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

M. Colom[1], B. Kerautret[2], N. Limare[1], P. Monasse[3], and J.M. Morel[1]

[1]CMLA, ENS Cachan [2]LORIA, Univ. de Lorraine [3]LIGM, Univ. Paris-Est

Worskshop RCBR 2015, NTMS 2015 July 29, Paris

1. The IPOL Journal: context of reproductible research (1)

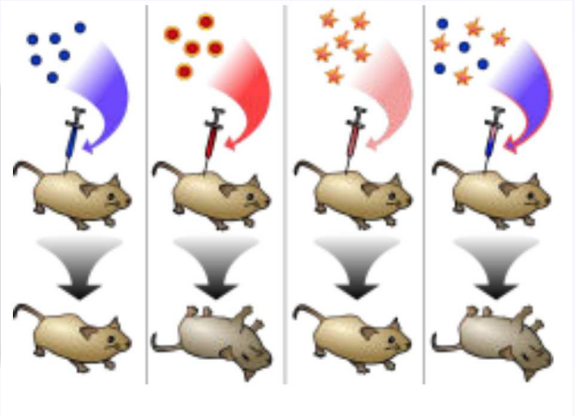
Reproductible research in sciences:

- *Theoretical scientists* share demonstrations;

1. The IPOL Journal: context of reproductible research (1)

Reproductible research in sciences:

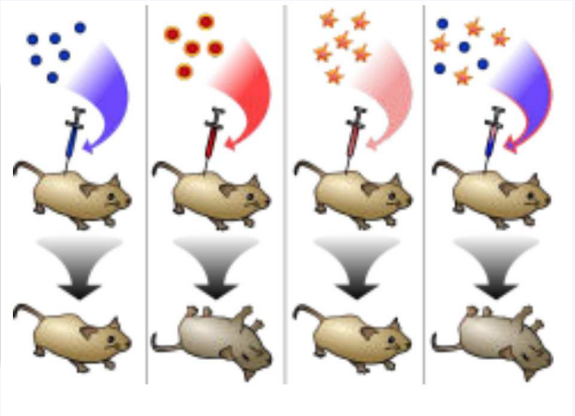
- *Theoretical scientists* share demonstrations;
- *Experimental scientist* share procedures;



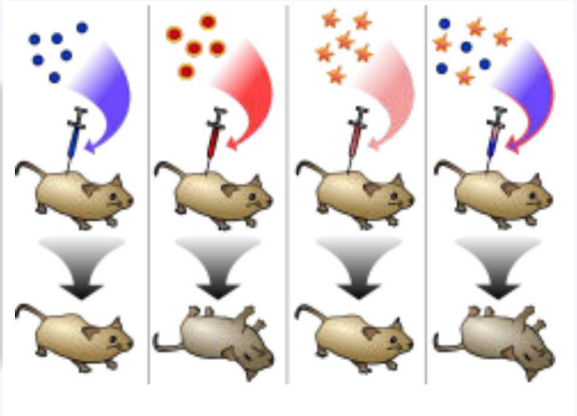
1. The IPOL Journal: context of reproductible research (1)

Reproductible research in sciences:

- *Theoretical scientists* share demonstrations;
- *Experimental scientist* share procedures;
- *Computational scientists... ?*



1. The IPOL Journal: context of reproductible research (1)



Reproductible research in sciences:

- *Theoretical scientists* share demonstrations;
- *Experimental scientist* share procedures;
- *Computational scientists... ?*

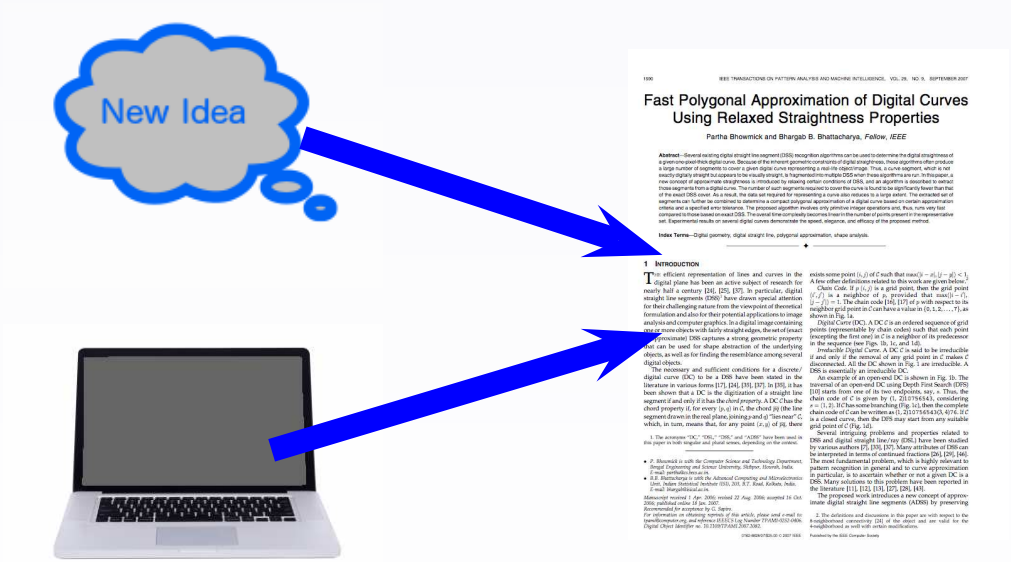
Computer Science:

- Description of methods/algorithms;
- description often limited (constraints on page limits);
- parameters not given or not well described;
- steps of pre/post processing missing.

1. The IPOL Journal: context of reproductible research (2)

Research in Computer Science:

1. New idea;
2. demonstration, implementation;
3. article publication.



Fast Polygonal Approximation of Digital Curves Using Relaxed Straightness Properties

Partha Bhowmick and Bhargab B. Bhattacharya, *Fellow, IEEE*

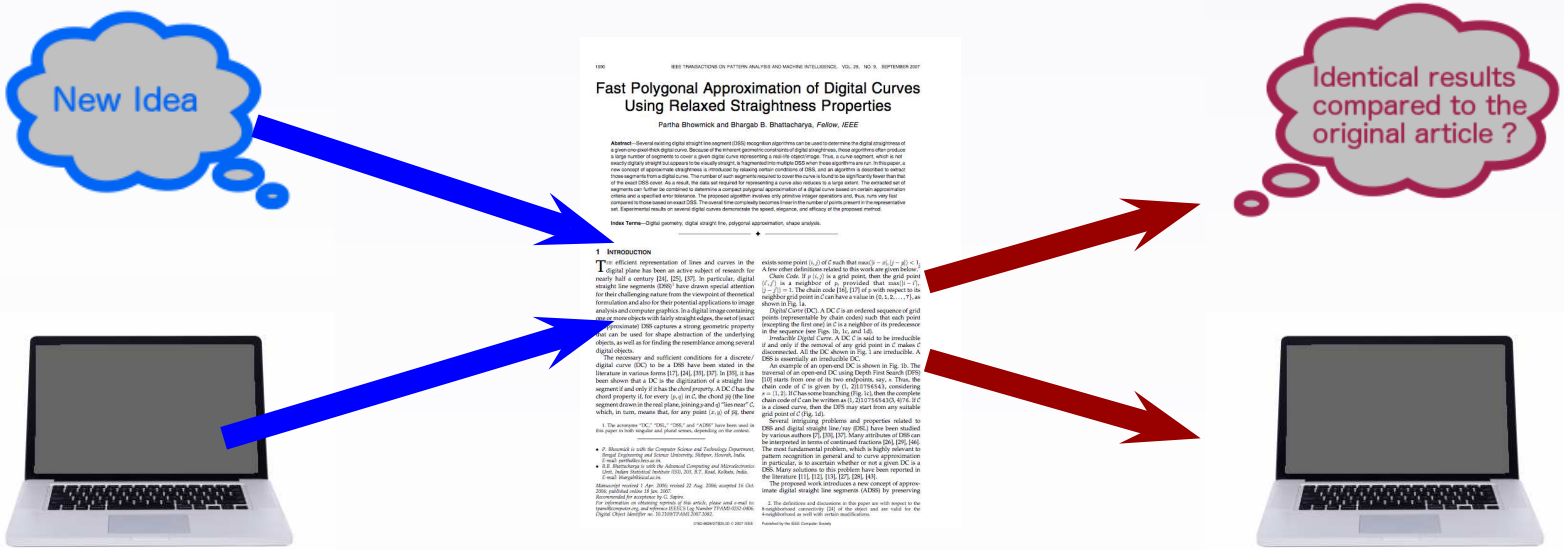
1. The IPOL Journal: context of reproductible research (2)

Research in Computer Science:

1. New idea;
2. demonstration, implementation;
3. article publication.

Reusable Research:

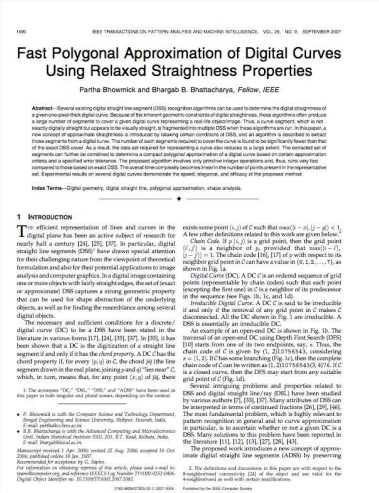
1. Article which seems interesting;
2. re-implement the algorithm;
3. conformity of the results with the original.



1. The IPOL Journal: context of reproductible research (3)

Frequent difficulties in computer science (image processing):

- Source code often not available (or not reviewed);
- quality/stability of the results not easy to analyze;
- testing with different input data not possible.



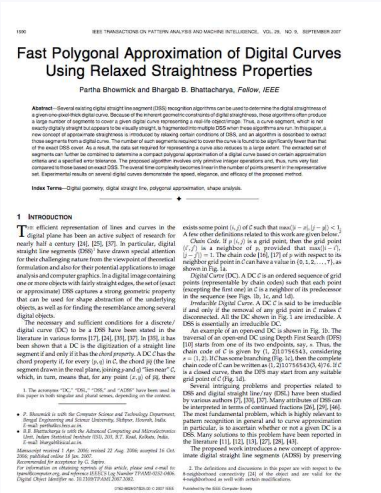
Fast Polygonal Approximation of Digital Curves Using Relaxed Straightness Properties

Partha Bhowmick and Bhargab B. Bhattacharya, *Fellow, IEEE*

1. The IPOL Journal: context of reproductible research (3)

Frequent difficulties in computer science (image processing):

- Source code often not available (or not reviewed);
- quality/stability of the results not easy to analyze;
- testing with different input data not possible.



Fast Polygonal Approximation of Digital Curves Using Relaxed Straightness Properties

Partha Bhowmick and Bhargab B. Bhattacharya, *Fellow, IEEE*

Consequences:

- Comparisons and experiments difficult;
- potential time loss for the reader;
- limits the diffusion of research.

1. The IPOL Journal: context of reproductible research (4)

Providing source code/data

- $\oplus$ A real added value for the publication;
- $\oplus$ increases the impact/comparisons;
- $\ominus$ software is not really acknowledged;
- $\ominus$ important effort (documentation, tests, user maintenance).

```
#include <iostream>
#include "DGtal/base/Common.h"
#include "DGtal/io/readers/VolReader.h"
#include "DGtal/io/Display3D.h"

#include "DGtal/io/DrawWithDisplay3DModifier.h"
#include "DGtal/images/ImageSelector.h"
#include "DGtal/images/imagesSetsUtils/SetFromImage.h"
#include "DGtal/helpers/StdDefs.h"
#include "ConfigExamples.h"


using namespace std;
using namespace DGtal;
using namespace Z3i;


int main( int /*argc*/, char** /*argv*/ )
{
  std::string inputFilename = examplesPath + "samples/Al.100.vol";
  //! [ExampleDisplay3DToOFF]
  Display3D<Space, KSpace> viewer;
  typedef ImageSelector < Z3i::Domain, int>::Type Image;
  Image image = VolReader<Image>::importVol(inputFilename);
  Z3i::DigitalSet set3d (image.domain());
  SetFromImage<Z3i::DigitalSet>::append<Image>(set3d, image, 0,255);

  viewer << set3d ;
  viewer >> "exportMeshToOFF.off";
 //! [ExampleDisplay3DToOFF]

  return 0;
}
```

1. The IPOL Journal: context of reproductible research (4)

Providing source code/data

- $\oplus$ A real added value for the publication;
- $\oplus$ increases the impact/comparisons;
- $\ominus$ software is not really acknowledged;
- $\ominus$ important effort (documentation, tests, user maintenance).

```
#include <iostream>
#include "DGtal/base/Common.h"
#include "DGtal/io/readers/VolReader.h"
#include "DGtal/io/Display3D.h"

#include "DGtal/io/DrawWithDisplay3DModifier.h"
#include "DGtal/images/ImageSelector.h"
#include "DGtal/images/imagesSetsUtils/SetFromImage.h"
#include "DGtal/helpers/StdDefs.h"
#include "ConfigExamples.h"


using namespace std;
using namespace DGtal;
using namespace Z3i;


int main( int /*argc*/, char** /*argv*/ )
{
  std::string inputFilename = examplesPath + "samples/Al.100.vol";
  //! [ExampleDisplay3DToOFF]
  Display3D<Space, KSpace> viewer;
  typedef ImageSelector < Z3i::Domain, int>::Type Image;
  Image image = VolReader<Image>::importVol(inputFilename);
  Z3i::DigitalSet set3d (image.domain());
  SetFromImage<Z3i::DigitalSet>::append<Image>(set3d, image, 0,255);

  viewer << set3d ;
  viewer >> "exportMeshToOFF.off";
 //! [ExampleDisplay3DToOFF]

  return 0;
}
```

Software Diffusion

- Specialized journals in software:
  Source Code for Biology and Medicine, Journal of Open Research Software, Computing in Science and Engineering, ...
- Diffusion platform:
  RunMyCode / Run&Share, Figshare, DataDryad, Harvard Dataverse, Ubiquity Metajournals, Zenodo (with doi) ...

$\Rightarrow$ no validation, no scientific review (reliability and durability problem).

1. The IPOL Journal: origin and motivation

Origin:

- Journal started in October 2009;
- under the initiative of Nicolas Limare, Jean-Michel Morel and the Image Processing team at the CMLA lab (ENS-Cachan);
- first article published in 2010.

1. The IPOL Journal: origin and motivation

Origin:

- Journal started in October 2009;
- under the initiative of Nicolas Limare, Jean-Michel Morel and the Image Processing team at the CMLA lab (ENS-Cachan);
- first article published in 2010.

Motivation [Limare & Morel 2009]:

- Reproductible research;
- new way to publish research results;
- allows everybody to test the algorithms;
  $\Rightarrow$ with their own images
- independent of the platform (the demos execute on the server side and the results are shown to the user using a web interface).

1. IPOL Journal: principle and current form (1)

Characteristics:

- Research journal in **image** processing;
- each article contains a description of **one algorithm** and its **source code**;
- association of each **article** with its **online demonstration**, with **archived** experiments;
- the peer-review process includes the **article**, **demo**, and **source code**;
- *Open Science* journal and **Reproductible Research**.

1. IPOL Journal: principle and current form (2)

Philisophy of the journal:

- Follows the guideline on reproductible research topics;
- reproductible research standard [Stodden 09a] [Stodden 09b];
- answer to credibility crisis in scientific computation (as pointed out by Donoho [Donoho *et al.* 09]).

1. IPOL Journal: principle and current form (2)

Philisophy of the journal:

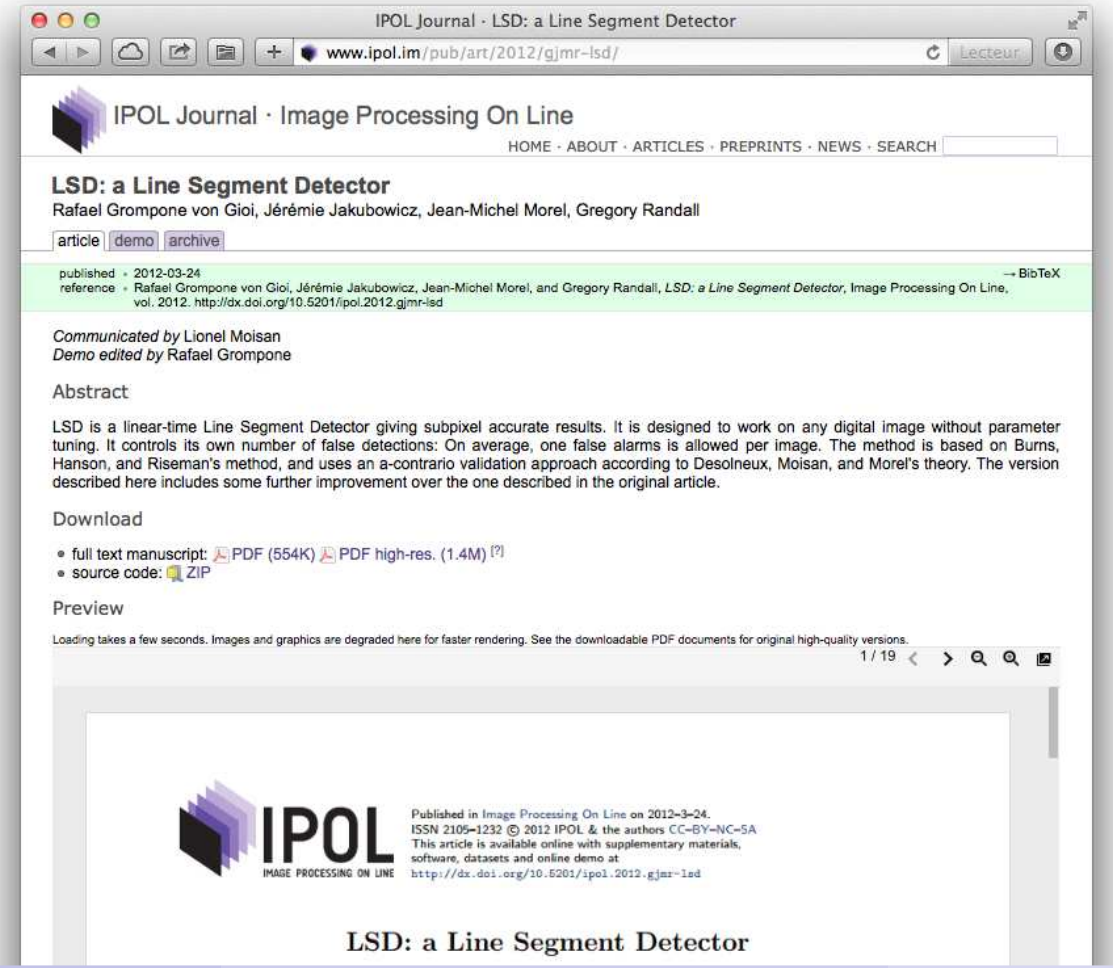
- Follows the guideline on reproductible research topics;
- reproductible research standard [Stodden 09a] [Stodden 09b];
- answer to credibility crisis in scientific computation (as pointed out by Donoho [Donoho *et al.* 09]).

What IPOL is not:

- IPOL publishes algorithms along with their implementation, but not compiled software;
- IPOL is not a software library (each code has minimal dependencies);
- IPOL is not a software or code diffusion platform.

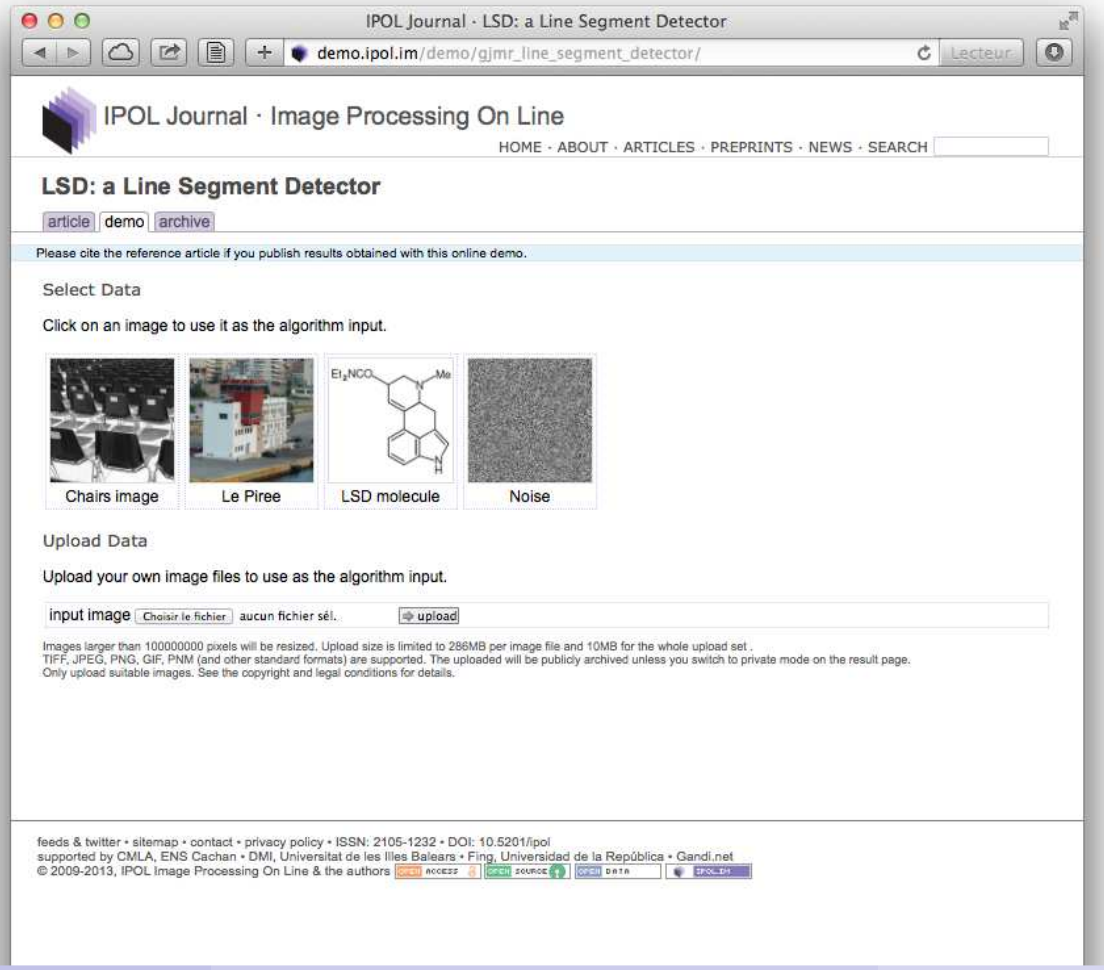
1. IPOL Journal: principle and current form (3)

- Current form: “classic” (with online PDF).



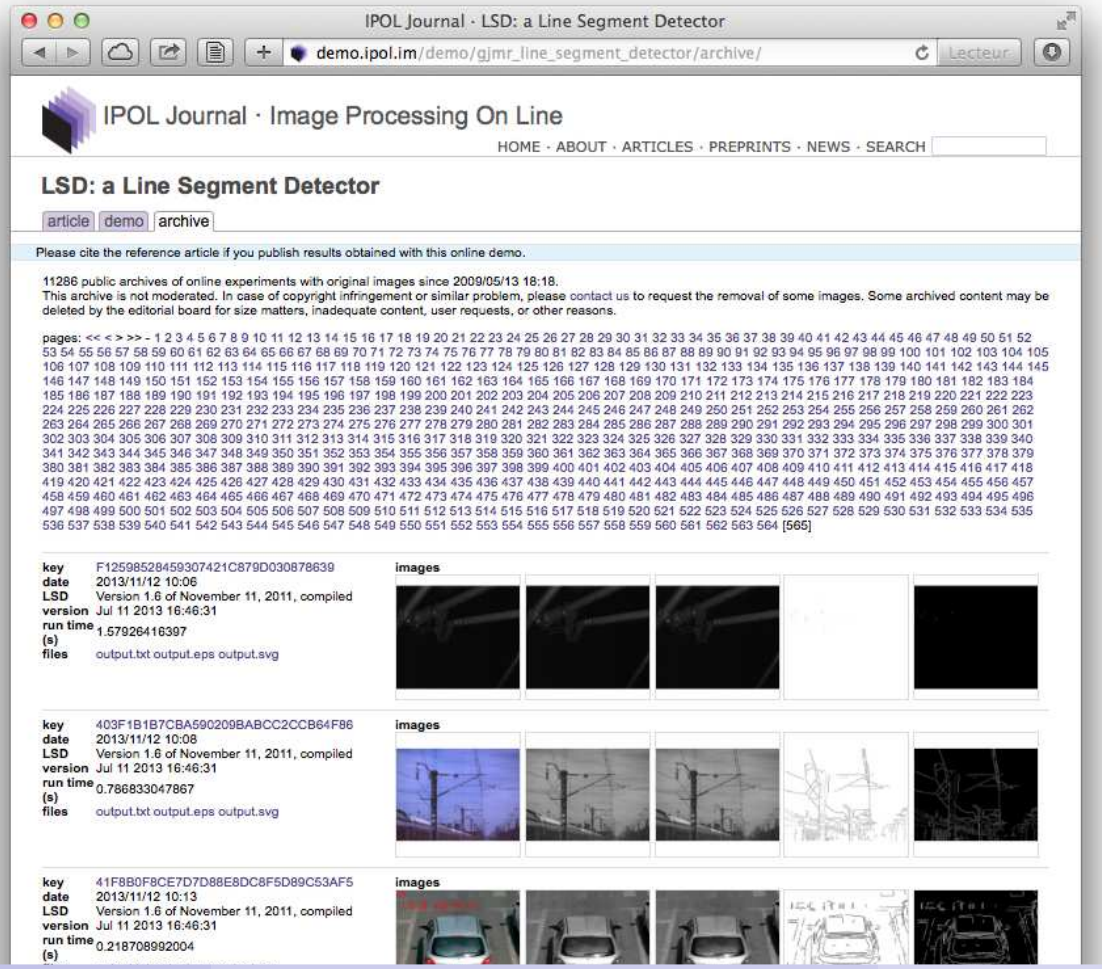
1. IPOL Journal: principle and current form (3)

- Current form: “classic” (with online PDF).
- Associated demos.



1. IPOL Journal: principle and current form (3)

- Current form: “classic” (with online PDF).
- Associated demos.
- Archive containing experiments with data uploaded by users.



1. IPOL Journal: editorial structure

Same aspects as a classical journal:

- Editorial project, editorial committee;
- articles, authors, editors;
- reviewing process and validation;
- ISSN, DOI;
- special issues;
- currently indexed by:
  Scirus, Google Scholar, DBLP, DOAJ, SHERPA/RoMEO, Héloïse, WorldCat, CrossRef, Ulrich, Index Copernicus, PBN, JGate, VisionBib, CVonline, JournalSeek, and NewJour.

1. IPOL Journal: editorial structure

Same aspects as a classical journal:

- Editorial project, editorial committee;
- articles, authors, editors;
- reviewing process and validation;
- ISSN, DOI;
- special issues;
- currently indexed by:
  Scirus, Google Scholar, DBLP, DOAJ, SHERPA/RoMEO, Héloïse, WorldCat, CrossRef, Ulrich, Index Copernicus, PBN, JGate, VisionBib, CVonline, JournalSeek, and NewJour.

Software point of view:

- Each article should propose an implementation;
- reviewing step, verification, validation, and publication;
- reviewer: **check the correspondence between the algorithm description in the article and code** (+ code readability and code documentation).

2. Scientific & technical achievements to establish a state of the art (1)

Image Denoising

- Papers on image denoising cover most of the state of the art in image denoising.
  $\Rightarrow$ analyze and finalize the often incomplete algorithms.
- Online implementation allows the first objective comparison.
- complete state of the art of denoising

$\Rightarrow$ See the noise Clinic [Lebrun *et al.* 15]

2. Scientific & technical achievements to establish a state of the art (1)

Image Denoising

- Papers on image denoising cover most of the state of the art in image denoising.
  $\Rightarrow$ analyze and finalize the often incomplete algorithms.
- Online implementation allows the first objective comparison.
- complete state of the art of denoising

$\Rightarrow$ See the noise Clinic [Lebrun *et al.* 15]

Stereovision

The stereovision category at IPOL contains five fundamental algorithms:

- Quasi-Euclidean Epipolar Rectification [Monasse 11].
- Kolmogorov & Zabih's graph cuts stereo matching algorithm [Kolmogorov *et al.* 14].
- Stereo disparity through cost aggregation with guided filter [Tan & Monasse 14].
- Integral images for block matching [Facciolo *et al.* 14].
- Bilaterally weighted patches for disparity map computation [Fernandez & Monasse 15].

$\Rightarrow$ Others in preparation.

2. Scientific & technical achievements to establish a state of the art (2)

Stereovision

- First workshop demo delivering a 3D digital elevation model of the ground from satellite images.
- available here: `http://dev.ipol.im/~carlo/ipol_demo/workshop_s2p`

3. Technical Issues Overcome Through the Development of IPOL

First problem: reference programming languages

The chosen languages must:

- have a **stable API**;
- be **used intensively** by reasearchers and the industry;
- have stable **standard libraries**.

3. Technical Issues Overcome Through the Development of IPOL

First problem: reference programming languages

The chosen languages must:

- have a **stable API**;
- be **used intensively** by reasearchers and the industry;
- have stable **standard libraries**.

Accepted languages and libraries

- First accepted languages: **ANSI C/C++**;
- **currently accepted** libraries: `libtiff`, `libjpeg`, `libpng`, `zlib`, `FFTW`, `GSL`, `Eigen`, `cblas`, and `clapack`;
- currently accepted `MATLAB` toolboxes: `Image Processing`, `Optimization`, `Wavelet`;
- new accepted frameworks: `MATLAB`, `Python` (with `NumPy` and `SciPy`);
- other languages might be considered and accepted.

Second problem: design an online demo system

A **demo system** was created for this purpose, from scratch. It has to **manage the execution** of each online experiment:

- The **parameters and result pages** are different in each demo;
- the **test images** are different in each demo;
- users must be able to **upload their own images**;
- it has to be **fast** enough to allow online execution $\rightarrow$ **Multiple CPU** system;
- it has to **archive** user experiments (input data, input parameters, and results).

Second problem: design an online demo system

A **demo system** was created for this purpose, from scratch. It has to **manage the execution** of each online experiment:

- The **parameters and result pages** are different in each demo;
- the **test images** are different in each demo;
- users must be able to **upload their own images**;
- it has to be **fast** enough to allow online execution $\rightarrow$ **Multiple CPU** system;
- it has to **archive** user experiments (input data, input parameters, and results).

Problems detected in the current system

The current system is **usable** and **functional**, but we detected **several problems**:

- Creating a new demo implies **coding** in Python and **designing HTML templates**;
- **non-scalable**, **too complex internal structure**, and with **lack of modularization**.

$\Rightarrow$ **Now moving towards a modular system with automatic demo generation.**

Problems in the current demo system

- Most of the **problems** related to the **architecture** of the system;
- system designed as an object-oriented **monolithic** kernel;
- **too complex**, with **tightly interface-coupled** components;
- **non-scalable**;
- **not easy to distribute** the system over different machines.

Problems in the current demo system

- Most of the **problems** related to the **architecture** of the system;
- system designed as an object-oriented **monolithic** kernel;
- **too complex**, with **tightly interface-coupled** components;
- **non-scalable**;
- **not easy to distribute** the system over different machines.

Improvements

We're currently working on **improving the system**

- **Modular architecture:**
  - Specialized standalone modules;
  - the monolithic kernel becomes a simple controller;
  - the core controller and the modules communicate via webservices.
- **Automatic demo generation:**
  - Each demo is specified using a simple textual description (name, type, and default value of the parameters and format of result page);
  - no need to code or design any page to build a new demo.

Quality of articles and a “natural selection”

Most **failed IPOL projects** aborted when:

- the described **algorithm** was **incomplete**;
- **did not give all the results** described in the paper;
- **run time** not reasonable;
- worked **only on a certain type of data**.

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

Identify algorithms representing a very substantial portion of the state of the art:

- Most **efficient algorithms** should be published;
- proposed theories and methods, **even though they are not** (or no longer) considered **the best**.

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

Identify algorithms representing a very substantial portion of the state of the art:

- Most **efficient algorithms** should be published;
- proposed theories and methods, **even though they are not** (or no longer) considered **the best**.

Some sections currently in IPOL

- Color and Contrast (10 articles)
- Denoising (15 articles)
- Demosaicking (6 articles)
- Interpolation (4 articles)
- Optical Flow (6 articles)
- Vision Through Turbulence (3 articles)
- . . .

Goal: cover must of their respective state of the art (performance and descriptions of main mathematical techniques)

Others sections

- 3D
- Blur
- Computational Photography
- Geometry
- Infrared
- Learning and Detection
- Inpainting
- Image Comparison
- PDE
- Stereovision
- Texture
- ...

→ **However, still incomplete!**

Example of advantages of a reproducible algorithms with Stereovision: **Middlebury stereo evaluation**

- One **method** [Fernandez & Monasse 15] was **implemented** from an algorithm on adaptive neighborhoods [Yoon & Kweon 06];
- the obtained **results were different!**
- it was discovered that a **post-processing** must be applied to the main algorithm;
- there is **no mention of a post-processing in the original article** [Yoon & Kweon 06]**!**

Criticism to IPOL

- **Excessive effort** (rigor, run on any data, code description);
- **excessive length** of the **peer review report** (including code review: slow and demanding);
- **large number of objects** to be published (article, source code, demo);
- no official **impact factor**;
- **frustrating to work on algorithms** designed by **others**;
- **restricted** number of **authorized libraries or toolboxes** (as `libjpeg`, `libtiff`, `FFTW`, `MATLAB Image Processing toolbox`, `...`)

Authors and publishers praised IPOL for:

- **Immediate impact** of their publication;
- **impact** due to the very **existence of the demo**;
- gain tangible industrial and academic **credibility**;
- facilitate obtaining **research funding** (ERC, ONR, ANR, DGA, CNES, FUI, . . .)

Conclusions (I)

- **Reproducible Research** redefines the **output of the research**: not only the article, but also the **source** code and the **data**.
- **IPOL** is a **complete** and **fully functional Reproducible Research journal**: **articles, demos, data**. Everything is **free** or **open source**.
- For the **authors** it takes **more effort** to write Reproducible Research articles, but:
  $\Rightarrow$ benefit is immediate (**credibility** and **number of citations**);
  $\Rightarrow$ for both the IPOL article and the one published in a different journal with a demo available in IPOL).
- It is important to **adapt the journal to the community** needs and usages:
  $\Rightarrow$ **accept** commonly used languages, libraries, and frameworks.

Conclusions (II)

Pseudo-code is the main production of IPOL over the source code itself:

- The pseudo-code **describes** the **significant parts** of the **algorithm**;
- it **does not contain all the details** needed to encode it using an actual programming language;
- the pseudo-code is aimed to be **readable**, and **reusable**;
  $\Rightarrow$ in general, it is between **one and two orders of magnitude shorter** than the actual source code.
- the pseudo-code is **unambiguous** to the mathematician reader.

What is Next? (I)

Facts:

- More than 5000 articles/year on Computer Vision and Image processing describing algorithms;
- however, about 200 of these articles would we enough to cover the CV and IP state of the art. Less than 400 if exhaustive.

IPOL is producing 40 articles/year $\rightarrow$ It can exhaust the state of the art (old and new) in about 7 years.

What is Next? (II)

Moving towards a new way to do research

- What is next, then? $\rightarrow$ Focus on incremental research.
- Antecedents: analysis of the genome.
- Cycle:
  1. review estabilished and published algoritms;
  2. combine them;
  3. improve them to achieve new and better results and applications.

In definitive, a new methodology to do research in CV and IP

Acknowledgement

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 I

[Stodden 09a] V. Stodden
The legal framework for reproducible scientific research: Licensing and copyright
Computing in Science & Engineering, vol. 11, no. 1, pp. 35–40, 2009.

[Stodden 09b] V. Stodden
Enabling reproducible research: Open licensing for scientific innovation
International Journal of Communications Law and Policy, Forthcoming, 2009.

[Buckheit & Donoho 95] J. B. Buckheit and D. L. Donoho
Wavelab and reproducible research
Springer, 1995.

[Gentleman 05] R. Gentleman
Reproducible research: A bioinformatics case study
Statistical applications in genetics and molecular biology, vol. 4, no. 1, 2005.

[Donoho *et al.* 09] 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.

[Donoho 10] D. L. Donoho
An invitation to reproducible computational research
Biostatistics, vol. 11, no. 3, pp. 385–388, 2010.

[Lebrun *et al.* 15] 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.

References II

[Monasse 11] P. Monasse
Quasi-Euclidean Epipolar Rectification
Image Processing On Line, vol. 1, 2011.

[Kolmogorov *et al.* 14] V. Kolmogorov, P.Monasse and P.Tan
Kolmogorov and Zabih's Graph Cuts Stereo Matching Algorithm
Image Processing On Line, vol. 4, pp. 220–251, 2014.

[Tan & Monasse 14] P. Tan and P. Monasse
Stereo Disparity through Cost Aggregation with Guided Filter
Image Processing On Line, vol. 4, pp. 252–275, 2014.

[Facciolo *et al.* 14] G. Facciolo, N. Limare, and E. Meinhardt-Llopis
Integral Images for Block Matching, Image Processing On Line, vol. 4, pp. 344–369, 2014.

[Fernandez & Monasse 15] L. Fernández Julià and P. Monasse
Bilaterally Weighted Patches for Disparity Map Computation
Image Process- ing On Line, vol. 5, pp. 73–89, 2015.

[Limare & Morel 11] 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.

[Gamma *et al.* 94] E. Gamma, R. Helm, R. Johnson and J. Vlissides
Design patterns: elements of reusable object-oriented software
Pearson Education, 1994.

References III

[Yoon & Kweon 06] 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.

[CGAL] Computational Geometry Algorithms Library,

`http://www.cgal.org`

[DGtal] DGtal: Digital Geometry tools and algorithms library,

`http://libdgtal.org`

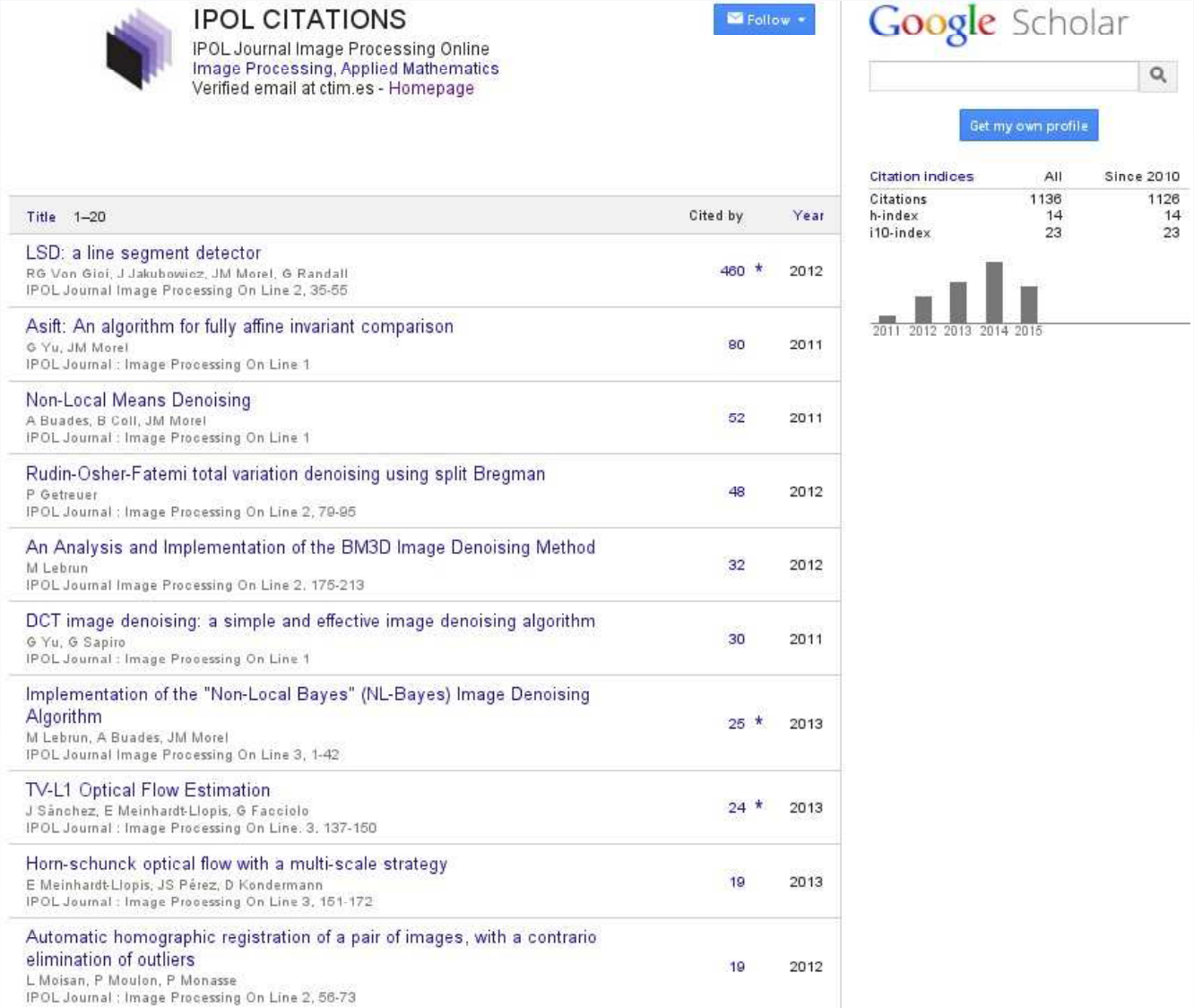
[Limare & Morel 2009] Limare N. and Morel, J-M (2009)

IPOL Project Presented at the CMLA Seminar

`http://www.ipol.im/news/20091022_cmla/s5.html`

CMLA ENS Cachan

IPOL citations (07/28/2015)



IPOL CITATIONS

Follow

IPOL Journal Image Processing Online
Image Processing, Applied Mathematics
Verified email at ctim.es - Homepage

Google Scholar

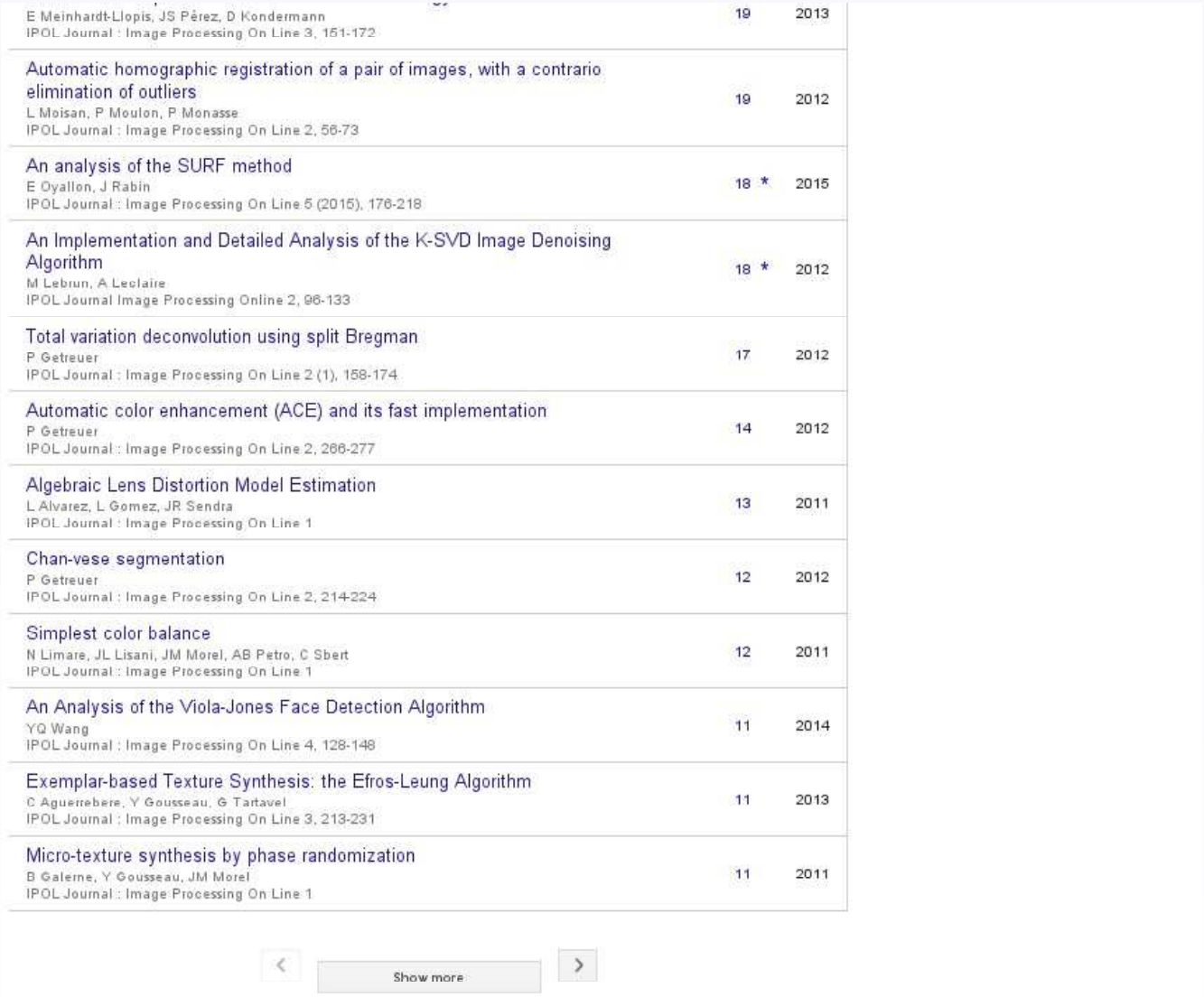
Get my own profile

Citation indices	All	Since 2010
Citations	1136	1126
h-index	14	14
i10-index	23	23

2011 2012 2013 2014 2015

Title 1–20	Cited by	Year
LSD: a line segment detector RG Von Gioi, J Jakubowicz, JM Morel, G Randall IPOL Journal Image Processing On Line 2, 35-55	460 *	2012
Asift: An algorithm for fully affine invariant comparison G Yu, JM Morel IPOL Journal : Image Processing On Line 1	80	2011
Non-Local Means Denoising A Buades, B Coll, JM Morel IPOL Journal : Image Processing On Line 1	52	2011
Rudin-Osher-Fatemi total variation denoising using split Bregman P Getreuer IPOL Journal : Image Processing On Line 2, 79-95	48	2012
An Analysis and Implementation of the BM3D Image Denoising Method M Lebrun IPOL Journal Image Processing On Line 2, 175-213	32	2012
DCT image denoising: a simple and effective image denoising algorithm G Yu, G Sapiro IPOL Journal : Image Processing On Line 1	30	2011
Implementation of the "Non-Local Bayes" (NL-Bayes) Image Denoising Algorithm M Lebrun, A Buades, JM Morel IPOL Journal Image Processing On Line 3, 1-42	25 *	2013
TV-L1 Optical Flow Estimation J Sánchez, E Meinhardt-Llopis, G Facciolo IPOL Journal : Image Processing On Line. 3, 137-150	24 *	2013
Horn-schunck optical flow with a multi-scale strategy E Meinhardt-Llopis, JS Pérez, D Kondermann IPOL Journal : Image Processing On Line 3, 151-172	19	2013
Automatic homographic registration of a pair of images, with a contrario elimination of outliers L Moisan, P Moulon, P Monasse IPOL Journal : Image Processing On Line 2, 56-73	19	2012

IPOL citations (07/28/2015)



E Meinhardt-Llopis, JS Pérez, D Kondermann IPOL Journal : Image Processing On Line 3, 151-172	19	2013
Automatic homographic registration of a pair of images, with a contrario elimination of outliers L Moisan, P Moulon, P Monasse IPOL Journal : Image Processing On Line 2, 56-73	19	2012
An analysis of the SURF method E Oyallon, J Rabin IPOL Journal : Image Processing On Line 5 (2015), 176-218	18 *	2015
An Implementation and Detailed Analysis of the K-SVD Image Denoising Algorithm M Lebrun, A Leclaire IPOL Journal Image Processing Online 2, 96-133	18 *	2012
Total variation deconvolution using split Bregman P Getreuer IPOL Journal : Image Processing On Line 2 (1), 158-174	17	2012
Automatic color enhancement (ACE) and its fast implementation P Getreuer IPOL Journal : Image Processing On Line 2, 266-277	14	2012
Algebraic Lens Distortion Model Estimation L Alvarez, L Gomez, JR Sendra IPOL Journal : Image Processing On Line 1	13	2011
Chan-vese segmentation P Getreuer IPOL Journal : Image Processing On Line 2, 214-224	12	2012
Simplest color balance N Limare, JL Lisani, JM Morel, AB Petro, C Sbert IPOL Journal : Image Processing On Line 1	12	2011
An Analysis of the Viola-Jones Face Detection Algorithm YQ Wang IPOL Journal : Image Processing On Line 4, 128-148	11	2014
Exemplar-based Texture Synthesis: the Efros-Leung Algorithm C Aguerrebere, Y Gousseau, G Tartavel IPOL Journal : Image Processing On Line 3, 213-231	11	2013
Micro-texture synthesis by phase randomization B Galerne, Y Gousseau, JM Morel IPOL Journal : Image Processing On Line 1	11	2011

< Show more >