

Image Processing on Line

<http://www.ipol.im>

A new way to publish algorithms?

A new way to organize research in an image analysis lab?

A new way to establish a state of the art?

Nicolas Limare (ENS Cachan),
Jean-Michel Morel (ENS Cachan)

Project partially funded by ERC, CNES, ONR N00014-97-1-0839

IPOL: <http://www.ipol.im>

The goal is to achieve « reproducible research » and therefore to publish:

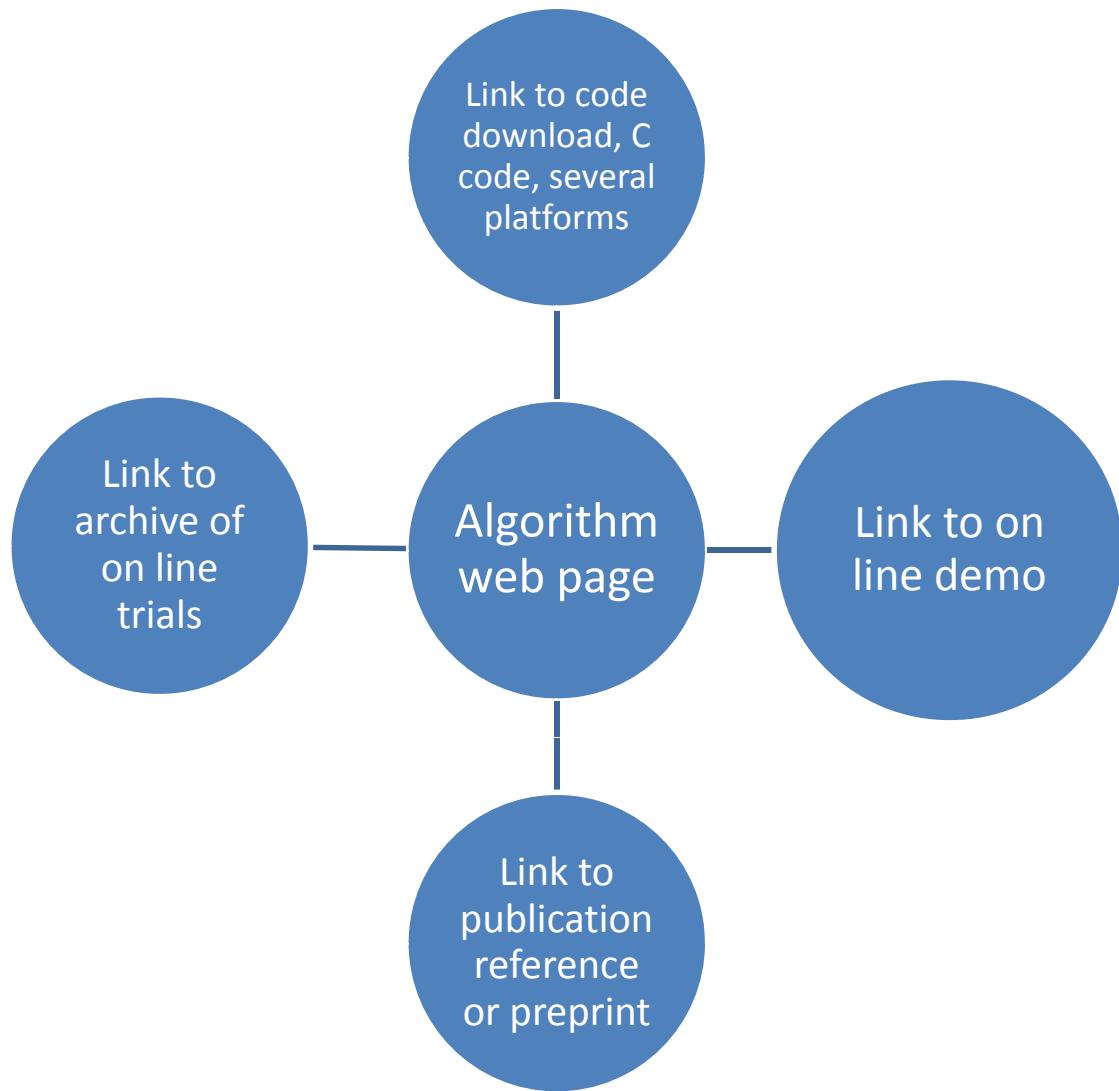
Litterary but complete and accurate description of each algorithm
Downloadable code (certified by referees to correspond to the description)
On line demo (also tested and certified by referees)
An open archive for each article with all on line experiments results

Advantages

Experiment sharing
Testing independent of any platform (no code download, no system requirement...)
Authors rewarded by a peer reviewed publication, complementary to a
classic « paper »

Online execution is particularly adapted to image processing because images and
video have standard formats and can be uploaded.

For each algorithm: a fourfold publication



Article = Web page + Online demo + Archive + Code

The screenshot shows a Mozilla Firefox browser window with the title bar "Affine SIFT (ASIFT) - Mozilla Firefox". The address bar displays the URL "http://mw.cmla.ens-cachan.fr/megawave/algo/asift/". The main content area shows a dark blue header with the text "AFFINE SIFT (ASIFT)". Below the header, there are several sections: "Contacts" (listing Jean-Michel Morel and Guoshen Yu), "References" (listing three academic papers), "Overview" (describing ASIFT as a fully affine invariant image comparison method), "When does it work?" (mentioning SIFT's limitations), and "Failure Cases" (linking to examples). A sidebar on the right lists links to "Contacts", "References", "Overview", "On Line Demo", "Software", "Dataset", "Examples", and "Failure Cases". The status bar at the bottom shows the URL "http://mw.cmla.ens-cachan.fr/megawave/algo/asift/" and the message "Terminé".

Fichier Édition Affichage Historique Marque-pages Outils ? | G Google

Page précédente Page suivante Actualiser Arrêter Accueil http://mw.cmla.ens-cachan.fr/megawave/algo/asift/ Aperçu Imprimer

Débuter avec Firefox À la une Google News France - ...

> mw > megawave > algo

AFFINE SIFT (ASIFT)

Contacts

- Jean-Michel Morel morel@cmila.ens-cachan.fr
- Guoshen Yu yu@cmap.polytechnique.fr

References

1. J.M. Morel and G.Yu, *ASIFT: A New Framework for Fully Affine Invariant Image Comparison*. SIAM Journal on Imaging Sciences, 2(2):438-469, 2009. [preprint](#)
2. G. Yu and J.M. Morel, *A Fully Affine Invariant Image Comparison Method*. Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), Taipei, 2009. [preprint](#)
3. J.M. Morel and G.Yu, *On the consistency of the SIFT Method*. Preprint, CMLA 2008-26, Sept 2008. [preprint](#)

Overview

A fully affine invariant image comparison method, Affine-SIFT (ASIFT) is introduced. While SIFT is fully invariant with respect to only four parameters namely zoom, rotation and translation, the new method treats the two left over parameters : the angles defining the camera axis orientation.

Against any prognosis, simulating all views depending on these two parameters is feasible. The method permits to reliably identify features that have undergone very large affine distortions measured by a new parameter, the transition tilt.

State-of-the-art methods hardly exceed transition tilts of 2 (SIFT), 2.5 (Harris-Affine and Hessian-Affine) and 10 (MSER). ASIFT can handle transition tilts up 36 and higher.

When does it work?

The SIFT method works to compare 2D objects or 3D objects with flat enough details, taken from similar view angles but at arbitrary distances.

The typical failure cases are:

htt

Terminé

4

Example : ASIFT: affine invariant image, SIFT scale invariant image comparison
(tools : heat equation, differential invariants), a contrario methods

megawave demo - ASIFT - Mozilla Firefox

Fichier Édition Affichage Historique Marque-pages Outils ? Google

Page précédente Page suivante Actualiser Arrêter Accueil http://mw.cmla.ens-cachan.fr/meg Aperçu Imprimer

Débuter avec Firefox À la une Google News France - ...

> mw > megawave > demo

ASIFT DEMO

ASIFT

This program performs the affine scale-invariant matching method known as ASIFT. Full details, examples and code are available on [a dedicated page](#).

Please select two images; color images will be converted into gray level.

The first step, low-resolution, should take about 20 seconds, maybe less for simple images.

upload and start

input	image	#1
	<input type="button" value="Parcourir..."/>	
input	image	#2
	<input type="button" value="Parcourir..."/>	

uncheck if you do not want to include these images in the public archive.

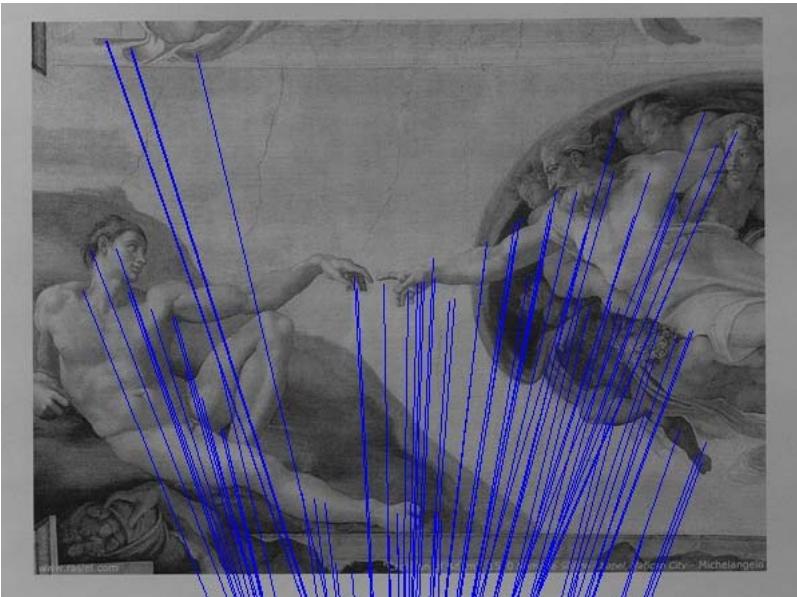
The maximum allowed file size is 1024Kb. TIFF, JPEG, PNG, PNM (and other) formats are recognized.
We may re-use the uploaded files for further analysis.

You can also try these proposed images.

select and start

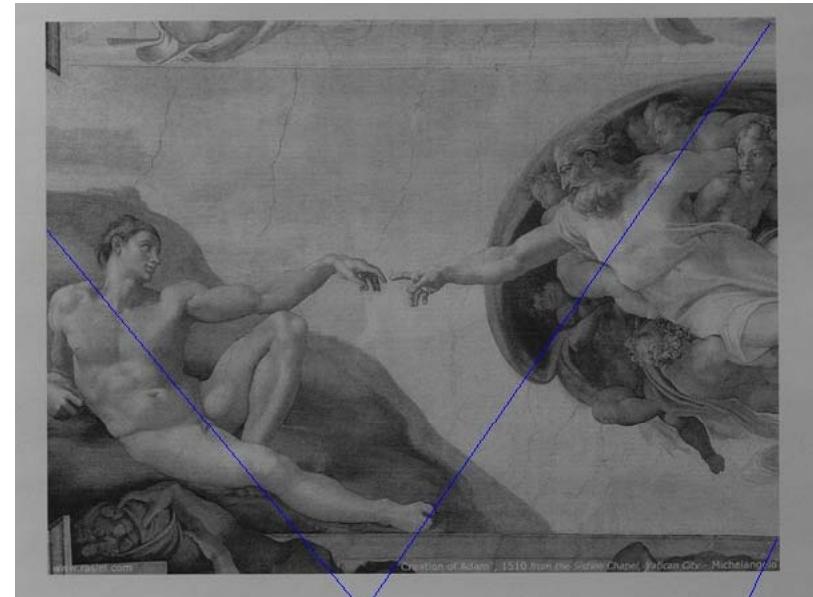
Terminé

Result of ASIFT:



<http://www.ipol.im>

Compared with SIFT:



Cachan, Heidelberg 2011

The page of each algorithm shows and explains the failure cases. For instance for SIFT and ASIFT, failure comparing objects with night and day illumination

Affine SIFT (ASIFT) - Mozilla Firefox

Fichier Édition Affichage Historique Marque-pages Outils ? G Google

Page précédente Page suivante Actualiser Arrêter Accueil http://mw.cmla.ens-cachan.fr/megawave/algo/asift/ Aperçu Imprimer

Débuter avec Firefox À la une Google News France - ...

Failure Cases

Day-and-night illumination change

All methods fail!

Six images of Notre-Dame under different illumination conditions are compared. The number of matches of ASIFT and SIFT are shown. (Harris-Affine, Hessian-Affine and MSER find less matches than SIFT.) Little view angle change is presented. The red arrows imply recognition failure.

In general, matching succeeds between day images and between night images. However, under day-and-night illumination change, all methods fail.

2. evening, with light 3 day, cloudy

1. day, sunny 4. day, sunny

ASIFT/SIFT

Image 1	Image 2	Image 3	Image 4	Image 5	Image 6
1. day, sunny	2. evening, with light	3 day, cloudy	4. day, sunny	5. night, with light	6. night, cloudy
142/45	5/0	119/51	0/0	131/48	0/5

Terminé

7

The online demo gives also access to the online archive. More than 8550 different images have been so far tried by on line users. They are grouped in pages of 20. Here are three examples tried by users, on a simple box, a building and a landscape.

megawave demo - asift archives - Mozilla Firefox

Fichier Édition Affichage Historique Marque-pages Outils ? G Google

Page précédente Page suivante Actualiser Arrêter Accueil http://mw.cmla.ens-cachan.fr/megawave/demo/asift/arch Aperçu Imprimer

Débuter avec Firefox À la une Google News France - ...

> mw > megawave > demo > asift

MEGAWAVE DEMO - ASIFT PUBLIC ARCHIVES

This archive is updated every hour. It is not moderated; in case of copyright infringement or similar problem, please contact us to request the removal some images.

pages : 50 100 150 200 250 300 350 400 450 500 550 600 650 700 750 800 850 900 950 1000 1050 1100 1150 1200 1250 1300 1350 1400 1450 1500 1550

- 2009-07-09 11:49:14 - 47d5c49e494d1c83bf5cc3fa76413a8e - 1.0
- 2009-07-07 16:23:46 - 032e34169a3eeaa7e40963a64c9311af - 1.0
- 2009-07-07 11:39:55 - abe00b893716d24ca5ac1fbb3659e57e - 1.0

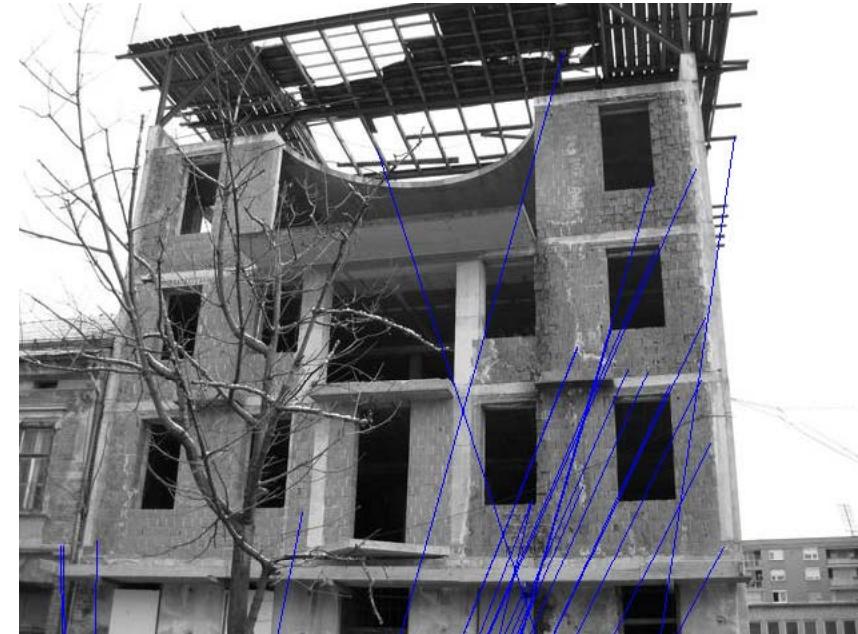
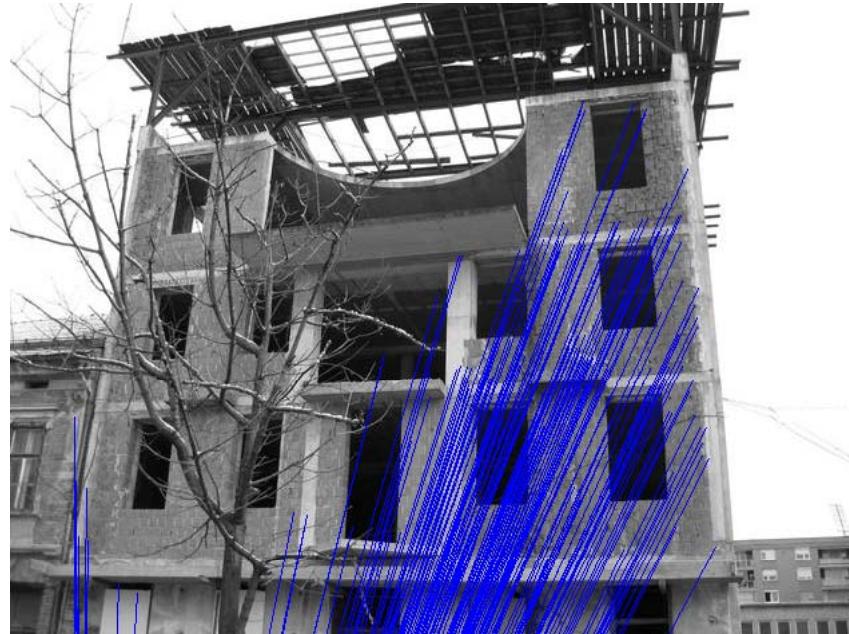
coords LR coords HR

http://mw.cmla.ens-cachan.fr/megawave/demo/asift/archive/47d5c49e494d1c83bf5cc3fa76413a8e/match_LR.jpeg

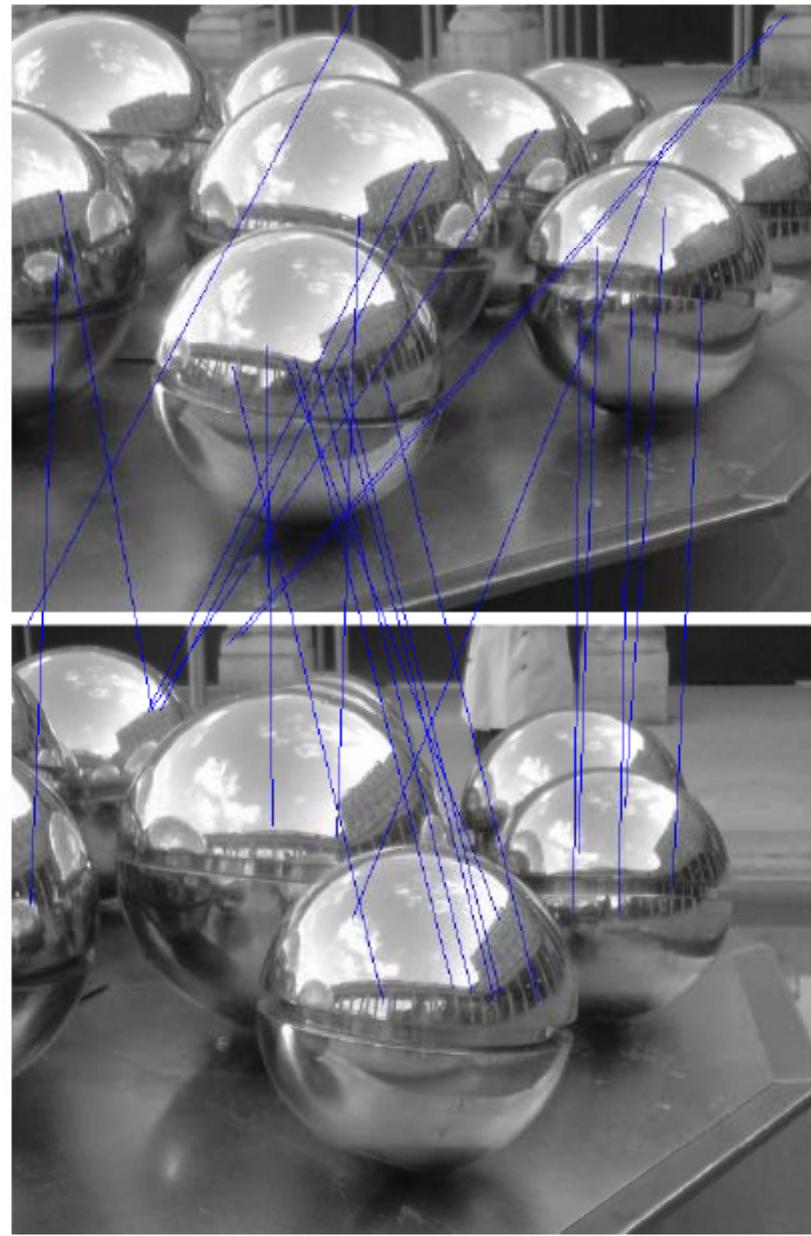
By a simple click a closer view of each experiment is available:

ASIFT result :

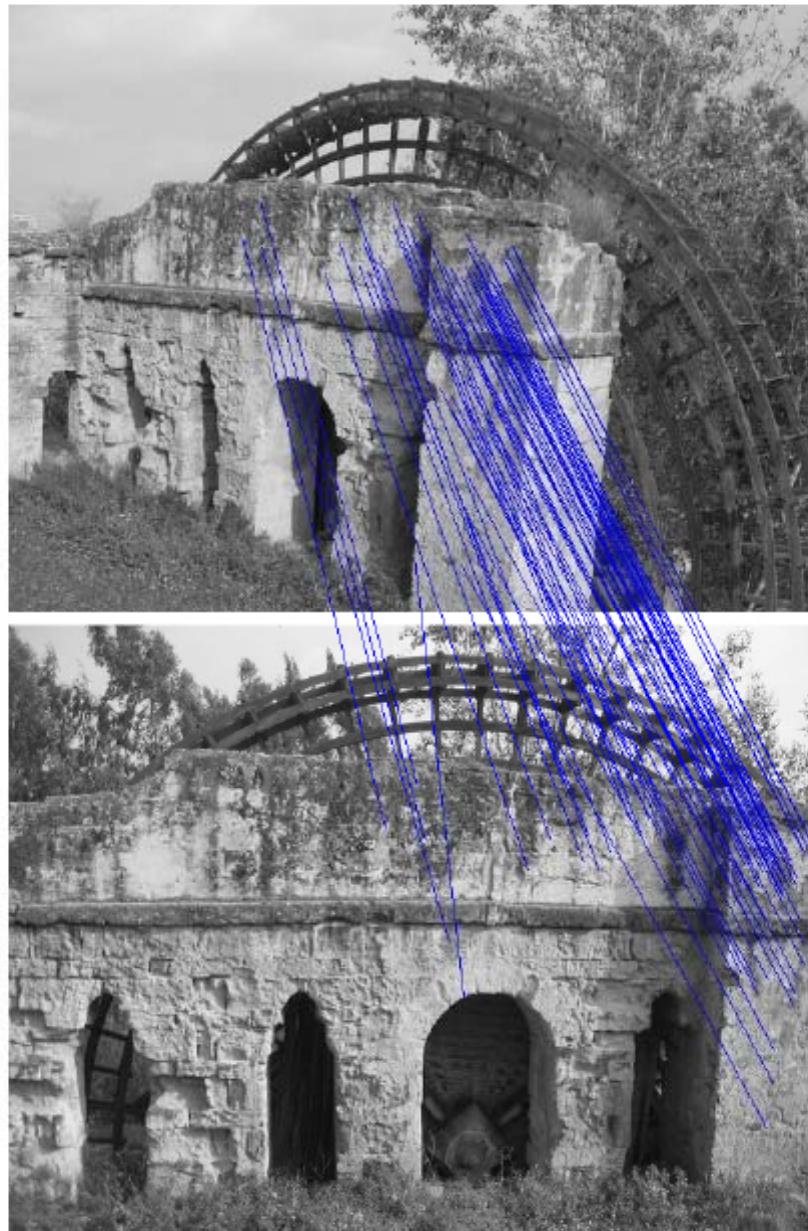
SIFT result :



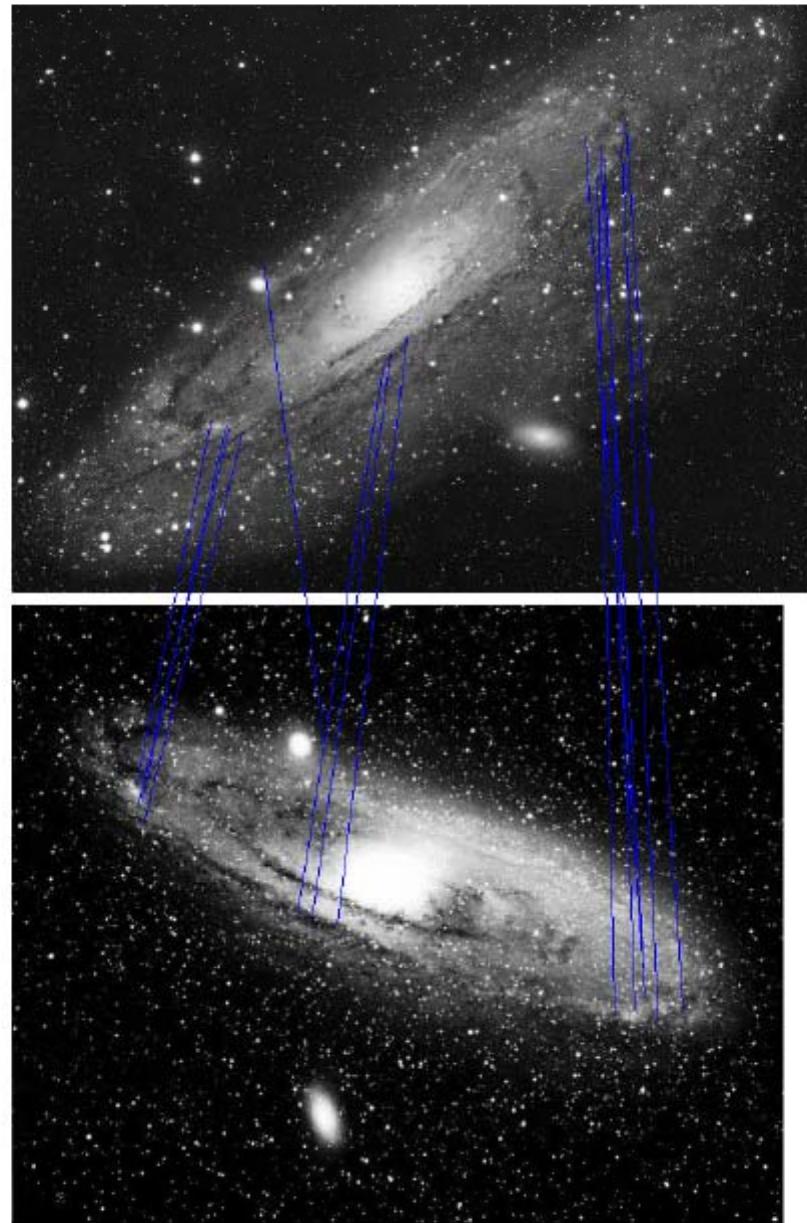
han, Heidelberg

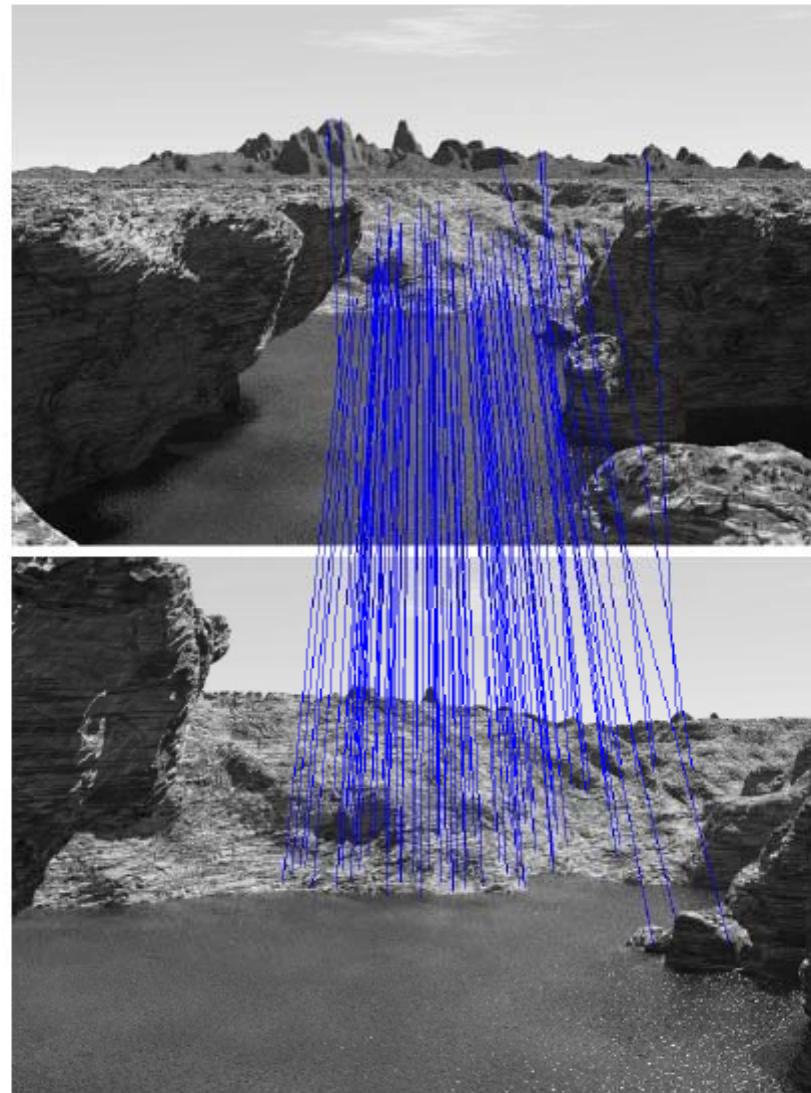


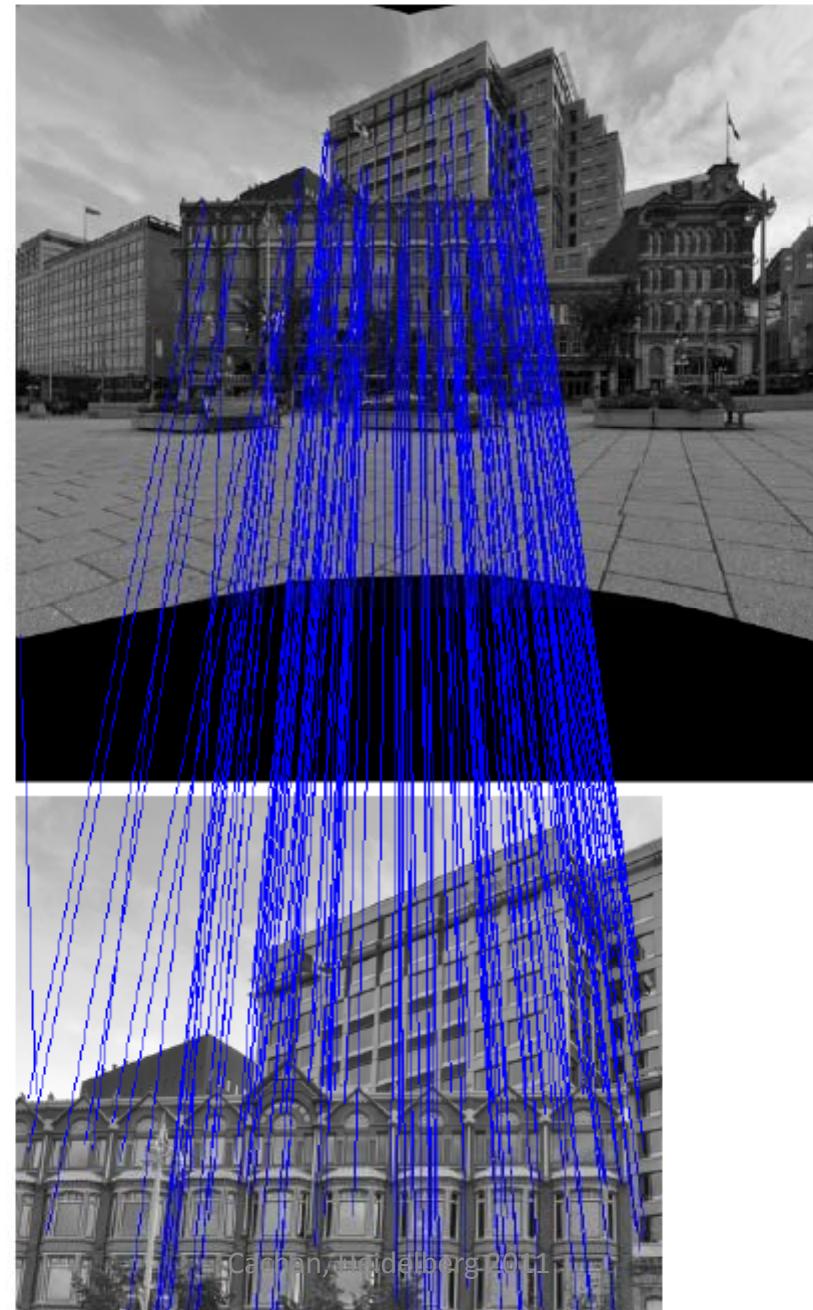


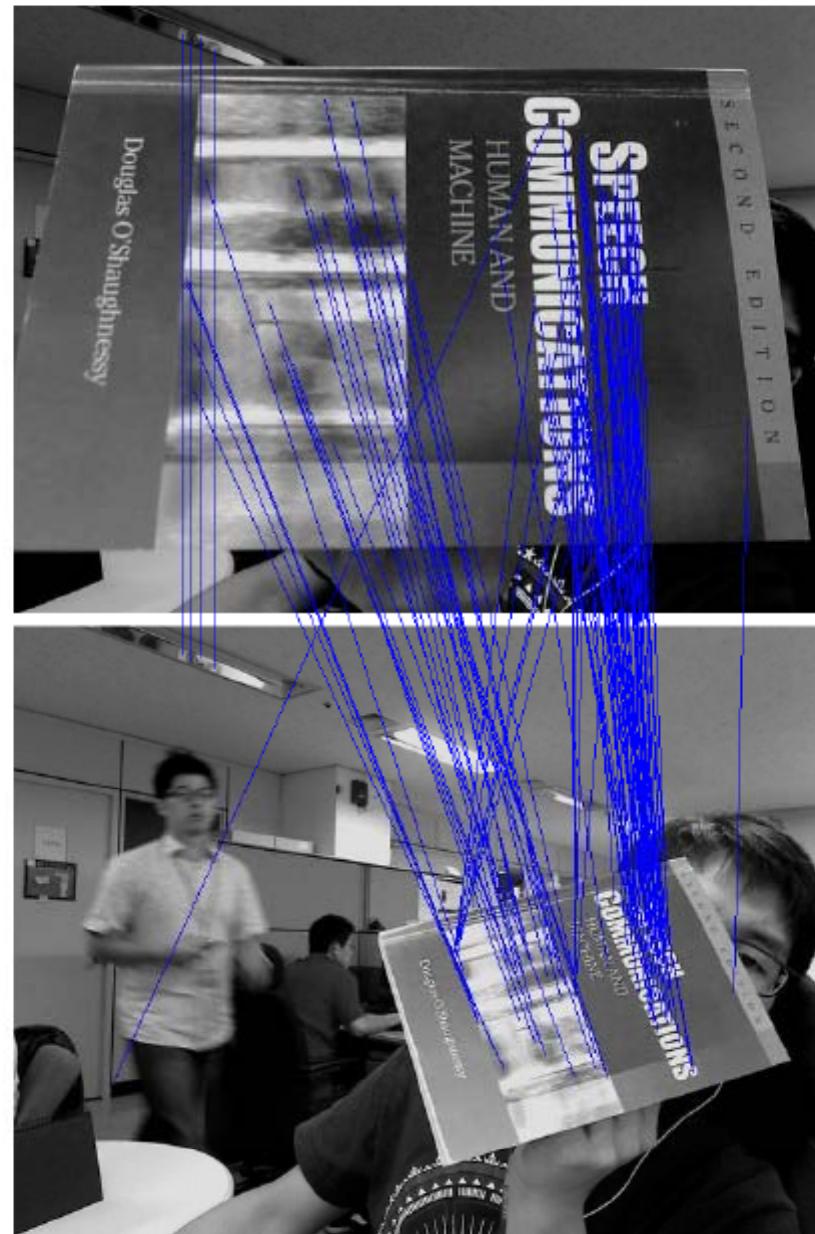


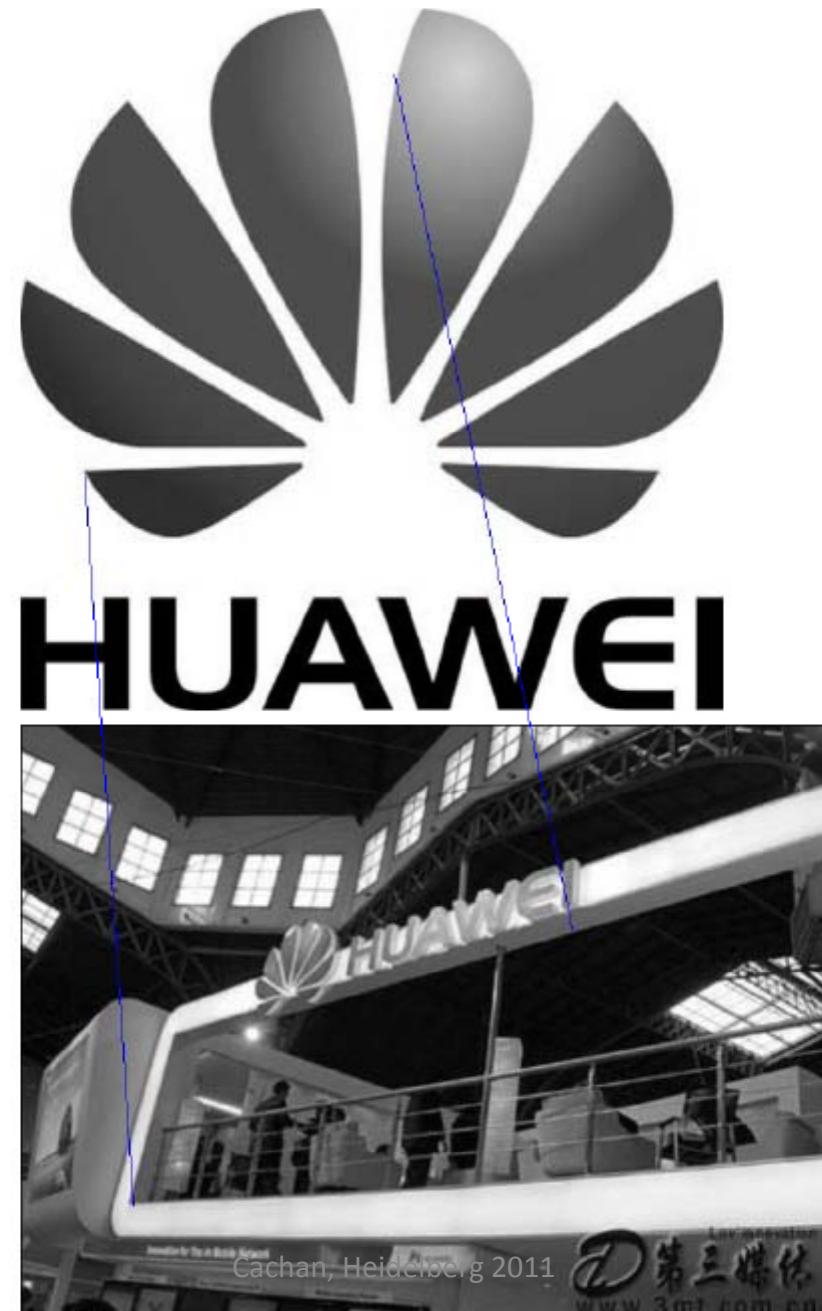


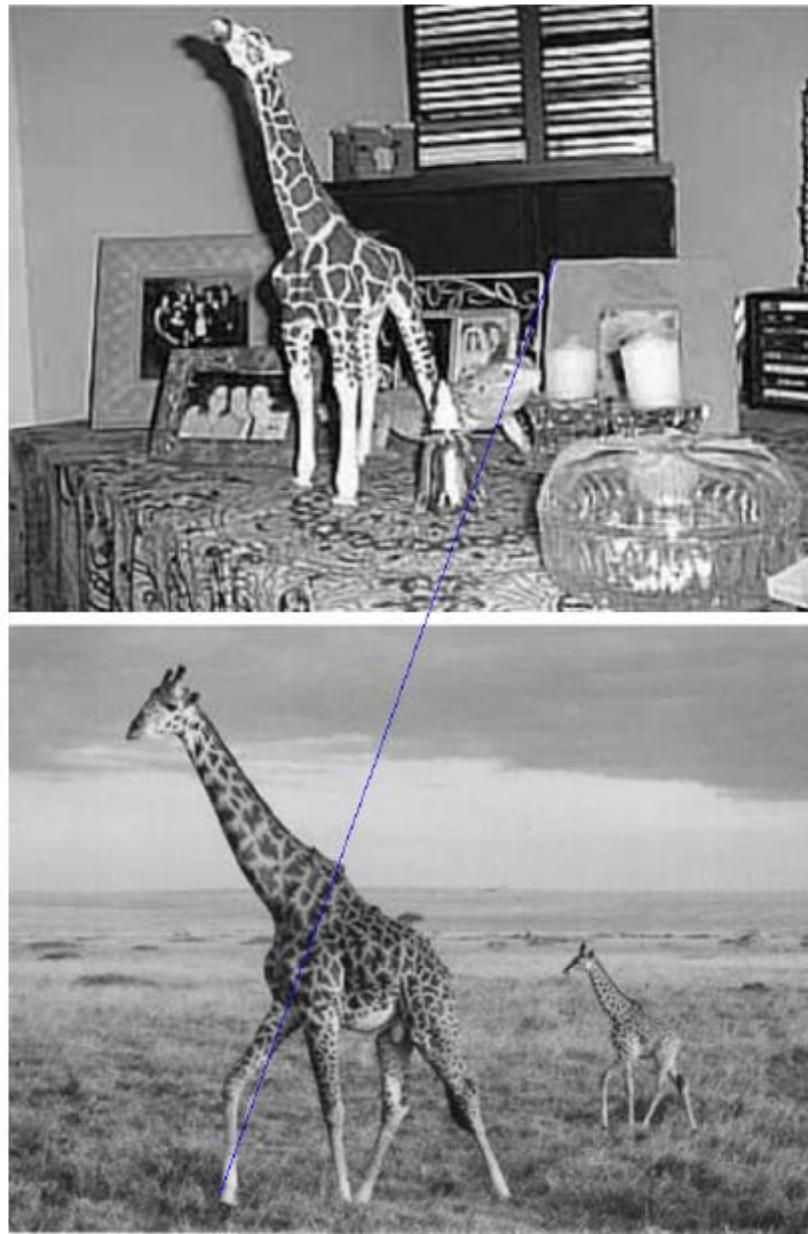
















Panorama (Caposele)



Pro Foto Caposele





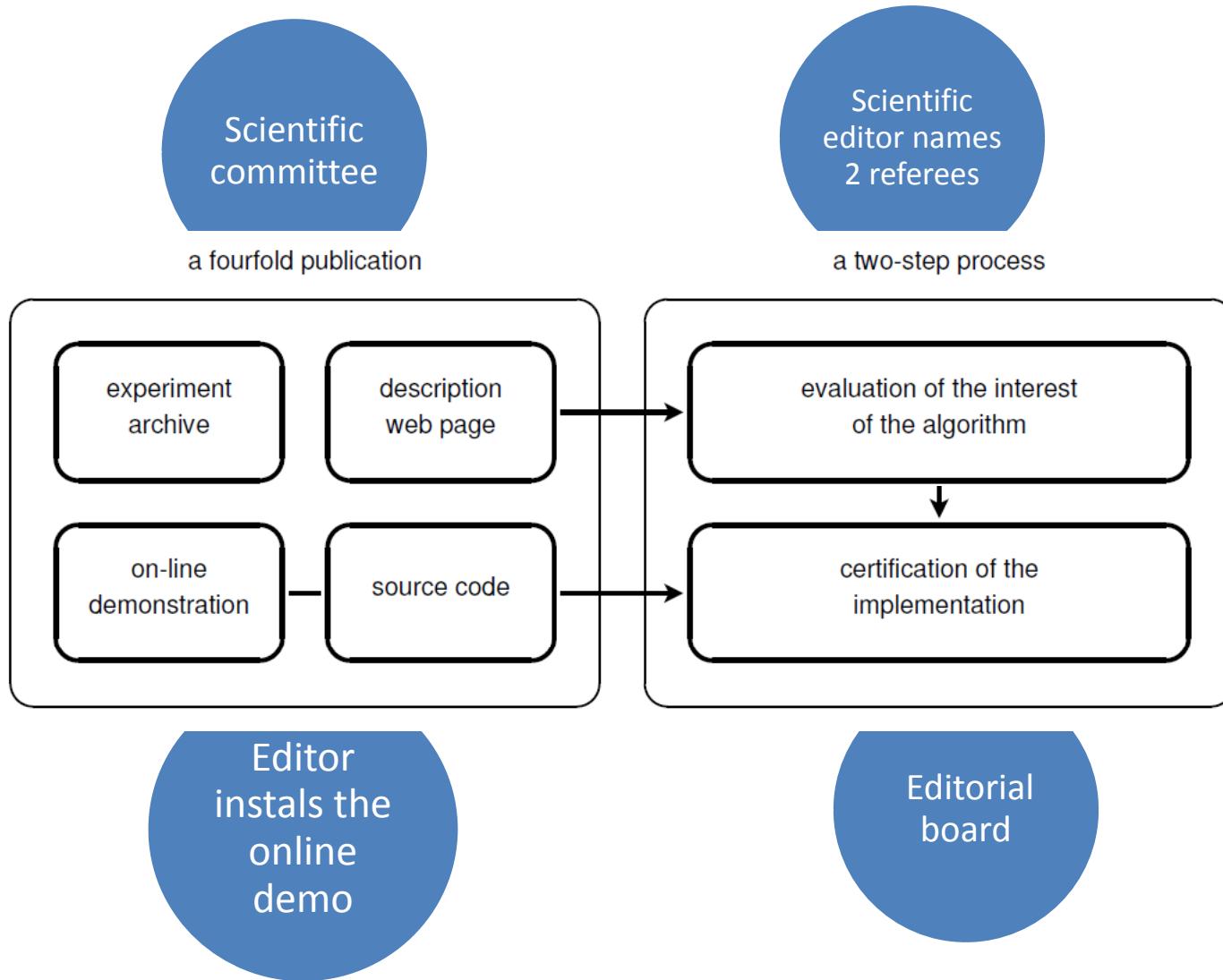
Online experiments: statistics on 14 articles

	Number of online experiments	Google Scholar
<i>SIFT + ASIFT :</i>	8295 experiments since 2009/04/27	52
<i>LSD Linear Segment detector:</i>	3966 experiments since 2009/04/27	13
<i>Retinex Poisson Equation:</i>	1189 experiments since 2009/04/27	3
<i>Micro-texture synthesis:</i>	1541 experiments since 2009/07/01	2
<i>Simplest color balance:</i>	851 experiments since 2009/08/28	0
<i>Cartoon + texture :</i>	873 experiments since 2009/09/07	1
<i>NL-means denoising :</i>	1800 experiments since 2009/11/28	473
<i>Image Curvature Microscope:</i>	513 experiments since 2010/01/15	2
<i>Lens distortion correction:</i>	700 experiments since 2010/05/06	5
<i>Self-similarity demosaicing:</i>	220 experiments since 2010/05/11	9
<i>Image interpolation with contour stencils :</i>	54 experiments since 2010/11/12	0
<i>Quasi-Euclidean Epipolar Rectification:</i>	39 experiments since 2010/11/30	17
<i>Image color cube dimensional filtering:</i>	36 experiments since 2010/12/07	0
<i>Finite difference schemes for MCM and AMSS:</i>	7 experiments since 2010/12/08	0

The archive contains more or less half of these experiments (the others are intentionally hidden by users)

Number of papers in preparation: 38

Publication submission and evaluation workflow



Means of the online journal

One online server, multicore (32), one test server for submissions (8 cores)
Editorial board (distinct from scientific committee): 15 experienced researchers with solid computer background.
Each paper signed by authors and an editor

Requirements on authors:

Execution in real/interactive time (less than 20 seconds)
Some (easy) parallel computing recommended (help provided by editors)
Writing in very standard code ANSI C, C++, free of any environment
A minimal number of libraries authorized (libtiff, libpng, fftw)
Additional workload on authors: two months (acceleration, code comments, demo design, testing)

A toolkit is provided to editors to program online demos.
Authors have deliver their algorithm along with a specification of its input, output.

Means of the online journal

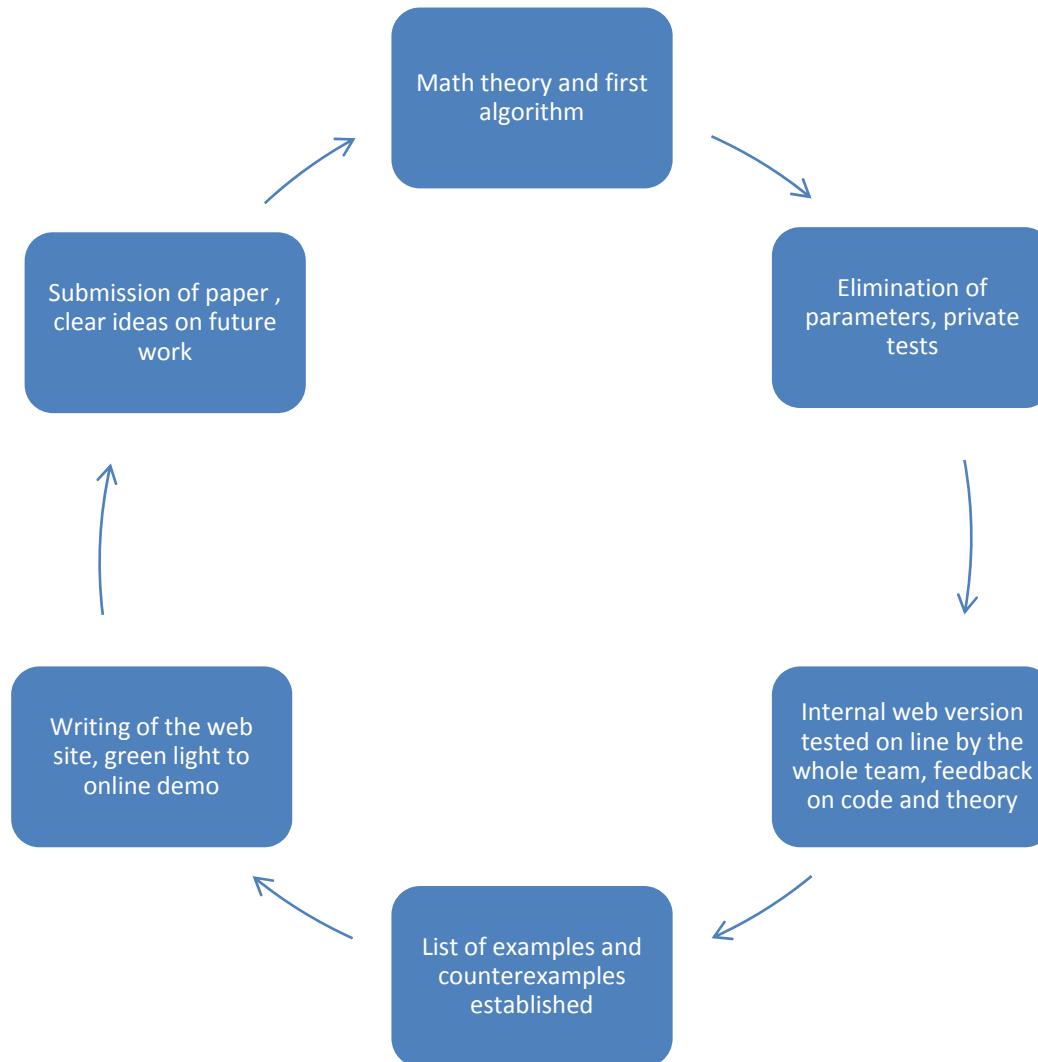
One online server, multicore (32), one test server for submissions (8 cores)
Editorial board (distinct from scientific committee): 15 experienced researchers with solid computer background.
Each paper signed by authors and an editor

Requirements on authors:

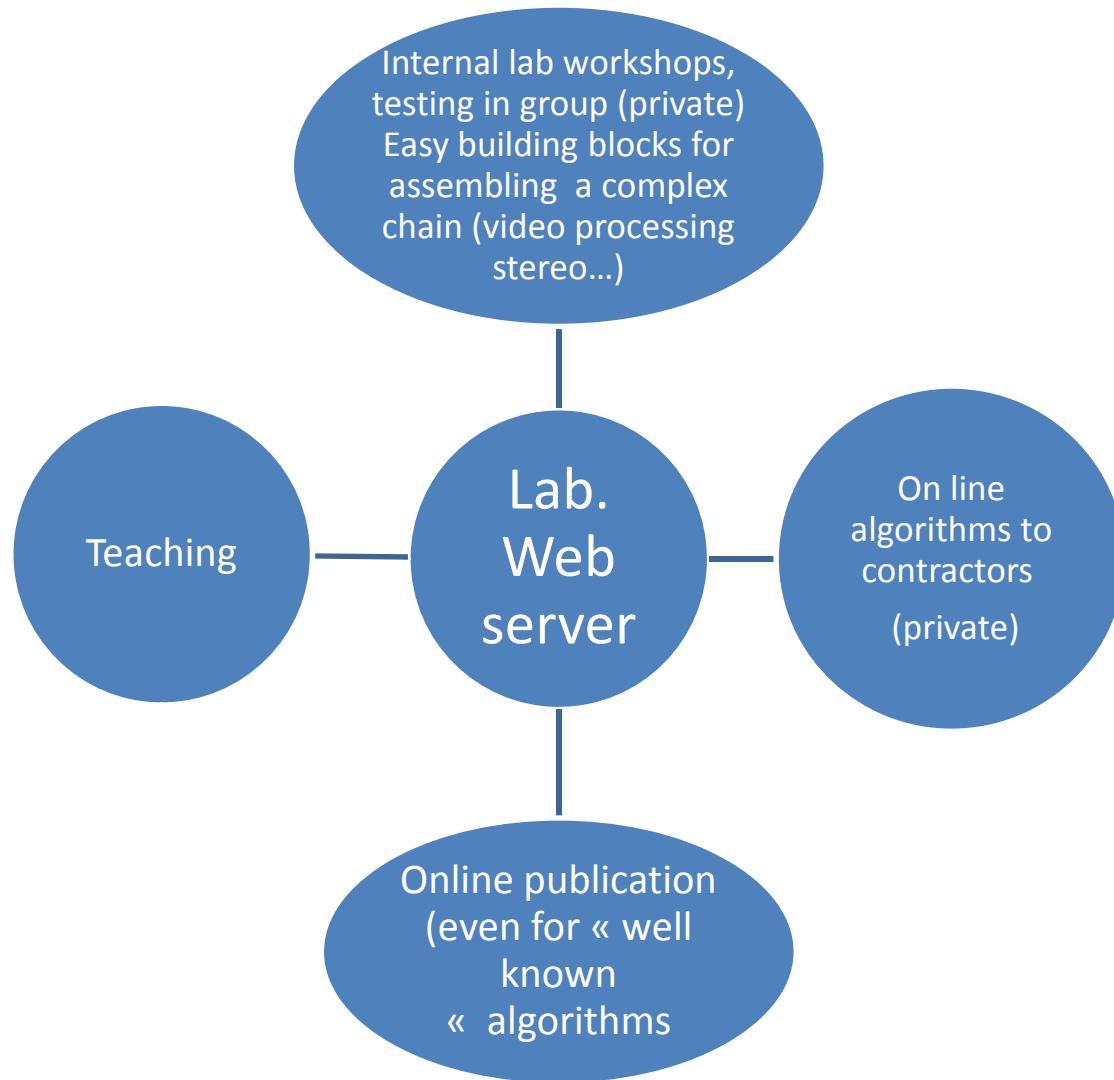
Execution in real/interactive time (less than 20 seconds)
Some (easy) parallel computing recommended (help provided by editors)
Writing in very standard code ANSI C, C++, free of any environment
A minimal number of libraries authorized (libtiff, libpng, fftw)
Additional workload on authors: two months (acceleration, code comments, demo design, testing)

A toolkit is provided to editors to program online demos.
Authors have deliver their algorithm along with a specification of its input, output.

Reorganization of the work flow in a research team



Four functions for a Web server



Plans and problems

Problem 1: the atom/molecule egg and hen problem

Submitted articles are atoms, but online demos are molecules because they must display some kind of end result

- . For example ASIFT uses SIFT and ORSA. Only the submitted atom is evaluated the
- . The rest of the routines must be at some point submitted, evaluated too:
(ORSA is currently submitted)
- . Stereo=calibration+ stereorectification + matching + reconstruction...
- . Image processing (noise, blur, color, contrast: requires reliable estimates)

Problem 2: Can a publication become two fold: journal paper + IPOL ?

-YES (1st agreement with SIAM J. of Image Science)

Problem 3: Already published algorithms can become research level IPOL articles?

-YES. An IPOL publication of a « know » algorithm + conception of the demo turns out to be substantial research

Problem 4: Licence GPL, others: If authors also want to sell licenses to industry, compatible?

-YES, you can see private licences along with a public licence

LSD: a Line Segment Detector - Mozilla Firefox

Fichier Édition Affichage Historique Marque-pages Outils ? | G Google

Page précédente Page suivante Actualiser Arrêter Accueil http://mw.cmla.ens-cachan.fr/megawave/algo/lsd/ Aperçu Imprimer

Débuter avec Firefox À la une Google News France - ...

> mw > megawave > algo

LSD: A LINE SEGMENT DETECTOR

Contacts

- Rafael Grompone von Gioi grompone@cmla.ens-cachan.fr
- Jérémie Jakubowicz jakubowi@cmla.ens-cachan.fr
- Jean-Michel Morel morel@cmla.ens-cachan.fr
- Gregory Randall randall@fing.edu.uy

References

1. R. Grompone von Gioi, J. Jakubowicz, J.-M. Morel, G. Randall, *LSD: A Fast Line Segment Detector with a False Detection Control*. IEEE Trans. on PAMI, 19 Dec. 2008. preprint

Overview

A linear-time line segment detector that gives accurate results, a controlled number of false detections, and requires no parameter tuning. The method is based in Burns et al. method, and uses an a contrario validation approach.

On-line Demo

An on-line demo that allows you to try LSD with your own images is available [here](#).

Software

Implementations in C programming language and Megawave2 framework are available [here](#).

Video

The video [here \(mp4 file, 56 MB\)](#) shows the result of applying LSD, frame per frame, to the original video [here](#).

Examples

Terminé

Line segment detector (LSD), no parameter. More than 3500 images on line trials in the archive. Tools: A contrario statistical method, level line analysis

megawave demo - lsd archives - Mozilla Firefox

Fichier Édition Affichage Historique Marque-pages Outils ? Google

Page précédente Page suivante Actualiser Arrêter Accueil http://mw.cmla.ens-cachan.fr/megawave/demo/lst/archiv Aperçu Imprimer

Débuter avec Firefox À la une Google News France - ...

> mw > megawave > demo > lsd

MEGAWAVE DEMO - LSD PUBLIC ARCHIVES

This archive is updated every hour. It is not moderated; in case of copyright infringement or similar problem, please [contact us](#) to request the removal of some images.

pages : 50 100 150 200 250 300 350 400 450 500 550 600 650 700 750 800 850

- ◆ 2009-09-30 11:34:20 - 95be8a74889a75a6ab137b38c581c8f7 - 1.0
- ◆ 2009-09-30 11:33:47 - 59c3d5663a82b971d0ba8bcc00fdbcc6 - 1.0
- ◆ 2009-09-30 11:10:44 - f73d5a6fb7b9c94d22cad8ca01590867 - 1.0

coords

Terminé

32

Examples





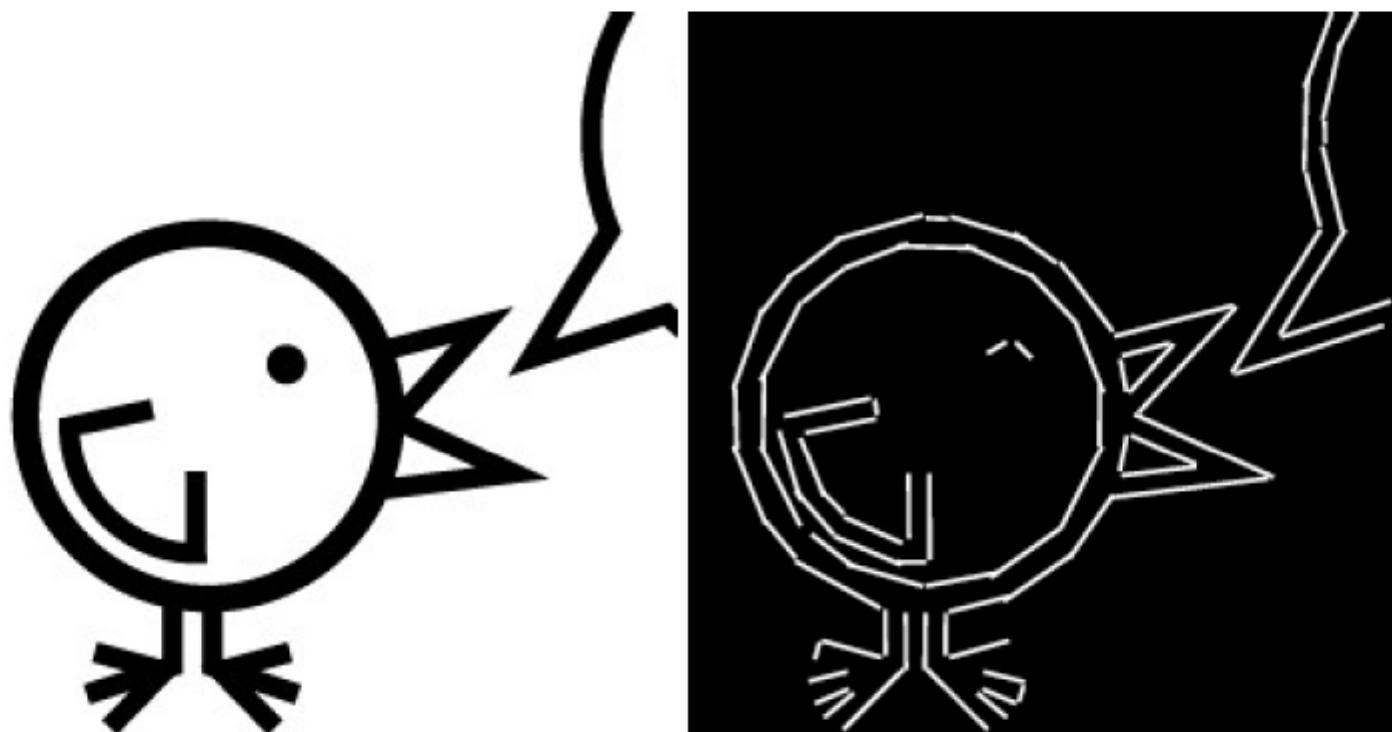
LSD: Some results in
the online archive



rg 2011

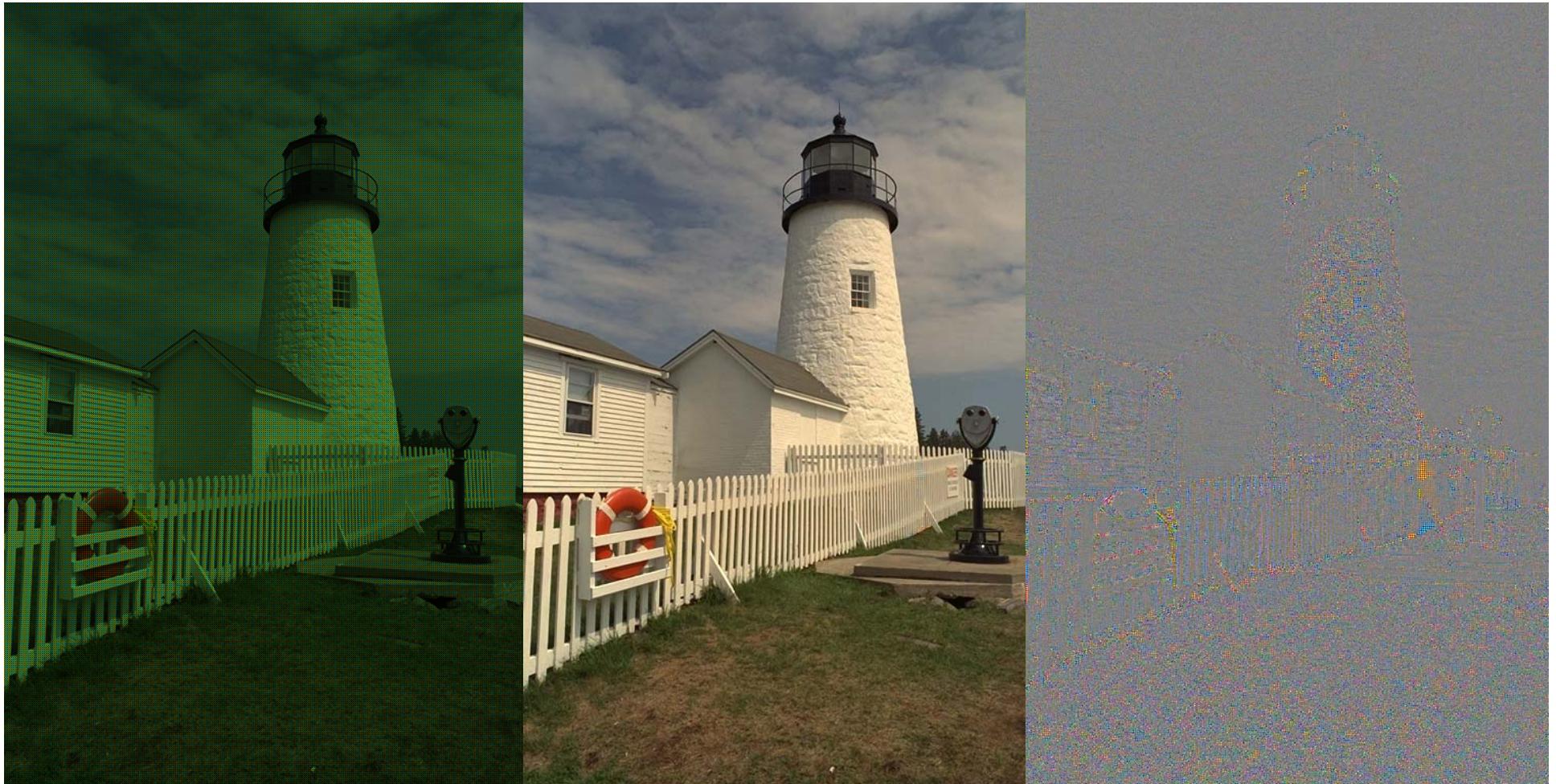
34

Examples

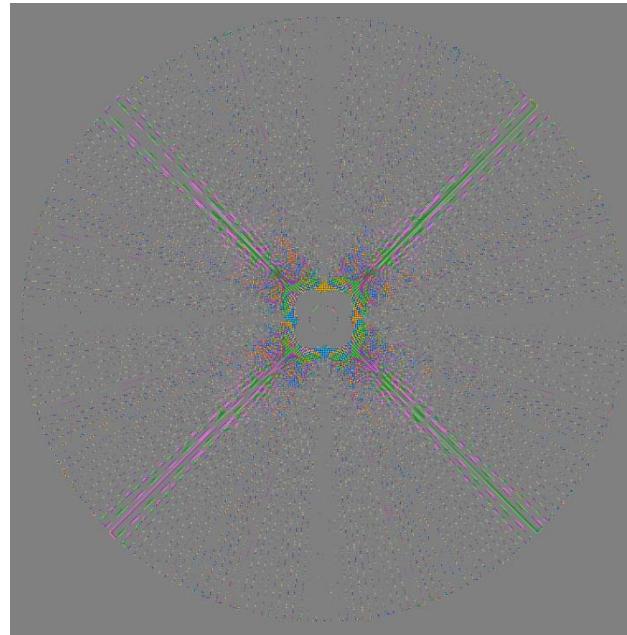
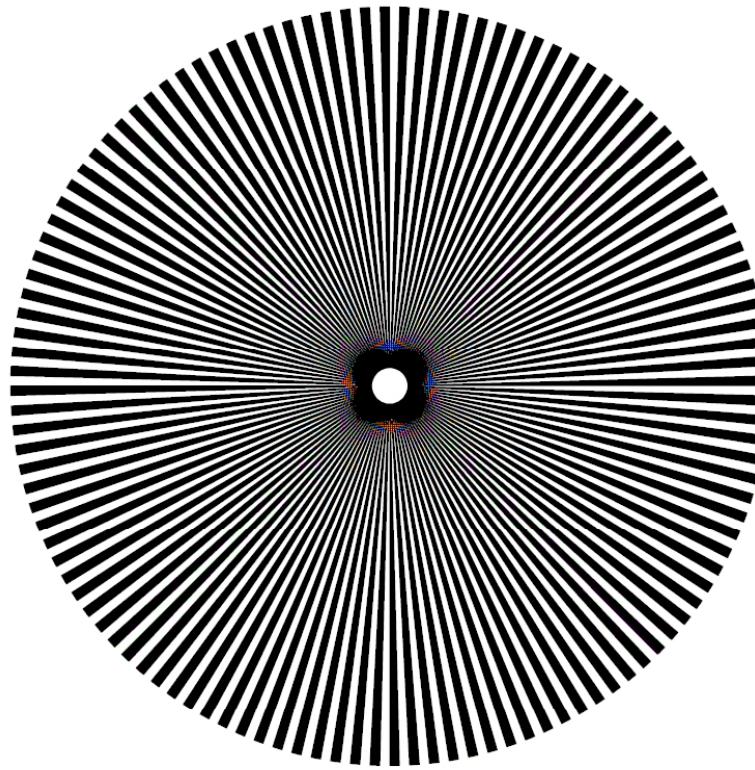
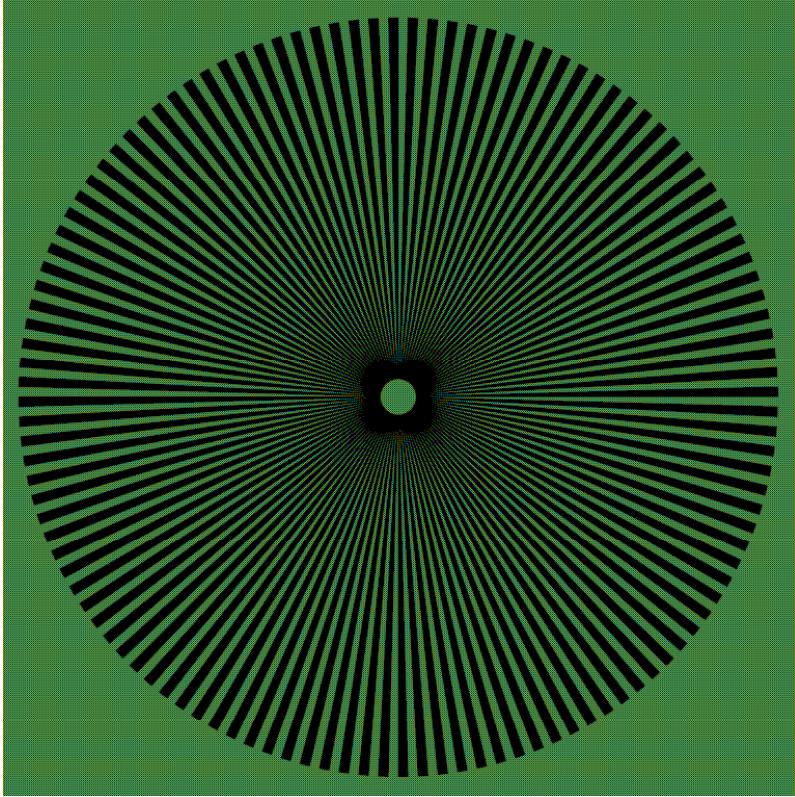


NL-means, archive on line. Tools: non linear partial differential equation, non local filters. Noise is automatically estimated.



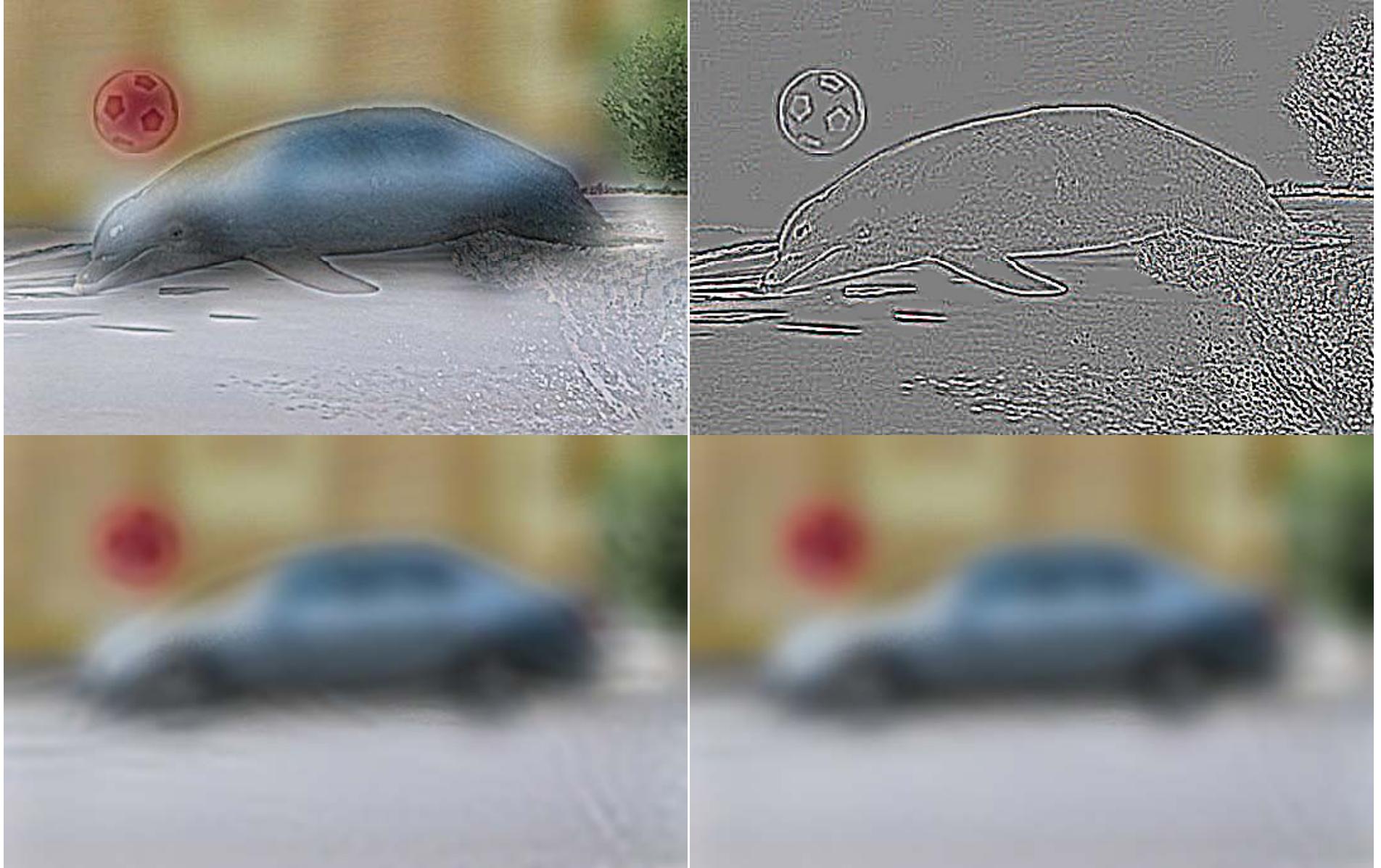


Demosaicking (zero parameter method) Tools: nonlocal methods, Fourier analysis, partial differential operators. From Archive on line

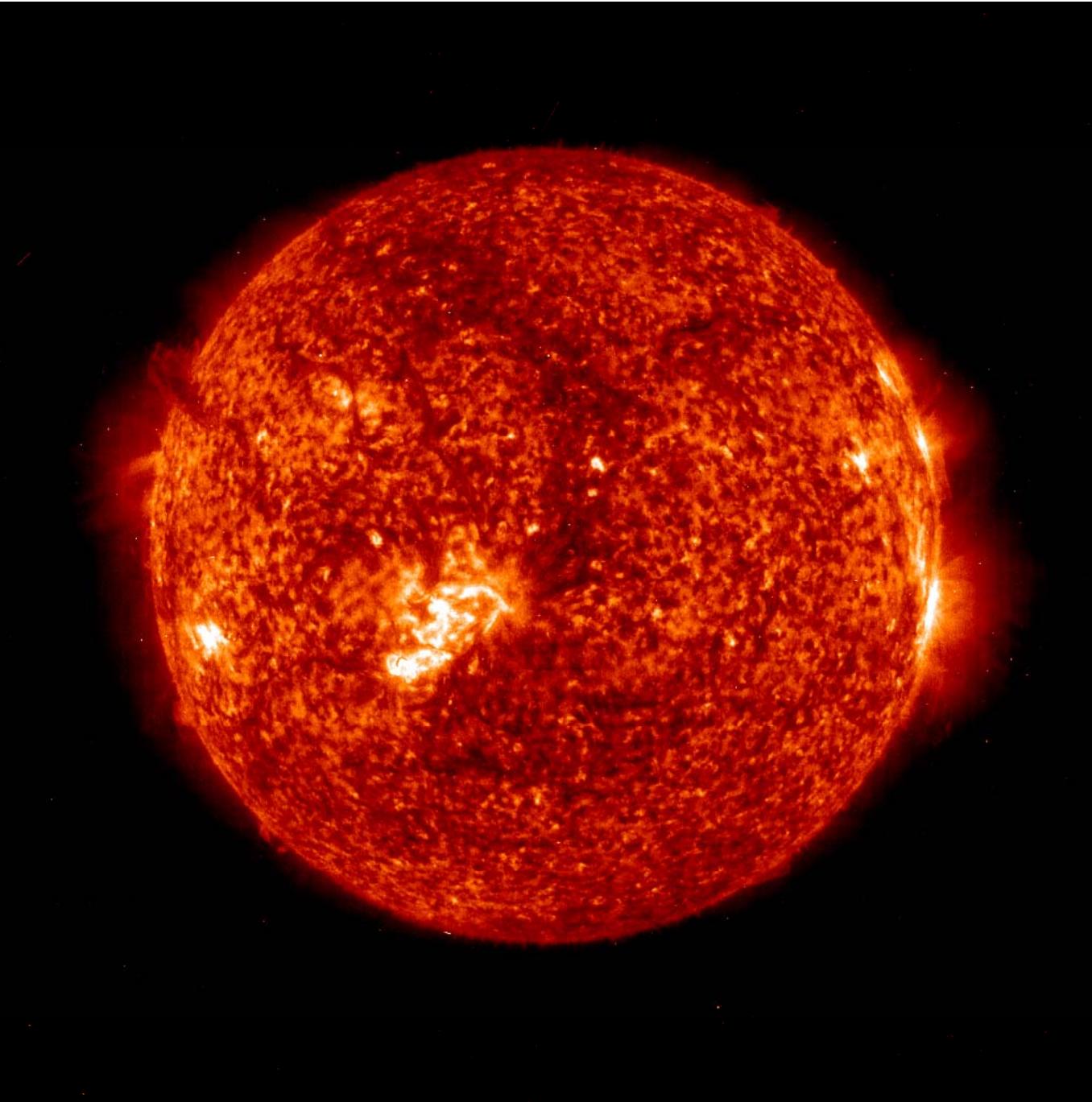


<http://www.ipol.im>

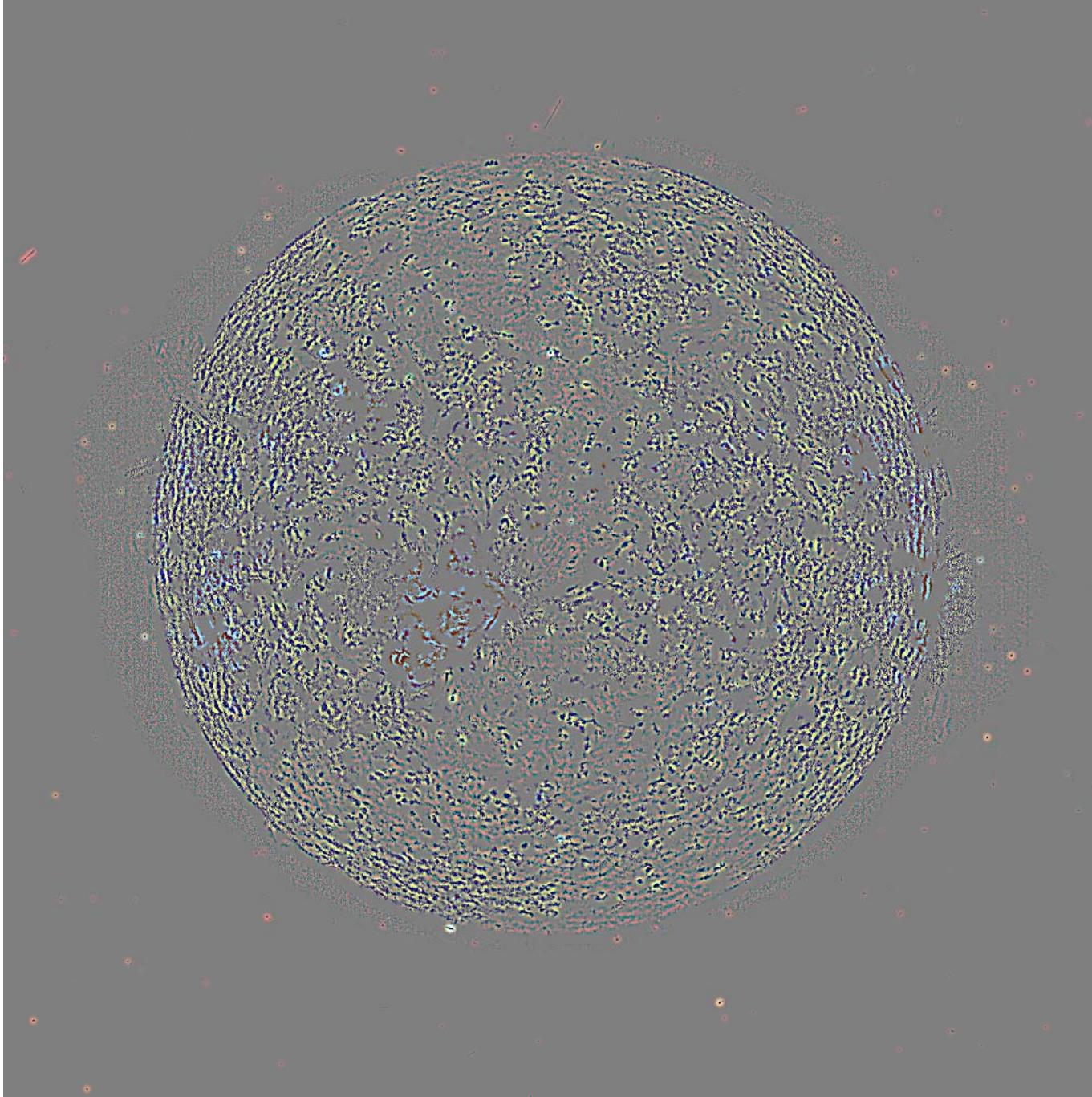
LSD: Some results in
the online archive



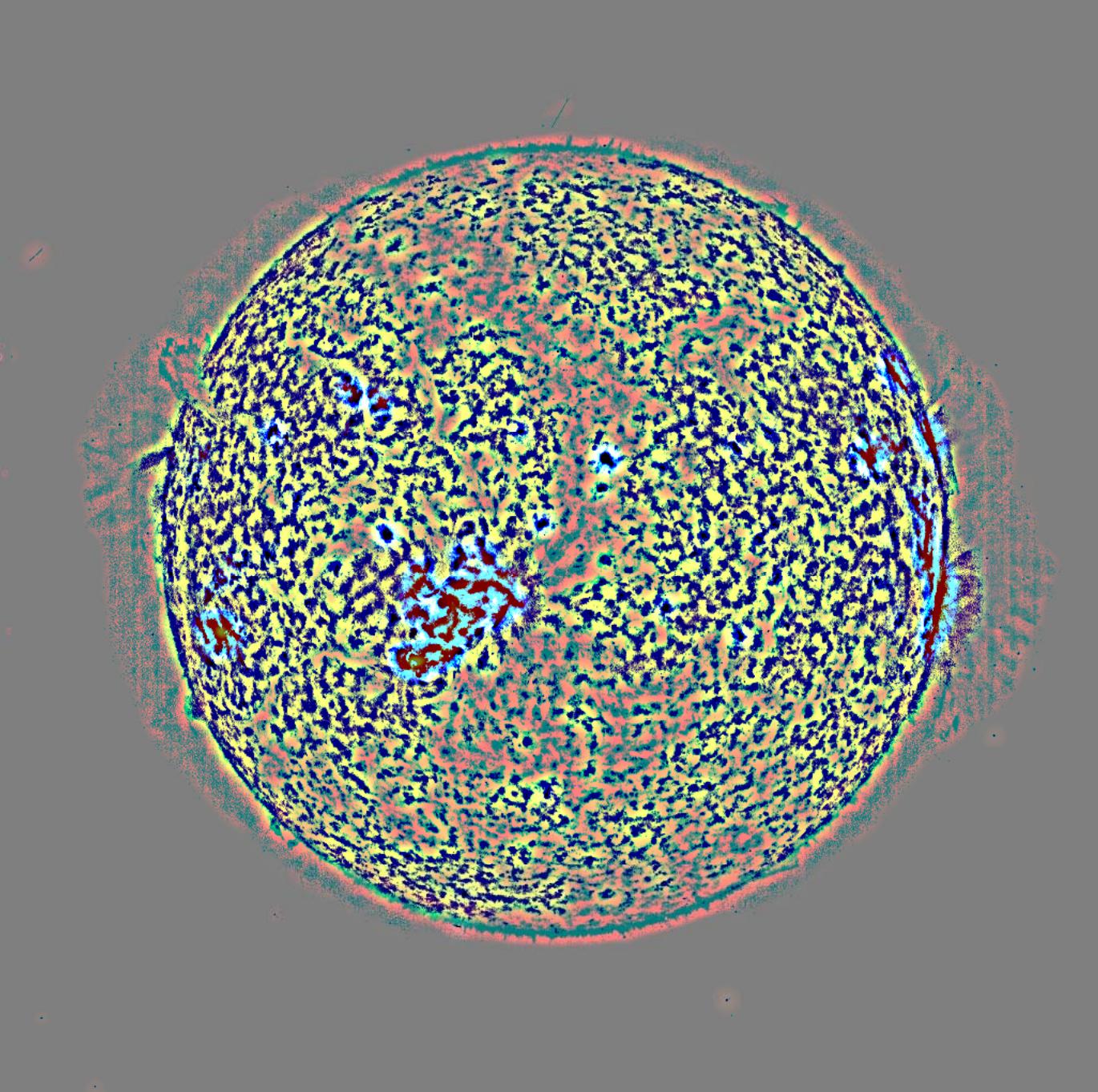
Cartoon+texture (IPOL Archive): method invented by Yves Meyer to decompose any image. One scale parameter. Tools : Fourier + functions with bounded variation



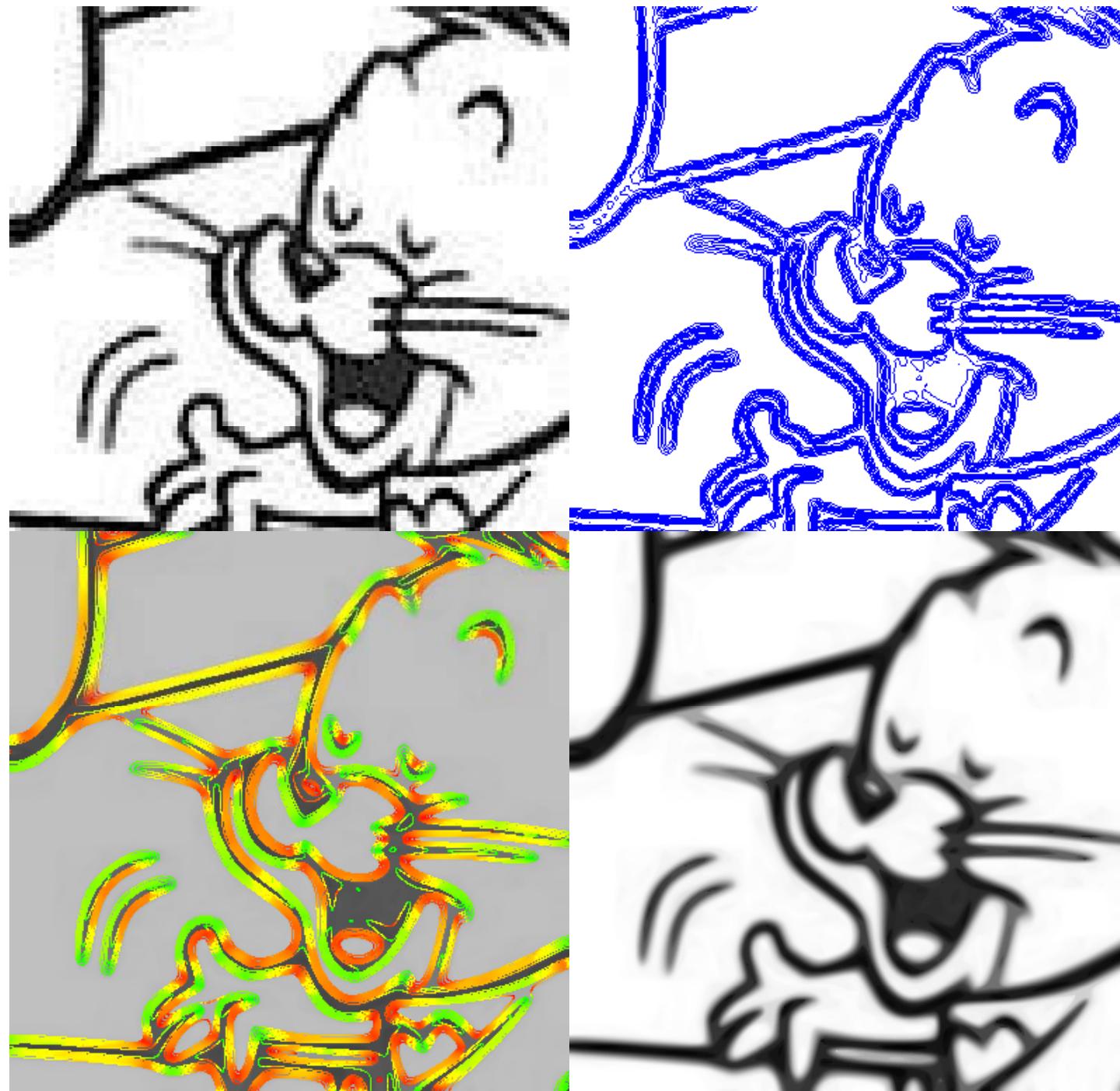
Cartoon+texture (IPOL
Archive)



Cartoon+texture (IPOL
Archive)



Cartoon+texture (IPOL
Archive)



Mean curvature motion (IPOL Archive).
Tools: Morse theory,
geometric partial differential equations

 Micro-Texture Synthesis by Phase Randomization - Mozilla Firefox

Fichier Édition Affichage Historique Marque-pages Outils ? G Google

Page précédente Page suivante Actualiser Arrêter Accueil http://mw.cmla.ens-cachan.fr/megawave/algo/random_phase_noise/ Aperçu Imprimer

Débuter avec Firefox À la une Google News France - ...

> mw > megawave > algo

MICRO-TEXTURE SYNTHESIS BY PHASE RANDOMIZATION

Contacts

- Bruno Galerne galerne@cmla.ens-cachan.fr
- Yann Gousseau gousseau@tsi.enst.fr
- Jean-Michel Morel morel@cmla.ens-cachan.fr

References

1. B. Galerne, Y. Gousseau and J.-M. Morel, *Random Phase Textures: Theory and Synthesis*, preprint CMLA N°2009-24, 2009.
Abstract [and pdf file](#) on CMLA preprint webpage.

Overview

The *Random Phase Noise (RPN)* algorithm synthesizes a texture from any original image by randomizing its Fourier phase. The *RPN* algorithm is able to reproduce the textures which are characterized by their Fourier modulus, namely the phase invariant textures.

The presented algorithm deals with color images and it is able to synthesize output textures having a larger size than the input samples.

Even though this texture synthesis algorithm only reproduce a limited class of textures, it has several good properties:

- It produces a micro-texture given any input image, and thus can be used to produce micro-texture versions of some macro-textures, or can also be used to design textures.
- The algorithm is perceptually stable: all the textures synthesized from the same input image look similar.
- The algorithm is fast.

On Line Demo: Try It!

An [on-line demo](#) of this algorithm is available.

The demo permits to upload a color texture sample and to replicate it in arbitrary size. Texture samples can be taken from existing databases, but to have still more realistic samples, you can extract them as homogeneous regions of a photograph, as shown below in [What are micro-textures?](#)

Données transférées depuis mw.cmla.ens-cachan.fr



Contacts
References
Overview
On Line Demo: Try It!
Source Code
Algorithm
Implementation
Micro-textures
Examples

Micro-Texture Synthesis by Phase Randomization - Mozilla Firefox

Fichier Édition Affichage Historique Marque-pages Outils ? G Google

Page précédente Page suivante Actualiser Arrêter Accueil http://mw.cmla.ens-cachan.fr/megawave/algo/random_phase_noise/#ii Aperçu Imprimer

Débuter avec Firefox À la une Google News France - ...

Examples

Below are some examples of satisfyingly well reproduced textures.

Original image



stone

RPN



stone simulated

Original image



wood

RPN



wood simulated

Terminé

http://mw.cmla.ens-cachan.fr/megawave/algo/random_phase_noise/#ii

Cachan, Freudentberg 2011

Micro-Texture Synthesis by Phase Randomization - Mozilla Firefox

Fichier Édition Affichage Historique Marque-pages Outils ? | G Google

Page précédente Page suivante Actualiser Arrêter Accueil | http://mw.cmla.ens-cachan.fr/megawave/algo/random_phase_noise/#i | Aperçu Imprimer

Débuter avec Firefox À la une Google News France - ...

Wood

Wood samples must be homogeneous in direction to be correctly emulated by RPN. Wood samples with knots or other conspicuous patterns fall logically in the [failure catalog](#).

Wood sample



RPN simulation



Show/hide **more wood examples**

Fabric

These fabric samples were picked from several web sites. Only homogeneous fabrics, with no printed on patterns are treated. RPN turns

Terminé

 Micro-Texture Synthesis by Phase Randomization - Mozilla Firefox

Fichier Édition Affichage Historique Marque-pages Outils ?  Google

Page précédente Page suivante Actualiser Arrêter Accueil   Aperçu Imprimer

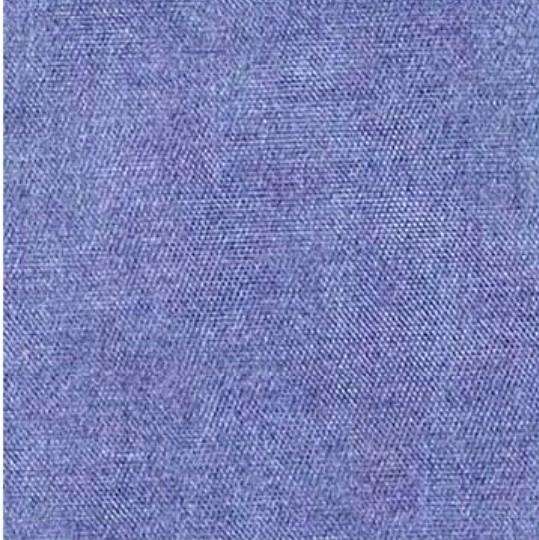
Débuter avec Firefox  À la une  Google News France - ...

These fabric samples were picked from several web sites. Only homogeneous fabrics, with no printed on patterns are treated. RPN turns out to work remarkably on jeans fabrics.

Fabric sample



RPN simulation



Show/hide [more fabric examples](#)

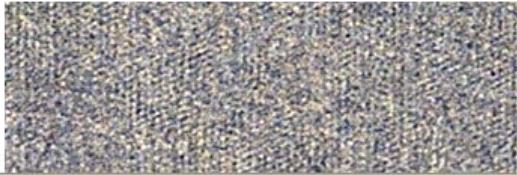
carpet

These carpet samples are taken from a single commercial website. Those with big patterns will be found in the [failure catalog](#).

Carpet sample



RPN simulation



Terminé 

Micro-Texture Synthesis by Phase Randomization - Mozilla Firefox

Fichier Édition Affichage Historique Marque-pages Outils ? G Google

Page précédente Page suivante Actualiser Arrêter Accueil http://mw.cmla.ens-cachan.fr/megawave/algo/random_phase_noise/#ir Aperçu Imprimer

Débuter avec Firefox À la une Google News France - ...

failure catalog

Most failures are **macro**-textures. For instance:

- textures containing periodic geometric patterns with large period,
- textures containing strong edges, such as veins in marble or cracks in bark
- textures containing definite shapes, such as knots in wood or fruit or visible leaves in foliage
- strictly periodic patterns, even with small period, where phase shifts cause aliasing effects
- failure also occurs when the sample texture contains different dominant directions in different areas. Then these directions are mixed by the random sampler.

Macro-texture sample



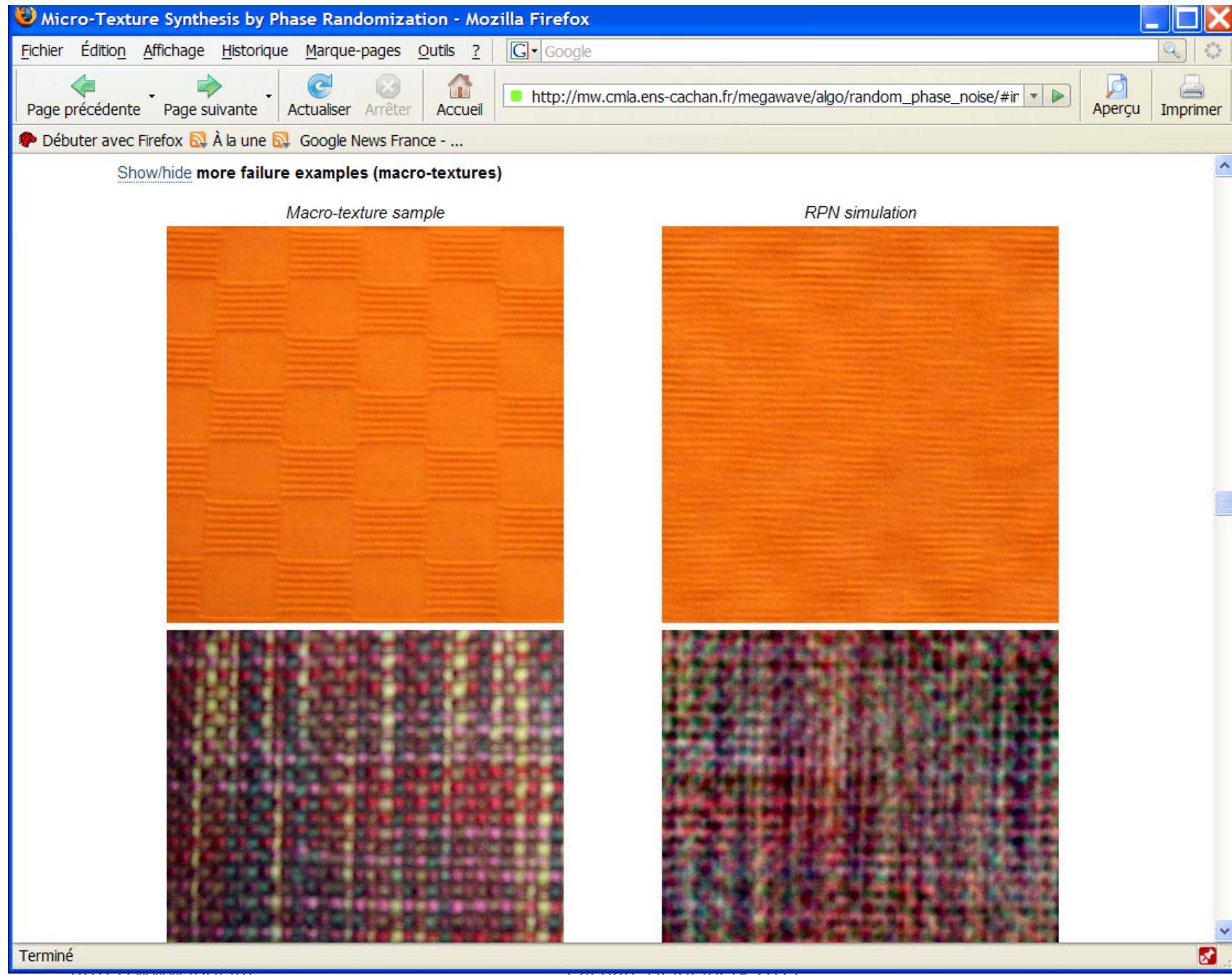
RPN simulation



Show/hide [more failure examples \(macro-textures\)](#)

edit recent changes history preferences

Terminé



The Retinex PDE : a model for color perception - Mozilla Firefox

Fichier Édition Affichage Historique Marque-pages Outils ? G Google

Page précédente Page suivante Actualiser Arrêter Accueil Aperçu Imprimer

Débuter avec Firefox À la une Google News France - ...

The Retinex PDE : a model... Erreur de chargement de la p...

> mw > megawave > algo

THE RETINEX PDE : A MODEL FOR COLOR PERCEPTION

Contacts

- Jean-Michel Morel morel@cmila.ens-cachan.fr
- Ana Belén Petro anabelen.petro@uib.es
- Catalina Sbert catalina.sbert@uib.es

References

1. Jean-Michel Morel, Ana Belén Petro and Catalina Sbert, *PDE Formalization of the Retinex Theory*. Submitted to IEEE Trans. on Image Processing.
2. Jean-Michel Morel, Ana Belén Petro and Catalina Sbert, *Fast Implementation of color constancy algorithms*. *Color Imaging XIV: Displaying, Processing, Hardcopy and Application*. Proc. of Electronic Imaging SPIE, vol 7241. January 2009.
preprint [on line article](#)

Overview

In 1964 Edwin H. Land formulated the Retinex theory, the first attempt to simulate and explain how the human visual system perceives color. His theory and an extension, the "reset Retinex" were further formalized by Land and McCann. Several Retinex algorithms have been developed ever since. These color constancy algorithms modify the RGB values at each pixel to give an estimate of the physical color independent of the shading.

Unfortunately, the Retinex original algorithm is both complex and not fully specified. Indeed, this algorithm computes at each pixel an average of a very large and unspecified set of paths on the image. For this reason, Retinex has received several interpretations and implementations which, among other aims, attempt to tune down its excessive complexity.

But, as shown in the references below, Retinex solutions satisfy a discrete linear partial differential equation in the Poisson form. This yields an exact and fast implementation of the Land-McCann theory using only two FFT's. Test the theory [on line on your own color images!](#)

Terminé

The Retinex PDE : a model for color perception - Mozilla Firefox

Fichier Édition Affichage Historique Marque-pages Outils ? G Google

Page précédente Page suivante Actualiser Arrêter Accueil http://mw.cmla.ens-cachan.fr/megawave/algo/retinex_pde Aperçu Imprimer

Débuter avec Firefox À la une Google News France - ...

The Retinex PDE : a model... Erreur de chargement de la p...

The PDE-Retinex Model

In the IEEE article it is proven that the output of the retinex algorithm proposed by Land and McCann is the solution of the discrete partial differential equation with Neumann boundary conditions

$$-\Delta_d u(i,j) + \frac{4}{\dim-1} u(i,j) = F(i,j)$$

where

$$-\Delta_d u(i,j) = u(i+1,j) + u(i-1,j) + u(i,j+1) + u(i,j-1) - 4u(i,j)$$

is the discrete Laplacian, $\dim = N M$ is the size of the image,

$$F(i,j) = f(I(i,j)-I(i+1,j))+f(I(i,j)-I(i-1,j))+f(I(i,j)-I(i,j+1))+f(I(i,j)-I(i,j-1))$$

and $f(x)$ is a threshold function, whose value is zero if $|x| < t$ and the identity in other case and I is the image to process. This function f eliminates the small variations of the intensity image I .

The parameter t (the threshold) is by default $t = 3$ but you can choose the value depending of the variations you want to eliminate.

The Algorithm

The output of the algorithm are two images: the first one is the white balance of the original color image (on each channel the darkest pixels are put to zero and the brightest ones are put to 255); the second image is the result of the Retinex PDE applied to the white balanced image.

The discrete partial differential equation is easily solved by fast Fourier transform. Applying the Fourier transform to the discrete partial differential equation yields

$$\hat{u}(k,l) \cdot \left(4 + \frac{4}{\dim-1} - 2 \cos\left(\frac{2k\pi}{N}\right) - 2 \cos\left(\frac{2l\pi}{M}\right) \right) = \hat{F}(k,l)$$

The algorithm is

1. Compute $F(i,j)$;
2. Compute Fourier transform of F by FFT;
3. Deduce the Fourier transform of u using the formula above;
4. Compute the final solution u using the inverse FFT.

Terminé



Fichier Édition Affichage Historique Marque-pages Outils ?

G Google

Page précédente Page suivante Actualiser Arrêter Accueil

http://mw.cmla.ens-cachan.fr/megawave/demo/retinex_pde/archive/

Aperçu Imprimer

Débuter avec Firefox À la une Google News France - ...

megawave demo - retinex...

Erreurs de chargement de la p...

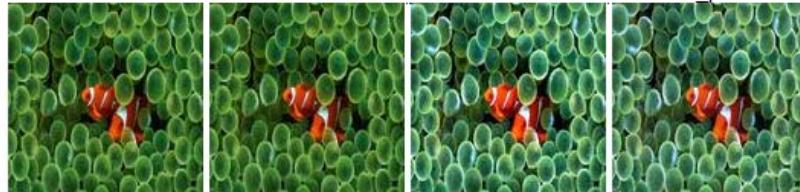
> mw > megawave > demo > retinex_pde

MEGAWAVE DEMO - RETINEX_PDE PUBLIC ARCHIVES

This archive is updated every hour. It is not moderated; in case of copyright infringement or similar problem, please [contact us](#) to request the removal of some images.

pages : 50 100 150 200 250 300

- ◆ 2009-09-30 13:06:46 - deb4db3fd0caf6011a63173b46542166 - retinex_pde version 2.04.20090613



- ◆ 2009-09-21 14:22:56 - 3ed7539eff1589eb14f1a51073bbcc47 - retinex_pde version 2.04.20090613



- ◆ 2009-09-14 11:49:03 - ce2cc1566bbfae101ffaeed2494a4972 - retinex_pde version 2.04.20090613



- ◆ 2009-08-28 06:23:12 - 1fabe644068a2fa7ebef1fbe93265e1e - retinex_pde version 2.04.20090613



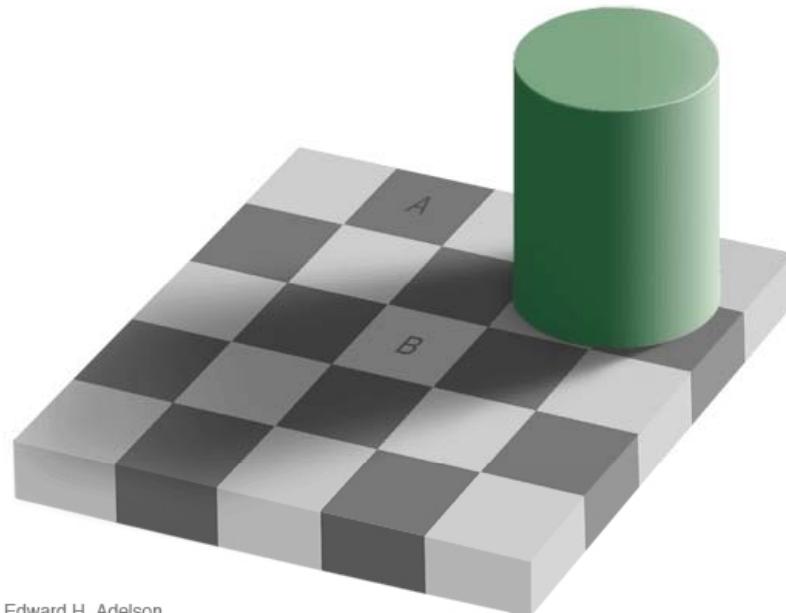
< >

Terminé

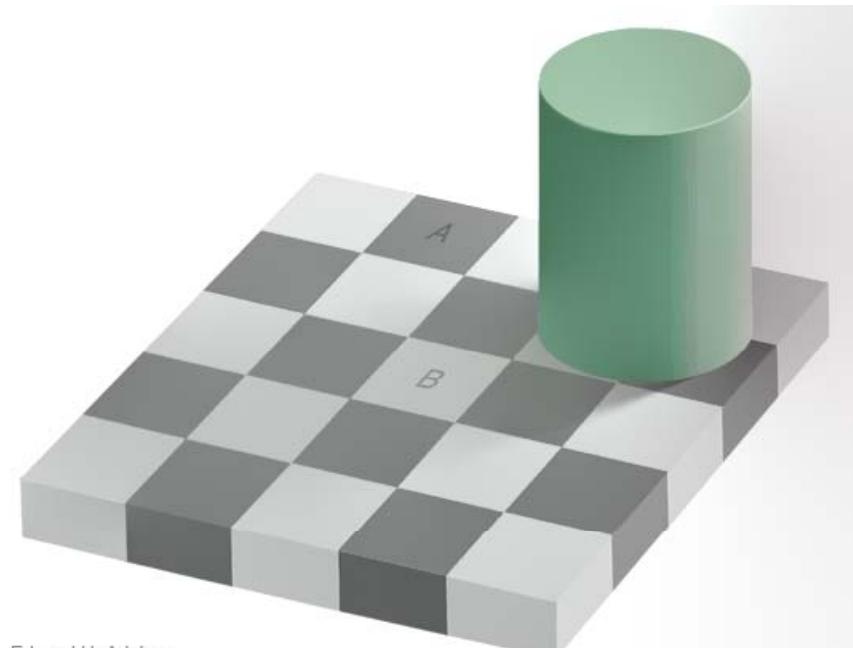


Some Results

The gray level of A and B is 120



The gray level of A is 145 and B is 190

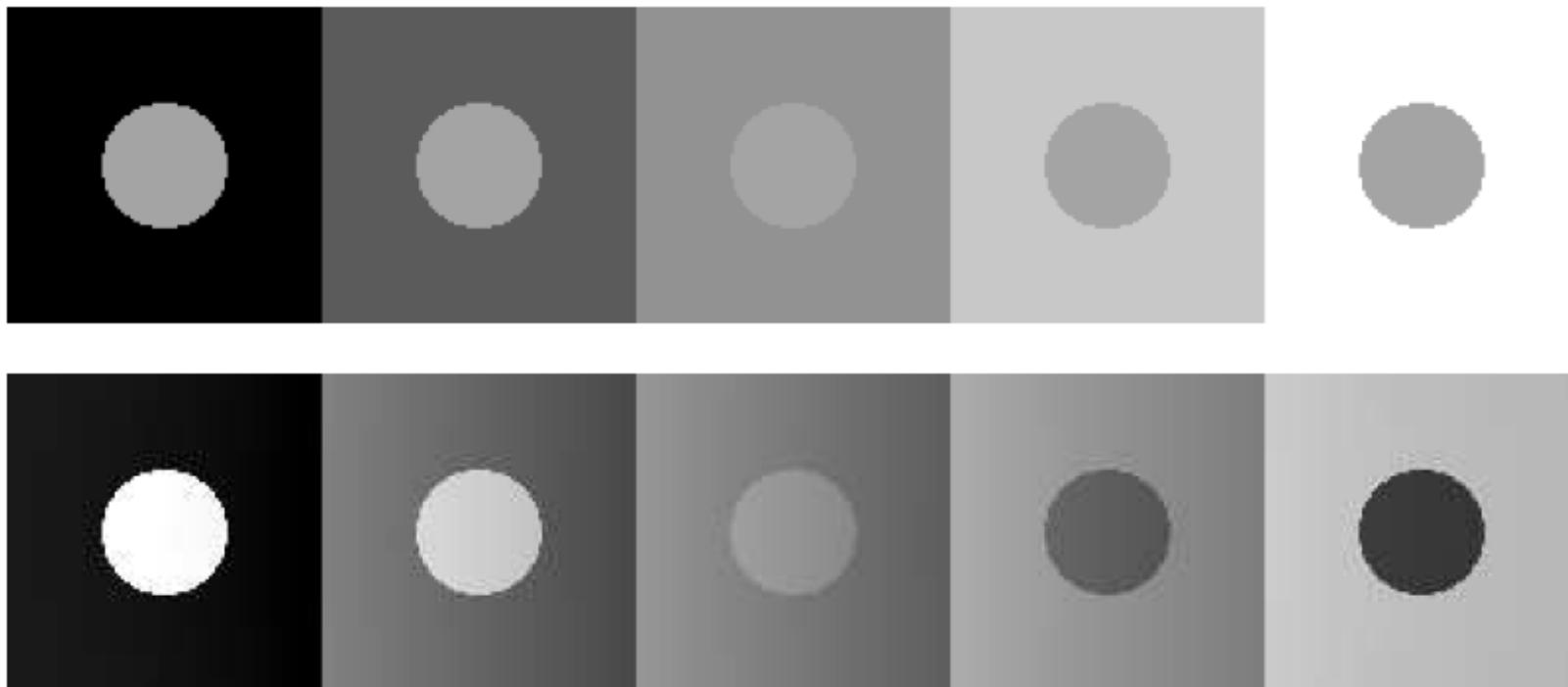


EI 2009

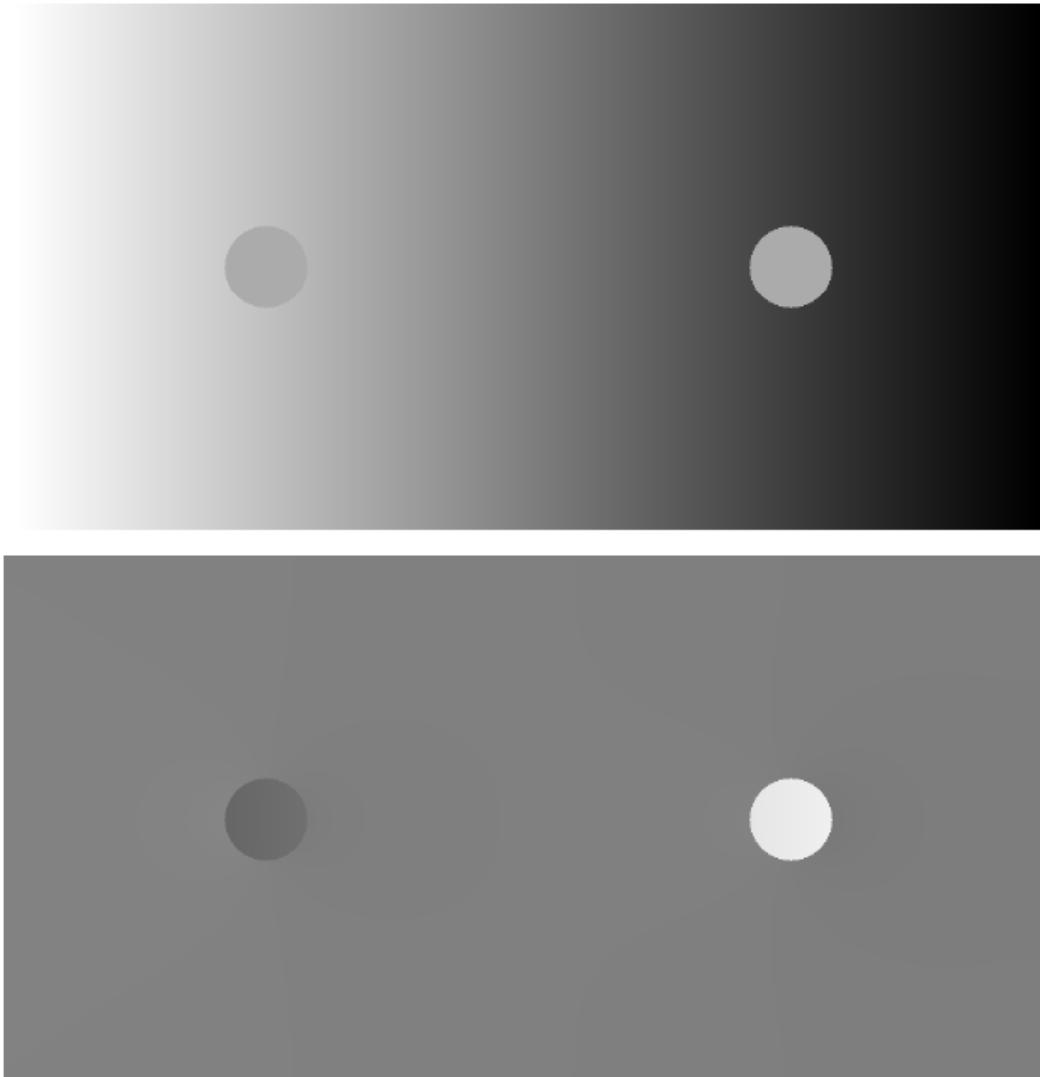
Retinex: color constancy ignoring shadows



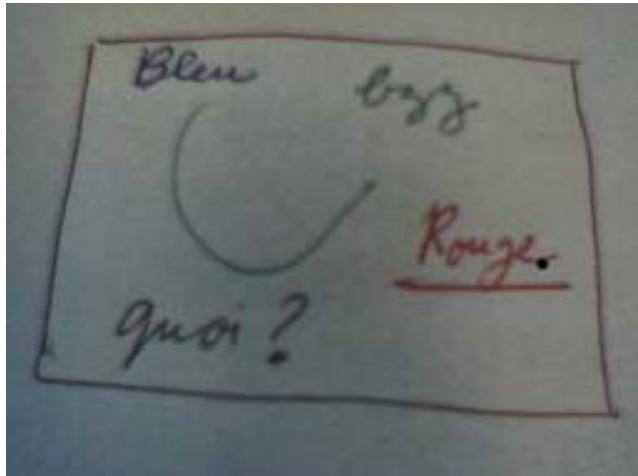
Retinex Theory



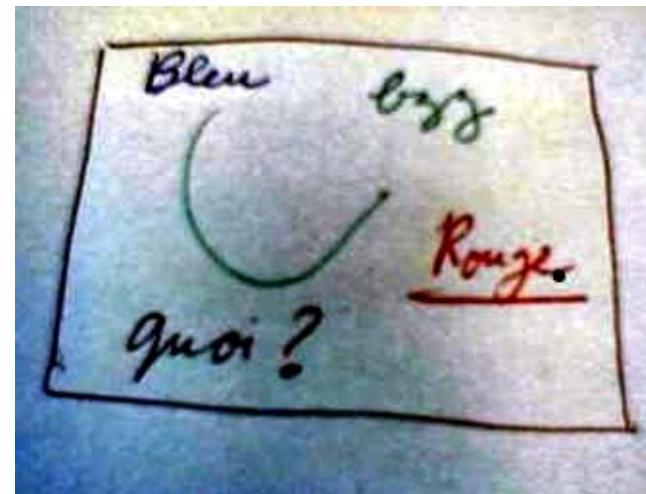
Retinex Theory



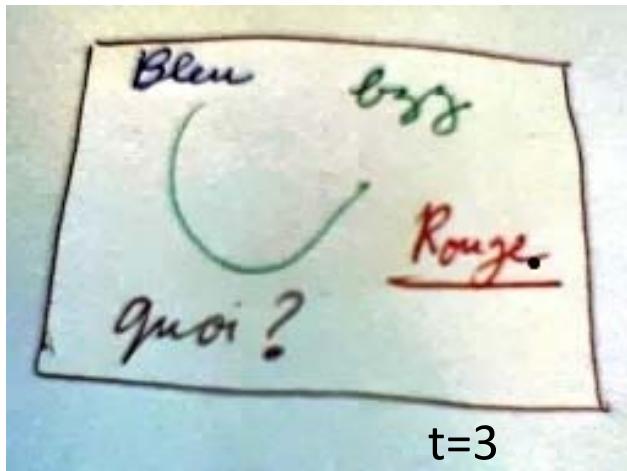
Retinex results



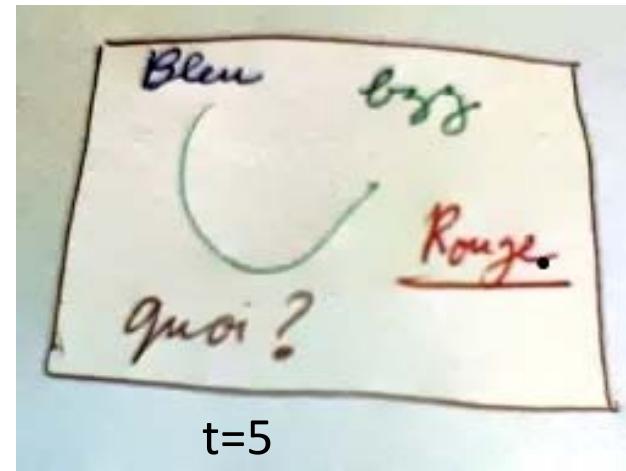
Original



Color balanced



t=3



t=5

EI 2009

Local contrast adjustment. Algorithm

Given an image f and with the values for T and α the proposed algorithm is fully automatic:

- Select the dark region Ω by the threshold T .
- Define the guidance vector field by

$$\mathbf{V} = \begin{cases} \nabla f & \text{in } R \setminus \Omega \\ \alpha \nabla f & \text{in } \Omega \end{cases}$$

where $\alpha \in [2, 3]$ and in the experiment we have took $\alpha = 2.5$.

- Solve the Poisson equation with Neumann boundary conditions using the Fourier transform as explained in a posterior section.

$$\Delta u = \operatorname{div} \mathbf{V}, \quad \text{over } R, \quad \frac{\partial u}{\partial \mathbf{n}} = 0 \quad \text{over } \partial R,$$

Local contrast adjustment. Examples



Local contrast adjustment. Examples



Local contrast adjustment. Examples



Local contrast adjustment. Examples



Local contrast adjustment. Examples



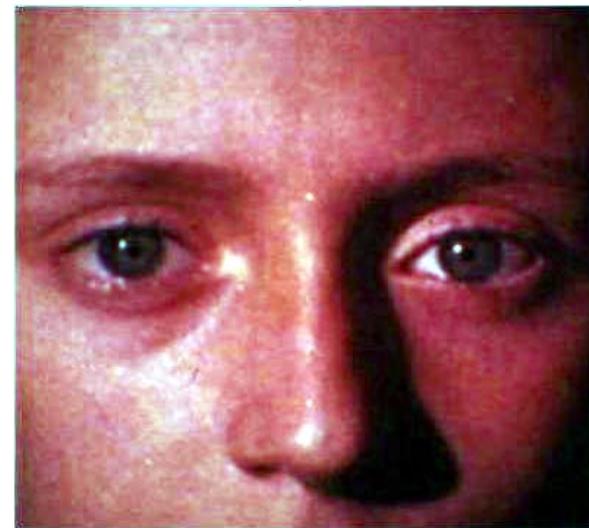
Local contrast adjustment. Examples



Experimental Results



Original Image



Colour balanced



<http://www.ipol.im>
Retinex t=10



Cachan, Heidelberg 2011
Retinex t=20

EI 2009



Original,
Color-Balanced,

Beltramio-Caselles-Provenzi,
Retinex_t=4



Beltramo-Caselles-Provenzi 1, 2, and 3
Original

Retinex t=2, Retinex t=10, Retinex t=20
Cachan, Heidelberg 2011

