



**HAL**  
open science

# Direct and inverse imaging. Applications to satellite design and to forgery detection.

Miguel Colom

► **To cite this version:**

Miguel Colom. Direct and inverse imaging. Applications to satellite design and to forgery detection.. Signal and Image Processing. ENS Paris-Saclay, 2019. tel-02418066

**HAL Id: tel-02418066**

**<https://theses.hal.science/tel-02418066>**

Submitted on 18 Dec 2019

**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.

# Direct and inverse imaging. Applications to satellite design and to forgery detection.

Habilitation à diriger les recherches de l'École normale supérieure Paris-Saclay (École normale supérieure de Cachan)

Spécialité: Sciences et technologie de l'information et de la communication

Thèse présentée et soutenue à l'ENS Paris-Saclay, le 20 novembre 2019,

par

**Miguel COLOM BARCO**

Composition du jury:

- Luis Álvarez  
PU Universidad de las Palmas de Gran Canaria — CTIM      Rapporteur
- Julie Delon  
PU Université Paris Descartes — MAP5      Rapporteur
- Présidente du jury
- Carola Schönlieb  
PU University of Cambridge — DAMPT      Rapporteur
- Andrés Almansa  
DR Université Paris Descartes — MAP5      Examineur
- Coloma Ballester  
TU Universitat Pompeu Fabra — GPI      Examineur
- Javier Portilla  
DR CSIC — Instituto de Óptica Daza de Valdés      Examineur



**Titre:** Direct and inverse imaging. Applications to satellite design and to forgery detection.

**Mots clés:** imagerie satellite, recherche reproductible, détection automatique falsification images

**Résumé:** Ce rapport d'habilitation à diriger des recherches est un condensé de mes recherches scientifiques récentes. Il couvre trois domaines principaux : (1) la conception de systèmes d'imagerie satellite, (2) la recherche reproductible et (3) la détection automatique de falsification d'images. Les recherches (1) et (3) portent sur les problèmes inverses, et (2) est une méthodologie scientifique qui est transversale à tout mon travail. Ce rapport traite principalement de mes recherches en cours et présente des résultats actuellement non publiés, ainsi qu'un résumé de certains travaux récents publiés dans des revues à comité de lecture. Le document est composé de trois parties, chacune commençant par une introduction générale au sujet et un résumé de mes contributions. Le dernier chapitre est consacré à mon plan de recherche et présente les projets dans lesquels je souhaite impliquer mes futurs doctorants et les aider à développer avec succès leur propre carrière scientifique.

**Title:** Direct and inverse imaging. Applications to satellite design and to forgery detection.

**Keywords:** satellite imaging, reproducible research, image automatic forgery detection

**Abstract:** This habilitation à diriger des recherches report is a digest of my recent scientific research. It covers three main areas: (1) design of satellite imaging systems, (2) reproducible research, and (3) automatic detection of forgery in images. Topics (1) and (3) deal with in inverse problems, and (2) is a transverse scientific methodology applied throughout my work. This report focuses on my ongoing research and presents (at the moment) unpublished results, and a summary of some selected recent works published in peer-reviewed journals. The document is made of three parts, each of them starting with a general introduction to the subject along with a summary of my contributions. The last chapter is devoted to my research plan and presents the projects in which I wish to involve my future PhD candidates to help them develop successfully their own scientific career.

École normale supérieure Paris-Saclay

61, avenue du Président Wilson

94235 Cachan, France

---

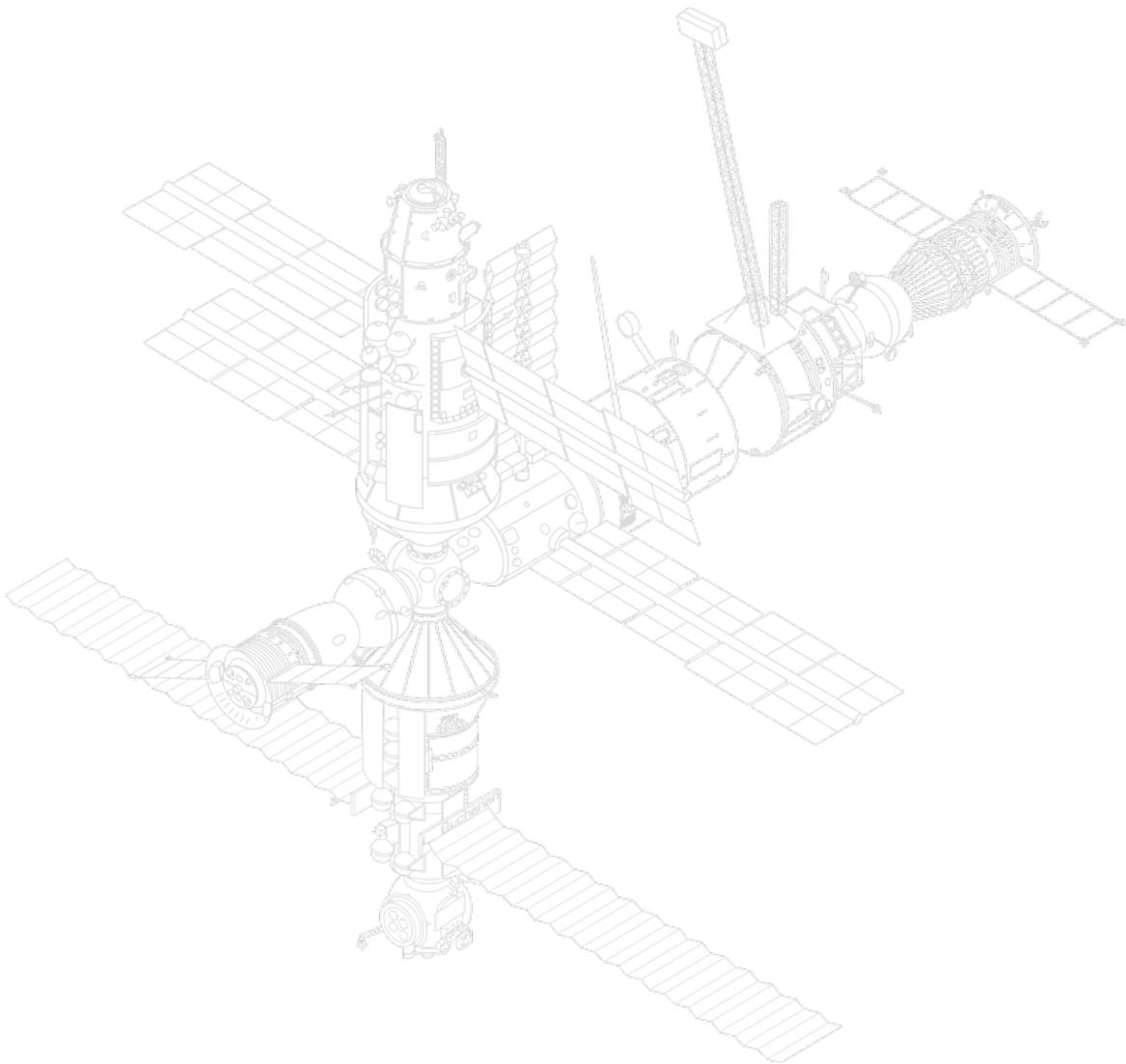


# Dedication

*Este documento de HDR está dedicado a ti, Lucas.*

*Qué menos, en muy poco tiempo tú me has dado ya mucho.*

*Como imagino que hay demasiado texto para tu gusto, he dejado algunas páginas en blanco para que puedas dibujar. Por favor, no te comas las “pinturetas”.*



## Acknowledgement

Writing this part of the report is quite pleasant. Indeed, I have had the pleasure to collaborate and to share all kinds of enriching experiences with so different people. Learning together, organizing workshops, supervising and being supervised, having fun, ...

I will start thanking Andrés Almansa, Luis Álvarez, Coloma Ballester, Julie Delon, Javier Portilla, Guillermo Sapiro, and Carola-Bibiane Schönlieb who kindly accepted to be part of the jury of my HDR. And a special acknowledgment for the three members that accepted to be the reviewers of the manuscript: Luis Álvarez, Julie Delon, and Carola-Bibiane Schönlieb. I hope you'll find it pleasing to read this summary of my research.

Including all the names of the PhD candidates and post-docs that I have met at CMLA would make this section absurdly long. Instead, I will restrict myself to my office's co-habitants: Thibaud Ehret, Axel Davy, and Valery Dewil, who made that room warm. And the rest of the lab, to make CMLA our big home (with a kitchen with good coffee and better conversations!)

Thanks to our great CMLA administrative team, that helps us with and even sometimes frees us from tedious paperwork. Without them our research activity would certainly slow down significantly. Thank you Véronique Almadovar for being always so professional, responsive, and fast (and for taking care of our last-minute project deadlines!). Alina Müller for managing all the details of our travel missions and helping us with all the procedures (and sometimes, so kind of doing it all instead of us). Virginie Pauchont for dealing with our complicated combinations of sandwiches, salads, and desserts every week, and more important... the free coffee! We will miss you when we move to our new location, by the way. We miss Micheline Brunetti too! Delphine Laverne for her administrative work with the students. And also to Sandrine Noury and Yamina Debbou, who arrived recently to CMLA. The IR team, with Christophe Labourdette and Atman Kendira. And Matthieu Christophe for being so incredibly responsive and helpful with any IT problem we might have!

A special thank you to the director of CMLA, Nicolas Vayatis, for having involved me in fresh projects (such as ML-Agora and others) and for his support and encouragement of reproducible research (one of the research axes) at CMLA. And of course to all the rest of *enseignants-chercheurs* of the lab: Thomas Alazard, Agnès Desolneux, Frédéric Dias (UCDublin), Alain Durmus, Gabriele Facciolo, Jean-Michel Ghidaglia, Argyris Kalogeratos, Enric Meinhardt, Yves Meyer, Mathilde Mougeot, Jean-Michel Morel, Frédéric Pascal, Vianney Perchet, Laure Quivy, Luba Tchertanov, Alain Trouve, Sandrine Dallaporta, Simon Halfon, Tuong-huy Nguyen, and Milan Perera. And the rest of researchers at CMLA (yes, the CMLA is big!) and CognAc-G (yes, the merger unit will be even bigger!).

Thank you very much to the many collaborators with whom I have or I have had the pleasure to work with in the IPOL team, developing our platform. I hope that what we have learned together in the project is useful to those who left the team to follow their own endeavours. Even if sometimes it is not easy for me to manage a team of quite competent people with their own (strong!) ideas, I really enjoy each of our discussions! Thank you for turning IPOL into a working reality. Also thanks to Jean-Michel Morel for his tireless work as one of the IPOL's Editors in Chief, José Luis Lisani for the careful copyediting of each of the articles IPOL published, Sandra Doucet for her editorial work (including *whipping* us when we get lazy as editors or authors!), and Eva Theumann also for her editorial management.

To Bertrand Kerautret for pushing and involving me in the organization of the two first editions of the Reproducible Research in Pattern Recognition (RRPR) workshops, along with Pascal Monasse. We have had to put a lot of effort in this initiative, but it is really satisfying to see that slowly it attracts more and more the interest of other researchers in this methodology. And I am happy that the third edition is ahead, and eager to go for it! Thank you also to all the rest of RRPR editors (Daniel Lopresti, Pascal Monasse, and Hugues Talbot) for their very valuable work. And to Springer for publishing the proceedings as part of the Lecture Notes in Computer Science series.

I thank the agencies that funded a great part of my research, including Centre National d'Études Spatiales (CNES, R&T action), European Research Council, Office of Naval Research (ONR), Direction Générale de l'Armement (DGA), and Agence Nationale de la Recherche (ANR). Also, to ENS Paris-Saclay for supporting me as a researcher.

A very special thank you to Bernard Rougé for introducing me into the satellite imaging topic in which I have been involved these last years (including the IASI-NG and SMOS-HR missions), and for helping me understand concepts of satellite imaging that were new to me. And thank you for being so kind of even letting me stay at your home several times when we traveled to Toulouse to meet with the CESBIO team! Thank you also Isabelle for her warmth.

Thank you to CESBIO's Eric Anterrieu, François Cabot, Ali Khazâal, Nemesio Rodríguez, and Yann Kerr for their collaboration and for providing SMOS data. And thanks also to CMLA's Gabriele Facciolo for his help and new ideas for the irregular sampling inversion.

Thanks to Tina Nikoukhah for being an excellent PhD candidate and for having accepted to be supervised by Jean-Michel Morel, Rafael Grompone and myself. Thank you Max Dunitz for accepting that I am your main PhD supervisor (I will do my best!), and thanks to Université Paris-Saclay for agreeing on my temporal HDR derogation to supervise Max. And thank you to Marina Gardella for accepting being also my PhD candidate, very soon. It is really exciting and satisfying to me to work with you all.

And finally, a very special thank you to Jean-Michel Morel for his absolute support in my scientific career and for being so generous with me not only professionally but also personally. If I do it right and am lucky enough, perhaps my own PhD candidates will someday think of me in a similar way.



# Contents

<b>1</b>	<b>Detailed curriculum vitæ</b>	<b>13</b>
1.1	Context and motivation . . . . .	13
1.2	Scientific and professional career . . . . .	14
1.3	Main research contracts, grants, and my role in them . . . . .	15
1.4	Doctoral thesis . . . . .	16
1.5	Education . . . . .	16
1.6	Publications and communications . . . . .	17
1.6.1	International peer-reviewed journal articles . . . . .	17
1.6.2	International peer-reviewed conferences . . . . .	17
1.7	Participation in the scientific life and collective responsibilities . . . . .	19
1.7.1	Congress organization . . . . .	19
1.7.2	Editorial boards and scientific committees . . . . .	19
1.7.3	Reviewer in international journals . . . . .	20
1.7.4	Invited talks . . . . .	20
1.7.5	Collaboration with other research teams . . . . .	21
1.7.6	Other . . . . .	21
<b>I</b>	<b>Design of satellite imaging systems</b>	<b>23</b>
<b>2</b>	<b>Introduction</b>	<b>25</b>
<b>3</b>	<b>Full-spectrum denoising of high-SNR hyperspectral images</b>	<b>29</b>
3.1	Introduction . . . . .	29
3.2	The noise model and its estimation . . . . .	31
3.2.1	Noise estimation . . . . .	32
3.2.2	Our real and simulated data . . . . .	33
3.3	The proposed denoising method . . . . .	34
3.3.1	Estimation of $C_n$ . . . . .	37
3.3.2	The conservative Lee correction . . . . .	38
3.3.3	Discussion on the choice of the hyperparameters . . . . .	39
3.4	Experimental validation of DBBD . . . . .	40
3.4.1	Evaluation with simulated USIs . . . . .	40
3.4.2	Evaluation with real USIs . . . . .	43
3.5	Consequences for the instrument's resolution . . . . .	45
<b>4</b>	<b>Ongoing research on the design of SMOS-HR</b>	<b>49</b>
4.1	Introduction to the questions of irregular sampling . . . . .	51
4.2	Model of the visibilities . . . . .	52
4.2.1	Discretization . . . . .	54
4.3	Optimizing the position of the antennas . . . . .	55
4.3.1	Separating antennas below the minimal allowed distance . . . . .	56
4.4	Calibration . . . . .	56

4.5	Irregular sampling and reconstruction . . . . .	57
4.5.1	Regularization of $G^*$ . . . . .	58
4.6	Evaluation . . . . .	58
4.6.1	Evaluation with a body of constant 300 K temperature . . . . .	60
4.6.2	Evaluation with an image of the Earth . . . . .	62
4.6.3	Point spread function of the instruments . . . . .	62
4.6.4	Convolution kernels . . . . .	66
4.7	Conclusions, perspectives, and future work . . . . .	66
<b>II</b>	<b>Contributions on reproducible research</b>	<b>69</b>
<b>5</b>	<b>Introduction</b>	<b>71</b>
<b>6</b>	<b>The present. IPOL: an online architecture for reproducible research</b>	<b>77</b>
6.1	Introduction . . . . .	77
6.2	State of the art in reproducible research platforms . . . . .	78
6.3	IPOL as a peer-reviewed scientific journal . . . . .	79
6.4	IPOL's system architecture . . . . .	80
6.4.1	IPOL's demo system modules . . . . .	82
6.5	Software engineering in the IPOL demo system . . . . .	83
6.5.1	Tools . . . . .	84
6.6	Editorial management: the Control Panel . . . . .	84
6.7	Automatic demo generation . . . . .	85
6.8	Conclusions . . . . .	86
<b>7</b>	<b>The future. Extending IPOL to machine-learning applications</b>	<b>89</b>
7.1	Plan to extend IPOL to machine-learning applications . . . . .	89
7.1.1	Extended architecture: support for applications . . . . .	90
7.1.2	Extension to new data types . . . . .	90
7.1.3	Comparison of algorithms . . . . .	92
7.1.4	Chaining of algorithms . . . . .	93
7.1.5	Machine learning applications . . . . .	94
7.1.6	System management with a unified tool . . . . .	96
7.2	Conclusions . . . . .	97
<b>III</b>	<b>Automatic detection of image forgery</b>	<b>99</b>
<b>8</b>	<b>Introduction</b>	<b>101</b>
<b>9</b>	<b>JPEG grid detection with controlled false alarms</b>	<b>105</b>
9.1	Introduction . . . . .	105
9.2	State of the art . . . . .	106
9.3	An <i>a-contrario</i> detector . . . . .	107
9.3.1	Grid extraction . . . . .	107
9.3.2	The voting process . . . . .	109
9.3.3	Statistical validation . . . . .	110
9.4	Results on several applications . . . . .	111
9.4.1	Grid detection . . . . .	111
9.4.2	Crop detection . . . . .	111
9.4.3	Copy-paste tampering detection . . . . .	112
9.5	Conclusion . . . . .	112
<b>10</b>	<b>JPEG grid detection based on the number of DCT zeros</b>	<b>115</b>

10.1	Introduction . . . . .	115
10.2	JPEG compression . . . . .	116
10.3	State of the art . . . . .	117
10.4	JPEG grid detection . . . . .	117
10.4.1	Voting process . . . . .	117
10.4.2	Statistical validation . . . . .	119
10.5	Forgery detection . . . . .	122
10.6	Experimental results . . . . .	123
10.6.1	Grid detection . . . . .	123
10.6.2	Forgery detection . . . . .	126
10.7	Conclusion . . . . .	128
<b>11</b>	<b>Research plan</b>	<b>129</b>



# Chapter 1

## Detailed curriculum vitæ

### **Miguel Colom Barco**

*Enseignant-chercheur* in École Normale Supérieure Paris-Saclay

Centre de Mathématiques et Leurs Applications (CMLA)

CMLA UMR CNRS 8536

### **Professional Address**

Bâtiment Laplace

CMLA - ENS Paris-Saclay

61 av. du président Wilson

94235 CACHAN cedex

**Website:** <http://mcolom.info>

### ***Garant:***

Prof. Jean-Michel Morel

[morel@cmla.ens-cachan.fr](mailto:morel@cmla.ens-cachan.fr)

## 1.1 Context and motivation

I am a permanent *enseignant-chercheur* at CMLA, ENS Paris-Saclay since 2018, specializing in the mathematical aspects and applications of image, video, and general signal processing.

I have worked deeply on the processing chain of digital cameras (from the acquisition of the raw image at the CMOS or CCD sensors to the final JPEG image) and especially in **denoising**. I worked with CNES (the French aerospace agency) and other agencies on improving the hyperspectral captors of Earth-observation satellites (**IASI**, **IASI-NG**), as well as on the design of new satellites (**SMOS-HR**). I plan to supervise a PhD student (Max Dunitz) on this subject starting in Oct. 2019.

In collaboration with the company **DxO**, I have contributed to improve smartphone digital photographs, particularly on denoising and noise estimation. Regarding satellite imaging, I proposed a noise estimation + denoising algorithm to CNES to be used on-ground to improve the images

acquired by Pleiades and OTOS satellites with **denoising**. If the procedure is done on-board, it **improves the compression performance**, along with other potential benefits.

I contributed to the **design of the new IASI-NG** satellite sensor and data acquisition. Our preliminary results show that we might **increase the resolution of the satellite by a factor of more than 7**. These results have been published in the proceedings of the **WHISPERS 2016 conference** (UCLA, California), and in the **Journal of the Optical Society of America A (JOSA A)**. Also in collaboration with CNES I am contributing to the development of the new **SMOS satellite** by characterizing the **gains of the antennas** and **de-aliasing, denoising, and increasing the resolution** of the acquired data.

I am also working on the **DEFALS** project, funded by the French **ANR**, which comprises the **detection of forgery and manipulation of images** from the analysis of the modified images itself, by analyzing its noise level curves, JPEG quantification, or the detection of duplicated zones in the images found by SIFT analysis, among other techniques. I am the co-adviser of a PhD student and I have also co-directed an M2 master (MVA) student, both related to this project.

I am involved in depth with the **Reproducible Research** initiatives at CMLA. Specifically, since 2014 I am in charge of a team of currently three **engineers** and two **postdocs** in order to improve the current system and develop the new version (the current one) of the demo system of the Image Processing On Line (**IPOL**) journal (<http://www.ipol.im>). Even if the journal is already a working system, the objective is to develop several **new features and improvements**, such as scalability, distributed computing, optimal data storage and data retrieval, move to a micro-service architecture, improvement of the UX of the interface, ensured updated documentation, among others. The full source code of the demo system itself is available<sup>1</sup> as free software.

Also, I foster the principles of Reproducible Research and good practices for reliable science with both the License and Master students in my classes in the Mathematics Department at ENS Paris-Saclay, as well with my co-advised students. More formally, I have published several articles, as well as organizing several workshops on the subject. I am one of the founding editors of IPOL, and part of its Editorial Board currently.

My plan for the future is to go on with my research project (see chapter 11) and to help new researchers in the field (PhD candidates) to develop their own scientific career. My profile with a strong background in both Mathematics and Engineering is, in my opinion, well adapted to the needs of nowadays applied research in signal and image processing. For these reasons, I am applying to obtain the *Habilitation à Diriger des Recherches* at ENS Paris-Saclay.

## 1.2 Scientific and professional career

### Scientific career:

- Sep. 2018 - now: *enseignant-chercheur* (permanent position) at CMLA, ENS Paris-Saclay
- Jan. 2017 - Sep 2018: **associate researcher** at CMLA, ENS Paris-Saclay
- Mar. 2016 - Dec. 2016: **CNRS postdoctoral researcher**
- Sep. 2014 - Dec. 2015: **CNRS data science engineer**. Funded by the **Paris-Saclay Center of Data Science**<sup>2</sup>.

### Professional career in the industry:

<sup>1</sup><https://github.com/mcolom/ipolDevel>

<sup>2</sup><http://www.datascience-paris-saclay.fr/>

- May 2007 - Jun. 2008: **senior software analyst** at Juniper Consulting S.L.<sup>3</sup>
- Nov. 2006 - May 2007: **software engineer** at SM2 Baleares S.A.<sup>4</sup>. Outsourced working at the Sol Meliá S.A.<sup>5</sup> client.
- Feb. 2004 - Nov. 2006: **software analyst and team coordinator** at Alta Tecnología Aplicada S.L. (AltecSoft)<sup>6</sup>.

## 1.3 Main research contracts, grants, and my role in them

CMLA, ENS Paris-Saclay contracts to which I have participated:

- **Twelve Labours (ERC advanced grant, 1,800,000 €, 2010-2015)**. My contribution: I published the state of the art on noise estimation in several IPOL articles and I took over the management of the whole online installation in 2014.
- **DxO-Labs denoising (annual contract, 80,000 €, 2011-2016)**. My contribution: I designed for DxO new unsupervised noise estimators and a first model of fully unsupervised denoising algorithm for JPEG images.
- **CNES IASI-NG (42,000 €, 2016)**. My contribution: I was the PI of this project and invented a new denoising method for hyperspectral images.
- **CNES SMOS-HR (120,000 €, 2017-2019)**. My contribution: I am the PI of this project and I am preparing the simulator of the new interferometer measuring ground temperature and water salinity in L-Band, that will succeed the current SMOS satellite. I develop new methods for denoising, calibration, removal of parasites, and super-resolution. These algorithms will operate partly on board and partly on the ground.
- **ONR (Office of Naval Research, 2013-2016, \$360,000)**. My contribution has been the development of the IPOL new online operating facility and the design of a very simple editor of online demos that new comers can learn in 30 minutes. Also, implementing new datatypes other than images, such as video and interactive 3D data. <http://www.ipol.im>
- **ANR (Agence Nationale de la Recherche) DEFALS project, from 09/2017 to 09/2020, 370,000 €)**. Senior researcher in this project of automatic and blind detection of falsifications in digital images. It implies developing image processing algorithms to determine the forged images and identify which zones were modified. This project is presented as a challenge for the four competing teams. <https://www.defals.fr>
- **Kayrros, from 06/2018 to 06/2021, 1,080,000 €)**. Kayrros is a start-up focused on the exploitation of recurrent remote sensing data for the extraction of indices of economic activity in the energy sector. Also, automatic analysis of time series of satellite images obtained by passing through the same orbit. Several subjects have been addressed, including detection of cloud cover, analysis of changes over time, as well as detection of structures. I participate in the project since the beginning of the collaboration.

→ **Incoming contracts and projects** CMLA, ENS Paris-Saclay contracts that are being negotiated and to start:

---

<sup>3</sup><https://www.ejuniper.com/>

<sup>4</sup><https://www.sm2baleares.es/>

<sup>5</sup><http://www.melia.com>

<sup>6</sup><https://cuiner.com/>

- **Chaire Industrial Data Analytics & Machine Learning (Atos, CEA and ENS Paris-Saclay).** In 2016 a collaboration agreement was signed between Atos, CEA, and ENS Paris-Saclay. Atos has expressed a large interest in the IPOL demo system as well as in the algorithms published by the journal. At this moment we are negotiating the terms of a collaboration which will benefit both Atos and IPOL. On one hand, Atos will benefit of the expertise of the CMLA to exploit the published algorithms and on the other hand IPOL will gain a larger visibility and use the supercomputing resources provided by Atos. This will allow IPOL to execute video-processing algorithms in a reasonable time. As the responsible for the IPOL demo system and the team of engineers which develop the new features in IPOL, my role in this collaboration is essential for the integration of the IPOL and Atos platforms.
- **BUF.** Founded in 1984, BUF<sup>7</sup> is a well-established French company of visual effects. Leader in Europe, it has developed its own software for VX which has been used in many movies and TV shows<sup>8</sup> I was contacted by them in June 2018 to give advise on image denoising and noise estimation, which has led to the current negotiation of a collaboration.

## 1.4 Doctoral thesis

**Title:** Multiscale Noise Estimation and Removal for Digital Images

**Advisors:** Dr. Antoni Buades (Universitat de les Illes Balears) and Prof. Dr. Jean-Michel Morel (CMLA, ENS Cachan)

**Jury:** Dr. Andrés Almansa (Université Paris Descartes, Paris, France), Prof. Bartomeu Coll (Université de les Illes Balears, Palma, Espagne), Prof. Gabriele Facciolo (ENS Paris-Saclay, France), Dr. José Luis Lisani (Université de les Illes Balears, Palma, Espagne), Dr. Enric Meinhardt (ENS Paris-Saclay, France).

My PhD thesis presents a **thorough study** of the **transformations of the noise** in a digital camera sensor (CMOS or CCD) **throughout the complete image signal processing pipeline** (acquisition of the raw mosaic, demosaicing, white balance, tone curve correction, and JPEG compression), and the problem of **estimating and reducing the noise**, especially from the raw image and also from the final **JPEG-compressed** image. The minimization of the noise is **blind**, in the sense that the only information available is the **noisy image itself**. Thus, my PhD thesis focused not only on the theoretical problem, but especially on a **practical computationally optimized solution**. In concrete, I developed **accurate noise estimators** for raw images and JPEG-encoded images, and applied them to the **denoiser** we built. This allows to **blindly remove the noise** from images with **unknown noise model** (with the minimal assumption that the noise depends on the intensity and not the position). All the **source codes** are available at the IPOL journal (<http://www.ipol.im>) as free software and come along with a demo that allows users to test the algorithms with their data. The **C++ code** of all methods is **parallelized with OpenMP** and is also available to download.

PhD thesis: [http://mcolom.info/download/thesis/thesis\\_miguel\\_colom.pdf](http://mcolom.info/download/thesis/thesis_miguel_colom.pdf)

## 1.5 Education

- 2010-2014: **PhD in Applied Mathematics.** Universitat de les Illes Balears (Spain)
- 2011-2015: **Degree in Computer Science.** Universitat Oberta de Catalunya (Spain)
- 2010: **M2 Master's in Mathematics/Vision/Learning ("MVA").** École Normale Supérieure de Cachan (France)
- 2009: **Master's in Free Software.** Universitat Oberta de Catalunya (Spain)

<sup>7</sup><http://buf.com>

<sup>8</sup>As for example Blade Runner 2049, Star Trek Discovery, The Lobster, and others.

- 1998-2003: **Degree in Telecommunication Engineering.** Universitat de les Illes Balears (Spain)

## 1.6 Publications and communications

### 1.6.1 International peer-reviewed journal articles

- M. Colom, J.M. Morel (2019). Full-Spectrum Denoising of High-SNR Hyperspectral Images. *Journal of the Optical Society of America A*. 36, 450-463. DOI:10.1364/JOSAA.36.000450.

- M. Lebrun, M. Colom, J. Darbon, and J.M. Morel (2018). Numerical Simulation of Landscape Evolution Models. *Image Processing On Line*. DOI:10.5201/ipol.2018.205.

- M. Colom, A. Buades (2016). Analysis and Extension of the PCA Method, Estimating a Noise Curve from a Single Image. *Image Processing On Line*. DOI:10.5201/ipol.2016.124.

- M. Colom, M. Lebrun, A. Buades, and J.M. Morel (2015). Nonparametric Multiscale Blind Estimation of Intensity-Frequency Dependent Noise. *IEEE Transactions On Image Processing*. DOI:10.1109/TIP.2015.24385

- M. Lebrun, M. Colom, and J.M. Morel (2015). Multiscale Image Blind Denoising. *IEEE Transactions On Image Processing*. DOI:10.1109/TIP.2015.2439041.

- M. Colom, A. Buades, J.M. Morel (2014). Nonparametric Noise Estimation Method for Raw Images. *Journal of the Optical Society of America A*. DOI:10.1364/JOSAA.31.000863.

- M. Colom, A. Buades (2014). Analysis and Extension of the Percentile Method, Estimating a Noise Curve from a Single Image. *Image Processing On Line*. DOI:10.5201/ipol.2013.90.

- M. Colom, A. Buades (2013). Analysis and Extension of the Ponomarenko et al. Method, Estimating a Noise Curve from a Single Image. *Image Processing On Line*. DOI:10.5201/ipol.2013.45.

- M. Lebrun, M. Colom, A. Buades and J. M. Morel (2012). Secrets of Image Denoising Cuisine. *Acta Numerica*, 21, pp 475-576. DOI:10.1017/S0962492912000062.

### 1.6.2 International peer-reviewed conferences

- N. Rodríguez-Fernández, E. Anterrieu, B. Rougé, J. Boutin, G. Picard, T. Pellarin, M.J. Escorihuela, A. Al Bitar, P. Richaume, A. Mialon, O. Merlin, C. Suere, J. Costeraste, B. Palacin, R. Rodríguez-Suquet, T. Decoopman, F. Cabot, M. Colom, J.-M. Morel, Y. H. Kerr. *A High Spatial Resolution SMOS Follow-on: SMOS-HR*. 2019 IEEE International Geoscience and Remote Sensing Symposium - IGARSS 2019. Yokohama, Japan - 28 July - 2 August, 2019.

- É. Anterrieu, N. Rodríguez-Fernández, B. Rougé, F. Cabot, P. Richaume, A. Khazaal, Y. H. Kerr, J.-M. Morel, M. Colom, J. Costerate, B. Palacin, R. Rodríguez-Suquet, T. Tournier, T. Decoopman, R. Caujolle, N. Jeannin, L. Costes, F. Payot. *Preliminary System Studies on a High-resolution SMOS Follow-on: SMOS-HR*. 2019 IEEE International Geoscience and Remote Sensing Symposium - IGARSS 2019. Yokohama, Japan - 28 July - 2 August, 2019.

- Tina Nikoukhah, Miguel Colom, Jean-Michel Morel, Rafael Grompone (2019). *Détection de Grille JPEG Par Compression Simulée*. GRETSI 2019: XXVII colloque du Groupement de Recherche en Traitement du Signal et des Images. (In peer-review).

- T. Nikoukhah, J. Anger, T. Ehret, M. Colom, J.M. Morel, R. Grompone (2019). JPEG Grid Detection Based on the Number of DCT Zeros and its Application to Automatic and Localized Forgery Detection. *CVPR 2019: Workshop on Media Forensics*. June 15-21, 2019, Long Beach, USA.
- T. Nikoukhah, R. Grompone, M. Colom, J.M. Morel (2018). Automatic JPEG Grid Detection with Controlled False Alarms, and its Image Forensic Applications. 1st IEEE International Workshop on "Fake MultiMedia" (FakeMM'18) April 10-12, 2018, Miami, FL, USA.
- T. Nikoukhah, R. Grompone, M. Colom, J.M. Morel (2018) Automatic JPEG Grid Detection with Controlled False Alarms, and its Image Forensic Applications. *1st IEEE International Workshop on "Fake MultiMedia" (FakeMM'18)*.
- M. Colom. *Extending IPOL to New Data Types and Machine-Learning Applications* RRPR 2018: 2nd Workshop on Reproducible Research in Pattern Recognition. International Conference on Pattern Recognition (ICPR).
- M. Colom, B. Kerautret, A. Krähenbühl *An Overview of Platforms for Reproducible Research and New Ways of Publications* RRPR 2018: 2nd Workshop on Reproducible Research in Pattern Recognition. International Conference on Pattern Recognition (ICPR).
- Y. Kerr, B. Rougé, J.M. Morel, M. Colom, N. Rodríguez-Fernández, E. Anterrieu, F. Cabot, A. Khazaal (2018) SMOS-HR mission: Towards a 10 km resolution radiometer. *15th Specialist Meeting on Microwave Radiometry and Remote Sensing of the Environment (MicroRad 2018)*.
- M. Arévalo, C. Escobar, P. Monasse, N. Monzón, M. Colom (2016). The IPOL Demo System: a Scalable Architecture of Microservices for Reproducible Research. *RRPR 2016: 1st Workshop on Reproducible Research in Pattern Recognition. International Conference on Pattern Recognition (ICPR)*. DOI:10.1007/978-3-319-56414-2\_1.
- M. Colom, G. Blanchet, A. Klonecki, O. Lezeaux, E. Pequignot, F. Poustomis, C. Thiebaut, S. Ythier, J.M. Morel (2016). BBD: a New Bayesian Bi-Clustering Denoising Algorithm for IASI-NG Hyperspectral Images. *8th Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing*. DOI:10.1109/WHISPERS.2016.8071745.
- M. Colom, B. Kerautret, N. Limare, P. Monasse, and J.M. Morel (2015). IPOL: a new journal for fully reproducible research; analysis of four years development. *7th IFIP International Conference on New Technologies, Mobility and Security*. DOI:10.1109/NTMS.2015.7266500.
- M. Colom, M. Lebrun, A. Buades, and J.M. Morel (2014). A non-Parametric Approach for the Estimation of Intensity-Frequency Dependent Noise. *IEEE International Conference on Image Processing (ICIP)*. DOI:10.1109/ICIP.2014.7025865.
- M. Lebrun, M. Colom, and J.M. Morel (2014). The Noise Clinic: a Universal Blind Denoising Algorithm. *IEEE International Conference on Image Processing (ICIP)*. DOI:10.1109/ICIP.2014.7025541.
- Miguel Colom, Gabriele Facciolo, Marc Lebrun, Nicola Pierazzo, Martin Rais, Yi-Qing Wang, and Jean-Michel Morel (2014). A Mathematical Perspective of Image Denoising. *Proceedings of the International Congress of Mathematicians*, Seoul. Volume 4, pp 1061-1085.  
→ Proceedings available at <http://www.icm2014.org/en/vod/proceedings.html>.
- M. Colom and J.L. Lisani (2004). An Improved Frequency-domain Technique for Robust Image Watermarking. *IWSSIP'04: IEEE International Workshop on Systems, Signals and Image Processing*, pp

## 1.7 Participation in the scientific life and collective responsibilities

### 1.7.1 Congress organization

I am part of the **Organization Team** and **Chair** of the **Workshop on Reproducible Research in Pattern Recognition (RRPR)**:

- First edition, 4 Dec. 2016 Cancún (México)
- Second edition, 20 Aug. 2018 Beijing (China)

The proceedings of both editions have been included in two books of the Lecture Notes in Computer Science published by Springer:

- First edition  
ISSN 0302-9743, ISSN 1611-3349 (electronic).  
Lecture Notes in Computer Science  
ISBN 978-3-319-56413-5, ISBN 978-3-319-56414-2 (eBook)  
DOI: <https://doi.org/10.1007/978-3-319-56414-2>
- Second edition  
ISSN 0302-9743, ISSN 1611-3349 (electronic)  
Lecture Notes in Computer Science  
ISBN 978-3-030-23986-2, ISBN 978-3-030-23987-9 (eBook)  
DOI: <https://doi.org/10.1007/978-3-030-23987-9>

### 1.7.2 Editorial boards and scientific committees

I am or have been a member of the following **editorial boards** and **scientific committees**:

- Editorial Board of the journal Image Processing On Line (IPOL);
- Program Committee of the 20th Iberoamerican Congress on Pattern Recognition (CIARP) 2015;
- Scientific Committee of the 21th Iberoamerican Congress on Pattern Recognition (CIARP) 2016;
- Scientific Committee of the 1st Workshop on Reproducible Research in Pattern Recognition (in ICPR 2016);
- Scientific Committee of the 2nd Workshop on Reproducible Research in Pattern Recognition (in ICPR 2018);
- Associate board member of TC18 IAPR Technical Committee on Discrete Geometry and Mathematical Morphology.

### 1.7.3 Reviewer in international journals

I have reviewed articles for the following journals:

- Journal of Mathematical Imaging and Vision
- SIAM Journal on Imaging Sciences
- IEEE Transactions on Image Processing
- IEEE Signal Processing Letters
- IEEE Transactions on Geoscience and Remote Sensing
- Applied Optics
- Journal of the Optical Society of America A (JOSA-A)
- OSA - Biomedical Optics Express
- Signal, Image and Video Processing
- Computer Vision and Image Understanding
- MicroRad 2018, 15th Specialist Meeting on Microwave Radiometry and Remote Sensing of the Environment
- Image Processing On Line
- Journal of Imaging
- Remote Sensing

### 1.7.4 Invited talks

I have been invited to give the following talks to communicate my research activity to the community:

- Journée Scientifique at Université Paris 13, 4 Dec 2018. Present and Future of the IPOL Journal. Machine Learning Applications.
- RRPR 2018: 2nd Workshop on Reproducible Research in Pattern Recognition, 20 Aug in Beijing (China). I'm co-organizing this workshop within the International Conference on Pattern Recognition (ICPR) 2018, along with Bertrand Kerautret, Pascal Monasse, and Jean-Michel Morel.
- Present and Future of the IPOL Journal. Machine Learning Applications, keynote, 20 Aug 2018, Beijing, RRPR workshop.
- An Overview of Platforms for Reproducible Research and New Ways of Publications, keynote, 20 Aug 2018, Beijing, RRPR workshop.
- RRPR 2016: 1st Workshop on Reproducible Research in Pattern Recognition, 4 Dec 2016 in Cancún, México. Co-organizer along with Bertrand Kerautret, Pascal Monasse, and Jean-Michel Morel.
- THORR Workshop (Training and Hands-On in Reproducible Research Workshop) at the 20th Iberoamerican Congress on Pattern Recognition, Friday 13 and Saturday 14 November, 2015, Montevideo (Uruguay). I taught the THORR workshop on Reproducible Research within CIARP 2015. It included topics such as source code in reproducible research, legal aspects, the IPOL journal, and discussion of particular cases.

- Open Software Initiative (OSI) day, October 26, 2015, Paris-Saclay Center for Data Science. Talk: Building a service-oriented platform for online physiological data analysis
- Computational Photography and Intelligent Cameras, February 4-6, 2015, IPAM, UCLA. I presented a poster on blind multiscale noise estimation and denoising with real images.
- *Co-conception de systèmes hybrides : quand l'instrumentation et les traitements numériques se rencontrent*, October 31, 2014, Télécom Paris. Talk: Characterization of noise throughout the camera processing chain and denoising of images with unknown noise model.
- Workshop on Reproducible Research in Signal, Image and Geometric Processing, from July 22 to 27, 2013. Fondation des Treilles.
- Young researchers' workshop on Statistics, Learning and Variational Methods in Imaging, 20-21 September 2012, DAMTP, University of Cambridge (UK). Talk: the Noise Clinic.

### 1.7.5 Collaboration with other research teams

I have collaborated with several research groups, including:

- TAMI (Mathematical Analysis and Processing of Images) group of Universitat de les Illes Balears, Spain.
- Facultad de Ingeniería of Universidad de la República, Uruguay.
- Laboratory for Scientific Image Analysis at the Facultad de Medicina, Santiago de Chile, Chile.
- Departamento de Computación at the Universidad de Buenos Aires, Argentina (from April 10 to April 21, 2015).
- AMI Research Group at the Universidad de Las Palmas de Gran Canaria, Spain.

### 1.7.6 Other

- I am a member of the MISS<sup>9</sup> workgroup organized by CMLA and CNES, on satellite imaging. I regularly expose the advances in my research to other colleagues.
- I have co-organized several activities in order to involve the scientific community in Reproducible Research. Namely, the two editions of the Workshop on Reproducible Research in Pattern Recognition within ICPR (the first edition in 2016 in Cancún and the second in 2018 in Beijing).
- I participated in the STIC - AmSud<sup>10</sup> *Programa regional de cooperación científico-tecnológica entre Argentina, Brasil, Chile, Colombia, Ecuador, Paraguay, Perú, Uruguay, Venezuela y Francia* as a senior researcher, involved mainly in the reproducible research aspects.
- I am the designer and lead supervisor of the IPOL platform, the main tool for reproducible research of CMLA. Also, I have helped to the development of IPOL spin-offs such as ML-Agora (work in progress).
- I am part of the jury to evaluate international students of the *Algorithmique et Fondements de la Programmation (AFP)* M2 research master (ENS Paris-Saclay). Date of the jury meeting: May 27, 2019.
- I participate regularly in the weekly seminars of the imaging group, the GTTI<sup>11</sup>.

<sup>9</sup>*Mathématiques de l'Imagerie Satellitaire Spatiale.*

<sup>10</sup><http://www.sticmathamsud.org/>

<sup>11</sup>*Groupe de Travail sur le Traitement d'Image.*

- Finally, I have a clear responsibility in the education of the future researchers and professionals studying at ENS Paris-Saclay as a *enseignant-chercheur*, with a service of 96 HETD per year. Currently I intervene in three courses:
  - **Introduction mathématique au traitement et à l’analyse des images digitales, et à leurs surprenantes applications** (mathematical introduction to signal processing and analysis of digital images, and to their surprising applications). Level: master 1. Image processing/analysis has become a mathematical science where one can find the reformulations and new usages of many mathematical tools as well as new mathematical models. This course covers mainly imaging at the discrete and continuous domains: the representation of the images by the Fourier-Shannon method, applying geometrical transformation, color and contrast and the dynamics of images, their histogram equalization, Retinex perception theory, solving partial derivatives equations, Poisson editing, heat equation, the scale-space representation and its discretization, and the SIFT method and variants such as A-SIFT. The course consists on lectures as well as TD (supervised exercises) and TP (supervised practical classes using demos of the IPOL journal).
  - **Optimisation numérique** (numerical optimization). Level: master 1. This course is an introduction to optimization designed to provide a minimum baggage to any future mathematician. It deals mainly with problems in finite dimension but covers a number of key concepts, from unconstrained optimization to problems with constrained equalities and inequalities as well as the duality for convex optimization problems. It focuses on algorithms of digital optimization. Specifically, this course covers the optimal-step gradient descent method, Newton and quasi-Newton methods, optimization under constraints, duality for the convex problems, and proximal algorithms. The lectures of the course are re-enforced with several sessions of TD and TP. The TP sessions consist in coding several optimization method with Python.
  - **Analyse numérique des EDO** (numerical analysis of ODEs<sup>12</sup>). Level: licence 3. This course focuses on quadrature methods, interpolation and solving nonlinear problems that are very useful for scheme building. The study of EDOs with delay, with integro-differential terms, or with boundary conditions are also introduced. This course also includes TD and TP sessions with exercises that the students need to solve by coding the algorithms in Python.

These three courses account for 46 HETD. The other 50 HETD are dedicated to supervising the research internships of M1 (master 1) and L3 (license 3) students.

---

<sup>12</sup>Ordinary differential equation.

# Part I

## Design of satellite imaging systems



# Chapter 2

## Introduction

Since many years now, CMLA enjoys a fruitful scientific collaboration with CNES (the French aerospace agency), as well as other international public agencies and private companies (Kayros and others) working on satellite imaging. My first serious research on satellite imaging was as a postdoc at CMLA, when Bernard Rougé proposed that I analyzed several images obtained by the two Pleiades satellites. In my PhD thesis I worked on the estimation of noise and denoising in natural images and it seemed that it could be possible to apply the same principles to Pleiades images, or perhaps also to hyperspectral images with hundreds or even thousands of channels.

Even if that kind of data seemed at first glance to be of a very different nature compared to natural images, in the end the same principles still governed. Indeed, principles of denoising such as exploiting the redundancy of the data or aggregating it were equally valid and could be applied or extended to other particular types of data. We managed to improve the Pleiades images and we submitted our work to CNES. Since then, I have worked in two main projects in collaboration with CNES, that I have chosen for being presented in detail in this report.

The first project, which is presented in chapter 3, is about denoising of high-SNR hyperspectral (up to hundreds of narrow spectral channels) and ultraspectral images (thousands of channels). Hyperspectral imaging has a wide range of applications. For instance, in agriculture there exist methods that can determine the contents of water or minerals in crops. And other really amazing applications, such as remote sensing and characterization of chemicals on the surface of other planets, or even measuring the abundance of specific elements in stars<sup>1</sup>! During this project I had the opportunity to attend the WHISPERS<sup>2</sup> 2016 conference in Los Angeles and to submit our work. One of the plenary talks was by Dr. Bethany L. Ehlmann, a research scientist and professor of Planetary Science from the NASA's Jet Propulsion Laboratory. Her talk "*Imaging spectroscopy for planetary science: new discoveries and a look to the future*" convinced me that indeed there was a promising future for what has been called *non-conventional imaging systems* and that image processing had an important role in that new scenario. The combination of new kinds of acquisition systems along with the proper image processing techniques could improve significantly the applications. The knowledge that some image processing techniques have a great potential to improve the data must necessarily lead to re-think the design of the instruments. In the work that I present in chapter 3 we showed that the surface of the sensor of the future IASI-NG satellite may be reduced in a factor more than 50 while keeping the same SNR.

In this project we wanted to have something *realistic* in the sense that our algorithms should work with real data, not only with over-simplified simulations (as sometimes one can find in the literature). Our main objective was to propose an algorithm for the new IASI-NG satellite's sensor (launch expected in 2021) for which one can only work with simulations. However, there does already exist

---

<sup>1</sup>I could learn more on this after some exciting talks with CESBIO's physicist Nemesio Rodríguez.

<sup>2</sup>Workshop on Hyperspectral Images and Signal Processing: an Evolution in Remote Sensing.

a version, the IASI satellite (launched in 2006) and therefore real data was available. Therefore, we could verify that our noise estimations actually corresponded to the physical model given by the Planck law. And indeed, both our estimation and the physical model matched accurately. Therefore, we had strong evidence that our noise estimation was physically consistent and therefore could be applied to IASI-NG. We published very satisfying results which show that the techniques I present here again should be taken into account to improve the IASI-NG instrument itself in the very design phase.

The second project is absolutely ongoing research. It consists on the design of a satellite that measures the humidity of the soil and the salinity of the oceans by means of *interferometry*. This technique goes back to the early XIX century with the famous interference experiment by Thomas Young where he showed that light from a unique source breaking through two different slits created an interference pattern. Later, that led to the general acceptance that light could be (also) regarded as a wave. Years later, in 1880, Albert A. Michelson started developing an improved instrument (nowadays known as the *Michelson interferometer*), that splits a light source with a mirror and forces it follow two different paths, each of them with a detector at the end. The phase shift of the two waves arriving to the two detectors is measured. This beautifully simple but proven powerful technique has had many applications, and perhaps the most impressive was the direct detection of gravitational waves with the LIGO<sup>3</sup> instrument (a sophisticated Michelson interferometer) in September 2015. A gravitational wave made the distance between the two 4-Km long arms of the instrument *slightly* different, thus allowing to observe an interference pattern (a *fringe*) in the detector.



Northern arm of the LIGO interferometer on Hanford Reservation. Source: Wikipedia's user *Umptanum*, public domain.

Since then, many other gravitational waves originated from the merger of black holes have been detected, and even the merger of neutron stars (GW170817, detected by both the LIGO and Virgo detectors in August 17, 2017).

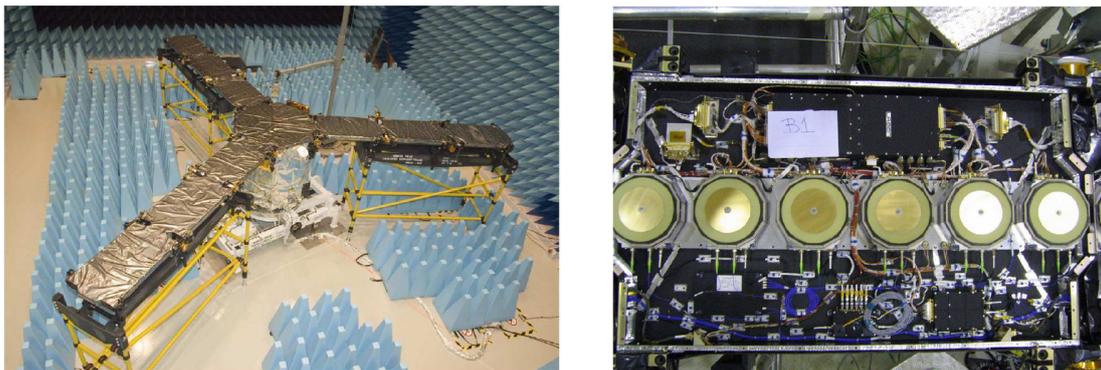
On 10 April 2019, the very first direct image of a black hole object (a supermassive one, in the center of the Messier 87 galaxy) and its vicinity was obtained by a large team of researchers working at the Event Horizon Telescope. The reconstruction combined interferometric reads from several different radiotelescopes, including VLA and ALMA. The very existence of this kind of astronomic object was a disputable topic not a long time ago. Even Albert Einstein published a paper in 1939 [1] where he concluded that they could not exist. However, in early 1916 Karl Schwarzschild had already found a non-trivial exact solution [2] to Einstein's field equations that showed that indeed they were theoretically feasible! The same way the Michelson–Morley experiment showed in 1887, by the means of

<sup>3</sup>Laser Interferometer Gravitational-Wave Observatory.

interferometry, strong evidence against the existence of aether, the image of the black hole in Messier 87 is probably the definitive proof for their existence, after being predicted by Einstein's general relativity theory and after the Schwarzschild finding of a particular solution.

Getting back to contemporary times, on November 2, 2009 the SMOS satellite was launched with the objective of studying the exchange processes between the surface of the Earth and the atmosphere and hopefully help develop better models of weather and climate. It is a project from ESA's Living Planet Programme in which the CESBIO team led by Yann Kerr at CNES had a major role for the design of the satellite, including the image processing algorithms and the exploitation of data produced. This satellite uses interferometry to deduce the brightness temperature<sup>4</sup> of the  $H_2O$  molecules. This information is given as input to a final model that, along with other measurements<sup>5</sup> provides an estimate of soil moisture and ocean salinity. The LICEF<sup>6</sup> antennas of the satellite are mounted on a Y-shaped structure, each of the bars containing several antennas that receive radio-frequency signals centered at the *L-band* (from 1 to 2 GHz).

The MIRAS<sup>7</sup> sensor of SMOS is based on an interferometric technique known as *aperture synthesis*. The angular resolution of a telescope depends on its diameter, and certainly it is unfeasible to deploy a very large single antenna attached to the satellite. Instead, an array of several antennas is used and the angular resolution is obtained from the maximum distance between all pairs of antennas. Each pair of antennas gives one Fourier coefficient of the brightness temperature. The following figure shows the three SMOS arms and a segment of the MIRAS instrument.<sup>8</sup>



On the left, the deployed MIRAS instrument at ESTEC's Maxwell EMC chamber. On the right, a MIRAS arm segment with 6 LICEF antennas.

SMOS was planned as a five-years mission (thus, until 2014), then was extended to 2017, and actually nowadays in 2019 still keeps operating. However, even if the satellite has fuel enough to operate some more decades, the specialists reached the conclusion that a spatial resolution of about  $10 \text{ Km}^2/\text{pixel}$  was required in order to develop reliable models for the Earth's water cycle and climate. That is one of the main reasons that pushed the current studies for SMOS-HR, given that SMOS has a resolution of  $40 \text{ Km}^2/\text{pixel}$  at nadir.

In late 2017 we signed a research contract with CESBIO (to end in August 2020) to help design the new SMOS-HR satellite. This research is still ongoing work, but nevertheless the first results are presented in chapter 4. In particular we propose solutions to the large spatial folding present in SMOS<sup>9</sup> by

<sup>4</sup>The brightness temperature (not to be confused with the physical temperature) is a function that depends on the wavelength and characterizes the radiation emitted by a body.

<sup>5</sup>Such as the speed of wind, and the physical temperature, among others.

<sup>6</sup>Lightweight Cost-Effective Front-end.

<sup>7</sup>Microwave Imaging Radiometer using Aperture Synthesis.

<sup>8</sup>Source: SPIE Digital Library, <https://www.spiedigitallibrary.org/conference-proceedings-of-spie/10565/105650Z/Fibre-optics-in-the-SMOS-mission/10.1117/12.2309115.full?SS0=1>

<sup>9</sup>MIRAS undersamples the scene in the Fourier domain, thus causing spatial folding in the reconstruction.

exploiting the redundant multi-snapshot nature of the acquired data, the influence of the noise in the reconstruction, the possibility of using different configurations, including the irregular sampling or the double-sampling (the last proposed by Bernard Rougé). We are convinced that unfolding and denoising the acquired data at the moment the overlapping snapshots are acquired (rather than aggregating the redundant information at the end) will lead to more reliable models to estimate the final soil moisture and ocean salinity.

This research will be reinforced with the incorporation of Max Dunitz to the project, supervised by Jean-Michel Morel and I. At this moment, Max Dunitz is finishing his M2's<sup>10</sup> research project in which he is working on projecting onto the terrestrial surface the locations coming from the satellite's coordinates system. Max Dunitz will start a PhD on this subject on October 2019, also under the supervision of Morel and I. Two L3<sup>11</sup> students, Hugo Marsan and Clément Monnier, worked on the problem of spatial folding, with positive results. At this moment we work on developing a realistic SMOS-HR simulator, that will combine the simplified (but still a required proof of concept) simulations by Marsan and Monnier with the more realistic projections from Dunitz. At the same time, we keep working on improving the reconstructions using irregular sampling. To the best of our knowledge, it will be the first time that irregular sampling is proposed for an Earth-observing satellite, but it has been used in other ground telescopes. The future SKA instrument also uses irregular sampling techniques [3].

One problem that we needed to address in this project was that although we could develop configurations of antennas that performed very well, for engineering reasons it was not feasible to deploy some of them in space. While this is not a problem for ground telescopes (say, VLA, ALMA, or SKA) it brings additional constraints that need to be taken into account. Also, the onboard calibration of the antennas needs to be taken into account, given that a configuration that cannot be calibrated is useless in practice. Chapter 4 presents our proposed solutions to these two problems.

This research is, in my opinion, again a point in favor for the design of hybrid instruments which combine accurate acquisition systems with advanced signal processing algorithms. We proved that indeed with IASI it was possible, and we hope that we will convince ESA, CNES, and other involved agencies that this hybrid approach is also the way to go for SMOS-HR.

---

<sup>10</sup>In the French system this is a second-year research masters that allows to start a PhD.

<sup>11</sup>Third year of bachelor's degree in France.

# Chapter 3

## Full-spectrum denoising of high-SNR hyperspectral images

After the WHISPERS 2016 conference we extended significantly our denoising method of high-SNR hyperspectral images and we published our results in the Journal of Optical Society of America A (JOSA A) [4] in 2019. This chapter is a summary of this research.

### Abstract

The high spectral redundancy of hyper/ultraspectral Earth-observation satellite imaging raises three challenges: a) to design accurate noise estimation methods, b) to denoise images with very high SNR, and c) to secure unbiased denoising. We solve (a) by a new noise estimation (b) by a novel Bayesian algorithm exploiting spectral redundancy and spectral clustering, and (c) by accurate measurements of the interchannel correlation after denoising. We demonstrate the effectiveness of our method on two ultraspectral Earth instruments, IASI and IASI-NG, one flying and the other in project, and sketch the major resolution gain of future instruments entailed by such unbiased denoising.

### 3.1 Introduction

Hyperspectral (HSI) and Ultraspectral Imaging (USI) have numerous applications, such as the characterization of the abundance of components found in planetary exploration [5], the measurement of the thermodynamic, chemical and climate variables of the Earth atmosphere through satellite sounders [6] but also, in the case of HSI, medical applications such as tumor detection [7], inspection of food and agricultural products [8], among many others. To make the reading easier, in the text we will refer from now to "HSIs" for both HSIs and USIs, without loss of generality. Hyperspectral imaging is generally characterized by a low resolution spatial sampling and high resolution spectral sampling, hence requiring unmixing algorithms to augment the spatial resolution and characterize the detected chemical or atmospheric events [9].

The data acquired by a hyperspectral instrument is affected by internal random perturbations such as the dark currents inside the sensor, readout errors, and absorption of electromagnetic energy by the semiconductors of the device, as well as by external sources, particularly photon noise due to the intrinsic quantum nature of light. All of these random perturbations of the measurement are known as *noise* in the acquired HSI.

With the exception of photon noise, the set of the other noise sources can be modeled as homoscedastic additive white Gaussian noise, whereas photon noise depends on the temperature of the scene and the frequency of the channel being observed. However, photon noise can be well characterized by

Planck's law. Thus one can formulate a complete and realistic noise model, which will be discussed in Sec. 3.2. Nevertheless, reestimating the noise components on board remains necessary.

Our main focus here is to reconsider and reformulate the noise estimation and the image denoising problem for high Signal-to-Noise Ratio (SNR) satellite hyperspectral images (where the noise is mainly governed by Planck's law), by taking advantage of the high redundancy of the hyperspectrum. The latest advances in the design of the hyperspectral sensors allow one to measure very small variations of the signal in narrow spectral channels with a very low noise added by the electronics (i.e. dark, readout, shot, or electromagnetic interference noises). Therefore, modern instruments deliver high-SNR images. Yet, precisely, this high SNR and an even higher one are needed to evaluate (for example) the tiny variations of some diluted chemical components in the atmosphere.

We shall see that the observation of several thousands spectral channels on many data makes it possible to estimate the noise model from the data themselves. We shall also be in a position to check that the estimated noise matches the physical model.

The new method we present here is based on some of the ideas sketched in our conference paper [10]. We show Peak Signal-to-Noise Ratio (PSNR) and correlation results, before and after denoising. Actually we introduce a similar but more robust metric, the Median Signal to Noise Ratio (MSNR), Eq. (3.9). Our main goal is to reach a bias-free denoising method, which is defined as a method introducing no correlation in the signal. This is particularly important for applications to the detection of small variations in temperatures and densities of chemical components, where denoising can be followed by sophisticated detection and unmixing methods [11, 12].

The classic denoising tools that have been tried on HSIs in the literature comprise: wavelet thresholding, total variation, 3D patch-based methods, coupled with decorrelation tools. We now review the most effective ones.

The authors of [13] and [14] applied denoising to HSIs (70 to 200 channels), assuming low or very low SNR. In this low SNR context their algorithm could still be compared to classic patch-based algorithms such as Video Block Matching 3D (VBM3D) [15]. As we shall see for high SNR, HSIs with thousands of channels, patch-based methods are no longer competitive. They underperform when compared to algorithms exploiting the strong spectral redundancy.

The method proposed in [16] simulates uniform Gaussian noise and some "sparse noise" which is not related to Planck's law, and therefore not adapted to hyperspectral imaging. This method involving 2-D total variation along the spatial dimension and 1-D total variation along the spectral dimension may be used for multi-spectral images and low SNR. Again, this method is not relevant for high SNR hyperspectral images, because the spatial total variation forces local spatial correlations. Similarly the denoising method in [17] and [18] involves a minimization of the 2-D total variation along the spatial dimension and of the 1-D total variation along the spectral dimension. This is again adapted to HSIs with low SNR. The authors only consider a combination of Gaussian and impulse noise which does not correspond to the physics of hyperspectral images. A more up to date approach is presented in [19]. This paper builds spatial-spectral dictionaries of patches to exploit the sparsity across the spatial-spectral domain, the high correlation across spectra, and the non-local self-similarity over space. The denoising energy model which depends on three parameters is minimized by an iterative method. This model is clearly related to what we will propose, but again used in a different framework: hyperspectral rather than USIs, with low SNR. The proposed iterative method would be far too slow for our purposes. The dictionary approach can be accelerated as proposed in [20]. This paper describes an attempt at a fast greedy method based on orthogonal matching pursuit. This again is done in the context of multispectral images with low SNR, and is not adapted to the high SNR setup of ultraspectral imaging. A recent method based also on dictionary learning and decomposable nonlocal tensors (TDL) [21] is nowadays one of the methods that reach the best performance for low and moderate SNR HSIs. It exploits two well-known intrinsic characteristics of HSIs: the spatial nonlocal autosimilarity and the global correlation across the spectrum. We will also compare our results with TDL [21] and VBM3D [15] and show that even the best methods of the state

of the art that are most performing in the literature are not adapted to the very high SNR images that are obtained by the next generation of ultraspectral sensors, including that of the future Infrared Atmospheric Sounding Interferometer Next Generation (IASI-NG) satellite.

We shall refer to a method based on 2-D non-subsampled shearlet transform (NSST) [22] and fully constrained least-squares unmmixing (FCLSU) [23]. This method splits the channels of the HSI between low and high noisy sets according to their spectral correlation, and then NSST is applied at each channel. Finally, the low-noise bands are denoised by thresholding shearlet coefficients and high-noise channels with FCLSU.

The method proposed by Chen [24] seems to be another state of the art method and it contains several arguments that we shall reuse here. Similarly to the present work, the authors consider hyperspectral images with signal-to-noise ratio (such as 600 : 1). They decorrelate the image information of hyperspectral data cubes from the noise by using principal component analysis (PCA). They argue that the first PCA channels contain a majority of the total energy of the HSI and that removing noise in the low-energy PCA channels will not harm the fine features of the data cubes. A 2-D bivariate wavelet thresholding method is therefore used to remove the noise for low-energy PCA channels, while instead a 1-D dual-tree complex wavelet transform denoising method is used to remove the noise of the spectrum of each pixel of the data cube.

The results of our denoising method, Dual Bispectral Bayesian Denoising (DBBD) outperform state-of-the-art methods such as [23] and [24]. The DBBD results are presented in Sec. 3.4.

Our plan is as follows. In section 3.2 we describe the noise model. It is valid for IASI and the future satellite IASI-NG. In section 3.2.2 we describe the simulation data used for IASI-NG. Section 3.3 details our proposed method for hyperspectral images with high SNR and large spectrum. Section 3.4 details the evaluation of the method on simulated and real hyperspectral satellite images with high SNR and large spectrum. Section 3.5 is a conclusion on the perspectives open to satellite design by high performance unbiased denoising.

## 3.2 The noise model and its estimation

Assuming a black body, the emission of photons is related only to its temperature  $T$  [25] and it can be modeled by a Poisson distribution whose variance  $\sigma^2$  is equal to its mean  $\mu$ . By definition of the Signal to Noise Ratio, we have  $\text{SNR} = \frac{\mu}{\sigma} = \frac{\lambda}{\sqrt{\lambda}} = \sqrt{\sigma^2} = \sigma$ . Since the Noise Equivalent Differential Temperature (NEDT) is defined as the change in temperature that yields an SNR ratio of unity, we can write that the standard deviation of the noise is equal to the Noise Equivalent Delta Radiance (NEDL), which is expressed as [26, 27, 28]:

$$\text{NEDL}(\nu, T) = \text{NEDT}(\nu, T) \times \left| \frac{\partial L(\nu, T)}{\partial T} \right|,$$

where  $L(\nu, T)$  is the black body radiance and  $\nu$  the corresponding wavenumber. The values of NEDT are an intrinsic characteristic of the acquisition sensor and they can be tabulated according to frequency and temperature values. In practice one assumes that the temperature of the black body is constant and it is fixed to a certain reference, typically  $T = 280 \text{ K}$  (background-limited system).

According to Planck's law,

$$L(\nu, T) = \frac{c_1 \nu^3}{e^{c_2 \nu / T} - 1},$$

where  $c_1$  and  $c_2$  are the first and the second radiation constants. Thus,

$$\frac{\partial L(\nu, T)}{\partial T} = \frac{c_1 c_2 \nu^4}{T^2} \frac{e^{c_2 \nu / T}}{(e^{c_2 \nu / T} - 1)^2}.$$

Finally, the standard deviation of the noise can be obtained by the following closed expression [29]:

$$\sigma_{\text{noise}}(\nu, T) = \text{NEDT}(\nu, T) \times \frac{c_1 c_2 \nu^4}{T^2} \frac{e^{c_2 \nu / T}}{(e^{c_2 \nu / T} - 1)^2}. \quad (3.1)$$

The physical details behind these well-known formulations are out of the scope of this article, and we refer the reader to [29], as well as to [30, 31] for more information.

Assuming a (non-available) ideal noise-free HSI or USI  $\mathbf{G}$ , the noisy HSI  $\tilde{\mathbf{G}}$  is obtained as

$$\tilde{\mathbf{G}}_\nu = \mathbf{G}_\nu + \mathcal{N}_\nu, \quad (3.2)$$

where  $\mathcal{N}_\nu$  is a Poisson random variable whose parameter is the NEDL.

### 3.2.1 Noise estimation

Sec. 3.2 discussed how to obtain the standard deviation of the noise in a particular image in the hyperspectral cube corresponding to a wavenumber  $\nu$ . While this physical model is valid for a black body, there are situations where it might be useful to estimate the noise from the data instead of assuming directly the ideal black body scenario. First, the radiometric characterization is not quite realistic. An Earth-observing scene in space-flight might not correspond to the one obtained in pre-flight calibration, where the instrument is calibrated using a sphere under constant lighting simulating a black body. The scene observed during space calibration is likely to have spectral radiance characteristics which do not correspond exactly to the black body assumption [32].

Second, the signal in the sensor might be censored or even get saturated if it overflows its dynamic range, causing a nonlinearity of the dependence of the variance of the noise with respect to the intensity when the signal approaches saturation points. This leads to censoring and clipping effects [33].

Third, the instrument mounted on the satellite might be made of several sensors with spectral overlapping. For example, in the case of the IASI-NG, the instrument is made of four hyperspectral sensors (four bands) with spectral overlapping of  $\pm 40 \text{cm}^{-1}$ . Thus, the variance of the noise at the intersections is halved by averaging.

For these reasons, instead of assuming the black body model directly, it may be recommended to estimate the noise model from the noisy hyperspectral image itself. We propose in Algo. 1 a simple procedure to estimate the noise from HSIs. Its adequacy will be verified on both simulated and real data.

The algorithm has two steps. First (lines 4–12) it finds for every channel  $i$  the channel  $j \neq i$  that gives the maximum correlation, equalizes the data in both channels to have the same mean, and computes a raw estimation of the noise variance from their difference. This empirical variance (computed on line 11) is likely to contain outliers, since the most correlated channel with respect to a given one can have a larger difference than the one caused by the small noise level.

The second step (lines 13–19) removes the outliers. The assumption is that the noise levels inside a short frequential window should not be very different, since they are linked to similar frequencies. Thus, the minimum value of the noise empirical standard deviation should be close to the real noise value. The obvious drawback of using the min operator is that if most of the values are close to the real noise level, then the obtained result might get a small underestimation bias, as can be observed in the wavenumbers from 150000 to 200000 in Fig. 3.1. This figure shows the noise estimation in a simulated IASI-NG with simulated added noise with the physical model, Eq. (3.1). This means that after the NEDT of the sensor had been measured in the laboratory and tabulated, we applied Planck's law to obtain the NEDL and therefore the frequency-dependent standard deviation (STD) of the noise. In red, one can see the standard deviation of the noise according to the wavenumber given by the physical model. In green, our estimation. Both curves match, and their accuracy is more than sufficient for denoising purposes.

---

**Algorithm 1** Estimate the noise level function of an HSI
 

---

```

1: HSI noise estimation
   Input  $\tilde{\mathbf{G}}$ : acquired noisy HSI
   Input  $w = 100$ : window size
   Output  $\hat{\sigma}$ : noise estimation

2:  $R \leftarrow \text{corrcoef}(\tilde{\mathbf{G}})$  ▷ Pearson cross-correlation coeffs.
3:  $\hat{\sigma} \leftarrow \text{zeros}(R.\text{shape}[0])$  ▷ First dimension: frequencies
4: # Raw noise estimation
5: for  $i \in \text{range}(R.\text{shape}[0])$  do
6:    $R[i, i] \leftarrow -\infty$  ▷ Exclude trivial diagonal maxs
7:    $j \leftarrow \text{argmax } R[i, :]$  ▷ Get most correlated frequency
8:    $k \leftarrow \text{mean}(\tilde{\mathbf{G}}_i) / \text{mean}(\tilde{\mathbf{G}}_j)$ 
9:    $\tilde{\mathbf{G}}_j \leftarrow \tilde{\mathbf{G}}_j \times k$  ▷ Mean equalization
10:   $\hat{\sigma}[i] \leftarrow \|\tilde{\mathbf{G}}_i - \tilde{\mathbf{G}}_j\| / \sqrt{2}$  ▷ Raw noise estimation
11: end for
12: # Final filtered estimation
13: for  $i \in \text{range}(0, \text{len}(\hat{\sigma}), w)$  do ▷ Non-overlapping window
14:    $j \leftarrow i + w - 1$ 
15:   if  $j \geq \text{len}(\hat{\sigma})$  then ▷ Take care of array boundary
16:      $j \leftarrow \text{len}(\hat{\sigma}) - 1$ 
17:   end if
18:    $\hat{\sigma}[i : j + 1] \leftarrow \min(\hat{\sigma}[i : j + 1])$  ▷ Assign min to window
19: end for
20: return  $\hat{\sigma}$ 

```

---

Fig. 3.2 shows a noise estimation of two real IASI USIs. In that case the noise was not simulated and added: It is the real noise present in the acquired USI. Here again the estimations match well the noise predicted by the physical model.

This procedure is different and actually simpler than the noise estimator proposed in [34], which is, as the authors claim, valid for extremely noisy hyperspectral applications. Here we are instead working on high SNR hyperspectral data.

In order to confirm that the noise estimation procedure was accurate, we performed two kinds of tests: one with simulated data (see Sec. 3.2.2 for the details on the data model) and added noise according to the physical model presented in Sec. 3.2, as well estimating the noise in real HSIs and comparing the obtained result with the physical noise model.

The noise estimation algorithm is based on the strong redundancy of HSIs, in particular on the large cross-correlations along the different spectral channels. Fig. 3.3 shows an example of the band B2 of an USI of IASI-NG. Note that we use the correlation and not the L2 or similar norms to compare the data along different frequencies since the gain (say, the atmospheric absorption) at different channels can be different. However, the Pearson cross-correlation of the data is robust to these global changes.

### 3.2.2 Our real and simulated data

We performed experiments on both simulated and real data. The simulated data model was obtained by the  $\alpha$ -IASI- $\beta$  radiative transfer model, originally developed for IASI by the European Space Agency EUMETSAT (European Organisation for the Exploitation of Meteorological Satellites). Yet this model is realistic for any nadir viewing satellite, aerial sensor, or ground observation [35].

Fig. 3.4 shows the simulated scene radiance at two different frequencies of the first band of USI #0 in our dataset. Fig. 3.5 corresponds however to real data obtained from an actual IASI instrument.

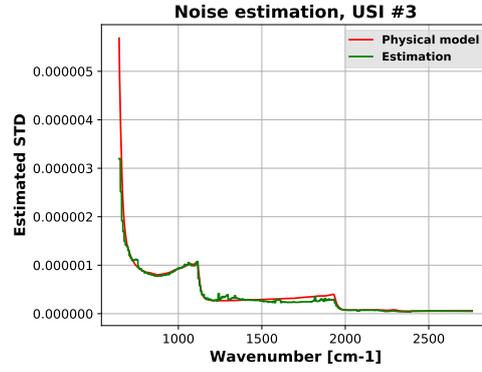


Figure 3.1: Noise estimation with simulated data and simulated added noise according to the physical model of the IASI-NG USI #3. In red, the standard deviation of the noise according to the wavenumber given by the physical model. In green, our estimation. Both curves match, and their accuracy is more than sufficient for denoising purposes. The noise estimation was obtained step-wise for each wavenumber.



Figure 3.2: Noise estimation in the real IASI USIs #7 and #9. The noise was estimated from the original USI data. Similar results were obtained with the other real IASI USIs. In red, the standard deviation according to the wavenumber given by the physical model, Eq. (3.1). In green, our estimation. The noise estimation was obtained step-wise for each wavenumber.

### 3.3 The proposed denoising method

This section describes our proposed *Dual Bispectral Bayesian Denoising* (DBBD) method. Its pseudocode is given in Algo. 2. Most of the frequencies of an HSI are highly correlated with the rest (see Fig. 3.3) with the exception of a few of them, and therefore also the acquired images (Fig. 3.4). A fundamental denoising principle is to exploit redundancy of the data, and particularly its self-similarity. The general principle invented by Lee [36] is to group pixels having a similar model. Groups of similar pixels are detected by comparing their position in the image domain and their values. In a recent image denoising method, the same principle was expanded by grouping pixels which have similar neighborhoods [37], before denoising them together. The reason for comparing patches instead of pixels is to ensure that the grouped pixels really obey the same model. In the case of hyperspectral images, the high number of values available at each pixel makes it unnecessary and actually counter-productive to use patches. Each pixel is indeed extremely well characterized by its spectral signature. We even found that the self similarity principle should be applied first in the frequency domain. This means that we treat frequencies as dual pixels and group them according to their similarities, before going back to the image domain and grouping the pixels in clusters by their similarity. Thus, we build frequency clusters for each HSI by grouping highly correlated frequencies. We perform this using the K-means clustering algorithm.

Each frequency cluster  $\Lambda_i$  therefore contains highly-correlated frequencies. The data at each of the frequencies is divided by the STD of the noise to equalize all frequency STDs to 1. The next step is to reduce the frequency domain dimensionality to  $N = 20$  dimensions by PCA. We verified empir-

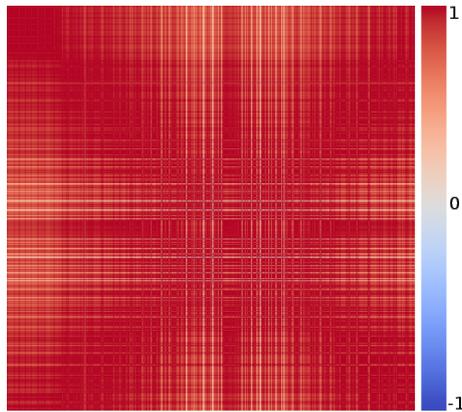
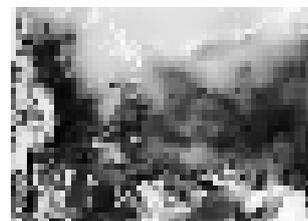


Figure 3.3: The normalized Pearson cross-correlation matrix for band B2 (from  $115000.0 \text{ m}^{-1}$  to  $195987.5 \text{ m}^{-1}$ ) in USI #1. The red color represents a high correlation, while white means low (*coolwarm* color map). Most of the channels are correlated with other many. Only a few frequencies are uncorrelated because of the presence of gases in the atmosphere at those particular frequencies.



(a)  $88362.5 \text{ m}^{-1}$



(b)  $96500.0 \text{ m}^{-1}$

Figure 3.4: Simulation of the images acquired by the IASI-NG instrument at different frequencies (USI #0, band B1). The intensity of each pixel represents the energy measured in  $\text{W}/\text{m}^2/\text{sr}/\text{m}^{-1}$  units.

ically that 20 PCs are enough to represent reliably the IASI-NG USIs (see Fig. 3.6). This is done in conformity with the literature which recommends a PCA reduction for an efficient representation of observations from high-resolution infrared sounders [38]. As pointed out in [39], one of the main advantages of the spectral domain over the spatial domain is the computation of low-noise basis vectors even from very noisy USIs. As these authors remark,

Spectral distributions can be very compactly represented and there is an overwhelmingly large number of spectra in a typical USI. As we also expect noise to occur randomly, the impact of noise on the greatest directions of variance in the large set of spectra in an USI should be minimal. Thus PCA can be used to obtain low-noise top basis vectors even when noise is high.

We denote by  $\tilde{\mathbf{A}}$  the pixels in  $\Lambda_i$  after dimensionality reduction.

For each cluster  $\Lambda_i$  we apply optimal Bayesian denoising [40] to each of the pixels of the HSI after PCA dimensionality reduction (only the first  $N$  principal components -PCs- are kept), where  $\tilde{P}$  is the noisy pixel and  $P$  the denoised pixel:

$$P = \bar{P} + [\mathbf{C}_{\tilde{P}} - \mathbf{C}_n] \mathbf{C}_{\tilde{P}}^{-1} (\tilde{P} - \bar{P}). \quad (3.3)$$

In this equation,  $\mathbf{C}_{\tilde{P}}$  is the empirical covariance matrix of the pixels most similar to the reference pixel in the cluster. We use the Pearson cross-correlation coefficient as the similarity criterion. The empirical covariance matrix  $\mathbf{C}_n$  of the noise after PCA transformation is required too. It needs to be computed only for the wavenumbers in  $\Lambda_i$ . Sec. 3.3.1 gives the details on how to compute  $\mathbf{C}_n$ .

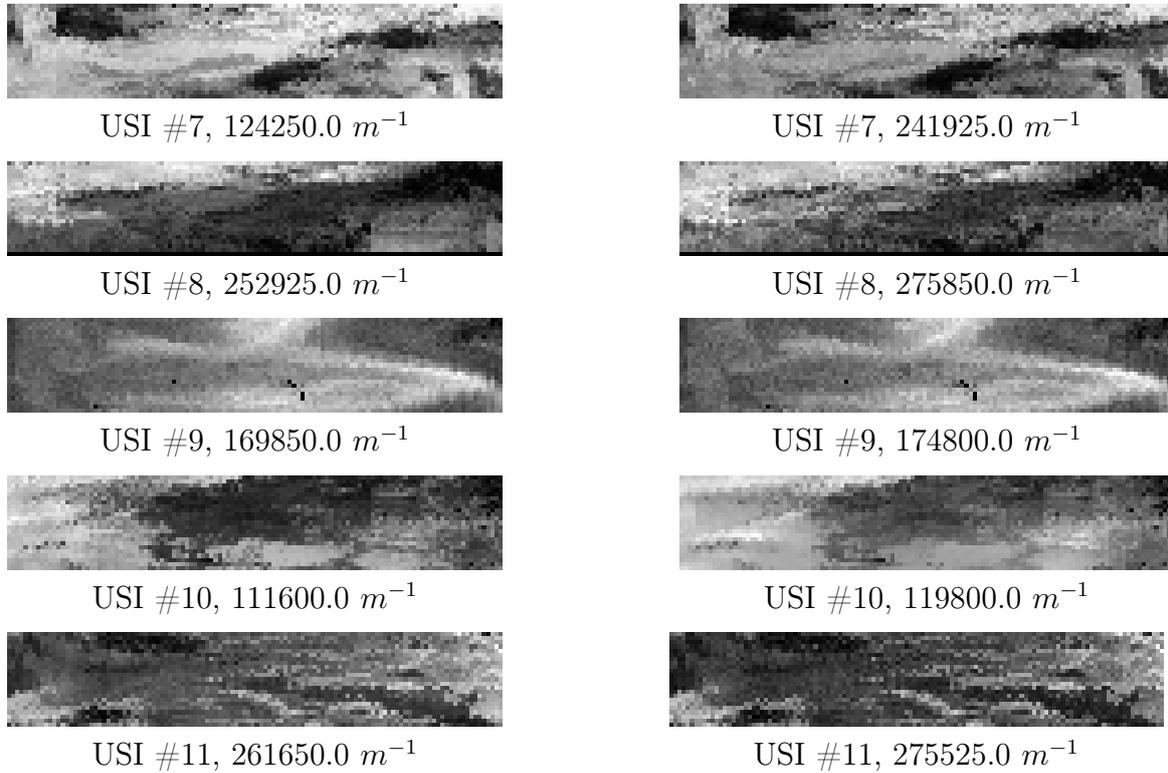


Figure 3.5: Two different frequencies of the USIs #7–#11 in our database. This corresponds to real IASI acquisitions, each of them of  $120 \times 23$  pixels.

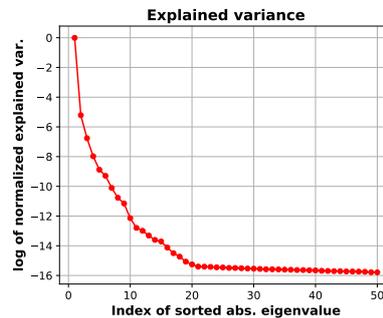


Figure 3.6: Decay of the normalized explained variance according to the sorted eigenvalues, in log scale. There is no significant improvement if more than 20 PCs are used. This plot was obtained with data from USI #0. Similar results are obtained with the others.

For each cluster  $\Lambda_i$ , we apply the optimal Bayesian estimator. This estimator is optimal under two assumptions:

- The frequency-dependent noise model is Gaussian and entirely characterized by its covariance matrix  $C_n$ ;
- The pixel model is equally frequency-dependent Gaussian and characterized by its covariance matrix  $C_{\tilde{p}}$ .

Assumption a) is realistic for spectrometers [26, 27]. Indeed, it is a classic result in imaging that the Poisson distribution can be well approximated by a Gaussian when the number of photons is large enough. Assumption b) is all the more true if our double grouping procedure, clustering first similar frequencies and second similar pixels is efficient. Very similar such pixels are only different by small variations caused by sampling and random local variations of all chemical components that can be accounted for by a Gaussian model.

At this point, the pixel has been partially denoised for the wavenumbers contained in the cluster, and only for the first  $N$  PCs. The next step is to denoise the rest of the PCs, which contains a very noisy signal but still some useful information. Thus, the rest of the spectrum encoded at those last PCs can be denoised, leading to a *full-spectrum* technique.

Here we apply the dual method of Chen [24] to the data in the cluster, which accounts for applying a dual-tree complex wavelet transform (DT-CWT) to the rest of the PCs and denoising them with Donoho's universal threshold, defined as

$$t = \sigma \sqrt{2 \log(N_x N_y)}, \quad (3.4)$$

where  $\sigma = 1$  is the STD of the noise (it is one because we first divided the noisy signal by the STD of the noise), and  $(N_x, N_y)$  is the size of the 2D image.

If we call  $c_{i,j,l}$  the wavelet coefficient at pixel  $[i, j]$  and level  $l$ , the denoised coefficient is obtained by soft thresholding as

$$\hat{c}_{i,j,l} = c_{i,j,l} (1 - t^2/|d_{i,j,l}|^2)^+, \quad (3.5)$$

where  $|d_{i,j,l}|^2 = (|c_{i-1,j,l}|^2 + |c_{i,j,l}|^2 + |c_{i+1,j,l}|^2)/3$  is the average of the absolute squared coefficients in a small neighborhood.

Each cluster contains only a part of the total set of wavenumbers. Thus each pixel is denoised partly at each iteration. After all iterations, the pixel is completely denoised. Finally, the denoised pixels are projected back (the data is expressed in the same axes as before the PCA rotation) and the complete denoised USI is obtained. At the end, all the PCs (the first  $N$  with optimal Bayesian denoising and the rest with DT-CWT thresholding) and all the wavenumbers have been denoised.

### 3.3.1 Estimation of $C_n$

Eq. (3.3) requires the empirical covariance matrix of the noise  $C_n$  on the PCA axes. This matrix is not available directly and must be estimated from the noisy data itself.

The rationale behind the estimation procedure presented here is that, after PCA, the first  $N$  PCs (in decreasing absolute eigenvalue) are dominated by the underlying signal, whereas the  $N$ -th and following PCs are mainly the contribution of the noise. We shall call  $M$  the number of wavenumbers in the current noisy cluster and  $C_n$  the empirical covariance matrix of the noise being estimated. For the PCs from  $N$  to  $M$  the signal is dominated by the noise and therefore the variance computed at these PCs corresponds mainly to the variance of the noise after PCA rotation [41].

For the first  $N$  PCs the underlying signal dominates and it is not possible to directly measure the variance of the noise. In order to estimate the noise, we make the following assumptions:

1. The PCA concentrated most of the variance of the noisy cluster  $\Lambda_i$  in the first PCs
2. Given that  $\Lambda_i$  was normalized in order that the noise had STD=1 along any wavenumber therefore (as a consequence of the previous point)  $C_n[0, 0] \approx 1$ .

The variance of the noise from the  $N$ -th PC is well estimated and therefore one can approximate the variance of the noise from the first PC to the  $N$ -th. We chose to use a simplest estimator (a linear function) given that we do not assume any priors, but it could be improved if additional information is available. For example, from the statistics of many observations of real IASI-NG images, but for the moment these data is not available. However, the simple linear estimator had a good performance, as shown in Sec. 3.4.

Finally, we obtain the estimated empirical covariance matrix of the noise as:

$$\mathbf{C}_n[i, j] = \begin{cases} 1 + \frac{i \times \text{var}(\Lambda_i[N, :]) - 1}{N} & \text{if } i = j \text{ and } i < N, \\ \text{var}(\Lambda_i[i, :]) & \text{if } i = j \text{ and } i \geq N, \\ 0 & \text{otherwise.} \end{cases} \quad (3.6)$$

DBBD performs a denoising which is adapted to the data itself and groups similar pixels in clusters. This covariance matrix is obtained for each of the clusters, given that the PCA rotation is performed at each of them.

Fig. 3.7 shows our estimated noise variance after PCA rotation, in one of the clusters. Both the estimation and the ground-truth curves match sufficiently to estimate an empirical covariance matrix of the noise,  $\mathbf{C}_n$ .

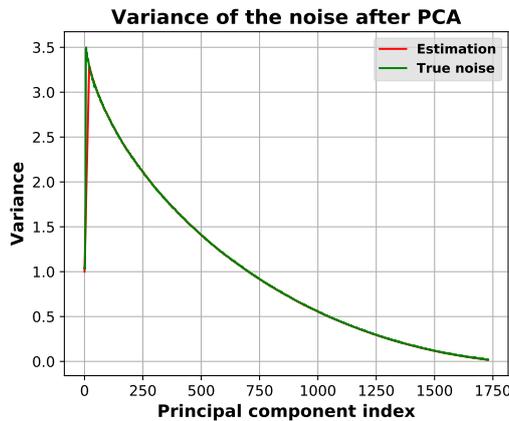


Figure 3.7: Estimated noise variance after PCA rotation for USI #6, band #4, cluster #2 (similar results are obtained with the other HSIs, bands, and clusters). In red: our estimation. In green: the true value. The curves match sufficiently to estimate an empirical covariance matrix of the noise,  $\mathbf{C}_n$ .

### 3.3.2 The conservative Lee correction

Eq. (3.3) performs totally blind denoising of each pixel of the input USI, for the first  $N$  PCs. It assumes that the  $K$  most similar hyperspectral pixels used to compute the covariance matrix  $\mathbf{C}_{\bar{P}}$  are samples of a Gaussian vector, their differences being due to noise and to a combination of feeble independent factors. The other PCs (assumed to contain mainly noise but also minimal detail information) are denoised by soft thresholding.

If the denoising process is perfect, the variance of the removed signal should match the variance of the noise. If its variance is larger than the variance given by the physical model, this means that the denoising is not only removing noise, but also removing details. Thus, in order to improve the denoised signal we compute a weighted sum of the denoised and of the noisy USI to put back details from the noisy image when the denoising has been detected as too *aggressive*. This technique is inspired by Lee's local filtering [36, 42], which estimates the true value of a pixel with a weighted sum of the noisy pixel and the average of other noisy pixels in a small neighborhood.

The weighted sum for a given wavenumber  $j$  is given by:

$$\hat{\mathbf{G}}'_j = \alpha \hat{\mathbf{G}}_j + (1 - \alpha) \tilde{\mathbf{G}}_j. \quad (3.7)$$

The weighting factor is obtained by solving a simple equation which sets the variance of the removed signal to the variance given by the model. The unique solution is

$$\alpha_j^2 = \frac{\sigma_{\mathcal{N},j}^2}{\text{var}(\tilde{\mathbf{G}}_j)} + \text{var}(\hat{\mathbf{G}}_j) - 2 \text{cov}(\tilde{\mathbf{G}}_j, \hat{\mathbf{G}}_j). \quad (3.8)$$

Our final denoising algorithm, DBBD, is summarized in Algo. 2.

---

**Algorithm 2** DBBD denoising of an HSI

---

```

1: # Input/output:
   Input  $\tilde{\mathbf{G}}$ : acquired noisy HSI, of size  $(N_x, N_y, N_z)$ 
   Input  $\sigma_n$ : standard deviations of the noise (wavenumber dep.)
   Input  $Q = 3$ : number of spectral clusters
   Input  $K = 400$ : number of similar pixels
   Input  $N = 20$ : number of PCA PCs kept
   Output  $\hat{\mathbf{G}}'$ : denoised HSI

2:  $\hat{\mathbf{G}} \leftarrow \text{zeros}(\tilde{\mathbf{G}}.\text{shape})$  ▷ Placeholder
3: Clusterize in  $Q$  spectral groups  $\Lambda_i, i \in [1, Q]$  ▷ K-means for cosine distance
4: for each cluster  $\Lambda_i, i \in [1, Q]$  do
5:   Noise normalization: divide each coordinate of each sample in  $\Lambda_i$  by its STD taken from  $\sigma_n$ 
6:    $\mathbf{Z} \leftarrow \text{PCA}(\Lambda_i)$  ▷ Matrix of eigenvectors of the PCA of  $\Lambda_i$ , in decreasing abs. eigenvalue order
7:    $\text{sigmas\_pca} \leftarrow \text{STD}(\Lambda_i[:])$  ▷ STD of each wavenumber after PCA: Eq. (3.6).
8:   # Process the first PC's (from 0 to  $N - 1$ ):
9:    $\tilde{\mathbf{A}} \leftarrow \tilde{\mathbf{A}}[0 : N]$  ▷ First  $N$  PCs
10:   $\mathbf{C}_n[i, j] \leftarrow \text{sigmas\_pca}[i]^2$  if  $i = j$  (0 otherwise) ▷ Get noise covariance matrix
11:  for each noisy pixel  $\tilde{P}$  do
12:     $\mathbf{V} \leftarrow$  set of the  $K$  most correlated pixels with  $\tilde{P}$ 
13:     $\bar{P} \leftarrow \text{mean}(\mathbf{V})$  ▷ Compute their empirical mean and
14:     $\mathbf{C}_{\tilde{P}} \leftarrow \text{cov}(\mathbf{V})$  ▷ covariance matrix
15:     $P \leftarrow \bar{P} + [\mathbf{C}_{\tilde{P}} - \mathbf{C}_n] \mathbf{C}_{\tilde{P}}^{-1} (\tilde{P} - \bar{P})$  ▷ Eq. (3.3)
16:     $\hat{\mathbf{A}} \leftarrow P$  ▷ Append denoised  $P$ 
17:  end for
18:   $\mathbf{Z}[0 : N] \leftarrow \tilde{\mathbf{A}}$  ▷ Store Bayesian denoised
19:  # Process the remaining PC's (from  $N$  to end):
20:   $\mathbf{D} \leftarrow \text{DTCWT}(\tilde{\mathbf{A}}[N :])$  ▷ DT-CWT of the remaining PCs
21:   $\mathbf{M}^2 \leftarrow (|\mathbf{D}_{i-1,j,l}^2| + |\mathbf{D}_{i,j,l}^2| + |\mathbf{D}_{i+1,j,l}^2|)/3$ 
22:   $\mathbf{T}[:] \leftarrow \text{sigmas\_pca}[:] \sqrt{2 \log N_x N_y}$  ▷ Universal threshold
23:   $\mathbf{Z}[N :] \leftarrow \mathbf{Z}[N :](1 - \mathbf{T}[:]^2 / \mathbf{M}^2[:])$  ▷ Pointwise soft thresholding
24:   $\hat{\mathbf{G}}(\Lambda_i) \leftarrow \text{PCA}^{-1}(\mathbf{Z})$  ▷ Invert PCA rotation
25:   $\hat{\mathbf{G}}(\Lambda_i) \leftarrow \hat{\mathbf{G}}(\Lambda_i) \times \sigma_n[\text{idx}(\Lambda_i)]$  ▷ Invert STD normalization
26: end for
27: # Apply conservative variance correction:
28: for  $j \in [0, \text{shape}(\hat{\mathbf{G}})[0] - 1]$  do ▷ Loop over wavenumbers
29:    $\alpha^2 \leftarrow \frac{\sigma_n^2[j]}{\text{var}(\tilde{\mathbf{G}}[j,:])} + \text{var}(\hat{\mathbf{G}}[j,:]) - 2 \text{cov}(\tilde{\mathbf{G}}[j,:], \hat{\mathbf{G}}[j,:])$  ▷ Eq. (3.8)
30:    $\hat{\mathbf{G}}'[j,:] \leftarrow \alpha \hat{\mathbf{G}}[j,:] + (1 - \alpha) \tilde{\mathbf{G}}[j,:]$  ▷ Weighted sum, Eq. (3.7)
31: end for
32: return  $\hat{\mathbf{G}}'$  ▷ Final denoised HSI

```

---

### 3.3.3 Discussion on the choice of the hyperparameters

DBBD has three hyperparameters: the number of spectral clusters  $Q$ , the number of similar pixels  $K$ , and  $N$  the number of PCA principal components kept.

These hyperparameters are fixed and depend on the application. For IASI-NG we found that the optimal values are  $Q = 3$ ,  $K = 400$ , and  $N = 20$ . The actual values depend on the characteristics of the hyperspectral image, including the SNR, the size of the image, and the number of channels per

pixel. The number of PCA components which is kept ( $N$ ) can be obtained from many observations of different HSIs and choosing the value for which adding more PCs has no significant influence on the explained variance. We chose  $N = 20$  from the experiment shown in Fig. 3.6. The number of spectral clusters depends on the number of channels. A low  $Q$  will create clusters whose pixels are not similar enough, and conversely a large  $Q$  will produce clusters with not enough samples. The exact value for  $Q$  depends again on the application and the number of channels of the HSI. We fixed  $Q = 3$  which produces 1410 channels/cluster. The number of similar pixels  $N$  that will be used to compute the covariance matrix of the noisy pixels depends directly on the number of pixels of the sensor and the characteristics of the image. It should be large enough to allow computing the inverse of  $\mathbf{C}_{\tilde{p}}$ . If  $K$  is too large  $\mathbf{C}_{\tilde{p}}$  will be computed with pixels which have a different spectral signature and the algorithm will fail to obtain a reliable model of the noisy. For IASI-NG we found that  $K = 400$  is a good compromise. Table 3.1 shows the MSNRs obtained when varying the hyperparameters of the algorithm for HSI #2. *Optimal* is the default case of IASI-NG with  $Q, N, K = \{3, 20, 400\}$ , case #1 with  $Q, N, K = \{8, 20, 400\}$ , case #2 with  $Q, N, K = \{3, 7, 400\}$ , and case #3 with  $Q, N, K = \{3, 20, 150\}$ . Deviations from the optimal hyperparameters cause a slight decrease of the performance. For other type of sensors the values for  $Q, K$ , and  $N$  need to be re-calibrated from the observations.

Optimal	Case #1	Case #2	Case #3
<b>63.44</b>	62.39	62.08	62.84

Table 3.1: MSNRs obtained when varying the hyperparameters of the algorithm for HSI #2. *Optimal* is the default case of IASI-NG with  $Q, N, K = \{3, 20, 400\}$ , case #1 with  $Q, N, K = \{8, 20, 400\}$ , case #2 with  $Q, N, K = \{3, 7, 400\}$ , and case #3 with  $Q, N, K = \{3, 20, 150\}$ . Deviations from the optimal hyperparameters cause a slight decrease of the performance.

## 3.4 Experimental validation of DBBD

Our denoising method was tested and validated on both simulated and real USIs.

### 3.4.1 Evaluation with simulated USIs

The reason to evaluate the method in simulations is that they allow to create an ideal USI where the physically correct noise model is respected. Thus, a reliable ground-truth (GT) is available. This yields a quantitative exact measurement of the performance of the method. To this aim we used simulated data of the IASI-NG satellite project [43, 6, 44]. The location of the seven simulated IASI-NG USIs is given in Fig. 3.8

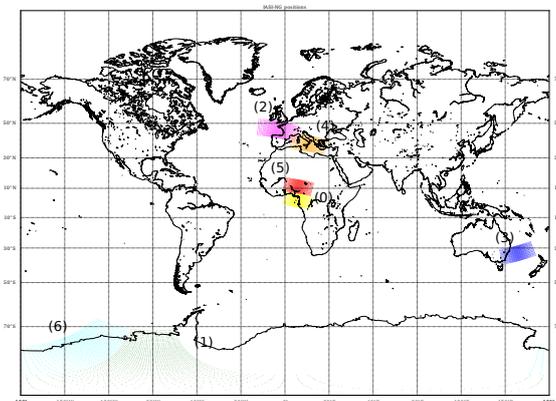


Figure 3.8: Location of the seven IASI-NG USIs.

The first performance test measures the PSNR of the denoised signal with respect to the noise-free USI. In order to avoid the effect of outlier radiance values, the median of the signal should be pre-

ferred to its maximum for the PSNR definition. This leads to using the MSNR between an USI  $\mathbf{T}$  and the reference noise-free USI  $\mathbf{G}$  at wavenumber  $j$  defined by

$$\text{MSNR}_j(\mathbf{T}, \mathbf{G}) = 10 \log_{10} \left[ \frac{\text{median}(\mathbf{T}_j)^2}{\text{MSE}(\mathbf{T}_j, \mathbf{G}_j)} \right], \quad (3.9)$$

where MSE is the Mean Squared Error.

Fig. 3.9 plots the respective MSNRs of the noisy USI, the DBBD result, dimensionality reduction by PCA band by band, and Chen's method [24]. We display results for USIs #2, #3, #4, and #5. Similar results were obtained for all simulated USIs. We used the following parameters in the comparison:  $N = 20$  in PCAbands,  $N = 20$  PCs and 4 DT-CWT levels in Chen. For VBM3D, the parameters are  $T_{2D}=5$ ,  $T_{3D}=7$ ,  $N=8$ ,  $N_f=4$ ,  $N_s=7$ ,  $N_{pr}=5$ ,  $N_b=2$ ,  $k=8$ ,  $kt=1$ ,  $p=6$ ,  $d=0.000754$ ,  $\lambda_{3D}=2.700000$ , and  $\tau=0.046136$  (first step of the algorithm) and  $T_{2D}=4$ ,  $T_{3D}=7$ ,  $N=8$ ,  $N_f=4$ ,  $N_s=7$ ,  $N_{pr}=5$ ,  $N_b=2$ ,  $k=7$ ,  $kt=1$ ,  $p=4$ ,  $d=0.000138$ ,  $\lambda_{3D}=0$ ,  $\tau=0.023068$  (second step). We refer the reader to [15, 45] for more details on the parameters. For TDL, the original source code sets the optimal values according to the level of noise. The IASI-NG USIs have 16920 channels and 2240 pixels/channel. VBM3D [15] improves slightly the noisy HSI, Chen's method [24] performs significantly well and beats the PCA band-by-band. DBBD (ours) outperforms all methods. TDL [21] has the worst performance, with a performance below the noisy HSI, confirming that TDL is not adapted to very-high SNR frequency-dependent HSIs.

Given that neither TDL nor VBM3D can deal with frequency-dependent noise, we multiplied each channel by the proper factor in order to equalize the STD of the noise along all the spectrum, as required by these methods. Since exact noise level is known for each wavenumber (estimated noise (Sec. 3.2.1), simulated (Sec. 3.4.1), or tabulated empirically from the NEDT of the sensor (Eq. (3.1))), it is straightforward to perform this STD equalization.

The observed MSNR increase of the PCA result with respect to the noisy USI shows that PCA separates efficiently noise from the signal. Since the USIs were simulated, we dispose of an exact GT. It is therefore possible to examine the portion of signal that the denoising procedure actually removed. We express this in terms of *removed signal* and not *removed noise* because unless the denoising algorithm is *perfect*, it will remove some details along with the noise, as well as introducing some bias in the result.

Since we know in advance that the noise is not correlated across different spectral channels, we can measure the Pearson auto-correlation of the removed signal ( $\hat{\mathbf{G}} - \mathbf{G}$ ), and compute histograms of correlation values (with the exception of the diagonal of the correlation matrix). Once the trivial ones at the diagonal have been excluded, two parameters of the histogram distribution need to be analyzed: its mean and its sharpness. Ideally, the distribution of correlation values should be a very sharp peak centered at zero (implying that there is no cross-correlation with other frequencies). Fig. 3.10 shows the histograms of the three USIs analyzed. As can be observed on the left, the cross-correlations of the signal removed by DBBD are centered at zero and their distribution is sharp and symmetric, which means that the method is not introducing artificial correlation in the denoised signal.

Table 3.2 compares the mean MSNR for the seven simulated IASI-NG USIs with respect to VBM3D [15], DBBD, Chen's method [24], PCA band by band, and TDL [21] and it shows that DBBD outperforms all other methods. Table 3.3 shows the evaluation with the SSIM [46] metric, where again DBBD outperforms the compared methods.

Table 3.4 gives our denoising factors. It is the factor by which the standard deviation of the noise has been divided for each of the simulated USIs. Reducing the standard deviation of the noise by the factor in (a) is theoretically equivalent to reducing the area of the sensor by the square of this factor while keeping the same SNR, or increasing the resolution of the sensor while keeping the same area. Column (c) gives the equivalent reduction in the area of the sensor.

USI #	Noisy	VBM3D [15]	DBBD	Chen [24]	PCA bands	TDL [21]
#0	41.76	44.96	<b>59.39</b>	58.13	57.91	40.46
#1	33.64	41.00	<b>52.27</b>	50.38	50.25	33.31
#2	45.89	51.55	<b>63.44</b>	62.04	60.65	44.91
#3	44.83	49.88	<b>63.22</b>	61.84	60.82	41.92
#4	44.51	50.25	<b>62.46</b>	61.09	60.12	39.91
#5	47.63	53.69	<b>65.08</b>	63.69	62.35	47.47
#6	40.06	49.39	<b>58.50</b>	56.91	56.65	39.88
Mean	42.62	48.39	<b>60.62</b>	59.15	58.30	41.12

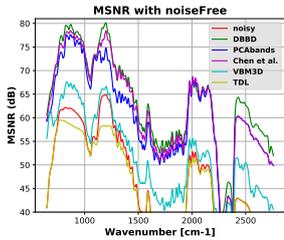
Table 3.2: Mean MSNR for the seven simulated IASI-NG USIs for VBM3D, DBBD, Chen’s method, PCA band by band, and TDL. The column labeled as *Noisy* corresponds to the MSNR when no denoising is applied. The fact that TDL shows a MSNR lower than the noise itself must be understood as this method not being adapted to these kind of high-SNR frequency-dependent noise HSIs, despite of being one of the best methods available.

USI #	Noisy	VBM3D [15]	DBBD	Chen [24]	PCA bands	TDL [21]
#0	22.73	23.47	<b>26.27</b>	26.12	25.75	22.48
#1	22.73	24.08	<b>26.19</b>	26.16	26.02	22.73
#2	22.73	23.85	<b>25.98</b>	25.81	24.92	22.27
#3	22.73	23.54	<b>26.06</b>	25.91	25.18	20.97
#4	22.73	23.75	<b>25.95</b>	25.83	24.93	20.94
#5	22.73	23.95	<b>26.04</b>	25.83	24.87	22.65
#6	22.73	24.14	<b>26.12</b>	26.11	25.89	22.72
Mean	22.73	23.80	<b>26.08</b>	25.96	25.26	21.80

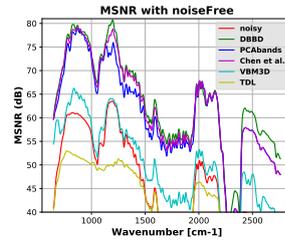
Table 3.3:  $|\log(1 - \text{SSIM})|$  for the seven simulated IASI-NG USIs for VBM3D, DBBD, Chen, PCA band by band, and TDL. *Noisy* refers to the case when no denoising is applied. The evaluation of TDL shows again that certain methods (including VBM3D) are not adapted to high-SNR frequency-dependent HSIs, even if they can be considered the best methods for low and medium SNR HSIs.

USI #	(a)	(b)	(c)	(d)
#0	7.492	56.124	-5.513-e5	0.021
#1	7.331	53.740	-5.346-e5	0.021
#2	6.663	44.400	-5.365-e5	0.021
#3	6.966	48.530	-5.442-e5	0.021
#4	6.406	41.037	-5.429-e5	0.021
#5	6.778	45.943	-5.207-e5	0.021
#6	7.125	50.767	-5.318-e5	0.021

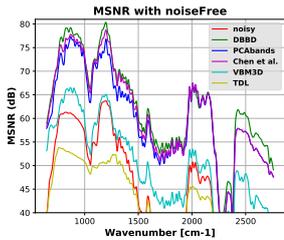
Table 3.4: (a) Standard deviation reduction factor after denoising with DBBD, (b) the associated surface reduction for the IASI-NG simulated USIs, (c) mean and (d) STD of the Pearson auto-correlation of  $\tilde{\mathbf{G}} - \hat{\mathbf{G}}'_v$  (the removed signal) excluding the diagonal. The noise is simulated according to the tabulated NEDT of the sensor and applying Eq. (3.1) to obtain its realistic frequency-dependent STD.



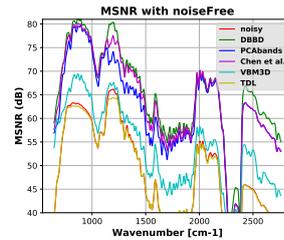
USI #2 (IASI-NG), MSNR



USI #3 (IASI-NG), MSNR



USI #4 (IASI-NG), MSNR



USI #5 (IASI-NG), MSNR

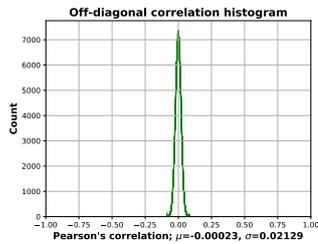
Figure 3.9: MSNR denoising performances of simulated USIs #2, #3, #4, and #5. In red: the MSNR corresponding to the noisy USI. In green: the MSNR of our DBBD method. In blue: the denoising result obtained by simply applying PCA dimensionality reduction with  $N = 20$  principal components band by band. Chen’s method [24] in magenta, [15] in cyan, and TDL in olive color. Since the curves oscillate much, they have been filtered with a Gaussian of standard deviation  $\sigma = 50$  for visualization and comparison purposes only. VBM3D [15] improves slightly the noisy HSI, Chen’s method [24] performs significantly well and beats the PCA band-by-band. DBBD (ours) outperforms all methods. TDL [21] has the worst performance, with a performance below the noisy HSI, confirming that TDL is not adapted to very-high SNR frequency-dependent HSIs.

Fig. 3.11 shows the noise reduction before and after DBBD denoising in terms of the NEDT parameter. It can be observed that the amount of noise decreases significantly after denoising. We show results for only one of the seven IASI-NG USIs, but similar results are obtained with the others.

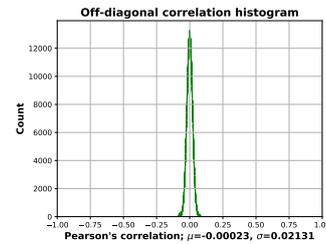
Our proposed method DBBD is especially adapted to hyperspectral images where the SNR is high and the number of channels is very large (several thousands), as in the case of the sensor of future satellites such as IASI-NG. To complete our study we have included a hypothetical scenario with a worse SNR and a reduced number of channels. We multiplied by 10 the STD of the noise and took one out of 70 wavenumbers (a total of 242). The aim of this is to have a quick review of these state-of-the-art methods in a scenario which is not adapted to DBBD. As can be observed in Table 3.5, in this scenario the difference between Chen and DBBD is not significant. The method by Chen should be preferred in this scenario, since it performed better in a majority of USIs. TDL shows competitive results, as expected. Indeed, when the noise is larger, it is easier to separate the noise from the signal, and if the number of channels is reduced TDL and other dictionary-based methods start to be more adapted to denoise such signals. Nevertheless, notice that DBBD learns a basis for each group of similar pixels, and that the decay of eigenvalues for each basis is very sharp. Hence, DBBD can be assimilated to a local dictionary learning.

### 3.4.2 Evaluation with real USIs

The evaluation of the denoising performance with real USIs can only be done through indirect methods such as the analysis of the correlations introduced (Sec. 3.4.1).



USI #2 (IASI-NG) autocor.



USI #3 (IASI-NG) autocor.

Figure 3.10: Histograms of the Pearson auto-correlation along different frequencies of the removed signal in USIs #2 and #3 (IASI-NG), band #3. As can be observed, the auto-correlations of the signal removed by DBBD are centered at zero and their distribution is sharp and symmetric. This gives strong evidence that the part of the signal that was removed by DBBD is indistinguishable from noise.

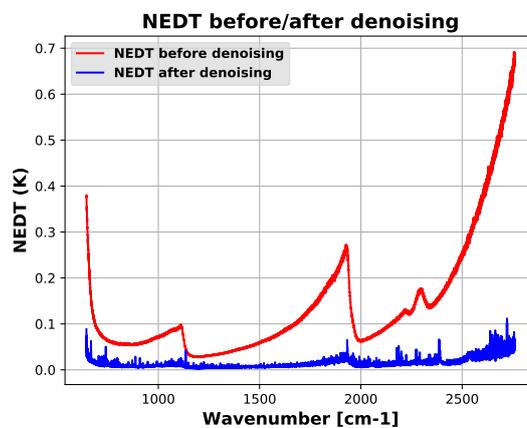


Figure 3.11: Noise reduction before and after denoising with DBBD with IASI-NG USI #6. In red: the model's theoretical NEDT of the noise according to the physical model. In blue: the NEDT of the noise after DBBD denoising. The amount of noise has decreased significantly.

We used the five real USIs shown in Fig. 3.5 acquired by the IASI instrument [47]. We will refer to these real USIs as USI #7, ..., #11. Each of the acquired images is  $120 \times 23$  pixels with 8461 spectral channels. Since these are real USI images, there is no GT to compare to, but the autocorrelation of the removed signal can be analyzed. As explained in Sec. 3.4.1, we expect that the distribution of the Pearson auto-correlation coefficients out of the diagonal is centered at zero and with small variance (it should be zero in the case of an ideal perfect denoiser). The introduced correlation remains small even if groups of channels have similar underlying information. This is particularly important for remote sensing applications which can benefit from increased SNR, for example for retrieval of atmospheric and surface temperatures or ozone atmospheric profiles only if no bias has been introduced [26, 48].

Table 3.6 gives the statistics of the histograms of the Pearson cross-correlations coefficients along all frequencies of the removed signal in real USIs #7 ... #11. For the real IASI data (8461 channels) Chen's method has values very similar to PCA-bands because with a reduced number of channels its thresholding simply set to zero the less significant PCs (as PCA does). The autocorrelation is better in DBBD. Note however that when the number of available channels is larger (as in the IASI-NG case, see Table 3.5, and also Tables 3.2 and 3.3), Chen clearly outperforms PCA. The goal of this experiment is to show that indeed these methods are valid to denoise ultraspectral images.

Statistics of the histograms of the Pearson cross-correlations coefficients along all frequencies of the removed signal in real USIs #7 ... #11. For the real IASI data (8461 channels) Chen's method has values very similar to PCA-bands because with a reduced number of channels its thresholding simply set to zero the less significant PCs (as PCA does). The autocorrelation is better in DBBD. Note how-

USI #	Noisy	VBM3D [15]	DBBD	Chen [24]	PCA bands	TDL [21]
#0	21.74	27.45	35.75	<b>35.90</b>	28.91	35.26
#1	13.83	24.71	<b>30.85</b>	29.53	25.31	29.47
#2	25.82	34.50	39.41	<b>39.85</b>	30.95	38.92
#3	24.79	32.51	39.58	<b>40.13</b>	29.86	38.26
#4	24.47	33.57	39.24	<b>39.35</b>	30.22	37.92
#5	27.67	37.18	40.53	<b>40.86</b>	30.72	40.81
#6	20.02	30.96	<b>36.56</b>	35.83	29.41	35.24
Mean	22.62	31.55	<b>37.42</b>	37.35	29.34	36.55

Table 3.5: Mean MSNR for the seven simulated IASI-NG USIs for VBM3D, DBBD, Chen's method, PCA band by band, and TDL in the case of low SNR and a reduced number of channels. We multiplied by 10 the STD of the noise and took one out of 70 wavenumbers (a total of 242). Now the difference between Chen and DBBD is not significant. The method by Chen should be preferred in this scenario, since it performed better in a majority of USIs. TDL shows competitive results, as expected.

USI #	Mean		STD	
	(a)	(b)	(c)	(d)
#7	-2.21e-5	6.88e-5	3338.96e-5	3446.04e-5
#8	-4.80e-5	63.16e-5	3452.47e-5	5675.18e-5
#9	-5.77e-5	11.81e-5	3327.54e-5	3722.74e-5
#10	-3.77e-5	32.29e-5	3516.86e-5	6539.90e-5
#11	9.91e-5	-9.23e-5	3297.69e-5	2510.47e-5
Mean	<b>-1.33e-5</b>	20.98e-5	<b>3386.71e-5</b>	3386.71e-5

Table 3.6: Statistics of the histograms of the Pearson cross-correlations coefficients along all frequencies of the removed signal in real USIs #7 . . . #11. (a) means in DBBD, (b) means in PCA-bands and Chen, (c) STDs in DBBD, and (d) STDs in PCA-bands and Chen. For the real IASI data (8461 channels) Chen's method has values very similar to PCA-bands because with a reduced number of channels its thresholding simply set to zero the less significant PCs (as PCA does). The autocorrelation is better in DBBD. Note however that when the number of available channels is larger (as in the IASI-NG case, see Table 3.5, and also Tables 3.2 and 3.3), Chen clearly outperforms PCA. The goal of this experiment is to show that indeed these methods are valid to denoise ultraspectral images.

ever that when the number of available channels is larger (as in the IASI-NG case, see Table 3.5, and also Tables 3.2 and 3.3), Chen clearly outperforms PCA. The goal of this experiment is to show that indeed these methods are valid to denoise ultraspectral images.

IASI measures wavenumbers from  $645 \text{ cm}^{-1}$  to  $2760 \text{ cm}^{-1}$  along 8461 channels. Therefore it has a spectral resolution of  $0.25 \text{ cm}^{-1}/\text{ch}$ . For example, the  $\text{NO}_2$  molecule can be detected between  $1290$  and  $1306.19 \text{ cm}^{-1}$  (a bandwidth of  $15.52 \text{ cm}^{-1}$  or equivalently 51 IASI channels). If a denoising method happened to correlate these 51 wavenumbers, it could prevent the detection of  $\text{NO}_2$  or produce false detection of other molecules.

### 3.5 Consequences for the instrument's resolution

In this paper we presented a new denoising method for hyperspectral and ultraspectral imaging based on the physical noise model given by Planck's law. We designed a noise estimation procedure adapted to real USIs and verified on simulated and real data that it retrieves the physical noise model. The noise estimation can favorably be used instead of the physical noise model as discussed in Sec. 3.2.1. It is a simple remedy for the insufficiency of the black body assumption, the differences between pre and post flight radiometric characterization, the signal saturation in the sensor, and finally the

band overlapping. It is well known that a bad noise model leads to bad unmixing results [49]. Our experiments substantiate the usefulness of the physical model and, still better, of its real estimation. They not only allow to remove the exact amount of noise, but also help keep low the correlation introduced in the denoised signal. Finally, the method is *full-spectrum* in the sense that it denoises the totality of the spectrum [24] and not only high-energy PCA channels.

We did three types of evaluations. The MSNR and Structural Similarity (SSIM) metrics are important indicators of success when a GT is available. But we saw that the autocorrelation of the removed signal is perhaps a still more faithful indicator, actually more reliable than the MSNR or SSIM. While the MSNR and SSIM give crucial information on the denoising performance (for simulated data), a high MSNR does not guarantee that the removed noise is uncorrelated.

We showed that the performance of both TDL and VBM3D was poor in terms of the MSNR and SSIM metric. This must be understood in the context of the problem of high-SNR frequency-dependent noise in the case of ultraspectral imaging, for which these methods are not adapted. They can be considered the state of the art and clearly they are the best methods for data with a lower SNR and with a noise model which does not depend on the frequency, but they are simply not adapted to this new scenario. These methods do not take into account the fact that the variance of the noise depends on the frequency in their model, while in our method it is deeply integrated. Of course, one can always try to stabilize the variance with the Anscombe transform [50, 51] or other specific variance stabilizing transforms (VST), but the performance of the stabilization depends greatly on the size of the data and the introduced errors might be too large for the high-SNR case. In general, it is preferable to avoid the VST and integrate the fact that the noise depends on the frequency in the model itself. Other methods, such as Chen's and DBBD are more adapted to this kind of high-SNR USI data.

Yet one of the most important goals of hyperspectral imaging is the characterization of materials and their abundances. The performance of unmixing algorithms might be altered by any unduly introduced correlation in the restored signal. A significant advantage of the correlation measurement is that it can be measured in absence of GT. This has led us to demonstrate that the correlation introduced by our denoising method is very low on IASI real data.

The fact that DBBD does not introduce a significant correlation in the denoised signal and that MSNR and classification results after denoising is largely improved raises an important question in satellite design: is its resolution adequate to its goals? Indeed, in Table 3.4 we showed that for IASI-NG the standard deviation of the noise can be reduced by a factor of the order of 7. This means that it might be possible to reduce the surface of the sensor by a factor of the order of 50 while keeping the same SNR.

Thus, we propose to improve the current hyperspectral sensors by means of image denoising algorithms, with a special focus on the future IASI-NG to be launched in 2021. It is typical for hyperspectral sensors to have a large spectral resolution, but a very small one spatially. Getting a large enough SNR is the reason for the scarce spatial resolution. An increase of the SNR would allow to increase the spatial resolution delivered by the satellite, assuming the same sensor, or to design a new sensor whose surface is adapted to the image processing algorithms that will be applied. The resulting optical apparatus (and satellite) would therefore be much cheaper for an equal final resolution and SNR after processing. Alternatively, a much higher resolution apparatus would be obtained for the same overall cost. These comments apply retrospectively to the flying IASI and to future projects like IASI-NG. In short, we propose to move towards hybrid systems combining the optics of the instrument with the potential of image denoising algorithms.

In a future work, we seek to get *a-priori* information on the natural correlations introduced by the instrument itself in the spectral channels, by analyzing the data acquired at the same zones along different satellite passes. This information would be added to the noise model and help measure the correlation introduced, thus further improving the results.

**Acknowledgement:** Centre National d'Etudes Spatiales (CNES, R&T action) for the access to IASI and IASI-NG data and advice, EUMETSAT and NOVELTIS for the simulated data, and the European Research Council (Advanced Grant *Twelve Labours*), Office of Naval Research (Grant N00014-97-1-0839), and Direction Générale de l'Armement (DGA) for their funding of the project.



# Chapter 4

## Ongoing research on the design of SMOS-HR

### Abstract

In 2009 ESA launches the SMOS (Soil Moisture and Ocean Salinity) satellite with the objective of studying the exchange processes between the surface of the Earth and the atmosphere and hopefully help develop better models of weather and climate. The satellite carries an interferometer known as MIRAS, a structure with three-arms and a total of 69 antennas. Each of the MIRAS antennas is mainly sensitive to electromagnetic emissions in the L-band (where  $H_2O$  dipoles emit because of their thermal agitation). The correlation of the signal received by each pair of antennas allows to estimate one Fourier coefficient of the brightness temperature scene (in the case, a part of the Earth and our galaxy). The final goal is to obtain the soil moisture (or ocean salinity) of the observed zone, which is obtained by using the brightness temperatures from MIRAS as the input of a different model. Unfortunately, the techniques used in SMOS only allow to reconstruct a brightness temperature image with a spatial resolution of  $40 \text{ Km}^2/\text{pixel}$  at nadir. This resolution does not allow for reliable climate models since they need of a resolution of about  $10 \text{ Km}^2/\text{pixel}$ . Moreover, the antenna's configuration of the current SMOS satellite is regular and spectrally under-sampled below the Shannon-Nyquist limit. Thus, the spatial reconstructions are strongly folded and part of the acquired information is lost. To address these problems, a new version of the satellite is being designed (SMOS-HR), and we collaborate with CNES's CESBIO to this purpose. We compare in this chapter two different approaches with the objective of increasing the resolution of the instrument: a regular configuration proposed by CESBIO (the *quincunx*) that still provides a spatially folded reconstruction, and an irregular (but yet calibrable) configuration proposed by CMLA which is free of folding and which we aim to improve the spatial resolution of the instrument and eventually reducing the number of antennas needed. The position of the antennas is obtained as a solution to a non-convex problem that aims to find a sampling that is uniform and without "holes", solved by the Particle Swarm Algorithm. In this chapter I show the very preliminary results of this ongoing research.

### Notation

The following symbols are used along this document. The symbols that are vectors or matrices are written in bold weight. For example,  $a$  represents the position a single antenna whereas  $\mathbf{a}$  is an iterable list of positions of several antennas.

- $\lambda$ : wavelength of the antenna. It is obtained as  $\lambda = c/f_0 \approx 0.21 \text{ m}$  where  $c \approx 299792458 \text{ m/s}$  is the speed of light and  $f_0 \approx 1413 \text{ MHz}$  the central frequency of the antenna.
- $\delta\xi$  (resp.  $\delta\eta$ ): spatial step of the antennas
- $\delta u$  (resp.  $\delta v$ ): Fourier step of the sampling frequencies

- $d$ : spacing between antennas in the case of regular configurations. It is the distance between the centers of two antennas. The diameter of the antennas is  $\lambda$ , and consequently they cannot get closer than  $0.21 \text{ m}$  to each other.
- $q$ : constant that depends on the geometry of the sampling. It is defined as  $q = \delta u / \delta \xi$ . It is  $q = 2$  when using Cartesian coordinates (for example, with the regular square),  $q = \sqrt{3}$  for a hexagonal sampling (SMOS), and  $q = \sqrt{2}$  for the quincunx. The minimum distance between antennas before spatial folding happens is related to  $q$ , as seen in Eq. (4.4).
- $a$ : position of one antenna
- $\mathbf{a}$ : list of antennas
- $\mathcal{N}$ : number of antennas of the instrument
- $N$ : the size of the reconstructed image is  $N_x \times N_y$
- $\vec{u}$ : baseline. It is the vector that goes from one antenna to another, in  $\lambda$  units.
- $\mathcal{D}_{\vec{u}}$ : a function<sup>1</sup> that maps all the baselines of the instrument with their multiplicity. Each baseline is assigned a unique key (see Algo. 11) that is used as an index to retrieve the corresponding multiplicity.
- $v$ : visibility
- $\mathbf{V}$ : list of visibilities
- $\nu$ : sampling frequency associated to a visibility
- $\boldsymbol{\nu}$ : list of sampling frequencies
- $m$ : multiplicity of a sampling frequency
- $M$ : a matrix indexed by the antennas whose entries  $[i, j]$  are 1 when there is a baseline joining antenna  $a_i$  with  $a_j$  and 0 otherwise
- $\Psi$ : matrix corresponding to the system of equations used to calibrate the instrument
- $G$ : matrix that when multiplied by  $T$  temperatures gives  $V$
- $G^*$ : the pseudo-inverse of  $G$  (adjoint operator)
- $T$ : vector of brightness temperatures
- $\alpha$ : rotation angle
- $\gamma = 1e - 6$ : step of the thin grid used to determine if two sampling frequencies should be considered the same
- $\mu$ : discretization step of the fine grid of antennas' positions. This step is increased progressively during the design of antenna's positions, until the system gets calibrable.
- $f$ : an unknown scalar related to the antenna (it can be the parameter that controls the amplitude in the ARP or the phase). It is used in the system of equations that describe the instrument to be calibrated.
- $\phi$ : an unknown related to a pair of antennas (amplitude or phase). Used in the system of equations describing the instrument being calibrated.
- $C$ : cost function that PSO aims to minimize to improve the frequency plane covering
- $D$ : local density of frequencies

---

<sup>1</sup>In most programming languages this function corresponds to a dictionary or an associative array.

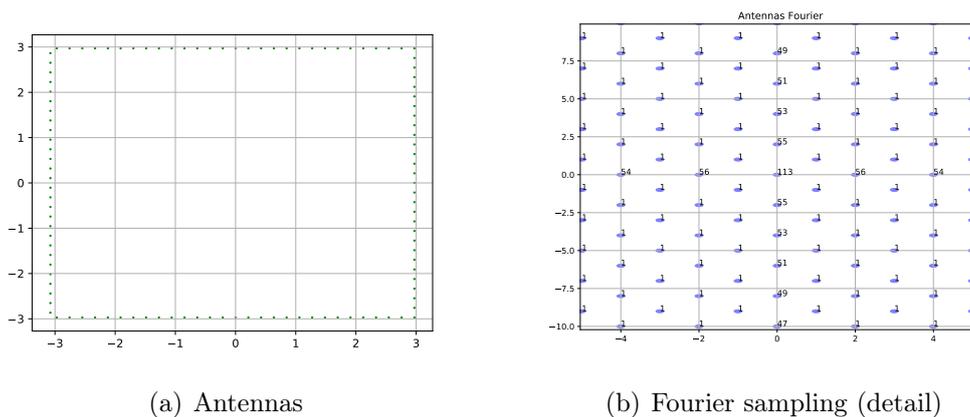
- $\mathbf{K}$ : convolution kernel used to obtain  $\mathbf{D}$  in the  $u - v$  domain
- $\Omega$ : prescribed Fourier domain that the set of visibilities  $\mathbf{V}$  has to cover as well as possible
- $h$ : penalty for the presence of "holes" in  $\Omega$
- $\gamma$ : hyperparameter
- $\odot$ : an integer representing a gridded sampling frequency
- $\zeta$ : matrix containing the distance of one antenna to any other
- $\delta\theta$ : angular resolution
- $\delta x$ : spatial resolution
- $D$ : maximum spatial distance between antennas in the instrument

## 4.1 Introduction to the questions of irregular sampling

The objective of this report is to evaluate two different architectures of interferometric networks for the future SMOS-HR satellite, the continuation of the current and operational SMOS as described the Kerr et al. founding paper [52]. Specifically we focus on these two configurations:

- The *quincunx*
- The irregular square

Algo. 9 gives the pseudo-code to build the quincunx configuration. Fig. 4.1 shows the quincunx configuration with 113 antennas on the left and a detail of the corresponding sampling frequencies on the right. When the quincunx configuration is mounted in the satellite it is rotated 45 degrees, but without loss of generality we will evaluate all of our configurations without the rotation. The name *quincunx* becomes clear after looking at the disposition of the sampling frequencies, similar to the dots of the side with number five in a die.



(a) Antennas

(b) Fourier sampling (detail)

Figure 4.1: Quincunx configuration with 113 antennas. Left: the spatial configuration. Right: the sampling frequencies in the Fourier domain (detail). When the quincunx configuration is mounted in the satellite it is rotated 45 degrees, but without loss of generality we will evaluate all of our configurations without the rotation. The name *quincunx* becomes clear after looking at the disposition of the sampling frequencies, similar to the dots of the side with number five in a die.

The irregular configuration is also based on a square shape (see the papers by Boone [53, 54] for a study on irregular configurations), but the position of the antennas are given by optimizing a cost

function of the local density of the samples in the frequency plane. The cost minimization aims to obtain a spectrum as regular as possible, without "holes", and avoiding redundant visibilities (of multiplicity more than one). The antennas are located freely along the perimeter of the square and a small movement (at most  $\pm\lambda$  meters) in the lateral direction is also authorized in order to improve the number of degrees of freedom in the final solution. Fig. 4.2 shows one irregular solution obtained with our method, where the maximum distance between antennas is 8.59 m, the length of the bar 6.83 m, and the width of the bar is 0.71 m. The width of the bar is chosen so the surface of any antenna lies over the surface.

In order to compare both configurations, we consider two instruments of the same size and using both 113 antennas. Note that to reach an spatial resolution of about  $10 \text{ Km}^2/\text{pixel}$  (one of the requirements craved in the SMOS-HR mission), a bigger instrument is required. Indeed, a spatial resolution of 10 Km assuming a satellite altitude of  $h = 750 \text{ Km}$  requires the maximal distance between antennas to be at least  $D = \frac{H\lambda}{10} = \frac{750\lambda}{10} = 15.75 \text{ m}$ . Therefore, the length of each bar would be  $15.75/\sqrt{2} = 11.14 \text{ m}$ .

For the regular quincunx case, this means having  $11.14 \times 4/d$  antennas, with  $d$  the distance between antennas. If we assume that the minimal distance between antennas is  $\lambda$  (the usual constraint), we would need a quincunx with 231 antennas. The actual setup for the quincunx configuration<sup>2</sup> is  $D = 17 \text{ m}$ , 231 antennas,  $d = \lambda$ , and a spatial resolution of 10.5 km after Hanning apodization. While it is possible to compute a pseudo-inverse with 231 antennas for the quincunx configuration, it was too memory-demanding for the irregular case (more than 128 Gb of RAM required, which is over the specifications of our current computers at CMLA). Nevertheless, since we compare two instruments of the same size and the same number of antennas, our comparison here is valid. In the short future we will compute the pseudo-inverse corresponding to the irregular configuration with a system with scaled up RAM.

An obvious problem of a purely irregular configurations is that in general it is not possible to find several pairs of antennas corresponding to the same baseline, thus making calibration impossible. In order to calibrate the proposed irregular configuration we followed the method of Camps et al. [55] and discretized the position of the antennas until the system of equations had a rank that allows solving it.

## 4.2 Model of the visibilities

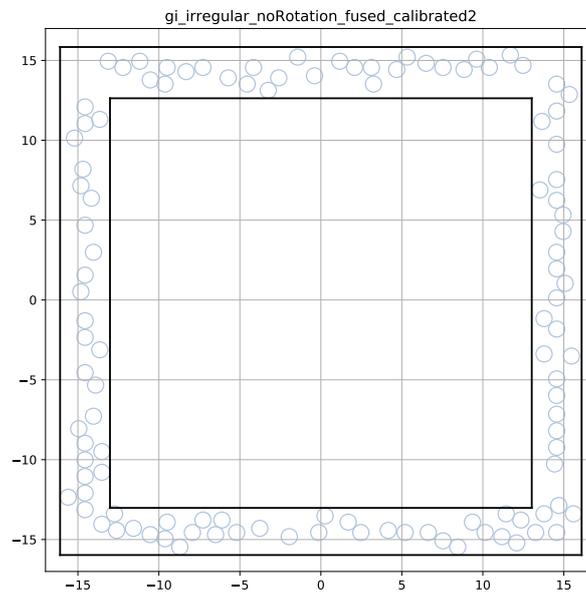
Given a set of sampling frequencies obtained from the corresponding baselines between two antennas divided by  $\lambda_0 \approx 0.21 \text{ m}$ , the visibilities are modeled in Eq. (4.1), which is derived from the Van Cittert-Zernike theorem [56, 57]. Neglecting the effect of the *fringe washing function*, the visibilities can approximated by

$$\mathbf{V}(i, j) = \iint_{\|(\xi, \eta)\|_2 \leq 1} \frac{F_i(\xi, \eta)F_j(\xi, \eta)T(\xi, \eta)}{\sqrt{1 - \xi^2 - \eta^2}} \exp[-2i\pi(\nu_{i,j}^u \xi + \nu_{i,j}^v \eta)] d\xi d\eta \quad (4.1)$$

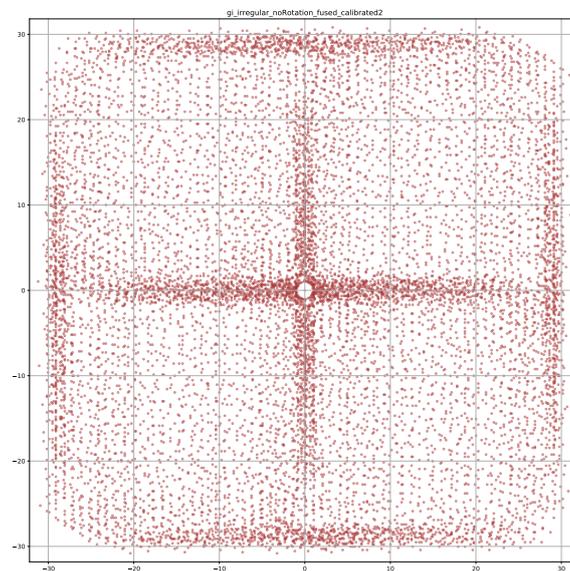
where  $i, j \in 1 \dots \mathcal{N}$  are the indices of two antennas  $a_i, a_j$ . The baselines are the vectors that join  $a_i$  to  $a_j$ ,  $\vec{u}_{i,j} = (a_i - a_j)/\lambda$ . The corresponding sampling frequencies are  $\nu_{i,j} = 2\vec{u}_{i,j}$ . We denote as  $\nu_{i,j}^u$  and  $\nu_{i,j}^v$  the two components of the vector in the  $u - v$  domain. Several baselines may correspond to the same sampling frequency, but each frequency is associated to a unique visibility. We will refer to the *multiplicity*  $m$  of a visibility to the number of baselines that give the same sampling frequency.

Here  $\xi$  and  $\eta$  are the director cosines, and  $F(\xi, \eta)$  the antenna radiation pattern (ARP). We adopted the simplification  $F_i = F_j = F$  in our simulations. In practice, each ARP will be different and the calibration procedure explained in Sec. 4.4 is required in a realistic setup. Thus, the visibility can be seen as the Fourier transform of the function  $\frac{F^2(\xi, \eta)T(\xi, \eta)}{\sqrt{1 - \xi^2 - \eta^2}}$ . Assuming that the antennas have been

<sup>2</sup>See *A high spatial resolution SMOS follow-on: SMOS-HR*, <https://lps19.esa.int/NikalWebsitePortal/living-planet-symposium-2019/lps19/Agenda/AgendaItemDetail?id=e26a134a-f160-4e98-8a1c-f12a646ddc07>



(a) Antennas



(b) Fourier sampling

Figure 4.2: One irregular solution obtained with our method. It has 113 antennas and it can be calibrated. On the left, the configuration of the antennas (in  $\lambda$  units). The maximum distance between antennas is 8.59 m, the length of the bar 6.83 m, and the width of the bar is 0.71 m. The width of the bar is chosen so the surface of any antenna lie over the surface.

calibrated, it suffices to compute the inverse Fourier transform and multiply by  $\frac{\sqrt{1-\xi^2-\eta^2}}{F^2(\xi,\eta)}$  to estimate  $\mathbf{V}(i,j)$ . The ARP is given by  $F(\xi,\eta) = \pi^2 \cos \theta$  with  $\theta = \text{asin}(\sqrt{\xi^2 + \eta^2})$ . Fig. 4.3 shows the function  $\frac{F^2(\xi,\eta)}{\sqrt{1-\xi^2-\eta^2}} = \pi^4(1 - \xi^2 - \eta^2)^{3/2}$ .

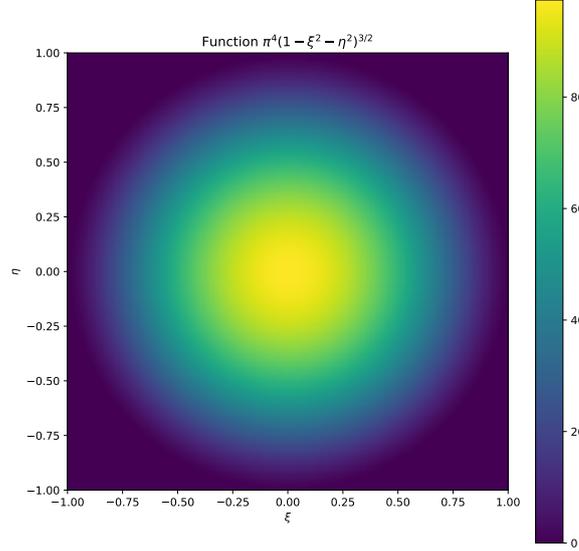


Figure 4.3: Function  $\frac{F^2(\xi,\eta)}{\sqrt{1-\xi^2-\eta^2}} = \pi^4(1 - \xi^2 - \eta^2)^{3/2}$ . It gets its maximum at the center and it decreases until it gets to zero when  $\|(\xi,\eta)\| = 1$ .

### 4.2.1 Discretization

In order to perform simulations, one needs to discretize Eq. (4.1) and take into account the different units used in the spatial and spectral domains. The *negative* frequency is also available and obtained as the baseline that goes in the inverse sense. Therefore, the antennas positions belong to the unit circle and the frequencies to the circle of radius 2. The fundamental relations between the support and the steps in the spatial and spectral domains has been already studied by Camps et al., Corbella et al. [58, 59] and developed in the PhD dissertations of Picard and Khazaal [60, 61].

Assuming that the ARP are known and equal, one can express and discretize Eq. (4.1) and obtain:

$$\mathbf{V}[i,j] = \sum_{x=0}^{N_x-1} \sum_{y=0}^{N_y-1} \frac{F^2[x,y]\mathbf{T}[x,y]}{\sqrt{1-\xi[x,y]^2-\eta[x,y]^2}} \exp\left(\frac{-i2\pi\nu_{i,j}^u x}{N_x} + \frac{-i2\pi\nu_{i,j}^v y}{N_y}\right) \quad (4.2)$$

with

$$\xi[x,y] = 2(x - N_x/2)/N_x,$$

$$\eta[x,y] = 2(y - N_y/2)/N_y,$$

$$\vec{u}_{i,j} = (a_i - a_j)/\lambda,$$

$$\nu_{i,j}^u = 2\vec{u}_{i,j}^u,$$

$$\nu_{i,j}^v = 2\vec{u}_{i,j}^v.$$

Spatial folding happens when the step in the Fourier domain is smaller than the step in the spatial domain, normalized by  $\lambda$  and taking into account  $q$  (see Eq. (4.4)) according to each particular sampling. The Fourier step is  $\delta u = \frac{\delta \xi q}{\lambda} = \frac{dq}{\lambda}$ . The minimal distance between antennas that avoids spatial folding is given in Eq. (4.3) (without loss of generality, the same is valid for the  $\eta$  axis).

$$d_{\min} = \lambda/q, \quad (4.3)$$

where

$$q = \delta u / \delta \xi \quad (4.4)$$

depends on each particular configuration. For example, in SMOS (hexagonal sampling,  $q = \sqrt{3}$ ) the antennas are equispaced with  $d = 0.875\lambda$ . However, the minimal distance to avoid spatial folding is  $d_{\min} = \lambda/\sqrt{3} = 0.58\lambda < d$  and therefore spatial aliasing is present in the reconstructed images.

We define  $T'[x, y] = \frac{F^2[x, y]T[x, y]}{\sqrt{1 - \xi[x, y]^2 - \eta[x, y]^2}}$  the *modified* brightness temperature (BT) with  $\mathbf{V}[u_b, v_b]$ . Finally, Eq. (4.5) gives the discretization of the visibilities.

$$\mathbf{V}[i, j] = \sum_{x=0}^{N_x-1} \sum_{y=0}^{N_y-1} T'[x, y] \exp\left(\frac{-i2\pi\nu_{i,j}^u x}{N_x} + \frac{-i2\pi\nu_{i,j}^v y}{N_y}\right). \quad (4.5)$$

Finally, this corresponds to the DFT of the modified brightness temperature, as shown in Eq. (4.6):

$$\mathbf{V}[u_b, v_b] = \text{DFT}(T'[x, y]). \quad (4.6)$$

The discretization of Eq. (4.1) into Eq. (4.6) is just an approximation since indeed  $T(\xi, \eta)$  is a continuous physical magnitude. However, in practice one needs to discretize the brightness temperature scene with a large  $N_x$  and  $N_y$  (a thin grid) in order to perform numerical simulations.

### 4.3 Optimizing the position of the antennas

A good spatial configuration of antennas is one that samples the  $u - v$  domain uniformly with most of the visibilities of multiplicity one. We model the interferometer as a sum of Dirac masses, each located at position  $a_i$  for each of the  $\mathcal{N}$  antennas.

The baselines are given by all differences  $a_i - a_j$ . We assume that the positions of the antennas have been multiplied by  $\lambda$  and that their positions have been fine-gridded in order that  $[a_i, a_j] \in \mathcal{Z}^2$ . Thus we shall define  $\mathbf{M}$  as the matrix that contains the multiplicity of each baseline, indexed by  $[a_i, a_j]$ , Eq. (4.7).

$$\mathbf{M}[a_i - a_j] = |\{a_i - a_j\}|, \quad (4.7)$$

where  $[i, j] \in [1 \dots \mathcal{N}]^2$  and  $|\cdot|$  is the cardinal operator.

Our concern now is to evaluate how good a distribution  $\mathbf{a}$  can be, given that  $\mathbf{M}$  has to cover as well as possible a prescribed Fourier domain  $\Omega$ . In short, we want  $\mathbf{M}$  to be spread on an  $\epsilon$ -net, namely ideally ensure that no point in  $\Omega$  is farther away from some point in  $\mathbf{M}$  than  $\epsilon$ , and that in addition, we do have an "economy" principle, namely that the points in  $\mathbf{M}$  do not clutter; they must all stand as far away from their neighbors as possible.

We call  $\Omega$  the goal domain (a square in our case), as this would be optimal for a satellite sweeping the ground, because in that way we get the same number of angular samples for all points in the swath. We want that the density of  $\mathbf{M}$  is as flat as possible. This can be tested by convolving  $\mathbf{M}$  by one or several Gaussian  $\mathbf{K}_\sigma$  and minimizing the variance of the result and the presence of "holes". How to choose  $\sigma$ ? Basically  $\mathbf{K}_\sigma$  will strongly penalize samples which are closer to each other than  $\sigma$ . Thus  $\sigma$  must be chosen of the order of the average distance between samples we want to achieve. Since we want to cover an area  $|\Omega|$  with  $|\mathbf{a}|$  samples, this target distance is of the order of  $\sigma = \sqrt{\frac{|\Omega|}{|\mathbf{a}|}}$ , meaning for example that a rectangle  $\Omega$  would be covered by  $|\mathbf{a}|$  squares with radius  $\sigma_0$ .

Obtaining such a configuration is a non-convex problem that can not be solved by classic algorithms which rely on minimizing the gradient, such as gradient descent or others. There are other algorithms that are not based on minimizing the gradient that reach good enough solutions. We used the Particle Swarm Optimization (PSO) [62] algorithm with the cost function given in Algo. 15. This cost function takes as input parameters:

- longitudes: a list of scalars that represent the position of each antenna in the square. We define a path that goes from one of the corners and turns around the square until it completes its perimeter. The endpoint of the path defines the coordinates of the antenna, in  $\lambda$ -units.
- lateralShifts: a list of scalars that represents the movement of the corresponding antenna in the lateral direction. It is given as a number between  $-1$  and  $1$  (the diameter of the antennas in  $\lambda$ -units).
- $K$ : the convolution kernel used to obtain a local density at each point of  $D$ .
- $\gamma_1, \gamma_2$ : the two hyperparameters of the algorithm. They depend on the configuration of the antennas and the size of the instrument, and they need to be found empirically.
- side: the side of the square, in  $\lambda$ -units.

The output is the value of that PSO intends to minimize.

Our algorithm proceeds as follows:

1. Read the coordinates of the antennas
2. Obtain all baselines and their multiplicity (Algo. 12).
3. Obtain a matrix  $J$  which fine-grids  $M$  (Algo. 13).
4. Compute the matrix that represents the density at each point as  $D = J * K$ , where  $K$  is the Gaussian convolution kernel.
5. Finally, compute the final cost function as  $C = \text{var}(D) + \gamma_1 h$ , with  $h = |\mathbf{D} < \gamma_2 \bar{D}| / |\mathbf{D}|$  (Algo. 15).

### 4.3.1 Separating antennas below the minimal allowed distance

In practice one cannot build the instrument if two antennas are closer than  $\lambda$ . Adding a penalty term in the cost function is not enough to ensure that no pair of antennas is below the minimal distance. Therefore one needs to explicitly look for pairs of antennas that violate the constraint. The position of one of the antennas in all violating pairs is then perturbed in the optimization process until there is no pair below the minimal distance. Algo. 16 details this procedure, which is applied to the solution obtained with PSO.

## 4.4 Calibration

Building a set of antennas with exactly the same physical properties is unfeasible, as they have anyway a different position and electromagnetic interaction with the satellite's frame. Therefore the exact gain and phase shift of each of the antennas needs to be determined experimentally, eventually once the antennas have been mounted in the instrument bars and the satellite is on its orbit as developed in Brown et al. [63].

Camps. et al. proposed the Redundant Space Calibration (RSC) method [55], where the raw observation (the amplitude of the phase) of the visibility given by two antennas  $a_i$  and  $a_j$  depends on the amplitude or phase of each of two antennas, as well on their cross-correlation. The raw phase observation from antennas  $a_i$  and  $a_j$  is

$$\phi_{i,j}^{\text{raw}} = f_i - f_j + \phi_{i,j}, \quad (4.8)$$

and equivalently for the gains

$$A_{i,j}^{\text{raw}} = G_i - G_j + A_{i,j}. \quad (4.9)$$

The phase shift of each of the antennas can be determined by solving the system of equations for each pair of antennas with (4.8) and the gains independently with the system of equations (4.9). The unknowns are the  $f_{(\cdot)}$  (resp.  $G_{(\cdot)}$ ), and  $\phi_{(\cdot,\cdot)}$  (resp.  $A_{(\cdot)}$ ). The RSC method assumes that  $\phi_{i,j} = \phi_{k,l}$  and  $A_{i,j} = A_{k,l}$  if  $a_i - a_j = a_k - a_l$  (the same baseline). Therefore, the system of equations can be solved only if each visibility is observed more than once. From now on we will only refer to equations on the phase shift  $\phi$ , since exactly the same can be applied to the gains, without loss of generality.

In matrix notation,

$$\phi^{\text{raw}} = (f_1 \ f_2 \ \dots \ \phi_{1,2} \ \phi_{1,3} \ \dots) \Psi, \quad (4.10)$$

where  $\Psi$  is a matrix which contains entries with  $+1$  (for  $f_i$  and  $\phi_{i,j}$ ) and  $-1$  (for  $f_j$ ). Each of the columns of  $\Psi$  represents one of the equations of the system.

As Camps et al. point out (see section 2 of [55]), one can assume that  $f_0 = 0$  and refer all the other phase shifts to that reference. Indeed, the system is invariant to adding a global phase-shift constant. Therefore, the system is determined if

$$\text{rank}(\Psi) \geq |f| + |\phi|, \quad (4.11)$$

where  $|f| + |\phi|$  is the number of unknowns.

However, in a purely irregular configuration there is no guarantee that the multiplicity of the visibilities is more than one, thus making the system of equations unsolvable. A simple solution is to introduce regularity in the position of the antennas until the system can be calibrated with the condition in Eq. (4.11). Our calibrated solution has a system of rank 318, 427 equations and 318 unknowns.

## 4.5 Irregular sampling and reconstruction

Assuming that the ARPs are known (and under the simplifying assumption that they are all equal), the Corbella formula (4.1) states that the visibilities are the Fourier transform of the (modified) brightness temperatures. After discretization, the vector of visibilities  $\mathbf{V}$  can be obtained by multiplying the corresponding Fourier matrix  $\mathbf{G}$  by the vector of brightness temperatures  $\mathbf{T}$ , as shown in Eq. (4.12).

$$\mathbf{V} = \mathbf{G}\mathbf{T}'. \quad (4.12)$$

We recall we defined the modified brightness temperature as  $\mathbf{T}' = \frac{F^2 \mathbf{T}}{\sqrt{1-x^2-y^2}} = \pi^4 (1-x^2-y^2)^{3/2} \mathbf{T}$ , and therefore that  $\mathbf{T} = \pi^{-4} (1-x^2-y^2)^{-3/2} \mathbf{T}'$ . Given a set of visibilities  $\mathbf{V}$  of irregular sampling frequencies one can obtain an approximation of  $\mathbf{T}'$  by using a pseudo-inverse of  $\mathbf{G}$ , that we will denote as  $\mathbf{G}^*$ :

$$\mathbf{T}' \approx \mathbf{G}^* \mathbf{V}. \quad (4.13)$$

Algo. 24 details the procedure to fill in the irregular Fourier matrix. It is possible to add the factor  $\pi^4 (1-x^2-y^2)^{3/2}$  directly in  $\mathbf{G}$ , but we observed that the results were slightly better if the ARP is corrected after obtaining the inverted image for the irregular configuration, rather than adding it in  $\mathbf{G}$  itself. In any case, both approaches are equally valid. See Anterrieu [64] for more details on recovering BTs with the pseudo-inverse technique.

The first index of the matrix  $\mathbf{G}$  is the spatial index in  $\{0 \dots N_x N_y - 1\}$  and the second index refers to the sampling frequencies  $\nu$ . Note that  $\mathbf{G}$  is not necessarily square (the number of reconstructed spatial points in general is generally lower than the number of sampling frequencies available) and therefore a pseudo-inverse is needed. We used the Moore-Penrose algorithm [65], which sets to zero the eigenvalues that are below a certain threshold in the SVD decomposition.

### 4.5.1 Regularization of $G^*$

The fact that  $q > 1$  necessarily (due to the physical size of the antennas) implies spatial folding due to undersampling in the regular cases and leads to an ill-posed problem in the case of irregular sampling. Indeed,  $G$  does not have all the needed frequencies and therefore one needs to regularize  $G$  in order to obtain an approximation of  $T'$  with minimal error.

The regularization of  $G$  has two effects: it introduces a reconstruction error (the small variations associated to the small eigenvalues can no longer be recovered) and makes the inverse matrix more robust to noise. Indeed, the small eigenvalues in  $G$  correspond to large eigenvalues in  $G^*$  and small perturbations lead to explosive reconstruction errors.

We observed experimentally that with the small level of radiometric noise (see Eq. (4.14)) the quincunx reconstruction is hardly sensitive to the threshold on the eigenvalues. We set a threshold of  $1e-11$ . On the other hand, with irregular sampling the reconstruction is very sensitive to the chosen threshold. After an exhaustive search, we found that the threshold 0.03 is optimal in our setup.

## 4.6 Evaluation

We evaluated the reconstruction with the quincunx and the irregular square with two scenes:

- A circle representing a body of constant 300K temperature and 0K outside the circle, and
- An RGB image of the Earth.

The RGB image of the Earth is not assumed to be a realistic scene of the brightness temperature of the planet. It is used only to test with a non-constant scene. The image was borrowed from NASA's *Blue Marble Next Generation* collection<sup>3</sup>, a composite of MODIS observations at a spatial resolution of 500 meters. This does not correspond to the operational parameters of SMOS-HR, but it is enough to compare both the quincunx and irregular square configurations. As for the image of the disk, it has an abrupt change when it goes from 300 K to 0 K and it is useful to evaluate the behavior of the instrument in the presence of extreme signal changes.

In our experiments we considered  $\mathcal{N} = 113$  antennas, which accounts for  $\mathcal{N}(\mathcal{N} - 1) + 1 = 12657$  visibilities. In our irregular solution we imposed a bit of regularity in order to be able to calibrate the instrument, and finally we ended up with 11667 unique sampling frequencies, slightly down from the maximal possible number. The number of pixels of the reconstructed image is chosen so the maximum sampling frequency can be expressed. If we denote by  $\nu_{\max}$  the sampling frequency of maximum module, then the reconstructed image must be contained in a circle whose diameter is  $\nu_{\max}$ . Since the shape of the set of sampling frequencies is square for both the quincunx and irregular configurations, some parts of the domain will not contain any sampling frequency.

The noise added to the visibilities has a standard deviation

$$\sigma = V(0,0)/\sqrt{m_i 2B\tau}, \quad (4.14)$$

where  $B = 20$  MHz,  $\tau = 1$  s, and  $m_i$  is the multiplicity of visibility  $i$ .

The angular and spatial resolutions of the instrument are given by Eq. (4.15) and Eq. (4.16) namely. In these equations  $D$  is the maximum spatial distance among all antenna pairs, and  $h = 750$  Km the altitude of the satellite.

$$\delta\theta = \frac{\lambda}{D}, \quad (4.15)$$

$$\delta x = h\delta\theta. \quad (4.16)$$

<sup>3</sup><https://visibleearth.nasa.gov/view.php?id=73751>

Config	Ants.	Unq. Fr.	$d$	$D$	$\delta\theta$	$\delta x$	Swath
Quincunx	113	6497	$\lambda$	8.40 m	1.45 deg	18.94 Km	3048.91 Km
Irreg. sq.	113	11667	$1.04\lambda$	8.59 m	1.42 deg	18.53 Km	3280.45 Km

Table 4.1: Summary of the characteristics of the quincunx and irregular square instruments. In the irregular case,  $d$  is computed as the average of the minimal distances from one antenna to any other.

We compared two different instruments (the quincunx and the irregular square configurations) with the same number of antennas and the same size. Since the quincunx has equally spaced antennas, the number of antennas has a direct impact on the geometric and spatial resolutions. Indeed, the maximum distance between two antennas depends on the size of the instrument, which depends also on the number of antennas. However, in the irregular square the antennas are not equally spaced and one can set the side of the bar while keeping a reduced number of antennas. For the quincunx with 113 antennas the longest distance between two antennas is 8.4 m. Thus the angular resolution is 1.45 deg and the spatial resolution is 18.94 Km. In the case of the irregular square the longest antennas' distance is 8.59 m and thus the angular resolution is 1.42 deg and the spatial resolution of 18.53 Km. Note that indeed the resolution of the quincunx can be improved by increasing the number of antennas (thus making the instrument larger), but our goal here is to compare both instruments when using a similar number of antennas.

Table 4.1 shows the characteristics of the two instruments. It shows:

- Config: the configuration (quincunx or irregular square)
- Ants.: number of antennas in the instrument
- Unq. Fr.: number of unique sampling frequencies
- $d$ : physical distance (spatial step) between antennas, in meters
- $D$ : largest distance among all pairs of antennas
- $\delta\theta$ : angular resolution
- $\delta x$ : spatial resolution
- Swath: the swath covered by the reconstructed image, including any folded area

The quincunx instrument is regularly undersampled and therefore the obtained BT images present large folding, which is not present in the case of the irregular square instrument. Therefore, one cannot compare the two obtained BT temperature images directly. For the quincunx we evaluate the performance within a squared zone far away enough from the folding border. It does not make any sense to evaluate the performances directly in the folded zone, which is to be discarded.

For the irregular square instrument there is a small reconstruction error due to the irregular sampling, that places  $\mathcal{N}$  replicas (whose intensity is divided by the average spatial step) of the BT image over itself, but there is no folding as in the classic regular sampling. Nevertheless, the larger field of view of the irregular sampling square approaches the border of the ARP, where the information is lost (multiplied by zero). The reconstruction error increases as  $\|(\xi, \eta)\| \rightarrow 1$ , thus limiting the spatial resolution. We evaluate the radiometric resolution of the instrument in slices in the  $(\xi, \eta)$  domain to determine the maximum reconstruction size allowed. In the irregular case,  $d$  is computed as the median of the minimal distances from one antenna to any other.

Section 4.6.1 evaluates the instruments when observing the constant temperature scene, and Section 4.6.2 evaluates the instruments with an image of the Earth. Section 4.6.3 shows the point spread function (PSF) of both instruments when observing a point source, and Section 4.6.4 the convolution kernels when all the visibilities equal 1.

### 4.6.1 Evaluation with a body of constant 300 K temperature

We consider that the BT is 300 K when  $\sin \theta < \frac{R_T}{R_T+H}$  and 0 otherwise. Here  $R_T$  is the radius of the Earth,  $h = 750$  Km the altitude of the satellite, and  $\theta = \text{asin}(\sqrt{\xi^2 + \eta^2})$ .

In the quincunx instrument, the antennas are equispaced  $d = \delta\xi = \lambda$  meters and the sampling frequencies are arranged in quincunx configuration, with  $q = \sqrt{2}$ . Thus, the minimal distance between antennas without spatial folding is  $d_{\min} = \lambda/\sqrt{2} = 0.15 < \lambda = 0.21$ . Therefore, the reconstructed image will exhibit a large folded zone. Fig. 4.4 shows the reference image of a body at 300 K and the scene after applying the ARP.

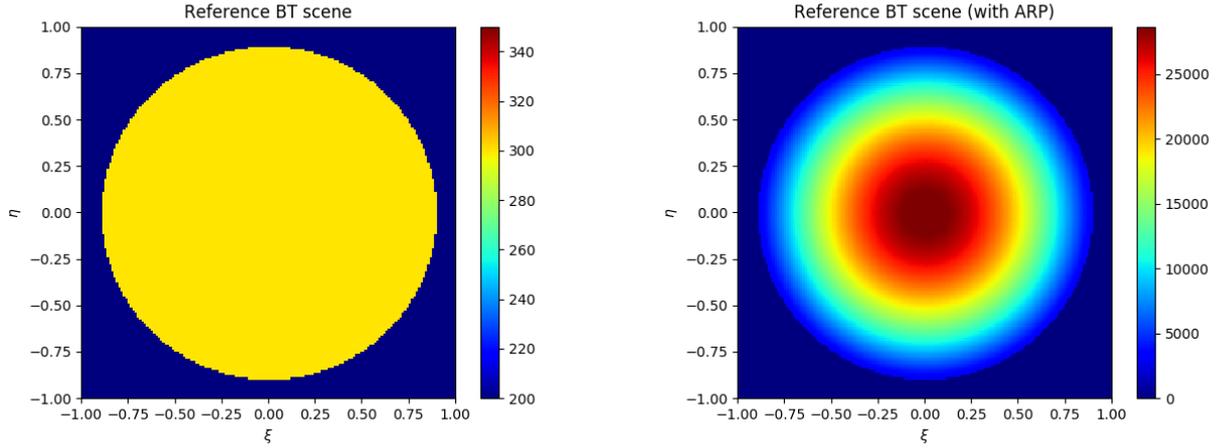


Figure 4.4: Left: reference image of a body at 300 K. Right: reference BT after applying the ARP.

Fig. 4.5 shows the reconstructed BT with the quincunx where the folding is clearly visible. The evaluation is performed inside the white rectangle. In the case of irregular sampling, the effect of sub-sampling the Fourier spectrum is no longer a convolution by a train of equispaced deltas, but the multiplication with some kernel in the Fourier domain (or the convolution by this kernel in the spatial domain). Thus, reconstructed image does not present folding in the classic sense (as in the case of the quincunx) but an interference is added. This interference can be regarded as an additive noise which is correlated with the image. Section 4.6.4 shows the corresponding convolution kernels.

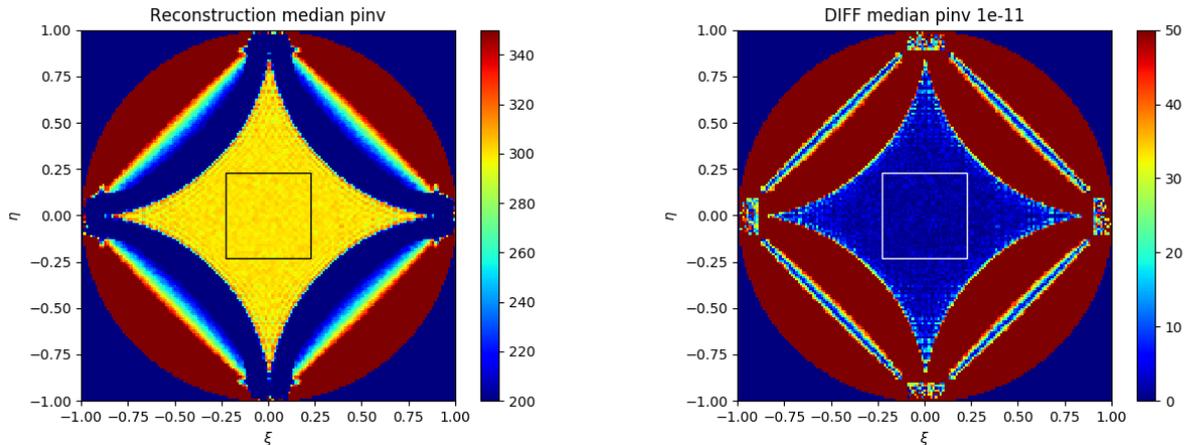


Figure 4.5: Reconstruction with the quincunx configuration and a body at 300 K. Large spatial folding is present. Left: reconstructed image. Right: absolute error. The absolute error measured within the black-square zone of the AF-FOV is 1.86 K, and the maximum absolute error is 6.03 K.

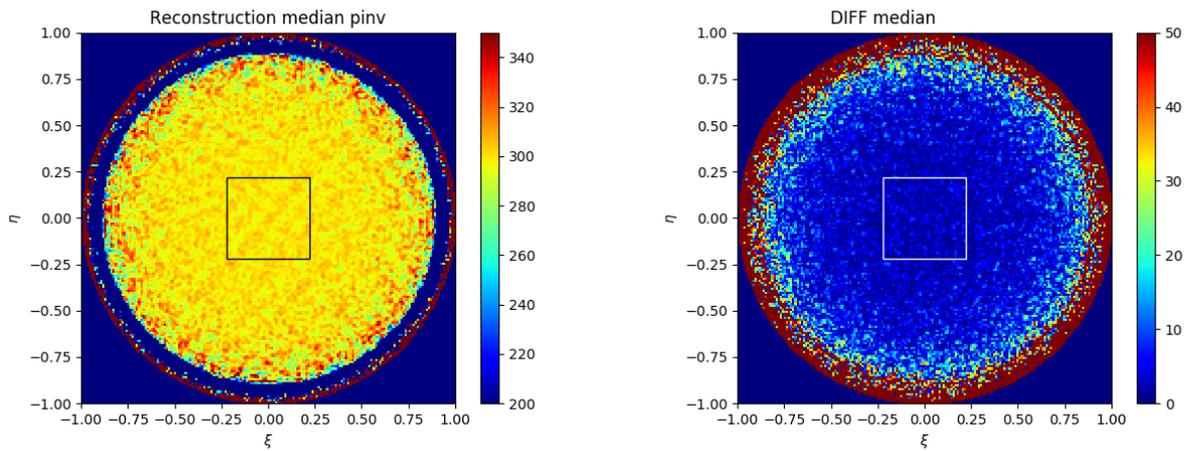


Figure 4.6: Reconstruction with the irregular square configuration and a body at 300 K. There is no observable folding, but a reconstruction error. Left: the reconstructed BT image. Right: the absolute difference between the reference and the reconstruction. The error increases as  $\|(\xi, \eta)\| \rightarrow 1$ . The absolute error measured within the black-square zone of the AF-FOV is 3.45 K, and the maximum absolute error is 10.98 K.

Instrument	RMSE-AF-FOV	Abs. max err. AF-FOV
Quincunx	1.86 K	6.03 K
Irregular square	3.45 K	10.98 K

Table 4.2: Summary of the evaluation of the quincunx and irregular square instruments for the constant (300 K) body scene.

When the sampling comes from a square configuration (regular or irregular) there is an unavoidable accumulation of visibilities in the horizontal and vertical axes. Since the number of visibilities is fixed, this necessarily means that some frequencies are missing in the rest of the spectrum.

Table 4.2 shows the summary of the evaluation of the quincunx and irregular square instruments. It shows:

- **RMSE-AF-FOV:** RMSE evaluated inside the white square show in figures 4.5 and 4.6.
- **Abs. max err. AF-FOV:** maximum absolute error evaluated in the square within the AF-FOV.

Note that the maximum absolute error is not a robust metric because of the random nature of the radiometric noise. Indeed, it is normal to find isolated reconstructed pixels with a large deviation from the expected value. The larger the reconstruction, the higher the probability to find such outliers. The RMSE is a better metric to evaluate the reconstructions, not that sensitive to the presence of extreme values.

About the minimal distance between antennas, in the case of the quincunx it is exactly  $d = \lambda$  since the antennas are equally spaced. For the irregular configuration, we computed it as the average of the median distances from one antenna to any other.

Since there is no folding in the reconstructed BT image obtained with the irregular instrument (but a global reconstruction error), it is possible to reconstruct the image close to the boundary when  $\|(\xi, \eta)\| \rightarrow 1$ . In Fig. 4.6 it can be observed that close to the boundary the error is large. Therefore, a legitimate question is whether this error is caused by the abrupt BT change (from 300 K to 0 K instantaneously) or on the contrary if it is due to the proximity to the boundary. To check out which is the cause, we added a square "hole" at 0 K inside. Fig. 4.7 shows the reconstruction and the absolute error. It can be observed that the large discontinuity causes an error of about 30 K and an oscillation pattern. However, it can be observed that the extent of the error caused by the discontinuity of

the square is smaller than the error in the boundary of the body at 300 K. Thus, we conclude that indeed the large discontinuity has an important impact in the reconstruction but that the large error observed in the boundary of the body at 300 K is caused by the combination of the discontinuity and the proximity to the zone where  $\|(\xi, \eta)\| \rightarrow 1$ , where the reconstruction should be discarded.

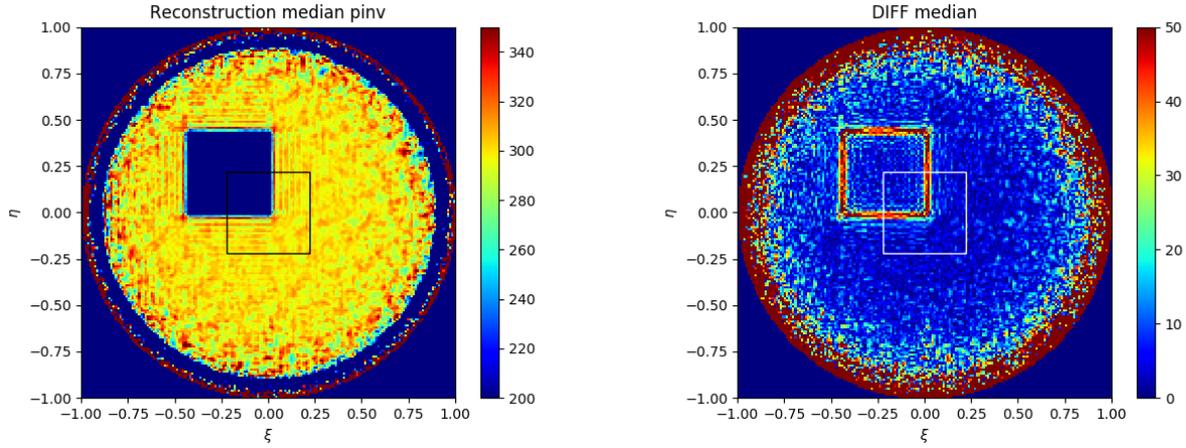


Figure 4.7: Reconstruction with the irregular square configuration and a body at 300 K with a "hole" added, to check out if the large error in the boundary is caused by the proximity to the boundary or if it is due to the large discontinuity from 300 K to 0 K. Left: the reconstruction. Right: the absolute error. It can be observed that the large discontinuity causes an error of about 30 K and an oscillation pattern.

Since the performance of the reconstruction depends on  $\|(\xi, \eta)\|$  we evaluated the RMSE in circular bands whose inner radius is  $r_1$  and the outer  $r_2$ . Table 4.3 shows the evaluation, with the RMSE within the band and the accumulated RMSE from  $r_1 = 0$  up to  $r_2$ .

#### 4.6.2 Evaluation with an image of the Earth

To evaluate with a non-constant image, we used an RGB scene of the Earth. This scene does not contain realistic values of BT, and it is used only to evaluate with a scene more realistic than the constant BT circle. We re-scaled the average of the three channels of the image so that the ocean has an average temperature of 280 K and the land a range of temperature that goes from 280 up to 344 K. This variability of temperatures in the land is larger to the real one, but nevertheless useful to evaluate the instruments in a difficult scenario.

Fig. 4.8 shows the reference image with and without applying the ARP. Fig. 4.10 show the reconstruction with the quincunx and irregular square instruments. Table 4.4 gives their evaluation.

Table 4.5 shows the evaluation by bands.

#### 4.6.3 Point spread function of the instruments

This section shows the PSF obtained with both instruments when observing a point source. We consider that the noise at the visibilities is an intrinsic characteristic of the instrument and thus it is added to the visibilities. Indeed, both the noise at the visibilities and the spatial folding are inherent to the quincunx instrument, independently from the scene. The irregular square instrument does not exhibit folding, but a reconstruction error that increases as  $\|(\xi, \eta)\| \rightarrow 1$  up to infinity. For both instruments, this reconstruction error is caused by the progressive loss of information caused by the ARP.

Fig. 4.11 shows the PSF of the quincunx and Fig. 4.12 the PSF of the irregular square. Both show a ring in when  $\|(\xi, \eta)\|$  is close to 1 since in that area the ARP has multiplied the signal almost by zero and therefore the reconstruction is impossible due to the presence of inherent radiometric noise.

$r_1$	$r_2$	RMSE $r_1 \rightarrow r_2$	RMSE $0 \rightarrow r_2$
0.00	0.03	4.31	4.31
0.03	0.07	2.98	3.64
0.07	0.10	3.24	3.51
0.10	0.14	3.07	3.40
0.14	0.17	3.26	3.37
0.17	0.20	3.43	3.38
0.20	0.24	3.65	3.42
0.24	0.27	3.31	3.41
0.27	0.31	3.30	3.39
0.31	0.34	3.66	3.42
0.34	0.37	3.89	3.46
0.37	0.41	3.84	3.49
0.41	0.44	4.21	3.55
0.44	0.47	4.41	3.61
0.47	0.51	4.55	3.67
0.51	0.54	4.97	3.75
0.54	0.58	5.06	3.83
0.58	0.61	5.68	3.93
0.61	0.64	6.24	4.06
0.64	0.68	7.11	4.21
0.68	0.71	7.51	4.37
0.71	0.75	9.98	4.62
0.75	0.78	11.69	4.93
0.78	0.81	14.37	5.32
0.81	0.85	19.02	5.87
0.85	0.88	31.08	6.84
0.88	0.92	71.05	9.22
0.92	0.95	92.77	12.20

Table 4.3: RMSE of the reconstruction of the scene of the body at 300 K with the irregular instrument, measured in bands with inner radius  $r_1$  and outer radius  $r_2$ . It shows the RMSE measured in the band as well as the RMSE from  $r_1 = 0$  up to  $r_2$ .

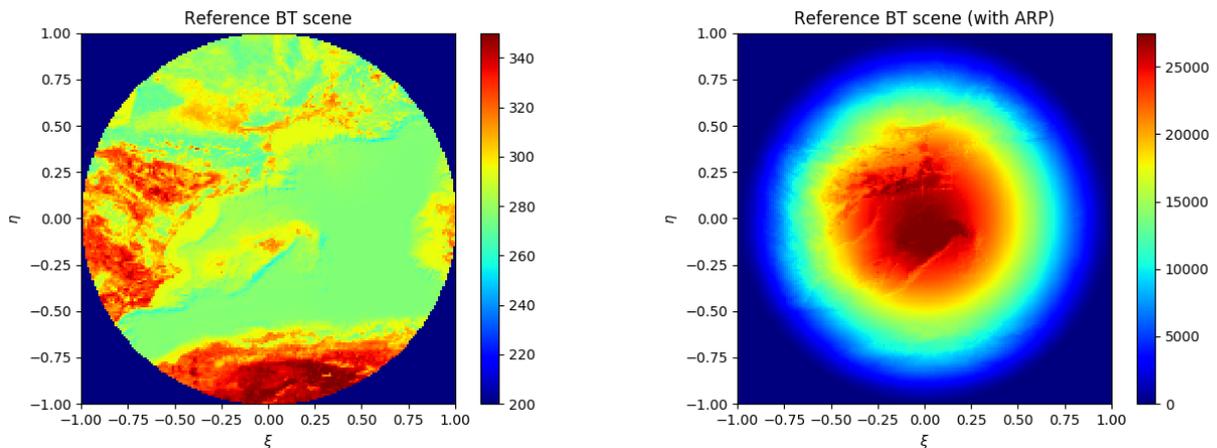


Figure 4.8: RGB scene of the Earth representing BT values. This scene does not contain absolutely realistic values of BT, and it is used only to evaluate with a scene more realistic than the constant BT circle. Left: reference BT image. Right: reference BT after applying the ARP.

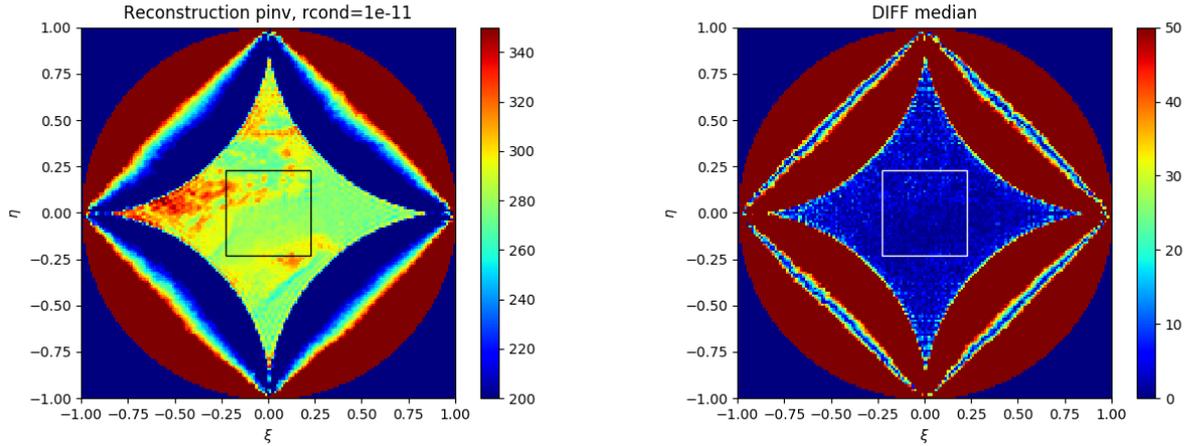


Figure 4.9: Reconstruction with the quincunx configuration and the image of the Earth shown in Figure 4.8. Left: reconstructed image. Right: absolute error. The RMSE within the black-square zone of the AF-FOV is 2.95 K, and the maximum absolute error is 13.11 K.

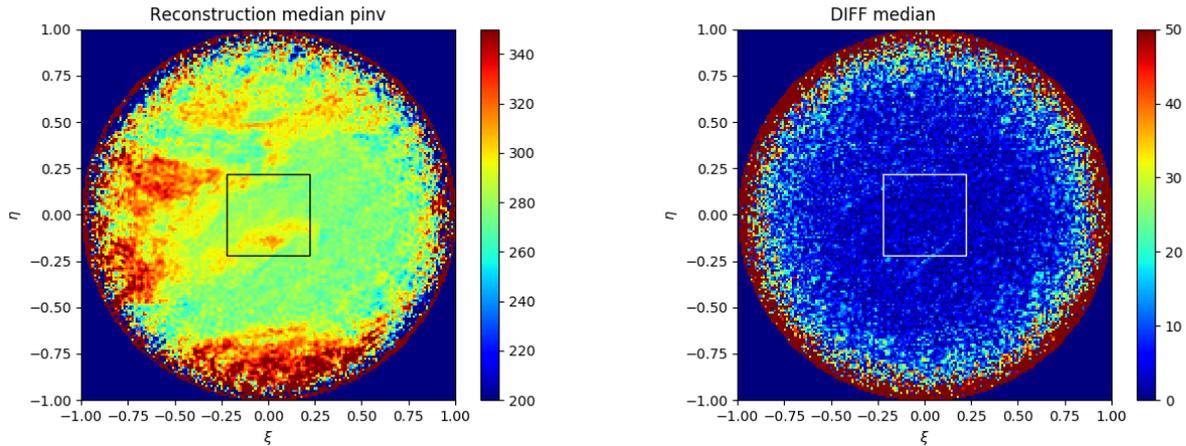


Figure 4.10: Reconstruction with the irregular square configuration and the image of the Earth shown in Figure 4.8. There is no observable folding, but a reconstruction error. Left: the reconstructed BT image. Right: the absolute difference between the reference and the reconstruction. The error increases as  $\|(\xi, \eta)\| \rightarrow 1$ . The RMSE within the black-square zone of the AF-FOV is 4.13 K, and the maximum absolute error is 15.76 K.

Instrument	RMSE-AF-FOV	Abs. max err. AF-FOV
Quincunx	2.95 K	13.11 K
Irreg. sq.	4.13 K	15.76 K

Table 4.4: Summary of the evaluation of the quincunx and irregular square instruments with the image of the Earth.

$r_1$	$r_2$	RMSE $r_1 \rightarrow r_2$	RMSE $0 \rightarrow r_2$
0.00	0.03	3.36	3.36
0.03	0.07	3.20	3.28
0.07	0.10	3.16	3.24
0.10	0.14	3.93	3.41
0.14	0.17	3.86	3.50
0.17	0.20	4.55	3.68
0.20	0.24	4.44	3.79
0.24	0.27	4.61	3.89
0.27	0.31	4.68	3.98
0.31	0.34	4.61	4.04
0.34	0.37	4.87	4.12
0.37	0.41	4.65	4.16
0.41	0.44	5.10	4.23
0.44	0.47	4.99	4.29
0.47	0.51	5.20	4.35
0.51	0.54	5.57	4.42
0.54	0.58	5.80	4.51
0.58	0.61	6.52	4.62
0.61	0.64	7.08	4.75
0.64	0.68	8.39	4.93
0.68	0.71	9.22	5.13
0.71	0.75	10.95	5.40
0.75	0.78	12.60	5.71
0.78	0.81	15.55	6.12
0.81	0.85	19.74	6.67
0.85	0.88	27.65	7.47
0.88	0.92	41.95	8.75
0.92	0.95	89.21	11.62

Table 4.5: RMSE of the reconstruction of the Earth BT scene with the irregular instrument, measured in bands with inner radius  $r_1$  and outer radius  $r_2$ . It shows the RMSE measured in the band as well as the RMSE from  $r_1 = 0$  up to  $r_2$ .

The irregular configuration exhibits a better PSF with respect to the quincunx, with a less evident cross-correlation pattern.

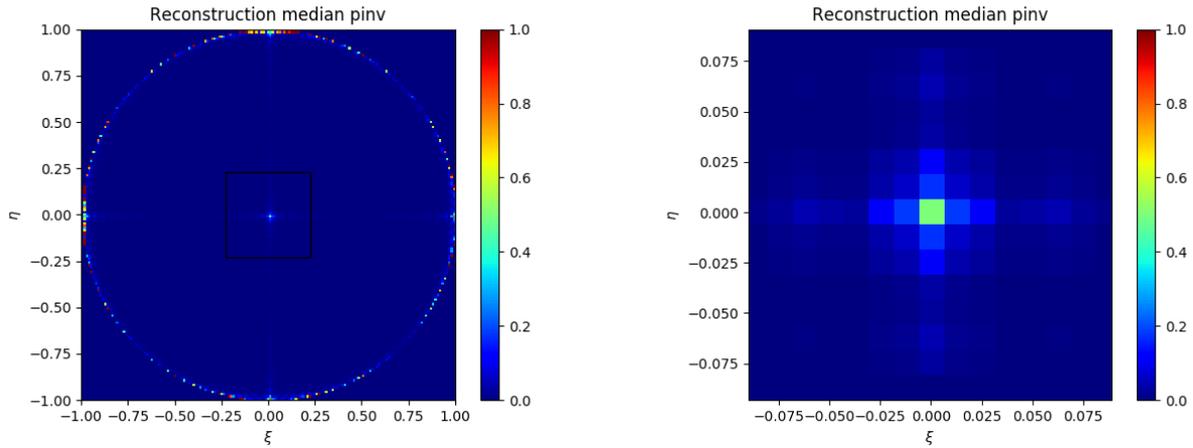


Figure 4.11: PSF of the quincunx instrument. On the left: PSF of the reconstructed image from the point source. On the right: detail of the PSF. A cross pattern can be observed at the center, stronger than the irregular configuration's PSF.

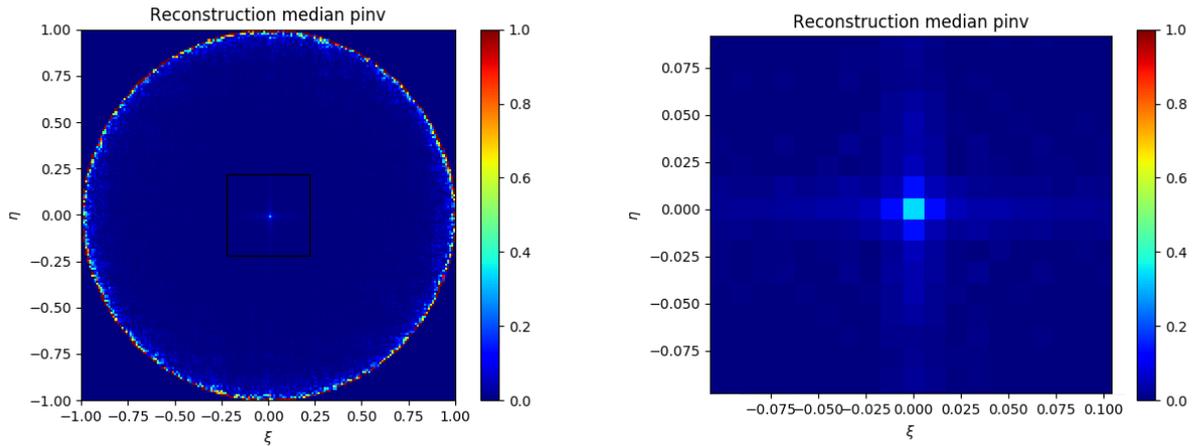


Figure 4.12: PSF of the irregular square instrument. On the left: PSF of the reconstructed image from the point source. On the right: detail of the PSF. This configuration shows a better PSF with respect to the quincunx, with a less evident cross pattern.

#### 4.6.4 Convolution kernels

We show the convolution kernels of both instruments. These kernels are obtained by setting all the visibilities to 1 and then reconstructing the corresponding BT image with Eq. (4.12). The kernel of the irregular configuration looks like noise, without any particular structure. The kernel of the quincunx clearly shows the folded areas. Note that the scale is different in both images. Fig. 4.13 shows the convolution kernels of the quincunx (left) and the calibrable irregular square (right).

## 4.7 Conclusions, perspectives, and future work

We have presented a comparison between the quincunx and the irregular square configurations of antennas. The reconstruction is performed by a pseudo-inverse with the Moore-Penrose algorithm. The scenes are multiplied by the ARP of the antennas, but no additional apodization (say, Hanning window or others) was applied. For the comparison we used two instruments of the same size and with the same number of antennas.

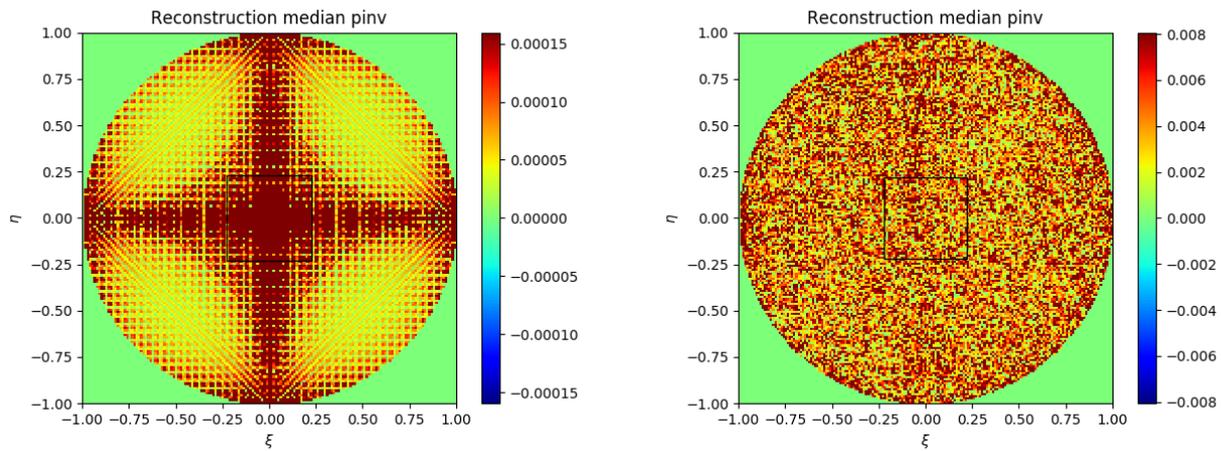


Figure 4.13: Left: convolution kernel of the quincunx. Right: convolution kernel of the irregular configuration. The kernel of the irregular configuration looks like noise, without any particular structure. The kernel of the quincunx clearly shows the folded areas. Note that the scale is different in both images.

In the case of the quincunx (regular sampling) we have classic aliasing surrounding the image (the same as in SMOS), and it is possible to find a small inner area which is alias-free. The error inside that alias-free zone that was small (1.86 K in the scene of piece-wise constant temperature and about 2.95 K for the scene of the Earth). The calibration approach to irregular configurations is different from regular configurations. In them some baselines are over-represented and instead some are critically under-represented. One of the reasons to use an irregular configuration is to reduce the number of duplicated visibilities. Yet one can easily perturb the irregular configuration to find a well-balanced set of replicated visibilities. This is done by progressively discretizing the position of the antennas, until the instrument can be calibrated.

With irregular sampling there is no completely folded area as in the regular case, but the reconstruction error is global. This reconstruction error (or irregular folding) consists of scaled replicas of the image added over itself multiple times. The error in the alias-free zone (the reference to locate this zone is the quincunx) in the irregular case was 3.45 K for the piece-wise constant image, and 4.13 K for the image of Earth. The BT image of the Earth we used was re-scaled in order that it had values similar to that of a real BT scene of the Earth.

We compared both configurations with a reduced number of antennas and a small instrument (thus, with a small spatial resolution), but nevertheless enough for the comparison. The quincunx has the advantage of a smaller reconstruction error in the AF-FOV zone but with large folding. The irregular configuration has a larger error, but without folding. To reach a spatial resolution of less than 10 Km a server with more than 128 Gb is needed to compute the pseudo-inverse directly with the Moore-Penrose method. We plan to obtain the required computational resources to compute a larger instrument.

The irregular reconstruction was performed by applying directly a pseudo-inverse to the visibilities, but this approach can be improved. As short term future work, we plan to improve the irregular reconstruction by adding weights which are proportional to the Voronoi cells corresponding to the sampling frequencies of each visibility in the  $u - v$  domain. Also, a de-convolution from the PSF or exploiting the knowledge of the convolution kernel is a hint for future research to improve the reconstruction from irregular samples.



## Part II

### Contributions on reproducible research



# Chapter 5

## Introduction

In 2009 David Donoho et al. point out that there is a *credibility crisis* [66] in scientific research, given that many researchers are unable to get the same results (published in well referenced journals) obtained by their colleagues. In 2011 the director of the oncology division at Amgen (an American biopharmaceutical company in California) tried to reproduce 53 of the most important oncology papers in the literature [67]. He failed to reproduce 47 of them. Bayer HealthCare in Germany confirmed recently that only about 25% of cancer research could be considered reproducible<sup>1</sup>.

In some disciplines (such as Biology) reaching reproducibility is not a simple task and indeed external conditions (for instance, the temperature of the room, the humidity, the health of the cell in a culture, or any unknown contamination, among others) might cause that the same experiments give different results. Because of that, biologists implement strict *protocols* that describe in detail a unique way to perform an experiment. Yet, it can be difficult to reproduce some results when they depend much on specific external conditions that may escape from the control of technicians and researchers. However, in computational sciences where the tools are *algorithms* and most of them are deterministic, in general the result depends only on the provided input. Therefore, there is no good excuse to reject a fully reproducible research. Signal processing (including image and video) is clearly one of the disciplines where full reproducibility can be reached, willing.

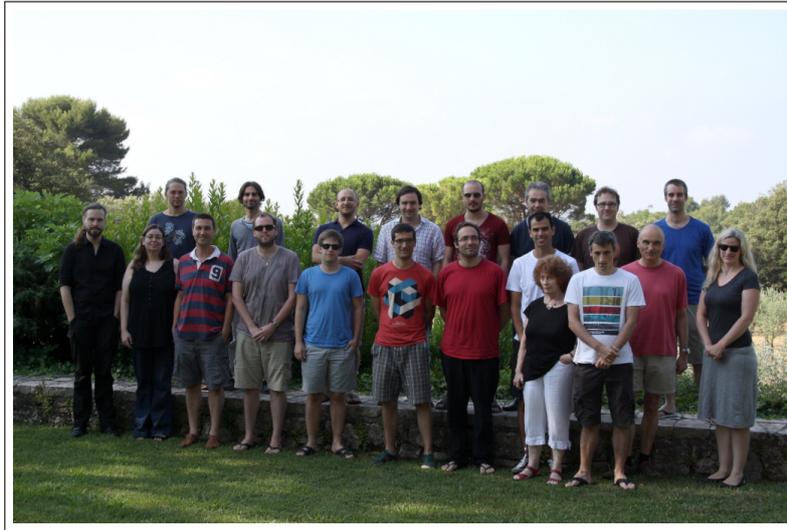
In 2010 the *Image Processing On Line* (IPOL) was funded after the initiative of Jean-Michel Morel, Nicolas Limare, and a group of founding editors (to which I proudly belong). The idea was simple: to create an image-processing journal where each of the articles describes with deep mathematical detail an algorithm. This way, any researcher could reproduce the results. The source code of the proposed method should also be published along the article and the reviewers would carefully check that the pseudo-code descriptions in the article indeed correspond exactly to what is implemented in the source code. Also, an online demonstration and one archive of experiments would come along with the article.

During his PhD dedicated to Reproducible Research in Image Processing [68], Nicolas Limare defined all the article's guidelines of the IPOL journal, its software guidelines, the list of accepted languages and libraries, the choice of the editorial software, a complete first version of the demo system, configured the mailing lists, helped authors to write their first articles, and instructed the first IPOL editors. It has always been really impressive to me that he managed to do *everything* without much help in a relatively short time!

After the first three years of IPOL working, the Editorial Board decided to discuss about the future of IPOL, including topics such as publishing image, sound, and video processing algorithms, 3D data, which languages and libraries should be allowed in IPOL, the review process, scalability of the demo system, speed-up of the demo creation process, and other relevant topics. The meeting took place from 22 to 26 of July 2013 at the idyllic estate of Fondation des Treilles in the south of France.

---

<sup>1</sup><https://www.nature.com/articles/nrd3439-c1>



Participants of the 2013 *Reproducible Research in Signal, Image, and Geometric Processing* meeting at Fondation des Treilles.

That meeting really convinced me that IPOL was indeed a *necessary* initiative and that even if it seemed much work ahead, it would be worth it. In 2015 I decided to dedicate part of my post-doc time to put in practice some of the ideas that were discussed in the 2013 meeting. I was especially worried (and that was a shared feeling with other IPOL editors) about the fact that creating a demo in IPOL was only possible to experts and that it took way too long. The editor was required to write a large Python interface over the IPOL demo system, to create HTML templates, and to code any interaction of the demo with the final users. When the system only had a few demos that was feasible, but soon it became a bottleneck. I believed in that defining a new demo should be as simple as declaring the number of inputs, the parameters (with their type, default value, and range), defining how to call the program along with the parameters, and what should be shown as results. The majority of demos could be defined with that minimal information, without the need of coding complicated interfaces.

There were some preliminary attempts to solve this problem, as the *automatic demo generator* proposed by Agustín Salgado in the 2013 meeting. The idea consisted in a tool that given a short description of the demo (the “DDL”, the Demo Description Lines) would write automatically the interfacing code. Indeed, that was a good idea in the right direction. I thought that the system should go a step further: it would not execute any interfacing code written by the editors (even if automatically generated). Instead, it should only read a DDL and create *on the fly* the demo pages. That would allow the editors to update the demos interactively in a simple way, and become accessible to non-expert authors.

However, doing so implied writing a whole new demo system and a dedicated development team. It was a very ambitious plan and despite this, or perhaps because of it, we decided to go on. I took the responsibility of creating a fully automatic new demo system, distributed on several server. I organized and supervised a team of students and engineers to that purpose. In February 2013 the first *commit*<sup>2</sup> was added to the GitHub repository<sup>3</sup>. Up to now, there are more than 3300 commits. So far the new demo system has reached many of the milestones and addressed almost all of the problems that were discussed in Les Treilles:

- Automatic audio and video demos are possible.
- New languages have been accepted, including MATLAB and Python (with support for virtual environments).
- The system is scalable. Now it supports to be distributed along several servers that can be connected/disconnected easily from the architecture. The bottleneck of having a unique server was solved as IPOL now uses a load balancer to decide which server should take care of an execution request.

<sup>2</sup>A contribution to the source code repository of a project.

<sup>3</sup><https://github.com/mcolom/ipolDevel>

- New demos can now be created in a few minutes. The DDL is now only a declarative description. Any demo support code is totally separated from the DDL and it can be uploaded as a separated package.
- An easy-to-use and editor-oriented tool was written to create and modify demos, the *Control Panel*.

I also participated to the improvement of the software guidelines of IPOL and simplified the review process.

Unlike the *heroic* work of Nicolas Limare, creating the new IPOL demo system from 2015 up to now has not been a single-person task, but that of a whole team. I supervised a large team of collaborators that are helping or have helped turning the designs of the new architecture into real working system.

The team is made from master interns, young or experienced engineers, to post-docs developing the IPOL demo system infrastructure and conceiving the scientific organization of IPOL papers. Here follows the complete list of engineers (alphabetical order) that I have supervised, their status when I advised them, and their current occupation:

- **Matías Abal.** Degree in Computer Science from ULPGC. IPOL engineer from 2017-03-29 to 2019-07-12. Now full-stack developer at Cycling Friendly. He worked in the development of the facility for video processing in IPOL, in particular the aspects of encoding, pre-processing/data transformation, and visualization. This has allowed the possibility of having online video demos in IPOL. Matías Abal discussed within our team and also with external senior researchers, as for example Javier Sánchez (from ULPGC) to implement video stabilization algorithms in IPOL.
- **Martín Arévalo.** *Tecnólogo en Informática* from UdelaR (Uruguay). Currently Instrumentation Technician at UdelaR. He continued the work of José Arrecio (see below). This was part of a scientific collaboration within the STIC AmSud program between France and participating research centers in South America. In this collaboration I supervised Martín Arévalo (and co-authors) to write the article *The IPOL Demo System: a Scalable Architecture of Microservices for Reproducible Research*. From Dec. 2016 to Mar. 2017.
- **José Arrecio.** Master's in Computer Science from UIB. Now senior software engineer at Everis Chile. His main task in IPOL was to solve a severe bottleneck in IPOL: creating new demos was extremely complex. He developed the first version of the *IPOL Control Panel* that now allows editors to create new demos in a few minutes. From Nov. 2015 to Feb. 2017.
- **Carlos Escobar.** Degree in Computer Science from ULPGC. Now Master's in Software and Systems from UPM (Spain). He worked mainly on a severe problem of the first version of the IPOL demo system: a single machine can run out of resources (allowed execution time, available RAM) when several executions are requested at the same time. Carlos Escobar worked in sharing the workload along different machines, thus allowing IPOL to have hundreds of concurrent executions per days with the possibility of adding/removing servers on demand. From Sep. 2016 to Jun. 2017.
- **Vincent Firmin.** Intern student from Epitech at CMLA. Now Master in Computer Science and working as devOps engineer at Healsy. He was the first contributor in the new architecture of microservices of IPOL and wrote the *Archive* module. It manages the experiments performed by the IPOL users and nowadays more than 500,000 experiments with their corresponding metadata are accounted. This has a large impact for the scientific community using IPOL, since it allows to quickly asses the real state of the art (which does not necessarily coincide with what the literature establishes), and which methods have a larger impact. From Apr. to Jul. 2015.
- **Frédéric Glorieux.** *Master de Lexicographie, Linguistique et Informatique* from Université Lille 3. Frédéric Glorieux had an interesting profile combining computer science and text analysis, and helped improve the Archive module started by Vincent Firmin. His prior knowledge on text

analysis was useful to better structure the data in the Archive. This has attracted the attention of the machine learning group at CMLA (Nicolas Vayatis, in particular), who indeed saw a scientific interest in exploiting the large amount of data in the archive to extract meaningful information. An MVA internship on Machine Learning is envisaged. From Jun. 2017 to Sep. 2018.

- **Stéphane Gratias.** Master's student at Université de Technologie de Troyes. He improved largely during his internship the IPOL Control Panel started by José Arrecio and Martín Arévalo. This has made even faster and ergonomic for IPOL editor to create new demos. From Jan. to Jul. 2018.
- **Karl Krissian.** Postdoc at CMLA. Now senior software engineer at GoPro. He worked on a solution for the problem of defining an online algorithm with the minimal required information from the editors. The solution is what we call now the *Demo Description Lines* (DDL), as general as possible abstract specification for the demos. Before the availability of the DDL the editors had to code their own interfaces, a paramount bottleneck for the evolution of IPOL. Thanks to Karl Krissian's work now IPOL can create online interfaces for the algorithms submitted by the journal's authors in a few minutes. As far as we know, this is an unique feature compared to other online execution platforms competitors as IEEE's Code Ocean, for example. From Jun. 2015 to Oct. 2016.
- **Héctor Macías.** Degree in Computer Science from ULPGC. IPOL engineer from 2017-03-29 (ongoing). He is working on the development of interactive controls in IPOL. They will allow IPOL to publish articles that implement method such as Poisson editing, lens correction, or inpainting, among others. The interactive controls are generic enough (masks, points, open or closed polygons) to allow deploying online a large variety of articles. The M1 students at ENS Paris-Saclay will benefit from the new controls, since they are currently used (a first prototype) in the TPs of the *Introduction mathématique au traitement et à l'analyse des images digitales et à leurs surprenantes applications* course.
- **Yanis Mammar.** Ongoing engineering studies at Epita. He worked on the migration of demos from the old IPOL demo system to the current one. This implied writing tools to port the contents of the archived experiments in the previous system to the corresponding module in new architecture. Also, writing the corresponding DDL and testing the demos in their new environment. Short 1-month internship.
- **Alexis Mongin.** Master's in Computer Science from Epitech. Now software engineer at YBC-Data. As well as Vincent Firmin, he worked when the architecture of IPOL was just started. During his internship he wrote the first IPOL module (*Blobs*), an abstraction that allows IPOL to treat any datatype as a single entity that can be pre-processed, converted, or archived along the IPOL's standard pipeline. This has allowed IPOL to expand its application field not only to images, but to video, sound or 3D processing. From Oct. 2016 to Mar. 2017.
- **Nelson Monzón.** PhD candidate at ULPGC. Now postdoctoral researcher at CMLA. The role of Nelson Monzón is double. On one hand he has contributed to the development of the platform itself, but more interestingly his scientific profile has allowed the design of research articles on optical flow and video processing. From Nov. 2015 (ongoing).

For all of them the **supervision rate** is **100 %**. Although I had already supervised other software-engineering projects in the past, their reach cannot be compared with the large impact of IPOL, with thousands of executions daily and millions of results archived. It is the most long-term project and with the largest team that I have ever supervised. During these years I have learned how to deal with a team of engineers with their own strong ideas and how to reach agreements when we have different technical points of views. Most of the time decisions were made in common in our weekly meetings, where every member of the team exposes the problem and proposed solutions. Although, sometimes I have to force particular design decisions since I have a more complete and general vision

of the project<sup>4</sup>. At the same time I perceive designing and developing the IPOL system as a task of high responsibility, the very professional work of the many members of the team have turned it into a really satisfying common endeavour.

In 2011 the first articles got published and today in 2019 IPOL continues to publish articles and it has become a reference in the field of image processing. IPOL journal publishing research articles since 2011 and as of 2019, it is the only research journal publishing verified implementations. 158 articles have been published, 6 other articles are accepted and in press, and 18 preprints are available. At this moment the archives store about 250,000 experiments performed with uploaded original data and the total number of online demo executions by the system exceeds 1,000,000. IPOL is a registered journal with ISSN (2105-1232) and DOI identifiers (10.5201/ipol), indexed by SCOPUS and within the Thomson-Reuters Emerging Sources Citation Index. IPOL is steered by the CMLA image processing group with an international editorial board and an agreement with SIAM Journal on Imaging Science for publications of pairs of articles, the theoretical part published in SIIMS and the implementation part in IPOL.

In this project I had the pleasure to organize and to be a chair in the two first editions of the Workshop on Reproducible Research in Pattern Recognition (RRPR) with Bertrand Kerautret and Pascal Monasse. The first edition on 2016 in Cancún (México) and the second in 2018 in Beijing (China). We published two books containing the proceedings (see section 1.7.1 for the details).

I have chosen to include in this report a summary of two published articles given that they reflect well the evolution of the platform. The first one, summarized in chapter 6, was published in 2016 when the architecture of the platform got completed and the first articles were published. It describes the current state of the IPOL demo system. On the other hand, chapter 7 discusses on the work-in-progress extension the current IPOL platform into a system capable of running general machine learning applications with heterogeneous data. The same way the idea of developing in 2015 a brand new system could be thought as too much ambitious (and now it is the production's IPOL demo system), I hope that we will manage to have the next version of the platform available in the following years.

---

<sup>4</sup>Most of the members of the engineering team are not necessarily formed in Image Processing or even in general Signal Processing. Indeed, after some years they reach a working knowledge of the state of the art of the discipline, but normally their specialty is mainly on general Computer Science.



# Chapter 6

## The present. IPOL: an online architecture for reproducible research

This chapter is a summary of the research and a description of the actual development tasks that followed the Les Treilles meeting in 2013, that led to the current IPOL demo system. It was published in [69].

### Abstract

We identified design problems related to the architecture, ergonomics, and performance in the previous version of the Image Processing on Line (IPOL) demonstration system. In order to correct them we moved to an architecture of microservices and performed many refactorings. This article first describes the state of the art in Reproducible Research platforms and explains IPOL in that context. The specific problems which were found are discussed, along with the solutions implemented in the new demo system, and the changes in its architecture with respect to the previous system. Finally, we expose the challenges of the system in the short term.

### 6.1 Introduction

Image Processing on Line (IPOL) is a research journal started in 2010 on Reproducible Research in the field of Signal Processing (mainly Image Processing, but also video, sounds, and 3D data), giving a special emphasis on the role of mathematics in the design of the algorithms [68]. This article discusses the current system after the changes that were anticipated in [70], and towards which direction it plans to move in the future.

As pointed by Donoho et al. [66], there is a crisis of scientific credibility since in many published papers it is not possible for the readers to reproduce exactly the same results given by the authors. The causes are many, including incomplete descriptions in the manuscripts, not releasing the source code, or that the published algorithm does not correspond to what actually is implemented. Each IPOL article has an online demo associated which allows users to run the algorithms with their own data; the reviewers of the IPOL articles must carefully check that both the description and the implementation match.

Since it started in 2010, the IPOL demo system has been continuously improved and according to usage statistics collected along these years, it has about 250 unique visitors per day. However, several problems of design and potential improvement actions were identified and, on February 2015, it was decided to build a second version of the system based on microservices [71]. Among these problems can be listed: the lack of modularity, tightly-coupled interfaces, difficulties to share the computational load along different machines, or complicated debugging of the system in case of malfunction.

The plan of the article follows. Sect. 6.2 discusses the state of the art in Reproducible Research and microservices platforms. Sect. 6.3 discusses the particularities of IPOL as a journal, and Sect.

6.4 presents the architecture of microservices of the new IPOL demo system. Sect. 6.5 reveals the software development methodologies in the software engineering process the IPOL team is applying internally. Sect. 6.6 refers to a particular tool we designed for the IPOL editors which allows them to manage the editorial process. Sect. 6.7 presents a very important novelty of the new system, which is the capability of quickly creating new demos from a textual description. Finally, Sect. 6.8 presents the conclusions.

## 6.2 State of the art in reproducible research platforms

Some very well-known platforms whose use is closely related to Reproducible Research exist nowadays. Some of them are domain-specific while others are more general.

In the case of Biology, the Galaxy project [72] is a platform for genomic research which makes available tools which can be used by non-expert users too. Galaxy defines a *workflow* as a reusable template which contains different algorithms applied to the input data. In order to achieve reproducibility the system stores: the input dataset, the tools and algorithms which were applied to the data within the chain, the parameters, and the output dataset. Thus, performing the same workflow with the same data ensures that the same results are obtained given that the version of all the elements is kept the same.

In the field of Document Image Analysis, the Document Analysis and Exploitation platform (DAE) was designed to share and distribute document image with algorithms. Created from 2012, the DAE platform also adds tools to exploit annotation and perform benchmarking [73].

Generic tools for Reproducible Research include the IPython tool and its notebooks. This mature tool created in 2001 allows to create reproducible articles by not only editing text in the notebook, but allowing code execution and creating figures *in situ*. This approach follows closely the definition of a "reproducible scientific publication" given by Claerbout and followed also by Buckheit and Donoho: *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* [74].

In 2014 the Jupyter project was started as a spin-off of IPython in order to separate the Python language part of IPython from all the other functionalities needed to run the notebooks, such as the notebook format, the web framework, or the message protocols. IPython turns then into just another computation kernel for Jupyter, which nowadays supports more than 40 languages that can be used as kernels<sup>1</sup>.

There are also other generic tools which can be seen as *dissemination* platforms since their main objective is to make source code and data widely available to the public. In this category we find for example Research Compendia<sup>2</sup> focused on reaching reproducible research by storing data, code, in a form that is *accessible, traceable, and persistent*, MLOSS<sup>3</sup> for machine learning, datahub<sup>4</sup> to create, register, and share generic datasets, and RunMyCode<sup>5</sup> to associate code and data to scientific publications. Compared to these platforms, IPOL differs from them in the sense that it is a peer reviewed journal, and not only a dissemination platform.

Regarding the system architecture of IPOL, it is built as a Service-Oriented Architecture (SOA) made of microservices. This type of architecture allows IPOL to have simple units (called *modules* in its own terminology) which encapsulate isolated high-level functions (see in Sect. 6.4.1). Specifically, we use the CherryPy framework to provide the REST HTTP [75] services. Microservices in distributed system are especially useful for those system which need to serve millions of simultaneous requests. A good example of SOAs made of microservices is the Amazon AWS API Gateway<sup>6</sup> used by millions

<sup>1</sup><https://github.com/ipython/ipython/wiki/IPython-kernels-for-other-languages>

<sup>2</sup><http://104.130.4.253/>

<sup>3</sup><http://mloss.org>

<sup>4</sup><https://datahub.io/>

<sup>5</sup><http://www.runmycode.org/>

<sup>6</sup><https://aws.amazon.com/api-gateway>

of users. Also, multimedia streaming services such as Netflix<sup>7</sup> which receives about two-billion daily requests or Spotify<sup>8</sup> are usually based on SOAs of microservices.

## 6.3 IPOL as a peer-reviewed scientific journal

IPOL is a scientific journal on mathematical signal processing algorithms (image, video, audio, 3D) which focuses on the importance of reproducibility. It differs from other classic journals in its editorial policy: each IPOL article must present a complete description of its mathematical details together with a precise explanation of its methods with pseudo-codes. These ones must describe exactly the implementation that achieves the results depicted in the paper. The idea is that readers with sufficient skills could implement their own version (in any programming language or environment) from the IPOL article. Furthermore, submitting an IPOL paper means to upload the manuscript coupled with the original source codes. Both are reviewed in depth by the referees to ensure the quality of the publication and that the pseudo-codes match exactly with the attached program, before the editor's decision. The publication process is divided in two stages: first, the reviewers evaluate the scientific interest, the experiments and the reproducibility of the work; secondly, if this evaluation is positive, the authors submit the original code and the online demo is published.

Each IPOL article contains [68]:

1. A description of *one algorithm* and its *source code*;
2. a PDF article associated with an *online demonstration*;
3. *archived experiments* run by users.

All these data, accessible through different tabs of the article webpage, make IPOL an open science journal in favor of reproducible research. The philosophy of the journal follows the guidelines on reproducible research topics, by obeying the standards of reproducible research [76, 77]. This is meant as an answer to the credibility crisis in scientific computation pointed out by Donoho et al. [66].

IPOL publishes algorithms along with their implementation, but not compiled/binary software. Neither is it a software library, since each code must have minimal dependencies. The objective of IPOL is not simply to be a software or code diffusion platform. In this sense, the code must be as transparent to the reader as possible, not using implementation tricks unless they are described in the article. It should be seen as a reference implementation, always preferring clarity over run time optimization.

The current form of an IPOL article is illustrated in Fig. 6.1. The first tab (a) presents the links to the low- and high-resolution PDF manuscripts, and also to the reference source code. An embedded PDF viewer presents a preview of the manuscript. The second tab (b) is the interface of the demonstration system, proposing some illustrative input data. The user can also upload its own data from this page. Clicking on one proposed input dataset or uploading a new one brings to a page presenting a list of adjustable parameters of the algorithm, and possibly an image selection tool, used for example for cropping an image too big for real-time processing by the system (almost all demonstrations are expected to achieve their processing in at most 30 seconds). A click on the "Run" button brings to a waiting page, while the server runs the author's code, which finally updates into a webpage showing the results. At this stage, the user has the option to re-run on the same input data but modifying the parameters, or to change the input data. Running the algorithm on newly uploaded input data proposes to archive them and their results. The archived data of tab (c) in Figure 6.1 have permanent URL. This facilitates online communication between distant collaborators working on an algorithm. The archived data allow to understand what usages are aimed at by visitors, can reveal failure cases not anticipated by the authors, etc. The amount of archived data can also serve as a crude measure of the interest an algorithm raises in the community, as a kind of substitute or complement to the number of citations in a standard journal. The most cited IPOL articles have also tens of thousands of archived online experiments.

<sup>7</sup><https://media.netflix.com/en/company-blog/completing-the-netflix-cloud-migration>

<sup>8</sup><http://es.slideshare.net/kevingoldsmith/microservices-at-spotify>



Figure 6.1: The current form of an IPOL article with its three tabs: (a) article with manuscript, online viewer, and link to source code, (b) demonstration system, and (c) archived experiments.

Each IPOL demo downloads and compiles by itself the source code. This ensures that the users can reproduce exactly the results claimed by the authors. However, the authors of the demo can additionally add scripts or data, which is not peer-reviewed, but is needed to show the results in the website. This allows to avoid mixing the peer-reviewed source code of the submitted method with support extra codes needed only by the demo. This approach differs from classic publishing, where the method and some details about the implementation are usually described but it is not possible to reproduce and thus confirm the published results.

Apart from this specific form of the articles, IPOL presents the same aspects as a classic scientific journal, with an editorial committee, contributors in the form of authors and editors, a reviewing process, an ISSN, a DOI, etc. Some special issues are proposed, for example some selected papers from the 16th IAPR International Conference on Discrete Geometry for Computer Imagery (DGCI) in 2011. There is also an agreement for publishing companion papers in SIAM Journal of Imaging Sciences (SIIMS) and IPOL, the first submission concentrating on the theory and general algorithm and the second one on practical implementation and algorithmic details. Note that originality of the algorithm is *not* a prerequisite for IPOL publication: the usefulness and efficiency of an algorithm are the decisive criteria. IPOL articles are indexed by all major indexers, such as Scirus, Google Scholar, DBLP, SHERPA/RoMEO, CVonline, etc.

The role of the reviewer is not restricted to the evaluation of the manuscript. The reviewers are also expected to test the online demonstration, check the algorithmic description in the article, and the source code. Most importantly, they must verify that the description of the algorithm and its implementation code match. An important requirement is that the code be readable and well documented.

## 6.4 IPOL's system architecture

The architecture of the new IPOL demo system is an SOA based on microservices. This change was motivated by the problems found in the previous version of the demo system. First, it was designed as a monolithic program<sup>9</sup> which made it quite easy to deploy in the servers and to run it locally, but at the cost of many disadvantages. Given that it was a monolithic system, it was difficult to split it into different machines to share the computational load of the algorithms being executed. A simple solution would be to create specialized units to run the algorithms and to call them from the monolithic code, but this clearly evokes the first step to move to a microservices architecture. Indeed, this first step of *breaking the monolith* [71] can be iterated until all the functions of the system have been delegated in different modules. In the case of IPOL, we created specialized modules and removed the code from the monolith until the very monolith became a module itself: the Core. This Core module is in charge of all the system and delegates the operations to other modules. Fig. 6.2 summarizes the IPOL modules and other components of the system.

Other problems we had in the previous version of the demo system got solved when we moved to the

<sup>9</sup>Of course, with a good separation of functionality among different classes.

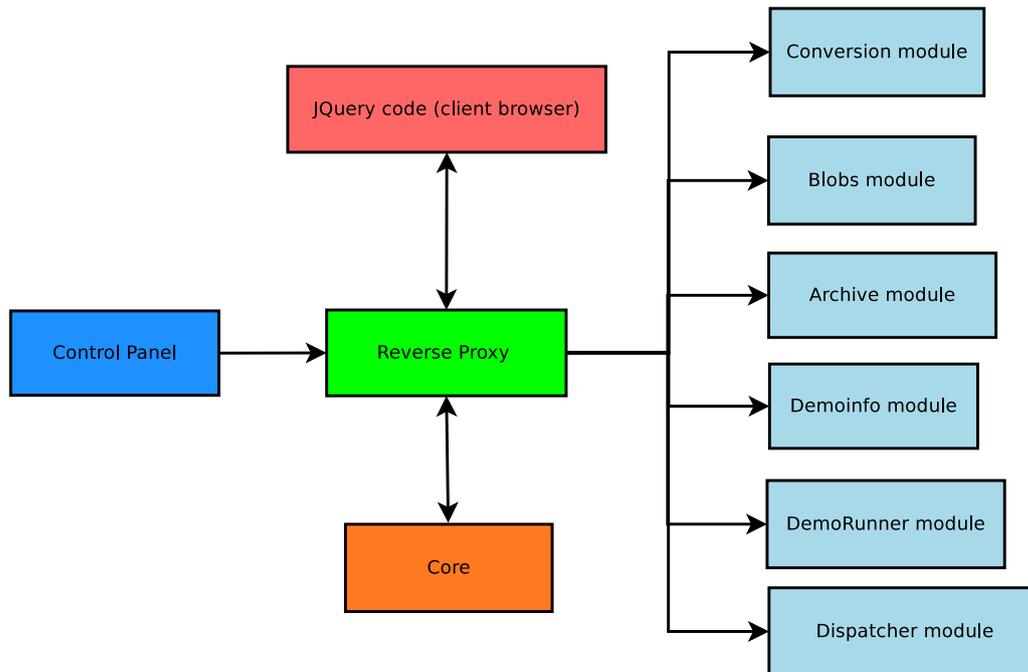


Figure 6.2: IPOL as a modular system.

microservices architecture. Since there is a loose coupling between the Core and the other modules, different members of the development team can work at the same time without worrying about the implementation details or data structures used in other parts of the system. Also, tracking down malfunctions is easier: since the Core centralizes all the operations, when a bug shows it can only be generated either at the Core or at the involved module, but not at any other part of the system. In the old system a bug could be caused by complex conditions which depend on the global state of the program, making debugging a complex task. And as noted before, the fact that the architecture of the system is distributed and modular by design makes it very natural and simple to have mechanisms to share the computational load among several machines.

Hiding the internal implementation details behind the interfaces of the modules is an essential part of the system, and it is needed to provide loose coupling between its components. The internal architecture of the system is of course hidden from the users when they interact with the system, but it is also hidden *from the inside*. This means that any module (the Core included) does not need to know the location of the modules. Instead, all of them use a published API.

Once the API is defined, the routing to the modules is implemented by a reverse proxy<sup>10</sup>. It receives the requests from the clients according to this pattern: `/api/<module>/<service>` and redirects them to the corresponding module. Fig. 6.3 shows how the API messages received by the proxy are routed to the corresponding modules, thus hiding the internal architecture of the system.

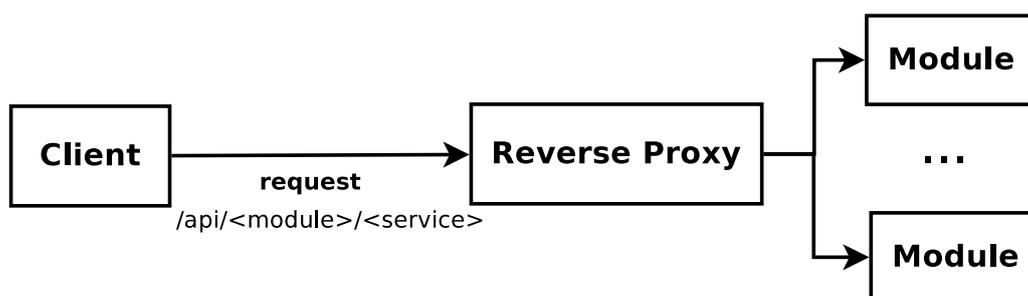


Figure 6.3: The reverse proxy routes the API messages to the corresponding modules.

<sup>10</sup>We use Nginx as the reverse proxy.

### 6.4.1 IPOL's demo system modules

The IPOL demo system is made of several standalone units used by the Core module to delegate specialized and well isolated functions. This section describes briefly these microservices modules.

#### Archive

The archive module stores all the experiments performed by the IPOL with their original data. The database stores the experiments and blobs, which are related with a junction table with a many-to-many relationship. It is worth noting that the system does not save file duplicates of the same blob, but detects them from their SHA1 hash.

This module offers several services, such as adding (or deleting) an experiment or deleting all the set of experiments related to a particular demo. The archive also has services to show particular experiments or several pages with all the experiments stored since the first use of the archive.

#### Blobs

Each demo of IPOL offers the user a set of default blobs which can be tagged and linked to different demos. Thus, the users are not forced to supply their own files for the execution of the algorithms. This module introduces the concept of *templates*, which are sets of blobs which can be associated to a particular demo. For example, this allows all the demos of an specific type (e.g., denoising) to share the same images as default input data. Instead of editing each demo one by one, the editors can simply edit their template to make changes in all the demos, and then particular changes to each specific demo.

#### Core

This module is the centralized controller of the whole IPOL system. It delegates most of the tasks to the other modules, such as the execution of the demos, archiving experiments, or retrieving metadata, among others.

When an execution is requested, it obtains first the textual description of the corresponding demo by using the Demo Description Lines (DDL) from the DemoInfo module and it copies the blobs chosen by the users as the algorithm's input. Then, it asks for the workload of the different DemoRunners and gives this information to the Dispatcher module in order to pick the best DemoRunner according to the Dispatcher's selection policy. The Core asks the chosen DemoRunner to first ensure that the source codes are well compiled in the machine and then to run the algorithm with the parameters and inputs set by the user. The Core waits until the execution has finished or a timeout happens. Finally, it delegates into the Archive module to store the results of the experiment. In case of any failures, the Core terminates the execution and stores the errors in its log file. Eventually, it will send warning emails to the technical staff of IPOL (internal error) or to the IPOL editors of the article (compilation or execution failure).

#### Dispatcher

In order to distribute the computational load along different machines, this module is responsible of assigning a concrete DemoRunner according to a configurable policy. The policy takes into account the requirements of a demo and the workload of all the DemoRunners and returns the DemoRunner which best fits. The DemoRunners and their workloads are provided by the Core. Figure 6.4 shows the communication between the Core, Dispatcher, and the DemoRunner modules.

Currently the Dispatcher implements three policies:

- **random:** it assigns a random DemoRunner
- **sequential:** it iterates sequentially the list of DemoRunners;
- **lowest workload:** it chooses the DemoRunner with the lowest workload.

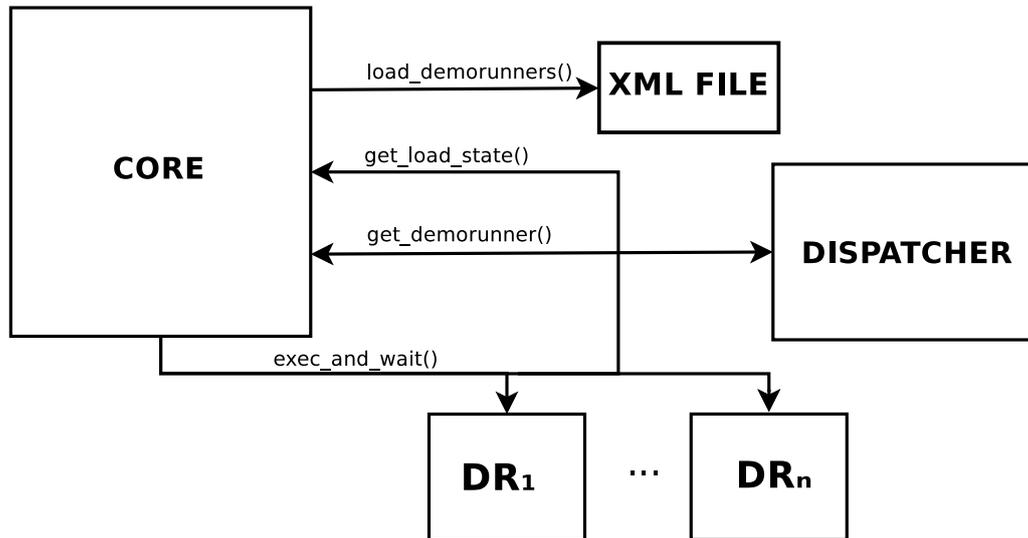


Figure 6.4: Communication between the Core, Dispatcher, and the DemoRunner modules.

Any policy selects only the DemoRunners satisfying the requirements (for example, having MATLAB installed, or a particular version of openCV).

### DemoInfo

The DemoInfo module stores the metadata of the demos. For example, the title, abstract, ID, or its authors, among others. It also stores the abstract textual description of the demo (DDL). All this information can be required by the Core when executing a demo or by the Control Panel when the demo is edited with its website interface.

It is possible that the demo requires non-reviewed support code to show results. In this case, the demo can use custom scripts to create result plots. Note that this only refers to scripts and data which is not peer-reviewed. In case they are important to reproduce the results or figures in the article, they need to be in the peer-reviewed source code package.

### DemoRunner

This module controls the execution of the IPOL demos. The DemoRunner module is responsible of informing the Core about the load of the machine where it is running, of ensuring that the demo execution is done with the last source codes provided by the authors (it downloads and compiles these codes to maintain them updated), and of executing the algorithm with the parameters set by the users. It takes care of stopping the demo execution if a timeout is reached, and to inform the Core about the causes of a demo execution failure so the Core can take the best action in response.

## 6.5 Software engineering in the IPOL demo system

The current IPOL project tries to follow the best practices in software engineering. Specifically, for this kind of project we found that Continuous Integration was a good choice in order to achieve fast delivery of results and ensuring quality. Continuous Integration is a methodology for software development proposed by Martin Fowler, which consists in making automatic integrations of each increment achieved in a project as often as possible in order to detect failures as soon as possible. This integration includes the compilation and software testing of the entire project.

It is a set of policies that, together with continuous deployment, ensures that the code can be put to work quickly. It involves automatic testing in both integration and production environments. In this sense, each contribution in the IPOL system is quickly submitted and several automatic test are performed. If any of these tests fail the system sends an email indicating the causes. Another advice of Continuous Integration is minimal branching. We use two. On one hand, master is the default

branch and where all the contributions are committed. It is used for the development, testing and this continuous integration; on the other hand, the prod branch is used only in the production servers. It is merged with master regularly. We use two different environments: integration and production. The integration server is where the master branch is pulled after each commit. The prod branch is used for the production servers and the code in this branch is assumed to be stable. However, the code in the integration server is also assumed to be stable and theoretically the code in the master branch could be promoted to production at any time once it has been deployed to the integration server and checked that it is fully functional and without errors.

Quality is perhaps the most important requirement in the software guidelines of the IPOL development team. The code of the modules must be readable and the use of reusable solution is advised [78]. The modules must be simple, well tested and documented, with loose interface coupling, and with proper error logging. Note that it is not possible to ensure that any part of the IPOL will not fail, but in case of a failure we need to limit the propagation of the problem through the system and to end up with diagnostic information which allows to determine the causes afterwards. Refactoring [79] is performed regularly and documentation is as important as the source code. In fact, any discrepancy between the source code and the documentation is considered as a bug.

### 6.5.1 Tools

The IPOL development team has created so far three main tools for the system administrators, developers, and editors to interact with the IPOL system. Some of their capabilities might be duplicated or overlapping with the Control Panel (for example, reading and modifying the DDL of the demos is a function implemented in the DDL tool and in the Control Panel, but they are still useful to perform massive changes or to automatize tasks).

#### Terminal

The Terminal is a small Python standalone application intended for system administrators which allows to start, stop, and query the status of each module. The current list of commands is: **start**: launches the specified module; **ping**: checks if the module is up; **shutdown**: stops the specified module; **info**: prints the list of available commands of the module; **modules**: displays the list of the modules of the system.

#### Control Panel

The Control Panel web application offers a unified interface to configure and manage the platform. It provides a navigation menu option for each module, which allows the editors to edit the demos or the modules directly (say, the add or remove images of a demo, or to delete experiments from the Archive upon request). Look at Sect. 6.6 for more information on the Control Panel.

#### DDL Tool

This tool is a standalone Python script which allows to read and write the DDLs of each demo. The main justification for this tool is to perform massive changes in the DDLs and automatize some needed tasks. It admits the following list of commands. **Read**: downloads the DDLs of the specified demos; **Read all**: downloads the DDLs of all the demos known by the system; **Write**: uploads the content of the file with the DDLs of the specified demos.

## 6.6 Editorial management: the Control Panel

The Control Panel is a Django web application which offers a unified interface to configure and manage the platform. Its graphical interface gives to the editors a menu with options to access the different modules available on the system. It provides many editing options to the users. The first option is the Status, that shows a list of the modules with summarized information about them, allowing the user to monitor if they are currently running. In second place there is an Archive Module

option to provide a list of the demos with stored experiments, as a result of an execution with original data. It allows the editor to remove specific experiments upon request (e.g., inappropriate images). There is also a Blobs Module option, which allows to add and remove blobs for a particular demo.

Additionally, the DemoInfo Module option permits the user to access information about the demos, authors and editors stored on the IPOL demo system, organized in three sections. The Demos section is the option selected by default, and makes it possible to edit the demo metadata, such as its ID, title, or the source code URL, the assigned editors, or its support scripts, among others. Fig. 6.5 shows a screen capture of the Control Panel application as shown in the browser.

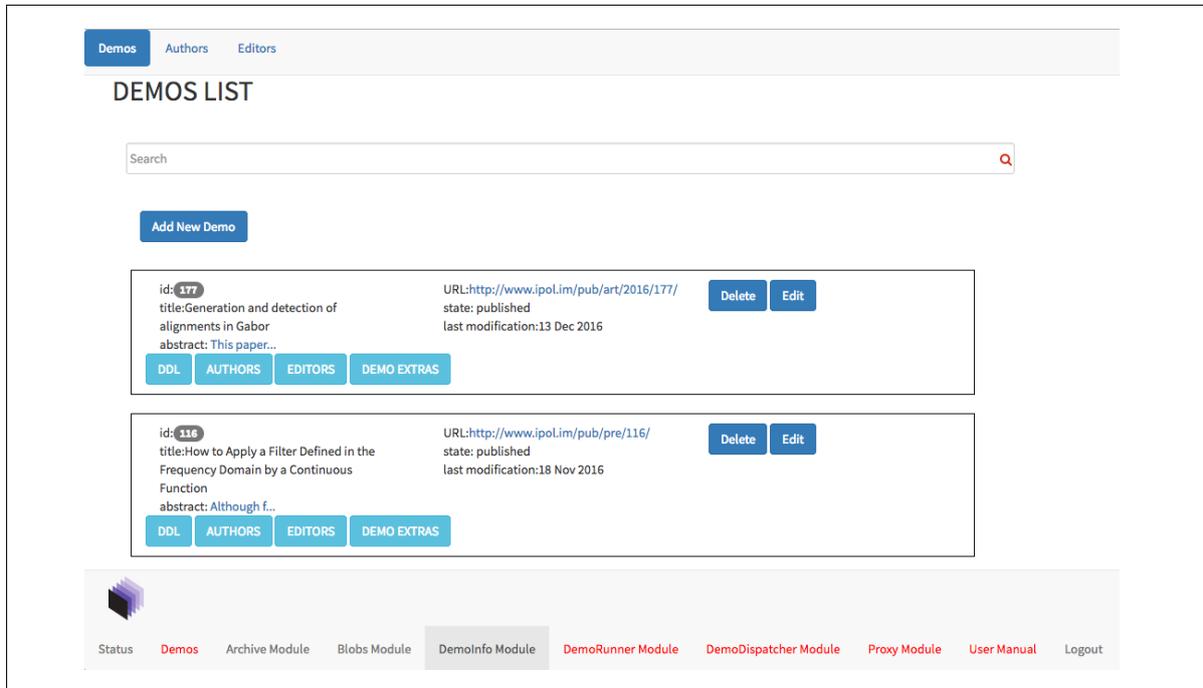


Figure 6.5: List of demos in DemoInfo Module option of the Control Panel.

## 6.7 Automatic demo generation

In the previous version of the IPOL demo system the demo editors had to write Python code to create a new demo. Specifically, to override some methods in a base demo class in order to configure its input parameters, to call the program implementing the algorithm, and also to design Mako HTML templates for the results page.

This approach does not really solve anything, since it simply moves the inability to generate demos from a simple description from the system to the demo editors. Since the Python code is written by the demo editors, it is prone to bugs which are specific to each demo. Moreover, fixing a bug in a demo does not prevent the others the have similar problems, with different implementations.

In fact, it is evident that this is a bad design, since to completely define a demo all that is needed is: (1) The title of the demo; (2) the URL where the system should download the source code of the demo; (3) the compilation instructions; (4) a description of the input parameters; (5) a description of what needs to be shown as results.

This information is not tied to any particular language or visualization technique, but it can be a simple abstract textual description. In IPOL we called this abstract textual description the Demo Description Lines (DDL). The IPOL editors only need to write such a description and the system takes care of making available the demo according to it. This not only avoids any coding problems (since there is nothing to code, but writing the short DDL), but also allows IPOL to have non-expert demo editors, and makes it possible to edit and publish demos quickly.

As an example, the following DDL listing is from a published IPOL demo:

```

1  { "archive": {
2    "files": {
3      "derivative1.png": " Derivative 1",
4      "sinc3.png": "Sinc 3"
5    },
6    "params": [
7      "a1", "a2", "orientation", "sigma"
8    ]
9  },
10 "build": {
11   "build1": {
12     "url": "http://www.ipol.im/pub/art/2016/116/filtering_1.00.zip",
13     "construct": "cmake filtering_1.00 && make -C filtering_1.00",
14     "move": "main_comparison"
15   }
16 },
17 "general": {
18   "demo_title": "How to Apply a Filter Defined in the Frequency Domain by a Continuous Function",
19   "xlink_article": "http://www.ipol.im/pub/art/2016/116/"
20 },
21 "inputs": [
22   {
23     "description": "input",
24     "dtype": "3x8i", "ext": ".png",
25     "max_pixels": "700*700", "max_weight": "10*1024*1024",
26     "type": "image"
27   }
28 ],
29 "params": [
30   {
31     "id": "a1",
32     "label": "x-component shifting",
33     "type": "range",
34     "values": {
35       "default": 0.25, "max": 1, "min": 0, "step": 0.05
36     }
37   },
38   (...)
39 ],
40 "results": [
41   {
42     "contents": {
43       "Derivative 1": "derivative1.png",
44       "Sinc 3": "sinc3.png"
45     },
46     "label": "Filtered and difference images",
47     "type": "gallery"
48   },
49 ],
50 "run": "main_comparison input_0.png -a $a1 -b $a2 -V -g $sigma -Q $orientation -e png"
51 }

```

## 6.8 Conclusions

The first version of the IPOL demo system has been working since the first article was published in 2010, with a total of 1434 citations and h- and i10-indexes of 20 and 36 respectively; its demo system is receiving about 250 unique visitors per day. While it is clear that the system is functional, some problems were detected: the system was difficult to debug to track down malfunctions, it suffered from tightly coupled interfaces, it was complicated to distribute the computational load among different machines, and the editors needed to write Python code to create and edit demos. These problems compromised the durability of the system at the same time they started to create a bottleneck that prevented to create and edit demos quickly.

The new system moved to a distributed architecture of microservices which solved many of these problems. It introduced however the typical problems of moving the complexity from the monolithic code to the boundaries of the microservices, but in general the balance has been quite positive. The system is made now of simple parts and the development team has gained flexibility due to the isolation of the microservices. Also, the editors are able now to quickly create and edit demos thanks to the abstract syntax of the DDL.

The next challenges for the very short term are to integrate new data types such as video, audio, and 3D, and the development team is quite optimistic about that, since the system is able to manage generic types (even if we refer to *images*, for the system they are simply *blobs*) and it comes down to a visualization problem in the website interface.

In conclusion, we managed to fix many of the problems found in the previous system by redesign and refactoring and now the system is ready to be expanded again, with a solid architecture and codebase.

## Acknowledgments

This work was partly funded by the European Research Council (advanced grant Twelve Labours), the Office of Naval Research (ONR grand N00014-14-1-0023), and the ANR-DGA project ANR-12-ASTR-0035.



# Chapter 7

## The future. Extending IPOL to machine-learning applications

The natural evolution of the IPOL demo system consists on being extended to support machine learning applications. However, this requires a full re-design of the architecture of the system. It comprises challenges such as the heterogeneity of the data, the lack of a common standard specifications, and introducing new steps. This new scenario is discussed in this chapter and a new architecture is proposed. This is a work in progress and part of my research plan for the next years. Parts of this chapter come from the article *Extending IPOL to New Data Types and Machine-Learning Applications* [80].

### Abstract

Image Processing On Line (IPOL) is a journal focused on mathematical descriptions of image processing (IP) / computer vision (CV) algorithms. Since the first article was published in 2010, it has started to become clear that the IP/CV discipline is mainly multi-disciplinary. For example, nowadays images are de-noised using convolutional neural networks (CNN), and fields such as neurophysiology need of the rudiments and techniques of IP/CV, general signal processing and artificial intelligence. IPOL wants to extend the capabilities of its demo system to cope with these needs. Specifically, in this article we review the state of the current demo system and its limitations. It enunciates a detailed project on how to build a more adapted system, and its minimal requirements: new data types, problematic heterogeneous data, the pre-processing and standardization, the possibility to chain different algorithms in a complex chain, and how to compare them.

### 7.1 Plan to extend IPOL to machine-learning applications

IPOL has been evolving since it was funded in 2010, from a very simple demo system which required the editors to write actual Python code and design HTML templates, to a system of a distributed system started in 2015 with load balancing along several machines and automatic demo generation from an abstract description syntax.

The next step is first to move to new data types (audio, video, 3D) and finally to allow different applications, including Machine Learning. Adapting the system to a wide-purpose platform will require some changes in the architecture of the system (new modules) and better data organization (Archive, mainly) but still the current modular architecture of the system will be able to support it. In any case, moving to new application fields and at the same time keeping a generalist platform will require strong engineering efforts and a stable team with permanent positions. This is not the case at this moment, and we need to work on this point to be able to extend IPOL with advanced features reliably.

The following sections present and discuss the steps needed to build the next evolution of the IPOL system.

### 7.1.1 Extended architecture: support for applications

There is a fundamental difference between the current IPOL demo system and the complete platform offering Software as a Service (SaaS) applications and learning capabilities: their execution time. The complete platform differs from a demo system in the fact that it allows applications whose lifetime<sup>1</sup> is largely over the short period of execution of a demo. In the case of a demo the input data is loaded, the algorithm executed, the results shown, and the demo has totally finished. In the case of algorithms which can learn from the archive, they need to be running indefinitely. Once they are run, they can receive events indicating the there is new data in the archive, or to explore it regularly.

We shall call these new processes with a large lifetime *Applications*. The system will have two different kinds of processes:

- Demos
- Applications

Nevertheless, the architecture of the new system needs not be redesigned in order to have these new functionalities, but new modules need to be added and they will share information within the system.

The first module which needs to be added is the equivalent to Core which delegates the demo operations along different specialized modules, but for the Applications. We will call this new module *AppCore* and rename the existing one to *DemoCore*.

The DemoCore will delegate some specialized functions in already existing modules, for example in Conversion or Blobs since they still be needed in the context of the applications. But new modules will be exclusive to AppCore, such as:

- *Databases*: storage and management of training and testing datasets
- *UserAccess*: lists of users and authorization management

The same way DemoCore offers an API of microservices that the web application uses to render demos in the client's browser, AppCore will offer microservices to be used by external applications. We can think for example of a MATLAB application or a Jupyter notebook which uses the AppCore's services to log in and then visualize classification results obtained using the corresponding learning databases. Fig. 7.1 shows the architecture of the complete platform.

Note that the platform is not tied to a particular visualization tool (say, a website) or to any specific framework (MATLAB, R, or the Android system, for example) but instead it offers generic web services that can be used with them. This way the platform remains generic but at the same time it can be customized with the frameworks preferred by its users.

### 7.1.2 Extension to new data types

The current demo system already allows to work with arbitrary data types. By design, we chose to refer to the data generically as *blobs*, instead of being attached to a particular data type such as images and others. The system manages the data in terms of blobs, and only a particular module ("Conversion") (see Fig. 6.2) needs to know about the actual details of the type to make conversions. This means that when adding a new data type only Conversion needs to be modified whereas the rest of the system modules remain the same.

The execution flow given an input blob is the following:

1. The blob arrives to Core module. The origin can be either one of the input blobs offered by the demo or a new blob uploaded by the user.

---

<sup>1</sup>Here we refer to the "wall clock time" of the process, not its CPU usage time. The application is expected to have a large lifetime, however mostly idle.

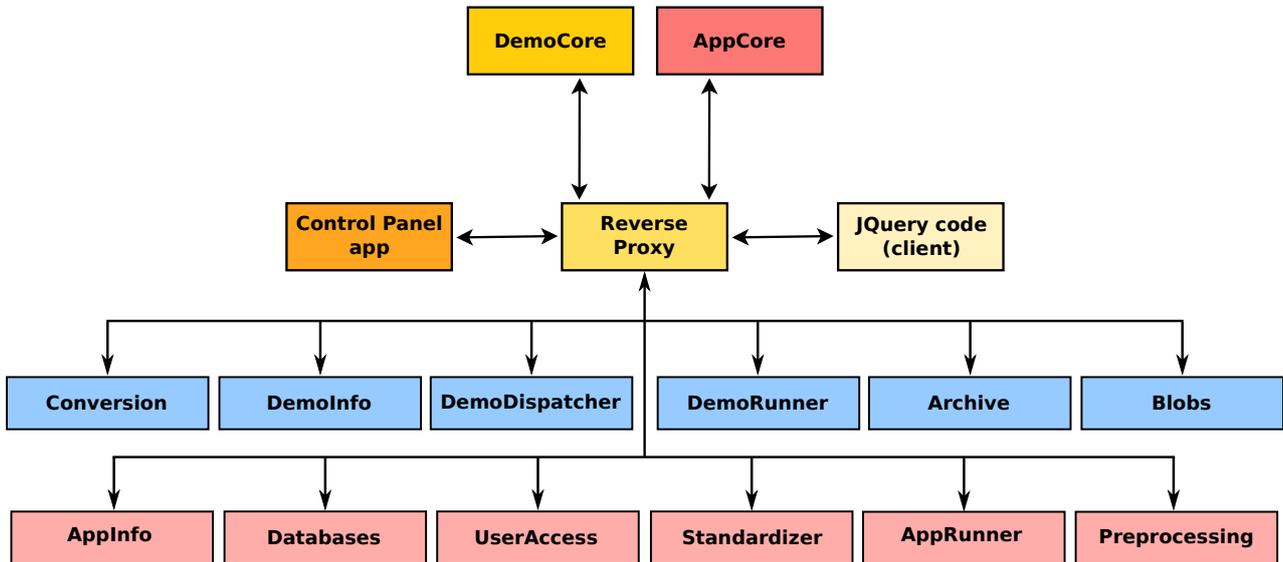


Figure 7.1: The complete modular architecture of the future IPOL platform.

2. Core examines the DDL to check if the demo authorizes data conversion and also it that conversion is actually needed. For example: reduce an image to the maximum allowed size and convert from JPEG to PNG.
3. In case the modification is authorized and needed, Core delegates the conversion of the blob to Conversion.
4. The Conversion module converts the blob to meet the requirements written at the DDL. Only in the context of Conversion the actual data type is taken into account. Outside it is only a generic *blob*.
5. Core goes on with the execution of the algorithm. With the original blob or with the converted.

The kind of conversions needed depend on the data type. For images, audio, video, and 3D they are basically two: resampling (or decimation) and format conversion. Other data types might need different and specific transformations.

While the conversion of the data itself is supported by the current system, depending on the application it might not be enough. Indeed, two more steps are needed:

- Data standardization
- Pre-processing

Given that the aim of the platform is to support multiple applications and data types, the input data is likely to be in different file formats and in different ranges. We can think for example in the data of two different accelerometer sensors: one of them could give values of the three spatial axes in  $m/s$  units, while the other might give only two of the three axes in  $cm/s$ . Before attempting the process the data, the system needs to put the input data in a common format.

Even if the data has been standardized, it might require pre-processing. For example, an algorithm might not work with missing (“NA” - Not Available) values, while for others this could be acceptable. The system might decide to pre-process the data in order to remove NA samples by interpolation, or simply to remove them before passing the data to the algorithm for execution.

The solution is to add two new modules to the system:

- *Standardizer*: to structure the data in a format understood by the system and the algorithms;

- *Preprocessing*: to perform pre-processing steps before execution according to the needs of the algorithm.

Core would systematically pass the data to these two modules to check if these steps are needed and to actually perform them in that case.

About the web interface, it needs to be expanded to support the visualization of the new data types. At the moment of writing this text, the system has full support for video processing and visualization and the interactive controls (masks for inpainting demos, points for Poisson editing, segments for lens correction, etc.) are expected to be finished in a few weeks. Fig. 7.2 shows the IPOL demo of the published article *Ball Pivoting Algorithm* for which the system shows the result with a 3D mesh renderer.

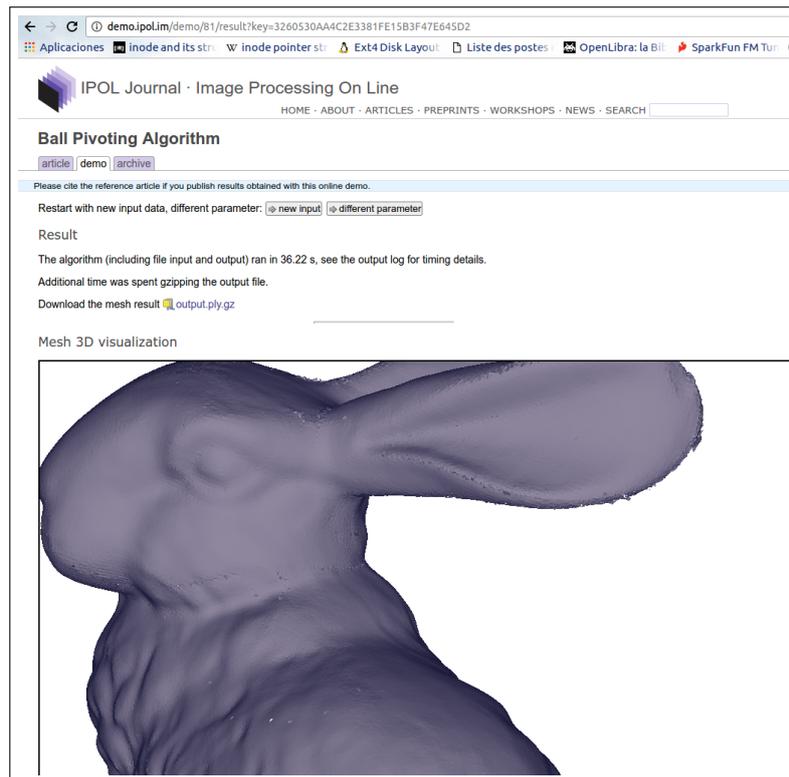


Figure 7.2: IPOL demo of the published article *Ball Pivoting Algorithm* showing the results with a 3D mesh renderer.

### 7.1.3 Comparison of algorithms

A desired feature in the future platform is the comparison of algorithms. At this moment every demo is an isolated unit which takes the input data, applies an algorithm with the configured parameters, and finally shows the results. However, being able to measure the performance of the algorithms is of great interest it this would allow to score them and to know which is the best parameter choice according to the algorithm and the input data.

In order to compare the algorithms a metric of quality must be available. This metric can depend solely in the results (for example, exploiting structural similarity in the case of images [46]) or a metric with respect to a given ground-truth.

The inputs and outputs of the algorithm might vary largely and therefore we need to define common data formats in the platform. Clearly, to compare two algorithms their outputs need to be encoded with compatible formats to allow automatic comparisons.

The role of Preprocessing Standardizer presented in Sec. 7.1.2 becomes clear here. Indeed, the comparison of algorithms feature relies on this two modules since the platform needs to:

- Define formats to homogenize the outputs of diverse algorithms. The Standardizer module will be the responsible for this task, delegated from the DemoCore or AppDemo controller modules.
- In the case of missing data or if pre-processing is needed, then Preprocessing will be invoked too.

Sometimes it will be needed to express the same data in different forms. For example, some metrics would need a temporal series while others could work only with their spectra. The role of Standardizer is to put the data in a flexible format when it is entered into the platform for the first time, not to perform data format conversions during execution (which is a different task and performed by Conversion, see Fig. 7.1).

The platform needs to provide large datasets to test the algorithms with and also several metrics to perform the comparisons. In the most favorable scenario, these datasets are previously annotated by experts and serve as a ground-truth to evaluate the algorithms. The module responsible for storing the testing and learning databases is the Databases module (Sec. 7.1.5).

#### 7.1.4 Chaining of algorithms

The *chaining* of algorithms consists in using the output of one algorithm as the input for another. For example, a de-noising algorithm could be applied to an input image before attempting object recognition. Another example may be an algorithm to interpolate missing samples in the data acquired from an accelerometer could be run in a first step before trying to determine the trajectory followed by the sensor. In the case of satellite images, we could think of a chain which first de-noises pairs of images with one algorithm, then rectifies them with a second algorithm, and finally performs a 3D reconstruction with yet another different algorithm.

This problem is utterly related to the comparison of algorithms sketched in Sec. 7.1.3, since the key to be able to connect one algorithm with a different one is that both have a common format for their input and outputs. The platform needs to define general formats for each possible input and output for the algorithm.

Some general data types that should be defined within the platform according to the current editorial needs:

- Temporal series of  $k$  dimensions, allowing missing values.
- Sets of points in a 3D point cloud. In general, sets of hyperpoints with an arbitrary number of points.
- Sets of edges and vertices to define 3D meshes. In general, edges and vertices defined in a hyperspace.
- Images and Video. In general,  $k$ -dimensional hypercubes of data.

Other generic formats might be defined according to the evolution of the platform and the needs of the users.

We propose to create input/output wrappers in the algorithms to adapt the data to the formats of the platform, to avoid forcing the implemented algorithm to follow specific formats. This way the platform would allow most of the algorithms available without the need of adapting them.

In the future system a final result might not come only from the execution of a single algorithm, but built from the concatenation of many.

### 7.1.5 Machine learning applications

Machine Learning algorithms have an extra step which makes them different to most of the Signal Processing algorithms: a first training step. Indeed, they normally proceed in three different phases:

1. A training (or *learning*) step with some examples and one or more metrics to evaluate the performance of the algorithm. For example, a neural network classifier intended for face recognition needs to be trained with many samples previously labeled as faces and non-faces. Typically the learning step is slow, requires many samples, and can be improved by adding more samples to the learning database.
2. A validation step, to ensure that the training is not biased.
3. Prediction. Once the algorithm has already learned according to the training samples (say, a neural network with well-adjusted weights or a SVM<sup>2</sup> classifier with the right hyperplane) it can proceed to predict the result given a sample that the algorithm has not seen before.

On the other hand, the assessment phase needs:

1. A metric (i.e. a formula), which typically is different from the metric used during training.
2. A test set, different from the training set.

The two-step process described above might not be directly applicable when the training of the neural network needs a complex customized training and therefore can not be reduced to uploading learning data and applying standard stochastic optimization. Also, depending on the application the learning step could consist on several phases.

For those complex cases we expect the learning to be performed offline and to work with two types of data:

- The learning set that must be kept for records and
- the generated neural network.

The researcher should upload a script (as *demoExtras*, see Sec. 7.1.6) governing the learning, in addition to data and to the back-propagation codes. In order to make it reproducible it would be needed that all the details on the training process are documented, including any random seeds, the hyperparameters, the initialization of the weights when learning (for example, Xavier's initialization), number of mini-batches, and in general any other learning parameters which are relevant to reproduce the same results.

The typical demos in the current IPOL system are not well adapted to this new schema since their cycle is simple to read the input data, apply some algorithm with the given parameters, and finally show the results. It is possible however to give some pre-learned databases, but the new platform will be able to update its own training datasets according to the input data from authorized users.

Let us present a typical case of use of the platform. In order to make it more understandable, we will explain it in a concrete application such as healthcare.

The platform will be used by persons with different roles, such as clinicians, technicians, and researchers from different fields (for example, physiologists, signal processing researchers, statisticians, expert systems researchers, etc). Their diverse interests converge at the platform. Doctors are mostly interested in obtaining relevant information after they have introduced data of a particular patient (for example, the possibility of discovering a neurological problem from the analysis of saccadic signatures in ocular movements [81]), as well as tracking the evolution of a patient over time.

---

<sup>2</sup>Support Vector Machine.

One of the objectives of the platform is to obtain substantial results by applying Machine Learning algorithms. In this example our aim would be helping clinicians extract relevant medical information from large and multidimensional datasets.

Doctors will decide if the data of their patients (after mandatory anonymization [82]) can be incorporated to the databases of the platform, or not. For example, if the clinician has labeled data it can be used as a ground-truth for many algorithms of the platform and thus over colleagues benefit from it. Or they can decide not to share but still use it as a tool with their own data.

Not all users will be authorized to upload training data (or at least, it needs to be peer-reviewed first), since the performance of the platform relies on the quality of its learning base and thus it needs to be carefully controlled. Thus, depending on the role of the users there will be several combinations of rights and authorization levels: possibility of uploading training data, possibility of executing the algorithms on uploaded test data, possibility to label an already existent dataset, possibility of creating groups of users with common right for certain datasets, etc. Instead of fixing the set of possibilities it is better to leave them open as configurable options and to set up rights for users or groups of users depending on their roles. As advanced in Sec. 7.1.1, the platform will use a new *Databases* module to store and manage the permissions for the training and test datasets.

In order to take advantage of both the practitioners' extended knowledge and the presence of numerous healthcare datasets, the platform could also include a crowdsourcing aspect (see e.g. [83]). Using trained classification or scoring algorithms, unlabelled data with the largest amount of uncertainty could be presented to the platform medical user for additional input – thus both improving the quality of the dataset, the performance of the algorithm, and presenting practitioner with interesting use cases.

In the case of executing an algorithm with a test dataset (say, to perform prediction), the execution will be similar to a typical IPOL demo. It can be sketched as follows:

1. The users upload or choose one of their datasets;
2. they chain several algorithms and configure their parameters;
3. the algorithm is run. The user can wait until it finishes, or it can be stored and recovered later;
4. the results are shown;
5. the results are archived for further analysis.

This is a complex interaction, perhaps for the most advanced users. However, most users (say, medical doctors) would just upload their data and wait for their automatic results. However, the system needs of course to be designed such a way that it copes with both the complex and common cases.

Technically, the request arrives to Core, which delegates the selection of the most available DemoRunner for the execution, DemoRunner will perform the execution, and then Core will retrieve the results. Finally, Core will send a command to Archive to store all relevant information on the experiment. See Fig. 6.4.

While the current system could support easily Machine Learning learning algorithm to perform *testing* (say, classification or prediction), it needs to be extended to reach the new architecture depicted in Fig. 7.1). Indeed, as explained in Sec. 7.1.1, the execution cycle of a simple demo does not allow to have a long learning phase.

This operative does not adapt well to the needed learning phase and thus a new entity (the *applications*). The applications are processes which are started by AppCore and instead of finishing after a fixed period has elapsed, they simply enter in a *sleep* mode. They can be woken up by AppCore for several reasons, as for example when new data arrives to Archive, or they can wake up by themselves at regular intervals to perform learning on the data available in the platform.

To better understand the concept of the applications, let us think of an algorithm to detect faces in pictures using an SVM. The users will upload their own pictures to the demo to check if the system

is able to detect or not the faces that might be present in the image. The users will provide feedback about if the system managed to detect a face in the right position (positive), or if it detected a face in a place where it was not any (false positive), or if it missed an actual face (false negative). To do it, the demo could use an SVM classifier whose hyperplane is shaped according to a learning step. Thus, the published algorithm will implement two tasks: testing (used by the demo), and training (performed by the application). We can imagine the algorithm's package as a ZIP file with two folders with independent codes, one for testing and the other for training.

Once the algorithm has been installed in the system, AppCore will start the corresponding application which uses the training code. It will wake up regularly the application with an event indicating that it has new feedback, and then the application will run the training code to update the SVM hyperplane. This will improve the performance of the demo. Once the training with the new data has finished the application will enter in a sleep mode until it is woken up by AppCore again.

Note that when we write *the AppRunner* or *the DemoRunner* modules, actually we refer to an instance of many of them. Indeed, while the other modules are unique in the system, the runners are many distributed processes. Each time Core needs to execute a process (say, a demo or a learning phase), it uses Dispatcher module to pick a particular runner and send the execution. For example, in the current system we have three distributed demoRunners available. One is physically installed at CMLA laboratory, and the other two are in the installations of an external provider. This is easily configured with simple configuration files.

In the case of the applications they will be also distributed along several servers. The AppCore will ask Dispatcher for an AppRunner, then AppCore will wake the runner up and ask to update its learning database when needed. Note that AppCore has the possibility of running several instances of the same learning process along different runners if necessary. For example, it can split the input data into several batches and make different runners process each of the parts at the same time.

Apart of adding these new AppCore and AppRunner modules to the architecture of the system, the existing Archive module will be improved to allow queries and receive structured information quickly. For example, we can think in a demo that runs a classification algorithm in images (let's say, it tells which is the animal in the image). If the demo configures Archive to store extra information about the inputs (say, level of JPEG compression, the mean intensity of the image, ...), it would be possible to extract meaningful information afterwards. For example, it how relevant is the compression level of the image when performing classification, or how images with very low contrast or saturated impact the performance. This could be done with a query language very similar to SQL.

Improving Archive will allow not only Machine Learning applications, but also for more complex Signal Processing algorithms in the demo system.

Finally, already existing Control Panel (Sec. 7.1.6) will be extended to allow the system operators to manage<sup>3</sup> the installed applications.

### 7.1.6 System management with a unified tool

A tool to allow non-technical administrators to manage the platform is needed. Such a tool is known as the *Control Panel* in the current system (see Sec. 6.6).

The Control Panel allows the IPOL editors to create new demos and modify any existing demos. This tool is a web application where the editors have access after typing their username and password and one of the design requirements is that it should be used by editors without any technical background. It allows however to add extra code to the demos (known as *demoExtras*) in order that advanced editors can customize them but nevertheless most of the demos are defined from a simple textual description (see Sec. 6.7).

In order to allow the complete system with the extensions presented, the Control Panel also needs to be completed. Specifically, it needs new sections to:

---

<sup>3</sup>Add, remove, start, stop, and configure user database rights.

- Install, configure, and remove Applications, in a similar way to what does already for the demos.
- Create, edit, and remove users.
- Add, remove, and configure the access right of the learning and testing databases for each user or group of users according to their roles in the platform.

The existence of a unified tool such as the Control Panel<sup>4</sup> and the use of an API (see Sec. 6.4) to interact with the system hides not only the implementation details but also the fact that the system is distributed along different machines. Indeed, the editors manage the system with a simple tool which hides all its complexity and lets them configure it quickly and easily.

## 7.2 Conclusions

The first version of the IPOL demo system has been working since the first article was published in 2010, with a total of 158 articles, 18 preprints, 3811 citations and h- and i10-indexes of 29 and 65 respectively. While it is clear that the system is functional, some problems were detected: the system was difficult to debug to track down malfunctions, it suffered from tightly coupled interfaces, it was complicated to distribute the computational load among different machines, and the editors needed to write Python code to create and edit demos. These problems compromised the durability of the system at the same time they started to create a bottleneck that prevented to create and edit demos quickly. These problems are already solved with the new version of the demo system, a distributed architecture of microservices.

Of course, even if the system is already functional it still needs strong engineering efforts and maintenance. With more than one hundred published demos, more than eight years of activity, about 250 unique visitors/day, 400 executions/day, more than 250,000 experiments performed by the users with their original data, IPOL has been proved to be a useful tool not only for research but also as a value-added tool for the industry.

The existence of similar initiatives such as Jupyter, Code Ocean, RunMyCode, and openCV indicate a great interest of the scientific community and the industry for this kind of platforms, for which IPOL is a clear precursor. These other platforms do not need to be understood as a threat for IPOL, but instead as an indication of the market trend. We refer the reader to [84] for a comparison and discussion on platforms that might be used to implement reproducible research. Nevertheless, IPOL needs to go a step beyond and move from a demo system to a wide-scope platform to build complete applications over it, following the Software as a Service model. Machine learning algorithms, the use of learning databases, and the ability of the platform to improve these databases is a unique business that is worth exploiting.

Without any doubt, this is an ambitious project but nevertheless realistic assuming that a stable engineering team is available. The fact that IPOL has been running for almost ten years already is the best proof of feasibility.

## Acknowledgments

I would like to thank Jean-Michel Morel, Nicolas Vayatis, Bertrand Kerautret, and Daniel Lopresti for their valuable comments that helped improve this work.

---

<sup>4</sup>Nowadays, even configuration tools like this are regarded as part of an experimental system, and techniques like A/B testing are used to improve the performance (usability) of the tool.



## Part III

# Automatic detection of image forgery



# Chapter 8

## Introduction

The term *fake news* is a recent neologism that designates the act of deliberately spreading misleading information via the traditional media, or more specifically on the Internet (social media mainly). Although the term is new, this phenomenon is not and indeed one can easily find many examples over time. For instance, in 1835 the New York's newspaper *The Sun* publishes a series of six articles where they claimed that a whole civilization was observed on the surface of the Moon (!) using a huge telescope that implemented a brand new observation technique (certainly, not the interferometric techniques of chapter 4).



A lithograph depicting the *ruby amphitheater* described in the series of articles as published in *The Sun*, with “its tall figures, leaping cascades, and rugged caverns”. Source: public domain.

This extraordinary information boosted the sales of the newspaper and the forged information spread rapidly. Edgar Allan Poe even accused the newspaper to have plagiarized his earlier work *The Unparalleled Adventure of One Hans Pfaall*. Moreover, these fake news and the short story of Allan Poe are mentioned in Jules Verne's *From the Earth to the Moon*. The fact that those amazing stories caught the interest of many people and relevant authors uncovers some of the principles of fake news: the large-scale diffusion of information, the aim to manipulate, and the inability of the readers to confront and to fact-check the information they consume.

The aim of fake news is wide, including selling more newspapers or attempting political manipulation. For example, in 2016 an Internet site published that Pope Francis officially endorsed Donald Trump for president. Quickly these news attracted a large flow of visits to the site (thus, increasing the earnings from the advertisements shown on the website), as well as increasing also their number of followers in the social media. This is a case with both economical and political motivations.

Nowadays there are millions of smartphones permanently connected to the Internet, several social networks with thousands of millions of users (Facebook, Twitter, Instagram, and others) where con-

tent can be posted instantly, as well as very powerful image and video editing tools available. It is a great broth to grow fake news!

The amazing narrative of The Sun describing the creatures that were observed on the Moon (including unicorns and goats) along with suggestive drawings was enough to stimulate the imagination of their readers up to the point of making them believe... or want to believe. But certainly nowadays a simple drawing is not enough and the public demands somehow more *tangible proofs*. And today in the connected and instantaneous communication era, a picture supposedly taken by a smartphone may be an evidence enough. After the dreadful shooting at Marjory Stoneman Douglas High School in Parkland (Florida, USA) in 2018, where seventeen persons were assassinated, a debate on gun control measures started. Many citizens protested against the current policy of the American administration, demanding more control on gun violence. One survivor of the attack, student Emma González, asked politicians to take action to prevent more deaths in the future, instead of simply sending their condolences and prayers. She also led other protests about school safety and against gun violence.

In this debate there are strongly confronted points of views, from citizens that plead for gun control to associations that find that a good solution is allow citizens to carry even more guns so they can defend themselves<sup>1</sup>. Emma González posted on the Internet a picture of herself tearing a poster of a gun-range target. It did not take long until that picture was vilely and carefully manipulated in order to forge a new fake image on which the original poster was replaced by the US constitution. The false image along with the false information turned into fake news and spread out of control all over many social media. Now the information consumers did not need to exercise their imagination as in 1835, but instead they had a picture that clearly was proving the "facts". Hopefully, there are methods that can help detect those manipulations and forgeries in digital images, as shown in the chapters of this part. In particular, the forgery of the Emma González original picture can be detected with the methods that we have developed<sup>2</sup> (chapters 9 and 10).

The CMLA lab was selected for participating in the ongoing *DEFALS*<sup>3</sup> challenge proposed by ARN (the French national research agency) and DGA (the French Directorate General of Armaments). The challenge is a competition where four groups try to obtain the best score detecting forged images. A database of images without any ground truth is provided to each of the teams, and for each image a decision is made (a binary decision telling if the image is forged or not, or a mask indicating the location of the forgery). The algorithms are *blind* in the sense that the only information available is the image file itself. For this challenge we are applying the techniques described in detail in chapters 9 and 10, as well as the noise level analysis of the images, chromatic aberration analysis, demosaicking traces analysis, copy-move attack detection, and others. All the methods that we develop will be published in peer-reviewed journals (including IPOL) so they can be available to the scientific community. Indeed, we think that reproducibility is paramount for the credibility of any technique which performs automatic detections. Part II of this HDR report fully discusses on reproducible research.

The fight against fake news has several fronts, from new laws to new technical means. Governments have enacted different laws to fight against the propagation of false information. In France we can cite one of the first attempts, with the *Loi du 29 juillet 1881 sur la liberté de la presse* (Law of July 29 1881 about press freedom), where the publication of false information in a journal was considered as a *press crime*. Even then, there was a debate about the press freedom and how to fight against fake news. In 1851 the editors of the Journal de St.-Quentin showed their worries<sup>4</sup>:

<sup>1</sup><https://www.washingtonpost.com/news/post-nation/wp/2018/03/24/nra-host-taunts-parkland-teens-no-one-would-know-your-names-if-classmates-were-still-alive/>

<sup>2</sup><http://cmla.ens-paris-saclay.fr/un-algorithme-qui-traque-automatiquement-les-images-falsifiees-20180324>

<sup>3</sup><http://defals.fr>

<sup>4</sup><https://www.retronews.fr/histoire-de-la-presse/echo-de-presse/2018/06/25/debut-xixe-premiere-agitation-autour-des-fausses>

*En quoi consiste une nouvelle fausse ? Une simple erreur de fait, comme il en échappe nécessairement à tous les journaux, constitue-t-elle un délit ? [...] La reproduction du bruit le plus insignifiant, le plus étranger à tout intérêt politique, le plus innocent aux yeux du public sera-t-elle coupable aux yeux de l'autorité ?*

What is false news? A simple error of fact, as it happens necessarily in all newspapers, does it constitute a crime? [...] The reproduction of the most insignificant noise, the most foreign to any political interest, the most innocent in the eyes of the public will it be guilty in the eyes of the authority?

The manipulation needs not actually consist in manipulating the images or videos or writing false information. It can be as subtle as simply taking a picture of a scene in such a way that important information is missing in the frame (thus, misleading the reader to a false interpretation), omitting relevant information.

More recently in France was validated by the *Conseil constitutionnel* in December 2018 the law against the manipulation of information, which has a special focus on the web and social media. The law gives some specific rules to decide when the diffusion of a piece of news should be stopped : (i) the information should be clearly false, (ii) it is massively and artificially spread, and (iii) it compromises the public peace or it attempts to manipulate a poll.

Press associations have also taken action against the spread of fake news. As a specific action, we can cite Fact Check<sup>5</sup> from AFP (the French press association), where fake news are debunked and their context explained to the readers. The tools that we have development have already been useful to expose fake news, as it is shown in the Fact Check website of the AFP<sup>6</sup>. Indeed, we plan to incorporate several of our algorithms into the AFP tools.

The fight against fake news is a competition (the same way DEFALS is a challenge). In this game we can find the economical and political manipulation interests of the part which produces fake news, the development of multimedia (images, video) forgery detection techniques by researchers, the counter-forensics measures which try to hide the traces of manipulation from the detection algorithms, the press associations that implement measures to fight against the spread of false information, among others.

New forgery techniques appear regularly, some of them as technically advanced as the recent *deep-Fake* based of deep learning, that allows the forger to exchange the faces of people appearing in videos. And recently, even to fake the speech<sup>7</sup>. Luckily, even these advanced techniques leave some traces that can be used as a signature of the manipulation, that can be detected by specialized algorithms<sup>8</sup>. Also, new image and video formats appear in the mark (HEIC, for instance), but again hopefully any manipulations will leave detectable traces.

In the following chapters we present our modest contribution to the detection of forgeries in JPEG-compressed images: two different published algorithms that allow to detect that a zone of the image has been copied from the same of a different image. Also, they can serve as the first step for other detection algorithms given that they locate the JPEG blocking grid. The main contribution of these methods is twofold:

- They provide a detection decision using the NFA (number of false alarms) technique. Many other methods simply give a visual result in the form of heatmaps that need to be interpreted by the experts. The methods presented here deliver a Boolean decision that indicates if there was a detection or not and, if so, where it is.
- The NFA provides a theoretical basis that ensures that the probability of a detection being a false positive is very low. Moreover, this probability can be adjusted depending on the needs of the applications.

<sup>5</sup><https://factcheck.afp.com/>

<sup>6</sup><https://factcheck.afp.com/no-not-photo-dangerous-building-under-construction-nairobi>

<sup>7</sup><https://www.technologyreview.com/f/613645/ai-fake-news-deepfakes-misinformation-united-nations/>

<sup>8</sup><https://www.helpnetsecurity.com/2019/08/09/zerofox-deepfake-detection-toolkit/>

The first author of this work is Tina Nikoukhah, as part of her PhD project.

In the research project sketched in chapter 11 I give some hints about our plan to detect forgery using the noise model, which will be one of the research topics of Marina Gardella (future PhD candidate).

# Chapter 9

## JPEG grid detection with controlled false alarms

In this work, published in [85] as the first-author work of Tina Nikoukhah, we detected forgeries based on the blocking traces after JPEG compression. Instead of providing a subjective *heatmap* that would require of the interpretation of experts, we developed an algorithm that outputs a localized detection decision based on the NFA. Later we found that the number of DCT zeros in the quantized JPEG matrix is a useful complement to this technique for the cases when this method is not able to decide (see chapter 10).

### Abstract

With the progress of image manipulation tools and the proliferation of fake news and images posted online on social networks, automatic identification of fake content is becoming indispensable. Lossy image compression leaves traces which can be used to recover the history of an image and to help decide about its authenticity. We propose a new JPEG grid detection algorithm. This operation is the first step of many forensic, anti-forensic, and deblocking algorithms. Our analysis is based on the detection of the blocking artifacts and is global and local at the same time. It retrieves the origin of the JPEG grid in all image regions and detects suspicious discrepancies. Our work is based on the *a-contrario* framework which reins in the over-detections caused by multiple testing. It also yields a Number of False Alarms (NFA) which gives extremely secure guarantees for tampering detection. We demonstrate the performance of the proposed method with both quantitative and visual results from well-known image databases.

### 9.1 Introduction

With the growth of social networks, the need for views and ratings has been increasing in the past decade. The evolution of technology has made it possible to publish false content in form of multimedia elements. More particularly, with image editing tools becoming more and more efficient and easy to use, forged images have become a great way to attract viewers.

Doctored photographs are very difficult to identify by visual examination [86]. More than 3.2 billion images are shared each day, which is 100 times more than the amount in 2011. Therefore, the credibility and trustworthiness of digital images has become an important matter, which has led experts to work on image forensic techniques. Indeed, finding digital fingerprints left by image processing and tampering can be used to determine whether an image has undergone modifications [87]. Many studies have been conducted to detect forgery in images [88, 89]. Most of them assume that almost no *a priori* information is available. Therefore, authentication of the image's history needs to be done from only the image itself and without the analysis of its metadata, headers, or file extension.

The digital life cycle of an image can be separated in three phases [90]: acquisition, coding and

editing. We will focus here on the coding phase. Indeed, the convenience of desiring smaller amounts of data to store and transmit leads to most digital cameras exporting in JPEG format [91], which is the most common format found online. JPEG images are involved in many forensics situations and the compression history is anyway interesting to recover.

The JPEG procedure starts by partitioning the image into  $8 \times 8$  non-overlapping blocks. The Discrete Cosine Transform (DCT) is then applied to each block. The DCT coefficients are quantized and finally losslessly encoded.

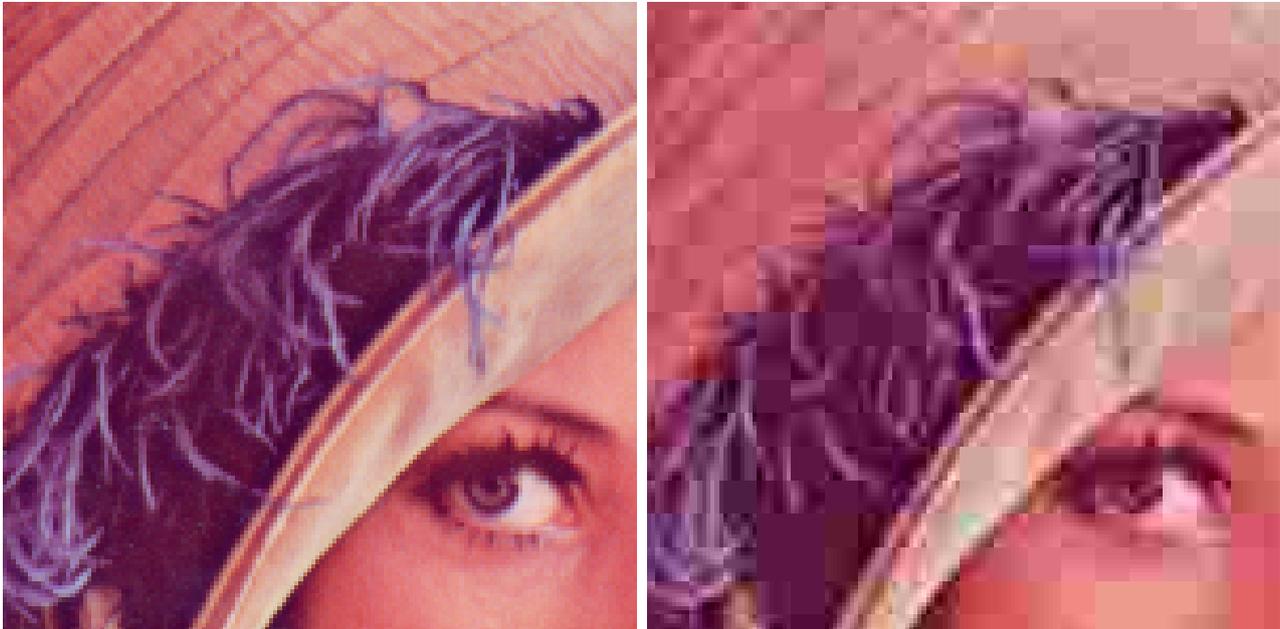


Figure 9.1: Close view of block artifacts for JPEG compressed image “Lena” with quality factor  $Q = 10$ .

Since the quantization process of JPEG compression is performed on each image block independently, blocking artifacts, as shown in figure 9.1, appear at block boundaries between adjacent blocks in the image. These characteristic compression traces can be analyzed both in the pixel domain and the transform domain. These artifacts, which degrade the quality of the image [92], yield very informative cues in forensic analysis.

The outline of the article follows. Section 9.2 reviews the state of the art on the techniques which are used to infer the JPEG compression history. Section 9.3 presents our method in the context of the *a-contrario* theory. This is our main theoretical contribution to the topic. Section 9.4 presents our results assuming different applications.

## 9.2 State of the art

A high number of forensic techniques have been designed to authenticate JPEG compression history. For the most part, these techniques have the same three step structure [93, 89]:

- grid detection, or block detection;
- quantization estimation;
- double JPEG compression detection.

Some techniques skip the first step because of a lack of a solid theoretic framework. Most methods are based on the analysis of DCT coefficients [94, 95] to estimate the original quantization table which is subsequently used locally to identify tampered areas. In the transform domain, block-based image coding schemes modify the histogram of transformed coefficients. In consequence several methods analyzing the shapes of these functions are proposed in the literature [96, 97].

The blocking artifacts have a regular pattern, since the quantization of the DCT coefficients is done separately on  $8 \times 8$  disjoint blocks. Image content and dynamic makes these discrepancies hard to detect in the Fourier domain [98]. This leads to work directly in the spatial domain, and in particular, on the luminance component  $I$ . Denoting by  $R$ ,  $G$  and  $B$  the color components of the pixel, the standard definition [99] of  $I$  is

$$I = 0.299 R + 0.587 G + 0.114 B.$$

The authors of [100] and [101] proposed algorithms which are based on the idea that if the image has been compressed, then the pixel differences across block boundaries are significantly different from those within blocks. The energy differences are compared to a threshold to deduce the presence of a prior compression. However, their method is a statistical estimate over the whole image and therefore gives a global result, not a local JPEG grid estimate.

The approach in [102] is capable of detecting and localizing tampered areas but is, nevertheless, sensitive to image content and suffers from high false detection rates. This is problematic for an automatic analysis.

Lin et al. [92] present a robust grid extraction method with an estimation based on a maximum likelihood method, introduced by [103]. Their forgery detection technique is based on two passes: one to estimate the main grid of the image and one to identify blocks which do not coincide with the global estimation. Their two-step technique is less sensitive to the image content and is capable of localizing tampered areas. Nevertheless their method depends on the tuning of several parameters (thresholds and attenuation values). This makes it difficult to render the detection method fully automatic.

A recent survey [89] states that “*all the approaches and methodologies [...] have the capacity to recognize fraud. In any case, a few algorithms are not viable regarding identifying actual forged regions. On the other hand some algorithms have a time complexity problem. So, there is a need to develop an effective (efficient) and accurate image forgery detection algorithm.*” A solid theoretical mathematical framework describing the statistical behavior of the quantities involved is also desired for each image forensic technique [104, 93]. Indeed, although some methods achieve excellent results in certain experimental settings, the absence of a generalized model might result in non-controllable performance when the setting is modified since the parameters of the methods change as well.

In this paper, we present an accurate method to estimate the grid origin of a JPEG image (globally and locally), which in most cases is, as mentioned previously, the first step of image forgery techniques. The method is based on three steps: extracting the block artifacts, decomposing the image into several voters, and evaluating the accuracy of the statistical estimation based on the *a-contrario* method. The estimation is controllable with an *a priori* number of false alarms for each detection which will be detailed in section 9.3.3. Furthermore, the proposed method does not suffer of a complexity problem since it can be parallelized. We show several applications for the method.

## 9.3 An *a-contrario* detector

### 9.3.1 Grid extraction

Let  $I$  be the  $X \times Y$  luminance component of the input image and  $I(x, y)$  the intensity value of pixel  $(x, y)$ , with  $0 \leq x \leq X - 1$  and  $0 \leq y \leq Y - 1$ .

The method [103] detects the presence of block artifacts by computing the absolute value of the gradient magnitude image. Indeed, the block artifacts are represented by horizontal and vertical abrupt changes in the luminance value image. The difference filter used is defined as follows

- horizontally:

$$dI(x, y) = |I(x, y) - I(x - 1, y)|;$$

- vertically:

$$dI(x, y) = |I(x, y) - I(x, y - 1)|.$$

Other authors [102] use second order differences defined as

- horizontally:

$$dI(x, y) = |2I(x, y) - I(x + 1, y) - I(x - 1, y)|;$$

- vertically:

$$dI(x, y) = |2I(x, y) - I(x, y + 1) - I(x, y - 1)|.$$

As can be seen in figure 9.2, the first difference and second difference filters are highly affected by the edges and textures in the image. The latter image dynamics is neither vertical nor horizontal. To reduce these interferences, a cross difference filter proposed in [105] is defined by

$$dI(x, y) = |I(x, y) + I(x + 1, y + 1) - I(x + 1, y) - I(x, y + 1)|.$$



Figure 9.2: Compressed image “Goldhill” with quality factor  $Q = 85$  and close view of its cross difference, first order difference and second order difference.

As figure 9.3 shows, the higher the compression quality, the dimmer the JPEG grid. This explains the limits of grid extraction methods. Other techniques [102, 103] add a nonlinear correction to enhance the JPEG artifact over the strong edges in the image. Thanks to the locality of our method, we shall not need this correction.

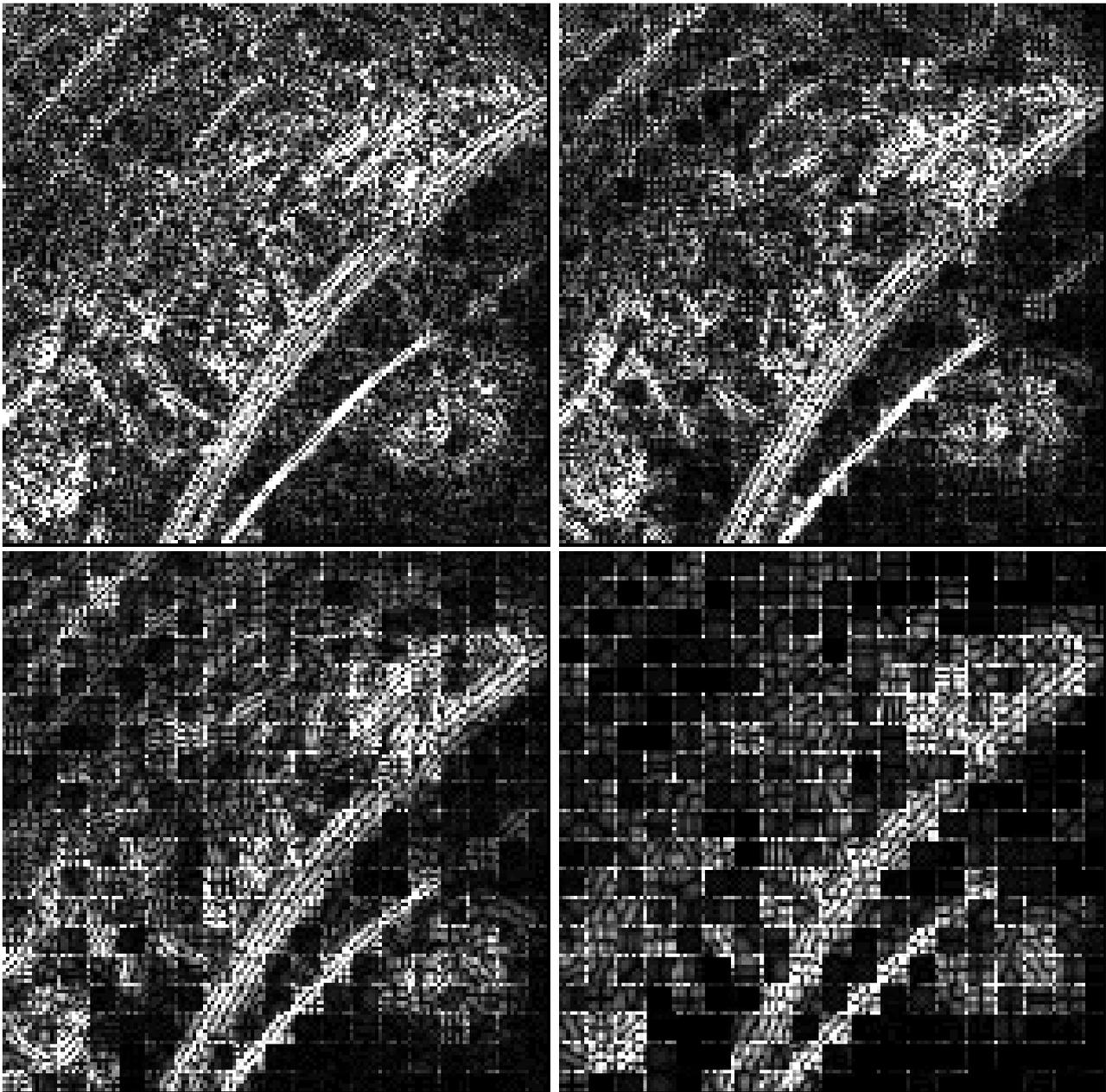


Figure 9.3: Comparison of cross images for different JPEG compression quality factors: 95, 85, 50 and 15.

### 9.3.2 The voting process

The voting process consists in decomposing the cross difference image into overlapping test blocks. Each block has a say and votes, independently, for its grid origin. The blocks are of size multiples of  $B_s$  (block size), which results in  $N$  independent (therefore it can be parallelized) tests to perform.

$$N = \frac{1}{4} \frac{X}{B_s} \left( \frac{X}{B_s} + 1 \right) \frac{Y}{B_s} \left( \frac{Y}{B_s} + 1 \right)$$

$B_s$  makes sense from  $B_s > 32$  so that each test block has at least 4 repetitions of the JPEG  $8 \times 8$  blocking artifact. This also implies that tampered regions as small as  $32 \times 32$  can already be detected. Of course if  $B_s$  is chosen too high for the application of forgery detection, it will not detect small modifications. Using overlapping test blocks is a form of multi-scale approach.

Each block  $b$  votes for the main grid by looking at the horizontal and vertical strict local maxima *separately*. Each direction (horizontal or vertical) has 8 different possible grid origins, since a typical JPEG block is of size  $8 \times 8$ . Algorithm 3 describes the voting process.

Algorithm 3 returns the total number of strict local maxima ( $n_x$  and  $n_y$ ) which represent the number

**Algorithm 3** vote - Block voting algorithm.

---

```

1:
    Input:  $b$ , block
    Output:  $v$ , vote
2: for all horizontal strict local maxima do
3:    $x \leftarrow$  first coordinate of strict local maximum
4:   vote_x [ $x \bmod 8$ ] ++
5: end for
6: for all vertical strict local maxima do
7:    $y \leftarrow$  second coordinate of strict local maximum
8:   vote_y [ $y \bmod 8$ ] ++
9: end for
10:  $n_x, n_y \leftarrow$  sum(vote_x), sum(vote_y) ▷ total number of local maxima
11:  $k_x, k_y \leftarrow$  max(vote_x), max(vote_y) ▷ number of votes of the chosen position
12:  $v \leftarrow n_x, n_y, k_x, k_y$  ▷  $v$  encodes all the information of the vote needed to compute the NFA later
13: return  $v$ 

```

---

of voters and the number of votes ( $k_x$  and  $k_y$ ) given for each block. Let us take the example of the “Goldhill” image, a  $512 \times 512$  grayscale image compressed with JPEG quality factor  $Q = 85$ , seen on the top left of figure 9.2. In this example  $B_s$  was set to 100 resulting in 225 test blocks. Table 9.1 shows the votes obtained on one block. The grid origin abscissa 0 has a strong relative majority. The second row shows the result for the horizontal extrema, with again a significant vote for the 0 position. The next section will explain the proposed detection criterion based on these statistics and controlling the number of false alarms.

	0	1	2	3	4	5	6	7
x	<b>604</b>	142	244	239	253	244	213	156
y	<b>509</b>	181	240	255	259	233	210	152

Table 9.1: Statistics extracted from an image block. The first row shows the number of local vertical extrema at each pixel abscissa, modulo 8.

### 9.3.3 Statistical validation

The proposed validation step is based on the *a-contrario* framework [106]: an event of interest is called meaningful if its occurrence is non-accidental, in the sense that the relation between its parts is too regular to be the result of an accidental arrangement of independent parts. Indeed, the grid estimation is based on the regularity of the pattern left by the JPEG compression.

The mathematical setting corresponds to a multiple testing procedure to control the expected number of false detections under a null model  $H_0$  [107]. The Number of False Alarms of the event  $e$  is defined as

$$\text{NFA}(e) = N_{test} P_{H_0}(e)$$

where  $N_{test}$  is the number of events to be tested and  $P_{H_0}(e)$  is the probability of observing an event  $e$  (or better) under the stochastic model  $H_0$ . An event  $e$  is called  $\epsilon$ -meaningful if and only if  $\text{NFA}(e) < \epsilon$ .

In our situation, the *a-contrario* framework determines whether a block’s vote is significant or not. Each block has two events to test: the horizontal and the vertical JPEG fingerprints. A test block is called significant when both of these events are  $\epsilon$ -meaningful, *i.e.*,  $\text{NFA}_x < \epsilon$  and  $\text{NFA}_y < \epsilon$ .

With  $N_{test} = N$  being the number of blocks, we have

$$\text{NFA}_x = N_{test} \mathcal{B}(n_x, k_x, 1/8),$$

$$\text{NFA}_y = N_{test} \mathcal{B}(n_y, k_y, 1/8),$$

where  $\mathcal{B}(n, k, p)$  is the binomial tail

$$\mathcal{B}(n, k, p) = \sum_{j=k}^n \binom{n}{j} p^j (1-p)^{n-j}.$$

As a simple convention, Desolneux et al. [106] suggest using  $\epsilon = 1$ , which is done in other fields using *a-contrario* methods [108]. Indeed, setting the value of  $\epsilon$  to 1 implies getting, on average, one false detection (one wrong block) per image.

In summary, for each test block we obtain a pair of NFA values for the most voted grid position. If both values are less than 1, we consider that the grid position is significant. This ensures that in an uncompressed image, there should be less than 1 false detection. The ensuing detection algorithm is summarized in Algorithm 4. Regarding the image “Goldhill” taken as an example, the result is an overwhelming vote for the grid position (0, 0). Indeed, 225 blocks over 225 blocks voted significantly for it. But the main point here is the NFA value of this detection. For the “most significant” block, with lowest NFA, this value is  $\text{NFA} = 10^{-785.487}$ . Most test blocks having extremely significant NFA values, ensures that the detection could “never happen by chance”.

---

**Algorithm 4** Summary of the proposed algorithm.

---

```

1:
   Input:  $I$ , RGB image
   Input:  $q$ , block size. The blocks are  $q \times q$  pixels.
   Output:  $\mathbf{S}$  Array that associates a block with a boolean indicating if it is significant
2:  $I \leftarrow$  Compute grayscale image ( $I$ )
3:  $I \leftarrow$  Compute cross difference image ( $I$ )
4:  $\mathbf{B} \leftarrow$  Decompose into blocks ( $I, q$ )
5: for  $b \in \mathbf{B}$  do
6:    $\mathbf{v} \leftarrow \text{vote}(b)$  ▷ Algorithm 3
7:    $\text{NFA}_x, \text{NFA}_y \leftarrow$  horizontal NFA( $\mathbf{v}$ ), vertical NFA( $\mathbf{v}$ )
8:    $\mathbf{S}[b] \leftarrow \text{NFA}_x < 1$  and  $\text{NFA}_y < 1$ 
9: end for
10: return  $\mathbf{S}$ 

```

---

## 9.4 Results on several applications

### 9.4.1 Grid detection

Grid detection is our main application as it represents the first step of most forgery detection algorithms. But this is not the only application. In image restoration, grid detection is also used to remove grid artifacts by a deblocking procedure [109]. To do so, it is useful to detect the grid in every case, and the hardest cases are when the compression level is low.

The challenge here is to detect a grid even for high quality compressed images. Table 9.2 shows that a very significant detection is possible up to  $Q = 95$ . Reliable detections for  $Q$  values up to 98 are also observed. On the other hand, the original uncompressed images do not produce detections (as expected). In contrast, the method proposed by [101], and applied to the same images than in Table 9.2, does not work for  $Q$  values over 90. The method proposed in [92] gets similar results to ours, yet requires to adjust properly two parameters, an attenuation value  $\alpha$  and a threshold  $\theta$ , while our method is parameter-free.

### 9.4.2 Crop detection

Table 9.3 reports the overall results, described in terms of correct percentage of the cropping position detection, depending on the compression ratio. The Kodak standard dataset [110] was used for that purpose, as it is good quality and compression-free. The detection rate decays significantly for  $Q \geq 90$ . Notice that cropping might have occurred just by chance with an origin compatible with

Image	Original	Q98	Q95	Q90
Barbara	$10^{0.822}$	$10^{0.016}$	$10^{-42.29}$	$10^{-280.40}$
Lena	$10^{1.257}$	$10^{1.335}$	$10^{-48.32}$	$10^{-418.30}$
Cameraman	$10^{0.159}$	$10^{-0.058}$	$10^{-0.338}$	$10^{-120.17}$
Goldhill	$10^{0.428}$	$10^{0.695}$	$10^{-39.16}$	$10^{-383.82}$
Peppers	$10^{2.203}$	$10^{0.056}$	$10^{-2.658}$	$10^{-131.47}$

Table 9.2: Results on standard images for several quality factors using the proposed algorithm.

the original grid. This actually happens one out of 64. Hence, there is a minimal of 1 over 64 false negative rate, which is clearly unavoidable.

Quality factor	Accuracy
$\leq 80$	100 %
90	91 %
95	70 %
99	41 %

Table 9.3: Results of the proposed method on cropped images of the Kodak database.

### 9.4.3 Copy-paste tampering detection

Again for this task, a detection based on a disparity in grid position in some block may fail with probability 1 over 64, when the copied area is placed so that its grid is aligned with the global grid. To test how detection can be based on JPEG grid misplacement, we used the database of tampered images [111]. Figures 9.4 and 9.5 represent tampered images and their ground truth. They come from the folder *C1\_panasonic* folder of the benchmark data [111], and were created by copying and pasting. The copied area is taken from the same image and its borders hidden in a smooth transition. However, we do not use this information for our detection, which would work equally well if the copied area came from a different JPEG image.

## 9.5 Conclusion

In this paper we have presented an accurate method for grid origin detection from a given image with no prior information. We proposed a way to validate this step with a fully unsupervised parameterless algorithm. Our grid only method is local enough to detect tampering such as crop and copy-paste, without any further step. It does not require extra information on the JPEG quantization and does not require involving the computation of DCT coefficients. In future work, we will aim at extending the method to detect locally double compression with a shifted JPEG grid (and therefore detecting a principal and a shifted grid).

Our method is only one of the steps of a tampering detection chain. For JPEG images, tampering detection attempts can go on, even if no discrepancy has been found in the grid origin throughout the image. But knowing the grid origin enables an accurate analysis of the statistics of block DCT coefficients, which is the classic next step in tampering detection. Being only one (significant) step in the detection chain, grid detection must be fully automatic and offer strong guarantees. For this reason, we also believe that the *a-contrario* methods, able to attach very small NFAs to detections, should also be used for the other detection tasks.



Figure 9.4: Image “Hedge” from database [111]: original, forged and results: over 1980 blocks, 1952 voted significantly for  $(0, 0)$  and 21 for  $(5, 4)$ .



Figure 9.5: Image “Supermarket” from database [111]: original, forged and results: over 1980 blocks, 1974 voted significantly for  $(0, 0)$  and 6 for  $(0, 5)$ .

# Chapter 10

## JPEG grid detection based on the number of DCT zeros

This work is the continuation of the research on forgeries after JPEG compression introduced in chapter 9, also the first-author work of Tina Nikoukhah. It was presented in the IEEE Conference on Computer Vision and Pattern Recognition Workshops and published in their proceedings [112].

### Abstract

This work proposes a novel method for detecting JPEG compression, as well as its grid origin, based on counting the number of zeros in the DCT of  $8 \times 8$  blocks. When applied locally, the same method can be used to detect grid alignment abnormalities. It therefore detects local image forgeries such as copy-move. The algorithm includes a statistical validation step which gives theoretical guarantees on the number of false alarms and provides secure guarantees for tampering detection. The performance of the proposed method is illustrated with both quantitative and visual results from well-known image databases and comparisons with state of the art methods.

### 10.1 Introduction

Image tampering is currently used massively on the web and continuously feeds fake news [113]. This issue has become important since digital image manipulation tools are available to the general public. Some social networks even allow to edit images and videos directly online. Since these platforms (Facebook, Instagram, Snapchat, etc.) delete all metadata for confidentiality reasons, there is a crucial need for journalists and mass-media producers to have access to tools for detecting forgeries from the image itself. Several such tools are readily available online, *Fotoforensics*, *Forensically*, *Ghiro* and *Reveal*<sup>1</sup> for instance. These tools provide a number of “tampering localization maps” [114] in the form of so-called image *heat maps* revealing suspicious alterations. However, those localization maps work only as trace enhancers and do not provide any solid proof of forgery. The result needs to be analyzed with care and interpreted by experts. We attempt here to fill in this gap by producing an automatic method using the tampering traces left in the image and taking an automatic decision about forgeries and their precise location. Indeed, even in absence of image metadata, much information about the history of the image is still present in the image itself. Many processes undergone by an image leave invisible but detectable traces all over the image. Following [115, 105, 101, 116, 113, 117, 118, 119], we address one of the most common processes, namely the JPEG compression. This compression leaves traces in the form of  $8 \times 8$  pixels blocking artifacts that produce a grid over the image.

Here we propose an automatic and statistically founded method for detecting the grid division of the

---

<sup>1</sup><https://fotoforensics.com>, <https://29a.ch/photo-forensics>, <https://www.getghiro.org>, <http://reveal-mklab.it>.gr

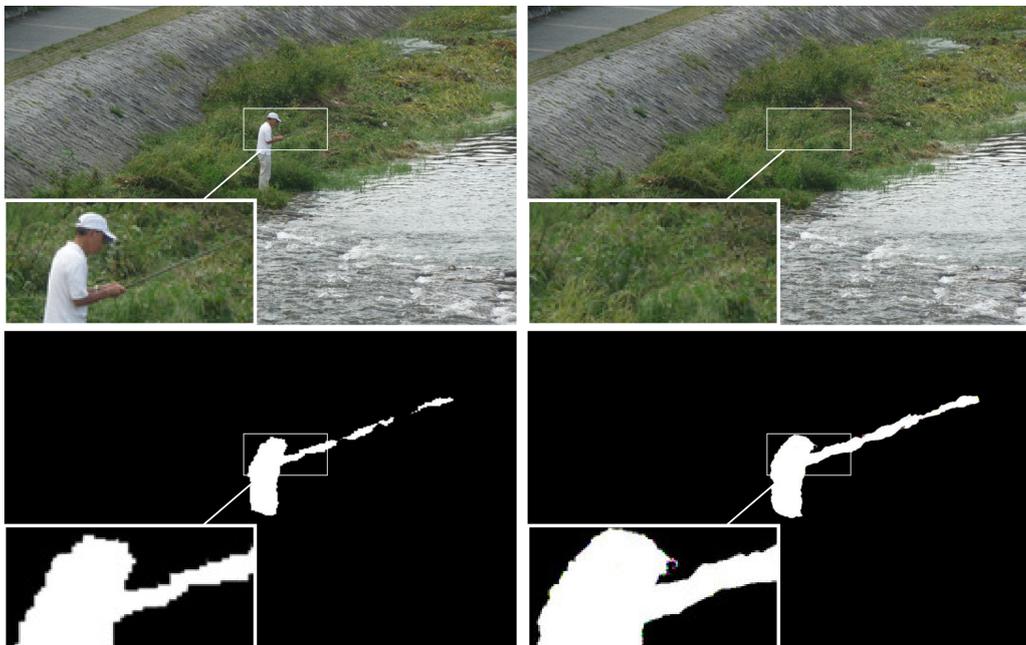


Figure 10.1: Result of the proposed method, from top left to bottom right: original image, forged image (input of the method), detected forgeries, ground-truth. The fisherman and the fishing rod have been removed. The resulting detected mask does not require visual interpretation.

image. The proposed method, hereafter referred to as ZERO, performs a global grid detection and then applies it more locally to detect forgeries. Indeed, image splicing generally breaks locally the original grid alignment. A locally detected grid therefore may contradict the global grid and become a reliable cue that a forgery took place. We evaluate ZERO against several state of the art algorithms on publicly available datasets. The detection does not require a human expert’s decision as shown in Figure 10.1, and is associated with theoretical guarantees.

The paper is organized as follows. A brief description of the JPEG compression algorithm is presented in Section 10.2, followed by a review of the state of the art in Section 10.3. In Section 10.4, the proposed method is described and its application to forgery detection is detailed in Section 10.5. Finally, evaluations of the performances of the method are presented in Section 10.6. A discussion concludes this article.

## 10.2 JPEG compression

The JPEG image format is widely used by most digital cameras (including smartphones) and by social networks to share images on the Internet [120, 91]. First, the image is converted from the  $RGB$  to the  $YC_bC_r$  colorspace. The  $Y$  channel is the luminance component while  $C_b$  and  $C_r$  correspond to the chroma components. For simplicity, we will focus on the  $Y$  channel<sup>2</sup> since it contains most of the image visual information, which we will refer to as the luminance image. The luminance image is then partitioned into non-overlapping  $8 \times 8$  pixel blocks and the 2D discrete cosine transform (DCT) type II is applied to each of these blocks. Due to the independent encoding of the blocks, pixel discontinuities are introduced across the block boundaries of the decompressed image.

Each of the  $8 \times 8$  blocks undergoes a quantization step performed in the spectral domain. A quantization table (related to the compression quality) provides a factor for each DCT component. At this step, some DCT coefficients are cancelled out when they have a small value relative to the quantization factor. After this step, all  $8 \times 8$  blocks have a number of zeros that depends both on the compression quality and on the image content. Finally, the quantized DCT coefficients are losslessly compressed by exploiting, among other things, the presence of zero values.

---

<sup>2</sup> $Y = 0.299R + 0.587G + 0.114B$ .

## 10.3 State of the art

There are several tools for detecting forgeries based on JPEG compression traces. The most famous ones (used by mass-media online) are ELA (Error Level Analysis) [121] and GHOST [116], which are very similar. Both attempt to detect JPEG compression ratio differences throughout the image. In JPEG forensics, the main methods are either based on the histograms of DCT coefficients [115, 119] or based on the detection of a higher contrast at the block edges [105, 101].

Three methods [117, 118, 122] are closely related to ours as they detect forgeries by locating inconsistencies of JPEG blocking artifacts. In [118], the image is filtered based on local derivatives, weak edges are detected, and the coherence with an aligned  $8 \times 8$  grid is measured. A feature corresponding to the local strength of the blocking pattern is extracted. Feature variations indicate local absence or misalignment of the grid and can be considered as a tampering cue. In [117], the authors use the artifact measure introduced by Fan and Queiroz [101]: their method evaluates multiple grid positions with respect to a fitting function. Areas with low contribution are identified as inconsistent with the main grid and therefore potentially tampered. An image segmentation step is introduced to differentiate between inconsistencies produced by tampering and those attributable to image content. In [122], the authors apply the filter proposed by Chan and Hsu [105] to reveal these blocking artifacts before using a statistical method to increase the reliability of the detection.

These methods make it possible to detect what is undoubtedly one of the most commonly used tampering schemes: the copy and paste of image parts which break the alignment of the original grid, either because of its location or because of transformations (scaling, rotation, etc.) of the manipulated area. Another way to alter an image is by simply cropping it to remove undesirable parts of the photographed scene. This method, frequent in photojournalism, can significantly alter the interpretation of a scene. To detect cropping, Li et al. [118] and Nikoukhah et al. [123, 122] detect the grid globally and exploit the fact that its origin may no longer be at  $[0, 0]$ . Our method, being based on the detection of the global JPEG grid, is also able to detect this type of manipulation.

Finally, besides the spectral and spatial methods, a third way has been introduced based on the principle that JPEG compression has the goal to minimize the file size [123]. Therefore, to detect the JPEG grid, the method compresses the image with all 64 possible grids and selects the one yielding the shortest file. Our method extends this idea as we decide to pick the likeliest JPEG blocks as those containing the largest number of zero DCT coefficients.

## 10.4 JPEG grid detection

As described before, the JPEG algorithm sets to zero some of the DCT coefficients of  $8 \times 8$  blocks. Based on this fact, the core of the proposed method is thus to count the total number of zeros of each hypothesized DCT block. In the presence of JPEG compression, this number should be maximum when the  $8 \times 8$  block is aligned with the JPEG grid. Indeed, non-aligned blocks include additional discontinuities due to blocking artifacts, leading to larger DCT coefficient values relative to the aligned block, see Figure 10.2 (right). A statistical test is used to confirm that a JPEG grid is actually present.

### 10.4.1 Voting process

The first thing to explain is how to determine which DCT coefficient are “zeros”. During JPEG decompression, an inverse DCT is performed on each block, transforming the integer DCT coefficients to real pixel values. Then, those pixel values are quantized to produce an integer image. Notice that this quantization step is different from the one during the compression. This pixel quantization step, of course, also modifies the corresponding DCT values. Thus, a DCT coefficient that was put to zero during compression does not keep an exact zero value after decompression. Yet it remains close to zero. We propose to count the number of coefficients with absolute values smaller than 0.5. This allows to discriminate zeros even when DCT coefficient quantization is at its finest rate, namely to integer values.

Each pixel may belong to 64 different overlapping  $8 \times 8$  blocks, as illustrated in Figure 10.2. In the

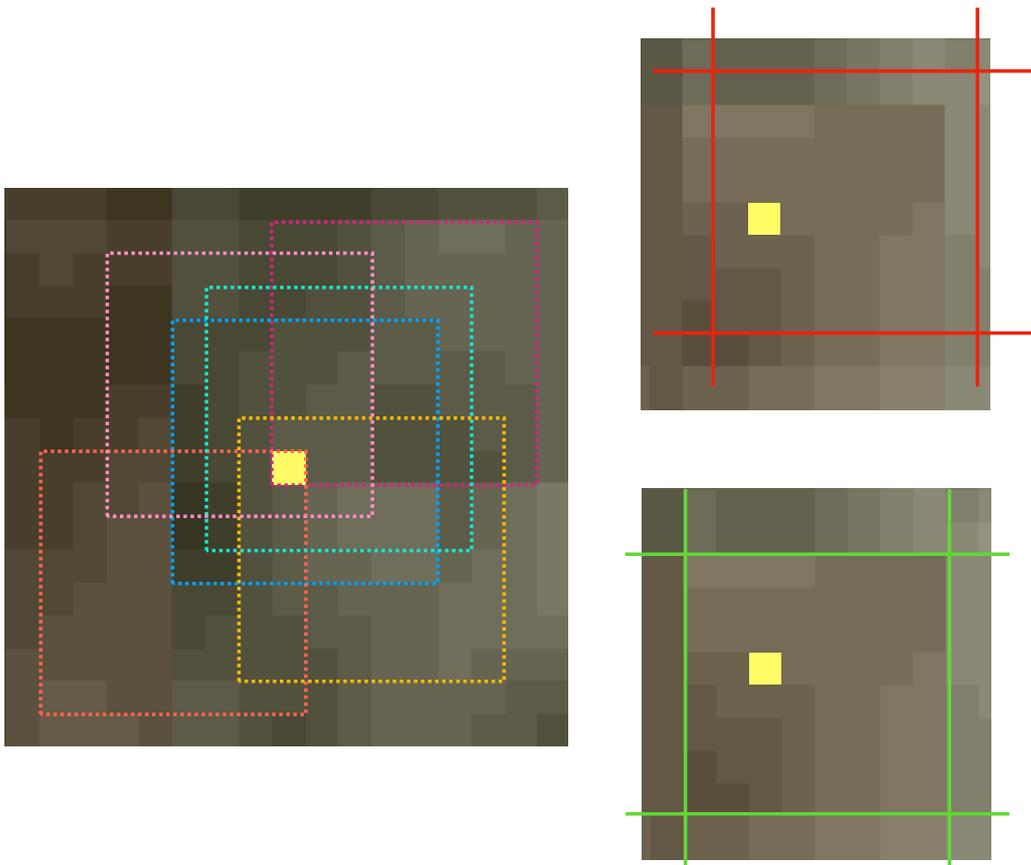


Figure 10.2: Each pixel (yellow) belongs to 64 different  $8 \times 8$  blocks of the image. Six of them were drawn in different colors on the left. Top right shows (in red) the position of a patch not aligned with the grid. Bottom right shows (in green) the position of the patch containing the pixel matching the JPEG grid.

first step of the method, those 64 blocks are evaluated for each pixel. The 64 DCT are computed as well as the corresponding number of zeros. Each pixel votes for the grid origin of the block with most zeros. In the case of a tie, the pixel does not vote.<sup>3</sup>

Performing the count as described requires computing 64 DCTs per pixel, but every block is shared with 64 other pixels and this can be exploited to avoid recomputing the DCT. Algorithm 5 describes the procedure. A table is used to keep track of the largest number of zeros found for each pixel. The DCT of every  $8 \times 8$  block in the image and its number of zeros are computed. Every pixel included in the block is checked and the table of zeros is updated when the current block has more zeros than in other blocks previously evaluated. The table of votes is also updated to the grid origin corresponding to the block with more zeros (`GridAlignedWith(b)`), or to `NON VALID` in case of a tie.

Figure 10.3 shows two examples of vote maps. On the left, the image is uncompressed and we observe a random vote map. On the right, the same is shown for a JPEG compressed image.

---

**Algorithm 5** Compute grid votes.
 

---

```

1:
    Input: luminance channel  $L$  defined on  $\Omega$ 
    Output: grid vote map
2: votes( $\Omega$ )  $\leftarrow$  NON VALID                                ▷ initialize votes
3: zeros( $\Omega$ )  $\leftarrow$  0                                       ▷ initialize number of zeros
4: for  $b \in \text{Blocks}_{8 \times 8}(\Omega)$  do                             ▷ loop on all  $8 \times 8$  blocks
5:      $d \leftarrow \text{DCT}(L(b))$                                        ▷ 2D DCT-II of the block
6:      $z \leftarrow \sum_{d_i \in d} \mathbb{1}_{|d_i| < 0.5}$                        ▷ number of zeros in the block
7:     for  $[x, y] \in b$  do
8:         if  $z = \text{zeros}[x, y]$  then                                ▷ tie, do not vote
9:             votes $[x, y] \leftarrow$  NON VALID
10:        else if  $z > \text{zeros}[x, y]$  then
11:            zeros $[x, y] \leftarrow z$ 
12:            votes $[x, y] \leftarrow \text{GridAlignedWith}(b)$ 
13:        end if
14:    end for
15: end for
16: return votes

```

---

## 10.4.2 Statistical validation

When analyzing a JPEG image, the most voted grid probably corresponds to the right one. But, even in uncompressed images, one of the grids will get more votes than the others, probably by a small margin. A statistical criterion is therefore needed to decide whether this prominence is caused by JPEG compression or not.

The proposed validation procedure is based on the *a contrario* theory [106], which relies on the non-accidentalness principle [124, 125]. Informally, this principle states that there should be no detection in noise. In the words of D. Lowe, “*we need to determine the probability that each relation in the image could have arisen by accident,  $P(a)$ . Naturally, the smaller that this value is, the more likely the relation is to have a causal interpretation*” [124, p. 39]. This principle has shown its practical use for detection purposes such as line segment detection [126], vanishing points detection [127], anomaly detection [128], or forgery detection [129, 122].

In our context, we need to assess the probability that a given grid origin gets a large number of votes purely by chance. To that aim, a stochastic null model  $H_0$  for the votes is required. It is here easily given by Laplace’s principle of indifference: in absence of JPEG compression, each of the 64 blocks containing a given pixel would have the same chance of being the one with the largest number of

---

<sup>3</sup>There is relevant information when two or more blocks have the same number of zeros. However, exploiting this information would make the statistical evaluation more complex. Given that the method is already reasonably sensitive, we preferred to keep a simple formulation.

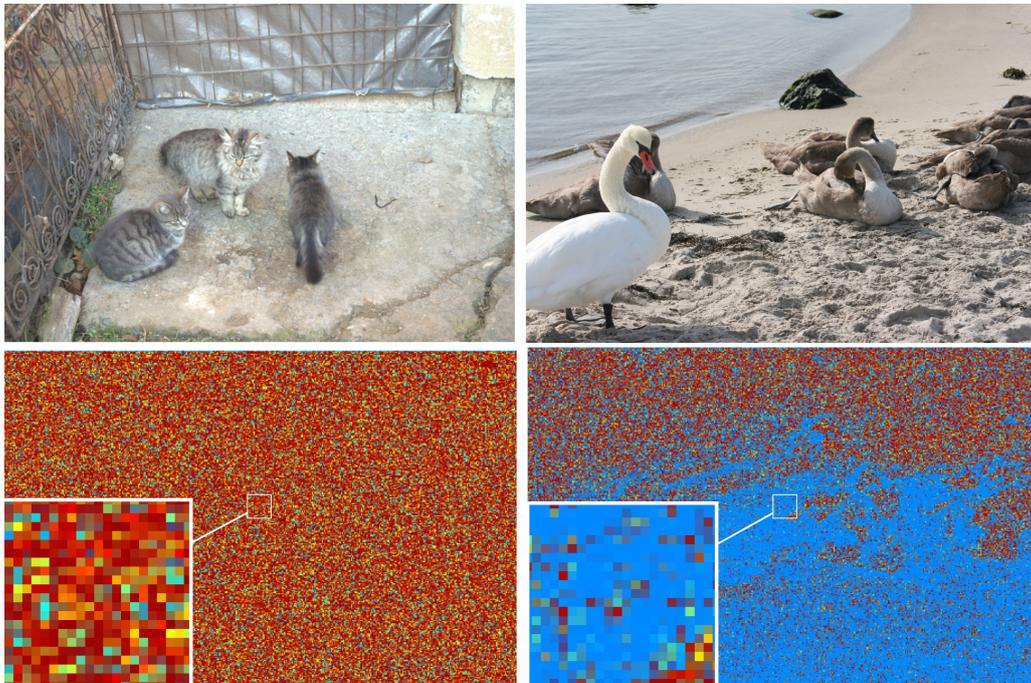


Figure 10.3: Left: Uncompressed image and its vote map. Right: JPEG compressed image and its vote map. In both cases, the pixels which return a NON VALID vote (a tie) are shown in dark red.

zeros; that would depend on the image content and there is no reason to suppose that it is synchronized with a particular  $8 \times 8$  grid origin. However, the votes of neighbor pixels are not independent, even in noise images. Indeed, there are always blocks that are local maxima of the number of zeros, and those blocks get the votes of every pixel belonging to it. Figure 10.4 shows a vote map obtained in an image of noise. Votes are correlated within a distance of 8 pixels; on the other hand, pixels at distance larger than eight are largely uncorrelated. Thus, we define a stochastic null model  $H_0$  for votes at distance eight in which votes are independent and uniformly distributed among all the 64 grid origins.

Let us suppose that we are observing a square patch of an image where the number of votes for a given valid grid origin is counted at a distance of eight pixels. Let us say that  $k$  votes are counted for that valid grid among a total of  $n$  votes. Under the null hypothesis  $H_0$ , votes for the given grid origin become Bernoulli random variables with probability  $\frac{1}{64}$ . So under  $H_0$ , the number of votes becomes a random variable  $K$  and, given the independence of votes (at distance larger than eight), it follows a binomial distribution of parameter  $p = \frac{1}{64}$ . Thus,

$$\mathbb{P}(K \geq k) = \mathcal{B}(n, k, p) = \sum_{j=k}^n \binom{n}{j} p^j (1-p)^{n-j},$$

where  $\mathcal{B}(n, k, p)$  is the tail of the binomial distribution. Given an observed number of votes  $k$ ,  $\mathbb{P}(K \geq k)$  is the probability of obtaining at least  $k$  votes under  $H_0$ . When this probability is small enough, there exists evidence to reject the null hypothesis and declare that a significant grid origin was found. However, the multiplicity of tests needs to be taken into account. Indeed, if 100 tests were performed, it would not be surprising to observe an event that appears with probability 0.01 under random conditions. The number of tests  $N_T$  needs to be included as a correction factor, as it is standard in statistical multiple hypothesis testing [107].

Following the *a contrario* methodology, we define the Number of False Alarms (NFA) of a candidate grid  $g$  on a given window  $w$  as

$$\text{NFA}(g, w) = N_T \mathbb{P}(K \geq k). \quad (10.1)$$

One can show [106] that under the null hypothesis  $H_0$  the expected number of false alarms with

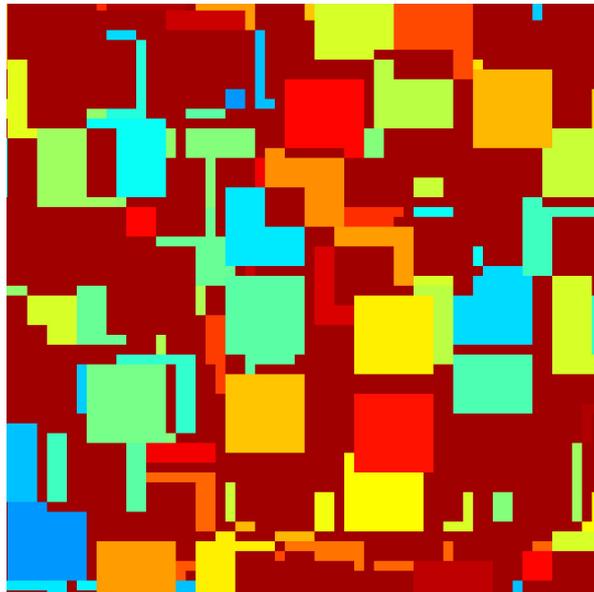


Figure 10.4: Zoom on a vote map for an image of Gaussian noise. Each color represents a vote for a given grid origin. The dark red color corresponds to NON-VALID votes (the ties). One can observe entire blocks of  $8 \times 8$  pixels voting for the same origin; this is the case when a block has a local maximum of number of zeros.

$\text{NFA}(g, w) < \varepsilon$ , is bounded by  $\varepsilon$ :

$$\mathbb{E}_{H_0} \left[ \sum_{(g,w) \in \mathcal{N}_T} \mathbb{1}_{\text{NFA}(g,w) < \varepsilon} \right] < \varepsilon, \quad (10.2)$$

where  $\mathcal{N}_T$  is the set of  $N_T$  tests. As a result,  $\varepsilon$  corresponds to the mean number of false detections per image under  $H_0$ . In most practical applications, the typical value  $\varepsilon = 1$  is suitable; we will set it once and for all in our application as well. With this choice, the expected number of false detections per image under  $H_0$  is guaranteed to be upper-bounded by 1.

The same criterion will be used for the whole image as well as for all sub-images. We want the tests to be selective enough to discriminate a JPEG grid using only a local region of the image. As we will see later, this also allows to detect local forgeries. Thus, every square window of a  $X \times Y$  pixels image is included in the family of tests and the 64 grid origins are tested for each one. Then, the number of tests can be approximated by

$$N_T = 64 \cdot XY \cdot \sqrt{XY}, \quad (10.3)$$

where  $\sqrt{XY}$  gives a rough estimation of the possible window sides, and  $XY$  gives the number of possible positions for the square windows of a given size. All in all, given a window to be analyzed, the grid origin with the maximum of votes is selected and its number of votes at distance eight pixels is counted. Then, the NFA is given by

$$\text{NFA} = 64 \cdot XY \cdot \sqrt{XY} \cdot \mathcal{B}(n, k, p). \quad (10.4)$$

A JPEG grid is detected when  $\text{NFA} < 1$ .

According to the theory, to test a grid we must count one vote out of eight in both directions. These are the votes inside the window with coordinates  $(x_0 + 8i, y_0 + 8j)$  for integers  $i$  and  $j$ . This test must be performed for all other 64 grids with  $x_0$  and  $y_0$  in  $\{0, 1, \dots, 7\}$ . A simpler way to evaluate a lower bound for the number of votes in the most voted grid is as follows. Instead of counting votes at distance of eight pixels for those offsets, we can count every vote and divide the number by 64. Indeed, let  $v$  be the total of votes in the window for the given grid. If those votes were equally distributed on the sub-samplings, one would have  $k = \frac{v}{64}$  for each of the sub-samplings. If not, necessarily one of the sub-samplings will have more votes. Hence we can deduce that there is at least one of those sub-samplings with  $k$  votes satisfying  $k \geq \frac{v}{64}$ . So by counting every vote and

dividing the count by 64 we are considering the worst case and we are sure that a detected grid is meaningful. Naturally, the count of votes for every pixel in the window is also divided by 64. Algorithm 6 describes the ensuing JPEG grid detection method.

---

**Algorithm 6** JPEG grid detection.
 

---

```

1:
    Input: image  $I$  of size  $X \times Y$ 
    Output: main grid  $G$ 
    Output: NFA of the main grid
2:  $L \leftarrow \text{Luminance}(I)$ 
3:  $\text{votes} \leftarrow \text{ComputeVotes}(L)$  ▷ algorithm 5
4:  $G \leftarrow \arg \max_{g \text{ is VALID}} \sum_{x,y} \mathbb{1}_{\text{votes}(x,y)=g}$  ▷ most voted valid grid
5:  $v \leftarrow \sum_{x,y} \mathbb{1}_{\text{votes}(x,y)=G}$  ▷ number of votes for main grid
6:  $\text{NFA} \leftarrow 64XY\sqrt{XY} \cdot \text{BinTail}\left(\frac{XY}{64}, \frac{v}{64}, \frac{1}{64}\right)$ 
7: if  $\text{NFA} < 1$  then
8:     return  $G, \text{NFA}$  ▷ JPEG grid found
9: else
10:    return  $\emptyset$  ▷ JPEG grid not found
11: end if

```

---

## 10.5 Forgery detection

Our JPEG grid detection can be performed globally but also at every square window of the image. When a local region has a meaningful grid that is different from the main one, it means that it is a foreign grid and so the result of a forgery. Indeed, when part of a JPEG image is copied and pasted, it retains its grid traces. In 63 over 64 times (assuming that the forger did not explicitly align the grid), the grid origin will not correspond to the main one, thus allowing its detection. This is true whether it is a case of copy-move from the same image or when the copied part is taken from a different JPEG image.

The same algorithm as described in Section 10.4 can be applied directly on every square window. But this would be computationally expensive. Instead, we propose a heuristic method using a greedy algorithm to accelerate the search for forged regions. Nevertheless, the final validation uses the same statistical test used for the global grid.

Algorithm 7 describes the method. The vote map is partitioned into connected regions sharing the same grid vote. A region growing algorithm is used for partitioning the vote map: starting from a seed pixel  $(x, y)$ , the neighbor pixels are iteratively aggregated when voting for the same grid. As Figure 10.4 shows, votes for the same grid often have gaps, so a relaxed notion of neighborhood is needed. A window with a meaningful grid origin must have a vote density of at least  $\frac{1}{64}$  for the right grid origin. Thus, the votes for the right grid should not be further away than eight pixels. To allow for some variation in the distribution, we set this neighborhood size a little larger and use  $W = 12$ .

Then, for each region with a valid grid origin different from the main one, a square bounding box is computed and the statistical test is performed with the NFA framework introduced before. When a meaningful region is found with a grid origin different from the main one, the pixels in the region (which all voted for the same grid) are marked in a forgery mask. Figure 10.5 shows an example. After a region is evaluated, its votes are marked as NON VALID to gain time by preventing the same pixels from being explored again. Due to variations in the number of votes, the raw forgery mask contains holes. To give a more useful forgery map, these holes are filled by a mathematical morphology closing operator [130] with a square structuring element of size  $W$  (the same as the neighborhood used in the region growing). Figure 10.5 shows an example of the final forgery mask.

Figure 10.6 illustrates some limitations of the proposed method. Forgeries are detected as regions in which the local grid does not agree with the global grid. This means that when the grid of the forged

**Algorithm 7** Forgery detection.

---

```

1:
   Input: grid vote map defined on  $\Omega$  of size  $X \times Y$ 
   Input: main grid  $G$ 
   Input: neighborhood size  $W = 12$ 
   Output: forgery mask
2:  $\text{mask}(\Omega) \leftarrow \text{FALSE}$  ▷ initialize forgery mask
3: for  $[x, y] \in \Omega$  do
4:   if  $\text{votes}[x, y]$  is VALID and  $\text{votes}[x, y] \neq G$  then
5:      $R \leftarrow \text{GrowRegion}(\text{votes}, x, y, W)$ 
6:      $B \leftarrow \text{BoundingBox}(R)$ 
7:      $N \leftarrow \max(B_x, B_y)$  ▷ size of square bounding box
8:      $\text{NFA} \leftarrow 64XY \sqrt{XY} \cdot \text{BinTail}\left(\frac{N^2}{64}, \frac{|R|}{64}, \frac{1}{64}\right)$ 
9:     if  $\text{NFA} < 1$  then ▷ Forgery found
10:       $\text{mask}(R) \leftarrow \text{TRUE}$  ▷ mark tampered region
11:     end if
12:      $\text{votes}[R] \leftarrow \text{NON VALID}$  ▷ do not test again in R
13:   end if
14: end for
15:  $\text{mask}(\Omega) \leftarrow \text{Closing}(\text{mask}, W)$  ▷ fill holes in mask
16: return mask

```

---

regions aligns perfectly with the global grid, our method will fail to detect the forgery. Nevertheless, this happens only once for every 64 positions. In a saturated region, the DCT coefficients of the blocks are all equal to zero except for the DC coefficient. The number of zeros are tied and the votes are all non-valid. This means that it is impossible to distinguish the JPEG grid in saturated regions. Since no valid JPEG grid can be found, it will never disagree with the global grid and therefore saturated parts of a forgery cannot be found. However, as soon as a part of the forgery is not saturated it can be detected as it is shown in Figure 10.6. Another limitation is when the forged region is too small. Since the statistical test must be satisfied to detect a forgery, there is a minimal detectable region size that depends on the image size, the JPEG compression quality and the image contents.

## 10.6 Experimental results

In this section we evaluate the proposed method on two tasks: global grid detection and forgery detection. Comparisons with other available methods are performed which illustrate the superiority of our approach, both in terms of low false rate and accuracy on true detections.

### 10.6.1 Grid detection

Grid detection is our main application as it represents the first step of most forgery detection algorithms but this is not the only application. In image restoration, grid detection is also used to remove grid artifacts by a deblocking procedure [109]. To this aim, it is useful to detect the grid in every case, and the hardest cases are when the quality factor is high. Our first evaluation was performed with cases where no detection should be obtained. This experiment is important to illustrate the main strength of the proposed algorithm: it gives a principled method to decide whether the image has undergone JPEG compression or not. The first dataset is composed of 200 images of noise following a Gaussian distribution. Two image sizes were generated:  $500 \times 500$  and  $1000 \times 1000$ . We also used the UCID [131] uncompressed image collections (886 images) and Kodak [110] (24 images). The method's performance is compared to three other methods in Table 10.1. ZERO is the sole to produce no false detection.

To illustrate the validity of the proposed approach, the second experiment is applied to 12 288 images generated from the Kodak [110] uncompressed image database. The 24 images (of size  $768 \times 512$ ) were

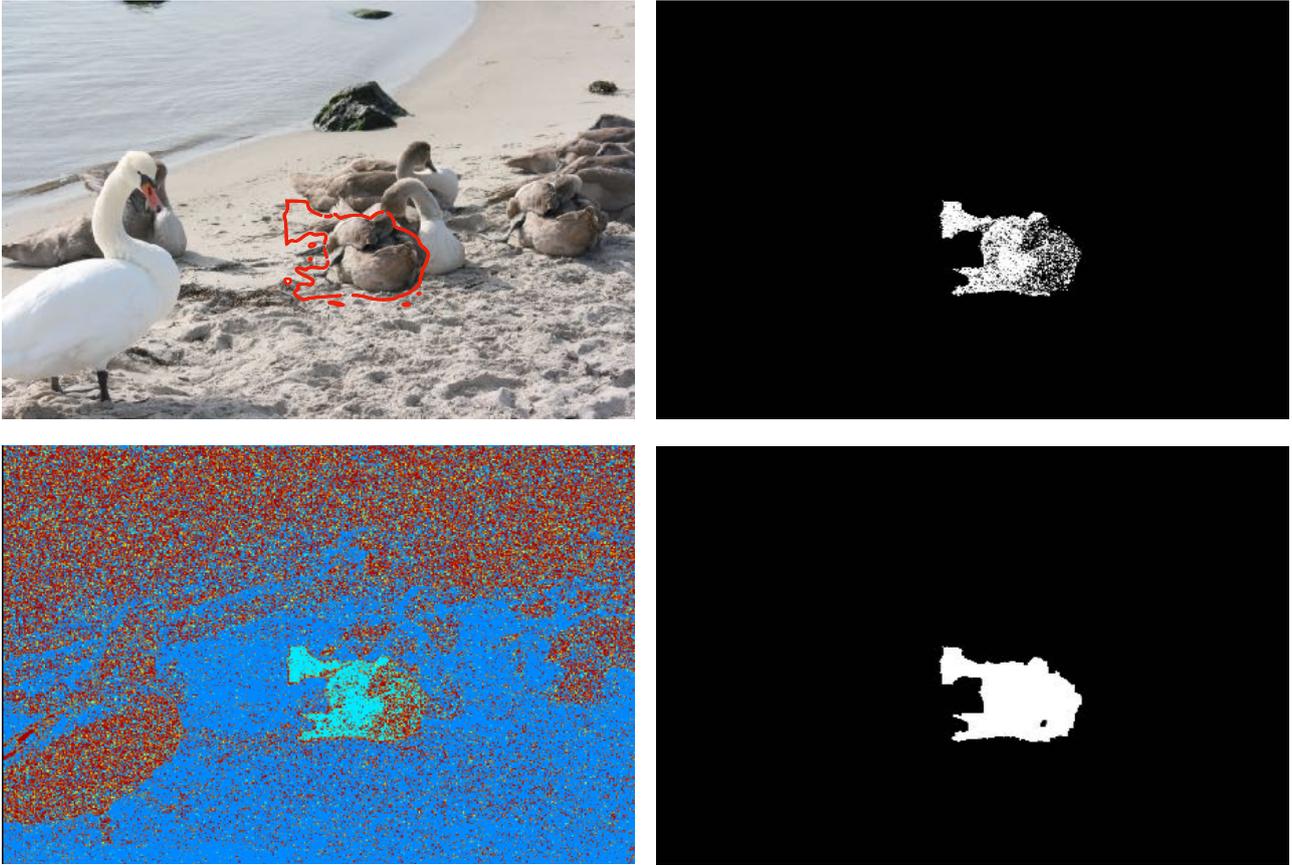


Figure 10.5: Up-left: an image with a tampered region indicated in red. Down-left: grid origin vote map. Up-right: raw forgery mask. Down-right: final forgery mask after holes filled by a mathematical morphology closing operation.

		Dataset		
		Noise	UCID [131]	Kodak [110]
BLK [118]	% true	—	—	—
	% false	100	100	100
GOD [122]	% true	—	—	—
	% false	0	0.6	0
SGOD [123]	% true	—	—	—
	% false	0	0.5	0
ZERO	% true	—	—	—
	% false	0	0	0

Table 10.1: Results of the proposed method compared to BLK, GOD and SGOD on uncompressed images. Percentage of images where a JPEG grid was detected. In this case, every detection is a false detection.

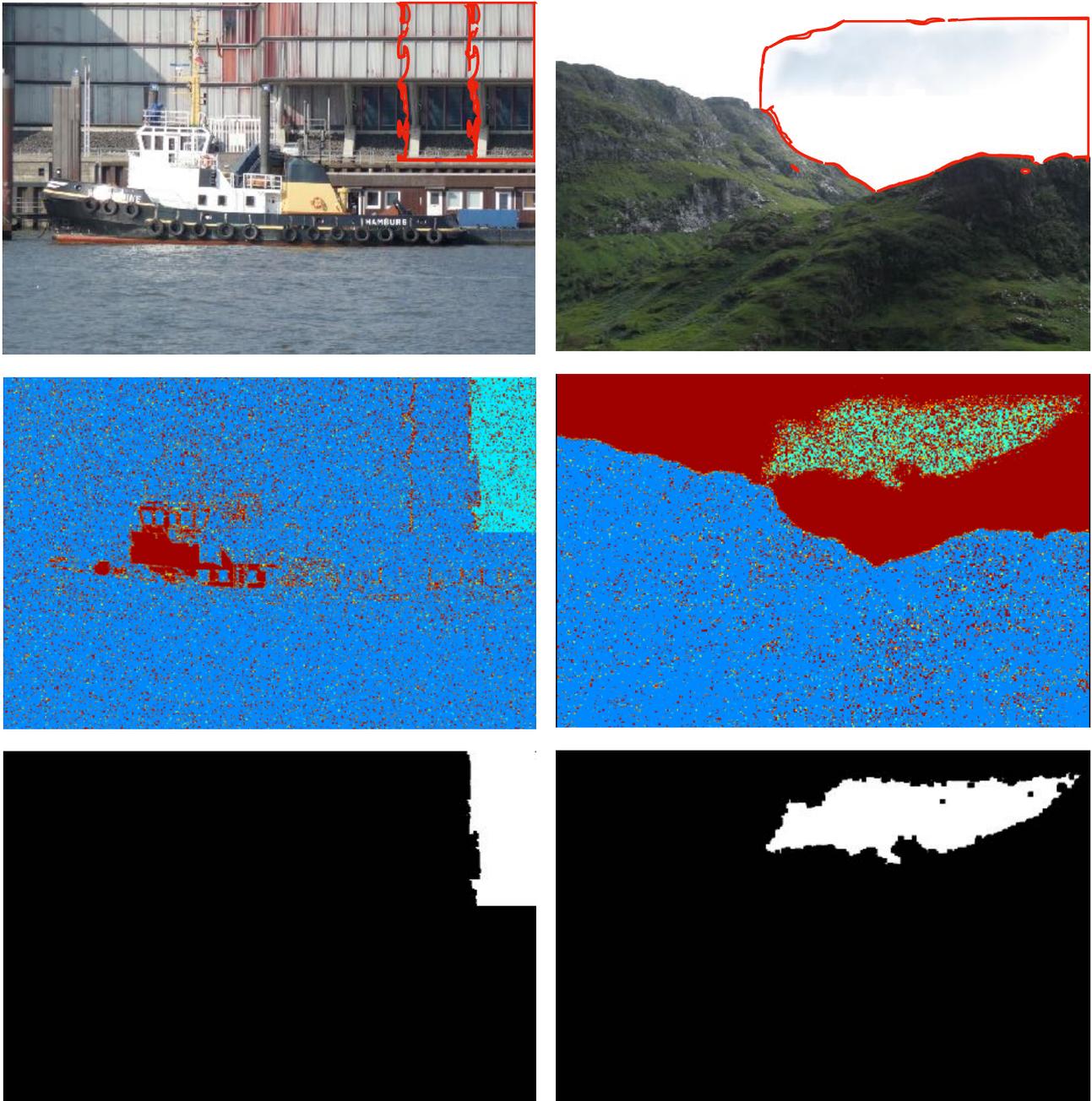


Figure 10.6: Up: images with tampered regions indicated in red. Middle: grid origin vote maps. Down: detected forgery masks. On the left, an example of a missed detection: one of the two forged regions was not detected because its local grid was correctly aligned with the global grid. On the right, an incomplete detection caused by saturation in the image.

		JPEG quality factor				
		$\leq 80$	90	95	98	99
BLK [118]	% true	97	95	85	31	0
	% false	3	5	15	69	100
GOD [122]	% true	100	91	70	55	41
	% false	0	0.003	0.05	0.06	0.1
SGOD [123]	% true	100	100	100	50	0
	% false	0	0	0	0	0
ZERO	% true	100	100	100	100	100
	% false	0	0	0	0	0

Table 10.2: Results of the proposed method compared to BLK, GOD and SGOD on 12 288 compressed and cropped images generated from the Kodak [110] database. Percentages of true and false JPEG grid detections.

compressed, using the `imagemagick` tool to increasing quality factors (50, 60, 70, 80, 90, 93, 95, 98 and 99), then cropped into the 64 different positions to test all possible grid positions. The results are shown in Table 10.2.

This application is useful by itself. Indeed, knowing if an image has undergone JPEG compression is important. Moreover, knowing its grid origin can tell whether an image has been cropped. Table 10.2 shows a perfect detection of JPEG compression and cropping after a JPEG compression even with a very high quality (99%). JPEG compression at 100% is not detectable by the proposed method as it does not increase the number of zeros in the DCT. Even in those cases, the proposed method gives no false detection.

## 10.6.2 Forgery detection

To test how detections can be based on JPEG grid misplacement, we used a database of tampered images created by copy-paste [132]. The copied area is taken from the same image and its borders hidden in a smooth transition. However, the proposed method would work exactly the same if the copied area came from a different JPEG image. A detection based on a disparity in the grid position in some blocks may fail with probability  $1/64$  when the copied area is placed so that its grid is aligned with the global grid.

The proposed algorithm gives back two types of important information: the main JPEG grid origin detection and the forgery detection when it is the case. The database [132] contains 48 pairs of original and forged images. On this dataset, we compare the results of GOD [122], CMFD [133] and ours. CMFD by Cozzolino et al. [133] proposes a method that allows a precise and accurate detection of a copy-move forgeries inside a single suspicious image. Therefore this method is particularly appropriate on the tested images and is used as reference for these types of forgeries. We used the implementation provided by [134]. Note that our method does not need the forged region to come from the same image. Both GOD [122] and ZERO are able to filter out the 11 pairs that are either not JPEG compressed, or with a quality factor of 100%. On the 37 remaining images, the proposed method is able to detect most forgeries with no false detection. The quantitative results are reported in Table 10.3. Regarding the two missed forgeries, one is caused by the copied area having the same grid origin as the global image, and the other one is due to its small size.

We also qualitatively compared our method to other forgery detection methods based on compression traces analysis in Figure 10.7. This allows us to compare with state-of-the-art methods that only produce heatmaps.

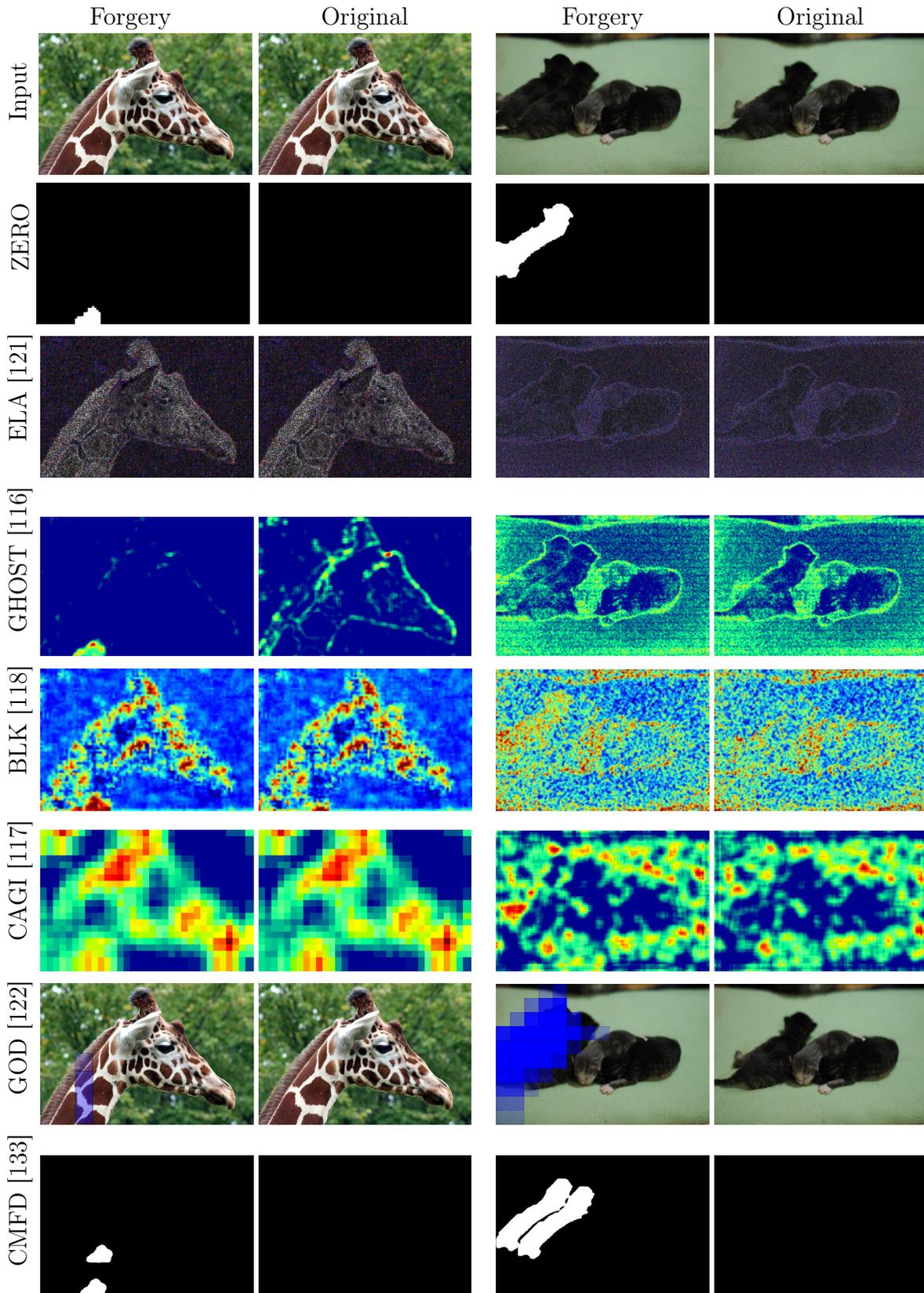


Figure 10.7: Results of the proposed method compared to the JPEG state-of-the-art methods. The first one produces a difference image, the three following heat maps and the last two masks. The methods are applied to the forged image and also its original source.

		ZERO	GOD [122]	CMFD [133]
Original	true	—	—	—
	false	0	6	5
Forged	true	35	23	44
	missed	2(+11)	21(+4)	4

Table 10.3: Quantitative results on the dataset [132], containing 48 pairs of original and forged images. 11 of the images are correctly detected as not JPEG by ZERO (4 for GOD). Since these image are not JPEG the forgery cannot be detected; they are reported in parenthesis.

## 10.7 Conclusion

This paper presented a novel JPEG grid detection and tampering localization method based on the number of zeros in the DCT blocks. It has a high accuracy detecting JPEG compression up to quality factor of 99%. It performs reliable reverse engineering and detects forgeries by giving an automatic, localized, and reliable result without requiring any human interpretation. The proposed algorithm is efficient; the bottleneck is the computation of the vote map, which has about the same complexity of 64 JPEG compressions. Color information and handling ties in the number of zeros will be explored in future work.

**Reproducibility.** The source code of the proposed method as well as an online demo are available at <https://github.com/tinankh/ZERO>

# Chapter 11

## Research plan

My research project for the following years is focused on three main axes:

- **Reproducible research**
- **Design of new Earth-observation satellites and improvement of existing ones**
- **Reverse engineering of images**, with diverse applications

Since the beginning of my scientific career, reproducible research has been my working methodology and indeed I believe strongly on the need to have verifiable and repeatable research results. Specifically in computational disciplines, where the scientific community needs to have access to the complete procedures (including source code and data) allowing to obtain the same published results.

I am one of the IPOL founding editors and I have supported the initiative from its very beginning for almost ten years now. At this moment reproducible research is one of the pursued objectives of CMLA and therefore I plan to apply it to my research and that of my PhD candidates. I have been actively working on the development of the IPOL platform since 2012. In February 2014 I started working on an advanced version of the demo system with the help of an engineering team that I supervise and coordinate. Nowadays IPOL accounts for more than 400 executions per day, more than one million of total executions, and about 250,000 experiments stored in the archives. These figures show the large impact of the IPOL initiative.

The next milestone is to extend the demo platform to a system capable of running machine-learning applications, a brand new scenario with heterogeneous data and new phases (such as the learning step) that are not taken into account in the current architecture. And again, this will not be the work of a single person, but of a team. The details of my proposal are given in chapter 7.

The final goal is to cover as much as possible the state of the art in signal processing (mainly image and video) as well as machine learning algorithms. The reproducible research principles necessarily lead to further exchanges between research groups working on image processing, they promote the validation and comparison of the algorithms, the reuse of implementations, and the progressive compilation of a verifiable state of the art. And I am convinced that encouraging my future PhD candidates to follow this philosophy will help them to develop their scientific career on a solid and reliable basis.<sup>1</sup>

The second part of my research project involves the design and improvement of Earth-observation satellites. Indeed, our work with IASI-NG indicates that by means of image denoising it would be possible to reduce the size of the sensor in a factor over 7. This of course opens the discussion on the possibilities of making hybrid systems in the sense of designing them as physical/optical

---

<sup>1</sup>So far Tina Nikoukhah has written already several IPOL demos demonstrating her algorithms on detection of falsifications, as well as making the source code available to the scientific community. Several IPOL articles are in preparation at this moment.

systems and taking into account the image processing techniques that might influence the design of the instrument.

I explored the concept of hybrid systems with our work with IASI-NG, and in our current project on SMOS-HR it became clear that it was the right direction to follow. SMOS is an Earth-observation satellite based on interferometry with the goal of measuring the humidity on ground and the salinity of the oceans and SMOS-HR is its next version, with a greatly improved spatial resolution as the main objective. We are working at this moment on this problem by exploring the possibilities of irregular sampling. This implies a non-convex optimization that can be solved with some optimization algorithms which do not depend on the gradient (such as Particle Swarm Optimization), as we show. Again, our methodology brings us to hybrid systems that take into account both the physics and the possibilities of mathematical techniques at several steps of the processing chain. For the SMOS-HR project we are confident that our results will persuade CNES and collaborating agencies to take into account our approach for the design of the future SMOS-HR satellite.

Specifically, our research plan is focused on solving the following problems:

1. Explore and develop new mathematical techniques of Fourier irregular sampling for SMOS-HR. Demonstrate that the obtained configurations can be calibrated;
2. Obtain a model of the brightness temperature (BT) function according to the incidence angle. Explore two possibilities: a local model (from consecutive snapshots) and a global model (from clustered observations along a year on different Earth biomes and seasons);
3. Perform a joint-reconstruction of the visibilities which takes into account the low dimension of the BT according to the incidence angle. This redundancy can be exploited in the process that transforms visibilities into BTs;
4. Correct the spatial aliasing in the BT map using a consecutive multi-snapshot strategy;
5. Reduce the influence of radiometric noise by using the low-dimension model of the BT signal.

Max Dunitz is to start in October 2019 a PhD thesis supervised by Jean-Michel Morel and myself<sup>2</sup>. We have an ongoing contractual collaboration with the CESBIO team at CNES. The details of this ongoing research and the first results are given in chapter 4.

In particular, the research plan for the PhD of Max Dunitz is the following:

- **First year**

1. State of the art of the several concepts of interferometric telescopes and studies carried out for the optimization and fusion of the acquired data. The literature will be particularly analyzed around the existing SMOS satellite and telescopes (*VAtacama Large Millimeter Array - ALMA Radio Telescope* [135]), and the future SKA (*Square Kilometer Array* [136]).
2. Study of the literature on the irregular sampling of the Fourier transform and methods of extension or completion of the spectrum (in particular extensions of the total variation method [137]). The PhD student will confirm the experiments proving that an optimized irregular configuration of antennas is better compared to a regular configuration, and will explore ways to prove it mathematically.
3. Research on the best non-convex optimization algorithms to optimize an irregular configuration.
4. Online publications and demos on IPOL allowing real-time application, and also conventional conferences and journals of image processing and mathematics applied to imaging and spatial imagery (ICIP, CVPR, TIP, JMIV, SIIMS, JOSA, SPRS, IGARSS, Remote Sensing, ...)

---

<sup>2</sup>My request for the temporal derogation of the HDR has already been accepted by Université Paris-Saclay, thus allowing me to become Max Dunitz's main director.

- **Second year**

1. Mathematical analysis of the folded satellite in regular configuration: demonstration on simplified planar models that the unfolding is possible. Extension to a spherical acquisition model.
2. Study and optimization of separable irregular configurations, allowing to obtain a sufficient sampling in the direction orthogonal to the trace of the satellite.
3. Implementation of a general procedure for the calibration of antennas in irregular configuration (extension of [138, 52]).
4. Creation of a first simplified simulator showing a signal-to-noise ratio increase in the evaluation tables, and the proof of concept the proposed unfolding technique.
5. Parametric modeling of the angular BT and implementation of the general inversion algorithm.
6. Publications, as shown in point 1 of the first year.

- **Third year**

1. Extension of the theory and algorithms to calculate also the temperature of the sky (view at the edge of field) and the atmospheric transparency effects.
2. In collaboration with CESBIO, implement the complete simulator.
3. Refined modeling of the BT using the vast capital of the ten years of SMOS acquisitions. Use of this *a priori* to refine the inversion (we will consider the training of a deep neural network to this purpose).
4. Publications, as shown in point 1 of the first year.

**Expected impact and perspectives:** a proof of usefulness of mathematics does not hurt: an advanced mathematical analysis will allow us to design a new type of satellite which is more economical and more efficient. The developments on the irregular Fourier analysis [54] and our analysis of the inversion on Earth<sup>3</sup> may be applicable to any interferometric instrument.

The CMLA has currently partnerships with CNES and three companies including two start-up companies in the Paris region, MBDA, Nam'R and Kayrros on various problems and applications of remote sensing and 3D reconstruction on large-scale satellite coverings. Daily and future analyzes of the soil by SMOS and SMOS-HR address important societal and economic questions, notably the prediction of droughts, floods, and harvests. Moreover, it is not excluded that other models of Earth observation satellites in other bands emerge from the expertise acquired by CMLA. Kayrros in particular, has the ambition to design and launch their own satellites.

The third axis is inverse engineering of images. During my PhD I deeply studied the estimation of noise along the several steps of the camera processing chain (raw image acquisition, demosaicing, white balance, gamma correction, and JPEG compression) and it became clear to me that it could be possible to retrieve the history of transformations of an image from just one sample, given that all the steps of the processing chain leave detectable traces. The noise level curves (which depend both on the intensity and the frequency) allow that reverse engineering, as well as other techniques such as the study of periodicity or recovering the CFA mosaic pattern, among others. Indeed, the complete information of the noise was needed in our *Noise Clinic* project, where we managed to denoise images with unknown noise model, even after JPEG compression. Figure 11.1 illustrates the evolution of the noise level functions according to each of the step of the camera's processing chain.

Yet another application of the recovery of the history of an image is the detection of manipulations and falsifications.

The problem of image forgery detection concerns much broader lines of evidence than just tools from automatic image analysis. These include: the declared source of the image, the political, economic,

---

<sup>3</sup>Max Dunitz is finishing at this moment his masters' thesis on this subject, projecting coordinates given on the satellite's reference coordinate system on the surface of the (spherical) Earth.

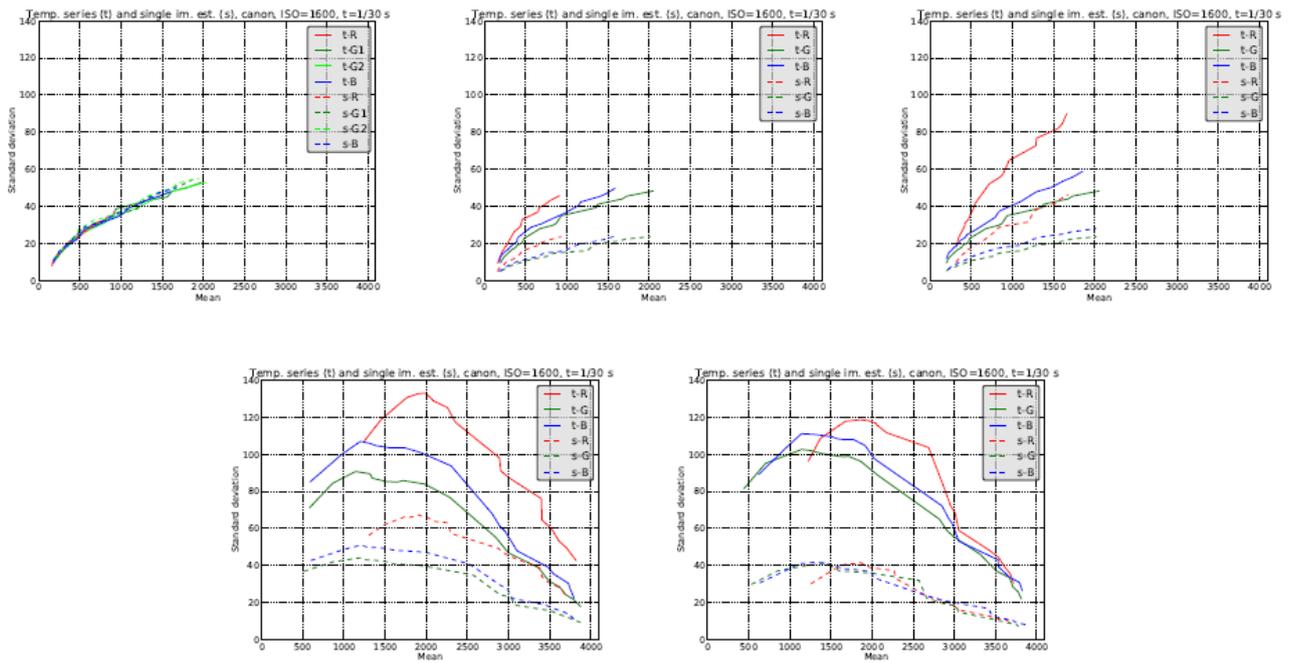


Figure 11.1: Illustration of the noise signature of a JPEG image at successive stages of its processing from a Canon raw image, ISO 1600,  $t = 1/30$ s. Successively: curves of the raw image, then after demosaicking, color balance, gamma correction, and JPEG compression. At the beginning the four color channels have the same noise curve (associating standard deviation of the noise at the signal level). After demosaicking, the curves of the color channels diverge. Gamma correction changes the shape of the noise curve. The final JPEG compression alters the noise ratios at different scales. All these signatures characterize the history of the image and allows an inverse engineering to detect alterations, or even reconstruct a falsely raw image to camouflage them.

legal conditions in which the image was produced, the supposition by the expert of malicious or intentional falsification, the information on the image source such as its EXIF or file headers' data of the device that produced it. Finally the detection requires a semantic analysis of the photograph and of the likelihood of the scene. The expert's eye is often crucial to detect inconsistencies in a scene. Hence the forensic analysis of an image cannot rely solely on automatic image processes. These are not able to translate the contextual cues that often make a forgery evident to an expert.

Our goal is therefore to develop algorithms applicable to any digital image given that these algorithms will produce by reverse engineering a complete history of the image. Adequate visualization tools will also reveal potential anomalies in the image processing chain and we will develop statistical tools associating with these anomalies error probabilities or false alarm number. This will enable to quantitative decisions, free of subjectivity.

The project has been divided into three phases, as it takes:

- Establish a digital history of the image, ie. the parameters of all operations that were performed on it from the raw image, in particular chromatic and optical aberration correction, denoising, demosaicking, gamma correction, color balancing, deblurring, interpolation, cropping, compression type, and compression parameters, (possibly) double compression.
- Categorize, implement, and take over the most promising image falsification algorithms for the addition or removal of local information (inpainting, Poisson editing, and others) and do the same for the anti-forensic operations.
- Finally develop automatic detection algorithms which, based on the alleged history of the image, detect any anomaly in this history. Furthermore algorithms detecting and analyzing internal image suspicious repetitions, produced by a copy-move tampering action.

Marina Gardella will start a PhD on detection of falsifications in images covering an area that does not overlap with the research of Tina Nikoukhah: the analysis of the noise level several zones of a eventually forged image. The number of detection techniques is so vast that it needs of a small coordinated team in order to succeed. Indeed, a reliable detector needs to find inconsistencies in the whole image processing chain, and these are some of the hints for research:

- Lighting inconsistencies;
- camera response function (CRF);
- JPEG blocking artifacts;
- estimated JPEG quantization matrices;
- copy-move attacks;
- forensics of chromatic aberration and its anti-forensics;
- noise inconsistencies;
- interpolation and re-sampling inconsistencies;
- detecting inconsistencies in encoders;
- detecting double compression;
- anti-forensic attacks on compression signatures;
- smart anti-forensic sampling;
- counter-antiforensic operations;
- mosaicing patterns;

- resampling traces in the Fourier domain;
- video forensics, anti-forensics, counter anti-forensics, and game theory for video frame. deletion detection.

There is much literature on the above subjects, and indeed we prepared an exhaustive bibliographic research to reveal the state of the art. We found methods based on the Photo Response Non-Uniformity (PRNU) [139, 140, 141] which are not a good match to our purposes (they need about 50 images taken with the original camera, which normally is not available), methods based on denoising of homoscedastic noise [142, 143, 144, 145] that cannot be applied after demosaicking or JPEG compression, methods where it is not well specified how to set the parameters in the general case or a decision threshold to perform automatic detections [146, 147, 148, 149, 150, 151]. In all the papers we reviewed we found that relevant discussion on applying these methods to real images was missing. Indeed, most of them deal with unrealistic simulated noise model, which are not compatible with real images that have undergone demosaicking and JPEG compression. In this project we will address these questions in order to develop algorithms that can be used with real images. Specifically we take into account the frequency dependence of colored noise and the detection at different scales. And, most importantly, the possibility of performing automatic detections blindly, without the need of human interpretation.

Marina Gardella is at this moment in a *pre-PhD* internship at CMLA before she starts officially her PhD supervised by Jean-Michel Morel, Pablo Musé, and myself. I shall now briefly discuss on very preliminary (but yet promising) results she obtained using a multi-scale approach with the Ponomarenko noise estimator [152].

Two different images were taken with digital Camera with different exposure time and ISO gain. We shall refer to this images as *Image 1* and *Image 2*. A part of Image 2 was copied over Image 1, creating a large falsification, very evident to the naked eye. The question is if it is possible to determine from the noise level functions such an evident difference. Note that the images have undergone the complete image processing chain of the camera up to JPEG compression. Figure 11.2 shows images 1 and 2, as well as the forgery. The size of both images is  $3072 \times 2304$  pixels. Image 1 was taken with a exposure time of 1.3 s and ISO 100. Image was taken with a exposure time of 0.17 s and ISO 208.

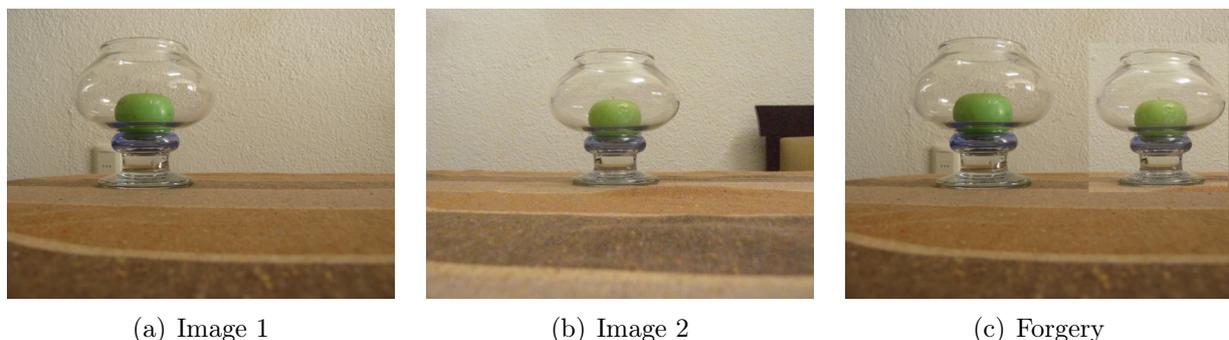


Figure 11.2: Original images 1 and 2, and the evident forgery. The size of both images is  $3072 \times 2304$  pixels. Image 1 was taken with a exposure time of 1.3 s and ISO 100. Image was taken with a exposure time of 0.17 s and ISO 208.

The variance of noise is related to the number of photons arriving to the camera's sensor<sup>4</sup>. Assuming that the images were *raw*<sup>5</sup> (which is not the case at all) and that the ISO is simply a multiplicative gain, one expects that Image 2 would be more noisy than Image 1 since the product of the exposure time by the ISO gain is 80 in Image 1 and 208 in Image 2. The initial Poisson-Gaussian noise model of a raw image states that the variance of the noise at each pixel is an affine function of its expectation. Yet this model is deeply altered by the image processing chain (demosaicking, color balance, gamma-correction, compression). Hence we can only make the following generic assumption:

<sup>4</sup>The photon emission of a body follows the Poisson distribution, where its variance is proportional to the mean intensity.

<sup>5</sup>Retrieved directly from the sensor's mosaic without passing through all the camera's processing chain.

**Postulate:** The noise variance at a pixel depends mainly on its expectation.

The algorithm that we are developing at this moment proceeds by considering overlapping blocks in the image and comparing the obtained noise level function with the function obtained from a global estimation in the eventually forged image. One expects that the falsification is not as important as to affect significantly the global estimation. In any case, a zone whose noise level is significantly smaller compared with the global estimation is a clear sign of falsification. On the other hand, if the noise level of a particular zone is higher, one needs to determine if that is caused by the presence of textures in the zone or if it caused by a falsification with an increased noise level.

In this simple example we show a falsification that consists on pasting a block with a majority of pixels coming from Image 2, which is more noisy. Figure 11.3 shows Blocks 1, 2, and 4 from the original Image 1 and Block 3 with a majority of pixels from Image 2 (fake source) and a minority from Image 1.

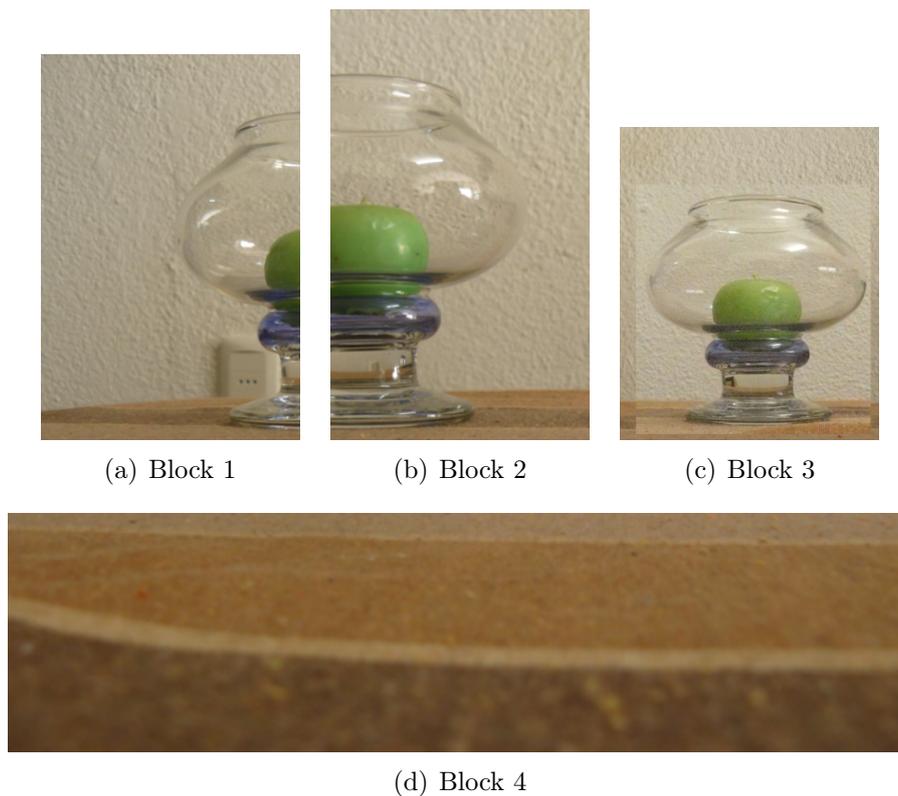


Figure 11.3: Blocks 1, 2, and 4 from the original Image 1 and Block 3 with a majority of pixels from Image 2 (fake source) and a minority from Image 1. Block 1 is  $977 \times 1465$ , Block 2 is  $877 \times 1469$ , Block 3 is  $1219 \times 1485$ , and Block 4 is  $2500 \times 667$  pixels.

In order to estimate the noise, we follow a multi-scale strategy, where  $S_0$  is the original-size image,  $S_1$  a zoom-out by averaging each block of  $4 \times 4$  pixels in  $S_0$ , and  $S_2$  is obtained by averaging  $S_1$  the same way. In general, the scale  $S_{i+1}$  is obtaining by averaging  $S_i$ . The effect of JPEG in the estimation is really minimal in  $S_2$  since the compression algorithm considers non-overlapping  $8 \times 8$  blocks.

Figure 11.4 shows the noise estimation at scale  $S_0$ . The noise level functions of the estimation in the original and forged blocks do not allow to tell that they come from different sources. This is expected, since the Ponomarenko estimator assumes that the noise is white, which is no longer the case after JPEG compression. Indeed, JPEG quantized much the high-frequency DCT coefficients thus removing most of the high-frequency noise.

Figure 11.5 shows the estimations at scale  $S_1$ . Some difference between the noise estimations according to the source can be noticed, especially in the red channel.

Finally figure 11.6 shows the estimations at scale  $S_2$ , where the influence of JPEG may be neglected. A simple observation of the curves allows to tell that indeed the source of Block is different to the

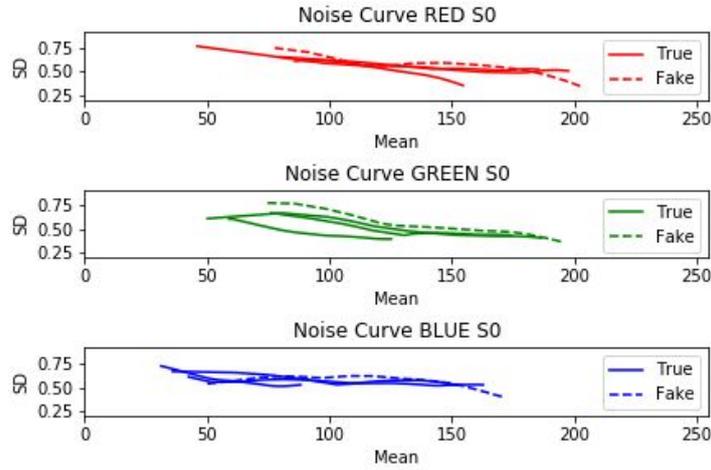


Figure 11.4: Noise estimation at scale  $S_0$ . The noise level functions of the estimation in the original and forged blocks do not allow to tell that they come from different sources. This is expected, since the Ponomarenko estimator assumes that the noise is white, which is no longer the case after JPEG compression. Indeed, JPEG quantized much of the high-frequency DCT coefficients thus removing most of the high-frequency noise.

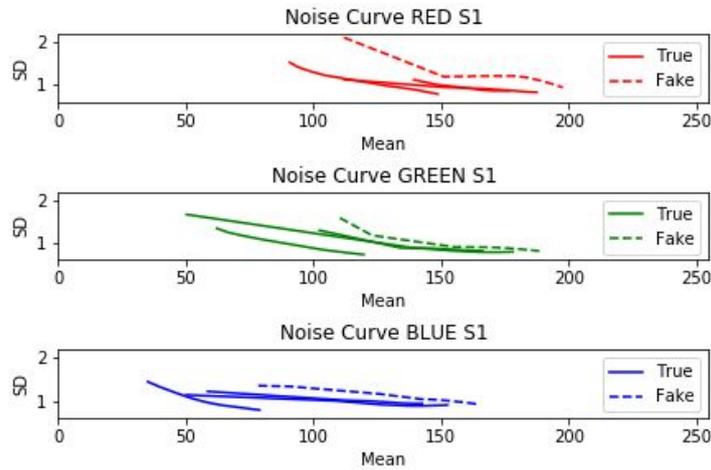


Figure 11.5: Noise estimations at scale  $S_1$ . Some difference between the noise estimations according to the source can be noticed, especially in the red channel.

other blocks 1, 2, and 4.

However, one of the goals is to perform automatic detections rather than letting a human observer subjectively decide. As introduced in chapters 9 and 10, we will apply the NFA theory to have fully objective detections. And moreover, to decide that a given image *does not* contain any falsification. This last statement makes tries to make clear the advantage of automatic detectors with respect to visual *heatmaps* that need to be interpreted by experts.

Although the preliminary results I have presented here are only a proof of concept, they give some hints about the research path we will follow.

The dependence of the noise variance on its expectation must be formulated at each scale, because the first two scales are deeply altered by demosaicking and JPEG compression. Because of that, all methods that we shall comment in the next share the following steps:

1. they estimate noise at each scale of the image;
2. they establish a correspondence between the value of the color channel and its noise variance;
3. they try to detect discrepancies of this correspondence, namely to find pairs of regions with similar color and different noise models.

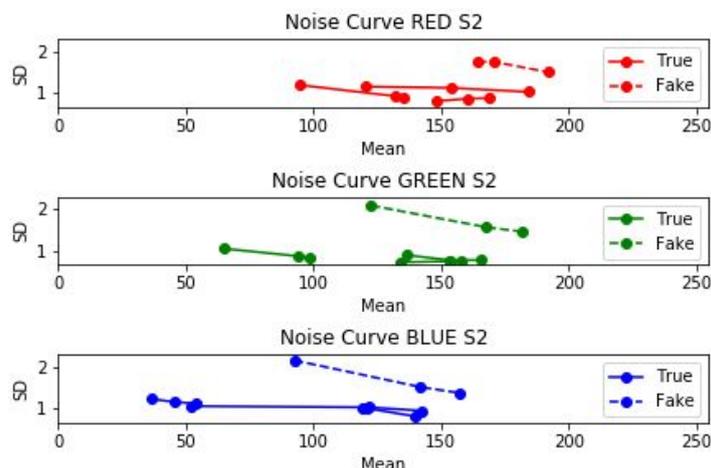


Figure 11.6: Noise estimations at scale  $S_2$ , where the influence of JPEG may be neglected. A simple observation of the curves allows to tell that indeed the source of Block is different to the other blocks 1, 2, and 4. However, our goal is to avoid any human interpretation and to have instead automatic detector based on the NFA.

Each step requires serious reflexion. We now analyze each of them.

**Noise estimation** First of all, many of the reviewed methods advocate for the application of a denoising algorithm to extract the noise. But this raises a hen and egg dilemma: to denoise we need an estimation of the noise model and to estimate noise, we need also to somehow single it out. It seems that we need to estimate the noise locally and by the most robust method. To that aim we propose to use the self-similarity concept according to which each image block contains at least another very similar block. Thus we propose to estimate noise at each pixel and for each channel by the simple following procedure:

- For each  $k \times k$  block centered at a pixel, find the closest block for the  $l^2$  norm within an  $m \times m$  neighborhood. Typically,  $k = 5$  or  $7$ , and  $m = k + 3$ .
- compute the average  $\mu$  of the block
- compute the variance  $\sigma$  of the noise block as half the variance of the difference of the block to its closest block

Clearly there will be blocks containing structure and it is recommended to discard them because their noise will be overestimated. One way to do so is simply to discard all blocks such that the ratio of their variance to the variance of their noise is too high. This should discard all blocks containing edges or texture.

**Establishing a correspondence between signal and noise** After the first step we dispose of a set of correspondences  $(\mu, \sigma)$ , for each of the three image channels. For each small interval of the channel's dynamic  $I_n = [n\tau, (n+1)\tau]$  ( $\tau = 2$  or  $4$  for example) we can therefore estimate a probability distribution for the set of variances  $\sigma$  such that  $\mu \in I_n$ .

**Establishing discrepancies** If for some  $\mu$  the distribution of the noise variance happens to be bimodal, this may indicate that the corresponding image blocks belong to two different groups, one from the original image and one from a part that has been spliced into the image. It is clear at this point that we must perform hypothesis testing to decide if the discrepancy is just casual or really meaningful:

-We can test if the set of variances obey a  $\chi^2$  as they should under a Gaussian assumption for the noise.

-We can test blocks with exceptional variance with respect to their interval, typically blocs with a variance more than  $\rho > 1$  times the average variance in their interval, or significantly less than  $\rho^{-1} < 1$

times the average variance in their interval.

-if many blocks that are grouped in a part of the image are discrepant blocks, this may indicate that they all belong to a spliced region.

It follows from the above considerations that we should visualize for each image all discrepant blocks, and then decide what final decision test should be performed. This is a research in progress and we hope that the techniques sketched here will lead to good detection scores at the DEFALS challenge. And following our reproducible research methodology, more importantly all the results will be published and the source code made available.

This concludes my research and pedagogical project for the following years, which definitively aims to help new researchers (PhD candidates) in the image/video and general signal processing fields to develop their own scientific career.

# Algorithmic descriptions

This appendix gives the pseudo-codes of the algorithms described in chapter 4.

---

**Algorithm 8** rotate: rotate a point around the origin

---

1:  
    **Input**  $(x, y)$ : coordinates of the point  
    **Input**  $\alpha$ : the rotation angle  
    **Output**  $(x_r, y_r)$ : the rotated coordinates  
2:  $x_r \leftarrow x \cos(\alpha) - y \sin(\alpha)$   
3:  $y_r \leftarrow x \sin(\alpha) + y \cos(\alpha)$   
4: **return**  $x_r, y_r$

---

---

**Algorithm 9** Build the quincunx configuration

---

1:  
    **Input** freqs: sampling frequencies  
    **Input**  $R$ :  $2R$  antennas per arm  
    **Input**  $d = \lambda$ : spacing of the antennas  
    **Output**  $P$ : set of antennas' positions  
2:  $P \leftarrow \emptyset$   
3:  $\beta \leftarrow 1/\sqrt{2}$   
4: **for**  $x \in [-R, \dots, R - 1]$  **do**  
5:      $x_r, y_r \leftarrow \text{rotate}(x, R, \pi/4)$  ▷ Algo. 8  
6:      $P \leftarrow (x_r d, y_r d)$   
7:      $x_r, y_r \leftarrow \text{rotate}(x, -R, \pi/4)$   
8:      $P \leftarrow (x_r d, y_r d)$   
9: **end for**  
10: **for**  $y \in [-R, \dots, R - 1]$  **do**  
11:      $x_r, y_r \leftarrow \text{rotate}(R, y, \pi/4)$   
12:      $P \leftarrow (x_r d, y_r d)$   
13:      $x_r, y_r \leftarrow \text{rotate}(-R, y, \pi/4)$   
14:      $y_r \leftarrow y_r - \beta$  ▷ Shift bar  
15:      $P \leftarrow (x_r d, y_r d)$   
16: **end for**  
17: **return**  $P$

---

---

**Algorithm 10** key2freq: convert a dictionary's key into a sampling frequency/baseline

---

1:  
    **Input**  $\odot$ : input dictionary key  
    **Output**  $\nu$ : frequency/baseline associated to the given key  
2:  $\gamma \leftarrow 1e - 6$   
3:  $\nu \leftarrow \gamma \times \odot$   
4: **return**  $\nu$

---

---

**Algorithm 11** freq2key: convert a frequency/baseline into a dictionary key

---

1:  
     **Input**  $\nu$ : frequency/baseline  
     **Output**  $\odot$ : dictionary's key of the given sampling frequency/baseline  
 2:  $\gamma \leftarrow 1e - 6$   
 3: **return**  $\lfloor \text{round}(\nu/\gamma) \rfloor$

---



---

**Algorithm 12** getBaselinesDict: given a list of antennas' positions, compute their baselines and multiplicities.

---

1:  
     **Input**  $\mathbf{a}$ : list of antennas' positions  
     **Output**  $\mathcal{D}_{\vec{u}}$ : a dictionary function that maps any of the baselines of the instruments with its multiplicity  
 2:  $\mathcal{D}_{\vec{u}} \leftarrow \emptyset$   
 3: **for**  $(x, y) \in \mathbf{a}$  **do**  
 4:      $\vec{u} \leftarrow x - y$   
 5:      $\odot \leftarrow \text{freq2key}(\vec{u})$   
 6:     **if**  $\odot \in \mathcal{D}_{\vec{u}}$  **then**  
 7:          $\mathcal{D}_{\vec{u}}[\odot] \leftarrow \mathcal{D}_{\vec{u}}[\odot] + 1$   
 8:     **else**  
 9:          $\mathcal{D}_{\vec{u}}[\odot] \leftarrow 1$   
 10:     **end if**  
 11: **end for**  
 12: **return**  $\mathcal{D}_{\vec{u}}$

---



---

**Algorithm 13** gridBaselines: create a sparse matrix which grids the given baselines

---

1:  
     **Input**  $\mathcal{D}_{\vec{u}}$ : Dictionary that maps all the baselines of the instrument with the corresponding multiplicities.  
     **Input**  $\delta = 1$ : gridding step  
     **Output**  $\mathbf{J}$ : matrix that contains the gridded baselines  
 2:  $\vec{u}_{\max} \leftarrow \text{argmax}_{\vec{u} \in \text{values}(\mathcal{D}_{\vec{u}})} \|\vec{u}\|_2$   $\triangleright$  Get baseline corresponding to the highest sampling frequency  
 3:  $\mathbf{J} \leftarrow \text{zeros}(\lceil 2\delta\vec{u}_{\max} + 1 \rceil)$   
 4:  $\mathbf{J}[\text{round}(\delta(\vec{u} + \vec{u}_{\max}))] \leftarrow \mathcal{D}_{\vec{u}}[\odot], \forall (\odot, \vec{u}) \in \mathcal{D}_{\vec{u}}$   $\triangleright$  Set multiplicities, with DC freq. centered  
 5: **return**  $\mathbf{J}$

---

---

**Algorithm 14** antennasInRectangle: given a list of longitudes along the perimeter of the rectangle and shift of each antenna in the lateral direction, this function computes the 2D coordinates of the antennas

---

```

1:
    Input longitudes: antenna's positions given as longitudes in the square, given in  $\lambda$  units
    Input lateralShifts: antenna's movement factor in the orthogonal bar direction, w.r.t. to  $\lambda$ 
    Input side: the side of the square
    Output a: list of antennas within the square structure
2:  $R \leftarrow \text{side} / 2$ 
3:  $\mathbf{a} \leftarrow \emptyset$ 
4:  $i \leftarrow 0$  ▷ Counter
5: for  $d \in \text{longitudes}$  do
6:    $x, y \leftarrow \text{longitudeToRectangle2D}(d, R)$ 
7:    $\mathbf{a} \leftarrow (x, y)$  ▷ Append
▷ Apply lateral movement
8:    $\text{side} \leftarrow \text{getSide}((x, y), R)$  ▷ Get side of the square to which the coordinate belongs. 0: right,
1: upper, 2: left, 3: bottom
9:   if  $\text{side} = 0$  or  $\text{side} = 2$  then
10:     $x \leftarrow x + \text{lateralShifts}[i]$ 
11:   else
12:     $y \leftarrow y + \text{lateralShifts}[i]$ 
13:   end if
14:    $\mathbf{a}[i] \leftarrow (x, y)$  ▷ Update position with the lateral shift
15:    $i \leftarrow i + 1$  ▷ Next antenna
16: end for
17: return  $\mathbf{a}$ 

```

---



---

**Algorithm 15** cost: the PSO optimization cost function

---

```

1:
    Input longitudes: antenna's positions given as longitudes in the square, given in  $\lambda$  units
    Input lateralShifts: antenna's movement factor in the orthogonal bar direction, w.r.t. to  $\lambda$ 
    Input K: kernel to compute the density by convolution
    Input  $\gamma_1, \gamma_2$ : hyperparameters
    Input side: the side of the square
    Output C: evaluated cost
2:  $\mathbf{a} \leftarrow \text{antennasInRectangle}(\text{longitudes}, \text{lateralShifts}, \text{side}/2)$ 
3:  $\mathcal{D}_{\bar{\mathbf{u}}} \leftarrow \text{getBaselinesDict}(\mathbf{a})$ 
4:  $\mathbf{J} \leftarrow \text{gridBaselines}(\mathcal{D}_{\bar{\mathbf{u}}})$  ▷ Grid the baselines
5:  $\mathbf{D} \leftarrow \mathbf{J} * \mathbf{K}$  ▷ Convolution
6:  $h \leftarrow |\mathbf{D} < \gamma_2 \bar{\mathbf{D}}| / |\mathbf{D}|$  ▷ Absence of holes measure.  $\bar{\mathbf{D}}$  is the mean of the values in  $\mathbf{D}$  and  $|\cdot|$  is the
cardinal operator.
7:  $C \leftarrow \text{var}(\mathbf{D}) + \gamma_1 h$  ▷ It penalizes holes in the spectrum and large global variance
8: return  $C$ 

```

---

---

**Algorithm 16** separateAntennas: separate antennas which are too close

---

```

1:
    Input  $\mathbf{a}$ : antenna's coordinates in  $\lambda$  units
    Output  $\hat{\mathbf{a}}$ : list of fused antennas

2:  $\hat{\mathbf{a}} \leftarrow \mathbf{a}$ 
3:  $\zeta \leftarrow \text{distanceMatrix}(\hat{\mathbf{a}})$  ▷ For example, function spatial.distance_matrix in SciPy
4:  $p, q \leftarrow \text{argmin}_{i \neq j} \zeta[i, j]$ 
5: while  $\zeta[p, q] < 1$  do ▷ The minimal distance is  $\lambda$ 
6:      $\hat{\mathbf{a}}[p] \leftarrow \hat{\mathbf{a}}[p] + \mathcal{N}(\mu = 0, \sigma = 0.03)$  ▷ Perturbate its position
7:      $\zeta \leftarrow \text{distanceMatrix}(\hat{\mathbf{a}})$ 
8:      $p, q \leftarrow \text{argmin}_{i \neq j} \zeta[i, j]$ 
9: end while
10: return  $\hat{\mathbf{a}}$ 

```

---



---

**Algorithm 17** isPositiveFreq: determine if the given frequency is “positive”.

---

```

1:
    Input  $\nu$ : sampling frequency
    Output: Boolean indicating if the given frequency is “positive”

2: return  $\nu^u \geq 0$  ▷ Check  $u$  component of the frequency

```

---



---

**Algorithm 18** absFreq: get the corresponding “positive” frequency

---

```

1:
    Input  $\nu$ : sampling frequency
    Output: the corresponding “positive” frequency

2: if isPositiveFreq( $\nu$ ) then
3:     return  $\nu$ 
4: else
5:      $p, q \leftarrow \nu$ 
6:     return  $(-p, -q)$ 
7: end if

```

---



---

**Algorithm 19** getBaselineDiscr: thin-grid the given baseline from the position of two antennas

---

```

1:
    Input  $\mathbf{a}$ : list of antennas' positions
    Input  $i$ : index of the 1st antenna
    Input  $j$ : index of the 2nd antenna
    Output: thin-gridded baseline

2:  $\gamma \leftarrow 1e - 6$ 
3: return  $\lfloor (\mathbf{a}_i - \mathbf{a}_j) / \gamma \rfloor \gamma$ 

```

---

---

**Algorithm 20** getPositiveBaselinesDict: given a list of antennas' positions, compute their "positive" baselines only and their multiplicities.

---

```

1:
   Input  $\mathbf{a}$ : list of antennas' positions
   Output  $\mathcal{D}_{\vec{u}}$ : a dictionary function that maps any of the baselines of the instruments with its
   multiplicity. Only "positive" baselines
2:  $\mathcal{D}_{\vec{u}} \leftarrow \emptyset$ 
3: for  $(x, y) \in \mathbf{a}$  do
4:    $\vec{u} \leftarrow \text{absFreq}(x - y)$ 
5:    $\odot \leftarrow \text{freq2key}(\vec{u})$ 
6:   if  $\odot \in \mathcal{D}_{\vec{u}}$  then
7:      $\mathcal{D}_{\vec{u}}[\odot] \leftarrow x, y, \mathcal{D}_{\vec{u}}[\odot] + 1$  ▷ Associate first antenna
8:   else
9:      $k, l, m \leftarrow \mathcal{D}_{\vec{u}}[\odot]$ 
10:     $\mathcal{D}_{\vec{u}}[\odot] \leftarrow k, l, m + 1$  ▷ It already exists: increments its multiplicity count
11:   end if
12: end for
13: return  $\mathcal{D}_{\vec{u}}$ 

```

---

**Algorithm 21** getRank: obtain the rank of the equations system.

---

```

1:
   Input  $\mathbf{a}$ : list of antennas' positions,  $\mathcal{N}$  its length.
   Output  $|f| + |\phi|$ : number of unknowns in the system
   Output  $\text{rank}(M)$ : rank of the system of equations
2:  $\Psi \leftarrow \text{zeros}(\cdot, \cdot)$ 
3:  $r \leftarrow 0$ 
4:  $f \leftarrow$  list of antennas' unknowns initialized to zero
5:  $\phi \leftarrow$  list of cross-antennas' unknowns initialized to zero
6:  $\mathcal{D}_{\vec{u}}^+ \leftarrow \text{getPositiveBaselinesDict}(\mathbf{a})$ 
7: for  $i \in [0, \dots, \mathcal{N} - 1]$  do
8:   for  $j \in [i + 1, \dots, \mathcal{N} - 1]$  do
9:      $\vec{u} \leftarrow \text{getBaselineDiscr}(\mathbf{a}, i, j)$  ▷ Obtain its positive baseline
10:    if  $\text{isPositiveFreq}(\vec{u})$  then
11:       $s \leftarrow +1$ 
12:    else
13:       $\vec{u} \leftarrow \text{absFreq}(\vec{u})$ 
14:       $s \leftarrow -1$ 
15:    end if
▷ Obtain a unique baseline antenna's pair and its multiplicity
16:     $p_i, p_j, m \leftarrow \mathcal{D}_{\vec{u}}^+[\vec{u}]$ 
17:    if  $m > 1$  then ▷ Consider only visibilities with multiplicity over 1
18:      if  $p_i > 0$  then
19:         $\Psi[r, i - 1] \leftarrow +1$ 
20:      end if
21:      if  $p_j > 0$  then
22:         $\Psi[r, j - 1] \leftarrow -1$ 
23:      end if
24:       $k \leftarrow |f| + \phi.\text{index}((p_i, p_j))$  ▷ Column index of the unknown in  $\Psi$ 
25:       $\Psi[r, k] \leftarrow s$  ▷ Set sign
26:       $r \leftarrow r + 1$  ▷ Next row
27:    end if
28:   end for
29: end for
30: return  $|f| + |\phi|, \text{rank}(M)$ 

```

---

---

**Algorithm 22** discretizeAntennaPosition: get the nearest gridded spatial position of the given antenna

---

1:  
     **Input**  $\mathbf{a}$ : the spatial location of the antennas  
     **Input**  $\mu$ : gridding factor  
     **Output**: gridded antenna positions  
 2: **return**  $[\lfloor a/\mu \rfloor \mu \forall a \in \mathbf{a}]$

---



---

**Algorithm 23** getCalibrable: discretize the position of the antennas until the configuration gets calibrable

---

1:  
     **Input**  $\mathbf{a}$ : list of antennas' positions  
     **Input**  $\mu_0$ : initial gridding factor  
     **Input**  $\Delta\mu$ : increment for the gridding factor  
     **Output**  $\mathbf{a}_\mu$ : calibrable configuration  
 2:  $\mu \leftarrow \mu_0 - \Delta\mu$   
 3: calibrable  $\leftarrow$  **False**  
 4: **while not** calibrable **do**  
 5:      $\mu \leftarrow \mu + \Delta\mu$   
 6:      $\mathbf{a}_\mu \leftarrow$  discretizeAntennaPositions( $\mathbf{a}, \mu$ )  
 7:     numUnknowns, rank  $\leftarrow$  getRank( $\mathbf{a}_\mu$ )  
 8:     calibrable  $\leftarrow$  (numUnknowns  $\leq$  rank)  
 9: **end while**  
 10: **return**  $\mathbf{a}_\mu$

---



---

**Algorithm 24** Fill in the irregular Fourier matrix

---

1:  
     **Input**  $\{(\mathbf{x}, \mathbf{y})\}$ : coordinates corresponding to the discretized director cosines inside the FOV  
     **Input**  $\nu$ : sampling frequencies  
     **Input**  $N_x, N_y$ : width and height in pixels of the reconstruction  
     **Output**  $\mathbf{G}$ : irregular Fourier matrix  
 2:  $\mathbf{G} \leftarrow$  zeros( $|\nu|, |\{(\mathbf{x}, \mathbf{y})\}|$ )  
 3: **for**  $(\nu^u, \nu^v) \in \nu$  **do** ▷ Iterate through all sampling freqs.  
 4:      $k \leftarrow 0$   
 5:     **for**  $(x, y) \in \{(\mathbf{x}, \mathbf{y})\}$  **do** ▷ Iterate through all discretized spatial positions.  
 6:          $\xi \leftarrow 2(x - N_x/2)/N_x$   
 7:          $\eta \leftarrow 2(y - N_y/2)/N_y$   
 8:          $\mathbf{G}[k, yN_x + x] \leftarrow -2i\pi\xi\nu^u - 2i\pi\eta\nu^v$   
 9:     **end for**  
 10:      $k \leftarrow k + 1$   
 11: **end for**  
 12:  $\mathbf{G} \leftarrow \exp(\mathbf{G})$   
 13: **return**  $\mathbf{G}$

---

# Bibliography

- [1] Albert Einstein. On a stationary system with spherical symmetry consisting of many gravitating masses. *Annals of Mathematics*, pages 922–936, 1939.
- [2] Karl Schwarzschild. Über das gravitationsfeld eines massenpunktes nach der einsteinschen theorie. *Sitzungsberichte der Königlich Preussischen Akademie der Wissenschaften (Berlin)*, 1916, Seite 189-196, 1916.
- [3] A. El-Makadema, N. Razavi-Ghods, and A. Brown. Scanning performance of SKA-low sparse array configurations incorporating realistic element patterns and sky noise contributions. In *2012 International Conference on Electromagnetics in Advanced Applications*, pages 844–847, Sep. 2012.
- [4] Miguel Colom and Jean-Michel Morel. Full-spectrum denoising of high-SNR hyperspectral images. *J. Opt. Soc. Am. A*, 36(3):450–463, Mar 2019.
- [5] Rebecca N. Greenberger, John F. Mustard, Bethany L. Ehlmann, Diana L. Blaney, Edward A. Cloutis, Janette H. Wilson, Robert O. Green, and Abigail A. Fraeman. Imaging spectroscopy of geological samples and outcrops: Novel insights from microns to meters. *GSA Today*, 25(12):4–10, 2015.
- [6] C. Crevoisier, C. Clerbaux, V. Guidard, T. Phulpin, R. Armante, B. Barret, Camy-Peyret C., J. P. Chaboureaud, P. F. Coheur, L. Crépeau, et al. Towards IASI-new generation (IASI-NG): impact of improved spectral resolution and radiometric noise on the retrieval of thermodynamic, chemistry and climate variables. *Atmospheric Measurement Techniques*, 7(12):4367–4385, 2014.
- [7] Brian S. Sorg, Benjamin J. Moeller, Owen Donovan, Yiting Cao, and Mark W. Dewhirst. Hyperspectral imaging of hemoglobin saturation in tumor microvasculature and tumor hypoxia development. *Journal of biomedical optics*, 10(4):044004–044004, 2005.
- [8] Renfu Lu and Yud-Ren Chen. Hyperspectral imaging for safety inspection of food and agricultural products. In *Photonics East (ISAM, VVDC, IEMB)*, pages 121–133. International Society for Optics and Photonics, 1999.
- [9] Nirmal Keshava. A survey of spectral unmixing algorithms. *Lincoln Laboratory Journal*, 14(1):55–78, 2003.
- [10] Miguel Colom, Gwendoline Blanchet, Andrzej Klonecki, Olivier Lezeaux, Eric Pequignot, Florian Poustomis, Carole Thiebaut, Sylvain Ythier, and Jean-Michel Morel. BBD: a new Bayesian bi-clustering denoising algorithm for IASI-NG hyperspectral images. In *8th workshop on Hyperspectral Image and Signal Processing : Evolution in Remote Sensing*, pages –, Los Angeles, USA, 08/2016 2016. IEEE.
- [11] Antonio Plaza, Jon Atli Benediktsson, Joseph W. Boardman, Jason Brazile, Lorenzo Bruzzone, Gustavo Camps-Valls, Jocelyn Chanussot, Mathieu Fauvel, Paolo Gamba, Anthony Gualtieri, et al. Recent advances in techniques for hyperspectral image processing. *Remote sensing of environment*, 113:S110–S122, 2009.

- [12] Antonio Plaza, Jon Atli Benediktsson, Joseph W. Boardman, John Brazile, Lorenzo Bruzzone, Gustavo Camps-Valls, Jocelyn Chanussot, Mathieu Fauvel, P.E. Gamba, A. Gaultieri, et al. Advanced processing of hyperspectral images. In *2006 IEEE International Symposium on Geoscience and Remote Sensing*, pages 1974–1978. IEEE, 2006.
- [13] Yong-Qiang Zhao and Jingxiang Yang. Hyperspectral image denoising via sparse representation and low-rank constraint. *Geoscience and Remote Sensing, IEEE Transactions on*, 53(1):296–308, 2015.
- [14] Yongqiang Zhao and Jinxiang Yang. Hyperspectral image denoising via sparsity and low rank. In *Geoscience and Remote Sensing Symposium (IGARSS), 2013 IEEE International*, pages 1091–1094. IEEE, 2013.
- [15] D. Kostadin, F. Alessandro, and E. Karen. Video denoising by sparse 3D transform-domain collaborative filtering. In *European signal processing conference*, volume 149. Tampere, Finland, 2007.
- [16] Behnood Rasti, Johannes R. Sveinsson, Magnus O. Ulfarsson, and Jon Atli Benediktsson. Hyperspectral image denoising using 3D wavelets. In *Geoscience and Remote Sensing Symposium (IGARSS), 2012 IEEE International*, pages 1349–1352. IEEE, 2012.
- [17] Hemant Kumar Aggarwal and Angshul Majumdar. Mixed gaussian and impulse denoising of hyperspectral images. In *Geoscience and Remote Sensing Symposium (IGARSS), 2015 IEEE International*, pages 429–432. IEEE, 2015.
- [18] Hemant Kumar Aggarwal and Angshul Majumdar. Hyperspectral image denoising using spatio-spectral total variation. *IEEE Geoscience and Remote Sensing Letters*, 13(3):442–446, 2016.
- [19] Ying Fu, Antony Lam, Imari Sato, and Yoichi Sato. Adaptive spatial-spectral dictionary learning for hyperspectral image denoising. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 343–351, 2015.
- [20] Minchao Ye, Yuntao Qian, and Qi Wang. Mt-omp for hyperspectral imagery denoising with model parameter estimation. In *Geoscience and Remote Sensing Symposium (IGARSS), 2013 IEEE International*, pages 1079–1082. IEEE, 2013.
- [21] Yi Peng, Deyu Meng, Zongben Xu, Chenqiang Gao, Yi Yang, and Biao Zhang. Decomposable nonlocal tensor dictionary learning for multispectral image denoising. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2949–2956, 2014.
- [22] Azam Karami, Rob Heylen, and Paul Scheunders. Band-specific shearlet-based hyperspectral image noise reduction. *IEEE Transactions on Geoscience and Remote Sensing*, 53(9):5054–5066, 2015.
- [23] Azam Karami, Rob Heylen, and Paul Scheunders. Denoising of hyperspectral images using shearlet transform and fully constrained least squares unmixing. In *8th workshop on Hyperspectral Image and Signal Processing : Evolution in Remote Sensing*, Los Angeles, USA, 08/2016 2016.
- [24] Guangyi Chen and Shen-En Qian. Denoising of hyperspectral imagery using principal component analysis and wavelet shrinkage. *Geoscience and Remote Sensing, IEEE Transactions on*, 49(3):973–980, 2011.
- [25] M. Planck. Ueber das gesetz der energieverteilung im normalspectrum. *Annalen der Physik*, 309(3):553–563, 1901.
- [26] Filipe Aires, Alain Chedin, Noelle A. Scott, and William B. Rossow. A regularized neural net approach for retrieval of atmospheric and surface temperatures with the IASI instrument. *Journal of Applied Meteorology*, 41(2):144–159, 2002.

- [27] F. Cayla, B. Tournier, and P. Hebert. Performance budgets of IASI options. Technical report, Tech. Rep. IA-TN-0B-5476-CNE, 21 pp. [Available at Centre National d'Études Spatiales, 2 place Maurice Quentin, 75039 Paris Cedex 01, France], 1995.
- [28] M. Jiménez, R. Díaz-Delgado, P. Vaughana, A. De Santis, A. Fernández-Renau, E. Pradao, and O. de la Cámara Gutiérrez. Airborne hyperspectral scanner (AHS) mapping capacity simulation for the Doñana biological reserve scrublands. In *Proceedings of the 10th International Symposium on Physical Measurements and Signatures in Remote Sensing, Davos, Switzerland*, pages 12–14, 2007.
- [29] Changyong Cao and Andrew K. Heidinger. Inter-comparison of the longwave infrared channels of MODIS and AVHRR/NOAA-16 using simultaneous nadir observations at orbit intersections. In *Earth Observing Systems VII*, volume 4814, pages 306–317. International Society for Optics and Photonics, 2002.
- [30] Changyong Cao, Michael Weinreb, and Jerry Sullivan. Solar contamination effects on the infrared channels of the advanced very high resolution radiometer (AVHRR). *Journal of Geophysical Research: Atmospheres*, 106(D24):33463–33469, 2001.
- [31] Stephen B. Campana, Joseph S. Accetta, and David L. Shumaker. *The infrared and electro-optical systems handbook (volume 5, Passive electro-optical systems)*. Infrared Information Analysis Center, 1993.
- [32] J.P. Rice, S. W. Brown, B.C. Johnson, and J.E. Neira. Hyperspectral image projectors for radiometric applications. *Metrologia*, 43(2):S61, 2006.
- [33] A. Foi. Clipped noisy images: Heteroskedastic modeling and practical denoising. *Signal Processing*, 89(12):2609–2629, 2009.
- [34] Masoud Farzam, Soosan Beheshti, and Masoud Hashemi. Adaptive noise variance estimation and intrinsic order selection for low SNR hyperspectral signals. *Electrical and Computer Engineering, Canadian Journal of*, 36(1):4–10, 2013.
- [35] Giuseppe Grieco, Carmine Serio, and Guido Masiello.  $\sigma$ -IASI- $\beta$ : a hyperfast radiative transfer code to retrieve surface and atmospheric geophysical parameters. *EARSeL eProceedings*, 12(2):149, 2013.
- [36] Jong-Sen Lee. Digital image smoothing and the sigma filter. *Computer vision, graphics, and image processing*, 24(2):255–269, 1983.
- [37] M. Lebrun, M. Colom, A. Buades, and J. M. Morel. Secrets of image denoising cuisine. *Acta Numerica*, 21(1):475–576, 2012.
- [38] A. D. Collard, A. P. McNally, F. I. Hilton, S. B. Healy, and N. C. Atkinson. The use of principal component analysis for the assimilation of high-resolution infrared sounder observations for numerical weather prediction. *Quarterly Journal of the Royal Meteorological Society*, 136(653):2038–2050, 2010.
- [39] Antony Lam, Imari Sato, and Yuuki Sato. Denoising hyperspectral images using spectral domain statistics. In *Pattern Recognition (ICPR), 2012 21st International Conference on*, pages 477–480. IEEE, 2012.
- [40] Marc Lebrun, Antoni Buades, and Jean-Michel Morel. A nonlocal Bayesian image denoising algorithm. *SIAM Journal on Imaging Sciences*, 6(3):1665–1688, 2013.
- [41] Stanislav Pyatykh, Jürgen Hesser, and Lei Zheng. Image noise level estimation by principal component analysis. *IEEE transactions on image processing*, 22(2):687–699, 2013.
- [42] Jong-Sen Lee. Refined filtering of image noise using local statistics. *Computer graphics and image processing*, 15(4):380–389, 1981.

- [43] F. Bernard, B. Calvel, F. Pasternak, R. Davancens, C. Buil, E. Baldit, C. Luitot, and A. Penquer. Overview of IASI-NG the new generation of infrared atmospheric sounder. In *International Conference on Space Optics*, volume 7, page 10, 2014.
- [44] Cyril Crevoisier, Cathy Clerbaux, Vincent Guidard, Eric Pequignot, and Frédérick Pasternak. IASI-NG: un concentré d’innovations technologiques pour l’étude de l’atmosphère terrestre. *La Météorologie*, 2014(86):3–5, 2014.
- [45] Marc Lebrun. An Analysis and Implementation of the BM3D Image Denoising Method. *Image Processing On Line*, 2:175–213, 2012.
- [46] Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.
- [47] EUMETSAT. IASI level 1: Product guide. <http://www.eumetsat.int/website/home/Data/Products/Level1Data/index.html>, 2014. [Online; accessed 30 May 2016].
- [48] F. Aires, W. B. Rossow, N. A. Scott, and A. Chédin. Remote sensing from the infrared atmospheric sounding interferometer instrument 2. simultaneous retrieval of temperature, water vapor, and ozone atmospheric profiles. *Journal of Geophysical Research: Atmospheres*, 107(D22), 2002.
- [49] K. Cawse-Nicholson, A. Robin, and M. Sears. The effect of spectrally correlated noise on noise estimation methods for hyperspectral images. In *Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS), 2012 4th Workshop on*, pages 1–4. IEEE, 2012.
- [50] Francis J Anscombe. The transformation of poisson, binomial and negative-binomial data. *Biometrika*, 35(3/4):246–254, 1948.
- [51] Markku Makitalo and Alessandro Foi. Optimal inversion of the anscombe transformation in low-count poisson image denoising. *IEEE transactions on Image Processing*, 20(1):99–109, 2011.
- [52] Yann H. Kerr, Philippe Waldteufel, J-P. Wigneron, JAMJ Martinuzzi, Jordi Font, and Michael Berger. Soil moisture retrieval from space: The soil moisture and ocean salinity (SMOS) mission. *IEEE transactions on Geoscience and remote sensing*, 39(8):1729–1735, 2001.
- [53] Boone, F. Interferometric array design: Optimizing the locations of the antenna pads. *Astronomy & Astrophysics*, 377(1):368–376, 2001.
- [54] Boone, F. Interferometric array design: Distributions of Fourier samples for imaging. *Astronomy & Astrophysics*, 386(3):1160–1171, 2002.
- [55] A. Camps, F. Torres, P. Lopez-Dekker, and S. J. Frasier. Redundant space calibration of hexagonal and Y-shaped beamforming radars and interferometric radiometers. *International journal of remote sensing*, 24(24):5183–5196, 2003.
- [56] A. Richard Thompson, James M. Moran, and George W. Swenson. *Van Cittert–Zernike Theorem, Spatial Coherence, and Scattering*, pages 767–786. Springer International Publishing, Cham, 2017.
- [57] Swenson G.W. Thompson A.R., Moran J.M. *Interferometry and Synthesis in Radio Astronomy*. Springer, Cham, 2017.
- [58] Adriano Camps, Mercè Vall-llossera, Ignasi Corbella, Nuria Ubeda, and Francesc Torres. Improved image reconstruction algorithms for aperture synthesis radiometers. *Geoscience and Remote Sensing, IEEE Transactions on*, 46:146 – 158, 02 2008.
- [59] Ignasi Corbella, Francesc Torres, Nuria Duffo, Israel Duran, Veronica Gonzalez-Gambau, and Manuel Martin-Neira. Wide field of view microwave interferometric radiometer imaging. *Remote Sensing*, 11(6):682, 2019.

- [60] Bruno Picard. *Téledétection de la surface terrestre par un radiomètre imageur à synthèse d'ouverture : principes de mesure, traitement des données interférométriques et méthodes de reconstruction régularisées*. PhD thesis, Université de Versailles - St Quentin, 2004.
- [61] Ali Khazâal. *Reconstruction d'image pour la mission spatiale SMOS*. PhD thesis, Université Toulouse III - Paul Sabatier, 2009.
- [62] J. Kennedy and R. Eberhart. Particle swarm optimization. In *Proceedings of ICNN'95 - International Conference on Neural Networks*, volume 4, pages 1942–1948 vol.4, Nov 1995.
- [63] Michael A. Brown, Francesc Torres, Ignasi Corbella, and Andreas Colliander. SMOS calibration. *IEEE Transactions on Geoscience and Remote Sensing*, 46(3):646–658, 2008.
- [64] E. Anterrieu. A resolving matrix approach for synthetic aperture imaging radiometers. *IEEE Transactions on Geoscience and Remote Sensing*, 42(8):1649–1656, Aug 2004.
- [65] Roger Penrose. A generalized inverse for matrices. In *Mathematical proceedings of the Cambridge philosophical society*, volume 51, pages 406–413. Cambridge University Press, 1955.
- [66] David L. Donoho, Arian Maleki, Inam Ur Rahman, Morteza Shahram, and Victoria Stodden. Reproducible research in computational harmonic analysis. *Computing in Science & Engineering*, 11(1):8–18, 2009.
- [67] C. Glenn Begley and Lee M. Ellis. Raise standards for preclinical cancer research. *Nature*, 483:531 EP –, Mar 2012. Comment.
- [68] Nicolas Limare. *Reproducible research, software quality, online interfaces and publishing for image processing*. PhD thesis, École Normale Supérieure de Cachan - ENS Cachan, 2012.
- [69] Arevalo M., Escobar C., Monasse P., Monzon N., and Colom M. The IPOL demo system: A scalable architecture of microservices for reproducible research. *Reproducible Research in Pattern Recognition*, 10214, 2016.
- [70] Miguel Colom, Bertrand Kerautret, Nicolas Limare, Pascal Monasse, and Jean-Michel Morel. IPOL: a new journal for fully reproducible research; analysis of four years development. In *2015 7th International Conference on New Technologies, Mobility and Security (NTMS)*, pages 1–5. IEEE, 2015.
- [71] Sam Neuman. *Building Microservices: Designing Fine-Grained Systems*. O'Reilly Media, 2015.
- [72] Jeremy Goecks, Anton Nekrutenko, and James Taylor. Galaxy: a comprehensive approach for supporting accessible, reproducible, and transparent computational research in the life sciences. *Genome biology*, 11(8):1, 2010.
- [73] Bart Lamiroy and Daniel Lopresti. The DAE platform: a framework for reproducible research in document image analysis. In *Proceedings of the first workshop on Reproducible Research in Pattern Recognition*, volume 10214 of LNCS. Springer, 2017.
- [74] Jonathan B. Buckheit and David L. Donoho. *Wavelab and reproducible research*. Springer, 1995.
- [75] Tim Berners-Lee, Roy Fielding, and Henrik Frystyk. Hypertext transfer protocol–HTTP/1.0. RFC 1945, RFC Editor, 1996.
- [76] Victoria Stodden. The legal framework for reproducible scientific research: Licensing and copyright. *Computing in Science & Engineering*, 11(1):35–40, 2009.
- [77] Victoria Stodden. Enabling reproducible research: Open licensing for scientific innovation. *International Journal of Communications Law and Policy*, Forthcoming, 2009.
- [78] Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides. *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison–Wesley, 2008.

- [79] Martin Fowler, Kent Beck, J. Brant, William Opdyke, and Don Roberts. *Refactoring: Improving the design of existing programs*. Addison-Wesley Reading, 1999.
- [80] Colom M. Extending IPOL to new data types and machine-learning applications. *Reproducible Research in Pattern Recognition*, 11455, 2019.
- [81] D. Purves, G. J. Augustine, and D. Fitzpatrick. *Neural Control of Saccadic Eye Movements*. Sunderland (MA): Sinauer Associates, 2001.
- [82] Thomas Neubauer and Johannes Heurix. A methodology for the pseudonymization of medical data. *International journal of medical informatics*, 80(3):190–204, 2011.
- [83] Antonio Foncubierta Rodríguez and Henning Müller. Ground truth generation in medical imaging: a crowdsourcing-based iterative approach. In *Proceedings of the ACM multimedia 2012 workshop on Crowdsourcing for multimedia*, pages 9–14. ACM, 2012.
- [84] M. Colom and B. Kerautret. An overview of platforms for reproducible research and new ways of publications. In *Reproducible Research in Pattern Recognition*, pages 25–39. Springer, 2018.
- [85] Tina Nikoukhah, Rafael Grompone von Gioi, Miguel Colom, and Jean-Michel Morel. Automatic JPEG grid detection with controlled false alarms, and its image forensic applications. In *2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, pages 378–383. IEEE, 2018.
- [86] H. Farid. Digital doctoring: how to tell the real from fake. *Significance*, 3(4):162–166, 2006.
- [87] Zhipeng Chen, Yao Zhao, and Rongrong Ni. Detection of operation chain: JPEG-resampling-JPEG. *Signal Processing: Image Communication*, 57(Supplement C):8 – 20, 2017.
- [88] Gajanan K. Birajdar and Vijay H. Mankar. Digital image forgery detection using passive techniques: A survey. *Digital Investigation*, 10(3):226 – 245, 2013.
- [89] Abhishek Kashyap, Rajesh Singh Parmar, Megha Agarwal, and Hariom Gupta. An evaluation of digital image forgery detection approaches. *CoRR*, abs/1703.09968, 2017.
- [90] Alessandro Piva. An overview on image forensics. *ISRN Signal Processing*, 2013(496701), 2013.
- [91] Gregory K. Wallace. The JPEG still picture compression standard. *IEEE Transactions on Consumer Electronics*, 1991.
- [92] Cheng-Shian Lin and Jyh-Jong Tsay. Passive forgery detection for JPEG compressed image based on block size estimation and consistency analysis. *Applied Mathematics and Information Sciences*, 9(2):1015–1028, 2015.
- [93] Thanh Hai Thai, Rémi Cogranne, Florent Retraint, and Thi-Ngoc-Canh Doan. JPEG quantization step estimation and its applications to digital image forensics. *IEEE Trans. Inf. Forensics Security*, 12(1), Jan. 2017.
- [94] Yanjun Cao, Tiegang Gao, Li Fan, and Qiting Yang. A robust detection algorithm for copy-move forgery in digital images. *Forensic Science International*, 214:33–43, 2012.
- [95] Zhouchen Lin, Jungeng He, Xiaoou Tang, and Chi-Keung Tang. Fast, automatic and fine-grained tampered JPEG image detection via DCT coefficient analysis. *Pattern Recognition*, 42:2492–2501, 2009.
- [96] W. Luo, J. Huang, and G. Qiu. JPEG error analysis and its applications to digital image forensics. *IEEE Transactions on Information Forensics and Security*, 5(3):480–491, Sept 2010.
- [97] S.M.Ye, Q.B. Sun, and E.C. Chang. Detecting digital image forgeries by measuring inconsistencies of blocking artifact. *IEEE International Conference on Multimedia and Expo, Beijing, China*, pages 12 – 15, 2007.

- [98] A. R. Bruna, G. Messina, and S. Battiato. Crop detection through blocking artefacts analysis. In Giuseppe Maino and Gian Luca Foresti, editors, *Image Analysis and Processing – ICIAP 2011*, pages 650–659, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.
- [99] M. Tkalcic and J. F. Tasic. Colour spaces: perceptual, historical and applicational background. In *The IEEE Region 8 EUROCON 2003. Computer as a Tool.*, volume 1, pages 304–308 vol.1, Sept 2003.
- [100] Zhigang Fan and Ricardo L. de Queiroz. Maximum likelihood estimation of JPEG quantization table in the identification of bitmap compression history. *ICIP*, pages 948–951, 2000.
- [101] Zhigang Fan and Ricardo L. de Queiroz. Identification of bitmap compression history: JPEG detection and quantizer estimation. *IEEE TIP*, 12(2), 2003.
- [102] Weihai Li, Yuan Yuan, and Nenghai Yu. Passive detection of doctored JPEG image via block artifact grid extraction. *Signal Processing*, 89(9):1821 – 1829, 2009.
- [103] W.S. Lin, S.K. Tjoa, H.V. Zhao, and K.J. Ray Liu. Digital image source coder forensics via intrinsic fingerprints. *IEEE Trans. on Information Forensics and Security*, 4(3):460 – 475, Sept 2009.
- [104] Cecilia Pasquini, Giulia Boato, and Fernando Pérez-González. Statistical detection of JPEG traces in digital images in uncompressed formats. *IEEE Transactions on Information Forensics and Security*, 12(12):2890 – 2905, Dec. 2017.
- [105] Yi-Lei Chen and Chiou-Ting Hsu. Image tampering detection by blocking periodicity analysis in JPEG compressed images. In *2008 IEEE 10th Workshop on Multimedia Signal Processing*, pages 803–808, Oct 2008.
- [106] A. Desolneux, L. Moisan, and J.-M. Morel. *From Gestalt Theory to Image Analysis*. Springer, 2008.
- [107] A. Gordon, G. Glazko, X. Qiu, and A. Yakovlev. Control of the mean number of false discoveries, Bonferroni and stability of multiple testing. *The Annals of Applied Statistics*, 1(1):179–190, 2007.
- [108] Rafael Grompone von Gioi, Jeremie Jakubowicz, Jean-Michel Morel, and Gregory Randall. LSD: a line segment detector. *IPOL*, 2012.
- [109] Jim Chou, M. Crouse, and K. Ramchandran. A simple algorithm for removing blocking artifacts in block-transform coded images. *IEEE Signal Processing Letters*, 5(2):33–35, Feb 1998.
- [110] Kodak Lossless True Color Image Suite. <http://r0k.us/graphics/kodak/>. [Online; accessed 20-December-2017].
- [111] Friedrich-Alexander-Universitat Erlangen-Nurnberg. Image Manipulation Dataset. <http://www5.cs.fau.de/research/data/image-manipulation/>. [Online; accessed 20-December-2017].
- [112] T. Nikoukhah, J. Anger, T. Ehret, M. Colom, J.-M. Morel, and R. Grompone von Gioi. JPEG grid detection based on the number of DCT zeros and its application to automatic and localized forgery detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 110–118, 2019.
- [113] Hany Farid. *Photo Forensics*. The MIT Press, 2016.
- [114] Denis Teyssou, Jean-Michel Leung, Evlampios Apostolidis, Konstantinos Apostolidis, Symeon Papadopoulos, Markos Zampoglou, Olga Papadopoulou, and Vasileios Mezaris. The InVid plug-in: web video verification on the browser. In *Proceedings of the First International Workshop on Multimedia Verification*, pages 23–30. ACM, 2017.
- [115] Tiziano Bianchi, Alessia De Rosa, and Alessandro Piva. Improved DCT coefficient analysis for forgery localization in JPEG images. In *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2444–2447. IEEE, 2011.

- [116] Hany Farid. Exposing digital forgeries from JPEG ghosts. *IEEE transactions on information forensics and security*, 4(1):154–160, 2009.
- [117] Chryssanthi Iakovidou, Markos Zampoglou, Symeon Papadopoulos, and Yiannis Kompatsiaris. Content-aware detection of JPEG grid inconsistencies for intuitive image forensics. *Journal of Visual Communication and Image Representation*, 54:155–170, 2018.
- [118] Weihai Li, Yuan Yuan, and Nenghai Yu. Passive detection of doctored JPEG image via block artifact grid extraction. *Signal Processing*, 89(9):1821–1829, 2009.
- [119] Shuiming Ye, Qibin Sun, and Ee-Chien Chang. Detecting digital image forgeries by measuring inconsistencies of blocking artifact. In *2007 IEEE International Conference on Multimedia and Expo*, pages 12–15. IEEE, 2007.
- [120] Eric Kee, Micah K. Johnson, and Hany Farid. Digital image authentication from JPEG headers. *IEEE Transactions on Information Forensics and Security*, 6(3-2):1066–1075, 2011.
- [121] Neal Krawetz and Hacker Factor Solutions. A picture’s worth . . . . *Hacker Factor Solutions*, 6, 2007.
- [122] Tina Nikoukhah, Rafael Grompone von Gioi, Miguel Colom, and Jean-Michel Morel. Automatic JPEG grid detection with controlled false alarms, and its image forensic applications. In *2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, pages 378–383. IEEE, 2018.
- [123] Tina Nikoukhah, Miguel Colom, Jean-Michel Morel, and Rafael Grompone von Gioi. Détection de grille JPEG par compression simulée. *preprint*, 2019.
- [124] David G. Lowe. Visual recognition from spatial correspondence and perceptual organization. In *IJCAI*, pages 953–959. Citeseer, 1985.
- [125] Andrew P. Witkin and Jay M. Tenenbaum. On the role of structure in vision. In *Human and machine vision*, pages 481–543. Elsevier, 1983.
- [126] Rafael Grompone von Gioi, Jeremie Jakubowicz, Jean-Michel Morel, and Gregory Randall. LSD: A fast line segment detector with a false detection control. *IEEE transactions on pattern analysis and machine intelligence*, 32(4):722–732, 2010.
- [127] José Lezama, Rafael Grompone von Gioi, Gregory Randall, and Jean-Michel Morel. Finding vanishing points via point alignments in image primal and dual domains. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 509–515, 2014.
- [128] Axel Davy, Thibaud Ehret, Jean-Michel Morel, and Mauricio Delbracio. Reducing anomaly detection in images to detection in noise. In *2018 25th IEEE International Conference on Image Processing (ICIP)*, pages 1058–1062. IEEE, 2018.
- [129] Quentin Bammey, Rafael Grompone von Gioi, and Jean-Michel Morel. Automatic detection of demosaicing image artifacts and its use in tampering detection. In *2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, pages 424–429. IEEE, 2018.
- [130] Jean Serra. *Image Analysis and Mathematical Morphology*. Academic Press, 1982.
- [131] Gerald Schaefer and Michal Stich. Ucid: An uncompressed color image database. In *Storage and Retrieval Methods and Applications for Multimedia 2004*, volume 5307, pages 472–481. International Society for Optics and Photonics, 2003.
- [132] Vincent Christlein, Christian Riess, Johannes Jordan, Corinna Riess, and Elli Angelopoulou. An evaluation of popular copy-move forgery detection approaches. *IEEE Transactions on information forensics and security*, 7(6):1841–1854, 2012.
- [133] Davide Cozzolino, Giovanni Poggi, and Luisa Verdoliva. Efficient dense-field copy-move forgery detection. *IEEE Transactions on Information Forensics and Security*, 10(11):2284–2297, 2015.

- [134] Thibaud Ehret. Automatic Detection of Internal Copy-Move Forgeries in Images. *Image Processing On Line*, 8:167–191, 2018.
- [135] Robert L. Brown, Wolfgang Wild, and Charles Cunningham. ALMA—the Atacama large millimeter array. *Advances in Space Research*, 34(3):555–559, 2004.
- [136] D. H. Schaubert, A. O. Boryssenko, A. Van Ardenne, J. G. Bij de Vaate, and C. Craeye. The square kilometer array (SKA) antenna. In *IEEE International Symposium on Phased Array Systems and Technology, 2003.*, pages 351–358. IEEE, 2003.
- [137] Javier Preciozzi, Andrés Almansa, Pablo Musé, Sylvain Durand, Ali Khazaal, and Bernard Rougé. A Sparsity-Based Variational Approach for the Restoration of SMOS Images From L1A Data. *IEEE Transactions Geosciences and Remote Sensing*, 55(5):2811–2826, feb 2017.
- [138] I. Corbella, F. Torres, A. Camps, A. Colliander, A. Martin-Neira, S. Ribo, K. Rautiainen, N. Duffo, and M. Vall-llossera. MIRAS end-to-end calibration: application to SMOS L1 processor. *IEEE Transactions on Geoscience and Remote Sensing*, 43(5):1126–1134, 2005.
- [139] Mo Chen, Jessica Fridrich, Miroslav Goljan, and Jan Lukáš. Determining image origin and integrity using sensor noise. *Information Forensics and Security, IEEE Transactions on*, 3(1):74–90, 2008.
- [140] Paweł Korus. Digital image integrity—a survey of protection and verification techniques. *Digital Signal Processing*, 71:1–26, 2017.
- [141] Hany Farid. Image forgery detection—a survey. 2009.
- [142] Yongzhen Ke, Qiang Zhang, Weidong Min, and Shuguang Zhang. Detecting image forgery based on noise estimation. *International Journal of Multimedia and Ubiquitous Engineering*, 9(1):325–336, 2014.
- [143] Thibaut Julliand, Vincent Nozick, and Hugues Talbot. Automated image splicing detection from noise estimation in raw images. 2015.
- [144] Babak Mahdian and Stanislav Saic. Using noise inconsistencies for blind image forensics. *Image and Vision Computing*, 27(10):1497–1503, 2009.
- [145] Thibaut Julliand, Vincent Nozick, and Hugues Talbot. Image noise and digital image forensics. In *International Workshop on Digital Watermarking*, pages 3–17. Springer, 2015.
- [146] Thibault Julliand, Vincent Nozick, and Hugues Talbot. Automatic image splicing detection based on noise density analysis in raw images. In *International Conference on Advanced Concepts for Intelligent Vision Systems*, pages 126–134. Springer, 2016.
- [147] Christophe Destruel, Vincent Itier, Olivier Strauss, and William Puech. Color noise-based feature for splicing detection and localization. In *2018 IEEE 20th International Workshop on Multimedia Signal Processing (MMSP)*, pages 1–6. IEEE, 2018.
- [148] Xunyu Pan, Xing Zhang, and Siwei Lyu. Exposing image splicing with inconsistent local noise variances. In *2012 IEEE International Conference on Computational Photography (ICCP)*, pages 1–10. IEEE, 2012.
- [149] Bo Liu and Chi-Man Pun. Splicing forgery exposure in digital image by detecting noise discrepancies. *International Journal of Computer and Communication Engineering*, 4(1):33, 2015.
- [150] Thibault Julliand, Vincent Nozick, Isao Echizen, and Hugues Talbot. Using the noise density down projection to expose splicing in JPEG images. 2017.
- [151] Hui Zeng, Yifeng Zhan, Xiangui Kang, and Xiaodan Lin. Image splicing localization using pca-based noise level estimation. *Multimedia Tools and Applications*, 76(4):4783–4799, 2017.
- [152] Miguel Colom and Antoni Buades. Analysis and extension of the Ponomarenko et al. method, estimating a noise curve from a single image. *Image Processing On Line*, 3:173–197, 2013.