.*, vol. 10, pp. 62–77, 2020.
- [19] T. Dagobert, R. Grompone von Gioi, C. de Franchis, and J.-M. Morel, "Cloud detection by disparity phase analysis for pushbroom satellite imagers," *Image Process., Preprint*, 2020.
- [20] C. Hessel, R. Grompone von Gioi, J. M. Morel, G. Facciolo, P. Arias, and C. de Franchis, "Relative radiometric normalization using several automatically chosen reference images for multi-sensor, multi-temporal series," *ISPRS Ann. Photogrammetry, Remote Sens. Spatial Inf. Sci.*, vol. V-2-2020, pp. 845–852, 2020. [Online]. Available: <https://www.isprs-ann-photogramm-remote-sens-spatial-inf-sci.net/V-2-2020/845/2020/>
- [21] H. Foroosh, J. B. Zerubia, and M. Berthod, "Extension of phase correlation to subpixel registration," *IEEE Trans. Image Process.*, vol. 11, no. 3, pp. 188–200, Mar. 2002.
- [22] T. Briand and A. Davy, "Optimization of image B-spline interpolation for gpu architectures," *Image Process.*, vol. 9, pp. 183–204, 2019.
- [23] M. Colom, "Extending IPOL to new data types and machine-learning applications," in *Reproducible Research in Pattern Recognition* (series Lecture Notes in Computer Science), B. Kerautret, M. Colom, D. Lopresti, P. Monasse, and H. Talbot, Eds., vol. 11455, Cham, Switzerland: Springer, 2019, pp. 3–24. [Online]. Available: http://link.springer.com/10.1007/978-3-030-23987-9_1

Miguel Colom received the Ph.D. degree in applied mathematics from Universitat de les Illes Balears, Palma, Spain, in 2014 and the HDR degree in sciences and technologies of information and communication from ENS Paris-Saclay, Cachan, France, in 2019.

He is a Senior Researcher with Centre Borelli, ENS Paris-Saclay. He is one of the IPOL founding editors, and the designer and responsible of the current IPOL demo system, where he is the Director of the IPOL's engineering team. He works on satellite design (specifically, in the SMOS-HR mission) in collaboration with CESBIO (CNES), as well as on detection of falsifications in images and video within the DEFALS project, funded by ARN. He is cosupervising three Ph.D. candidates on these subjects. His main research interests include reproducible research, as well as image and general signal processing, and artificial intelligence.

Tristan Dagobert received the Ph.D. degree in applied mathematics from the Centre de Mathématiques et de Leurs Applications, cole Normale Supérieure Paris-Saclay, Cachan, France, in 2017.

He is currently an Associate Researcher with Centre Borelli. His research interests include satellite image processing and artificial intelligence.

Carlo de Franchis received the Ph.D. degree in applied mathematics from ENS Paris-Saclay, Cachan, France, in 2015.

He is a Researcher with Centre Borelli and a Technical Lead with Kayros. He worked on multi-view stereo 3-D mapping with satellite images with ENS Paris-Saclay. His research interests include remote sensing and computer vision.

Rafael Grompone von Gioi received the Ph.D. degree in applied mathematics from the École Normale Supérieure de Cachan, Gif-sur-Yvette, France, in 2010.

He is currently a Researcher with Centre Borelli, Paris-Saclay, France. His research focuses on computer vision and in particular on statistical methods for pattern detection.

Charles Hessel received the B.Sc. and M.Sc. degrees in electronics, electrotechnics, and automation from University Toulouse III Paul Sabatier, Toulouse, France, in 2012 and 2014, respectively, and the Ph.D. degree in applied mathematics from École Normale Supérieure Paris-Saclay, Cachan, France, in 2018.

He was a Ph.D. student at DxO, Boulogne-Billancourt, France, and Centre Borelli, snf École Normale Supérieure Paris-Saclay. He is currently a Postdoctoral Researcher with École Normale Supérieure Paris-Saclay and Kayrros.

Dr. Hessel received a CIFRE Scholarship from the French Ministry of Higher Education, Research and Innovation.

Jean-Michel Morel received the Ph.D. degree in applied mathematics from University Pierre et Marie Curie, Paris, France, in 1980.

He has been a Professor of Applied Mathematics with the cole Normale Supérieure Paris-Saclay, Cachan, France, since 1997. His research interests include the mathematical analysis of image processing and on detection theory in very large image series.

Prof. Morel is a cofounder of *Image Processing on Line* (www.ipol.im), cofounded in 2011, the first journal publishing reproducible algorithms, software, and online executable articles. He is a Laureate of 2013 Grand Prix INRIA Acadmie des Sciences, 2015 CNRS mdaille de innovation, 2015 IEEE Longuet-Higgins prize, and is 2017 Doctor honoris causa of Universidad de la Repblica, Montevideo.