IPOL: a new journal for fully reproducible research; analysis of four years development

Miguel Colom, Bertrand Kerautret, Nicolas Limare, Pascal Monasse, Jean-Michel Morel

IPOL: a new journal for fully reproducible research; analysis of four years development

Miguel Colom *, Bertrand Kerautret †‡, Nicolas Limare *, Pascal Monasse §, and Jean-Michel Morel *
* CMLA, ENS-Cachan, 61, Av. du Président Wilson, 94235 Cachan, France,
Email: {colom, limare, morel}@cmla.ens-cachan.fr
† Université de Lorraine, LORIA, UMR 7503, Vandoeuvre-lés-Nancy, F-54506, France
‡ CNRS, LORIA, UMR 7503, Vandoeuvre-lés-Nancy, F-54506, France
Email: bertrand.kerautret@loria.fr
§ Université Paris-Est, LIGM (UMR 8049), ENPC, F-77455 Marne-la-Vallée, France
Email: monasse@imagine.enpc.fr

***Abstract*—After four years of development of the Image Processing On Line journal (IPOL), this article presents a first analysis and overview of its scientific and technical development. The main issues met and overcome from the beginning of the journal are described with a focus on the purpose of the journal to establish a state of the art on the main Image Processing topics. The evolution of the online demonstration is also presented with a first analysis of author/publisher criticism, which led to a proposal for a new modular architecture of its demo system.**

Keywords: reproducible research; image processing; online demo; peer-review; system architecture; scientific publications.

I. The IPOL journal

This article discusses the development of the online journal IPOL (Image Processing on Line), which seems to be one of the first journals which publishes peer-reviewed articles associating each of them with an online execution demo which allows users to test the algorithms with their own images and parameter values. Both the source code of the algorithm and the source code of the demo are also peer-reviewed.

The journal was 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 3D reconstruction from several photographs of a scene, and automatic analysis and detection of features in images.

IPOL (www.ipol.im) seems to be the first journal for Image Processing which requires an online, executable, and reproducible demonstration of the algorithm being published. In the past three years, from 2011 to 2014, this journal has gone from an experimental web platform to an established journal with 80 articles published, 10 submitted and 30 in preparation.

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.

The philosophy of this journal follows the guidelines on the reproducible research topics also presented as 'Reproducible Research Standard' by Stodden [1], [2]. Before the last 10 years, the initiatives on reproducible research were not really common in the Image Processing domain. In the field of wavelet analysis, the initiative of Buckheit and Donoho highlights the reproducibility with the development of some *Matlab* routines called "*WaveLab*" defined as a base to reproduce all content of their published wavelet articles [3]. Later a similar approach was also provided by Fadili et al. with the "*MCALab*" routine dedicated to the *Matlab* implementation of image decomposition and inpainting algorithms. These approaches were also studied in the signal processing field [4] and in the cloud computing domain [5]. We can also mention the work of Gentleman which proposes a new modern format of publication with a mixture of code and text applied in bioinformatics [6].

Rooted in a credibility crisis, as pointed out by Donoho et al. in the scientific computation field [7], the initiative of the IPOL journal should potentially imply numerous benefits as those described by Donoho [8]: improve work habits, improved teamwork, great impact, less inadvertent competition, greater continuity, and cumulative impact.

The IPOL model requires authors to write the article including pseudo-code of the algorithm and release associated open source code. The code does not need to be optimized, but must implement faithfully the described algorithm, and is designed to be a representative implementation. The demonstration can also be written by the authors, but help can be asked from the technical staff. The reviewers role is to read carefully the article and the source code, and especially verify that they are in accordance. For example all parameters of the algorithm must be detailed in the article. Reviewers also test the demonstration with the proposed data, and can try it with their own uploaded data.

II. Some scientific and technical achievements of the journal to establish a state of the art

A. Image denoising

The papers on image denoising cover most of the state of the art in image denoising. The papers on this topic analyze and finalize the often incomplete algorithms reported in the literature and have posted their implementations online for the first objective comparison. These results are contained in the denoising section of the IPOL website. To summarize the content and impact of this section, we can mention:

- A complete state of the art of denoising giving the best noise reduction algorithms based on the different theories: wavelets, neural networks, sparse representations, Bayesian methods, variational methods. The direct comparison of the algorithms corrects what was thought to be an established hierarchy of the methods.
- The *Noise Clinic* [9], the first blind denoising algorithm to process any image, estimate its noise without further information, and denoise it correctly.
- The first generic noise model adaptable to most processed images (used for the Noise Clinic).

B. Stereovision

The stereovision category at IPOL contains five fundamental algorithms [10], [11], [12], [13], [14], and three others are in preparation. They include classic and emblematic algorithms for disparity map computation from a stereo pair. A notable IPOL workshop shows online the first complete processing chain that takes a pair or triplet of satellite images and delivers a 3D digital elevation model of the ground (http://dev.ipol.im/~carlo/ipol_demo/workshop_s2p).

III. Technical issues overcome through the development of IPOL

Most of these issues were resolved by N. Limare as part of his thesis dedicated to Reproducible Research in Image Processing [15].

A. Reference programming language

Some rules were needed to precise how algorithms can be implemented and how they should be evaluated by reviewers. The IPOL editorial board prepared a set of software guidelines with the list of authorized programming languages and libraries, and basic criteria of software quality expected by the journal. The first accepted programming language was C/C++, since these compiled languages are well standardized, reliable, fast, common, and stable libraries exist. Two issues remain in author requests. The first is a demand for Matlab support which is to become effective in 2015. The second is related to the use of less known libraries, which can still be used by authors if they are provided entirely within their source code.

B. Online publication

It was necessary to design a web interface managing all online articles. Its first classic role was to serve articles in PDF format and the corresponding source code package. But the main software design questions were raised by the online interface of the algorithmic demos, applicable on any image submitted by the users. The system also stores the results of every user experiment in the archives of each algorithm. This program managing all articles was designed in 2009 by N. Limare [16] and revised in 2013, before being placed in the hands of other editors.

It was necessary to design a web interface adapted to each paper and its particular demo. This framework is a Python-based webserver handling inputs and generating web pages in result of every user interaction with a demo: selecting or updating a file, changing a parameter of the algorithm, choosing a part of an image, a point, a line, etc.

The editorial question then was: how to manage the editorial burden of designing a different web page for each of the articles? For the first articles it was designed by one person (N. Limare), who then provided guidelines for the subsequent articles. Fortunately, it became soon clear that the number of types of articles and online demos was limited. This is the basis for the next step, an automated demo generation from a limited catalog of standard interactions and display feature, which is sketched in the next section.

C. Improvements in the demo system architecture

After more than four years of existence of IPOL, we could identify some parts of the initial architecture of the demo system which need to be improved or extended.

In its first version, the demo system was designed as an object-oriented monolithic kernel which performs each of the required operations, including web interaction, format conversion, or managing the archive of experiments. Therefore, the source code of the demo system becomes too complex, with tightly interface-coupled components. And it is not easy to distribute the system over different machines.

To solve these problems, the next version of the demo system will have a modular architecture, where each specialized functionality will be put in a separate module. The monolithic kernel becomes then a controller, which executes operations in the specialized modules using a webservice.

Perhaps the most important question is how to ensure backward compatibility with the existing system. Right now, more than one hundred demos are running and any modification made into the system should never make it stop working. In order to do it, the demos will contain a version number and a Factory method pattern [17] will be used to create the modular architecture while keeping the old functionality. For demos in version #1 (current system), the factory will simply instantiate the same objects which are used now. For new demos using version #2, the factory will instead create objects which communicate with the corresponding modules using their webservice. Thanks to the Factory pattern the controller does not need to know the concrete type and will work transparently using generic classes.

With the new modular architecture, it will be possible to distribute the system along several machines and improve its scalability. Also, it will make easier to work on each individual module, since its code will be totally decoupled from the demo controller and will communicate using the proper webservice API.

D. Quality of articles and a "natural selection"

IPOL articles must be impeccable: every proposed algorithm needs to be reproduced online on any given image chosen or uploaded by readers. Each algorithm must be described in detail by the authors to check its conformity to the corresponding source code. The online execution of each algorithm must take no more than 30 seconds to spare the patience of readers (this often requires parallelization). Finally the results of the algorithm need to be useful, interesting, and competitive.

The vast majority of the algorithms published in Image Processing and Computer Vision journals are far from meeting the IPOL's publication criteria. Most failed IPOL projects aborted when the authors realized that the algorithm described was incomplete, did not give all the results described on paper, was not feasible in a reasonable run time, or only worked on a certain type of data but not on *any* image. For example, many algorithms for stereovision presuppose a pair of fronto-parallel cameras. This is certainly a convenient simplification, but reduces the algorithm to a *proof of concept*, unsuited for any practical application on an actual stereo pair. Other authors with a potentially suitable algorithm were put off by the work of cleaning up and editing of their code to achieve a publishable code. It quickly became apparent that the IPOL journal would not reject a significant proportion of submissions, since most non-viable projects fail even before they get submitted.

E. Progress towards the establishment of a full state of the art in each of the main sections of the journal

The IPOL journal is currently divided into 19 sections, each containing an average of 4 articles and preprints. The editorial board tried to identify and steer the publication of algorithms representing each a very substantial portion of the state of the art. When choosing the methods to include, two criteria were followed. First, the most efficient algorithms must be present, and second all theories and methods that were proposed for a given problem should be represented, even though they are not (or no longer) considered the best.

The sections Color and Contrast (10 articles), Denoising (15 articles), Demosaicking (6 articles), Interpolation (4 articles), Optical Flow (6 articles) and Vision Through Turbulence (3 articles) cover most of their respective state of the art, both in terms of best performance and of description of the main mathematical techniques. The other sections (3D, Blur, Computational Photography, Geometry, Infrared, Learning and Detection, Inpainting, Image Comparison, PDE, Stereovision, Texture) contain each at least three fundamental algorithms, but are still incomplete with respect to the state of the art. The sections are listed in alphabetical order, but can also be grouped according to the application purpose.

The sections Blur, Calibration, Denoising, Color and Contrast, Demosaicking and Interpolation handle the main issues raised by the treatment of a raw image arriving at a camera, including the calibration of the camera itself. The 3D and Stereovision sections address the main issues of 3D reconstruction from a set of images. Other sections focus on analysis methods to extract information: detection of segments, faces, comparison of two images with invariant viewpoint and lighting, texture analysis, texture synthesis, restoration of missing parts (Inpainting), motion analysis (Optical Flow). The last sections address classic or new acquisition methods (Computational Photography, Infrared.)

The advantages of a reproducible algorithm should be obvious, but let us illustrate it. In stereovision, a reference benchmark for disparity map computation is known as the Middlebury stereo evaluation (http://vision.middlebury.edu/stereo/). More than 150 methods are put in competition on four stereo pairs, very few of them providing a complete algorithm or a source code. Authors just upload their images to the website, results are compared to the ground truth and they get ranked in the public database. During the implementation of two classic methods using adaptive neighborhoods [12], [14], it was discovered that a post-processing must be applied to the main algorithm presented in [12] in order to reproduce similar results (there is no mention of a post-processing in the original article [18]). Such a post-processing like the guided filter algorithm [19] implemented in [14], slightly degraded performances. Moreover even if its results are faithful to the Matlab code provided by the authors, the benchmarked results are obtained using their GPU implementation (not public), which obviously does not follow exactly the presented algorithm.

F. Criticism to IPOL

Authors and publishers have criticized IPOL for:

i. the excessive effort required to arrive at a reproducible article (rigor, acceleration, running on any data) and to describe the code and deliver perfectly readable code;
ii. the excessive length of the peer review (code reviewers are slow and very exigent);
iii. the large number of objects to be published with the article: PDF file, source code, viable demo written in Python;
iv. the journal being four years old did not yet receive an official impact factor. IPOL has already more than 500 citations on Google scholar for an average existence of two years for its articles (2011, 2012, 2013, 2014). Articles published in 2011 and 2012 have been cited more than 20 times on average, with the top one reaching 395 citations.
v. to some authors it is frustrating to work so hard to elucidate algorithms designed by others. Even when those algorithms are inventive and original, they are generally incomplete and require considerable research effort. Nevertheless, the symbolic benefit of this work goes back in good part to the originators.
vi. the restricted number of authorized libraries to support source code (like *libjpeg*, *libtiff*, *fftw*, etc.) can be a criticism in particular from authors who are already invested in a library of a specific domain like computer vision with *OpenCV* [20], computational geometry with *CGal* [21] or discrete geometry with *DGtal* [22]. Moreover, such library contributions are in general reviewed before integration in their main framework. Initially, the IPOL contribution relying on such a library was possible only with the inclusion of the library source code in the contribution code archive (as done for several articles of

the special issue on *DGCI 2011* conference [23], [24]). However, this restriction is going to be removed with the already allowed use of *Matlab* based demonstration and the use (under study) of *OpenCV*. The other libraries might be accepted in the future if needed.

G. Authors and publishers praised IPOL for:

vii. the immediate impact of their publications, reflected in the archive of experiments showing strong interest and use of the demos for technical issues far beyond the interests of the image processing community;
viii. authors would like to link the experiments in the archive to their application domains and reveal what type of data remains a challenge;
ix. the promise that this impact will continue due to the very existence of the demo;
x. the fact that authors publishing these articles gain a tangible industrial and academic credibility, which greatly facilitates obtaining research funding. As we observed, research funding agencies (ERC, ONR, ANR, DGA, CNES, FUI, ...) increased significantly their stake in the research teams using the journal because of the reproducibility of their research.

IV. Conclusion: pseudo-code is the main production

Conceptually, the main IPOL innovation is that the publication of algorithms forces researchers to write a very compact, fully reproducible pseudo-code and enables a universal disclosure of algorithms. Even the best written code is not easily readable and reusable, just because of its necessary prolixity: no understatement or ellipsis is possible when programming. A pseudo-code can be shorter by one or two orders of magnitude, while remaining unambiguous for a mathematician programmer. In sum, the mathematical definition of functions is often much shorter than their program counterpart. While the computer program is perishable, the pseudo-code is not. Programming languages, program libraries, and operating systems change. This actually causes recurrent maintenance problems. But the pseudo-code remains. The pseudo-code of IPOL articles occupies between one and four pages maximum per article. It can be read in a few hours and reprogrammed in any language in reasonable time. Thus, it will appear as the main production of the journal. Noticeably, some code published at IPOL has been licensed to several companies.

V. What is next? Outline of a new way of doing research

A. The incremental research

More than 5,000 articles are published per year (including major conferences, specialized press, and newspapers) on Computer Vision and Image Processing. Approximately 100,000 articles have been published on the subject in the last thirty years. Most of these articles describe one or more algorithms. Nevertheless, a list of barely 100 algorithms would cover most of Computer Vision and Image Processing existing goals. For each of these objectives there may be at most three or four variants or different competitive approaches. It follows that the state of the art of the subject might be eventually described rigorously with some 200 articles, and exhaustively with less than 400. Thus, a yearly production of some 40 articles at IPOL might exhaust the state of the art, old and new in seven more years.

Does it mean that if the new publishing method proposed by IPOL succeeds, it will seal the end of Image Processing? Not at all! Indeed, it should encourage new types of articles focusing on incremental research, where established and published algorithms (e.g. at IPOL) will be revised, combined, improved to achieve more sophisticated results and applications. This virtuous cycle, which exists in other disciplines such as the analysis of the genome or the numerical analysis of partial differential equations, has not yet been kick-started in Image Processing. Experiments which took years in a biological lab ten years ago sometimes might nowadays take months or even days because each new lab technique is soon industrialized. This should by all means happen in all disciplines producing algorithms, where it is particularly easy to make finalized research available to all. IPOL simply demonstrates that it is technically possible and even easy to do so, and that it boosts the teams that adopt this work methodology.

Acknowledgments

Work partly founded by the European Research Council (advanced grant Twelve Labours), the Office of Naval research (ONR grant N00014-14-1-0023), and ANR-DGA project ANR-12-ASTR-0035.

References

[1] V. Stodden, “The legal framework for reproducible scientific research: Licensing and copyright,” *Computing in Science & Engineering*, vol. 11, no. 1, pp. 35–40, 2009.
[2] ——, “Enabling reproducible research: Open licensing for scientific innovation,” *International Journal of Communications Law and Policy, Forthcoming*, 2009.
[3] J. B. Buckheit and D. L. Donoho, *Wavelab and reproducible research*. Springer, 1995.
[4] P. Vandewalle, J. Kovacevic, and M. Vetterli, “Reproducible research in signal processing,” *Signal Processing Magazine, IEEE*, vol. 26, no. 3, pp. 37–47, 2009.
[5] J. T. Dudley and A. J. Butte, “Reproducible in silico research in the era of cloud computing,” *Nature biotechnology*, vol. 28, no. 11, p. 1181, 2010.
[6] R. Gentleman, “Reproducible research: A bioinformatics case study,” *Statistical applications in genetics and molecular biology*, vol. 4, no. 1, 2005.
[7] D. L. Donoho, A. Maleki, I. U. Rahman, M. Shahram, and V. Stodden, “Reproducible research in computational harmonic analysis,” *Computing in Science & Engineering*, vol. 11, no. 1, pp. 8–18, 2009.
[8] D. L. Donoho, “An invitation to reproducible computational research,” *Biostatistics*, vol. 11, no. 3, pp. 385–388, 2010.
[9] M. Lebrun, M. Colom, and J.-M. Morel, “The Noise Clinic: a Blind Image Denoising Algorithm,” *Image Processing On Line*, vol. 5, pp. 1–54, 2015.
[10] P. Monasse, “Quasi-Euclidean Epipolar Rectification,” *Image Processing On Line*, vol. 1, 2011.
[11] V. Kolmogorov, P. Monasse, and P. Tan, “Kolmogorov and Zabihs Graph Cuts Stereo Matching Algorithm,” *Image Processing On Line*, vol. 4, pp. 220–251, 2014.
[12] P. Tan and P. Monasse, “Stereo Disparity through Cost Aggregation with Guided Filter,” *Image Processing On Line*, vol. 4, pp. 252–275, 2014.
[13] G. Facciolo, N. Limare, and E. Meinhardt-Llopis, “Integral Images for Block Matching,” *Image Processing On Line*, vol. 4, pp. 344–369, 2014.

[14] L. Fernández Julià and P. Monasse, "Bilaterally Weighted Patches for Disparity Map Computation," *Image Processing On Line*, vol. 5, pp. 73–89, 2015.

[15] N. Limare, "Reproducible research, software quality, online interfaces and publishing for image processing," Ph.D. dissertation, École Normale Supérieure de Cachan-ENS Cachan, 2012.

[16] N. Limare and J.-M. Morel, "The ipol initiative: Publishing and testing algorithms on line for reproducible research in image processing," *Procedia Computer Science*, vol. 4, pp. 716–725, 2011.

[17] E. Gamma, R. Helm, R. Johnson, and J. Vlissides, *Design patterns: elements of reusable object-oriented software*. Pearson Education, 1994.

[18] K.-J. Yoon and I. Kweon, "Adaptive support-weight approach for correspondence search," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 4, pp. 650–656, 2006, http://dx.doi.org/10.1109/TPAMI.2006.70.

[19] C. Rhemann, A. Hosni, M. Bleyer, C. Rother, and M. Gelautz, "Fast cost-volume filtering for visual correspondence and beyond," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2011, pp. 3017–3024. [Online]. Available: http://dx.doi.org/10.1109/CVPR.2011.5995372

[20] G. Bradski, *Dr. Dobb's Journal of Software Tools*, 2000.

[21] "CGAL, Computational Geometry Algorithms Library," http://www.cgal.org.

[22] "DGtal: Digital geometry tools and algorithms library," http://dgtal.org.

[23] B. Kerautret and J.-O. Lachaud, "Meaningful Scales Detection: an Unsupervised Noise Detection Algorithm for Digital Contours," *Image Processing On Line*, vol. 4, pp. 98–115, 2014.

[24] D. Coeurjolly, B. Kerautret, and J.-O. Lachaud, "Extraction of Connected Region Boundary in Multidimensional Images," *Image Processing On Line*, vol. 4, pp. 30–43, 2014.