"Methods for solving regularized inverse problems : from non-Euclidean fidelities to computational imaging applications"

Degraux, Kévin

Abstract
Many branches of science and engineering are concerned with the problem of recording signals from physical phenomena. However, an acquisition system does not always directly provide the high-quality signal representations that a given application may require. Signal processing and the study of inverse problems offer a set of powerful tools to recover a good signal quality from altered raw measurements. After a preliminary overview of the field, this thesis presents three main contributions. They involve, in various degrees, the interconnected constituents of a regularized inverse problem: the forward model, the prior or regularization, the data fidelity, and the recovery method. The first and most theoretical contribution focuses on recovering a key structural property of a sparse signal, its support. It discusses guarantees associated to a convex optimization method with atypical fidelity, e.g., using a non-Euclidean norm. The second part introduces a method for learning a convoluti...

Document type: Thèse (Dissertation)

Référence bibliographique

Degraux, Kévin. Methods for solving regularized inverse problems : from non-Euclidean fidelities to computational imaging applications. Prom. : Jacques, Laurent
METHODS FOR SOLVING REGULARIZED INVERSE PROBLEMS:
FROM NON-EUCLIDEAN FIDELITIES TO COMPUTATIONAL IMAGING APPLICATIONS

Kévin Degraux
ICTEAM Institute
Université catholique de Louvain

This dissertation is submitted for the degree of
Ph.D. in Engineering Sciences

Louvain School of Engineering

September 2017
Ph.D. Committee

Prof. Laurent Jacques, supervisor, Université catholique de Louvain, Belgium
Prof. Gabriel Peyré, École Normale Supérieure, Paris, France
Prof. Ulugbek S. Kamilov, Washington University, St. Louis, USA
Prof. Gauthier Lafruit, Université libre de Bruxelles, Belgium
Prof. Christophe De Vleeschouwer, Université catholique de Louvain, Belgium
Prof. David Bol, president, Université catholique de Louvain, Belgium

All unpublished material in this document is © 2017 of the author, all rights reserved.
ACKNOWLEDGEMENTS

I would like to thank my amazing advisor, Laurent Jacques, without whom this thesis would never have seen the light of day. During these last five years, Laurent has been my mentor and guide. He is not only a model of scientific integrity, mathematical rigor and hard work; he is also incredibly available, supporting, kind and generous. He has helped me in every aspect of my early career as a researcher and he always has the right advice to give. I thank him for introducing me to the image and signal processing community and to so many interesting and talented people; for helping me get funding, write and proofread papers, sometimes even late at night, and prepare my presentations and posters; for giving so many insights in such a wide variety of research problems; for taking so much time and energy to help me going forward in my projects. Finally, I thank him for all the small things that made my stay in the ISPGroup so pleasant and rewarding.

I am also grateful to Gabriel Peyré who kindly accepted to be part of my thesis committee and particularly for hosting me for three months in Paris. Gabriel has been an excellent host and I thank him for the many fruitful discussions we had. I would also like to thank everyone I met during my stay in the Ceremade Lab and Inria (Quentin, Clarice, Vincent, Maxime, Aude, Jean, Lénaïc, Arnaud, Luca, Marco, Camille, Raphaël, Raphaël, Michael, Fang, and all the others) for adopting me so fast and making me feel at home. My thanks of course also go to Jalal Fadili who hosted me and made me feel welcome in Caen. With Gabriel, Jalal played a great part in the making of the third chapter of this thesis and working with them has been truly stimulating.

Next, I would like to express my gratitude to Ulugbek Kamilov who hosted me at MERL in Boston. Working with Ulugbek was a real
pleasure and the whole stay was a fantastic experience. I also thank him for accepting to be part of the jury. I also thank everyone at MERL and in particular Petros Boufounos, Hassan Mansour, Dehong Liu and the rest of the Multimedia team, Musau and Elisabeth, all the interns (Daniel, Maxime, Yorgos, Vahan, Ryan, ...) and everyone I met during my stay in Boston.

I would like to warmly thank Valerio Cambareri for sharing my first thesis project on Hyperspectral Imaging. It has always been pleasant and easy to collaborate with Valerio. He also became a direct colleague, during his post-doc in our lab, and a friend. My gratitude also goes to Gauthier Lafruit who kindly accepted to be a member of the jury. Gauthier was one of the main instigators and coordinators of my collaboration with Valerio and IMEC. With them, I also thank Bert Geelen and the Integrated Imaging team at IMEC who also took part in this Hyperspectral Imaging projet.

I would like to thank Christophe De Vleeschouwer for agreeing to be part of my thesis committee and my jury. Christophe has always had great advice from the very first day, when I came to him to look for a master’s thesis project, to this day for my job search. I also thank David Bol for agreeing to be the president of the jury.

I would like to give special thanks to Stéphanie, Benoît, and Simon, my office mates, and François and Pierre-Yves who not only helped me proofread my thesis but also gave advice and feedback on my posters and talks, helped me solve math problems and took professional looking pictures of me. We had countless stimulating chats about the concerning state of the world, research, math, physics, bad jokes and coffee. My thanks extend to Amir, Chun-Lei, Adriana, Cédric, Damien, Thomas, Augustin, Arnaud, Prasad, Giorgios, Amit, Quentin, Mohieddine, Pascaline, David, Esther, and everyone who worked during my thesis in the the Stévin-ELEN building (not forgetting Alice, the coffee machine). I would also like to thank Jean, François, Brigitte, Isabelle, Nathalie, and all the administrative and technical staff. I also thank Claude
Oestges, Bruno Dehez, Christophe Craeye, but also Simon, Maxence, Corentin and Joachim for sharing with me the teaching supervision of the ELEC1101 project during those four years, which I really enjoyed. In general, I thank everyone in the ELEN department and in the ICTEAM institute with whom I had the chance to interact during these four years. I also thank the F.R.S-FNRS for the funding, without which this Ph.D. would not have been possible.

With all my heart, I thank my parents, my brother and my friends for their constant support. Most of all, I thank Elisabeth who has always been there for me. She helped me countless times, always giving me sound advice, proofreading my writings and finding the perfect formulations. She gave me strength and motivation when I was down, and shared my joy and enthusiasm in good times.
ABSTRACT

Many branches of science and engineering are concerned with the problem of recording signals from physical phenomena. However, an acquisition system does not always directly provide the high-quality signal representations that a given application may require. Signal processing and the study of inverse problems offer a set of powerful tools to recover a good signal quality from altered raw measurements.

After a preliminary overview of the field, this thesis presents three main contributions. They involve, in various degrees, the interconnected constituents of a regularized inverse problem: the forward model, the prior or regularization, the data fidelity, and the recovery method.

The first and most theoretical contribution focuses on recovering a key structural property of a sparse signal, its support. It discusses guarantees associated to a convex optimization method with atypical fidelity, e.g., using a non-Euclidean norm. The second part introduces a method for learning a convolutional dictionary, which is to be used as a multimodal imaging prior. This constitutes, indeed, a practical way of sharing information between several imaging modalities, such as depth and light intensity, thus enhancing the reconstruction quality associated to each inverse problem. The last contribution revolves around the design of two multispectral compressive imaging strategies using spectrally filtered sensors. The first scheme relies on a generalized inpainting formulation in the multispectral volume, while the second system leverages the principles of compressed sensing from coded optical convolutions. The chapter studies and compares these two sensing models and discusses implementation challenges and tradeoffs. Each part of the thesis involves a detailed analysis that is validated by numer-
ical experiments. The dissertation concludes with a discussion about the perspectives and open questions.
# Table of Contents

Nomenclature xxi

1 Introduction 1
   1.1 Motivations .............................................. 1
   1.2 Outline and contributions ................................. 4

2 Preliminaries 7
   2.1 Regularized inverse problems ............................ 7
       2.1.1 Forward model ....................................... 8
       2.1.2 Low complexity priors ............................... 15
       2.1.3 Sensing model and embedding ......................... 25
   2.2 Recovery methods ......................................... 32
       2.2.1 General optimization formulation ..................... 33
       2.2.2 Non-convex recovery methods ......................... 35
       2.2.3 Convex recovery methods ............................. 41
       2.2.4 Algorithms for convex optimization ................. 46
       2.2.5 Dictionary Learning ................................. 51

3 Sparse Support Recovery with Convex Fidelity Constraint 57
   3.1 Introduction .............................................. 58
       3.1.1 Sparse regularization with convex fidelity constraint 58
       3.1.2 Dual Certificates .................................... 60
       3.1.3 Main result for sparse support recovery ............. 62
       3.1.4 Relation to Prior Works .............................. 64
3.2 Preliminaries and main result .................................. 65
3.2.1 Noiseless support stability .................................. 65
3.2.2 Model subspace and restricted injectivity conditions 66
3.2.3 Formal statement of the main result ....................... 71
3.3 Proofs .............................................................. 74
3.3.1 Proofs of the lemmas and subdifferential decom-
   posability .......................................................... 74
3.3.2 Proof of Theorem 5 ............................................ 83
3.4 Numerical experiments ........................................... 91
3.5 Conclusion ......................................................... 94

4 Online Convolutional Dictionary Learning for
   Multimodal Imaging .................................................. 95
4.1 Introduction ......................................................... 96
4.1.1 Main Contributions ............................................ 97
4.1.2 Related Work ................................................... 99
4.2 Proposed Method .................................................. 100
4.2.1 Problem Formulation ........................................ 100
4.2.2 Online Convolutional Dictionary Learning
   Algorithm ............................................................ 104
4.2.3 Dictionary update ............................................. 105
4.2.4 Implementation details ..................................... 107
4.3 Experimental Evaluation ........................................ 110
4.4 Conclusion .......................................................... 117

5 Multispectral Compressive Imaging Strategies using
   Fabry-Pérot Filtered Sensors ....................................... 119
5.1 Introduction ......................................................... 120
5.1.1 Main Contributions ............................................ 121
5.1.2 Related Work ................................................... 122
5.2 Preliminaries ....................................................... 125
5.2.1 Fabry-Pérot Filtered Sensors ............................... 125
5.2.2 Forward model and analysis prior ......................... 127
## Table of contents

5.2.3 Recovery Method ............................................. 128
5.3 Multispectral Compressive Imaging by Generalized In-painting ............................................. 132
5.3.1 Image Formation Model ..................................... 132
5.3.2 Simulations ..................................................... 135
5.3.3 Experiments ..................................................... 138
5.4 Multispectral Compressive Imaging by Out-of-Focus Random Convolution ..................................... 140
5.4.1 Image Formation Model ..................................... 140
5.4.2 Non-idealities and practical considerations .............. 145
5.4.3 Sensing matrix implementation ............................. 151
5.4.4 Simulations ..................................................... 153
5.5 Final Comparison .................................................. 155
5.6 Conclusion ......................................................... 160

### 6 Conclusions

6.1 Summary ......................................................... 163
6.2 Perspectives and open questions .............................. 166

References ............................................................. 173

Appendix A Elements of Convex Optimization .................. 195
NOMENCLATURE

Symbols

\( x \)  A (column) vector (lowercase boldface).

\( \Phi \)  A matrix (uppercase boldface).

\( c \)  A scalar quantity (lowercase, integers denoting a number of objects may be uppercase).

\( C \)  A set (calligraphic, with some exceptions).

\([n]\)  The set of indices \( \{1, \ldots, n\} \).

\( I, I^c \)  An index set \( I \subseteq [n] \) and its complement \( I^c = [n] \setminus I \) (uppercase letters, with some exceptions).

\( \#I, |I| \)  Cardinality of a finite set \( I \).

\( F(\cdot), f(\cdot) \)  A scalar-valued function (can be uppercase, lowercase).

\( \Gamma(\cdot) \)  A vector-valued or matrix-valued function (boldface).

\( \|x\|_\alpha, \|x\|_\beta \)  The \( \ell_\alpha \)-norm and its dual \( \ell_\beta \)-norm for \( \alpha, \beta \in [1, +\infty] \) such that \( \frac{1}{\alpha} + \frac{1}{\beta} = 1 \), or quasinorm for \( \alpha \in (0, 1] \), \( \|x\|_\alpha \triangleq (\sum_{i=1}^{n} |x_i|^\alpha)^{\frac{1}{\alpha}} \) and \( \|x\|_\infty \triangleq \max_{i\in[n]} |x_i| \). The shorthand \( \|x\| = \|x\|_2 \) may be used for \( \alpha = 2 \).

\( B_\alpha \)  The \( \ell_\alpha \)-ball \( B_\alpha \triangleq \{x \mid \|x\|_\alpha \leq 1\} \).
The induced $\ell_a, \ell_b$-operator norm of matrix $\Phi$,
$$\|\Phi\|_{a,b} \triangleq \sup_{\|x\|_a=1} \|\Phi x\|_b.$$ The shorthand $\|\Phi\| = \|\Phi\|_2 = \|\Phi\|_{2,2}$ may be used for $a, b = 2$.

The support of $x$, $\text{supp}(x) \triangleq \{i \mid |x_i| \neq 0\}$.

The $\ell_0$-(pseudo)norm of a vector $\|x\|_0 \triangleq \# \text{supp}(x)$.

The set of $s$-sparse signals, of synthesis $s$-sparse signals in dictionary $D$ and of analysis $\kappa$-cosparse signals in analysis dictionary $A$.

The hard and soft thresholding operators.

The saturation support of $x$, $\text{sat}(x) \triangleq \{i \mid |x_i| = \|x\|_\infty\}$.

The sub-differential of a convex function $f$.

The $\ell_\alpha$-error of the best $s$-terms approximation to $x$ (in the canonical basis and in dictionary $D$).

The extended reals $\bar{\mathbb{R}} \triangleq \mathbb{R} \cup \{+\infty\}$.

The convex conjugate of a function $f \in \Gamma_0(\mathbb{R}^n)$.

The set of proper convex lower semi-continuous (l.s.c.) functions over $\mathbb{R}^n$.

The gradient of $f : \mathbb{R}^n \to \mathbb{R}$.

The covariant derivative of $f : \mathcal{T} \to \mathbb{R}$ with respect to the subspace $\mathcal{T} \subseteq \mathbb{R}^n$ at $x \in \mathcal{T}$, defined, for all $u \in \mathbb{R}^n$, as $\langle u, \nabla_{\mathcal{T}} f(x) \rangle = \frac{d}{dt} f(P_{\mathcal{T}}(x + tu))|_{t=0}$.

The Hessian of $f : \mathbb{R}^n \to \mathbb{R}$.

The matrix equivalent of the covariant Hessian of $f : \mathcal{T} \to \mathbb{R}$ with respect to the subspace $\mathcal{T} \subseteq \mathbb{R}^n$ at $x \in \mathcal{T}$, i.e., the auto-adjoint bilinear map, defined, for all $u \in \mathbb{R}^n$, by $\langle u, \nabla_{\mathcal{T}}^2 f(x) u \rangle = \frac{d^2}{dt^2} f(P_{\mathcal{T}}(x + tu))|_{t=0}$. 
Nomenclature

\[ \nabla \Gamma \in \mathbb{R}^{n \times m} \] The Jacobian of \( \Gamma : \mathbb{R}^n \to \mathbb{R}^m \).

\[ \nabla_i \Gamma \in \mathbb{R}^{q \times m} \] The Jacobian with respect to the \( i \)th input variable of size \( q \) of \( \Gamma : \mathbb{R}^n \to \mathbb{R}^m \).

Span(\( C \)) The subspace spanned by the vectors of a nonempty set \( C \).

aff(\( C \)) The affine hull, i.e., the smallest affine subspace in which the nonempty set \( C \) lies.

bd(\( C \)) The boundary of a nonempty set \( C \).

int(\( C \)) The interior of a nonempty set \( C \).

par(\( C \)) The subspace parallel to a nonempty convex set \( C \), \( \text{par}(C) \triangleq \mathbb{R}(C - C) \).

rbd(\( C \)) The relative boundary of a nonempty set \( C \), i.e., the boundary of \( C \) relative to aff(\( C \)).

ri(\( C \)) The relative interior of a nonempty set \( C \), i.e., the interior of \( C \) relative to aff(\( C \)).

\( \iota_\mathcal{C}(\cdot) \) The indicator function \( \iota_\mathcal{C}(u) \) associated to the set \( \mathcal{C} \) is 0 if \( u \in \mathcal{C} \) and \( +\infty \) otherwise.

\( \text{prox}_f(\cdot) \) The proximal operator associated to function \( f \).

\( \gamma_\mathcal{C}(\cdot) \) Gauge associated to the non-empty convex set \( \mathcal{C} \).

\( \gamma^\circ_\mathcal{C}(\cdot) \) Polar gauge of a gauge \( \gamma_\mathcal{C} \) associated to the non-empty convex set \( \mathcal{C} \).

\( \mathcal{C}^\circ \) Polar set of the non-empty convex set \( \mathcal{C} \).

\( \Phi^*, \Phi^\top \) The conjugate transpose (or adjoint) and the transpose of a matrix \( \Phi \).
diag(\(x\)) The diagonal matrix whose diagonal entries are the entries of \(x\).

diag(\(A, B, C\)) The block diagonal matrix formed by stacking the matrices \(A, B\) and \(C\) diagonally without overlap.

diag_n(\(A\)) The block diagonal matrix formed by repeating \(n\) times \(A\) diagonally, without overlap.

\(\Phi^{-1}, \Phi^\dagger, \Phi^+\) The inverse, a left or right inverse and the Moore-Penrose pseudo-inverse of \(\Phi\).

\(\delta_i\) The canonical vector of index \(i\).

\(P_V\) The orthogonal projector onto a subspace \(V\).

\(\text{Id}\) The identity matrix. The dimension should always be clear from the context. Note that \(\text{Id}_I\) is a zero-padding operator (the output has zeros on \(I^c\)) and \(\text{Id}^S\) is a selection operator.

\(\text{svd}\) For a matrix \(\Phi \overset{\text{svd}}{=} U \Sigma V^*\) denotes the singular value decomposition. \(\Sigma\) is omitted when all singular values are equal to 1. Moreover \(U = V \Leftrightarrow \Phi = \Phi^*\).

\(x^I, x^V, x^V\) A vector restricted to an index set \(I\), projected on a linear subspace \(V\) or expressed in an orthonormal basis \(V\). For instance if \(x^V \overset{\Delta}{=} P_V x \in V \subseteq \mathbb{R}^n\) is the vector projected on the subspace \(V\) of dimension \(v\), let \(V \in \mathbb{R}^{n \times v}\) be an orthonormal basis of \(V\), such that \(P_V \overset{\text{svd}}{=} V V^*\), then \(x^V \overset{\Delta}{=} V^* x \in \mathbb{R}^v\) is the vector expressed in \(V\).

\(\Phi^S, \Phi^I, \Phi^S_I\) For two index sets \(S \subseteq [m], I \subseteq [n]\), the submatrix of \(\Phi \in \mathbb{R}^{m \times n}\) restricted to the rows indexed by \(S\) or the columns indexed by \(I\) or both, respectively.
**Nomenclature**

Φ<sub>T</sub>, Φ<sub>V</sub>, Φ<sup>T</sup><sub>V</sub> For two linear subspaces \( T \subseteq \mathbb{R}^m \) and \( V \subseteq \mathbb{R}^n \), we have \( \Phi^T \triangleq P_T \Phi \), \( \Phi_V \triangleq \Phi P_V \), \( \Phi^{T,V} \triangleq P_T \Phi P_V \in \mathbb{R}^{m \times n} \)

\*i.e.*, the matrix \( \Phi \in \mathbb{R}^{m \times n} \) whose columns are projected on \( T \) or whose rows are projected on \( V \) or both. Combinations with index sets, \( e.g., \Phi^T_I \) are allowed.

Φ<sub>T</sub>, Φ<sub>V</sub>, Φ<sup>T</sup><sub>V</sub> For two orthonormal bases \( T \in \mathbb{R}^{m \times t} \) and \( V \in \mathbb{R}^{n \times v} \), \( \Phi^T \triangleq T^* \Phi \), \( \Phi_V \triangleq \Phi V \), \( \Phi^{T,v} \triangleq T^* \Phi V \in \mathbb{R}^{t \times v} \)

\*i.e.*, the matrix \( \Phi \in \mathbb{R}^{m \times n} \) whose columns are expressed in terms of their coefficients in \( T \) or whose rows in terms of their coefficients in \( V \) or both. Combinations with index sets are allowed, \( e.g., \Phi^{T,S} \) for \( I \subset [m] \) and \( S \subset [t] \).

\([x]_i, x_i, [\Phi]_{i,j}, \varphi_{i,j}\) The \( i^{th} \) entry of a vector \( x \) or the entry \((i,j)\) of a matrix \( \Phi \).

\(x * y\) The full convolution between \( x \) and \( y \) (either 1-D or 2-D according to the context).

\(x \odot y\) The circular convolution between \( x \) and \( y \) (either 1-D or 2-D according to the context). The smaller vector is padded with zeros to match the size of the bigger vector.

\(x \check{*} y\) The valid convolution between \( x \) and \( y \) (either 1-D or 2-D according to the context). It can be computed either from the full convolution, \( x * y \), restricted to indices such that \( x \) and \( y \) overlap entirely, or from the circular convolution \( x \odot y \) restricted to the appropriate indices.

\(\mathcal{N}(\mu, \sigma^2), \mathcal{U}(\delta_l, \delta_u)\) The Gaussian distribution with esperance \( \mu \) and variance \( \sigma^2 \) and the uniform distribution between \( \delta_l \) and \( \delta_u \).
For some probability distribution \( \mathcal{P} \), \( x_i \overset{i.i.d.}{\sim} \mathcal{P} \) indicates that for all \( i \in [n] \), the entries \( x_i \) of a vector \( x \in \mathbb{R}^n \) are independent and identically distributed according to the distribution \( \mathcal{P} \).

\[ \mathcal{N}(\mu, \sigma^2)^{m \times n} \]

\[ \Phi \overset{i.i.d.}{\sim} \mathcal{N}(\mu, \sigma^2)^{m \times n} \]

indicates that the entries of the matrix \( \Phi \in \mathbb{R}^{m \times n} \) follow \( \varphi_{i,j} \overset{i.i.d.}{\sim} \mathcal{N}(\mu, \sigma^2) \).

\[ \mathbb{P}(B) \]

The probability of event \( B \) to happen.

\[ \mathbb{E}[X] \]

The expectation of a random variable \( X \).

**Acronyms / Abbreviations**

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADMM</td>
<td>Alternating Directions Method-of-Multipliers</td>
</tr>
<tr>
<td>AWGN</td>
<td>Additive White Gaussian Noise</td>
</tr>
<tr>
<td>BP</td>
<td>Basis Pursuit</td>
</tr>
<tr>
<td>BPDN</td>
<td>Basis Pursuit DeNoising</td>
</tr>
<tr>
<td>CA</td>
<td>Coded Aperture</td>
</tr>
<tr>
<td>CASSI</td>
<td>Coded Aperture Snapshot Spectral Imaging</td>
</tr>
<tr>
<td>CoSaMP</td>
<td>Compressive Sampling Matching Pursuit</td>
</tr>
<tr>
<td>CP</td>
<td>Chambolle-Pock</td>
</tr>
<tr>
<td>CS</td>
<td>Compressed Sensing</td>
</tr>
<tr>
<td>CT</td>
<td>X-Ray Computed Tomography</td>
</tr>
<tr>
<td>DCT</td>
<td>Discrete Cosine Transform</td>
</tr>
<tr>
<td>DFT</td>
<td>Discrete Fourier Transform</td>
</tr>
<tr>
<td>DOF</td>
<td>depth-of-field</td>
</tr>
<tr>
<td>DWT</td>
<td>Discrete Wavelet Transform</td>
</tr>
<tr>
<td>Acronym</td>
<td>Definition</td>
</tr>
<tr>
<td>---------</td>
<td>------------</td>
</tr>
<tr>
<td>FB</td>
<td>Forward-Backward</td>
</tr>
<tr>
<td>FFT</td>
<td>Fast Fourier Transform</td>
</tr>
<tr>
<td>FISTA</td>
<td>Fast Iterative Shrinkage Thresholding Algorithm</td>
</tr>
<tr>
<td>FPA</td>
<td>Focal Plane Array</td>
</tr>
<tr>
<td>FP</td>
<td>Fabry-Pérot</td>
</tr>
<tr>
<td>FWHM</td>
<td>Full Width at Half Maximum</td>
</tr>
<tr>
<td>GMM</td>
<td>Gaussian Mixture Model</td>
</tr>
<tr>
<td>HS</td>
<td>hyperspectral</td>
</tr>
<tr>
<td>HSI</td>
<td>hyperspectral imaging</td>
</tr>
<tr>
<td>HTP</td>
<td>Hard Thresholding Pursuit</td>
</tr>
<tr>
<td>IHDN</td>
<td>Iterative Hard DeNoising</td>
</tr>
<tr>
<td>IHT</td>
<td>Iterative Hard Thresholding</td>
</tr>
<tr>
<td>i.i.d.</td>
<td>independent and identically distributed</td>
</tr>
<tr>
<td>ISTA</td>
<td>Iterative Soft Thresholding Algorithm</td>
</tr>
<tr>
<td>Lasso</td>
<td>Least absolute shrinkage and selection operator</td>
</tr>
<tr>
<td>l.s.c.</td>
<td>lower semi-continuous</td>
</tr>
<tr>
<td>MFISTA</td>
<td>Monotone FISTA</td>
</tr>
<tr>
<td>MRI</td>
<td>Magnetic Resonance Imaging</td>
</tr>
<tr>
<td>MSE</td>
<td>Mean Squared Error</td>
</tr>
<tr>
<td>MS</td>
<td>multispectral</td>
</tr>
<tr>
<td>MSRC</td>
<td>Multispectral Random Convolution</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>MSVI</td>
<td>Multispectral Volume Inpainting</td>
</tr>
<tr>
<td>NSP</td>
<td>null space property</td>
</tr>
<tr>
<td>ODCT</td>
<td>Oversampled DCT</td>
</tr>
<tr>
<td>ONB</td>
<td>orthonormal basis</td>
</tr>
<tr>
<td>PET</td>
<td>Positron Emission Tomography</td>
</tr>
<tr>
<td>PFE</td>
<td>Partial Fourier Ensemble</td>
</tr>
<tr>
<td>PMF</td>
<td>Probability Mass Function</td>
</tr>
<tr>
<td>PSF</td>
<td>Point Spread Function</td>
</tr>
<tr>
<td>PSNR</td>
<td>Peak Signal-to-Noise Ratio</td>
</tr>
<tr>
<td>QCS</td>
<td>Quantized Compressed Sensing</td>
</tr>
<tr>
<td>RIC</td>
<td>Restricted Isometry Constant</td>
</tr>
<tr>
<td>RIP</td>
<td>Restricted Isometry Property</td>
</tr>
<tr>
<td>RME</td>
<td>Random Matrix Ensemble</td>
</tr>
<tr>
<td>RNSP</td>
<td>robust null space property</td>
</tr>
<tr>
<td>ROI</td>
<td>region of interest</td>
</tr>
<tr>
<td>RV</td>
<td>Random Vector</td>
</tr>
<tr>
<td>SLM</td>
<td>Spatial Light Modulator</td>
</tr>
<tr>
<td>SNR</td>
<td>Signal-to-Noise Ratio</td>
</tr>
<tr>
<td>TV</td>
<td>Total Variation</td>
</tr>
<tr>
<td>UDWT</td>
<td>Undecimated Discrete Wavelet Transform</td>
</tr>
<tr>
<td>ULS</td>
<td>Union of Low-dimensional Subspaces</td>
</tr>
</tbody>
</table>
Nomenclature

URA  Uniformly Redundant Array
VIS  visible light
Chapter 1

Introduction

1.1 Motivations

Data acquisition constitutes an essential part of many disciplines of science and engineering. From medicine to physics, from chemistry to biology, from industrial processes and robotics to environmental sciences and astronomy; all these fields heavily rely on an accurate, fast and reliable acquisition and representation of signals from the surrounding world. In this thesis, we are mainly interested in signals that exist within a definite boundary in space, time, etc. In more precise mathematical terms, those signals have a compact domain. An instance of particular importance is that of images. When capturing such a signal, one aims at getting the most useful, reliable and interpretable information out of the sensing device. Unfortunately, signal acquisition is not always as seamless and banal as taking a picture of one’s dessert with a smartphone; the physics and configuration of the device do not always allow to directly reach the quality requirements of the targeted application. In those cases, signal processing is indispensable to make sense, and get the most scientific value out, of the acquired data.

In signal processing, a popular approach is to build a mathematical model of the sensing process, commonly referred to as the forward model.
We represent the ideal target signal (or scene) by an unknown vector$^1$, $x_0$. The sensing instrument response is modeled, as precisely as possible, as an operator, $\Phi(\cdot)$. The recorded measurements are then, naturally, modeled as $y \approx \Phi(x_0)$, the result of the unknown scene being observed through the sensing operator. The approximation sign stems from the fact that despite our efforts, the forward model might be inaccurate or, as is more commonly assumed, that the observations might be affected by noise. The goal is then to provide an estimate, $\hat{x}$, i.e., guess as precisely as possible what was the actual input of the process. In other words, the goal is to “invert” the forward model. This general approach is thus called an inverse problem.

Of course, in most situations, one cannot straightforwardly compute an inverse, $\Phi^{-1}(y)$ and use it as the estimate. First, even if in some cases an inverse operator, $\Phi^{-1}(\cdot)$, may actually exist, it is often not robust to noise. Because the observations are imperfect, and if the operator is badly conditioned, computing an inverse will not give an accurate estimate. Second, and most importantly, the sensing operator is often not one-to-one (i.e., not injective), meaning that several, sometimes wildly different, inputs may produce the exact same measurements (the operator is not invertible since the inverse is not unique). This is where regularization comes into play. Regularizing the inverse problem (or solving the regularized inverse problem) amounts to find a nice, stable criterion, to choose from the estimation candidates. The criterion is called a prior model, because it amounts to making hypotheses on the unknown target, based on prior information on its origin, for instance, its inherent, natural structure. Figure 2.1 illustrates the principle of a regularized inverse problem.

One important, emblematic example of prior model$^2$ that has gained a tremendous amount of popularity in the last decades, is that of sparsity and sparse models. It consists in assuming that the unknown vector

---

$^1$Formal definitions are given in Chapter 2  
$^2$Other examples, along with references, can be found in Chapter 2.
can be (almost) perfectly described by a very small number of linearly combined basic elements, called atoms. Sparsity has been successfully used for years in signal compression and is one of the main rationale behind immensely popular standards such as JPEG for images, MP3 for audio, and many more.

Common examples of regularized inverse problems are denoising, superresolution, inpainting, and de-blurring, which aim at mitigating or erasing unintentional alterations that occurred during acquisition. These examples thus focus on post-acquisition restoration of compromised data. It is also possible to co-design the sensing device along with an inverse problem formulation for recovering high quality data, including features that would otherwise be inaccessible. When the acquired signals are images, this kind of co-design is called computational imaging. In the same vein, compressed sensing stands for inverse problems in which the sensing operator has been purposely designed to compress the unknown
signal: it reduces the amount of measured data compared to what would be required for a direct sampling of the target scene (the famous Nyquist rate). Classical compressed sensing theory leverages randomness of the sensing operator and sparsity of the unknown target to provide recovery guarantees.

This dissertation is about methods for solving regularized inverse problems. Part of the thesis is dedicated to providing theoretical recovery guarantees for a general class of methods for solving sparsity-regularized inverse problems with so-called non-Euclidean fidelities. The rest of the thesis focuses on computational imaging applications. We study how to build a good model of the sensing operation and how to design or learn prior models adapted to a certain class of target signals. We also study how to co-design an acquisition device with a reconstruction procedure in order to get better performances than through direct imaging.

### 1.2 Outline and contributions

**Chapter 2** introduces key concepts and definitions that are useful in the next chapters. It provides a general background by citing classical results and many references related to the state of the art in inverse problems and associated research topics. It also provides a brief, non-exhaustive overview of recovery methods, among which are convex and non-convex optimization formulations and algorithms.

**Chapter 3** studies sparse support recovery (see Chapter 2 for definitions), when using a compact convex constraint for data fidelity. Specifically, this theoretical work analyzes the solutions of a convex, constrained $\ell_1$-regularized minimization program, which is routinely used for solving sparse inverse problems. It extends well known results for the usual Euclidean fidelity, commonly known as the least-squares solution, to more general convex constraints including, among others, $\ell_\alpha$-losses for
\(\alpha \geq 1\), mixed norms and the nuclear norm. It characterizes in particular support stability or instability of perturbed solutions, and provides an explicit way of predicting their extended supports and sign patterns. The main theoretical findings unify two different situations: non-smooth losses, including polyhedral losses such as \(\ell_1\) or \(\ell_\infty\) constraints, and (partially) smooth losses, such as the \(\ell_2\)-norm, and mixed norms. The theory is built upon classical results from convex analysis and convex optimization, coupled with a rich framework introduced in [Vaiter et al., 2015].

**Chapter 4** proposes a novel approach for multimodal computational imaging. The method is designed to learn data-adaptive convolutional dictionaries for both sparsely representing the signals and reconstructing them from their linear measurements. Specifically, we develop a new online convolutional dictionary learning method suitable for working with large-scale datasets, where the linear measurements of indirectly observed data can be streamed along time.

Our key contributions in this chapter are summarized as follows: We provide a new formulation for multimodal computational imaging, incorporating a convolutional joint sparsity prior and a Total Variation (TV) regularizer. In this formulation, the high resolution images are determined by solving an optimization problem, where the regularizer exploits the redundancies across different modalities. On the other hand, we develop an online convolutional dictionary learning algorithm. By accommodating an additional TV regularizer in the cost, the algorithm is able to learn the convolutional dictionary in an unsupervised fashion, directly from the noisy measurements. We validate our approach for joint intensity–depth imaging.

**Chapter 5** introduces two acquisition device architectures for multispectral (MS) compressive imaging. Both proposed computational imaging techniques use a dedicated sensor which integrates Fabry-Pérot (FP)
spectral filters at pixel level to record measurements in several bands of
the spectrum of light. In particular, they keep a relatively low system-
level complexity as they do not include any dispersive element, unlike
most existing methods for compressive MS imaging.

The first technique provides a simple integration of the FP filtered
sensor in a computational imaging scheme. It is linked to inpainting
and super-resolution as its goal is to fill in the missing pixels, removed
from the multispectral volume by spectral filtering. The second pro-
posed imaging technique adds a layer of complexity allowing higher
compression ratios, in link with compressed sensing theory. By introduc-
ing spatial light modulation before the sensor, it implements random
convolutions, which have several benefits in terms of mathematical and
computational properties.

We use the same approach in both cases to solve the associated in-
verse problem. Specifically, we propose a redundant analysis signal
prior in a convex formulation. The convex problem is then solved with
an efficient algorithm. Through numerical simulations, we explore vari-
ous tradeoffs in different realistic setups. We highlight some practical
guidelines and discuss their complexity tradeoffs to integrate these
schemes into computational imaging systems.

Chapter 6, finally, summarizes the main contributions of each chapter
and discusses perspectives and open questions.
CHAPTER 2

PRELIMINARIES

The goal of this chapter is to provide a brief overview of the area of research in which this thesis lies. We dive deeper into some of the points that are most relevant to this work and give references about the topics that are related but either too complex to be described in this introductory material or not crucial for understanding the contributions in subsequent chapters.

The chapter is divided in two sections. The first one describes the class of problems that we address in this thesis: the so-called regularized inverse problems. In particular, it defines the forward model, the prior information used to regularize those problems and the mathematical aspects associated to the sensing model and the interactions with the prior. The second section is about the optimization methods used for solving those regularized inverse problems.

2.1 Regularized inverse problems

We introduce and formalize here the class of problems that we tackle in this thesis. Note that in this section in particular, and in this dissertation in general (with the notable exception of Chapter 3 which is more abstract), we focus mainly on imaging applications, even though most considerations are generalizable to any kind of discrete signal. We start
by giving a formal definition of the noisy linear forward model. That mathematical expression is precisely the equation that we try to invert in an inverse problem. We also describe a few important noise models and give a few words about where and how one can encounter inverse problems in the field of imaging. Next, we present the low complexity models that are usually used to regularize those inverse problems. We end this section by presenting a few important mathematical characterizations of the sensing model.

2.1.1 Forward model

As informally introduced in Chapter 1, the forward model is a mathematical expression modeling the physical reality of an acquisition process. The aim of solving the associated inverse problem is to find an accurate estimate $\hat{x} \in \mathbb{R}^n$ of the target image $x_0 \in \mathbb{R}^n$. Those are usually modeled as vectors where each entry implicitly corresponds to a specific spatial position in a specific frame or channel. The forward model represents, to some extent, the acquisition procedure that transforms the physical reality in a set of measurements, $y \in \mathbb{R}^m$. Its crafting entails the careful design of the acquisition method, i.e., the engineering of the physical path between the scene and the sensor recording the measurements. It also requires a precise definition of a target image $x_0 \in \mathbb{R}^n$, i.e., the field of view, number of pixels, number of channels, modalities, etc. Finally, it requires an accurate modeling and calibration of the physical reality to cope with non-idealities such as a blur, the noise, etc. In general, one can write the forward model as

$$y = \Phi(x_0) + w(x_0),$$

(2.1)

where $\Phi(\cdot) : \mathbb{R}^n \rightarrow \mathbb{R}^m$ is the known, i.e., modeled, sensing operator and $w(\cdot) : \mathbb{R}^n \rightarrow \mathbb{R}^m$ is the unknown (possibly signal-dependent) additive

$^1$This general formulation also encompasses other types of non-additive noise such as, e.g., Poisson noise, typically written as $y = \mathcal{P}(\Phi(x_0))$, or quantization distortion,
2.1 Regularized inverse problems

noise. In order to simplify the study, we ignore the potential data dependency of the noise term and make the standard assumption that the sensing operator only operates linear combinations of the pixels of the target image and is thus fully equivalent to a matrix multiplication. The noisy linear forward model, illustrated on Figure 2.1, reads

\[ y = \Phi x_0 + w, \]  

(2.2)

where \( \Phi \in \mathbb{R}^{m \times n} \) and \( w \in \mathbb{R}^m \). This equation constitutes a set of \( m \) linear equations with \( n + m \) unknowns, that is \( n \) for the signal of interest \( x_0 \) and \( m \) for the noise term \( w \). The difficulty in solving an inverse problem lies in finding a criterion to select the best solution in the huge space of potential candidates. In other words, given a sensing matrix \( \Phi \), a measurement vector \( y \) and a noise model we need to devise a strategy to find the best reconstruction \( \hat{x} \), i.e., one that is as close as possible to the target \( x_0 \).

**Noise models**

The noise term \( w \), obviously unknown, is usually modeled as a random vector with a given or estimated distribution. The most commonly used noise model is the additive white Gaussian noise. This is the noise model

\[ y = \mathcal{Q}(\Phi(x_0)), \]  

for a quantizer \( \mathcal{Q}(\cdot) \). In that case, one has the equivalence with (2.1) by writing, e.g.,

\[ y = \Phi(x_0) + (\mathcal{P}(\Phi(x_0)) - \Phi(x_0)). \]
that is assumed by default as it occurs in any electronic sensing device, see for instance [Janesick, 2001]. Due to its isotropic nature, it is often associated to the to the Euclidean norm (or $\ell_2$-norm) or equivalently to the least squares data fidelity function (see Section 2.2) which corresponds, in Bayesian terms, to a maximum likelihood of the Gaussian distribution.

Another type of noise sometimes called impulse or salt-and-pepper noise results in sparse outliers in the data. This type of noise comes, for instance, from faulty sensing units or transmission errors [Bovik, 2005]. A related but different noise distribution is the Laplace distribution which gives rise to rare but relatively high noise peaks and a low but non-zero noise floor. A Bayesian analysis of Laplace noise promotes the use of a (non-Euclidean) $\ell_1$-norm fidelity function. However, because of its polyhedral geometry (see Chapter 3), the $\ell_1$-norm is in reality more suited to sparse outliers [Jacques, 2013; Nikolova, 2004a; Studer et al., 2012a].

Finally, when the bit depth of the analog-to-digital converters in an imaging sensor is not sufficient, quantization of the measurements can introduce a distortion that can be modeled as an additive noise which approximately follows a uniform distribution [Gonzalez and Woods, 2007]. The fact that this type of noise is inherently bounded tends to favor the use of yet another non-Euclidean fidelity constraint: the $\ell_\infty$-ball. However, this polyhedral norm entails saturating residual entries in the reconstruction and might not be the best for this type of noise [Jacques et al., 2011].

Chapter 3 is devoted to study the effect of atypical (i.e., non-Euclidean) noise priors on the stability of the structure (namely, the sparse support) of the reconstructed signal.

For completeness, let us mention Poisson noise even though it is not studied in this thesis. Also called shot noise, it is another important noise model which occurs in low-illumination conditions when the recorded signal is proportional to a relatively low number of particles, e.g., photons, hitting the sensing units (e.g., photo-diodes). The Poisson
distribution depends on its mean and therefore shot noise cannot be modeled as a data-independent additive noise vector. Such type of noise has been broadly studied in imaging, see for example [Gonzalez and Woods, 2007; Jähne, 2002].

Several flavors of inverse problems in imaging

A discrete linear sensing operator is sufficient to model many different types of image acquisitions. Some of them are introduced here, and some are illustrated on Figure 2.2.

The simplest and most obvious inverse problem is denoising [Donoho and Johnstone, 1994; Rudin et al., 1992; Starck et al., 2002a, 2007], i.e., when the sensing matrix is the identity $\Phi = \text{Id}$ and the reconstruction procedure aims at mitigating the effect of the noise term, $w$.

When measurements are missing, i.e., $\Phi$ is a binary mask (or restriction) operator, for example, when some regions of the image are occluded or some pixels are corrupted, the problem is called inpainting [Bertalmio et al., 2003; Degraux et al., 2015; Elad et al., 2005]. The aim is to interpolate the available data in order to fill in the missing pixels. In traditional digital color imaging or, as described in Chapter 5, in some form of snapshot multispectral imaging, different channels (colors) of the image are filtered out with an alternating grid (or mosaic) pattern (e.g., the Bayer pattern [Bayer, 1976]) at the sensor level. This filtering can also be modeled as a selection operator leading to an inpainting problem. This particular instance of inpainting is called demosaicking [Baone, 2006] (or demosaicing [Condat, 2008]).

A less direct observation model takes into account the optical blur that may occur between the scene and the sensor. The effect is usually modeled by the blurred image that would be recorded when observing a bright spot. The blurry spot is then used as a convolution kernel called the Point Spread Function (PSF) of the system. The sensing matrix $\Phi$ is then a convolution operator and the corresponding inverse problem is called deconvolution [Taylor et al., 1979] when the PSF is known and blind
Figure 2.2 Five examples of measurements images associated to different inverse problems. Top left is the target image, common to all five examples. For denoising, an additive white gaussian noise was added to the color channels of the image. For deblurring, a motion blur was used. For inpainting, random blocks of 8 by 8 pixels were masked (in white) from the image. The bottom left image depicts the magnitude of the Discrete Fourier Transform (DFT) (in logarithmic scale), and the bottom right, a random selection of Fourier coefficients with variable density sampling for achieving Compressed Sensing (CS) (see Section 2.1.3).
2.1 Regularized inverse problems

deconvolution [Ayers and Dainty, 1988; González et al., 2014] when it is unknown. This encompasses deblurring, for instance, to compensate for a motion blur [Elad and Or, 2001; Krahmer et al., 2006] or an optical lens defect [Biemond et al., 1990], and super-resolution [Candès and Fernandez-Granda, 2014; Demanet et al., 2013; Duval and Peyré, 2015; Gerchberg, 1974], i.e., to break the so-called diffraction limit, e.g., in a telescope [Starck et al., 2002b] or a microscope [Agard, 1984].

In tomography, for example Magnetic Resonance Imaging (MRI) [Lustig et al., 2007], X-Ray Computed Tomography (CT) and Positron Emission Tomography (PET) [Herman, 2009], confocal microscopy [Pawley, 1996], seismic tomography [Lin and Herrmann, 2007], and many others or in interferometry, e.g., Fourier transform spectroscopy [Stroke and Funkhouser, 1965] or astronomical radio interferometry [Wiaux et al., 2009], the image is observed through a series of indirect measurements. The operator $\Phi$ in this case is a discrete transform, approximating the continuous physical response of the acquisition device. When the number of measurements $m$ is equal to the number of pixels in the target image $n$, $\Phi$ is sometimes equivalent to an orthonormal transform which can be implemented using a fast algorithm. In MRI for example, the observations are samples of the 2-D Fourier transform plane. Under appropriate circumstances, the corresponding sensing operator is accurately approximated by the Discrete Fourier Transform (DFT) which is of course implemented with the Fast Fourier Transform (FFT) algorithm.

In a different regime, the number $m$ of (linearly independent) samples is much smaller than the number $n$ of pixels of the target image, i.e.,

$$m \ll n.$$  \hspace{1cm} (2.3)

This is particularly useful when $n$ is large in order to mitigate the acquisition time or the amount of data that needs to be acquired, transmitted and stored. As in inpainting, the forward model constitutes an under-determined set of equations. In that case, however, and in particular
when those indirect measurements are subsampled at random, we call this type of inverse problem *Compressed Sensing (CS)* [Candes et al., 2006b; Donoho, 2006]. The term compressed sensing includes in fact many types of under-determined inverse problems. The main distinctive feature of CS is that, roughly speaking, all the measurements are supposed to carry the same amount of information about the target image. In order to achieve this, the idea is to scramble the target. That way, the information about local features of the target (edges, corners, details, texture,...) is spread across all the measurements instead of being grouped in small clusters of pixels. Some sorts of indirect imaging such as tomography or interferometry achieve well this scrambling effect [Lustig et al., 2007; Moshtaghpour et al., 2017; Wiaux et al., 2009]. However, it is also possible to design a compressive device in order to maximize this effect. In Chapter 5 for example, we design an optical setup that scrambles the spatial information of a multispectral target scene before recording compressive measurements on a low resolution\(^2\) sensor.

Finally, sometimes a same target scene can be observed in different modalities, using different sensors. Therefore, it might make sense to exchange information between the modalities in order to improve the imaging performances. Some of the most common applications of multimodal imaging include remote sensing [Ma et al., 2014], biomedical imaging [Fatakdwala and *et al.*, 2013], and high-resolution depth sensing [Diebel and Thrun, 2005]. See also [Castorena et al., 2016; Liu et al., 2016; Wei et al., 2015]. In Chapter 4, we present an application where a high resolution image of light intensity can ease the inpainting associated to a low resolution depth map sensor. This kind of information transfer between modalities is called *sensor fusion*.

\(^2\)Note that we use the term *resolution* to denote the number of pixels (as in *display resolution*) and not the ability to resolve detail (as in *optical resolution*).
2.1.2 Low complexity priors

As we have seen, a criterion is needed in order to discriminate the reconstruction candidates and get the best possible reconstruction $\hat{x}$ that respects the forward model. As a general intuitive guideline, we want to favor simplicity over complexity. A signal coming from the natural world always has some kind of structure. In that regard, a natural signal is *simpler* than a meaningless, unstructured, random noise signal. The main idea behind *regularization*, is to associate to every candidate a cost that represents its complexity. The cost function thus defined is used in the recovery methods presented in Section 2.2 to select the simplest vector. This section introduces the different classical signal models that implement this notion of complexity.

**Sparse representations**

The notions of *sparsity* and *sparse representations* have been around for a long time in computational harmonic analysis and popularized in the 1990s in the field of image compression, see, e.g., [Antonini et al., 1992; Daubechies, 1990] and also [Gribonval and Nielsen, 2006; Mallat, 2009] and references therein. It is the simple idea that a signal can be “well” represented by a few linear parameters, *i.e.*, the linear combination of a few, say $s \ll n$, elementary signals or *atoms* $\psi_i \in \mathbb{R}^n$,

$$x_0 \approx \sum_{i=1}^{s} \alpha_i \psi_i. \quad (2.4)$$

Following [Foucart and Rauhut, 2013], let us start by precisely defining the notion of *sparsity*. We use the shorthand notation $[n] \triangleq \{1, \ldots, n\}$ and for a set $I \subseteq [n]$ we note $|I|$ or $\# I$, its cardinality and we write $I^c \triangleq [n] \setminus I$, its complement.
**Definition 1** (Support, $\ell_0$-“norm”, $s$-sparse vector). The support of a vector $x \in \mathbb{R}^n$ is the set of its nonzero entries, i.e.,

$$\text{supp}(x) \triangleq \{ i \in [n] \mid x_i \neq 0 \}.$$

The $\ell_0$-“norm” of $x$ denotes the cardinality of the support, i.e.,

$$\|x\|_0 \triangleq \# \text{supp}(x). \quad (2.5)$$

Finally, a vector $x$ is called $s$-sparse if

$$\|x\|_0 \leq s.$$

**Remark 1.** The $\ell_0$-“norm$^3$, can be viewed as the limit of the $\ell_\alpha$-(quasi)-norm to the power $\alpha$ when $\alpha \to 0$, i.e.,

$$\|x\|_\alpha^\alpha \triangleq \sum_{i=1}^{n} |x_i|^\alpha \to_{\alpha \to 0} \sum_{i=1}^{n} \delta(x_i \neq 0) = \|x\|_0,$$

where $\delta(B) = 1$ if condition $B$ is true and 0 otherwise.

We define the set of $s$-sparse vectors, e.g., in $\mathbb{R}^n$ (the ambient dimension should always be clear from the context), as

$$\Sigma_s \triangleq \{ x \in \mathbb{R}^n \mid \|x\|_0 \leq s \}.$$

However, in practice, a signal is rarely exactly sparse. This is why we define the weaker notion of *compressibility* which has several possible measures and definitions. A first definition is to call $(s, \varepsilon)$-compressible any vector $x$ such that

$$\# \{ i \in [n] \mid |x_i| \geq \varepsilon \} \leq s.$$

---

$^3$Sometimes it is abusively called the $\ell_0$-norm even though it is not a norm nor even a pseudo-norm because $\|ax\| \neq |a| \|x\|$. See Appendix A of [Foucart and Rauhut, 2013] for precise definitions.
Another possibility is to call a vector, $x$, *compressible* if its entries, ordered by decreasing magnitude as $|x_{j_1}| \geq |x_{j_2}| \geq \cdots \geq |x_{j_n}|$, decay like a power law, *i.e.*, for $1 \leq s \leq n$ there are some constants $c_1, c_2 > 0$ such that

$$|x_{j_s}| \leq c_2 s^{-c_1}.$$  

This definition is equivalent to another one that uses the error associated to the *best $s$-term approximation*.

**Definition 2 (Best $s$-term approximation).** For $\alpha \in \{1, 2\}$, the $\ell_\alpha$-error$^4$ of a best $s$-term approximation $x_s$ to a vector $x \in \mathbb{R}^n$ is defined by

$$\sigma_s(x)_\alpha \triangleq \inf \left\{ \|x - x_s\|_\alpha \mid x_s \in \Sigma_s \right\}. \quad (2.6)$$

Moreover, let $\mathcal{H}_s(\cdot)$ denote the hard thresholding operator which sets all but the $s$ largest entries$^5$ to zero. A best $s$-term approximation $x_s$ satisfying the definition above is given by

$$x_s = \mathcal{H}_s(x).$$

A signal is compressible in the sense defined above if and only if $\sigma_s(x)_2$ follows a decreasing power law of $s$, *i.e.*, for some constant $c_3 > 0$ and the same constant $c_1 > 0$ as above,

$$\sigma_s(x)_2 \leq c_3 s^{-c_1 - 1/2}.$$  

Note that Theorem 2.5 from [Foucart and Rauhut, 2013] tells us that the $\ell_2$-error of a best approximation to any vector $x$ is always bounded by

$$\sigma_s(x)_2 \leq \frac{1}{2} \|x\|_1 s^{-1/2}.$$  

$^4$The definition is sometimes given for any $\alpha > 0$ but the two cases considered here are the only one that are actually used in this thesis.

$^5$If the choice is not unique, *i.e.*, if the input vector has some entries that are identical, we arbitrarily pick any of the equivalent choices.
To link those definitions with (2.4) and what we called sparse representations, we need to define the dictionary $D = (\psi_1, \ldots, \psi_p)$ with $p$ atoms such that $p \geq n \gg s$. It is therefore convenient to group the coefficients $\alpha_i \in \mathbb{R}$ in an $s$-sparse vector $\alpha \in \mathbb{R}^p$. Therefore, (2.4) also reads $x_0 \approx D\alpha$.

The simplest form of dictionaries is when the atoms form an orthonormal basis (ONB), which includes the canonical basis. In that case, we denote by $Q \triangleq D$ the corresponding matrix to emphasize the fact that it is unitary, i.e., $QQ^* = Q^*Q = \text{Id}$. The best $s$-sparse approximation to a vector $x$ in that basis $Q$ is then simply formed with the $s$ largest coefficients in that basis, i.e.,

$$\alpha_s = \arg\min_{\alpha} \{ \| x - Q\alpha \|_2 \mid \alpha \in \Sigma_s \} = \mathcal{H}_s(Q^*x),$$

and of course, $x_s = Q\alpha_s$. The different notions of compressibility generalize nicely as well to any ONB. The two most common examples of (real) orthonormal bases used to expose sparsity (or compressibility) in images are the Discrete Cosine Transform (DCT), used for instance in the old and very popular JPEG compression standard [Bofill and Zibulevsky, 2001] and the Discrete Wavelet Transform (DWT) used in JPEG-2000 [Taubman and Marcellin, 2002]. Both transforms are illustrated on Figure 2.3, where only the 10% biggest coefficients were kept and all the others were set to zero.

When $p > n$, the dictionary $D$ is called redundant. Using redundant dictionaries offers much more flexibility than being restricted to sparsity in an ONB (see e.g., [Candès et al., 2011]). A dictionary that is well adapted to a specific class of signals will logically give better sparse representations than an orthonormal, but suboptimal basis. Therefore, the achievable approximation error,

$$\sigma^D_s(x)_2 \triangleq \inf \{ \| x - D\alpha \|_2 \mid \alpha \in \Sigma_s \},$$
Figure 2.3 The best $s$-sparse approximation in two different ONBs with $s/n = 0.1$. Left column is the DCT. Right column is the DWT with Daubechies-4 wavelets and 5 levels. First row shows the coefficients $\alpha_s = H_s(Q^*x_0)$ in the transformed domains (magnitude in logarithmic scale, black is used for zero-valued coefficients). The second row shows the corresponding reconstructed images in the spatial domain $x_s = Q\alpha_s$. 
will more likely be small. However in that case, hard thresholding does not in general give the best $s$-sparse $\alpha$ and one has to resort to convex optimization or to iterative algorithms such as the ones described in Section 2.2 to find the best sparse representations.

One could argue that there are several classes of redundant dictionaries. A first class is formed by those that come directly from computational harmonic analysis such as oversampled time-frequency representations, e.g., oversampled DCT, Gabor transforms, non-orthogonal wavelets extensions (e.g., ridgelets [Candès, 1998], curvelets [Candès et al., 2001; Donoho and Duncan, 2000; Starck et al., 2002a], shearlets [Candès and Donoho, 2004], contourlets [Do and Vetterli, 2005], etc.) and the redundant or scale-invariant or Undecimated Discrete Wavelet Transform (UDWT) [Starck et al., 2007] or from a signal model such as Gaussian mixtures [Yu et al., 2012]. See also [Mallat, 2009] for an overview on this type sparse representations. For some of these dictionaries, for instance those associated with wavelets, there are efficient algorithms allowing to compute the transform with a low computational complexity (e.g., [Cody, 1992], [Candès et al., 2006a], etc.).

Another class is the unions of orthonormal bases such as, for instance, the Dirac-DCT frame, useful when different types of sparsity are mixed together [Rauhut et al., 2008a]. Note that a dictionary $D \in \mathbb{R}^{n \times p}$ is called a frame of $\mathbb{R}^n$ when there are positive real numbers $a$ and $b$ such that $a \leq b$ and for each $x \in \mathbb{R}^n$,

$$a \|x\|^2 \leq \|D^*x\|^2 \leq b \|x\|^2.$$  

Moreover, when $a = b$, $D$ is called a tight frame (or a Parcival frame if $a = b = 1$) and respects $DD^* = aI_d$.

Yet another class is made of dictionaries that are directly crafted from a physical model of reality. For example, in hyperspectral imaging (HSI), there exist databases that describe typical spectra of light associated to typical materials. This can be used to sparsely represent hyper-pixels
in an hyperspectral image that are a mix of several of these endmembers [Keshava and Mustard, 2002]. This method is sometimes called source separation or unmixing.

Finally, it is also possible to automatically learn a dictionary that is best suited to a certain class of signals. In the example of source separation in HSI, one may not know the spectral database in advance. In that case, the problem of learning it during the reconstruction is called blind source separation [Arberet, 2010; Bofill and Zibulevsky, 2001]. In a larger scope in imaging, spatial sparse representations can be learned. Dictionary learning (and sparse coding) is a whole field of research in itself [Mairal et al., 2014; Olshausen and Field, 1997], to some extent akin to machine learning. The contributions of Chapter 4 belong to this field which is introduced in more details in Section 2.2.5.

**Union of Low-dimensional Subspaces and cosparsity**

Sparse representations as described above, where a signal $x_0$ is synthesized from a combination of a limited number of atoms, is called the synthesis model. From the perspective of linear algebra, the set which contains all the vectors that have a $s$–sparse representation in a given dictionary $D$ is called a Union of Low-dimensional Subspaces (ULS) [Blumensath and Davies, 2009b; Lu and Do, 2008]. Specifically, if we define $\mathcal{I}$ as the set of all supports $I$ of cardinality $s$ and $D_I$ the dictionary restricted to the columns that belong to the support, the ULS is explicitly defined as

$$\Sigma^D_s \triangleq \bigcup_{I \in \mathcal{I}} \text{Im}(D_I),$$

i.e., the union of all image subspaces spanned by any $s$ columns of $D$. The dimension of those subspaces is at most $s$ (exactly $s$ if any $s$ columns of $D$ are linearly independent). It is precisely because $\Sigma^D_s \subset \mathbb{R}^n$ is much smaller than $\mathbb{R}^n$ that we are able to use sparse representations to pick a good signal estimate.
Synthesis sparsity is far from being the only type of ULS. Another popular ULS model is called *analysis* sparsity [Elad et al., 2006] or *cosparsity* [Nam et al., 2013]. The rationale behind it is to count how many atoms from the dictionary have a non-zero scalar product with \( x_0 \). A good, structured, signal candidate \( \hat{x} \) is one who has a small analysis sparsity \( s^* \triangleq \|D^*\hat{x}\|_0 \). This model is of course equivalent to synthesis sparsity as long as the dictionary is an ONB (since in that case \( D^*D\alpha = \alpha \)). For a redundant dictionary however, even for a tight frame, this is no longer the case since \( D^*D \neq \text{Id} \). When using analysis sparsity in the rest of the text (in particular in Chapter 5), we will adopt the lighter notation \( A \triangleq D^* \) to denote the Analysis dictionary. Notice that in that case, the tight frame property becomes \( A^*A = a\text{Id} \). In fact, for analysis sparsity, the ULS is actually described through the atoms that are orthogonal to \( x_0 \). The indices of those atoms form the *cosupport* of \( x_0 \) in \( A \). Specifically, let \( I^c \) be the set of all possible *cosupports* \( I^c \subset [p] \) of cardinality \( \kappa = p - s^* \) and \( A^{I^c} \) the analysis dictionary restricted to the rows that belong to the cosupport. Then we define the \( \kappa \)-cosparse ULS associated to \( A \) as

\[
K^A_\kappa \triangleq \bigcup_{I^c \in I^c} \ker(A^{I^c}),
\]

i.e., the union of all *kernel subspaces* associated to any \( \kappa \) rows of \( A \). Note that the dimension of those subspaces is not the (analysis) sparsity \( s^* \) but is instead derived from the cosparsity \( \kappa \) as \( \dim(\ker(A^{I^c})) \geq n - \kappa \) (equality is reached if any \( \kappa \) rows of \( A \) are linearly independent). Again, the set \( K^A_\kappa \subset \mathbb{R}^n \) is much smaller than \( \mathbb{R}^n \), which justifies the use of analysis sparsity as a signal prior.

Analysis sparsity has been successfully used in many forms and for many inverse problems such as image denoising [Starck et al., 2002a], contrast enhancement [Starck et al., 2003], the so-called *morphological component* (e.g., cartoon and texture) separation [Starck et al., 2004], superresolution [Farsiu et al., 2004], fusion [Kluckner et al., 2010], compressed sensing [Candès et al., 2011] etc. The most popular and broadly
used analysis dictionary is probably⁶ \( A = L \), the discrete gradient (or finite differences) operator, leading to the celebrated TV regularization [Rudin et al., 1992]. Another very effective redundant analysis prior, used in Chapter 5 for spatial regularization is the UDWT, which can somewhat be viewed as a generalization of TV beyond single-level Haar wavelets.

At this point, let introduce the notion of vectors in general direction that will be useful in Chapter 3. We actually borrow the concept of points in general position from algebraic geometry (see, e.g., [Yale, 1968]) and adapt it to the context of dictionaries.

**Definition 3** (Vectors in general direction). A set of \( p \) vectors \( \{ \psi_i \in \mathbb{R}^n \}_{i \in [p]} \) is said to be in general direction if and only if all subsets of \( s \) vectors, for any \( s \leq \min(n, p) \), are linearly independent.

Note that the above definition also allows the number of vectors to be smaller than the dimension of the ambient space, in which case, the corresponding vectors simply are linearly independent. When \( p > n \), however, it is equivalent to requiring that the matrix \( D = (\psi_1, \ldots, \psi_p) \) has a full sparse rank, i.e.,

\[
\text{spark}(D) \triangleq \min_{\alpha \neq 0} \{ \| \alpha \|_0 \text{ s.t. } D\alpha = 0 \} = n.
\]

To obtain the correspondence with the classical definition of general position in geometry, we need to add the origin to the set of points corresponding to the vectors (hence the word direction instead of position in our definition). Another equivalent definition is that a finite set of vectors in \( \mathbb{R}^n \) is in general direction if and only if any subspace of \( \mathbb{R}^n \) of dimension \( s < n \) cannot contain more than \( s \) vectors of the set. Note finally that the concept of general position is also used in [Giryes et al., 2014; Nam et al., 2013] to simplify the study of redundant dictionaries in the cosparse analysis model.

⁶We use \( L \) in reference to the L-shape of the finite differences of order one in two dimensions, i.e., for a pixel of index \((i, j)\), it reads \((x_{i+1,j} - x_{i,j}, x_{i,j+1} - x_{i,j})\).
More low complexity priors

Classical sparse representations (synthesis and analysis) do not impose any assumption regarding the location of the non-zero coefficients (or the zeros in the cosparse model) in the coefficient vector. In some cases however, those non-zero coefficients actually form predictable patterns and are activated in groups. This phenomenon is called group sparsity [Baron et al., 2005; Kowalski, 2009; Tropp, 2006b; Tropp et al., 2006; Yuan and Lin, 2006] and is one example of structured sparsity [Bach et al., 2011; Baraniuk et al., 2010]. Given a partitioning $G = \{g_1, \ldots, g_{n_G}\}$ of $[p]$, in $n_G$ groups $g_i \subset [p]$ of sizes $n_{g_i}$, group sparsity can be defined as

$$
\|\alpha\|_{G,0} \triangleq |\{i \mid \|\alpha_{g_i}\| > 0, g_i \in G\}|,
$$

i.e., the number of groups $g_i$ that have at least one non-zero coefficient in them. For instance, in the case of a multichannel image such as multispectral or hyperspectral images, the different channels usually have a similar sparse support in the same dictionary [Golbabaee and Vandergheynst, 2012]. That specific case, when the groups are non-overlapping, is sometimes called joint sparsity. The isotropic TV regularization,

$$
\|x\|_{TV,iso} \triangleq \sum_{i=1}^{n} \|Lx\|_2,
$$

also uses a joint sparsity prior with the discrete gradient analysis dictionary. Group sparsity with overlap and sparsity with graph structure can also be useful [Jacob et al., 2009]. For example, in a multi-scale representation such as the wavelet transform, the activation of a coefficient in an arbitrary scale often entails the activation of neighbor coefficients in the other scales, in a tree-like structure.

In some cases, e.g., in multichannel images, instead of representing the signal as a big vector, it makes sense to split that vector and stack the pieces in a matrix of similar columns. Similarity, and in particular, linear dependency between columns or between rows, leads to a matrix
that has a low rank. Low rank is also quite a successful signal prior which has originally been introduced in signal processing for matrix completion [Candès and Recht, 2009]. Note that it is also a ULS (with an infinite number of subspaces, see, e.g., [Tirer and Giryes, 2017]). It has been proven useful, in combination with other (e.g., sparsity) priors, for regularizing signals that have a slowly varying component such as a fixed background in a video sequence (e.g., [Otazo et al., 2015; Pairet et al., 2016]) or for high dimensional signals coming from a linear mixture of a limited number of sources, i.e., the source-mixing model, used for instance in Hyperspectral Imaging [Golbabae and Vandergheynst, 2012].

There are, of course, other kinds of low complexity priors in signal processing and this list is not intended to be exhaustive. Let us finish this section by mentioning that some work has been done to give an abstract definition of complexity of a signal from an information theoretic point of view. In order to propose a universal prior, Baron and Duarte [2012] use Kolmogorov’s complexity as a formal definition of the information content of a signal. Even though it is not computationally tractable, it is an elegant way to formally capture the philosophical essence of all those low complexity priors.

2.1.3 Sensing model and embedding

This section focuses on the sensing matrix $\Phi$ and in particular on random matrices and some results developed during the last ten years in the field of compressed sensing. As explained in the book of Foucart and Rauhut [2013], randomness is the key, in the state of the art, to provide rigorous proof that we can sample a signal in an underdetermined way while still being able to recover it stably. A different, yet connected, approach leverages incoherence between measurements, i.e., the fact that because they are not correlated, they all carry about the same amount of useful information.
Random sensing and Restricted Isometry Property

The main idea behind the results presented here is that there are under-determined random sensing matrices that are embeddings of the ULSs introduced in the previous section, i.e., while reducing the dimensionality, they approximately preserve distances between signals from the same ULS. For simplicity, we focus here on $\Sigma_s \triangleq \Sigma_s^{\text{Id}}$, the set of $s$-sparse signals in $\mathbb{R}^n$. The embedding property associated to $\Sigma_s$ is called the Restricted Isometry Property (RIP) [Candès and Tao, 2005].

**Definition 4 (Restricted Isometry Constant).** The Restricted Isometry Constant (RIC) $\delta_s$ of order $s$ of a matrix $\Phi \in \mathbb{R}^{m \times n}$ is the smallest $\delta \geq 0$ such that for all $s$-sparse signal $x \in \Sigma_s$,

$$
(1 - \delta) \|x\|^2 \leq \|\Phi x\|^2 \leq (1 + \delta) \|x\|^2.
$$

Moreover, we say that $\Phi$ satisfies the RIP of order $s$ if $\delta_s < 1$ is small.

Let us note that the RIP of order $2s$ is sufficient to guarantee that if the noiseless system $\Phi x = y$, has an $s$-sparse solution $x_0 \in \Sigma_s$, then it is unique in $\Sigma_s$ even if $m < n$. Indeed, let $x_0$ and $x'_0$ be two $s$-sparse solutions of $\Phi x = y$, then $(x_0 - x'_0) \in \Sigma_{2s}$ and $(1 - \delta_{2s}) \|x_0 - x'_0\|^2 \leq \|\Phi x_0 - \Phi x'_0\|^2 = 0$ and therefore $x_0 = x'_0$.

This property is also very useful to prove stable recovery from under-determined noisy systems (see for instance [Candès et al., 2006b] and Section 2.2). However, it was shown in [Tillmann and Pfetsch, 2014] that computing the RIC of an arbitrary matrix is NP-hard. To this day, it has therefore been necessary to use stochastic matrices and a powerful set of probabilistic tools in order to circumvent this difficulty. The foundations of CS primarily used Gaussian matrices, i.e., random sensing matrices in which each entry is independent and identically distributed (i.i.d.) following a Gaussian distribution $\mathcal{N}(0, \frac{1}{m})$. It was soon generalized to subgaussian distributions (with proper normalization) that are defined as follows:
Definition 5 (subgaussian matrix). A subgaussian matrix $\Phi \in \mathbb{R}^{m \times n}$ is a matrix whose entries $\varphi_{i,j}$ are independent, zero-mean subgaussian random variables with same variance and subgaussian parameters $\beta, \theta > 0$, i.e., for all $(i, j) \in [m] \times [n],$

$$\mathbb{P}(\varphi_{i,j} \geq t) \leq \beta \exp(-\theta t^2) \text{ for all } t > 0.$$  

This includes, but is not restricted to, Bernoulli random matrices and Gaussian random matrices.

In particular it was proved (see e.g., [Foucart and Rauhut, 2013] and references therein) that such matrices, satisfy $\delta_s \leq \delta$ with probability at least $1 - \epsilon$ provided

$$m \geq \frac{c}{\delta^2} \left( s \log \left( \frac{n}{s} \right) + \log \left( \frac{2}{\epsilon} \right) \right),$$

where $c > 0$ is a constant depending only on the subgaussian parameters. Note finally that this statement generalizes well for the RIC associated to matrices $\Phi Q$ where $Q$ is any fixed ONB. This is why we say that subgaussian random matrices are universal.

Random Basis Ensembles and mutual coherence

Building a sensing device which implements a full (sub)Gaussian sensing matrix is not practical. First, it is sometimes simply not possible to implement due to physical or other constraints of the application. When it is possible, in addition to complex hardware, it requires a huge memory or a highly efficient random number generator just to store or generate on the fly the matrix itself. To give an order of magnitude, imagine, e.g., that $x_0$ is a relatively modest 1024 by 1024 image (1M pixels). A dense Gaussian matrix with say $m = n/16$ rows at 32 bits of precisions represents 8GB of data. Moreover, reconstruction always requires to handle the matrix and compute several times (hundreds or
thousands of times) matrix-vector products (see Section 2.2) which can be costly for dense matrices.

These considerations motivate the use of structured sensing matrices that have nice computational properties. A typical example is the random partial Fourier basis. Let $F \in \mathbb{C}^{n \times n}$ be the (2-D for images) DFT\(^7\). The random partial Fourier basis or partial Fourier ensemble is defined as

$$\Phi = R_m F,$$

where $R_m$ is a random restriction operator which selects $m$ rows of $F$. Storing the matrix amounts to store the $m$ restriction indices while applying the matrix to a vector is considerably less costly (using the FFT) than a dense matrix vector product. This matrix also provably satisfies the RIP with probability at least $1 - \epsilon$, at the cost of a higher sample complexity, for some $c > 0$ depending on $\delta$ and $\epsilon$,

$$m \geq cs \log^3(s) \log(n),$$

given by [Cheraghchi et al., 2013] (which improves the original bound from [Candes and Tao, 2006]). Actually, similar results hold when $F$ is replaced by another ONB, $U^*$, provided that the appropriate sparsity basis $Q$ is chosen. For instance, the Hadamard basis and the discrete orthonormal wavelet basis also have a fast transform algorithm but $Q = \text{Id}$ is not the best choice of sparsity basis for these two ONBs. This appeals to the notion of coherence and in particular mutual coherence introduced in [Donoho and Huo, 2001] and defined as follows:

\(^7\)Note that in that case, $\Phi \in \mathbb{C}^{m \times n}$ is complex. For the most part of this thesis, we have restricted the discussion to matrices and vectors defined on $\mathbb{R}$. The Fourier transform and partial Fourier ensembles, as well as the diagonal matrices associated to the Fourier transform of convolution kernels will be the only exceptions for which matrices are defined on $\mathbb{C}$. However, most (if not all) CS results that are stated in this chapter for real matrices actually hold for complex matrices. The few differences that exist in some results are discussed, e.g., in [Foucart and Rauhut, 2013].
2.1 Regularized inverse problems

**Definition 6 (Mutual coherence).** Given an orthonormal sparsity basis $Q = (\psi_1, \ldots, \psi_n)$ and another basis $U = (u_1, \ldots, u_n)$, the mutual coherence between $Q$ and $U$ is the value

$$\mu(U, Q) \triangleq \max_{i,j} |\langle u_i, \psi_j \rangle|.$$ 

Two ONBs, $U$ and $Q$, are compatible, in the sense that it is a good idea to randomly sample the coefficients in $U$ of a vector that is sparse in $Q$, as long as $U$ and $Q$ are *incoherent*, i.e., have a small mutual coherence. Intuitively it means that a vector that is sparse in $Q$ is going to be dense in $U$ (and vice versa) meaning that its information content is going to be spread in all coefficients in $U$. This is linked with the famous Heisenberg’s uncertainty principle as explained by Donoho and Stark [1989] for Fourier and by Elad and Bruckstein [2002] for any pair of bases. As it happens, the DFT basis is maximally incoherent with the canonical basis but, for instance, it is the noiselet transform [Coifman et al., 2001] that is maximally incoherent with the wavelet basis [Mallat, 2009]. Sample complexity for the RIP associated to $\Phi_Q = R_m U^* Q$ expressed as a function of $\mu(U, Q)$ can be found, e.g., in [Jacques and Vandergheynst, 2010] and references therein. A more extensive set of results on random basis ensembles are presented in Chapter 12 of [Foucart and Rauhut, 2013]. Let us note at this point that random Fourier ensembles are probably the biggest success story regarding applications of compressed sensing as it allows to drastically reduce the acquisition time and/or complexity which is a huge advantage for instance in MRI [Lustig et al., 2007] but it also finds lots of useful applications e.g., in interferometry [Auria et al., 2013].

**Spread Spectrum and Variable Density Sampling**

Because the Fourier basis is not universal and because of the uncertainty principle, it has been observed that simply sampling the Fourier space uniformly at random is not always the best strategy when the signal
is not sparse in the canonical basis but, for instance, in a wavelet basis. We can interpret this as being due to a non-uniform distribution of the power spectral density of the class of signals of interest. Many natural images, for instance, have much more energy in the lower frequencies than in the higher ones (as can be observed on the DFT of Figure 2.2). One way to circumvent this is to spread the energy across the Fourier domain by pre-multiplying the signal with a spread spectrum pattern (or chirp sequence) before applying the DFT [Puy et al., 2012]. Another strategy, originally introduced as a heuristic to boost the performances of compressive MRI [Lustig et al., 2007], is to sample the Fourier domain with a non-uniform probability density. It has since been broadly studied by different authors both in theory by [Adcock et al., 2015; Jones et al., 2016; Krahmer and Ward, 2014; Puy et al., 2011] and taking into account application constraints, e.g., by [Boyer et al., 2014; Chauffert et al., 2014]. Note that those theoretical advances allow once again to generalize beyond Fourier, e.g., using the concepts of local coherence or asymptotic incoherence between bases.

Random Convolutions

Another type of structured sensing matrices, introduced by Romberg [2007, 2009] are random partial circulant matrices i.e., random convolutions. It is a well known fact that circulant matrices (and block-circulant matrices if we are talking about 2-D convolutions), i.e., circular convolutions, are diagonalizable in the Fourier basis. Hence, those partial circulant matrices enjoy the following structure

$$\Phi = R_m F^* \Sigma F,$$

where $\Sigma \in \mathbb{C}^{n \times n}$ is a complex diagonal matrix$^8$ and $R_m$ and $F$ are defined above. The computation and memory complexity associated to

$^8$cfr. footnote 7. The diagonal $\text{diag}(\Sigma)$ is assumed to respect the appropriate conjugate symmetry so that its inverse DFT, $F^* \text{diag}(\Sigma) \in \mathbb{R}^n$, is real.
those matrices is therefore similar to partial Fourier bases which makes them interesting for practical applications. Suppose that $\mathbf{R}_m$ is arbitrarily fixed (deterministic) but the convolution kernel, i.e., the inverse DFT of the diagonal of $\Sigma$, is a Gaussian or a Rademacher sequence i.e., a random sequence of $+1$ and $-1$ with equal probability. Then Rauhut et al. [2012] prove that $\frac{1}{\sqrt{m}}\Phi$ satisfies the RIP of order $s$ with probability at least $1 - \epsilon$ as long as, for a universal constant $c > 0$,

$$m \geq \max \left\{ \frac{c}{\delta^2} s \log^2(s) \log^2(n) \log(\frac{1}{\epsilon}), \frac{c}{\delta^2} s^{3/2} \log^{3/2}(n) \right\}.$$

Some applications using this kind of structured matrices have appeared in computational imaging. For instance [Robucci et al., 2010] compute random convolutions in the analog electronic domain and [Jacques et al., 2009] implement the Rademacher convolution directly on the sensor array, using shift registers. Amazingly, it is also possible to realize those random convolutions using light and masks in the optical domain, e.g., using coded aperture [Marcia et al., 2009] or off-focus spatial light modulation [Björklund and Magli, 2013]. This last technique is ported to compressive multispectral imaging and studied in detail in Chapter 5 along with strategies and tradeoffs.

More embeddings

The RIP has been mainly studied for sparse signals in an orthonormal basis. However, It is worth mentioning that a lot of these results have been generalized to other ULS. Let us mention for instance the extensions to redundant dictionaries [Rauhut et al., 2008b], the D-RIP for an analysis prior [Candès et al., 2011], and the rank-RIP [Candès and Plan, 2011; Recht et al., 2010] associated to the low rank ULS.

Recently, stable embedding results have been extended to non-linear acquisition such as 1-bit compressed sensing in which the Binary $\epsilon$-Stable Embedding property [Jacques et al., 2013b] guarantees angle preservation. Even more recently, a Quantized-RIP [Jacques and Cam-
bareri, 2017] has been proved for Quantized Compressed Sensing (QCS) using a dithered quantizer (see, e.g., Boufounos et al. [2015] for a review about QCS).

Finally, as we discuss in Section 2.2, the RIP and associated embedding properties, which hold uniformly over the signal class, are not the only type of tools that can be used to guarantee recovery. In particular, we will see that it is possible to develop slightly weaker results that are non-uniform, in the sense that they are not uniformly valid for all $x_0$ but depend on the assumption that $x_0$ is fixed.

## 2.2 Recovery methods

After having introduced regularized inverse problems we present, in this section, the methods that can be used to solve them. Recovery or reconstruction is naturally formulated as an optimization program and we start by describing what is meant by that.

In order to solve the optimization program in question, we need an algorithm. The first class of algorithms, that we consider next, is associated to non-convex optimization. They divide in two main types: hard thresholding methods and greedy algorithms. As we will see, in that case the quality of an estimate does not really depend on the optimization formulation itself but more on the chosen algorithm and its parameters.

The second class of methods is associated to convex optimization. Before presenting the formulations and guarantees we give some elementary theoretical background in that field. Most notably, in the convex case, algorithms usually converge to the global optimum. For this reason, problem formulations and associated guarantees are decoupled from algorithms in two distinct subsections.
2.2 Recovery Methods

2.2.1 General optimization formulation

The general formulation that we adopt throughout this thesis is a regularized regression of the form

$$\hat{x} \in \text{Argmin}_{x \in \mathbb{R}^n} F(y - \Phi x) + R(x), \quad (\mathcal{P}^R_F)$$

where $F : \mathbb{R}^m \rightarrow \mathbb{R}$ is the data fidelity (or just fidelity) term and $R : \mathbb{R}^n \rightarrow \mathbb{R}$ is the regularization term. Note that both $F$ and $R$ have values in the extended reals $\mathbb{R} = \mathbb{R} \cup \{+\infty\}$ which allows optimization practitioners to elegantly model a constraint as a general penalty functions: To model a constraint set $C$, we use the indicator function defined as

$$\iota_C(x) = \begin{cases} 
0 & \text{if } x \in C \\
+\infty & \text{if } x \notin C \end{cases}.$$  \hspace{1cm} (2.10)

The data fidelity $F$ is chosen according to the noise model (discussed in Section 2.1.1) so that it takes a small value when the reconstruction residual $r(x) \triangleq \Phi x - y$ is small enough and preferably “looks like” the noise $w$, that is, has a similar distribution. The regularization function $R$, is an image of the complexity of $x$ according to the signal prior, e.g., sparsity, introduced in Section 2.1.2. It allows to choose the most appropriate $\hat{x}$ among all the possible $x$ with small fidelity $F$. Note that the capital A in $\text{Argmin}$ and the $\in$ symbol indicate that the solution to $(\mathcal{P}^R_F)$ is not always unique (see for instance [Hiriart-Urruty and Lemaréchal, 2001]), in which case we will content ourselves with one of the solutions that we will consider as equally acceptable as all the others. Moreover, in most cases, an exact solution will not be recovered in a finite time by the optimization algorithms. At best, when the problem is convex, the algorithm will converge to an estimate that is within acceptable tolerance of a true solution. When the problem is not convex however, most often the only algorithmic guarantee is that the estimate is within acceptable tolerance of a stationary point which could well be a saddle point or a lo-
cal minimum, see for instance [Beck and Hallak, 2016]. It must be noted however that some special cases (such as in CS) still warrant recovery of the true solution up to bounded errors even for some non-convex algorithms.

**Uniform and non-uniform recovery guarantees**

Recovery guarantees are at the heart of the research associated to inverse problems and CS in particular. They consist in assessing conditions on the forward model such that the estimate \( \hat{x} \) is close to the actual target vector \( x_0 \). There are two main strategies for proving recovery results. The first one uses the uniform properties of the sensing matrix introduced in Section 2.1.3. Those recovery results were mainly developed in the context of CS. Uniform means that the guarantee holds for all target vector \( x_0 \), once the sensing matrix has been picked at random. Mathematically, we can state uniform results (for some low-complexity set \( \mathcal{X} \)) as:

\[
P(\Phi \text{ is such that } \forall x_0 \in \mathcal{X}, (\hat{x} - x_0) \text{ is small}) \geq 1 - \epsilon
\]

where the probability is taken over the distribution of \( \Phi \). Results using the RIP introduced in Section 2.1.3 are inherently of this kind.

The other class of recovery results is called non-uniform because they require the target vector \( x_0 \) to be fixed. Then, they asses the results on probabilities over \( \Phi \) such as

\[
\text{for any fixed } x_0 \in \mathcal{X}, \quad P(\Phi \text{ is such that } (\hat{x} - x_0) \text{ is small}) \geq 1 - \epsilon.
\]

Those results are mathematically weaker, since they particularize to every signal \( x_0 \), but they usually provide tighter bounds, e.g., on the required number of measurements. Sometimes, they even give explicit quantitative non-probabilistic conditions on \( \Phi \) that depend on \( x_0 \) (and from which probabilistic results can be derived afterward). Results
presented in Chapter 3 are from this last type. Note that this type of results can also be used with deterministic matrices.

2.2.2 Non-convex recovery methods

Non-convex optimization is a very broad and complicated topic (see, e.g., [Attouch et al., 2010]). Solving the optimization program in that case, has to be understood in the weaker sense of finding an estimate \( \hat{x} \) which is sufficiently close to the target \( x_0 \) but not necessarily the global optimum of the optimization program. In particular, the optimization formulation is usually just a guide or an interpretation of the goal of an algorithm. We give here just a tiny glimpse at what seems most relevant to the topic of this thesis regarding non-convex optimization methods for solving inverse problems.

Non-convex \( \ell_0 \) regularization

The simplest regularization function associated to a synthesis sparsity prior, is the \( \ell_0 \)-“norm” (2.5). It is most often encountered in constrained form leading to the non-convex problem

\[
\min_{x \in \mathbb{R}^n} F(y - \Phi x) \quad \text{s.t.} \quad \|x\|_0 \leq s, \tag{2.11}
\]

where \( s \) is an upper bound on the size of the support of the estimate \( \hat{x} \) which should be close to the number of entries \( x_0 \) that are significantly bigger than zero. This amounts to assume that \( x_0 \) is compressible (see Section 2.1.2). Note that in order to simplify the discussion, we have considered here that the dictionary is the canonical basis. If it is not the case, we can always implicitly include it in the sensing operator as

\[
\min_{\alpha \in \mathbb{R}^p} F(y - \Phi D \alpha) \quad \text{s.t.} \quad \|\alpha\|_0 \leq s. \tag{2.12}
\]

In general, solving this problem or any problem involving the \( \ell_0 \)-“norm” implicitly requires to select the best support which is, if one has to
test them all, a combinatorial problem. Solving this problem exactly is therefore known to be NP-hard [Natarajan, 1995]. It has, nevertheless, been broadly studied along with algorithms designed to solve it, i.e., to find estimates that are reasonably close to the solution. It is most often associated with the sum of squares fidelity function \( \frac{1}{2} \| \Phi x - y \|_2^2 \) but some, for instance [Beck and Hallak, 2016], have studied a more general case. Another line of work, e.g., [Nam et al., 2013], has also studied generalization of \( \ell_0 \)-“norm” problems to other ULSs.

In the following, we present two successful families of algorithms, the greedy pursuit algorithms and the hard thresholding algorithms, that have been used to solve \( \ell_0 \)-“norm” problems and their generalizations and we give some guarantees associated with them.

**Greedy algorithms**

The idea behind greedy algorithms is to solve exactly a small optimization problem at each step. Each of these small sub-problems locally improves the estimate but they do not take the “big picture” into account. The goal of these greedy algorithms compared to other methods is clearly to gain in computational efficiency while still providing optimal error guarantees. The firstly introduced algorithms of the sort were Matching Pursuit [Mallat and Zhang, 1993], and Orthogonal Matching Pursuit [Pati et al., 1993] for computing optimal sparse decompositions in non orthogonal dictionaries.

The steps described here are based on the Compressive Sampling Matching Pursuit (CoSaMP) algorithm [Needell and Tropp, 2009] and more precisely on its recent generalization [Tirer and Giryes, 2017] to any union \( \Xi_s \subset \mathbb{R}^n \) of subspaces of dimension \( s \), but all those greedy pursuit algorithms roughly take the same steps. It starts with \( x^0 = 0 \) as the initial estimate and \( \mathcal{X}^0 = \{0\} \) as the initial subspace\(^9\) and iterates as follows. At iteration \( t \), compute a proxy \( a^{[t]} \) of the signal \( x_0 \) by back

\( \footnote{Think of it as the empty support set if you consider the sparse ULS.} \)
projecting the current residual

\[
a^{[t]} \leftarrow \Phi^* (y - \Phi x^{[t]}).
\]

Then, identify the subspace\(^{10}\) \(A^{[t]}\) of dimension \(2s\) closest to \(a^{[t]}\),

\[
A^{[t]} \leftarrow \arg\min_{A \subset \Xi_{2s}} \left\| a^{[t]} - P_A a^{[t]} \right\|_2,
\]

where \(P_A\) is the orthogonal projection on subspace \(A\). Restrict the problem by projecting the rows of \(\Phi\) on the subspace \(\bar{\mathcal{X}}^{[t]} = \mathcal{X}^{[t]} + A^{[t]} \subset \Xi_{3s}\) (Minkowski sum\(^{11}\)) and compute the next signal estimate \(x^{[t+1]}\) by minimizing \(F\) on that subspace\(^{12}\)

\[
\tilde{x}^{[t]} \leftarrow \arg\min_{x \in \bar{\mathcal{X}}^{[t]}} F (y - \Phi P_{\bar{\mathcal{X}}^{[t]}} x),
\]

and projecting the solution on the best subspace \(\mathcal{X}^{[t+1]}\) of dimension \(s\)

\[
\mathcal{X}^{[t+1]} \leftarrow \arg\min_{\mathcal{X} \subset \Xi_s} \left\| \tilde{x}^{[t]} - P_{\mathcal{X}} \tilde{x}^{[t]} \right\|_2, \quad x^{[t+1]} \leftarrow P_{\mathcal{X}^{[t+1]}} \tilde{x}^{[t]}.
\]

As an example, we give uniform recovery guarantees offered by CoSaMP for \(F = \frac{1}{2} \| \cdot \|_2^2\) and \(\Xi_s = \Sigma_s\). It was originally derived by Needell and Tropp [2009] with the condition \(\delta_{4s} < 0.17157\) and then improved in [Foucart and Rauhut, 2013] in the present form.

**Theorem 1** (CoSaMP uniform recovery). Suppose that \(y = \Phi x_0 + w\) where \(\Phi\) satisfies the RIP of order \(8s\) with RIC \(\delta_{8s} \leq 0.4782\). Then at iteration \(t\), the algorithm CoSaMP, produces a \(2s\)-sparse approximation \(x^{[t]}\) that satisfies

\[
\left\| x_0 - x^{[t]} \right\|_2 \leq \frac{c_1}{\sqrt{s}} \sigma_s(x_0)_1 + c_2 \| w \|_2 + 2 \rho^t \| x_0 \|_2,
\]

\(^{10}\)for sparsity, this amounts to find the \(2s\) largest entries.

\(^{11}\)which translates to the union of supports in the case of sparsity.

\(^{12}\)if \(F = \frac{1}{2} \| \cdot \|_2^2\), this amounts to compute the pseudo-inverse as in \(\tilde{x}^{[t]} \leftarrow (\Phi P_{\bar{\mathcal{X}}^{[t]}})^+ y\).
for some constant $c_1, c_2 > 0$ and $0 < \rho < 1$ depending only on $\delta_{8s}$ and where $\sigma_s(x_0)_1$ is the $\ell_1$-error of the best $s$-sparse approximation to $x_0$ as defined by (2.6). In particular, the sequence of $x^{[t]}$ converges to $\hat{x}$ satisfying

$$\|x_0 - \hat{x}\|_2 \leq \frac{c_1}{\sqrt{s}}\sigma_s(x_0)_1 + c_2 \|w\|_2.$$  (2.13)

This bound is really interesting and representative of most uniform stability results obtained for sparse recovery methods. The first term is associated to the best $s$-sparse approximation error, measured in $\ell_1$ norm. This importantly means that the algorithm is stable to inexact sparsity (i.e., compressibility). The second term is associated to the additive noise $w$, which means that the method is robust. Note that the second bound is valid not only for $t \to \infty$ but also after a sufficiently large number of iterations, as soon as $2\rho^t \|x_0\|_2$ is small enough to be absorbed in constants $c_1$ and $c_2$.

Note that computing the optimal subspaces $A^{[t]}$ and $X^{[t]}$ in the generalized version of CoSaMP presented here is not always as easy as in the case of sparsity in an orthonormal basis. Therefore, for instance for synthesis sparsity in a redundant dictionary, Davenport et al. [2013] propose to use approximate projections instead. This idea has been reused by other, e.g., in [Giryes et al., 2014] for the cosparse analysis model. In both cases, they still recover comparable guarantees despite the approximation.

**Hard thresholding algorithms**

The second class of non convex recovery methods is the hard thresholding algorithms\footnote{Note that hard thresholding algorithms are also sometimes called greedy. In a sense, they are a subclass of the above.}. The iterations of these algorithms are somewhat simpler. They consist in a first gradient descent step on the fidelity cost function, with a fixed or variable (see, e.g., [Blumensath and Davies, 2008]).
step-size $\mu_t > 0$:

$$a^{[t]} \leftarrow x^{[t]} + \mu^{[t]} \Phi^* \nabla F(y - \Phi x^{[t]}).$$

The second step implicates a support selection through hard thresholding, i.e., setting all but the largest entries to zero. Iterative Hard Thresholding (IHT) [Blumensath and Davies, 2009a] uses the best $s$-term approximation to $a^{[t]}$,

$$x^{[t]} \leftarrow \mathcal{H}_s(a^{[t]}).$$

At the cost of an increased complexity, similarly to greedy algorithms, Hard Thresholding Pursuit (HTP) [Foucart, 2011] replaces the hard thresholding step by fidelity optimization reduced to the support of the best $s$-term approximation

$$x^{[t]} \leftarrow \arg\min_{x \in \mathbb{R}^n} \left\{ F(y - \Phi x) \text{ s.t. } \text{supp}(x) = \mathcal{H}_s(a^{[t]}) \right\}.$$

For the classical $\ell_2$-$\ell_0$ case, both algorithms also provide uniform guarantees. The result was derived with the RIP in [Blumensath and Davies, 2009a] for the first time for IHT and then improved and unified with HTP in [Foucart, 2011].

**Theorem 2** (IHT and HTP uniform recovery). Suppose that $y = \Phi x_0 + w$ where $\Phi$ satisfies the RIP of order $6s$ with RIC $\delta_{6s} \leq 1/\sqrt{3}$. Then at iteration $t$, IHT or HTP (where the sparsity bound is set to $2s$ instead of $s$) produce a $2s$-sparse approximation $x^{[t]}$ that satisfies

$$\left\| x_0 - x^{[t]} \right\|_2 \leq \frac{c_1}{\sqrt{s}} \sigma_s(x_0)_1 + c_2 \|w\|_2 + 2\rho^t \|x_0\|_2,$$

for some constant $c_1, c_2 > 0$ and $0 < \rho < 1$ depending only on $\delta_{6s}$ and where $\sigma_s(x_0)_1$ is the $\ell_1$-error of the best $s$-sparse approximation to $x_0$ as defined by (2.6). In particular, the sequence of $x^{[t]}$ converges to $\hat{x}$ satisfying

$$\|x_0 - \hat{x}\|_2 \leq \frac{c_1}{\sqrt{s}} \sigma_s(x_0)_1 + c_2 \|w\|_2.$$
Note that we find exactly the same result as for CoSaMP, except for the different requirement on the RIC.

Similar algorithms have been proposed and studied for other forms of ULS such as synthesis sparsity in a redundant dictionary Giryes and Elad [2013] and analysis cosparsity Giryes et al. [2014]. For these versions, similar guarantees are usually given, by assuming the corresponding RIP-like property holds. However, they require an efficient method to compute (at least approximately) projections on the ULS, which is not always easy as mentioned above. To alleviate this difficulty, we used in [Degraux et al., 2015] a technique\footnote{The technique was inaccurately coined Pseudo-inverse IHT. It was realized afterwards that the inverse UDWT is in general not actually the Moore-Penrose pseudo-inverse of the direct transform. What was called the pseudo-inverse $A^\dagger$ of the analysis frame $A$ (which was formed with the UDWT) is actually not. Indeed, $AA^\dagger$ is not Hermitian and therefore $A^\dagger \neq (A^*A)^{-1}A^* \equiv A^+$. It is nevertheless a left inverse.} that we shall call Iterative Hard DeNoising (IHDN) to obtain analysis sparsity in a UDWT-based redundant frame $A$. Instead of projecting on the cosparselike ULS, we used hard thresholding of the analysis coefficients $\mathcal{H}_s(A^\dagger a[t])$, followed by the inverse transform $A^\dagger$ (which is a left inverse of $A$). If you ignore the multi-channel aspects described in the paper and just consider that $A$ is the UDWT, this hard thresholding step is known as cycle spinning or translation invariant wavelet hard denoising [Mallat, 2009]. We will call IHDN, any such algorithm which uses a hard denoising method instead of simple hard thresholding. Note that, as far as we know, uniform guarantees do not exist yet for IHDN in general.

In the same vein, let us mention that hard thresholding algorithms have been used to solve non-linear inverse problems such as 1-bit compressed sensing with Binary IHT [Jacques et al., 2013b], signal declipping with Consistent IHT [Kitic et al., 2013], and quantized compressed sensing with Quantized IHT [Degraux, 2013; Jacques et al., 2013a].
2.2.3 Convex recovery methods

In this section, we present the most classical convex formulations for sparse recovery and the like. Algorithms that can be used to solve them are presented in the next section and a brief theoretical background on convex optimization can be found in Appendix A. Since, in convex optimization, those algorithms are designed to converge with arbitrary precision\(^{15}\) to an exact solution of the problem they tackle, we can disregard the choice of the algorithm for now and just consider the exact solutions of the optimization program when giving recovery guarantees.

### \(\ell_1\) minimization and uniform guarantee

We have seen that greedy or thresholding based methods are good methods for direct \(\ell_0\)-"norm" minimization in order to recover sparse solutions of an underdetermined linear system \(y_0 = \Phi x_0\). A very popular and extensively studied alternative is \(\ell_1\)-norm minimization coined the Basis Pursuit (BP) problem [Chen et al., 1998]

\[
\min_x \|x\|_1 \quad \text{s.t.} \quad \Phi x = y_0.
\]

(2.14)

It can be viewed as the convex relaxation of the \(\ell_0\)-"norm" minimization problem [Tropp, 2006a]. BP is known to have sparse solutions but assumes that the observations \(y_0\) are exact. For tackling noisy systems of equations, \(y = \Phi x_0 + w\), Chen et al. [1998] also propose to use a quadratic penalty in what they call Basis Pursuit DeNoising (BPDN),

\[
\min_x \frac{1}{2} \|\Phi x - y\|_2^2 + \lambda \|x\|_1.
\]

(2.15)

---

\(^{15}\)Let it be noted that it has recently been observed in [Ben-Artzi et al., 2015] that in finite precision arithmetic, this is in fact not always true, even in convex optimization for solving inverse problems. We will ignore this in the present dissertation.
Since the seminal work of Candès et al. [2006b] in the context of CS, the BPDN problem has been often used in its constrained form

$$\min_{\mathbf{x}} \|\mathbf{x}\|_1 \quad \text{s.t.} \quad \|\Phi\mathbf{x} - \mathbf{y}\|_2 \leq \tau,$$  \hspace{1cm} (2.16)

while the lagrangian form (2.15) is often referred to as the Least absolute shrinkage and selection operator (Lasso). The term Lasso was introduced earlier by Tibshirani [1995] to refer to the $\ell_1$-constrained form

$$\min_{\mathbf{x}} \frac{1}{2} \|\Phi\mathbf{x} - \mathbf{y}\|_2^2 \quad \text{s.t.} \quad \|\mathbf{x}\|_1 \leq \zeta.$$  \hspace{1cm} (2.17)

In any case, all three of those formulations are equivalent in the sense that even though, there is no known explicit relationship, there provably exists continuous bijections between the parameters $\tau$, $\lambda$ and $\zeta$, at least as long as they are in some range of interest (for instance there exists a $\lambda_{\text{min}}$ such that for all $0 < \lambda \leq \lambda_{\text{min}}$, the solution to (2.15) is the same as the solution to (2.14), i.e., (2.16) with $\tau = 0$). Precisely, $\lambda$ is related to the Lagrange multiplier associated to the constraint in (2.17) and to the reciprocal of the multiplier of the constraint in (2.16). The constrained form (2.16) of this classical optimization program is depicted on Figure 2.4.

**Figure 2.4** A schematic view of BPDN in its constrained form (2.16) in $\mathbb{R}^3$. 
2.2 Recovery methods

Traditionally, (2.15) and (2.17) are considered easier to solve because they directly match the canonical form of quadratic programming, for which there is a plethora of efficient algorithms. However, if \( \tau \) in (2.16) has an easy interpretation in terms of noise level, \( \zeta \) is a priori harder to estimate and \( \lambda \) lacks a quantitatively interpretable meaning. In order to cope with this, van den Berg and Friedlander [2009] proposed to recursively solve (2.17) to find the \( \zeta \) that corresponds to the targeted noise level \( \tau \) with a Newton-based root-finding procedure. This method is used in the excellent spgl1 software package for Matlab [van den Berg and Friedlander, 2007] which is suited for large scale optimization.

The basic ingredient to guarantee stable and robust recovery of a sparse vector with BPDN is called the robust null space property [Foucart and Rauhut, 2013].

**Definition 7 (Robust null space property).** The matrix \( \Phi \in \mathbb{R}^{m \times n} \) is said to satisfy the robust null space property (RNSP) with constants \( 0 < \rho < 1 \) and \( \eta > 0 \) relative to a support \( I \subset [n] \) if for all \( v \in \mathbb{R}^n \)

\[
\|v^I\|_1 \leq \rho \|v^{I^c}\|_1 + \eta \|\Phi v\|_2,
\]

where \( v^I \) and \( v^{I^c} \) are the complementary parts of \( v \) respectively in the support \( I \) and its complementary \( I^c \). It particularizes to the null space property (NSP) if one only considers \( v \in \text{Ker}(\Phi) \setminus \{0\} \) (i.e., for which \( \|\Phi v\|_2 = 0 \)) and only requires

\[
\|v^I\|_1 < \|v^{I^c}\|_1.
\]

The NSP is a necessary and sufficient condition for any vector \( x_0 \) with support \( I \) to be solution of BP (2.14) with \( y_0 = \Phi x_0 \). A non-uniform version (i.e., which only holds for a fixed vector \( x_0 \)) called identifiability and associated to (A.6) is used in Chapter 3. On the other hand, when the NSP or the RNSP hold for all support \( I \) of cardinality \( |I| \leq s \), they are called NSP or RNSP of order \( s \). The following results (Theorems 6.13 and 4.22 from [Foucart and Rauhut, 2013]) state that the RIP guarantees the RNSP which in turn guarantees robust recovery.
Theorem 3 (RIP and RNSP). Suppose that the RIC of order $2s$ of the matrix $\Phi \in \mathbb{R}^{m \times n}$ satisfies
$$ \delta_{2s} < \frac{4}{\sqrt{41}} \approx 0.6246 $$
then $\Phi$ satisfies the RNSP of order $s$.

Theorem 4 (BPDN uniform recovery). Suppose that $\Phi$ satisfies the RNSP of order $s$. Then for any $x_0 \in \mathbb{R}^n$, a solution $\hat{x}$ of (2.16) with $y = \Phi x_0 + w$, and $\|w\|_2 \leq \tau$, for some constants $c_1, c_2 > 0$ approximates the vector $x_0$ with error
$$ \|x_0 - \hat{x}\|_2 \leq \frac{c_1}{\sqrt{s}} \sigma_s(x_0)_1 + c_2 \|w\|_2. $$

We recognize the exact same bound as for CoSaMP and IHT but the requirements on the RIC are much weaker in this case. This suggests that $\ell_1$-minimization is in fact more powerful than greedy methods. Note also that even though they are not explicitly given here, the constants $c_1$ and $c_2$ are different for every case and can be found in the cited references.

Analysis $\ell_1$ and other convex relaxations

For recovery using analysis based regularization with redundant analysis dictionaries, Candès et al. [2011] proposed the analysis-$\ell_1$ problem
\[
\min_x \|A x\|_1 \quad \text{s.t.} \quad \|\Phi x - y\|_2 \leq \tau, \tag{2.18}
\]
sometimes called Analysis-BPDN. They provided similar uniform recovery guarantees as for BPDN by using the D-RIP. This analysis-$\ell_1$ problem (2.18) is used in Chapter 5 (with an additional convex constraint). Let us also mention the penalized form
\[
\min_x \frac{1}{2} \|\Phi x - y\|_2^2 + \lambda \|A x\|_1, \tag{2.19}
\]
often coined generalized Lasso, which has the same relationship with (2.18) as (2.15) with (2.16).
There also exists other types of convex relaxations associated to some of the priors listed in Section 2.1.2. For minimizing the rank of a matrix, Fazel [2002] proposed to use the nuclear norm also called trace norm, \( \text{i.e.,} \) the sum of all singular values. Uniform guarantees were given in [Fazel et al., 2008]. Baraniuk et al. [2010] proposed a series of convex relaxations for model-based CS which defines a very general form of structured sparsity. It is also possible to combine several of those convex cost functions, see \( \text{e.g.,} \) [Golbabaee and Vandergheynst, 2012; Starck et al., 2010]. Let us finally mention that it is always possible to add more constraints on the signal we want to recover. For instance, if we know the dynamic range of the pixels of a target image, we can constrain the optimization problem to that range.

**Sparsistency and model identification**

In addition to uniform results stated above, a line of work has developed non-uniform guarantees for convex recovery of regularized inverse problems. In particular, by using notions like the dual certificates (introduced in Appendix A), it has allowed to obtain tight conditions for sparse support recovery under small noise and non-uniform stability under arbitrary noise. Those results are non-uniform and deterministic, \( \text{i.e.,} \) they assume that \( x_0 \) and \( \Phi \) are fixed and that \( x_0 \) is sparse with support \( \text{supp}(x_0) = I \). The exact support stability (also called sparsistency) of (2.15), \( \text{i.e.,} \) when the solutions \( \hat{x} \) have the same support \( \text{supp}(\hat{x}) = I \), has been proved by several authors. In the signal processing literature, this result can be traced back to the early work of Fuchs [2004]. In the statistics literature, sparsistency is also proved in [Zhao and Yu, 2006] in the case where \( \Phi \) is random, the result of support stability being then claimed with high probability.

A condition, often coined the irrepresentable condition in the statistics and machine learning literature guarantees exact support stability and is associated to the existence of a non-degenerate dual certificate. These results have been extended recently in [Duval and Peyré, 2015]
to the case where the support $I$ is not strictly stable, i.e. $I \not\subseteq \text{supp}(\hat{x})$. The contribution of Chapter 3 extends further those results to cases where the fidelity $F$ is more general than the Euclidean penalty. Note that sparsistency-like results have been proved for many low-complexity regularizers beyond the $\ell_1$-norm. Let us quote among others: analysis sparsity [Vaiter et al., 2013], the group-lasso [Bach, 2008a], the nuclear norm [Bach, 2008b], the total variation [Vaiter et al., 2013] and a very general class of partly-smooth regularizers [Vaiter et al., 2014]. In those cases, instead of sparse support recovery, we talk about the more general notion of model identification.

2.2.4 Algorithms for convex optimization

In this section, we present two generic algorithms that are used in Chapters 4 and 5 of this thesis to solve some of the convex optimization problems presented above. Both algorithms use proximal splitting and proximal operators defined by (A.1).

**Forward-backward algorithms**

The first and simplest algorithm is called Forward-Backward (FB) algorithm, also known as proximal gradient method which, according to [Parikh and Boyd, 2013], was first mentioned in [Bruck Jr., 1975]. Consider the problem

$$\min_{x \in \mathbb{R}^n} f(x) + g(x)$$

where $f, g \in \Gamma_0(\mathbb{R}^n)$ and $f$ is differentiable. The Forward-Backward iteration is

$$x^{[t+1]} \leftarrow \text{prox}_{\mu[t]g}(x^{[t]} - \mu[t]\nabla f(x^{[t]})),$$

where $\mu^{[t]} > 0$, is the step size that is either fixed or chosen by line search [Beck and Teboulle, 2010]. Note that if $g(x) = \lambda \|x\|_1$, we have $\text{prox}_{\mu[t]g}(v) = S_\lambda(v)$, i.e., the soft thresholding or shrinkage operator...
Algorithm 1 Monotone FISTA [Beck and Teboulle, 2009b]

1: procedure MFISTA
2: Input: $x^{[0]}$, $\mu \leq \frac{2}{L(F)}$
3: $v^{[1]} \leftarrow x^{[0]}$
4: while the stopping criterion is not met, do
5: $u^{[t]} \leftarrow \text{prox}_{\mu R}(v^{[t]} + \mu \Phi^* \nabla F(y - \Phi v^{[t]}))$
6: $\omega^{[t+1]} \leftarrow \frac{1+\sqrt{1+4(\omega^{[t]})^2}}{2}$
7: $x^{[t]} \leftarrow \arg\min_{x \in \{u^{[t]}, x^{[t-1]}\}} F(y - \Phi x) + R(x)$
8: $v^{[t+1]} \leftarrow x^{[t]} + \frac{\omega^{[t]}}{\omega^{[t+1]}}(u^{[t]} - x^{[t]}) + \frac{\omega^{[t]-1}}{\omega^{[t+1]}}(x^{[t]} - x^{[t-1]})$

defined by (A.4). The algorithm is then called Iterative Soft Thresholding Algorithm (ISTA) [Daubechies et al., 2004].

Interestingly, when applied to the non-convex $\ell_0$-constrained problem, i.e., $g(x) = \iota_{\|\cdot\|_0 \leq s}(x)$, FB gives rise to the Iterative Hard Thresholding algorithm described in Section 2.2.2. Indeed, by extending the definition of proximal operators to non-convex functions, we find\footnote{Note that as $H_s(\cdot)$, the prox of a non-convex function can be set valued.}

$$\text{prox}_{\iota_{\|\cdot\|_0 \leq s}}(v) = \pi_{\Sigma_s}(v) = H_s(v),$$

and FB becomes exactly IHT. Of course, the convergence proof that is valid for convex problems no longer holds in that case.

There also exist several accelerated versions of FB, in the sense that they have a better worst-case convergence rate than the original method [Nesterov, 1983]. The general accelerated FB method reads

$$v^{[t+1]} \leftarrow x^{[t]} + \omega^{[t]}(x^{[t]} - x^{[t-1]}),$$
$$x^{[t+1]} \leftarrow \text{prox}_{\mu^{[t]} g}(v^{[t+1]} - \mu^{[t]} \nabla f(v^{[t+1]})),$$

where $0 \leq \omega^{[t]} < 1$ is the over-relaxation parameter. One of these accelerated versions, first applied to $\ell_1$-minimization, was proposed by Beck and Teboulle [2009a] and named Fast Iterative Shrinkage Thresholding
Algorithm (FISTA). They particularized it to solve the TV regularization problem in [Beck and Teboulle, 2009b] and added a simple check to boost the performances, which lead to the Monotone FISTA (MFISTA). This last version of FB is used in Chapter 4 to solve the “sparse coding step” of the dictionary learning algorithm. In Algorithm 1, we describe MFISTA applied to the problem (\(P^R_F\)) where \(F\) is assumed differentiable and \(L(F)\) is the Lipschitz constant of \(\nabla F\).

Finally, let us mention that FB was generalized to minimize the sum of an arbitrary number of functions under the name Genearlized Forward-Backward [Raguet et al., 2013].

ADMM

The Alternating Directions Method-of-Multipliers (ADMM) was originally proposed by [Gabay and Mercier, 1976; Glowinski and Marroco, 1975]. In its general formulation, it is designed to solve problems in the form

\[
\min_{x \in \mathbb{R}^n, v \in \mathbb{R}^m} f(x) + g(v) \quad \text{s.t.} \quad Hx + Gv = b
\]

where \(f \in \Gamma_0(\mathbb{R}^n), g \in \Gamma_0(\mathbb{R}^p), \ b \in \mathbb{R}^m\) is fixed, \(H \in \mathbb{R}^{m \times n}\) and \(G \in \mathbb{R}^{m \times p}\) are linear operators. We define the augmented Lagrangian function

\[
L_\rho(x, v, p) \triangleq f(x) + g(v) + p^*(Hx + Gv - b) + \frac{\rho}{2} \|Hx + Gv - b\|_2^2,
\]

where \(p \in \mathbb{R}^m\) is called the dual variable or the vector of Lagrange multipliers. For some fixed step size parameter \(\rho > 0\), the algorithm consists of the iterations

\[
x^{[t+1]} \leftarrow \arg\min_x L_\rho(x, v^{[t]}, p^{[t]})
\]

\[
v^{[t+1]} \leftarrow \arg\min_v L_\rho(x^{[t+1]}, v, p^{[t]})
\]

\[
p^{[t+1]} \leftarrow p^{[t]} + \rho(Hx^{[t+1]} + Gv^{[t+1]} - b)
\]
Algorithm 2 ADMM as recast in [Almeida and Figueiredo, 2013]

1: procedure ADMM-2
2: Input: $x^{[0]}, \mu_1, \ldots, \mu_J > 0, v^{[0]}, u^{[0]}$
3: while the stopping criterion is not met, do
4: for $j = 1$ to $J$ do
5: $\zeta^{[t]}_j \leftarrow H_j x^{[t]} - u^{[t]}_j$
6: $v^{[t+1]}_j \leftarrow \text{prox}_{g_j/\mu_j}(\zeta^{[t]}_j)$
7: $u^{[t+1]}_j \leftarrow v^{[t+1]}_j - \zeta^{[t]}_j$
8: $x^{[t+1]} \leftarrow \left(\sum_{j=1}^J \mu_j H_j^* H_j\right)^{-1} \sum_{j=1}^J \mu_j H_j^* (v_j^{[t+1]} + u_j^{[t+1]})$

See [Boyd, 2010] for a complete review of this algorithm with interesting reformulations and practical tips for its use in a wide variety of problems. For the class of problems we are interested in, a convenient form can be found in [Almeida and Figueiredo, 2013]. It is tailored to handle an arbitrary sum of composed functions, i.e., problems of the form

$$\min_{x \in \mathbb{R}^n} \sum_{j=1}^J g_j(H_j x),$$

where $g_j \in \Gamma_0(\mathbb{R}^{m_j})$ and where $H_j \in \mathbb{R}^{m_j \times n}$ are linear operators. The reformulation of ADMM is coined ADMM-2 and is given here as Algorithm 2 with a slight reordering of the steps. This algorithm is guaranteed to converge for any step sizes $\mu_j > 0$ as long as (2.20) has a solution and provided that the matrix $G \triangleq (H_1^*, \ldots, H_J^*)^T$ has full column rank. This algorithm appears to be particularly useful if one has an efficient method to compute the inverse of

$$\sum_{j=1}^J \mu_j H_j^* H_j,$$

and an efficient method to compute $\text{prox}_{g_j/\mu_j}(\cdot)$ for every term $g_j$ of the objective function.
Note that the variable \( u^{[t]} \) in Algorithm 2 corresponds to the rescaled dual variable \( p \) of the original formulation. One has to take that into account when using the method described in §3.4.1 of [Boyd, 2010] to use an adaptive stepsize. As for FB algorithms, an over-relaxation is possible as described in §3.4.1 of the same reference. In the case of Algorithm 2, this translates by replacing the quantity \( H_j x^{[t]} \) (for all \( j \)) by

\[
\omega^{[t]} H_j x^{[t]} + (1 - \omega^{[t]}) v_j^{[t]}
\]

with \( 0 < \omega^{[t]} < 2 \) the relaxation parameter (values \( \omega^{[t]} \in [1.5, 1.8] \) can improve convergence, see [Boyd, 2010] and references therein).

Algorithm 2 is used in Chapter 5 to efficiently solve a challenging deconvolution problem regularized with analysis sparsity in a rescaled tight frame and an additional range constraint.

**More proximal methods**

We can cite a few other proximal methods that are not used in this dissertation but are closely connected. Let us mention the Douglas-Rachford splitting algorithm [Douglas and Rachford, 1956; Eckstein and Bertsekas, 1992; Lions and Mercier, 1979], see also [Combettes and Pesquet, 2007], which can be viewed as a particular case of ADMM but was introduced earlier.

Another quite useful algorithm is the Chambolle-Pock (CP) algorithm [Chambolle and Pock, 2010]. CP is actually formulated as a solver for the generic saddle-point problem \((S^R_F)\) where both \( F \) and \( R \) can be non-differentiable\(^{17}\). The big advantage of that algorithm is that when a function \( f \) composed with a linear operator \( H \), i.e., \( f(Hx) \), appears in the objective, it does not require to compute the proximal operator associated to \( f(Hx) \) but only to \( f(v) \). Moreover, as opposed to Algorithm 2, it does not require from \( H \) anything else than being able to compute \( Hx \) and \( H^* v \) for any input vectors \( x \) and \( v \). It is an advantage

\(^{17}\)The formulation in CP’s paper is equivalent to set \( y = 0 \), without loss of generality.
when the operator $H$ is not a tight frame and more generally when it is hard to invert. Like FISTA with respect to FB, CP uses over-relaxation to accelerate the theoretical convergence of an old algorithm called Arrow-Urwicz [Arrow et al., 1958]. Similarly to ADMM-2 and GFB, CP was generalized to a sum of an arbitrary number of functions in [González et al., 2014].

CP was used in [Degraux et al., 2015] where it is numerically compared against IHDN on an hyperspectral imaging problem. However, in Chapter 5 of this thesis, where we study a similar problem, we discovered that it is in fact more efficient to use the structure of $\Phi$ and $A$ and ADMM.

We can also cite PPXA [Combettes and Pesquet, 2011] which is useful to distribute the computations if the objective function can be split in many terms. The reader can also refer to the monograph [Parikh and Boyd, 2013] and references therein for a more extensive review of proximal methods.

### 2.2.5 Dictionary Learning

This final section is dedicated to dictionary learning, which is the main subject of the contribution presented in Chapter 4. As explained in Section 2.1.2, a significant part the research associated to inverse problems and computational harmonic analysis is dedicated to design the best dictionary $D \in \mathbb{R}^{p \times n}$ for finding sparse representations of a given class of signals. In their early work in neuroscience, Olshausen and Field [1996, 1997] have proposed a method that achieves this automatically. Dictionary learning has gained in popularity when the K-SVD algorithm was introduced [Aharon et al., 2006]. Since then, it has become a standard tool for various tasks in image processing such as, e.g., image restoration [Mairal et al., 2009], super-resolution [Yang et al., 2012], multispectral image fusion [Wei et al., 2015] and phase retrieval [Tillmann et al., 2016]. We begin by formalizing the problem of learning a dictionary $D$ that aims at providing good sparse representations for signals coming from
a certain dataset. In particular, we explain the classical block-based approach and confront it with the convolutional approach that is used in Chapter 4. Finally, we explain the concept of online learning and the original algorithm proposed by Mairal et al. [2009]. Note that dictionary learning is a very broad topic and we only cover part of it here. More details and references can be found for instance in the review of Mairal et al. [2014].

Dictionary learning formulation

The traditional approach formulates the problem as follows. Given a dataset of \( N \) training samples, \( \{x_1, \ldots, x_N\} \), find a dictionary \( D \) such that, on average, sparse representations of the samples in that dictionary give a small approximation error, i.e.,

\[
\min_{D \in \mathcal{D}} \sum_{i=1}^{N} \frac{1}{2} \|x_i - D\alpha_i\|_2^2 + R(\alpha_i),
\]

where the set \( \mathcal{D} \subset \mathbb{R}^{p \times n} \) is constraining the dictionary, e.g., to have atoms in the unit ball, the \( \alpha_i \in \mathbb{R}^p \) are often called the codes associated to the samples \( x_i \) and \( R \) is a regularizer, e.g., a sparsity inducing cost function such as the \( \ell_1 \)-norm.

Because both \( D \) and \( \alpha_i \) are optimized jointly, the problem is not convex, even if \( R \) is chosen to be a convex regularizer. However, when either variable is fixed and we optimize over the other one, the sub-problem becomes convex (provided \( R \) and the set \( \mathcal{D} \) are both convex). This type of problems is called bi-convex and it finds many applications beyond dictionary learning such as matrix factorization\(^\text{18}\) [Paatero and

\(^{18}\)Note that, using the notations \( X \triangleq (x_1, \ldots, x_N) \in \mathbb{R}^{n \times N} \) and \( A \triangleq (\alpha_1, \ldots, \alpha_N) \in \mathbb{R}^{p \times N} \) (2.21) is equivalent to the following matrix factorization problem,

\[
\min_{D \in \mathcal{D}} \min_{A} \frac{1}{2} \|X - DA\|_2^2 + R'(A),
\]
2.2 Recovery methods

Tapper, 1994], blind deconvolution [Ayers and Dainty, 1988; Bell and Sejnowski, 1995], blind calibration [Bilen et al., 2013; Cambareri and Jacques, 2016; Ling and Strohmer, 2015]. One classical method to tackle bi-convex problems is called alternate minimization which precisely consists in solving alternatively each convex problem by fixing the other variable to the previously found solution. In the context of dictionary learning, the steps are often called dictionary update when optimizing with respect to $D$, and sparse coding step when optimizing with respect to the codes $\alpha_i$.

**Block-based and convolutional dictionary learning**

Importantly, because we optimize over every entry of $D \in \mathbb{R}^{n \times p}$, it is important to limit the size $n$ and number $p$ of the atoms in the dictionary for the method to be tractable. Especially considering that the number $N$ of training samples should in principle be much bigger than the number of atoms $p$. For this reason, when the dictionary is trained on images, the latter are usually partitioned in patches of small size (typically on the order of $10 \times 10$ pixels) so as to reduce the computational load of the method and increase the number of samples. This is referred to as patch-based dictionary learning.

Patch-based atoms have several limitations. Firstly, they are not translation invariant. The same feature appearing in two patches but at a slightly shifted locations with respect to the center of the patch will have to be represented by two different atoms. Another problem is that they are inherently local. The optimization formulation (2.21) does not take into account the locations of the different patches relative to one another nor even the fact that they possibly come from the same image. Another issue arises once the dictionary has been learned, when actually using it to regularize an inverse problem. It is then necessary to break down the target image in patches, do the reconstruction of

$$R'(A) = \sum_{i=1}^{N} R(\alpha_i).$$
every patch individually and recombine the reconstructed patches at the end. Even if this potentially allows for massive parallelization of the computations, this can have several negative effects on the quality and speed of the reconstruction. In order to avoid blocking artifacts, it is actually advisable to use overlapping patches and the recombination is usually done by computing the mean of the overlapping regions. The number of overlapping patches can quickly grow as the size of the target image increases. All these considerations are discussed, e.g., in [Mairal et al., 2014] and references therein.

In order to mitigate these issues, some [Wohlberg, 2016; Yang et al., 2010; Zeiler et al., 2010] have proposed a convolutional approach where the atoms are convolution kernels and the dictionary is a structured sum-of-convolutions linear operator. This convolutional approach is the one adopted in Chapter 4, where it is detailed. In a few words, as opposed to the patch-based approach, the size of the training samples, of the convolutional atoms and of the target image are totally decoupled. Moreover, their convolutional nature makes them translation invariant. Finally, the optimization is now global and once the dictionary is learned, the reconstruction can be done in one step over the entire image.

**Online dictionary learning**

The dictionary learning problem (2.21) can also be seen as an expected risk minimization [Mairal et al., 2014]. Indeed, let

\[ J_x(D) \triangleq \min_{\alpha \in \mathbb{R}^p} \frac{1}{2} \| x - D\alpha \|_2^2 + R(\alpha), \]

the loss function associated to \( D \), given an input data \( x \). The dictionary should be such that the value of \( J_x \) is small for all vectors in the training set \( \{x_1, \ldots, x_N\} \). As \( N \) becomes larger and larger, the empirical average
over the dataset tends *almost surely* (a.s.) to the expectation,

\[
\frac{1}{N} \sum_{i=1}^{N} J_{x_i}(D) \xrightarrow{a.s.} \mathbb{E}_x[J_x(D)],
\]

taken with respect to the unknown distribution of \( x \) [Bottou and Bousquet, 2011]. Explicitly, one should solve the problem

\[
\min_{D \in \mathcal{D}} \mathbb{E}_x[J_x(D)].
\]

This viewpoint allows to use stochastic optimization methods such as *stochastic gradient descent* [Bottou, 2010] and the *online dictionary learning* algorithm [Mairal et al., 2010] summarized in Algorithm 3. In a few words, online learning has several advantages over direct methods. The principle is to optimize over one or a few samples \( x_i \) at a time by streaming the data instead of solving \( N \) subproblems at each sparse coding step. This results in smaller memory and computation complexity, allowing to drastically increase the size of the training set. Finally, if the samples are ordered and streamed and along time, we can use a *forgetting factor* mechanism which allows to automatically adapt the dictionary to changes in the distribution of \( x \) with time. This online dictionary learning approach is adopted in Chapter 4.
Algorithm 3 Online Dictionary Learning [Mairal et al., 2010]

1: **procedure** ONLINE DL
2: **Input:** Stream of data $t \mapsto x^{[t]}$, initial dictionary $D^{[0]}$.
3: $M^{[0]} \leftarrow 0^{p \times p}$; $B^{[0]} \leftarrow 0^{n \times p}$;
4: **while** streaming data, **do**
5: Draw a training sample $x^{[t]}$ from the training set;
6: Sparse coding step:
7: $\alpha^{[t]} \leftarrow \arg\min_{\alpha \in \mathbb{R}^p} \frac{1}{2} \|x^{[t]} - D^{[t]} \alpha\|^2_2 + R(\alpha)$;
8: Update memory:
9: $B^{[t]} \leftarrow (1 - \frac{1}{t})B^{[t-1]} + \frac{1}{t} x^{[t]} \alpha^{[t]}$;
10: $M^{[t]} \leftarrow (1 - \frac{1}{t})M^{[t-1]} + \frac{1}{t} \alpha^{[t]} \alpha^{[t]}$;
11: Dictionary update initialized with $D^{[t-1]}$:
12: $D^{[t]} \leftarrow \arg\min_{D \in \mathcal{D}} \frac{1}{2t} \sum_{i=1}^t \|x^{[i]} - D \alpha^{[i]}\|^2_2$; (using $B^{[t]}$ and $M^{[t]}$)
In this chapter, we study the support recovery guarantees of underdetermined sparse regression using the $\ell_1$-norm as a regularizer and a compact convex set of the residual for data fidelity, including $\ell_\alpha$ losses for $\alpha \geq 1$, mixed norm losses, the nuclear norm loss and others. Those $\ell_\alpha$ losses are routinely used, e.g., to account for gaussian noise ($\ell_2$ loss), but also sparse ($\ell_1$ loss) or uniform ($\ell_\infty$ loss) noise models.

Support recovery is important in practice: the indices of the non-zero entries are, in some applications, more physically meaningful than their magnitude. For instance, the source localization problem [Malioutov et al., 2005; Tang and Nehorai, 2010] can be formulated as a spectrum estimation problem, where the location of the non-zero elements is the most relevant information. In compressive radar imaging [Baraniuk and Steeghs, 2007; Herman and Strohmer, 2009], the position and velocity of a target are determined by its location in the time-frequency domain. In sparse linear regression in statistics [Larsson and Selén, 2007], the support of the parameter vector corresponds to the most relevant samples or features in a dataset.

While support recovery performances are well known for the $\ell_2$ loss, a theoretical analysis of support stability is lacking for other fidelities. In this part of the thesis, we extend the existing theory from the usual
\$l_2\$ fidelity constraint to a polyhedral or more general compact convex sets of the residual. We derive a sharp condition, which ensures that the support of the vector to recover is stable to small additive noise in the observations, as long as the loss constraint size is tuned proportionally to the noise level. A distinctive feature of our theory is that it also explains what happens when the support is unstable. While the support is not stable anymore, we identify an extended support and show that this extended support is stable to small additive noise. To exemplify the usefulness of our theory, we give a detailed numerical analysis of the support stability/instability of CS recovery with these different losses. This highlights different parameter regimes, ranging from total support stability to progressively increasing support instability. Part of the results and comments presented in this chapter are based on [Degraux et al., 2016], extended to more general loss functions. Except from the numerical simulations at the end, the results are presented in increasing order of complexity. The introduction gives the basic ingredients and an informal description of the main result. The second section introduces the tools that are used and formally states the main theorem. The third section focuses on proving the results introduced in the second section.

### 3.1 Introduction

#### 3.1.1 Sparse regularization with convex fidelity constraint

This chapter studies sparse linear inverse problems of the form (2.2) introduced in Chapter 2, i.e.,

\[ y = \Phi x_0 + w, \]

where the sensing matrix \( \Phi \in \mathbb{R}^{m \times n} \) is in general rank deficient corresponding to a noisy underdetermined linear system of equations, i.e., typically in the high-dimensional regime where \( m \ll n \). The target
vector $x_0$ is assumed $s$-sparse with support $I \triangleq \text{supp}(x_0)$ and no assumption is made on the distribution of the noise term $w$.

Let $F$ be a nonempty compact convex set in $\mathbb{R}^m$ such that $0 \in \text{int}(F)$ and let $F(\cdot)$ be the finite valued coercive gauge associated to $F$ as defined in Appendix A (Definition 14). In order to recover a sparse vector $x_0$, as explained in Chapter 2, a popular regularization is the $\ell_1$-norm. In that case, let $\tau \geq 0$ and consider the following constrained sparsity-promoting optimization problem

$$\min_{x \in \mathbb{R}^n} \left\{ \|x\|_1 \text{ s.t. } (y - \Phi x) \in \tau F \right\},$$

or equivalently,

$$x_\tau \in \text{Argmin}_{x \in \mathbb{R}^n} \left\{ \|x\|_1 \text{ s.t. } F(y - \Phi x) \leq \tau \right\}. \quad (P_{\tau F}^r(y))$$

Specifically, our goal is to study the stability of the support $\text{supp}(x_\tau)$ of minimizers $x_\tau$ of $(P_{\tau F}^r(y))$. In particular, we provide a sharp analysis that allows one to control the deviation of $\text{supp}(x_\tau)$ from $\text{supp}(x_0)$ if $F(w)$ is not too large and $\tau$ is chosen proportionally to $F(w)$. In order to avoid trivialities, through the chapter, we assume that the problem $(P_{\tau F}^r(y))$ is feasible, which is the case if $w \in \tau F$ or equivalently $\tau \geq F(w)$.

The set $F$ is said to be smooth if and only if $F$ is continuously differentiable everywhere except at the origin. In contrast, $F$ is polyhedral when it can be defined by a finite number of affine inequalities. Even though the ambition of this chapter is to present the results in a unified framework, smooth (or partly smooth) and polyhedral sets tend to behave quite differently in the proofs.

A special attention is given to a very important particular case: when $F(u) = \|u\|_\alpha$, the $\ell_\alpha$-norm, and $F = B_\alpha$, the unit $\ell_\alpha$-ball. The case $\alpha = 2$ corresponds to the usual $\ell_2$ loss function, which entails a smooth constraint set, and has been studied in depth in the literature (see Sec-

\footnote{Note that, in the context of CS, one can still say something if $\tau < F(w)$, see, e.g., [Brugiapaglia and Adcock, 2017].}
tions 2.2.3 and 3.1.4 for an overview of related results). It is also smooth for $\alpha > 1$ and $\alpha < \infty$. In contrast, the cases $\alpha \in \{1, +\infty\}$ correspond to very different setups, where the loss function $\|\cdot\|_\alpha$ is polyhedral (i.e., defined by a finite set of affine inequalities) and non-smooth. As previously noted in Section 2.1.1, the case $\alpha = 1$ corresponds to a “robust” loss function, and is important to cope with impulse noise or outliers contaminating the data (see for instance [Jacques, 2013; Nikolova, 2004b; Studer et al., 2012b]). At the extreme opposite, the case $\alpha = +\infty$ is typically used to handle uniform noise such as in quantization (see for instance [Boufounos et al., 2015; Jacques et al., 2011; Moshtaghpour et al., 2016]). At the end of the chapter, we compare $\text{supp}(x_r)$ and $\text{supp}(x_0)$ for $F(u) = \|u\|_\alpha$ with $\alpha \in [1, +\infty]$ numerically in a CS experiment.

Other examples of finite coercive gauges include mixed norms, the nuclear norm and general bounded polyhedral constraints. Mixed norms are associated to structured sparsity (see Chapter 2 and references therein), which could be useful to regularize a problem in the presence of structured noise. The nuclear norm is useful if the noise term can be represented as a low-rank matrix. Polyhedral constraints, finally, connect this theoretical framework to classical results in Linear Programming (see for instance [Boyd and Vandenberghe, 2004] and in particular the list of references given in the Bibliography of Chapter 4 of that book), given that the $\ell_1$-norm is itself polyhedral.

3.1.2 Dual Certificates

Since this chapter is dedicated to the $\ell_1$-regularization, we can particularize some definitions given in Appendix A. Let us start by the noiseless problem $(\mathcal{P}_0^R)$ with observations $y_0 = \Phi x_0$. In our case, it becomes the Basis-Pursuit problem (2.14) that we repeat here for convenience,

$$
\min_x \|x\|_1 \quad \text{s.t.} \quad \Phi x = \Phi x_0. \tag{\mathcal{P}_0^0(\Phi x_0)}
$$
Note that the equality constraint, $\Phi x = \Phi x_0$, is equivalent to requiring $F(\Phi x - \Phi x_0) = 0$, but the basis pursuit problem is, obviously, independent of $F$. The set of dual certificates (A.6) in turn particularizes to

$$D_{x_0} \triangleq \{ p \in \mathbb{R}^m \mid \Phi^* p \in \partial \| x_0 \|_1 \} = \{ p \in \mathbb{R}^m \mid \Phi^* p = \text{sign}(x_0^I), \| \Phi^* p \|_\infty \leq 1 \}.$$ (3.1)

Note the splitting of the subdifferential $\partial \| x_0 \|_1$ in a set of equalities and a set of inequalities. We recall that $x_0$ is solution to $(P^0(\Phi x_0))$ if and only if $D_{x_0}$ is not empty. In that case, $x_0$ is called identifiable (also see Section 2.2.3 for the connexion with the null space property).

**Minimal $F^\circ$ certificate.** Let $F^\circ$ be the polar gauge of $F$, i.e., the support function of the set $F$. For example, when $F = \| \cdot \|_\alpha'$ then $F^\circ = \| \cdot \|_\beta$ with $\frac{1}{\alpha} + \frac{1}{\beta} = 1$. We have seen in Chapter 2 that this polar gauge has a dual relationship with the constraint set $F$. Our main theoretical finding (Theorem 5) states that the stability (and instability) of the support of $x_0$ is characterized by the following specific subset of certificates

$$p_F \in \text{Argmin}_{p \in D_{x_0}} F^\circ(p).$$ (3.3)

We call such a certificate $p_F$ a minimal $F^\circ$ certificate. Note that for the smooth case, this $p_F$ is actually unique but when $F$ is polyhedral, it might not be the case. Note also that $p_F$ is independent of $y$ and in particular independent of the noise $w$.

Associated to such a minimal $F^\circ$ certificate, we define the extended support as

$$J \triangleq \text{sat}(\Phi^* p_F) = \{ i \in \{1, \ldots, n\} \mid |(\Phi^* p_F)_i| = \| \Phi^* p_F \|_\infty \}.$$ (3.4)

Note that, from the definition of $D_{x_0}$, one always has $I \subseteq J$, hence the name. In the literature, the extended support has also been called
62 Sparse Support Recovery with Convex Fidelity Constraint

\[ y = \Phi x_0 + w \]

Figure 3.1 Relationships between important quantities. An arrow \( A \to B \) means “B depends on A”.

the equicorrelation set (see, e.g., [Tibshirani, 2013] for the \( \ell_2 \)-penalty). Figure 3.1 schematizes the relationships between important quantities introduced here.

3.1.3 Main result for sparse support recovery

The ingredients defined just above are enough to give a first informal digest of the main result. The way it is presented here is meant to give an intuitive view of what this chapter is about. The formal statement of Theorem 5 is presented in Section 3.2.3.

Assume \( x_0 \) is identifiable and \( p_F \) is a minimal \( F^\circ \) certificate and let \( \overline{x} \doteq \min_{i \in I} |x_{0,i}| \). The result states that as long as the minimal signal-to-noise ratio, \( \overline{x}/\tau \), is large enough and \( \tau \) is chosen in proportion to the noise level \( F(w) \), which is referred to as the low noise regime (see (3.9)), then there is a solution \( x_\tau \) to \( (\mathcal{P}_F(w)) \) supported exactly in the extended support \( J \). In other words, \( J = \text{supp}(x_\tau) \), i.e., it indicates the set of indexes that will be activated in the signal estimate when a small noise \( w \) is added to the observation. Note in particular that this solution \( x_\tau \) has the correct sign pattern \( \text{sign}(x_\tau^I) = \text{sign}(x_0^I) \), but might exhibit outliers if \( J \setminus I \neq \emptyset \). The situation when \( I = J \) characterizes the exact support stability (“sparsistency”).

In order to shed light on this result, we show on Figure 3.2, a small simulated CS toy-example for \( F = \| \cdot \|_\infty \). The parameters are \( n = 20 \), \( m = 10 \) and \( |I| = 4 \) and \( x_0 \) and \( \Phi \) are generated as in the experiment of Section 3.4 and we use CVX/MOSEK [Grant and Boyd, 2008, 2014] at best precision to solve the optimization programs. First, we observe
that $x_0$ is indeed identifiable by solving $(P^0(\Phi x_0))$. Then we solve (3.3) to compute $p_{\ell_{\infty}}$ and predict the extended support $J$. Finally, we add uniformly distributed noise $w$ with $w_i \overset{i.i.d.}\sim U(-\delta, \delta)$ and $\delta$ chosen appropriately to ensure that the hypotheses hold and we solve $(P^F_{\tau}(y))$. Observe that as we increase $\tau$, new non-zero entries appear in $x_\tau$ but because $w$ and $\tau$ are small enough, as predicted, we have $\text{supp}(x_\tau) = J$.

Scope and limitations. A first comment on our analysis concerns larger noise regime. Though interesting, arbitrarily large noise is out of the scope of the present work. Let us note that no other results in the literature (even for $\ell_2$) provide any insight about sparsistency in the large noise regime. In that case, we are only able to provide bounds on the distance between $x_0$ and the recovered vector but this is the subject of a forthcoming paper.

Finally the only assumption on the noise models is that $F(w)$ is small. In order to establish links between the results and the noise statistics, and also in order to obtain probabilistic non-uniform guarantees, we would need further analysis of the constant involved in the main theorem and some additional constraints on $\Phi$. However, our result is a big step towards the understanding of the solutions behavior and can definitely help in such analysis.
3.1.4 Relation to Prior Works

To the best of our knowledge, Theorem 5 is the first to study the support stability guarantees by minimizing the $\ell_1$-norm with a general, possibly non-smooth loss function. The smooth case $F = \|\cdot\|_2$ is however much more studied, and in particular, the associated support stability results we state here are now well understood. Note that most of the corresponding literature studies in general the penalized form, \( \min_x \frac{1}{2} \|\Phi x - y\|^2 + \lambda \|x\|_1 \) instead of our constrained formulation \((P_{\tau F}(y))\). In the case $F = \|\cdot\|_2$, since the loss is smooth, this distinction is minor and the proof is almost the same for both settings. However, for polyhedral cases such as $F = \|\cdot\|_\alpha$ with $\alpha \in \{1, +\infty\}$, it is crucial to study the constrained problems to be able to state our results. The support stability (also called sparsistency, corresponding to the special case $I = J$ of our result) of \((P_{\tau F}(y))\) in the case $F = \|\cdot\|_2$ has been proved by several authors in slightly different setups. In the signal processing literature, this result can be traced back to the early work of J-J. Fuchs [Fuchs, 2004] who showed Theorem 5 when $F = \|\cdot\|_2$ and $I = J$. In the statistics literature, sparsistency is also proved in [Zhao and Yu, 2006] in the case where $\Phi$ is random, the result of support stability being then claimed with high probability. The condition that $I = J$, \textit{i.e.}, that the minimal norm certificate $p_{\ell_2}$ (for $\alpha = \beta = 2$) is saturating only on the support, is often coined the \textit{irrepresentable condition} in the statistics and machine learning literature. These results have been extended recently in [Duval and Peyré, 2015] to the case where the support $I$ is not stable, \textit{i.e.} $I \subsetneq J$. Note that sparsistency-like results have been proved for many \textit{low-complexity} regularizers beyond the $\ell_1$-norm. Let us quote among others: the group-lasso [Bach, 2008a], the nuclear norm [Bach, 2008b], the total variation [Vaiter et al., 2013] and a very general class of \textit{partly-smooth} regularizers [Vaiter et al., 2014]. Let us also point out that one of the main sources of application of these results is the analysis of the performance of CS problems, where the randomness of $\Phi$ allows
to derive sharp sample complexity bounds as a function of the sparsity of \( x_0 \) and \( n \), see for instance [Wainwright, 2009]. As we have stressed in Chapter 2, these support recovery results are different from those obtained using tools such as the Restricted Isometry Property and alike in many respects. For instance, the guarantees they provide are uniform, though they usually lead to quite pessimistic worst-case bounds, and the stability is measured in \( \ell_2 \) sense.

### 3.2 Preliminaries and main result

At this point, it is necessary to bring a bit more theoretical background, to understand the formal statement (Theorem 5). Let us first particularize the dual problem \((\mathcal{D}^R_F)\) to our case. The convex conjugate of the \( \ell_1 \)-norm is the indicator function of the \( \ell_\infty \)-ball and the conjugate of the indicator of the constraint set \( \iota_F \) is the polar gauge \( F^\circ \). Therefore the dual problem reads

\[
\max_{p \in \mathbb{R}^m} \{ \langle y, p \rangle - \tau F^\circ(p) \text{ s.t. } \| \Phi^* p \|_{\infty} \leq 1 \}. \quad (\mathcal{D}^R_F(y))
\]

The general methodology that we adopt is to propose a primal-dual pair \((x, p)\) and verify under what conditions they are solution to \((\mathcal{P}^R_F(y))\) and \((\mathcal{D}^R_F(y))\). More details on this can be found in Section 3.3. For now, it is important to note that the subdifferential \( \partial F^\circ(p) \), which will appear in the optimality conditions, is going to play a central role in the results.

#### 3.2.1 Noiseless support stability

In the case of noiseless observations \((w = 0)\) and when \( \tau > 0 \), the following general lemma whose proof can be found in Section 3.3 associates to a given dual certificate \( p_F \) an explicit solution of \((\mathcal{P}^R_F(\Phi x_0))\), i.e., the problem \((\mathcal{P}^R_F(y))\) in which the observations are noiseless but the size \( \tau \) of the constraint is not 0. This formula depends on a so-called Lagrange multiplier vector \( v_F \in \mathbb{R}^n \), which will be instrumental to state Theorem 5.
Lemma 1 (Noiseless solution). We assume that $x_0$ is identifiable, i.e. it is a solution to $(\mathcal{P}^0(\Phi x_0))$, and consider $\tau > 0$. Then there exists a $v_F \in \mathbb{R}^n$ supported on $J$ such that

$$\Phi_J v_F^J \in \partial F^0(p_F) \quad \text{and} \quad -\text{sign}(v_F^\hat{J}) = \Phi_J^* p_F \quad (3.5)$$

where we denoted $\hat{J} \triangleq J \setminus I$. If $\tau$ is such that $0 < \tau < \frac{\bar{x}}{\|v_F\|_\infty}$, with $\bar{x} = \min_{i \in I} |x_{0,i}|$, then a solution $\bar{x}_\tau$ of $(\mathcal{P}_F^\tau(\Phi x_0))$ with support equal to $J$ is given by

$$\bar{x}_\tau = x_J^0 - \tau v_F^J. \quad (3.6)$$

Moreover, its entries have the same sign as those of $x_0$ on its support $I$, i.e., $\text{sign}(\bar{x}_\tau^I) = \text{sign}(x_0^I)$.

Let us remark that this lemma almost provides already the result that we are aiming for. Theorem 5 will, so to say, analyze the perturbations around $\bar{x}_\tau^I$ when some noise $w \neq 0$ is added to the mix. An important question that arises, though, is whether $v_F$ can be computed explicitly. This is the topic of the next section.

3.2.2 Model subspace and restricted injectivity conditions

Let us define the model subspace and model vector introduced in [Vaiter, 2014] (also published in [Vaiter et al., 2015]). They generalize, beyond $F^0(p) = \|p\|_1$, the link between $\partial \|p\|_1$ and the support and sign of $p$.

Definition 8 (Model subspace and model vector). For any $p \in \mathbb{R}^m$, let $S_p \triangleq \text{par}(\partial F^0(p))$ be the subspace parallel to $\partial F^0(p)$. The model subspace (or model tangent subspace) associated to the polar gauge $F^0$ at point $p$ is defined as

$$T_p \triangleq S_p^\perp.$$ 

It uniquely defines a model vector, $e_p$, as

$$\mathbf{P}_{T_p} \partial F^0(p) = \{e_p\}.$$
For $p = p_F$, we shorten the notations and write $(T_F, S_F, e_F) \triangleq (T_{p_F}, S_{p_F}, e_{p_F})$.

Those fundamental quantities, are shown on Figure 3.3 (top) for an arbitrary polyhedral polar gauge $F^\circ$. Note that for any $p \in \mathbb{R}^m$, we have $p \in T_p$. The proof can be found in Proposition 3.1 of [Vaiter, 2014] along with other interesting properties. The model vector is also equivalent to the covariant derivative\(^2\) (see, e.g., [Lewis and Zhang, 2013]) of the restricted polar gauge $F^\circ : T_p \to \mathbb{R} : p^T \mapsto F^\circ(p^T)$, i.e.,

$$e_p \triangleq \nabla_{T_p} F^\circ(p) \in T_p,$$

defined, for all $u \in \mathbb{R}^m$, as

$$u^* e_p = \langle u, \nabla_{T_p} F^\circ(p) \rangle = \frac{d}{dt} F^\circ(p + tu^T)|_{t=0}.$$

Note that by definition of $T_p$, the covariant derivative with respect to $T_p$, i.e., the model vector, always exists (and is always unique).

**Smooth gauge.** When a gauge $F^\circ$ is differentiable at $p$, the subdifferential only contains one vector $\partial F^\circ(p) = \{ \nabla F^\circ(p) \}$. Therefore we have $S_p = \{0\}$ and $T_p = \mathbb{R}^m$. This happens, for instance, when $F^\circ = \| \|_\beta$ for $\beta \in [1, \infty[$ at $p \neq 0$. The opposite situation happens at the origin for any finite polar gauge $F^\circ$. Indeed, we have $\partial F^\circ(0) = \mathcal{F}$ (see Proposition 3). Therefore, since by assumption, $0 \in \text{int}(\mathcal{F})$, we have $S_0 = \mathbb{R}^m$ and $T_0 = \{0\}$. This second situation is only mildly interesting since, in practice, we only consider cases where $p \neq 0$.

**\(\ell_1\)-norm.** The special case $F = \| \|_\infty$, $F^\circ = \| \|_1$, is depicted on the bottom left part of Figure 3.3. As expected in that case, if we define $S \triangleq \text{supp}(p_{\ell_\infty})$, we exactly have that an ONB of

\(^2\)The covariant derivative is defined in a more general setting in [Lewis and Zhang, 2013] with respect to a manifold $\mathcal{M}$ and its tangent subspace $T_{\mathcal{M}}$. In our case, the manifold is $\mathcal{M} = T_p$ and coincides with its tangent subspace $T_{\mathcal{M}} = T_p$, which simplifies the definition.
Figure 3.3 Model tangent subspace $\mathcal{T}_p$ in $\mathbb{R}^2$ for some polyhedral constraint (above), for $F = \|\cdot\|_{\alpha}$ with $(\alpha, \beta) = (\infty, 1)$ (below left) and for $(\alpha, \beta) = (1, \infty)$ (below right).
\( T_{\ell_\infty} = \{ u \in \mathbb{R}^m \mid \text{supp}(u) = S \} \) is given by \( T_{\ell_\infty} = \text{Id}_S \) and the model vector is exactly \( e_{\ell_\infty} = \text{sign}(p_{\ell_\infty}) \).

\( \ell_\infty \)-norm. For \( F = \| \cdot \|_1, F^0 = \| \cdot \|_\infty \), depicted on the bottom right part of Figure 3.3, we define \( Z \triangleq \text{sat}(p_{\ell_1}) \) and \( s_{\ell_1} \triangleq \text{Id}_Z \text{sign}(p_{\ell_1}^Z) \). It is a classical result that the expression of the subdifferential of \( \| \cdot \|_\infty \) at \( p_{\ell_1} \neq 0 \) is

\[
\partial \| p_{\ell_1} \|_\infty \triangleq \{ u \in \mathbb{R}^m \mid u^Z = 0, \langle u^Z, s_{\ell_1}^Z \rangle = 1, \text{sign}(u^Z) = s_{\ell_1}^Z \}.
\]

Proposition 3.5 from [Vaiter, 2014] gives an expression for \( (S_{\ell_1}, T_{\ell_1}, e_{\ell_1}) \) that reads,

\[
S_{\ell_1} = \{ u \in \mathbb{R}^m \mid u^Z = 0 \text{ and } \langle u^Z, s_{\ell_1}^Z \rangle = 0 \}
\]

\[
T_{\ell_1} = \{ u \in \mathbb{R}^m \mid u^Z = \rho s_{\ell_1}^Z \text{ for } \rho \in \mathbb{R} \}
\]

\[
e_{\ell_1} = \frac{1}{|Z|} s_{\ell_1},
\]

and one can verify that we have an ONB of \( T_{\ell_1} \) given by

\[
T_{\ell_1} = (\text{Id}_{Z^c}, \sqrt{|Z|} e_{\ell_1}). \quad (3.7)
\]

Restricted injectivity condition. For the general case, let \( t_F \triangleq \text{dim}(T_F) \). Let \( T_F \in \mathbb{R}^{m \times t_F} \) be an ONB of \( T_F \) so that \( P_{T_F}^{nd} = T_F T_F^* \) and let \( \Phi_{T_F}^\dagger \triangleq P_{T_F} \Phi_J \in \mathbb{R}^{m \times |J|} \) and \( \Phi_{T_F}^{\dagger*} \Phi_J \triangleq T_F T_F^* \Phi_J \in \mathbb{R}^{t_F \times |J|} \) and \( e_{T_F} \triangleq T_F e_F \in \mathbb{R}^{t_F} \) (see nomenclature). Using this notation, \( v_F^J \) is uniquely defined and expressed in closed-form as

\[
v_F^J = \Phi_{T_F}^{\dagger*} e_F = \Phi_{T_F}^{\dagger*} e_{T_F}, \quad (3.8)
\]

if and only if the following restricted injectivity condition holds,

\[
\text{Ker } \Phi_{T_F}^{\dagger*} = \{0\}. \quad (\text{INJ}_F)
\]

Note that for any basis \( T_F \) of \( T_F \), \( \text{Ker } \Phi_{T_F}^{\dagger} = \text{Ker}(\Phi_{T_F}^{T_F}) \).
If $F^\circ$ is differentiable at $p_F$, which implies $\mathcal{T}_F = \mathbb{R}^m$, then the restricted injectivity condition is $\text{Ker}(\Phi_J) = \{0\}$. Lemma 2 (deterministic part) and Lemma 3 (probabilistic part) guarantee that it holds almost surely with mild assumptions on the distribution of $\Phi$. For the special case where $\mathcal{F}$ is polyhedral, Lemma 2 and Lemma 3 also ensure that $(\text{INJ}_F)$ almost surely and that, on top of it, we have $\dim(\mathcal{T}_F) = |J|$. They are proved in Section 3.3.

**Lemma 2.** Assume that $x_0$ is identifiable. Let $t_F \triangleq \dim(\mathcal{T}_F)$ and let $T_F \in \mathbb{R}^{m \times t_F}$ be an ONB of $\mathcal{T}_F$. If there exists a basis $T_F$ such that the $t_F$ columns of $\Phi_{J^*}^T e_T^F$ together with the saturation vector, $\eta_T^j \triangleq \Phi_{J^*}^T p_F \in \{-1, 1\}^{|J|}$, are in general direction (see Definition 3). Then, $t_F \geq |J|$ and $(\text{INJ}_F)$ holds.

Furthermore, assume that $F^\circ$ is not differentiable at $p_F$. If the $n$ columns of $\Phi_{J^*}^T F$ together with $e_T^F$ are in general direction\(^3\), then $t_F \leq |J|$ and $\Phi_{J^*}^T F$ is full row rank. Therefore, if both assumptions hold, $t_F = |J|$ and $\Phi_{J^*}^T F$ is square and invertible.

**Lemma 3.** If $\Phi$ is randomly drawn from a continuous distribution with i.i.d. entries, e.g., Gaussian, then as soon as $x_0$ is identifiable, the first condition of Lemma 2 hold with probability 1 over the distribution of $\Phi$.

Moreover, if $\mathcal{F}$ is polyhedral and arbitrarily fixed, then the second condition of Lemma 2 holds as well with probability 1.

**Remark 2.** The second condition of Lemma 3 comes from the fact that in the polyhedral case, the number of possible model vectors is finite since each possible $e_F$ is associated to a face of $\mathcal{F}$. As opposed to the polyhedral case, if $F^\circ$ is differentiable at $p_F$, the set of possible model vectors (in function of $\Phi$) is not discrete and $e_F$ may continuously depend on $\Phi$. This is why the second assumption of Lemma 2 does not hold in general. In particular, it is usually the case that $|J| < m$ (which is, so to say, the goal of sparse regularization). In that case,

\[^3\text{Note that since linear independence is invariant under rotation, this second assumption is independent of the chosen basis } T_F.\]
Table 3.1 Model tangent subspace, inverse of the restricted sensing matrix and model vector for $F = \| \cdot \|_\alpha$.

<table>
<thead>
<tr>
<th>$\alpha$</th>
<th>$T_F \subseteq \mathbb{R}^m$ such that ${ u \in \mathbb{R}^m }$</th>
<th>$\Phi_JF^+\Phi_J^T$</th>
<th>$e_F \in T_F \subseteq \mathbb{R}^m$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\infty$</td>
<td>$\text{supp}(u) = S$</td>
<td>$\Phi_J^{-1}$</td>
<td>$\text{Id}<em>S \text{sign}(p</em>{\ell_\infty}^S)$</td>
</tr>
<tr>
<td>1</td>
<td>$u^Z = \rho \text{sign}(p_{\ell_1}^Z)$ for any $\rho \in \mathbb{R}$</td>
<td>$\Phi_J^{T_{\alpha_{-1}}}^{-1}$</td>
<td>$\frac{1}{</td>
</tr>
<tr>
<td>2, $\infty$</td>
<td>$-\Phi_J^{T_{\alpha_{+1}}}^{-1}$</td>
<td>$-\frac{p_{\ell_2}}{|p_{\ell_2}|_2}$</td>
<td>$\nabla |p_{\ell_\alpha}|_\beta$</td>
</tr>
</tbody>
</table>

$\Phi_JF^+\Phi_J^T = \Phi_J\Phi_J^+ = P_{\text{Im}(\Phi_J)}$, which is, in general (and in particular when $|J| < m$), not equal to $P_T = \text{Id}$.

To conclude this section, Table 3.1 summarizes the quantities introduced here for $F = \| \cdot \|_\alpha$ in the three specific cases $\alpha \in \{1, 2, +\infty\}$ and for the general smooth case $\alpha \in ]1, +\infty[$.

### 3.2.3 Formal statement of the main result

Our main contribution is Theorem 5 below. A similar result is known to hold in the case of the $\ell_2$ loss ($\alpha = 2$, see Section 3.1.4). The theorem extends it to general smooth losses (such as $\ell_\alpha$-norms with $\alpha \in ]1, +\infty[$), polyhedral losses (such as $\ell_\alpha$-norms with $\alpha \in \{1, +\infty\}$) and beyond. The proof is detailed in Section 3.3. It is important to emphasize that the proof strategy is significantly different from the classical approach developed for $\ell_2$.

**Theorem 5.** Suppose that $x_0$ is a solution to $(P^0(\Phi x_0))$, i.e. $x_0$ is identifiable, and let $p_F$ be a minimal $F^0$ certificate (see (3.3)) with associated extended support $J$ (see (3.4)). Let $t_F \triangleq \dim(T_F)$. Suppose that $t_F \geq |J|$, that the restricted injectivity condition (IN$J_F$) is satisfied and $v_F^J$ can be explicitly computed (see (3.5)) and, when $t_F < m$, $v_F^J$ is such that $\Phi_J v_F^J \in \text{ri}(\partial F^0(p_F))$. Then, there exist constants $c_1, c_2 > 0$ depending only on $\Phi$ and $p_F$ such that,
for any \((w, \tau)\) satisfying

\[
F(w) < c_1 \tau \quad \text{and} \quad \tau \leq c_2 x \quad \text{where} \quad x \triangleq \min_{i \in I} |x_{0,i}|, \tag{3.9}
\]

the following holds.

**Case 1 ("non-smooth" case):** If \(t_F = |J|\), i.e., if \(\Phi_J^T F \) is invertible (see Lemma 2), then a solution of \((\mathcal{P}_\tau F(y))\) with support equal to \(J\) is given by

\[
x^J_\tau (w) = x^J_0 + \Phi_J^{-1} w - \tau v^J_F . \tag{3.10}
\]

**Case 2 ("smooth" case):** If \(t_F > |J|\), let \(Z \in \mathbb{R}^{t_F \times t_F - |J|}\) be an ONB of \(\ker(\Phi_J^T F^*)\) and define the mapping,

\[
e^T_F p : \mathbb{R}^{t_F - |J|} \to \mathbb{R}^{t_F} : z \mapsto e^T_F p(z) = T_F^* \nabla F \circ (T_F Z z + p_F) .
\]

If \(e^T_F p(z)\) is locally continuous and one-to-one (injective) around \(z = 0\), then there exists a neighborhood \(W \subseteq \mathbb{R}^{t_F}\) of \(0\) (depending on \(\tau\)) and a locally continuous one-to-one mapping \(z(w^T_F)\) such that \(z(0) = 0\) and for all \(w^T_F \in W\), a solution to \((\mathcal{P}_\tau F(y))\) with support \(J\), using \(v^J_F(w^T_F) = \Phi_J^T e^T_F p(z(w^T_F))\), is given by

\[
x^J_\tau (w^T_F) = x^J_0 + \Phi_J^T w^T_F - \tau v^J_F(w^T_F) , \tag{3.11}
\]

such that \(x^J_\tau (0) = \bar{x}^J.\) If moreover \(e^T_F p(z)\) is locally \(C^1\) around \(z = 0\), then the mapping \(x^J_\tau (w^T_F)\) is locally \(C^1\) and has the following first order expansion around \(w^T_F = 0\),

\[
x^J_\tau (w^T_F) = x^J_0 - \tau v^J_F + \Phi_J^T \left( \text{Id} - E_0 (Z^* E_0)^+ \right) w^T_F + \mathcal{O}(\|w^T_F\|^2) , \tag{3.12}
\]

with \(E_0 \triangleq \nabla e^T_F p(0)\), the Jacobian matrix of the mapping at \(z = 0\). The Big-\(O\) notation must be understood as entry-wise, i.e., each entry of the residual vector is \(\mathcal{O}(\|w^T_F\|^2)\). Finally, if the restricted polar gauge, \(F^\circ : T_F \to \mathbb{R} : p^T_F \mapsto \)
3.2 Preliminaries and main result

$F^{\circ}(p^F_T)$, is locally $C^2$ around $p_F \in T_F$, we can use the covariant Hessian\(^4\), and its matrix equivalent $H_0 \triangleq \nabla^2_{T_F} F^{\circ}(p_F) \in \mathbb{R}^{m \times m}$. The Jacobian matrix then reads $E_0 = T^*_p H_0 T_F Z$.

Case 1 is only applicable in the non-smooth case, e.g., when the assumptions of Lemma 2 are satisfied. Lemma 3 gives sufficient conditions for the assumption $t_F = |J|$ to be valid almost surely if $F$ is polyhedral. In that case the constants $c_1$ and $c_2$ are explicitly given in the proof.

Case 2, on the other hand, is more general and holds in particular when $F$ is smooth, i.e., when $T_F = \mathbb{R}^m$ in which case $e_p(z) = \nabla F^{\circ}(Zz + p_F)$. We keep the notations general in order to account for gauges that are neither polyhedral nor smooth, such as mixed norms for instance, but for which the model subspace $T_F$ is stable to small noise.

Note that the Hessian $\nabla^2 F^{\circ}(\cdot)$ (or the covariant Hessian $\nabla^2_{T_F} F^{\circ}(\cdot)$) is not always well behaved. For instance, for $F^{\circ} = \|\cdot\|_\beta$, with $\beta \in [1, +\infty[$, introducing the entry-wise notation $(p)^a \triangleq \text{sign}(p)|p|^a$ for $a \in \mathbb{R}$, the gradient at $p \neq 0$ reads

$$\nabla \|p\|_\beta = \|p\|_\beta^{1-\beta} (p)^{\beta-1}.$$  

In turn, the entry $(i, j)$ of the Hessian reads, for all $i \neq j$,

$$\left[ \nabla^2 \|p\|_{\beta} \right]_{i,j} = (1 - \beta) \|p\|_{\beta}^{1-2\beta} (p_i)^{\beta-1} (p_j)^{\beta-1},$$

and the $i^{th}$ diagonal entry,

$$\left[ \nabla^2 \|p\|_{\beta} \right]_{i,i} = (\beta - 1) \|p\|_{\beta}^{1-2\beta} (\|p\|_{\beta}^{\beta} - |p_i|^\beta)|p_i|^\beta-2.$$  

It is easy to see that when $\beta \in [1, 2[$, $\lim_{p_i \to 0} \left[ \nabla^2 \|p\|_{\beta} \right]_{i,i} = +\infty$ and not easy to guarantee that $p_i \to 0$ never happens for any entry of $p = Zz + p_F$ or even $p = p_F$. This is why, firstly, we only require that $e^T_F(z)$ is locally

\(^4\) see the proof and e.g., [Lewis and Zhang, 2013] for a precise definition. See also [Mordukhovich and Rockafellar, 2012] and references therein for more information on second-order differentiability and the generalized Hessian.
differentiable (as opposed to $\nabla F^\circ(p)$) and moreover, we leave the door open for $e_p^T(z)$ to be non-differentiable, in which case the dependency in $w$ of the proposed solution is not explicit, but the results concerning support stability still hold.

Finally, in the special case $F^\circ = \|\cdot\|_2$, we have, for $p \neq 0$, $T_p = \mathbb{R}^m$,

$$\nabla \|p\|_2 = \frac{p}{\|p\|_2}, \quad \text{and} \quad e_p(z) = \frac{Zz + p_F}{\|Zz + p_F\|_2},$$

and, since $\Phi_j^+Z = (\Phi_j^* \Phi_j)^{-1} \Phi_j^*Z = 0$,

$$v_p^J(w) = \frac{\Phi_j^+(Zz(w) + p_F)}{\|Zz(w) + p_F\|_2} = \frac{\Phi_j^+p_F}{\|Zz(w) + p_F\|_2}.$$ 

Therefore, the solution reads

$$x^J_\tau_2(x) = x_0^J + \Phi_j^+w - \frac{\tau}{c(w)} \Phi_j^+p_F,$$  \hspace{1cm} (3.13)

with a scalar $c(w) = \|Zz(w) + p_F\|_2$ that should not be too difficult to isolate, when injecting (3.13) in the optimality conditions.

### 3.3 Proofs

In this section, we prove Theorem 5, i.e., the main result of this chapter along with a few associated results such as Lemma 1 from Section 3.2.1, Lemma 2 and Lemma 3 from Section 3.2.2, and we state and prove Proposition 1, which allows to split $\partial F^\circ(p)$ in an equality on $T_p$ and an inequality on $S_p$, analogously to (3.2) with $\partial \|x_0\|_1$ (defining the set of certificates $D_{x_0}$).

#### 3.3.1 Proofs of the lemmas and subdifferential decomposability

Before that, let us complete the picture about duality and give the first order optimality conditions that are central in the proof. Using $\iota_F$, the indicator function associated to $F$, we can rewrite the original constrained
problem \((\mathcal{P}^F(y))\) as the equivalent unconstrained problem
\[
\min_{x \in \mathbb{R}^n} \left\{ \|x\|_1 + \iota_F \left( \frac{y - \Phi x}{\tau} \right) \right\}.
\]
The convex conjugate (see Appendix A) of the indicator function \(\iota_F\) is the polar gauge \(F^\circ\). We can therefore rewrite \((\mathcal{P}^F(y))\) as the equivalent saddle-point problem,
\[
\min_{x \in \mathbb{R}^n} \max_{p \in \mathbb{R}^m} \left\{ \|x\|_1 + \langle y - \Phi x, p \rangle - \tau F^\circ(p) \right\}.
\]
\((\mathcal{S}^F(y))\)

The first order optimality conditions read in that case
\[
\begin{bmatrix}
0 \\
y
\end{bmatrix} \in \begin{bmatrix}
\partial \|\cdot\|_1 & -\Phi^* \\
\Phi & \tau \partial F^\circ
\end{bmatrix} \begin{bmatrix}
x \\
p
\end{bmatrix},
\]
(3.14)
where \(\partial \|\cdot\|_1\) and \(\partial F^\circ\) are non-linear set-valued operators. Note that we already used the definition of \(\partial \|\cdot\|_1\) to define the set of certificates \(\mathcal{D}_{x_0}\) (3.2). Similarly, denoting \(I_x = \text{supp}(x)\), the first row of (3.14) is equivalent to require that
\[
\Phi_{I_x}^* p = \text{sign}(x_{I_x}^I) \quad \text{and} \quad \|\Phi^* p\|_\infty \leq 1.
\]
(3.15)
The dual problem \((\mathcal{D}^F(y))\) was already given in Section 3.2.2.

**Proof of Lemma 1** Let us rewrite the problem (3.3) by introducing the auxiliary variable \(\eta = \Phi^* p\) as
\[
\min_{p, \eta} \left\{ F^\circ(p) + \iota_{B_\infty}(\eta) \text{ s.t. } \eta = \Phi^* p, \eta^I = \text{sign}(x_0^I) \right\},
\]
(3.16)
where \(\iota_{B_\infty}\) is the indicator function of the unit \(\ell_\infty\) ball. Define the Lagrange multipliers \(v\) and \(z^I\) and the associated Lagrangian function
\[
L(p, \eta, v, z^I) = F^\circ(p) + \iota_{B_\infty}(\eta) + \langle v, \eta - \Phi^* p \rangle + \langle z^I, \eta^I - \text{sign}(x_0^I) \rangle.
\]
The first order optimality conditions for $p$ and $\eta$ read

$$\Phi v \in \partial F^\circ(p) \quad \text{and} \quad -v - z \in \partial \iota_{\mathcal{B}_\infty}(\eta),$$

where, by definition, $z^J = 0$. From the normal cone of $\mathcal{B}_\infty$ at $\eta$ on its boundary, the second condition reads

$$-v - z \in \left\{ u \mid u^J = 0, \text{sign}(u^J) = \eta^J \right\},$$

where $J = \text{sat}(\eta) = \text{sat}(\Phi^*p)$. Since $I \subseteq J$, $v$ is supported on $J$. Moreover, on $\tilde{J} = J \setminus I$, we have $-\text{sign}(v^{\tilde{J}}) = \eta^{\tilde{J}}$. As $p_F$ is a solution to (3.16), we can define a corresponding vector of Lagrange multipliers $v_F$ supported on $J$ such that $-\text{sign}(v^{\tilde{J}})_F = \Phi^{\tilde{J}}p_F$ and $\Phi_{J}v^{J}_F \in \partial F^\circ(p_F) - v$.

To prove the lemma, it remains to show that $\bar{x}_\tau$ is indeed a solution to $(\mathcal{P}^\tau_F(y))$, i.e., it obeys (3.14) for some dual variable $\hat{p}$. We will show that this is the case with $\hat{p} = p_F$. Observe that $p_F \neq 0$ as otherwise, it would mean that $x_0 = 0$, which contradicts our initial assumption of non-zero $x_0$. We can then directly see that the second row of (3.14) is satisfied. Indeed, noting $y_0 \triangleq \Phi x_0$, we can write

$$y_0 - \Phi_{J} \bar{x}_\tau = \tau \Phi_{J}v^{J}_F \in \tau \partial F^\circ(p_F).$$

By definition of $p_F$, we have $\|\Phi^*p_F\|_\infty \leq 1$. In addition, it must satisfy $\Phi_{J}^*p_F = \text{sign}(\bar{x}^J_\tau)$. Outside $I$, the condition is always satisfied since $-\text{sign}(v^{\tilde{J}})_F = \Phi^{\tilde{J}}p_F$. On $I$, we know that $\Phi_{I}^*p_F = \text{sign}(x^I_0)$. The condition on $\tau$ is thus $|x_0,i| > \tau |v_{F,i}|$, for all $i \in I$, or equivalently,

$$\tau < \frac{\bar{x}_\tau}{\|v^{J}_F\|_\infty}.$$  

Proof of Lemma 2: By the first assumption, for any subset of indices $T' \subseteq [t_F]$ we have

$$|T'| < |J| \Rightarrow \eta^{J'}_F \notin \text{Im} \left( (\Phi_{T'}^*)_{T'} \right).$$

(3.17)
The existence of $p_F$ is implied by the identifiability of $x_0$ and we have the following,

$$\exists p_F \Rightarrow \exists z, \Phi_j^{T_F} z = \eta_F^j \ (3.17) \Rightarrow t_F \geq |J|.$$  

Indeed, suppose to the contrary, that $t_F < |J|$. Then, by (3.17), the system $\Phi_j^{T_F} z = \eta_F^j$ would have no solution in $z$, which would contradict the existence of $p_F$. Indeed, $z = p_F^{T_F}$ is a solution by definition of $p_F \in T_F$. Since the $t_F$ columns of $\Phi_j^{T_F}$ are in general direction, it is full row rank. Therefore, $\Phi_j^{T_F}$ is full column rank, which is equivalent to (INJ$_F$). By the second assumption, for any subset of indices $J' \subseteq [n]$, we have

$$|J'| < t_F \Rightarrow e_F^{T_F} \notin \text{Im}(\Phi_{J'}^{T_F}).$$  

(3.18)

As established by Lemma 1, the existence of $v_F$ is implied by the identifiability of $x_0$ and we have

$$\exists v_F \Rightarrow \exists v^J, \Phi_j^{T_F} v^J = e_F^{T_F} \ (3.18) \Rightarrow |J| \geq t_F.$$  

(3.19)

Indeed, suppose that, to the contrary, $|J| < t_F$. Then, by (3.18), the system $\Phi_j^{T_F} v^J = e_F^{T_F}$ has no solution in $v^J$, which contradicts the existence of $v_F$. Since the $|J|$ columns of $\Phi_j^{T_F}$ are in general direction, it is full row rank. Therefore, under both assumptions, $|J| = t_F$ and $\Phi_j^{T_F}$ is square and invertible.

**Corollary 1.** If the conditions of Lemma 2 are satisfied, we have

$$\Phi_j^{T_F} \Phi_{j}^{T_F +} = P_{T_F}.$$  

(3.20)

**Proof.** We can develop

$$\Phi_j^{T_F} (\Phi_j^{T_F})^{-1} = \text{Id}$$

$$\Phi_j^{T_F} (\Phi_j^{T_F})^{+} = \text{Id}$$

$$T_F^* \Phi_j (T_F^* \Phi_j)^{+} = \text{Id}.$$
Multiplying to the left and to the right by $T_F$ and $T_F^*$ on both sides of the equality gives

$$\Phi^T J (T_F^* \Phi J)^+ T_F^* = P_{T_F}.$$  

Since $T_F$ has orthonormal columns and $T_F^* = T_F^+$, we have $(T_F^* \Phi J)^+ T_F^* = (T_F^* \Phi J)^+$, and finally

$$\Phi^T J \Phi^T J = P_{T_F}.$$  

\[\square\]

Proof of Lemma 3  By definition of $J$, we know that $\eta_{j_i}$ is a sign vector of dimension $|J|$, i.e., $\eta_{j_i} \in \{-1, 1\}^{|J|}$. That finite set is independent of $\Phi$. Similarly, $T_F$ belongs to a finite set of subspaces that only depends on the definition of the polytope $F^\circ$. As explained in Chapter 2 (Proposition 3), the subdifferential of $F^\circ(p_F)$ is characterized by the face of $F$ exposed by $p_F$, i.e., the intersection of $F$ and the supporting hyperplane $\{ u \in \mathbb{R}^m | \langle u, p_F \rangle = F^\circ(p_F) \}$. We can always define the polytope $F$ by the intersection of a finite number $q$ of half spaces as

$$F \triangleq \{ u \in \mathbb{R}^m | \langle f_i, u \rangle \leq 1, i \in [q] \}.$$  

(3.21)

Let $Q = \{ i \in [q] | \langle f_i, u \rangle = 1, \text{for any } u \in \partial F^\circ(p_F) \}$. The subspace $S_F$ parallel to the exposed face can be written as

$$S_F = \{ u \in \mathbb{R}^m | \langle f_i, u \rangle = 0, \forall i \in Q \}.$$  

(3.22)

Therefore, the subspace $T_F$, is spanned by these $f_i$ as

$$T_F = \text{Span}(\{ f_i \}_{i \in Q}).$$

Even if the selection of $T_F$ may depend on the particular instance of $\Phi$, the set of all possible $T_F$, i.e., the set of all possible $Q$, is finite and the
the columns of $\Phi$ are drawn independently of the $f_i$. If $\Phi$ is randomly drawn from a continuous distribution with i.i.d. entries, its $n$ columns are in general direction in $\mathbb{R}^m$ with probability 1. In particular, since the number of possible $T_F$ is finite, with probability 1 there exists a basis $T_F$ such that the $n$ columns of $\Phi^{T_F}$ are in general direction in $\mathbb{R}^{t_F}$ and the $t_F$ columns of $\Phi_{J_F^*}^{T_F}$ are in general direction in $\mathbb{R}^{|J_F|}$. For any subset of indices $T' \subseteq [t_F]$ such that $|T'| < |J|$, the subspace spanned by the columns of $(\Phi_{J_F^*}^{T_F})_{T'}$ will be a hyperplane of dimension $|T'|$ in $\mathbb{R}^{|J_F|}$. The probability that one of these hyperplanes intersects the finite set $\{-1, 1\}^{|J_F|}$ is 0. This is why $\eta^J_F \notin \text{Im}((\Phi_{J_F^*}^{T_F})_{T'})$ and the columns of $\Phi_{J_F^*}^{T_F}$ together with $\eta^J_F$ are in general direction. We follow a similar argument for the second assumption. First, note that $e_F$ and by extension $e_{T_F}^{T_F}$ (for an arbitrarily fixed basis) can only take a finite number of values, depending only on the face of $F$ exposed by $p_F$. More precisely, to every possible subset of indices $Q$, corresponds a model vector $e$ uniquely defined by

$$ e \in \text{Span}(\{f_i\}_{i \in Q}), $$

$$ \langle f_i, e \rangle = 1 \quad \text{for all} \quad i \in Q. $$

For any subset of indices $J' \subseteq [n]$ such that $|J'| < t_F$, the subspace spanned by the columns of $\Phi_{J'}^{T_F}$ will be a randomly oriented hyperplane of dimension $|J'|$ in $\mathbb{R}^{t_F}$. The probability that one of these hyperplanes intersects any of the finitely many possible $e^{T_F}$ is 0. This is why $e_{J'}^{T_F} \notin \text{Im}(\Phi_{J'}^{T_F})$ and the columns of $\Phi_{J'}^{T_F}$ together with $e_{J'}^{T_F}$ are in general direction. 

Like the proof of Lemma 1, the proof of Theorem 5 will consist in finding a primal-dual pair that satisfies the first order optimality condition (3.14). It will be convenient to split the subdifferential $\partial F^0(p)$ in a similar way, i.e., with an equality on $T_p$ and an inequality on $S_p$. As it happens, Theorem 3.1 of [Vaiter, 2014] (Theorem 1 of [Vaiter et al.,
establishes a general way to similarly split the differential of any proper convex, bounded from below, continuous function $J(x)$. Proposition 3.12 of [Vaiter, 2014] (Proposition 5 of [Vaiter et al., 2015]) establishes a similar result for the particular case of a strong gauge, which is, unfortunately, too restrictive for our need because it rules out $\|\cdot\|_\infty$. Since the first result is a little bit too general and the second one is too restrictive, we state and prove the case of a finite valued gauge $\gamma_C$ associated to the closed convex set $C$, i.e., such that $0 \in \text{int}(C)$. Note that Proposition 3 from Chapter 2 almost establishes the result we need.

**Proposition 1** (Decomposability for a finite gauge). Let $C \subseteq \mathbb{R}^n$ be a closed convex set such that $0 \in \text{int}(C)$ and let $\gamma_C$ be the associated finite gauge. Let $\bar{S}_x = \text{aff}(\partial \gamma_C(x))$, $S_x = \text{par}(\partial \gamma_C(x)) = \bar{S}_x - \bar{T}_x$, $T_x = S_x^\bot$ and $e_x = P_{T_x} \partial \gamma_C(x)$. Then the subdifferential of $\gamma_C$ at $x$ reads

$$\partial \gamma_C(x) = \{ u \in \mathbb{R}^n \mid u^T_x = e_x \text{ and } \gamma_C^o(u^{S_x} + e_x) \leq 1 \},$$

or equivalently, for any $f \in \text{ri}(\partial \gamma_C(x))$,

$$\partial \gamma_C(x) = \{ u \in \mathbb{R}^n \mid u^T_x = e_x \text{ and } \gamma_C^o - f(u^{S_x} - f^{S_x}) \leq 1 \}.$$  

**Proof.** By projecting both sides of $u \in \partial \gamma_C(x)$ on $T_x$ and $S_x$, we find,

$$u^T_x = e_x \text{ and } u^{S_x} \in \partial \gamma_C(x) - e_x.$$  

Since from Proposition 3 we have $\partial \gamma_C(x) = C^o \cap \bar{S}_x$, the second condition is equivalent to

$$u^{S_x} + e_x \in C^o \cap \bar{S}_x.$$  

Since $e_x \in \bar{S}_x \cap T_x$ (Proposition 1 of [Vaiter et al., 2015]), for any $u \in \mathbb{R}^n$, we have that $u^{S_x} + e_x \in \bar{S}_x$. Therefore, it is also equivalent to

$$u^{S_x} + e_x \in C^o$$

---

5 The definition, (e.g., Definition 3.4 [Vaiter, 2014]), is not important for understanding the rest. As a simple example, $\|\cdot\|_1$ is a strong gauge while $\|\cdot\|_\infty$ is not.
which gives the first expression, by definition of $\gamma_C^\circ$. Removing $f \in \text{ri}(\partial \gamma_C(x))$ on both sides gives

$$u^{S_x} - f^{S_x} \in C^\circ - f,$$

noting that $f = e_x + f^{S_x}$ since $f \in \partial \gamma_C(x)$. This gives the second expression.

\begin{remark}
In comparison, in the more general Theorem 1 of [Vaiter et al., 2015], the inequality part is $\gamma_{\partial \gamma_C(x)} - f(u^{S_x} - f^{S_x}) \leq 1$ for any $f \in \text{ri}(\partial \gamma_C(x))$. The gauge $\gamma_{\partial \gamma_C(x)} - f$ is then called the subdifferential gauge with respect to $f$. In the case of a finite gauge, $\partial \gamma_C(x) - f = (C^\circ - f) \cap S_x$ so that the inequality can be replaced by $\gamma_C^\circ - f(u^{S_x} - f^{S_x}) \leq 1$. However, in general $\gamma_C^\circ - f(u^{S_x} - f^{S_x}) \neq \gamma_C(u^{S_x})$. Equality holds precisely when $\gamma_C$ is a strong gauge.

\end{remark}

$\ell_1$-norm. We can directly check that this holds for the simple case of the $\ell_1$-norm, whose subdifferential at $x$ reads

$$\partial \|x\|_1 = \left\{ u \in \mathbb{R}^n \mid \text{sign}(x^{I_x}), \|u\|_\infty \leq 1 \right\},$$

where $I_x = \text{supp}(x)$. Indeed, let $T_x = \text{Id}_{I_x}$ be an ONB of $T_x$ the equality condition reads $u^{I_x} = \text{sign}(x^{I_x})$. Assuming the equality condition holds, the inequality condition, $\|\text{Id}_{I_x} u^{I_x} - \text{sign}(x)\|_\infty \leq 1$, is satisfied if, and only if $\|u\|_\infty \leq 1$.

$\ell_\infty$-norm. A more enlightening example, is the $\ell_\infty$-norm, which is a finite gauge but not a strong gauge (see Proposition 3.15 of [Vaiter, 2014]). Let $Z_p = \text{sat}\{p\}$, $s_p = \text{Id}_{Z_p} \text{sign}(p^{Z_p})$ and recall that the expression of the subdifferential of $\|\|_\infty$ at $p \neq 0$ is

$$\partial \|p\|_\infty \triangleq \left\{ u \in \mathbb{R}^m \mid u^{Z_p} = 0, \langle u^{Z_p}, s^{Z_p} \rangle = 1, \text{sign}(u^{Z_p}) = s^{Z_p} \right\}.$$
Let an ONB $T_p$ be defined, as in (3.7), by $T_p = (\text{Id}_{Z_p}, \frac{1}{|Z_p|} s_p)$. We can write the equality condition,

$$T_p^* u = T_p^* e_p \iff u_{Z_p} = 0$$

and

$$\langle s_p, u_{Z_p} \rangle = 1,$$

which gives exactly the first part of $\partial \|p\|_\infty$. Assume it holds for $u$, then we must show that the inequality condition from Proposition 1,

$$\|u^{S_p} + e_p\|_1 \leq 1,$$

holds if, and only if $\text{sign}(u_{Z_p}) = s_{Z_p}$. Note that $S_p = T_p^\perp = \text{Ker}(T_p^*)$.

$$u^{S_p} = u - u_{T_p} = (\text{Id} - T_p T_p^*) u.$$

$$u - T_p T_p^* u = \text{Id}_{Z_p} u_{Z_p} - \frac{1}{|Z_p|} s_p \langle s_p, u_{Z_p} \rangle = \text{Id}_{Z_p} u_{Z_p} - \frac{1}{|Z_p|} s_p,$$

where the second equality comes from the assumption $\langle s_p, u_{Z_p} \rangle = 1$.

The condition becomes

$$\|u^{S_p} + \frac{1}{|Z_p|} s_p\|_1 = \|u_{Z_p}\|_1 \leq 1.$$

To conclude, note that if $\text{sign}(u_{Z_p}) = s_{Z_p}$ then $\|u_{Z_p}\|_1 = \langle s_p, u_{Z_p} \rangle = 1$, which, of course, is $\leq 1$. Conversely, if $\text{sign}(u_{Z_p}) \neq s_{Z_p}$, then, $\|u_{Z_p}\|_1 > \langle s_p, u_{Z_p} \rangle = 1$. This proves that the proposed splitting is equivalent to the definition of $\partial \|p\|_\infty$.

First order optimality for a finite gauge. In the case of $\partial F^\circ(p)$, this decomposability property reads

$$\partial F^\circ(p) = \{ u \in \mathbb{R}^m \mid u_{T_p} = e_p \text{ and } F(u^{S_p} + e_p) \leq \tau \}.$$

We can therefore rewrite the second optimality condition as

$$y_{T_p} - \Phi_{l_x}^T x^I = \tau e_p \text{ and } F(y^{S_p} - \Phi_{l_x}^{S_p} x^I + \tau e_p) \leq \tau.$$
Note that, since \( p \in T \) we can also rewrite (3.15),
\[
\Phi_{I^x}^T p = \text{sign}(x_{I^x}) \quad \text{and} \quad \left\| \Phi_{I^x}^T p \right\|_\infty \leq 1.
\] (3.24)

### 3.3.2 Proof of Theorem 5

We will consider Case 1 first \((t_F = |J|)\), which holds for (but is not restricted to) polyhedral constraints and then prove the theorem for Case 2 \((t_F > |J|)\), which holds for (but is not restricted to) smooth constraints. Our proof consists in verifying that the vector supported on \( J \) defined by (3.10), (3.11), or (3.12), depending on the situation, is indeed a solution to \((P_{\tau F}(\Phi x_0 + w))\) for an appropriate regime of the parameters \((\tau, F(w))\). We need to show that the first order conditions (3.14), or equivalently (3.24) and (3.23), hold for some \( \hat{p} \) that is a solution of the dual problem \((D_{\tau F}^r(\Phi x_0 + w))\) and where \( \hat{x} \) is the proposed \( x_{\tau 1} \) or \( x_{\tau 2} \) accordingly. For Case 1, we show that under the small noise hypotheses, this holds for \( \hat{p} = p_F \), i.e., \( p_F \) is solution of \((D_{\tau F}^r(\Phi x_0 + w))\).

For Case 2, however, one cannot assume, in general, that \( \hat{p} = p_F \) as soon as \( w \neq 0 \). Nevertheless, we will show that if \( F(w) \) is not too large, defining \( \hat{p}(z) = T_F z + p_F \), there exists a locally continuous mapping \( \mathbb{R}^m \rightarrow \mathbb{R}^{t_F - |J|} \times \mathbb{R}^{|J|} : w \mapsto (z, \hat{x}^J)(w) \) around the noiseless solution \((0, (0, \hat{x}^J_0))\) (see Lemma 1). This mapping is implicitly defined by \( \hat{x}^J(w) = x^J_0 + \Phi_{J^c}^{T\tau} w + \Phi_{J^c}^{T\tau} - \tau \Phi_{J^c}^{T\tau} e_p^{T\tau}(z(w)) \) and under the small noise hypotheses, the pair \((\hat{x}(w), \hat{p}(z(w)))\) is an optimal primal-dual pair.

**Case 1 \((t_F = |J|)\)** Let us start by verifying the equality part of (3.24) with \( \hat{p} = p_F \), i.e., \( \Phi_J^* p_F = \text{sign}(x^J_{\tau 1}) \). On the support \( I \), we need to verify that \( \text{sign}(x^I_{\tau 1}) = \Phi_I^* p_F = \text{sign}(x^I_0) \), where the last equality comes from the definition of a dual certificate (3.2). Noting \( Id^I_J \) the restriction from \( J \) to \( I \), we have
\[
\text{sign} \left( x^I_0 + Id^I_J \Phi_{J^c}^{T\tau} w + \Phi_{J^c}^{T\tau} - \tau w_F \right) = \text{sign} \left( x^I_0 \right)
\]
as soon as
\[ \left| \left( \Phi_j^{-1} \Phi_j w \right) \right|_i - \tau v_F^i < |x_0^i| \quad \forall i \in I. \]

It is sufficient to require
\[
\left\| \text{Id}_I^f \Phi_j^{-1} w - \tau v_F^I \right\|_\infty < \bar{x},
\]
\[
b F(w_F^T) + \tau \left\| v_F^I \right\|_\infty < \bar{x},
\]
with \( \bar{x} = \min_{i \in I} |x_0^i| \) and
\[
b \triangleq \sup_{v \in \mathbb{R}^F} \frac{\left\| \text{Id}_I^f \Phi_j^{-1} v \right\|_\infty}{F(T_F v)} = \sup_{u \in F \cap T_F} \left\| \text{Id}_I^f \Phi_j^{T_F} u \right\|_\infty.
\]

Injecting the fact that \( F(w_F^T) \leq F(w) < c_1 \tau \) (the value of \( c_1 \) will be derived later), we get the condition
\[
\tau (bc_1 + \nu) \leq \bar{x},
\]
with \( \nu = \left\| v_F^I \right\|_\infty = \left\| \text{Id}_I^f \Phi_j^{-1} e_F \right\|_\infty \leq b \) (since \( e_F \in F \cap T_F \)). Rearranging the terms, we obtain
\[
\tau \leq \frac{\bar{x}}{bc_1 + \nu} = c_2 \bar{x},
\]
which guarantees \( \text{sign}(x_{i-1}^I) = \text{sign}(x_0^I) \). Outside \( I \), defining \( \text{Id}_j^J \) as the restriction from \( J \) to \( \tilde{J} \), we must have
\[
\Phi_j^{*} p_F = \text{sign} \left( \text{Id}_j^J \Phi_j^{-1} w - \tau v_j^I \right).
\]

From Lemma 1, we know that \( -\text{sign}(v_j^I) = \Phi_j^{*} p_F \), so that the condition is satisfied as soon as
\[
\left| \left( \Phi_j^{-1} w \right) \right|_j < \tau |v_{F,j}| \quad \forall j \in \tilde{J}.
\]
Noting $v = \min_{j \in J} |v_{F,j}|$, and

$$b' \triangleq \sup_{u \in F \cap T} \left\| \text{Id}_J \Phi_J^{-1} w_T \right\|_{\infty},$$

we get the sufficient condition for (3.24),

$$\left\| \text{Id}_J \Phi_J^{-1} w_T \right\|_{\infty} < \tau v, \quad F(w) < \tau \frac{v}{b'}.$$  \hspace{1cm} (c1a)

Note that in case of exact support stability, i.e., when $I = J$, then $\tilde{J} = \emptyset$ and condition (c1a) is automatically satisfied.

We can now verify (3.23). It is direct to verify that $x_{\tau_1}^J$ satisfies the equality part,

$$y_T - \Phi_T^J x_{\tau_1}^J = \tau e_F.$$  \hspace{1cm} (3.25)

Indeed, injecting the expression of $x_{\tau_1}^J$ in the left member and using Corollary 1 and the fact that $\Phi_J v_J \in \partial F^o(p_F)$ gives

$$y_T - \Phi_T^J x_0^J + \tau v_T + \Phi_T^J w_T$$

$$= y_T - \Phi_T^J x_0^J + \tau \Phi_T^J v_T - \Phi_T^J \Phi_T^J w_T$$

$$= \Phi_T^J (y - \Phi x_0 - w) + \tau \Phi_T^J v_T$$

$$= \tau e_F.$$ 

On $S_F$, the inequality part of (3.23) reads

$$F(y_S^F - \Phi_J^S x_{\tau_1}^J + \tau e_F) \leq \tau$$

Moreover, since by assumption, $\Phi_J v_F^J \in \text{ri}(\partial F^o(p_F))$, let $\tilde{F} \triangleq F - \Phi_J v_F^J$ and let $\tilde{F} = \gamma_{\tilde{F}}$ be the gauge associated with it. Then, by Proposition 1, it also reads

$$\tilde{F}(y_S^F - \Phi_J^S x_{\tau_1}^J - \tau \Phi_J^S v_F^J) \leq \tau.$$
Injecting the proposed expression for \( x^{J}_{\tau 1} \) gives
\[
\tilde{F}(y^{S_F} - \Phi_{J}^{S_F}(x^{J}_{0} - \tau v^{J}_{F} + \Phi_{J}^{T_F} w^{T_F}) - \tau \Phi_{J}^{S_F} v^{J}_{F}) \leq \tau,
\]
\[
F(w^{S_F} - \Phi_{J}^{S_F} \Phi_{J}^{T_F} w^{T_F}) \leq \tau.
\]

Let \( a > 0 \) be defined by
\[
a \triangleq \sup_{u \in F} \tilde{F}((P^{S_F} - \Phi_{J}^{S_F} \Phi_{J}^{T_F})u),
\]
which is finite because \((P^{S_F} - \Phi_{J}^{S_F} \Phi_{J}^{T_F})u \in S_{F}\) and \(0 \in \text{ri}(\bar{F} \cap S_{F})\). It is thus sufficient to require
\[
a F(w) \leq \tau. \tag{3.26}
\]

Finally, we get the sufficient condition for (3.23),
\[
F(w) \leq \frac{\tau}{a}, \tag{c1b}
\]
which, together with (c1a), gives the value of \( c_{1} \). This ensures that \( x_{\tau 1} \) is solution to \((P_{F}^{T}(\Phi x_{0} + w))\) and \( p_{F} \) solution to \((D_{F}^{T}(\Phi x_{0} + w))\), which concludes Case 1.

Remark 4. Note that from the definition of \( v^{J}_{F} \), we know that \( \Phi_{J} v^{J}_{F} \in \partial F^{o}(p_{F}) = F \cap \bar{S}_{F} \) but we have to assume \( \Phi_{J} v^{J}_{F} \in \text{ri}(\partial F^{o}(p_{F})) \). If \( \Phi_{J} v^{J}_{F} \in \text{rbd}(\partial F^{o}(p_{F})) \), then \( a = +\infty \) and no stability guarantee can be obtained (unless assuming more elaborate conditions on \( w \) instead of \( F(w) \)).

Case 2 \((t_{F} > |J|)\) Recall that, for \( Z \in \mathbb{R}^{t_{F} \times t_{F} - |J|} \), an ONB of \( \text{Ker}(\Phi_{J}^{T_F}) \), the mapping \( e_{p}^{T_{F}} : \mathbb{R}^{t_{F} - |J|} \to \mathbb{R}^{t_{F}} \) is defined by
\[
e_{p}^{T_{F}}(z) = T_{F}^{*} \nabla_{T_{F}} F^{o}(T_{F}^{*} Z z + p_{F}).
\]
Note that $e_p^{T_F}(0) = e_F^{T_F}$. We must prove that there exists a mapping $w^{T_F} \mapsto z(w^{T_F})$ and a neighborhood $\mathcal{W} \subseteq \mathbb{R}^{t_F}$ of $0$ such that for all $w^{T_F} \in \mathcal{W}$, using $v_p^{I}(w^{T_F}) = \Phi_j^{T_F+} e_p^{T_F}(z(w^{T_F}))$,

$$x_{\tau_2}^{J}(w^{T_F}) = x_0^{J} + \Phi_j^{T_F+} w^{T_F} - \tau v_p^{J}(w^{T_F}), \quad (3.27)$$

together with

$$\hat{p}^{T_F}(w^{T_F}) = p_F^{T_F} + Z z(w^{T_F}), \quad (3.28)$$

form an optimal primal-dual pair $(x_{\tau_2}(w^{T_F}), \hat{p}(w^{T_F}))$. In particular, this implicitly means that the model subspace $T_F$ is stable to noise. A particular case to keep in mind is when $F^o$ is differentiable at $p_F$, in which case $T_F = \mathbb{R}^m$. Let the mapping $\Gamma : \mathbb{R}^{t_F} \times \mathbb{R}^{t_F} \to \mathbb{R}^{t_F}$ be defined by

$$\Gamma((z, x^{J}), w^{T_F}) = \Phi_j^{T_F} x^{J} + \tau e_p^{T_F}(z) - w^{T_F} - \Phi_j^{T_F} x_0^{J}. \quad (3.29)$$

First, note that $\Gamma((z, x^{J}), w^{T_F}) = 0$ implies the equality part of (3.23). Also note that the noiseless solution satisfies $\Gamma((0, \bar{x}_\tau^{J}), 0) = 0$. The remaining of the proof divides in three parts. First, we prove that $\Gamma((z, x^{J}), w^{T_F}) = 0$ when $e_p^{T_F}$ is locally continuous and injective. Then we prove $\Gamma((z, x^{J}), w^{T_F}) = 0$ when $e_p^{T_F}$ is locally $C^1$ and give the first order expansion of the solution. Finally, we prove that the proposed solutions respect the optimality conditions.

**Without differentiability.** Let us begin by the case without differentiability. Assume that $e_p^{T_F}$ is locally continuous one-to-one in a neighborhood of $z = 0$. Then, invoking the *implicit functions theorem for one-to-one mappings without derivative* [Jittorntrum, 1978; Kumagai, 1980], there exists an open neighborhood $\mathcal{V}_1 \subseteq \mathbb{R}^{t_F}$ of $0$ such that there is a unique locally continuous one-to-one mapping $w^{T_F} \mapsto (z(w^{T_F}), x^{J}(w^{T_F}))$, such that $z(0) = 0$ and $x^{J}(0) = \bar{x}_\tau^{J}$, that respects $\Gamma((z(w^{T_F}), x^{J}(w^{T_F})), w^{T_F}) = 0$ for all $w^{T_F} \in \mathcal{V}_1$. 

With differentiability. Now let us assume that $e^T_p$ is locally $C^1$ in a neighborhood of $z = 0$. Let $\nabla_1 \Gamma \in \mathbb{R}^{t_F \times t_F}$ be the Jacobian matrix of $\Gamma$ in its first argument, $(x^J, z) \in \mathbb{R}^{t_F}$, given by

$$
\nabla_1 \Gamma((z, x^J), w^{T_F}) = \left( \tau \nabla e^T_p(z), \Phi^T_J \right),
$$

where $\nabla e^T_p(z) \in \mathbb{R}^{t_F \times (t_F - |J|)}$ is the Jacobian matrix of $e^T_p$ at $z$.

Note that the Jacobian matrix $\nabla_2 \Gamma \in \mathbb{R}^{t_F \times t_F}$ of $\Gamma$ in its second argument, $w^{T_F} \in \mathbb{R}^{t_F}$, is given by

$$
\nabla_2 \Gamma((z, x^J), w^{T_F}) = -\text{Id}.
$$

Assume that $\nabla_1 \Gamma$ is non-singular at point $((0, x^J), 0)$, i.e.,

$$
\nabla_1 \Gamma((0, \bar{x}^J_\tau), 0) = \left( \tau \nabla e^T_p(0), \Phi^T_J \right) = \left( \tau E_0, \Phi^T_J \right),
$$

is invertible. Then, invoking the classical implicit functions theorem [Dontchev and Rockafellar, 2014], there exists an open neighborhood $\mathcal{V}_1 \subseteq \mathbb{R}^{t_F}$ of $0$ such that there is a locally $C^1$ mapping $w^{T_F} \mapsto (z(w^{T_F}), x^J(w^{T_F}))$, such that $z(0) = 0$ and $x^J(0) = \bar{x}^J_\tau$, that respects $\Gamma((z(w^{T_F}), x^J(w^{T_F})), w^{T_F}) = 0$ for all $w^{T_F} \in \mathcal{V}_1$. Moreover, its Jacobian matrix, at $w^{T_F} = 0$, reads

$$
\nabla(z, x^J)(0) = \nabla_1 \Gamma((0, \bar{x}^J_\tau), 0)^{-1}.
$$

In particular, the Jacobian matrix of $x^J(w^{T_F})$, at $w^{T_F} = 0$, reads

$$
\nabla x^J(0) = \left( P_{\text{Ker}(\tau E_0)} \Phi^T_J \right)^+
$$

$$
= \left( (\text{Id} - E_0 E_0^+) \Phi^T_J \right)^+,
$$
and the Jacobian matrix of $z(w^T_F)$, at $w^T_F = 0$, reads

$$\nabla z(0) = \left( P_{\text{Ker}(\Phi^*_j F^T)} \tau E_0 \right)^+$$

$$= (ZZ^* \tau E_0)^+ = \frac{1}{\tau} (Z^* E_0)^+ Z^*$$

From there we can compute the Jacobian matrix, with respect to $w^T_F$, of $v^J_p(w^T_F) = \Phi^T_F + e^T_F(z(w^T_F))$ at $w^T_F = 0$. By using the chain rule, we get

$$\nabla v^J_p(0) = \Phi^T_F + \nabla e^T_F(z(0)) \nabla z(0)$$

$$= \frac{1}{\tau} \Phi^T_F E_0 (ZZ^* E_0)^+$$

Finally, putting the puzzle together, we get the first order expansion for $x^J_{\tau_2}(w^T_F)$, either from $\nabla x^J(0)$,

$$x^J_{\tau_2}(w^T_F) = x^J(0) + \nabla x^J(0) w^T_F + O(||w^T_F||^2)$$

$$= x^J_0 - \tau v^J_p + (Id - E_0^+ E_0) \Phi^T_F^+ w^T_F + O(||w^T_F||^2),$$

or from $\nabla v^J_p(0)$,

$$x^J_{\tau_2}(w^T_F) = x^J_0 + \Phi^T_F^+ w^T_F - \tau v^J_p(w^T_F)$$

$$= x^J_0 + \Phi^T_F^+ w^T_F - \tau v^J_p - \tau \nabla v^J_p(0) w^T_F + O(||w^T_F||^2)$$

$$= x^J_0 - \tau v^J_p + \Phi^T_F^+ (Id - E_0^+ (ZZ^* E_0)^+) w^T_F + O(||w^T_F||^2).$$

We chose to present this second formula in the theorem as it allows to easily isolate the difference with Case 1. Note that one should be able to verify that

$$\left( (Id - E_0^+ E_0^+) \Phi^T_F^+ \right)^+ = \Phi^T_F^+ (Id - E_0^+ (ZZ^* E_0)^+),$$

leading to equivalence between both expressions.
If the restricted polar gauge, $F^\circ : T_F \to \mathbb{R} : p^T_F \mapsto F^\circ(p^T_F)$, is locally $C^2$ around $p_F \in T_F$, we can use the covariant Hessian [Lewis and Zhang, 2013] (see also Footnote 4), and its matrix equivalent $H_0 \triangleq \nabla^2_{T_F} F^\circ(p_F) \in \mathbb{R}^{m \times m}$. For a fixed $p_F \in T_F$, the covariant Hessian with respect to $T_F$ is the bilinear auto-adjoint mapping, $\nabla^2_{T_F} F^\circ(p_F) : T_F \times T_F \to \mathbb{R}$, defined for all $u \in T_F$ by

$$\langle \nabla^2_{T_F} F^\circ(p_F) u, u \rangle = \frac{d^2}{dt^2} F^\circ(p_F + tu|_{T_F})|_{t=0}.$$

Because $T_F$ is a linear subspace (instead of a more general smooth manifold as in [Lewis and Zhang, 2013]), $\nabla^2_{T_F} F^\circ(p_F)$ is equivalent to a matrix in $\mathbb{R}^{m \times m}$ which trivially extends its definition to all $u \in \mathbb{R}^m$. Using the chain rule, we compute the Jacobian matrix,

$$E_0 = \nabla e^T_F(0) = T^*_F \nabla^2_{T_F} F^\circ(p_F) \nabla p(0) = T^*_F H_0 T_F Z.$$

**Conclusion of Case 2.** To conclude Case 2, it remains to show that the proposed pair is indeed solution by verifying (3.24) and the inequality part of (3.23). To achieve that, we inject the proposed (implicit) formula, exactly as for Case 1. However, since we already assumed $w^T_F \in \mathcal{V}_1$ for some neighborhood, it is somewhat pointless to look for explicit constants $c_1$ and $c_2$ as for Case 1. Moreover, now, $v^j_p(w^T_F)$ depends on the noise vector, and such explicit constants would be much harder to obtain. Observe, first, that the inequality part of (3.24) always holds because $\Phi^T_F \hat{p}^T_F(w^T_F) = 0$ and therefore,

$$\Phi^T_F \hat{p}^T_F(w^T_F) = \Phi^T_F p_F.$$

Second, note that, for $w^T_F = 0$, both remaining conditions are satisfied. In particular, the equality part of (3.24), now, reads

$$\Phi^T_F p_F = \text{sign}(x^0_F + \Phi^T_F w^T_F - \tau v^j_p(w^T_F)).$$
Since \( e_p(z(w^{T_F})) \) is locally continuous around 0, so is \( v^J_p(w^{T_F}) \). Therefore, there must exist an open neighborhood \( \mathcal{V}_2 \) around 0 (depending on \( \tau \)) such that for all \( w^{T_F} \in \mathcal{V}_2 \), the sign pattern remains identical.

The inequality part of (3.23) is only relevant when \( t_F < m \). It becomes

\[
F(w^{SF} - \Phi^T J_F w^{T_F} + \tau \Phi^T J_F v^J_p(w^{T_F}) + \tau e_p(z(w^{T_F}))) \leq \tau.
\]

In that case, remember that we assume that \( \Phi^T J_F v^J_p \in \text{ri}(\partial F^\circ(p_F)) \). It means that for \( w^{T_F} = 0 \), the inequality is strictly satisfied. Therefore, by local continuity of all the quantities involved, there must be an open neighborhood \( \mathcal{V}_3 \) around 0 (depending on \( \tau \)) such that for all \( w^{T_F} \in \mathcal{V}_3 \), the inequality remains satisfied. Taking \( \mathcal{W} = \mathcal{V}_1 \cap \mathcal{V}_2 \cap \mathcal{V}_3 \) concludes the proof of Theorem 5.

3.4 Numerical experiments

In order to illustrate support stability in Lemma 1 and Theorem 5, we address numerically the problem of comparing \( \text{supp}(x_\tau) \) and \( \text{supp}(x_0) \) in a CS setting where we adopt the \( \ell_\alpha \)-norm as the fidelity penalty function. Theorem 5 shows that \( \text{supp}(x_\tau) \) does not depend on \( w \) (as long as it is small enough); simulations thus do not involve noise. All computations are done in Matlab, using CVX [Grant and Boyd, 2008, 2014], with the MOSEK solver at “best” precision setting to solve the convex problems. We set \( n = 1000, m = 900 \) and generate 200 times a random sensing matrix \( \Phi \in \mathbb{R}^{m \times n} \) with \( \Phi_{ij} \sim \text{iid } \mathcal{N}(0, 1) \). For each sensing matrix, we generate 60 different \( s \)-sparse vectors \( x_0 \) with support \( I \) where \( s \triangleq |I| \) varies from 10 to 600. The non-zero entries of \( x_0 \) are randomly picked in \( \{\pm 1\} \) with equal probability. We adopt the notation, \( J_{\ell_\alpha} \triangleq \text{sat}(\Phi^* p_{\ell_\alpha}) \) (see (3.4)), in order to draw a distinction between the extended supports associated to different values of \( \alpha \). Note that it only
depends on $\text{sign}(x_0)$ so that the magnitude of the non-zero entries of $x_0$ will only affect the bounds in (3.9). For each case, we verify that $x_0$ is identifiable and for $\alpha \in \{1, 2, \infty\}$ (which correspond to $\beta \in \{\infty, 2, 1\}$), we compute the minimum $\ell_\beta$-norm certificate $p_{\ell_\alpha}$, solution to (3.3) and in particular, the support excess $J_{\ell_\alpha} \triangleq \text{sat}(\Phi^* p_{\ell_\alpha}) \setminus I$. It is important to emphasize that there is no noise in these simulations and that $J_{\ell_\alpha}$ is independent of $\tau$. As long as the hypotheses of the theorem are satisfied, we can predict that $\text{supp}(x_\tau) = J_{\ell_\alpha} \subset I$ without actually computing $x_\tau$, or choosing $\tau$, or generating $w$. Put differently, those experiments are valid for any value of $\tau$ and any realization of $w$, respecting the hypotheses of the theorem.

We define a support excess threshold $s_e \in \mathbb{N}$ varying from 0 to $\infty$. On Figure 3.4 we plot the probability that $x_0$ is identifiable and $|J_{\ell_\alpha}|$, the cardinality of the predicted support excess, is smaller or equal to $s_e$. It is interesting to note that the probability that $|J_{\ell_\infty}| = 0$ (the bluest horizontal curve on the right plot) is 0, which means that even for extreme sparsity ($s = 10$) and a relatively high $m/n$ rate of 0.9, the support is never predicted as perfectly stable for $\alpha = \infty$ in this experiment. We can observe as a rule of thumb, that a support excess of $|J_{\ell_\infty}| \approx s$ is much more likely. In comparison, $\ell_2$ recovery provides a much more likely perfect support stability for $s$ not too large and the expected size
3.4 Numerical experiments

Figure 3.5 (best observed in color) Sweep over \( \frac{1}{\alpha} \in [0, 1] \) of the empirical probability (bright yellow is probability 1, dark blue is probability 0) as a function of \( s \) that \( x_0 \) is identifiable and \(|\tilde{J}_{\ell_\alpha}| \leq s_e\) for three values of \( s_e \). The dotted red line indicates \( \alpha = 2 \).

of \( \tilde{J}_{\ell_2} \) increases slower with \( s \). Finally, we can comment that the support stability with \( \ell_1 \) data fidelity is in between. It is possible to recover the support perfectly but the requirement on \( s \) is a bit more restrictive than with \( \ell_2 \) fidelity. As previously noted, Lemma 1 and its proof remain valid for smooth loss functions such as the \( \ell_\alpha \)-norm when \( \alpha \in ]1, \infty[ \). Therefore, it makes sense to compare the results with the ones obtained for \( \alpha \in ]1, \infty[ \). On Figure 3.5 we display the result of the same experiment but with \( 1/\alpha \) as the vertical axis. To realize the figure, we compute \( p_{\ell_\alpha} \) and \( \tilde{J}_{\ell_\alpha} \) for \( \alpha \) corresponding to 41 equispaced values of \( 1/\alpha \in [0, 1] \). The probability that \(|\tilde{J}_{\ell_\alpha}| \leq s_e\) is represented by the color intensity. The three different plots correspond to three different values for \( s_e \). On this figure, the yellow to blue transition can be interpreted as the maximal \( s \) to ensure, with high probability, that \(|\tilde{J}_{\ell_\alpha}| \) does not exceeds \( s_e \). It is always (for all \( s_e \)) further to the right at \( \alpha = 2 \). It means that the \( \ell_2 \) data fidelity constraint provides the highest support stability. Interestingly, we can observe that this maximal \( s \) decreases gracefully as \( \alpha \) moves away from 2 in one way or the other. Finally, as already observed on Figure 3.4, we see that, especially when \( s_e \) is small, the \( \ell_1 \) loss function has a small advantage over the \( \ell_\infty \) loss.
3.5 Conclusion

In this chapter, we provided sharp theoretical guarantees for stable support recovery under small enough noise by $\ell_1$ minimization under a fidelity constraint defined by a finite coercive gauge of the residual.

Those results extend well known results for the usual Euclidean fidelity to more general convex constraints, including, amongst other, $\ell_\alpha$-losses for $\alpha \geq 1$, mixed norms and the nuclear norm. Unlike the classical setting where the $\ell_2$ data loss is smooth and twice differentiable, our analysis reveals the difficulties arising from non-smoothness, which necessitated a novel proof strategy.

Though we focused here on $\ell_1$ minimization, our analysis can be extended to more general regularization terms as was done in [Vaiter et al., 2014] for the $\ell_2$ penalty. More perspectives and open questions are discussed at the end of the thesis, in Chapter 6.
Chapter 4

Online Convolutional Dictionary Learning for Multimodal Imaging

Computational imaging methods that can exploit multiple modalities have the potential to enhance the capabilities of traditional sensing systems. In this chapter, we propose a new method that reconstructs multimodal images from their linear measurements by exploiting redundancies across different modalities. Our method combines a convolutional group-sparse representation of images with TV regularization for high-quality multimodal imaging. We develop an online algorithm that enables the unsupervised learning of convolutional dictionaries on large-scale datasets that are typical in such applications. We illustrate the benefit of our approach in the context of joint intensity-depth imaging. The work presented in this chapter has been accepted at the IEEE International Conference on Image Processing 2017 and a preprint is available [Degraux et al., 2017].
4.1 Introduction

Multimodal imaging systems acquire several measurements of an object using multiple distinct sensing modalities. Often, the data acquired from the sensors is jointly processed to improve the imaging quality in one or more of the acquired modalities. Such imaging methods have the potential to enable new capabilities in traditional sensing systems, providing complementary sources of information about the object. As mentioned in Chapter 2, some applications of multimodal imaging include remote sensing [Dalla Mura et al., 2015], combining modalities such as synthetic aperture radar, LIDAR, optical and thermal range, multispectral and hyperspectral devices; biomedical imaging, for instance [Fatakdawala and et al., 2013], where fluorescence lifetime imaging, ultrasound backscatter microscopy, and photoacoustic imaging are combined for imaging cancer in live patients, or [Guérit et al., 2016], where CT based geometrical priors are used for blind deconvolution of PET images; and high-resolution depth sensing [Diebel and Thrun, 2005] (also considered in this work, see Section 4.1.2 for more references) where information from an intensity image is used to increase a depthmap resolution.

We consider a joint imaging inverse problem with multiple noisy linear measurements

\[ y_\ell = \Phi_\ell x_\ell + w_\ell, \]  

(4.1)

where for each modality \( \ell \in [L], \) \( y_\ell \in \mathbb{R}^{m_\ell} \) denotes the corresponding measurement vector, \( x_\ell \in \mathbb{R}^{n} \) denotes the unknown image, \( \Phi_\ell \in \mathbb{R}^{m_\ell \times n} \) denotes the sensing matrix, and \( w_\ell \in \mathbb{R}^{m_\ell} \) denotes the noise in the measurements. The images \( \{x_\ell\}_{\ell \in [L]} \) correspond to the same physical object viewed from different modalities. For example, each \( x_\ell \) may represent a different color channel, spectral band, or a type of sensor. For simplicity, we assume that the desired dimension of the images is the same across all modalities and that acquisition devices are perfectly registered. The
key insight used in our paper is that information about a single modality exists, in some form, in other modalities. This information can be exploited to improve the quality of multimodal imaging, as long as it can be extracted from the measurements.

4.1.1 Main Contributions

In this work, we propose a novel approach based on jointly sparse representation of multimodal images. Specifically, we are interested in learning data-adaptive convolutional dictionaries for both reconstructing and representing the signals. The convolutional dictionary learning is performed in an unsupervised way, i.e., given only the linear measurements of the signals. The main benefits of a convolutional approach are that the resulting dictionary is translation invariant and leads to a sparse representation over the entire image. This, however, comes with the increase in the computational cost, which we address by developing a new online convolutional dictionary learning method suitable for working with large-scale datasets. This online formulation behaves, in some way, as a dynamical system, illustrated in Figure 4.1 (see Section 4.2 for more explanations), where internal state variables are used to keep track of previously reconstructed signals, in a memory-efficient way. If a non-stationary data stream is given to the online learning algorithm, the latter is able to automatically adapt the convolutional dictionary to the changing data distribution. Our key contributions are summarized as follows:

• We provide a new formulation for multimodal computational imaging, incorporating a convolutional joint sparsity prior and a TV regularizer. In this formulation, the high resolution images are determined by solving an optimization problem, where the regularizer exploits the redundancies across different modalities.

• We develop an online convolutional dictionary learning algorithm. By accommodating an additional TV regularizer in the cost, the algorithm is able to learn the convolutional dictionary in an unsupervised
fashion, directly from the noisy measurements. The dictionary is convolutional which provides several advantages over a patch-based approach, such as translation invariance and the fact that the size of the atoms does not restrict the size of the associated regularized inverse problem. The online nature of the algorithm makes it appealing for applications that entail very large datasets or dynamically streamed, possibly non-stationary, datasets.

• We validate our approach for joint intensity-depth imaging on two numerical experiments. The first one compares the proposed approach with two competitive intensity-depth fusion algorithms, in an extensive set of simulations over various examples coming from the Middlebury dataset [Scharstein et al., 2014], which provides accurate groundtruth depthmaps and high-resolution intensity images. The second experiment aims at demonstrating the dynamic capability of the online algorithm by presenting the results of reconstruction over a depth-intensity video sequence, during which the joint convolutional dictionary visibly improves itself, and the reconstructed depthmap quality, with time.
4.1.2 Related Work

Dictionary learning was introduced by the early work of Olshausen and Field [1996, 1997] in neuroscience, in which they discovered that it is possible to automatically extract, from natural images, a dictionary for sparse representations that mimics the known behavior of the primary visual cortex of mammals. With the introduction of efficient algorithms such as the K-SVD [Aharon et al., 2006], it has become a standard tool for various tasks in image processing such as denoising and demosaicking [Mairal et al., 2009], super-resolution via coupled dictionary [Yang et al., 2012], hyperspectral and multispectral image fusion [Wei et al., 2015], or phase-retrieval from the squared magnitude of complex linear measurements [Tillmann et al., 2016]. Our approach builds upon two prior lines of research. One is related to convolutional sparse representations, also known, from the perspective of machine learning, as translation-invariant sparse coding (with max pooling) [Yang et al., 2010], and deconvolutional networks [Zeiler et al., 2010]. Recent work in that line include the development of efficient algorithms, e.g., based on ADMM [Wohlberg, 2016] (see perspectives in Chapter 6). The other line of work is on online dictionary learning (or, from a more general point of view, matrix factorization), introduced by [Mairal et al., 2010], and recently extended in [Mensch et al., 2016a,b, 2017] for partially observed samples. Online learning relies on an input stream of data samples rather than utilizing the full dataset at once. This allows, among other things, to handle huge datasets. Since our method relies on TV regularization, introduced in [Rudin et al., 1992], it is also related to TV-based imaging algorithms such as the fast gradient-based algorithm from [Beck and Teboulle, 2009b], used in this work, the mode general approach of [Afonso et al., 2010], and the recent parallel proximal algorithm from [Kamilov, 2017]. Specifically, our method is derived from the patch-based dictionary learning algorithm from [Mairal et al., 2010], introduced in Section 2.2.5, and uses the popular FISTA, described in
Section 2.2.4 for reconstructing images from measurements. Our method is validated on the problem of joint intensity–depth imaging, also considered in [Diebel and Thrun, 2005], using Markov random chains; in [Kopf et al., 2007], as a joint bilateral upsampling; in [Kamilov and Boufounos, 2016], where several frames of a video sequence are used in a motion adaptive regularization; in [He et al., 2013], which introduced a guided filtering algorithm; and in [Castorena et al., 2016], which used a weighted TV approach, also useful for automatic registration. Our method is compared against those last two methods in Section 4.3. Let us highlight that addressing the joint intensity–depth imaging problem with dictionary learning has also been done, in particular, in [Tosic and Drewes, 2014; Tosic and Frossard, 2009] which uses traditional sparse coding and, more recently, [Liu et al., 2016] which uses convolutional dictionaries for representing multiple modalities. Our approach extends earlier work by performing multi-modal image reconstruction with convolutional dictionaries and developing a dedicated online unsupervised learning algorithm for large-scale settings.

4.2 Proposed Method

4.2.1 Problem Formulation

The underlying assumption in our approach is that a jointly sparse convolutional model can accurately approximate the images \{x_\ell\} as

\[ x_\ell \approx D_\ell \alpha_\ell \triangleq \sum_{k=1}^{K} d_{\ell k} \ast \alpha_{\ell k}, \quad (4.2) \]

where \{d_{\ell k}\} is the set of \(LK\) convolutional filters in \(\mathbb{R}^{p}\), \(\ast\) denotes convolution, and \{\alpha_{\ell k}\} is the set of coefficient maps in \(\mathbb{R}^{\hat{n}}\) (see Section 4.2.4 for the link between \(n\) and \(\hat{n}\)). Note that \(D_\ell\) and \(\alpha_\ell\) denote the concatenation of all \(K\) dictionaries and coefficient maps, respectively. Given the complete dictionary \(D = (D_1, \ldots, D_L)\), we can define our imaging
problem as the following joint optimization,

\[ (\hat{x}, \hat{\alpha}) = \arg\min_{x, \alpha} \{ \mathcal{C}(x, \alpha; D \mid y, \Phi) \}, \]

where the cost function \( \mathcal{C} \) is given by

\[ \mathcal{C}(x, \alpha; D \mid y, \Phi) \triangleq \frac{1}{2} \| y - \Phi x \|_2^2 + \frac{\rho}{2} \| x - D\alpha \|_2^2 \]

\[ + \lambda \| \alpha \|_{2,1} + \tau R(x), \]

with \( y \triangleq (y_1^T, \ldots, y_L^T)^T, x \triangleq (x_1^T, \ldots, x_L^T)^T, \) and \( \alpha \triangleq (\alpha_1^T, \ldots, \alpha_L^T)^T\) and \( \Phi \triangleq \text{diag}(\Phi_1, \ldots, \Phi_L) \) denoting the block diagonal sensing matrix. The first quadratic term in (4.4) measures the data-fidelity, while the second controls the approximation quality of the dictionaries.

The first regularization term

\[ \| \alpha \|_{2,1} \triangleq \sum_{k=1}^{K} \sum_{j=1}^{\hat{n}} \| \alpha_{.kj} \|_2 \]

imposes group- or joint-sparsity of coefficients across \( L \) modalities. Here, \( \alpha_{.kj} \in \mathbb{R}^L \) denotes the vector formed by the aligned entries of the coefficient maps associated with kernel \( k \) for every modality \( \ell \). Specifically, this regularizer promotes the co-occurrence of image features, encoded
by the dictionary $D$, in all the modalities as schematized in 1-D on Figure 4.2.

The second regularizer in (4.4) corresponds to the isotropic TV penalty [Rudin et al., 1992],

$$R(x) \triangleq \sum_{\ell=1}^{L} \sum_{j=1}^{n} \| L [L x]_j \|_2,$$

(4.6)

where $L$ denotes the discrete gradient operator. Unsupervised learning of dictionaries from $y$ is complicated when the imaging problem is ill-posed. The goal of including the TV regularizer is to assist this learning. In practice, we observed significant improvement in quality when TV was included, both during learning and reconstruction. Finally, the positive constants $\rho$, $\lambda$, and $\tau$ are parameters controlling the tradeoff between the data fidelity and regularization.

Notice that, in this formulation, the regularized signal, $x$, and the sparse coefficient maps, $\alpha$, are only tied together by the quadratic term, $\frac{\rho}{2} \| x - D \alpha \|_2^2$, referred to as the coupling term. The difference with traditional sparse coding is that $x$ is not a fixed, given sample but is as much an optimization variable as $\alpha$. Therefore, $x$ is jointly regularized through that “weak” coupling term and through the TV-regularizer $R(x)$, while from the viewpoint of the sparse code, $\alpha$, the coupling term acts as a fidelity cost function and the regularization term is $\| \alpha \|_{2,1}$. Using the two-level splitting approach of (4.4) has several advantages over a more classical “synthesis” formulation (with a –non classical– additional analysis regularizer), such as

$$\frac{1}{2} \| y - \Phi D \alpha \|_2^2 + \lambda \| \alpha \|_{2,1} + \tau R(D \alpha),$$

where the hard constraint $x = D \alpha$ is implicitly imposed. First, the proposed cost function allows more granularity, thanks to the coupling
parameter $\rho$. Note that, imposing $x = D\alpha$ can be viewed as a limit case\footnote{It is, however, not recommended in practice to set extremely high values for $\rho$ as it would likely destabilize the numerical scheme.} where $\rho \to +\infty$. Second, computing the proximal operator associated to the regularizer, $\lambda \|\alpha\|_{2,1} + \tau R(D\alpha)$, would anyway require to use a splitting method (see, e.g., [Combettes and Pesquet, 2011]). In comparison, as described hereafter, the prox of the separable function, $\lambda \|\alpha\|_{2,1} + \tau R(x)$, is easy to compute.

Finally, it also has computational advantages, linked to the convolutional nature of $D$, as explained further in Section 4.2.4.

The joint optimization program in (4.3) is a convex problem. To solve it, we use the monotonic variant of FISTA [Beck and Teboulle, 2009b] described in Algorithm 1 in Section 2.2.4. In particular, we split $\mathcal{E}(x, \alpha; D|\Phi, y)$ into its smooth quadratic part,

$$\frac{1}{2}\|y - \Phi x\|^2_2 + \frac{\rho}{2}\|x - D\alpha\|^2_2,$$  \hspace{1cm} (4.7)

and the non-smooth regularization term that is separable in $x$ and $\alpha$,

$$\lambda \|\alpha\|_{2,1} + \tau R(x).$$  \hspace{1cm} (4.8)

The proximal operator associated with $\lambda \|\alpha\|_{2,1}$ is equal to

$$\left[\text{prox}_{\lambda \|\cdot\|_{2,1}}(\alpha)\right]_{k.j} = \left(\|\alpha_{k.j}\|_2 - \lambda\right) + \frac{\alpha_{k.j}}{\|\alpha_{k.j}\|_2}. \hspace{1cm} (4.9)$$

While the proximal of TV does not have a closed-form solution, it can be efficiently implemented [Beck and Teboulle, 2009b].

On a final note, let us remark that since, in general, $D\hat{\alpha} \neq \hat{x}$, the proposed method, in fact has two possible outputs: $D\hat{\alpha}$ or $\hat{x}$. This was omitted on Figure 4.1 for the sake of simplicity but another “output stream” arrow could be drawn from the sparse code itself.
4.2.2 Online Convolutional Dictionary Learning Algorithm

Suppose the input data is streamed so that at every time step $t \in \mathbb{N}$ we get a pair $(y^{[t]}, \Phi^{[t]})$. The learning procedure attempts to minimize (4.4) for all $t$, jointly for $x, \alpha$ and $D$. Specifically, let $J^t(D) \triangleq \min_{x,\alpha} \left\{ \mathcal{C}(x, \alpha; D | y^{[t]}, \Phi^{[t]}) \right\}$, then this amounts to solving

$$\min_{D \in \mathcal{D}} \left\{ \mathbb{E} \left[ J^t(D) \right] \right\}, \quad (4.10)$$

with respect to $D$, where the expectation is taken over $t$. Note that, to compensate for scaling ambiguities, we restrict the optimization of $D$ to a closed convex set $\mathcal{D}$. Specifically, $\mathcal{D}$ is the set of convolutional dictionaries that have kernels in the $\ell_2$ ball, i.e., $\|d_{\ell k}\|_2 \leq 1$.

The joint optimization program in (4.10) is difficult to solve directly. Thus, we use an alternating minimization procedure. In particular, at iteration $t$, given the current dictionary $D^{[t-1]}$, and a new pair of data $(y^{[t]}, \Phi^{[t]})$, we first solve

$$\left( x^{[t]}, \alpha^{[t]} \right) \leftarrow \arg\min_{x,\alpha} \left\{ \mathcal{C}(x, \alpha; D^{[t-1]} | y^{[t]}, \Phi^{[t]}) \right\}, \quad (4.11)$$

using the method presented in Section 4.2.1. Then, following the principle of [Mairal et al., 2010], we use all the previous iterates and chose $D$ to minimize a surrogate of $\mathbb{E} \left[ J^t(D) \right]$ given by

$$\frac{1}{t} \sum_{i=1}^{t} \mathcal{C}(x^{[i]}, \alpha^{[i]}; D | y^{[i]}, \Phi^{[i]}). \quad (4.12)$$

This second step, performed using a block gradient descent on the kernels $\{d_{\ell k}\}$, is described in Section 4.2.3. The complete learning algorithm is summarized in Algorithm 4.
Algorithm 4 Online Convolutional Dictionary Learning

1: procedure ONLINECDL
2: Input: Stream of data \( t \mapsto (y[t], \Phi[t]) \), initial dictionary \( D[0] \).
3: \( M[0] \leftarrow 0; \ b[0] \leftarrow 0; \)
4: while streaming data, do
5: Draw a pair \((y[t], \Phi[t])\) from the stream;
6: Sparse coding step:
7: \((x[t], \alpha[t]) \leftarrow \arg\min_{x, \alpha} \{ C(x, \alpha; D[t-1] | y[t], \Phi[t]) \}; \)
8: Update memory (4.18) and (4.19):
9: \( b[t] \leftarrow (1 - \frac{1}{t}) b[t-1] + \frac{1}{t} A[t] x[t]; \)
10: \( M[t] \leftarrow (1 - \frac{1}{t}) M[t-1] + \frac{1}{t} A[t] A[t]; \)
11: Dictionary update (4.14), (4.15), (4.17) initialized by \( D[t-1] \):
12: \( D[t] \leftarrow \arg\min_{D \in \mathcal{D}} \left\{ G^t(D | M[t], b[t]) = \frac{1}{2t} \sum_{i=1}^t \|x[i] - D\alpha[i]\|_2^2 \right\} \)

4.2.3 Dictionary update

Keeping \( x[i] \) and \( \alpha[i] \) fixed for \( 1 \leq i \leq t \), the only term in \( C \) that depends on \( D \) is the quadratic coupling penalty \( \frac{p}{2} \|x[i] - D\alpha[i]\|_2^2 \). Therefore, we can equivalently minimize \( \frac{1}{2t} \sum_{i=1}^t \|x[i] - A^i d\|_2^2 \), for each modality \( \ell \).

Since everything is separable in \( \ell \), in the remainder we drop the subscript for notational clarity. Note that, since the convolution operation is commutative and the \( \alpha[i] \) are fixed, we can rewrite

\[
D\alpha[i] = \sum_{k=1}^K d_k \ast \alpha_k[i] = \sum_{k=1}^K \alpha_k[i] \ast d_k = A[i] d, \quad (4.13)
\]

where \( A[i] = (A^i_1, \ldots, A^i_K) \in \mathbb{R}^{n \times Kp} \) is the sum-of-convolutions linear operator and \( d = (d^T_1, \ldots, d^T_K)^T \). In order to minimize

\[
G^t(d) = \frac{1}{2t} \sum_{i=1}^t \|x[i] - A[i] d\|_2^2,
\]
subject to $\|d_k\|_2 \leq 1$, as in [Mairal et al., 2010], we apply a projected block-coordinate gradient descent. The algorithm starts for $s = 0$ with $d_{t,0}^{[t,0]} \leftarrow D^{[t-1]}$ and iteratively applies the following two steps for all $k \in [K]$,

$$\tilde{d}_{k}^{[t,s]} \leftarrow d_{k}^{[t,s-1]} - \frac{1}{L_k} \nabla d_k G^t \left( \begin{array}{c} \vdots \\ \tilde{d}_{k}^{[t,s]} \\ \vdots \\ d_{k}^{[t,s-1]} \\ \vdots \end{array} \right)$$  \hspace{1cm} (4.14)$$

$$d_{k}^{[t,s]} \leftarrow \frac{\tilde{d}_{k}^{[t,s]}}{\max\{\|\tilde{d}_{k}^{[t,s]}\|, 1\}}$$  \hspace{1cm} (4.15)$$

until convergence or until a maximum number of iterations is reached. Note that $\nabla d_k$ denotes the partial gradient

$$\nabla d_k G^t(d) = \frac{1}{t} \sum_{i=1}^t A_k^{[i]}^* (A_k^{[i]} d - x^{[i]}),$$  \hspace{1cm} (4.16)$$

and $L_k \triangleq \left\| \frac{1}{t} \sum_{i=1}^t A_k^{[i]}^* A_k^{[i]} \right\|_2$ is the Lipschitz constant of $\nabla d_k G^t(d(d_k))$, with respect to the dictionary atom $d_k$. Importantly, we can take advantage of all the previous iterates to compute this gradient. Indeed, we can write it as

$$\nabla d_k G^t(d) = M_k^{[t]} d - b_k^{[t]},$$  \hspace{1cm} (4.17)$$
where the *memory vector* and the symmetric *memory matrix*,

\[
b^{[t]} = \left( \begin{array}{c} b^{[t]}_1 \\ \vdots \\ b^{[t]}_K \end{array} \right) \triangleq \frac{1}{t} \sum_{i=1}^{t} A^{[i]*} x^{[i]},
\]

\[
M^{[t]} = \left( \begin{array}{c} M^{[t]}_1 \\ \vdots \\ M^{[t]}_K \end{array} \right) \triangleq \frac{1}{t} \sum_{i=1}^{t} A^{[i]*} A^{[i]},
\]

with \(M^{[t]}_k \triangleq (M^{[t]}_{k,1}, \ldots, M^{[t]}_{k,K}) = (A^{[i]*}_k A^{[i]}_1, \ldots, A^{[i]*}_k A^{[i]}_K)\), are computed recursively from the previous iterates as

\[
b^{[t]} \leftarrow \frac{t-1}{t} b^{[t-1]} + \frac{1}{t} A^{[t]*} x^{[t]},
\]

\[
M^{[t]} \leftarrow \frac{t-1}{t} M^{[t-1]} + \frac{1}{t} A^{[t]*} A^{[t]}.
\]

The aforementioned Lipschitz constant thus reads \(L^{[t]}_k = \|M^{[t]}_{k,k}\|_2\). For the sake of illustration, in Figure 4.1 and Algorithm 4, we used the notation \(G^{t}(D|M^{[t]},b^{[t]}) \triangleq G^{t}(d) = \frac{1}{2t} \sum_{i=1}^{t} \|x^{[i]} - A^{[i]} d\|_2^2\) to emphasize the fact that \(M^{[t]}\) and \(b^{[t]}\) entirely define it and suffice to optimize it over \(D\).

### 4.2.4 Implementation details

**Convolutional implementation.** A naive implementation of the online convolutional learning algorithm would require to store \(M^{[t]}\), which is a dense symmetric \(K_p \times K_p\) matrix. By definition, \(A^{[i]}_k \in \mathbb{R}^{n \times p}\) is a convolution operator whose columns and rows are restricted to match the size of its input \(d_k\) and output \(x\), respectively. Similarly, \(A^{[i]*}_k \in \mathbb{R}^{p \times n}\) is a restricted convolution operator whose kernel \(\tilde{\alpha}^{[i]}_k\) is the flipped version of \(\alpha^{[i]}_k\). Therefore, if the size of \(\alpha^{[i]}_k\) is chosen such that it implements a full convolution, then the operator \(A^{[i]*}_k A^{[i]}_k\) corresponds to a restricted
convolution with a kernel $\tilde{\alpha}_{k}^{[i]} * \alpha_{k'}^{[i]}$. Importantly, the restrictions to a $p \times p$ operator imply that only part of the kernel, of size proportional to $p$, is actually used. We denote this effective part by $R_{p}(\tilde{\alpha}_{k}^{[i]} * \alpha_{k'}^{[i]})$ where $R_{p}$ is the selection operator. This implies that $M_{kk'}^{[t]}$ is itself a convolution operator with kernel

$$c_{k,k'}^{[t]} \triangleq \frac{1}{t} \sum_{i=1}^{t} R_{p}(\tilde{\alpha}_{k}^{[i]} * \alpha_{k'}^{[i]}). \quad (4.22)$$

Therefore, an efficient implementation of $M^{[t]}$ only requires to store or convolve with those $K^2$ kernels. Note that this computational trick is a big argument in favor of decoupling $\frac{1}{2} \|y - \Phi x\|_2^2$ from $\frac{1}{2} \|x - D \alpha\|_2^2$. By contrast, using the same technique to minimize $\frac{1}{2} \|y - \Phi D \alpha\|_2^2$ instead, requires the explicit storage of a dense symmetric $Kp \times Kp$ matrix.

When $\Phi$ is a mask and the dictionary is not convolutional, Mensch et al. [2016b] adopt a different strategy which consists in approximating the surrogate function.

**Unknown boundary conditions.** Similarly to [Almeida and Figueiredo, 2013] and to what is done in Chapter 5, we used *unknown boundary conditions* for safely implementing all the convolutions as *full convolutions*\(^2\), taking full advantage of the FFT diagonalization, without suffering from zero-boundaries artifacts. Even though this is omitted in the rest of the chapter for the sake of clarity, the sensing matrix $\Phi$ in (4.4) was, in fact, replaced by $\tilde{\Phi}R_{\tilde{n}}$, where $R_{\tilde{n}} \in \mathbb{R}^{\tilde{n} \times n}$ is a restriction operator from an extended domain of size $\tilde{n}$ (i.e., including the unknown boundaries) such that $n > \hat{n} > \tilde{n}$. The effective operator\(^3\) $\tilde{\Phi} \in \mathbb{R}^{m \times \tilde{n}}$ denotes the actual linear measurements of an acquisition instrument with a field of view of size $\tilde{n}$, which is the region of interest for the reconstruction, and

\(^2\)themselves implemented as zero-padded circular convolutions.

\(^3\)We loosely abuse that notation in the rest of the chapter, by using $\Phi$ where, rigorously speaking, we should write $\tilde{\Phi}$. This is meant as a simplification to streamline the discussion. The same goes for the experiments where $R_{\tilde{n}}$ is implicitly applied before evaluating the PSNR.
corresponds to the valid part of the convolutions. Said differently, the optimization variable \( x \in \mathbb{R}^n \) has a border that is never observed by \( \tilde{\Phi} \), resulting from the full convolutions, \( d_k \ast \alpha_k \), of the kernels \( d_k \) with the coefficient maps \( \alpha_k \), for \( k \in [K] \). That border is thrown away at the end of the reconstruction by simply selecting \( R_{\tilde{n}} \in \mathbb{R}^n \).

As an illustrating 1-D example, schematized on Figure 4.3, consider a target signal \( x_0 \) of size \( \tilde{n} \), and a single-atom dictionary \( d \) of size \( p \). The extended optimization variable, \( x \), of size \( n \) is such that \( x_0 = R_{\tilde{n}} x \) but the boundary, outside of \( R_{\tilde{n}} \), is not a priori known. The coefficient map, \( \alpha \), of size \( \hat{n} = n - p + 1 \), is so that the full convolution \( d \ast \alpha \) is of size \( \hat{n} + p - 1 = n \) and the valid convolution \( d \ast \alpha \) (see nomenclature) is of size \( \hat{n} - p + 1 = \tilde{n} \).

**Data centering.** Patch-based dictionary learning is known to be more effective when the input data is centered to have zero mean. While this is not required, it is common to pre-process the data by removing the means of the training patches [Mairal et al., 2014]. Accordingly, we first estimate a local mean, *i.e.*, a low-pass component, \( x^{lo} \) of the data \( x \). The remaining component is the high-pass image \( x^{hi} \triangleq x - x^{lo} \). We aim to learn the sparse synthesis model \( x^{hi} \approx D\alpha \). To do so, we adapt the method above, replacing the coupling term in (4.4) by \( \frac{\rho}{2} \| x - x^{lo} - D\alpha \|_2^2 \). Specifically, we use a mask-aware low-pass filter to estimate \( x^{lo} \) from \((y, \Phi)\) where \( \Phi \) is a mask operator. Let \( \mathcal{L}(\cdot) \) be a low-pass filter and \( \varphi \in \{0, 1\}^n \) the mask (*i.e.*, the diagonal of \( \Phi \)). Then, for every pixel \( [\mathcal{L}(\varphi)]_j > 0 \), we compute \( [x^{lo}]_j = \frac{[\mathcal{L}(y)]_j}{[\mathcal{L}(\varphi)]_j} \). We use the nearest neighbor
interpolation to fill the remaining pixels where \([\mathcal{L}(\varphi)]_j = 0\). Note that this method for estimating \(x_{\text{lo}}\) from \(y\) depends on the structure of the sensing operator \(\Phi\) and is not always directly applicable.

**Mini-batch extension.** Similarly to [Mairal et al., 2010], we enhance the algorithm by performing the sparse coding step on a few samples between every dictionary update. This is particularly appealing when we can process several samples in parallel. It is worth noting that there is a tradeoff between the number of input samples per mini-batch and their size. A big input sample contains a lot of redundant information and leads to a slower coding step. Conversely, a mini-batch of a few small but diverse samples is faster and can mitigate the effect of a single iteration biasing towards a specific scene.

**Forgetting factor.** In the first few iterations of the algorithm, the initial dictionary may not be informative for effective sparse representation. Thus, the corresponding coefficient maps \(\alpha^{[t]}\) are inaccurate, compared to coefficient maps computed with later iterates. Consequently, we also introduce a forgetting factor \(\gamma \geq 0\), which allocates more weight to newer samples than to older ones. In practice, during the update of the memory vector (4.20) and matrix (4.21), we weigh the old and new ones respectively by \(\theta[\ell] \triangleq (1 - \frac{1}{t})^{1+\gamma} \) and \(1 - \theta[\ell]\).

### 4.3 Experimental Evaluation

To evaluate our multimodal imaging method, we focus on joint intensity-depth reconstruction. We consider two modalities \((L = 2)\), where \(\Phi_1 = \text{Id}\) is the sensing matrix associated to an intensity image and \(\Phi_2 = \text{diag}(\varphi)\) is a random mask setting to zero a fraction of the depth map pixels. To generate \(y_\ell\), Gaussian noise of 30dB PSNR is added to \(\Phi_\ell x_\ell\) on both modalities. Our quality metric is the prediction PSNR over the missing pixels of \(x_2\). Since the reconstruction method (4.3)
optimizes over two coupled but distinct variables ($\hat{x}, \hat{\alpha}$), two options are available for the prediction: either using $\hat{x}$ or using $D\hat{\alpha} + x^{lo}$, with $x^{lo}$, the low-pass image defined in Section 4.2.4. In our experiments the second solution provides better performance. We rely only on subsampled data for both training and reconstruction, and set the number of convolutional kernels to $K = 32$ and size $p = 15 \times 15$. All simulations were implemented in Python 3.5, using the NumPy 1.11, SciPy 0.18 and OpenCV 3.1 libraries.

**Comparison with other methods**

In this experiment, we used the *Middlebury* dataset [Scharstein et al., 2014] to simulate a joint depth–intensity multimodal imaging scenario. The 2014 Stereo datasets with ground truth from Middlebury College is a high quality set of 23 still life scenes, each comprising two color images from a pair of stereo (left and right) cameras, and an accurate depth groundtruth. It is, originally, meant for evaluating stereo-matching algorithms, *i.e.*, methods for extracting the depth from a pair of images sharing the same baseline. For the purpose of this work, the 23 scenes were pre-processed so as to only keep the left camera image, which was de-saturated (*i.e.*, to only keep the intensity information instead of color), and the high resolution groundtruth depthmaps. In order to harmonize the process, the scenes, that we shall refer to as frames, were all reshaped to $480 \times 672$ pixels (with the standard OpenCV bicubic reshaping method) per modality.

We first train a global dictionary using 160 mini-batches, each consisting of 8 randomly selected patches of size $45 \times 46$ in all the 23 scenes from the dataset. Then, to process each specific frame, the dictionary is specialized with 120 mini-batches of 8 patches sampled from that frame. Finally, the full ($480 \times 672$) frame is reconstructed by solving (4.3) with the specialized dictionary. Note that, in contrast to patch-based dictionary learning, the input patches are not necessarily of the same size as the dictionary kernels. In fact, the full frame could be used to train
Table 4.1 Average PSNR for various subsampling rates on 23 images from the Middlebury 2014 dataset.

<table>
<thead>
<tr>
<th>Method</th>
<th>2×</th>
<th>3×</th>
<th>4×</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linear</td>
<td>30.72 dB</td>
<td>30.39 dB</td>
<td>29.97 dB</td>
</tr>
<tr>
<td>Guided Filter</td>
<td>34.18 dB</td>
<td>33.46 dB</td>
<td>32.76 dB</td>
</tr>
<tr>
<td>Weighted TV</td>
<td>36.85 dB</td>
<td>35.58 dB</td>
<td>34.77 dB</td>
</tr>
<tr>
<td>Proposed</td>
<td><strong>37.13 dB</strong></td>
<td><strong>35.73 dB</strong></td>
<td><strong>34.88 dB</strong></td>
</tr>
</tbody>
</table>

the convolutional dictionary. However, using medium-sized patches instead, helps to accelerate the training.

Table 4.1 compares the average performance of our reconstruction procedure with three alternative approaches, listed in increasing complexity: linear interpolation, guided filtering [He et al., 2013], and weighted TV [Castorena et al., 2016]. Note that, similarly to our method, the guided filter and the weighted TV both use intensity information as a guide for depth estimation. Linear interpolation is presented as an intuitive indicator of the order of magnitude of the improvement. Figure 4.4 gives a more detailed summary of the same experiment. Even though it is often closely followed by weighted TV, the proposed method outperforms the other fusion methods on the majority of scenes.

Figure 4.5 shows the specialized dictionary and the reconstruction results for the Backpack image. One can clearly recognize the typical image features manifested in the learned kernels. Some kernels present sharp edges or corners. Others show more elaborate gridded features. Most paired kernels have striking similarities between depth and intensity in terms of shape, orientation, and alignment. These results highlight the ability of our method to learn multimodal convolutional dictionaries directly from noisy and subsampled data.

Parameters. The parameters were hand-tuned using heuristics for every method in order to achieve the best average performance and have a fair comparison. For our method we found that the parameters $\sigma^2 = 10^{-3}$,
Figure 4.4 Methods comparison on the Middlebury dataset [Scharstein et al., 2014] with 50%, 33% and 25% random sampling rate and some noise (30 dB). The vertical axis is the reconstruction PSNR (in dB) of the corresponding method. The base line is guided filtering. A star indicates the best algorithm for each scene (no star means that guided filtering performed best). The horizontal axis indicates the official name of each scene. The sorting order (as a function of the performances of the guided filter) is arbitrary.
Figure 4.5 The top-left and top-right images are the intensity and depth modalities from the Backpack image. White pixels correspond to missing pixels due, for example, to occlusions. Bottom-left image shows the trained dictionary with each pair of corresponding intensity-depth kernels grouped. Bottom-right shows the reconstruction of the region highlighted in red for $2 \times$ subsampling.

$\rho = 2/(n\sigma^2), \lambda = 200/(\hat{n}K\sqrt{L}), \gamma = 10$ give satisfactory results. For each modality, a different $\tau_1 = 40/n, \tau_2 = 80/n$ was set to account for difference in the amount of texture in depth and in intensity. An additional parameter, $\mu_\ell = 2/(m_\ell\sigma^2)$, multiplies the data-fidelity term to account for differences in the number of pixels measured in each modality. They were omitted in the text for the sake of clarity. The parameters for the guided filter were $r = 2$ and $\gamma_{\text{gdf}} = 10^{-3}$. The parameters for weighted TV were $\tau_{\text{TV}} = 125 m_2\sigma_n^2/n$ and $\gamma_{\text{TV}} = 1$ where the weights on each
pixel of the depth gradient are computed as $w_j = \exp(-\gamma_{TV} \|Lx_1\|_2)$.

The tolerance threshold for the stopping criterion (for every algorithm) was set to $\epsilon = 10^{-4}$ on the relative difference of the objective functions between two iterates. The maximum number of iterations was set to 100 for the inner loop computing the proximal operator of TV and for the dictionary update, 500 for weighted TV and 5000 for solving (4.3) in our method. The maximum number of iterations for weighted TV and for (4.3) was rarely reached.
Online training on a video

In order to demonstrate the online capability of our learning algorithm, we use the Road intensity–depth video sequence from the 3D Video dataset from [Zhang et al., 2009]. This dataset comprises three video sequences (one of which is called Road) acquired with a laterally moving camera. For each video sequence, the authors of the dataset have computed the depthmap associated to each frame using the bundle optimization method described in their paper, which optimizes over the whole video sequence. For the purpose of this demonstration, we consider that their method has provided a sufficiently accurate depth video sequence, which is thus considered, here, as a depth groundtruth. A pre-processing step consisted in desaturating the color video sequence and selecting a 512 × 512 region of interest.

For this experiment, we use a mask with 2× subsampling and add a 30 dB Gaussian noise. We start with a dictionary filled with Dirac deltas. On each 512 × 512 frame of the video, we extract 8 randomly chosen 50 × 50 patches and perform one mini-batch iteration of the learning algorithm. Then, we reconstruct the full frame using the current dictionary, $D[t]$, and compare with the performances obtained with the initial dictionary, $D_0$, of deltas ($\Delta\text{PSNR} = \text{PSNR}_{D[t]} - \text{PSNR}_{D_0}$). Figure 4.6 presents the evolution, as the video is streamed, of the PSNR improvement. As visually evident, the energy of the residual tends to decrease as the dictionary improves, especially near object edges (observe the road sign which appears in all three frames). Note that a temporary drop of quality might be observed when an unexpected feature appears in the scene, corresponding to a change in the data distribution. If the dictionary is able to adapt to the new feature, the decrease is then compensated.
4.4 Conclusion

The contributions of this chapter combine several ideas. The original online dictionary learning method [Mairal et al., 2010] introduced the key concept of online learning. However, their patch-based approach used directly observed data. The method we proposed here, on the other hand, uses convolutional atoms which requires a careful implementation to preserve the algorithm efficiency. We apply joint sparsity of learned convolutional atoms as a way to transfer information between different modalities. Moreover, we introduce an additional prior to regularize the inverse problem associated to indirectly observed data, making it an unsupervised method. Those are considerable generalizations that enhance the potentiality of the original online algorithm.

Nevertheless, the proposed method has a few important shortcomings. First, it is considerably more complex than the two methods against which it is compared. The learning process constitutes a computational overhead and the reconstruction process is significantly slower than guided filter and slightly more expensive than weighted TV. Second, the proposed method has numerous parameters and can be quite sensitive to their choice. The high computational cost of learning and reconstruction may constitute an obstacle to using cross validation for parameter selection.

We have, however, demonstrated on a numerical example that it is able to outperform those competitors in some situations. The proposed method is able to capture intensity-depth feature correspondences by learning appropriate joint atoms. Moreover it has the interesting capability of continuously adapting and improving, an interpretable, specialized, convolutive dictionary. Once it has been learned, the dictionary can potentially be re-purposed for other regularization tasks on similar images.

Some perspectives for future work are given at the end of the thesis in Chapter 6.
CHAPTER 5

MULTISPECTRAL COMPRESSIVE IMAGING STRATEGIES USING FABRY-PÉROT FILTERED SENSORS

In this chapter, we propose and study two acquisition device architectures for MS compressive imaging. Both proposed imaging techniques use a dedicated sensor which integrates Fabry-Pérot spectral filters to record measurements in several narrow bands of the spectrum of light.

The first, and most straightforward technique is, depending on the point of view, akin to inpainting, demosaicking or super-resolution. It aims at filling the gaps, introduced by spectral filtering at pixel level, in the multispectral volume. The second proposed imaging technique adds a layer of complexity allowing higher compression ratios. By introducing spatial light modulation before the sensor, it implements random convolutions, which have several benefits in terms of mathematical and computational properties (see Chapter 2).

Interestingly, we use the same approach in both cases to solve the associated inverse problem. Specifically, we use an analysis signal prior in a convex formulation. The convex problem is then solved with an efficient algorithm. Through extensive numerical simulations, we explore various tradeoffs in different realistic setups. We highlight some practical guidelines and limitations for both methods allowing to proceed towards an informed implementation.
Part of the material presented in this chapter has been published in [Degraux et al., 2014, 2015]. The rest will be submitted for a future journal publication.

5.1 Introduction

Multispectral (MS) imaging consists in capturing the light intensity, $X_0(u, v, \lambda)$, of an object or scene as it varies along its 2-D spatial coordinates $(u, v)$ and over different wavelengths $\lambda$, i.e., along a 1-D domain representing the light spectrum as measured into a few wavelength intervals or bands. This information is sampled in a 3-D data volume or cube which allows for accurate classification or segmentation of specific constituents in an object or scene from their spectral profile. Hence, MS imagers find diverse applications in remote sensing, optical sorting, biology, and astronomy.

A classic approach to MS imaging is to spatially or spectrally multiplex the MS cube over a 1-D or 2-D imaging sensor (i.e., a Focal Plane Array (FPA)). This is done by scanning the cube, so that parts or slices of the latter are relayed and acquired by the sensor during several snapshots (for a review see, e.g., [Sellar and Boreman, 2005]). Such systems require either spectral filters, dispersive elements (e.g., refractive prisms or diffraction gratings) which separate spatially the light spectrum, or high-precision mechanical parts to physically scan the object or scene being acquired. Those methods thus entail trade-offs between system-level complexity and cost, spectral and spatial resolution, and acquisition time.

Recently, single-snapshot MS imagers were developed to rapidly acquire a MS cube, thus avoiding motion artifacts and enabling video sampling rates [Hagen and Kudenov, 2013]. Among such imagers, we focus on those based on FP filtered sensors [Geelen et al., 2014; Lambrechts et al., 2014], i.e., standard CMOS imaging sensors on top of which an array of per-pixel spectral filters is overlaid. This recent
technology generalizes RGB filter arrays [Bayer, 1976] (i.e., a three-band filter bank) to filter banks using an arbitrary number of bands (e.g., a few tens with narrowband profiles [Lambrechts et al., 2014]). However, the sensor’s pixel count is then partitioned by the number of bands. The array thus imposes a reduction in the MS cube’s spatial resolution.

5.1.1 Main Contributions

This chapter investigates MS imaging strategies based on CS (see Chapter 2), in order to cope with the spatial resolution limitation of snapshot spectral imagers. Specifically, our work contributes to advancing the field of compressive MS imaging in the following senses:

- We propose two snapshot MS imaging architectures, Multispectral Volume Inpainting (MSVI) and Multispectral Random Convolution (MSRC), which involve no dispersive elements in the optical path, as they maintain a relatively low system-level complexity and minimal number of components. They are designed with a low-resolution, low-pixel count FP filtered sensor jointly with the principles of CS.
- The first architecture, MSVI, leverages generalized inpainting, as discussed in [Degraux et al., 2015], to provide a simple integration of the FP filtered sensor in a computational imaging scheme. This architecture entails no mixing in the spectral or spatial domains, as it performs a spatio-spectral subsampling of the MS cube. It only relies on the redundancy of MS cubes to obtain a high-resolution recovery from their subsampled measurements when the scenes follow a sparse model prior.
- The second architecture, MSRC, leverages random convolution, as discussed in [Degraux et al., 2014], to provide spatial-domain CS by means of an out-of-focus Coded Aperture (CA), while preserving the spectral-domain resolution fixed by the FP filters. This choice stems from the observation that the low number of narrowband filters (e.g., 16) deposited on our FPA already sets a low spectral-domain resolution.
The analysis is also paired with a discussion on the analysis-sparse signal prior, the associated convex optimization formulation and fine-tuned ADMM algorithm (introduced in Chapter 2) that scales well for the recovery of MS cubes given the measurements of both architectures.

Both architectures are numerically compared in terms of achievable recovery performances, while we also discuss their complexity trade-offs and design guidelines to integrate these schemes into realistic imaging systems.

Our findings and their extensive numerical exploration corroborate how a conspicuous reduction in the number of measurements w.r.t. the Nyquist-rate representation of $X_0(u, v, \lambda)$ is made possible by both architectures while preserving high Signal-to-Noise Ratio (SNR).

### 5.1.2 Related Work

#### Compressive Spectral Imaging

The use of CS for MS imaging schemes dates back to [Gehm et al., 2007; Sun and Kelly, 2009]. The most popular application of CS to spectral imaging is the Coded Aperture Snapshot Spectral Imaging (CASSI) framework [Arce et al., 2014; Gehm et al., 2007; Wagadarikar et al., 2008, 2009; Wu et al., 2011] with its many variants summarized hereafter. Single-disperser CASSI focuses the scene on a CA that is used to provide random modulation, i.e., the random spatial occlusion of areas of the scene. A shearing of the spatio-spectral information in the scene is then provided by a refractive prism and the processed light is recorded by a standard imaging sensor. While [Wagadarikar et al., 2008] shows high spectral-domain accuracy after image recovery at the expense of a relatively lower one in the spatial domain, double-disperser CASSI [Gehm et al., 2007] achieves opposite performances in terms of spatial versus spectral accuracy, but uses several optical components with non-trivial geometric alignment and system calibration.
A close line of work in [Correa et al., 2014, 2015] proposes a snapshot spectral imaging architecture featuring spectral filters. Their imager is still based on CASSI, but the shearing is followed by spectral filters which provide spatio-spectral subsampling in a random arrangement similar to that discussed in [Degraux et al., 2015]. As the filters are wideband, they integrate the sheared cube w.r.t. the spectral domain, while our FP filters are narrowband. Non-snapshot spectral imaging architectures based on CS were also recently proposed [August et al., 2013; Fowler, 2014; Sun and Kelly, 2009].

Indeed, CASSI and its variants target a relatively large number of bands and are intrinsically capable of achieving high spectral resolution thanks to dispersive optics. However, when the spectral domain is well represented by a few tens of bands, the application of spectral-domain mixing (i.e., of spectral-domain CS) will be less effective than spatial-domain mixing, especially when considering FP filtered sensors with a few tens of narrowband filters (e.g., [Geelen et al., 2015]) whose high selectivity excludes spectral-domain super-resolution attempts. Hence, we specifically focus on schemes that target only a few bands; for which the application of a dispersive element is made unnecessary by the use of FP filters.

**Compressive Imaging by Random Convolution**

Since its introduction, CS has been envisioned to provide image acquisition at reduced sensor resolution [Romberg et al., 2008] or shorter acquisition times [Lustig et al., 2007] (see also the tutorial in [Willet et al., 2011], and references therein). In particular, our work is related to CS by random convolution, by which the sensing operation performs a spatial-domain convolution of the scene by a random coded aperture (i.e., an array of square apertures randomly positioned on an opaque screen) which acts as a linear random spatial filter, i.e., as in coded aperture imaging [Dicke, 1968; Fenimore and Cannon, 1978]. More recently, the sensing operator that corresponds to randomly selecting pixels from
the output of a random convolution was shown to comply with the theoretical requirements of CS in [Rauhut, 2010; Rauhut et al., 2012; Romberg, 2009]. This operation was also featured in recent imaging architectures [Björklund and Magli, 2013; Jacques et al., 2009; Marcia et al., 2009]. The appeal of convolution-based schemes is in that they allow for a fast application of the sensing operator, i.e., the compressive measurements can be formed in one frame of a full imaging sensor, as opposed to single-pixel camera designs [Chan et al., 2008; Romberg et al., 2008; Sun and Kelly, 2009], where the compressive measurements are collected by multiplexing in time a single photo-detector. However, the intrinsic snapshot capability of random convolution architectures is generally paid by the higher correlation between compressive measurements, because of their spatial adjacency and the necessity of taking into account optical-level non-idealities such as the PSF of the optical elements.

For the acquisition of MS cubes, we propose the application of spatial-domain CS by random convolution using an extension of the low-complexity snapshot imaging architecture proposed by Björklund and Magli [Björklund and Magli, 2013]. In particular, their architecture uses a coded aperture placed out-of-focus to provide random convolution using essentially the latter aperture, an imaging sensor and relay optics. Due to its configuration, this convolution scheme will, however, suffer from the effect of non-negligible spatial-domain correlation between adjacent compressive measurements as sampled on the imaging sensor. To counter this effect, the authors of [Björklund and Magli, 2013] advocated the use of a larger coded aperture, and low fill-factor or large-pitch detector arrays.
5.2 Preliminaries

5.2.1 Fabry-Pérot Filtered Sensors

The class of imaging sensors at the core of this work are comprised of a standard, panchromatic CMOS sensor designed to operate in the visible light (VIS) range of wavelengths (i.e., 400–700 nm), on which a layer of monolithically integrated Fabry-Pérot interferometers [Fabry and Perot, 1901] is deposited. The FP interferometer is an optical element comprised of two partially reflecting mirror layers separated by a translucent layer; this ensemble forms an optical resonator or cavity, whose spectral-domain effect on the incoming light is that of a band-
pass filter. Its center wavelength is controlled by the mirror distance, while the filter’s Full Width at Half Maximum (FWHM) (essentially its pass-band) is controlled by the reflectivities of the two mirror layers (the physics of this resonator are well described in [Hernandez, 1986]). The cavity can be designed to yield a very narrow-band profile (about 10 nm) and to cover the area of a single CMOS sensor pixel.

Once the filter profiles are designed, the FP filters can be distributed either in a mosaic layout [Geelen et al., 2014], where a group of different cavities per-pixel is repeated in a $4 \times 4$ or $5 \times 5$ mosaic pattern (Figure 5.1b), or by partitioning the sensor in a tiled layout, where the sensor is partitioned in large areas with the spectral filter for a specific wavelength deposited on top of them [Geelen et al., 2013] (Figure 5.1d). While it is possible to envision architectures that use tiled layouts [Degraux et al., 2014; Geelen et al., 2014] we here focus on mosaic designs, as they will allow a reduction of the correlation between measurements taken on adjacent sensor pixels. Such a sensor, [Geelen et al., 2014], was designed and prototyped at IMEC$^1$ and will be referred to as IMEC’s sensor in the following. The use of an external spectral cutoff filter removing anything outside the VIS range allows one to obtain a filter bank such as the one depicted in Figure 5.1a. These profiles were generated for illustration purposes based on calibration measurements taken at IMEC, with a monochromator laser sweep on the mosaic sensor. The raw data was post-processed to only keep the main lobe of each filter response. In particular, smaller secondary modes, which sometimes appear at harmonic locations of the spectrum (see [Hernandez, 1986] and [Geelen et al., 2014] for another example), were removed from the figure for the sake of clarity. In this work, we consider an idealized situation, ignoring such secondary modes.

Hereafter, our architectures will consider a sensor featuring a 16-band filter bank with uniformly spaced center wavelengths between 470 – 620 nm in the VIS range, either placed in a regular $4 \times 4$ mosaic pat-

$^1$See www.imec-int.com.
tern (Figure 5.1b), or with a randomly-assigned arrangement, hereafter dubbed random pattern (Figure 5.1c). Manufacturing such a random pattern should not pose any major difficulty compared to the mosaic pattern. Yet, to our knowledge, that random pattern has not been implemented in practice. In simulations (Section 5.3.2), the random pattern is generated by permuting, uniformly at random over the entire FPA, the assigned locations of the filters. The same number of each filter as on the mosaic sensor is thus preserved, but their arrangement is scrambled across the sensor.

5.2.2 Forward model and analysis prior

Both studied architectures entail a noisy linear acquisition process, summarized by the usual generic forward model (see Section 2.1.1),

\[ y = \Phi x_0 + w. \]  

(5.1)

Let \( \mathbf{X}_0 \in \mathbb{R}^{n_u \times n_v \times n_\lambda} \) represent a discretized MS cube in its 2-D spatial and 1-D spectral domains, equivalently represented by its vectorization \( x_0 \triangleq \text{vec}(\mathbf{X}_0) \in \mathbb{R}^n, n = n_un_vn_\lambda \). The linear sensing operator is represented in matrix form by \( \Phi \in \mathbb{R}^{m \times n} \) where \( m \triangleq m_um_v \). It yields a set of compressive measurements that are captured by the sensor array, \( \mathbf{Y} \in \mathbb{R}^{m_u \times m_v} \) or in vectorized form \( y \triangleq \text{vec}(\mathbf{Y}) \in \mathbb{R}^m \). The additive noise vector \( w \in \mathbb{R}^m \) is considered bounded in \( \ell_2 \)-norm by \( \tau \).

As any optical system design based on regularized inverse problems, the schemes we develop in this chapter must leverage a prior model for the signal being acquired, e.g., by means of a transform that maps the MS cubes to points in a low-complexity set. As opposed to the learned synthesis dictionary of Chapter 4, we choose, here, to use a non-adaptive analysis transform (see Chapter 2, Section 2.1.2). In order to jointly enforce that low-complexity prior in both spatial and spectral domains, we separately apply two linear transforms in the spatial and spectral domains, as denoted respectively by \( \mathbf{A}_{uv} \) and \( \mathbf{A}_{\lambda} \). This amounts
to constructing a separable transform by the Kronecker product $A \triangleq A_{uv} \otimes A_\lambda$.

For the spatial domain transform, $A_{uv}$, we chose a 2-D UDWT which forms a shift-invariant, separable, and overcomplete wavelet transform comprised of non-critically sampled Daubechies-4 wavelets in each direction [Mallat, 1999; Starck et al., 2007]. The scaling coefficients or so-called approximation level is inherently not sparse as it contains the low-pass approximation of the image. One option could be to simply discard the corresponding coefficients from the transform. We found however that the slowly varying spatial information tends to help at capturing well the redundancy between bands. Instead of throwing it away, we use a 2-D discrete cosine transform to concentrate the low-pass information in a few coefficients. The wavelet filters are chosen with length 8 and in 3 levels per direction, resulting in an analysis transform $A_{uv} \in \mathbb{R}^{10n_u n_v \times n_u n_v}$.

The 1-D discrete cosine transform is chosen for the spectral domain transform, $A_\lambda \in \mathbb{R}^{n_\lambda \times n_\lambda}$, given that we will be focusing on MS cubes captured from natural scenes, whose spectral profiles tend to be smooth.

### 5.2.3 Recovery Method

In order to invert (5.1) and find the estimate $\hat{x}$ of the MS cube, using the analysis-sparsity prior, we use the $\ell_1$-analysis [Candès et al., 2011] convex problem introduced in Section 2.2.3, with an additional range constraint $R \triangleq [x_{\text{min}}, x_{\text{max}}]^n$, which reads

$$\hat{x} \triangleq \arg\min_{x \in \mathcal{R}} \|Ax\|_1 \quad \text{s.t.} \quad \|y - \Phi x\|_2 \leq \tau.$$  \hspace{1cm} (5.2)

The constrained form of the problem has the advantage, over its penalized Lagrangian form, of being easily interpretable in terms of noise level. A good noise estimate or, for synthetic experiments, a noise oracle is sufficient for setting the regularization parameter $\tau$. This greatly sim-
5.2 Preliminaries

plifies the parameter tuning step which can otherwise be cumbersome in such a high dimensional setup.

For solving (5.2), we use Algorithm 2 (see Section 2.2.4), which is an adaptation of the ADMM to problems of the form of (2.20), i.e.,

\[
\min_z \sum_{j=1}^{J} g_j (H_j z). \tag{5.3}
\]

where \( g_j \) are convex l.s.c. functions and \( H_j \) are linear operators. In [Almeida and Figueiredo, 2013], in order to handle the boundary conditions, the authors propose to add the missing columns in the sensing operator and treat the border as unknown, as opposed to constant or periodic. Doing so, they stabilize the estimation of the convolutive filter in blind deconvolution, while recovering a block-circulant structure of the convolution operator. This allows to diagonalize the matrix with the FFT, as was done in Chapter 4 (see Section 4.2.4). Building over these ideas, as will be further explained for both architectures in Sections 5.3.1 and 5.4.3, we can add rows and columns to \( \Phi \) to accelerate the computations without actually deviating from the original image formation model. We define an extended sensing matrix \( \bar{\Phi} \in \mathbb{R}^{\bar{m} \times \bar{n}} \) (with \( \bar{m} \geq m \) and \( \bar{n} \geq n \)) and corresponding restriction operators \( R_m \in \{0, 1\}^{m \times \bar{m}} \) and \( R_n \in \{0, 1\}^{n \times \bar{n}} \) such that

\[
\Phi = R_m \bar{\Phi} R_n^*. \tag{5.4}
\]

The additional rows and columns of the extended matrix are chosen to make sure that \( \bar{\Phi}^* \bar{\Phi} + \mu \text{Id} \) is easy to invert for any \( \mu > 0 \). Furthermore, the analysis transform \( A \) described in the previous section is actually a scaled tight frame, i.e., there exists a diagonal weighting matrix \( \Omega \) such that \( A \triangleq \Omega \tilde{A} \) and \( \tilde{A}^* \tilde{A} = \text{Id} \). In order to make use of those interesting properties of \( A \) and \( \Phi \), we first define

\[
\tilde{\Omega} \triangleq \begin{bmatrix} \Omega \\ 0_{\bar{n} - n} \end{bmatrix} \quad \text{and} \quad \tilde{A} \triangleq \begin{bmatrix} \tilde{A} R_n \\ R_n^c \end{bmatrix}, \tag{5.5}
\]
where $R_n^c \in \{0, 1\}^{(\bar{n} - n) \times \bar{n}}$ is the complementary restriction of $R_n$ and $0_{n-\bar{n}}$ is the matrix of zeros of size $\bar{n} - n$ by $\bar{n} - n$. This ensures that the tight frame property $\bar{A}^* \bar{A} = \text{Id}$ is preserved. Let $\bar{x} \triangleq R_n^* x$, i.e., a zero-padded version of $x$, and let $\bar{\alpha} \triangleq \bar{A} \bar{x}$ and $\bar{z} \triangleq \bar{\Phi} \bar{x}$. Note that $\|\bar{\Omega} \bar{A} \bar{x}\|_1 = \|A x\|_1$ and $\|y - R_m \bar{\Phi} \bar{x}\|_2 = \|y - \Phi x\|_2$. Using the change of variable, $\bar{x} = R_n^* x$, and imposing $R_n^c \bar{x} = 0$, (5.2), equivalently reads

$$\hat{x} = R_n \underset{\bar{x} \in \mathbb{R}^{\bar{n}}}{{\text{argmin}}} \|\bar{\Omega} \bar{A} \bar{x}\|_1$$

s.t. $\|y - R_m \bar{\Phi} \bar{x}\|_2 \leq \tau$, $R_n \bar{x} \in \mathcal{R}$ and $R_n^c \bar{x} = 0$.

Let $\iota_{\|y - \cdot\| \leq \tau}$ denote the indicator function of the constraint set $\{z \mid \|y - z\| \leq \tau\}$; let $\iota_{\mathcal{R}}$ denote the indicator of the range constraint, $R_n \bar{x} \in \mathcal{R}$, and let $\iota_{\{0\}}$ be the indicator of the $\{0\}$ set, translating the constraint $R_n^c \bar{x} = 0$. In order to match the required form, (5.3), the problem (5.6) is then equivalently split in $J = 3$ functions, namely,

$$g_1(\bar{\alpha}) \triangleq \|\bar{\Omega} \bar{\alpha}\|_1 \quad \text{with} \quad H_1 \triangleq \bar{A}$$

$$g_2(\bar{z}) \triangleq \iota_{\|y - \cdot\| \leq \tau}(R_m \bar{z}) \quad \text{with} \quad H_2 \triangleq \bar{\Phi}$$

$$g_3(\bar{x}) \triangleq \iota_{\mathcal{R}}(R_n \bar{x}) + \iota_{\{0\}}(R_n^c \bar{x}) \quad \text{with} \quad H_3 \triangleq \text{Id}.$$ 

The corresponding proximal operators all admit a very simple closed form expression that can be efficiently evaluated. See Chapter 2 and references therein for more details on proximal operators. Moreover, we have

$$\sum_{j=1}^3 \mu_j H_j^* H_j = \mu_1 \left( \bar{\Phi}^* \bar{\Phi} + \frac{\mu_2 + \mu_3}{\mu_1} \text{Id} \right),$$

which, as will be specifically explained for both architectures, is easily invertible.

**Simulation parameters and dataset**

In every experiment presented in this chapter, we used $\mu_1 = 50 \frac{\rho}{\|\bar{\Phi}\|^2}$, $\mu_2 = \mu_3 = \rho$, with $\rho = 40$. The dynamic range of the images is normalized to $x_{\text{min}} = 0$ and $x_{\text{max}} = 1$. The maximum number of iterations was
set to $n_{\text{iter}} = 2000$ but the tolerance of $\text{tol} = 5.10^{-5}$ for the relative $\ell_2$-norm difference between two iterates was always achieved before $n_{\text{iter}}$. Note that the $\mu_j$ parameters were manually tuned in order to reach a reasonably fast convergence but they do not critically affect the recovery quality once convergence has been reached. The $\frac{\rho}{\| \Phi \|^2}$ factor in $\mu_1$ is a heuristic normalization of $\Phi^* \Phi$ compared to $\text{Id}$ in (5.7). For initializing the algorithm, we use a 3-D linear interpolation of $Y$ in the 3-D $(u, v, \lambda)$ space to get an estimate $Y_{\text{lin}} \in \mathbb{R}^{m_u \times m_v \times n_\lambda}$. Let $y_{\text{lin}} = \text{vec}(Y_{\text{lin}})$ so that $y = R_m y_{\text{lin}}$ (but $y_{\text{lin}} \neq R^*_m y$). Then we use Tikhonov regularization,

$$x_{\text{init}} \triangleq ((\Phi^* R_n + \tau^2 \text{Id})^{-1}) \Phi R_n y_{\text{lin}},$$

that we solve using the conjugate gradients algorithm with a tolerance of $10^{-3}$ and maximum ten iterations.

In order to provide a unified, quantitative and qualitative comparison framework, we used, for both methods, the same MS dataset. The dataset was created from eight of the 32 multispectral volumes of size $512 \times 512 \times 31$ from the CAVE dataset [Yasuma et al., 2010] by carefully selecting a $256 \times 256 \times 16$ region of interest (ROI) in each of them. In the spectral direction, the ROI is consistently 470 nm through 620 nm, which roughly corresponds to the actual FP filters from IMEC’s sensor. The spatial ROI was chosen in each image in order to capture the most interesting features. Two examples, namely balloons (centered at pixel $(255, 128)$), and chart and stuffed toys (centered at $(230, 280)$), are shown on Figure 5.2 and used throughout the rest of the chapter for qualitative comparisons. The other chosen samples, used to produce average PSNR curves, were feathers $(256, 256)$, jelly beans $(256, 256)$, glass tiles $(256, 256)$, stuffed toys $(256, 256)$, superballs $(200, 236)$, and beads $(256, 256)$. 
5.3 Multispectral Compressive Imaging by Generalized Inpainting

The first architecture, coined Multispectral Volume Inpainting (MSVI), is presented in this section. We start by describing in detail how the measurements are formed and recorded on the snapshot FP filtered sensor. Then, we compare two FP-filter layouts, mosaic and random, in a numerical simulated experiment. Lastly, we present actual results obtained when applying our method to data acquired with IMEC’s mosaic snapshot imager.

5.3.1 Image Formation Model

For this method, we must, first, choose a target spatial resolution (i.e., the spatial resolution of the reconstructed volume) $n_u \times n_v$, for discretizing the continuous reality of the observed scene, $X_0(u, v, \lambda)$, as briefly introduced in Section 5.1. In this work, in order not to make the
5.3 Multispectral Compressive Imaging by Generalized Inpainting

Interpolation operator
\( U_p \)
FP filters
\( M_\ell \)

\( X_0 \in \mathbb{R}^{n_u \times n_v \times n_\lambda} \)
Target

\( X_{\text{up}} \in \mathbb{R}^{m_u \times m_v \times n_\lambda} \)
Upscaled

\( Y \in \mathbb{R}^{m_u \times m_v} \)
Focal plane

Figure 5.3 MSVI forward model.

problem harder than it needs to be (see footnote), we choose the target resolution to be smaller (not larger) than the spatial pixel count of the sensor \( m_u \times m_v \), i.e., \( n_u \leq m_u \) and \( n_v \leq m_v \), even though \( m_u m_v = m \leq n = n_u n_v n_\lambda \). The spatial Nyquist rate, associated to the observed scene, \( X_0(u, v, \lambda) \), is then assumed to be achieved by that chosen resolution \( n_u \times n_v \). If it is not the case, i.e., if the scene is not spatially band limited, this assumptions entails the need for an optical low-pass filtering at the Nyquist frequency corresponding to an \( n_u \times n_v \) discretization grid\(^2\). This practice is common for stabilizing RGB demosaicking [Keelan, 2002]. This can be done for instance with birefringent filtering [Acharya et al., 1994] or by slightly defocusing the objective lens of the imager, assuming that the observed scene is well contained in the optical depth-of-field (DOF).

Let \( X_{\text{up}} \in \mathbb{R}^{m_u \times m_v \times n_\lambda} \) be an upscaled version of the scene \( X_0 \), i.e., matching the FPA pixel count \( m_u \times m_v \). Numerically, we can obtain this \( X_{\text{up}} \) by using a smooth and separable interpolation function, represented here by the linear operator \( U_p \in \mathbb{R}^{m \times n_u n_v} \) (applied separately to every band). For instance, in this work, we used a Lanczos interpolation kernel. Since the Nyquist rate is assumed to be achieved by \( n_u \times n_v \), both

\(^2\)Note that in principle, there is no hard limitation for choosing \( n_u \times n_v \) larger than \( m_u \times m_v \). However, to avoid aliasing, this time on the discretizing sensor grid, the low-pass cutoff must be at maximum that \( m_u \times m_v \) discretization grid’s Nyquist frequency. Recovering more pixels would then just result in an additional badly conditioned or ill-posed super-resolution problem, where there is little hope to recover the lost high frequencies.
\(X_{\text{up}}\) and \(X_0\) represent perfectly the same continuous reality \(X_0(u, v, \lambda)\). These considerations allow to de-couple the number of FPA pixels from the target scene resolution and therefore to choose the subsampling rate \(m/n\) by changing one or the other.

Let us now denote by \(M_\ell \in \{0, 1\}^{m \times m}\) the diagonal mask operator selecting all the FPA pixels corresponding to the FP filters of index \(\ell \in [n_\lambda]\). Since every FPA pixel is sampling one and only one spectral band, we have \(\sum_{\ell \in [n_\lambda]} M_\ell = \text{Id}\) and \(M_\ell M_{\ell'} = 0\) for \(\ell \neq \ell'\). Note that the concatenation (used hereafter), \(M \triangleq (M_1, \ldots, M_{n_\lambda})\) is, in effect, a restriction operator rather than a mask. The sensing matrix then reads \(\Phi \triangleq (\Phi_1, \ldots, \Phi_{n_\lambda})\) with \(\Phi_\ell \triangleq M_\ell \text{Up}\). This forward model is schematized on Figure 5.3.

Set aside the clear affiliation with inpainting, some links can be made with the random basis ensembles introduced, in the context of CS, in section 2.1.2. In the spectral direction, the sparsity basis is the DCT and the sampling basis, if we ignore the upsampling operator, is the canonical basis which is incoherent with the DCT and therefore a priori a good choice. On the other hand, in the spatial dimension, loosely speaking, the sparsity “basis” is a wavelet transform which is not maximally incoherent with the canonical basis. This gives a good intuition of why there is room for improvement and justifies the exploration of the second method in Section 5.4. However, rigorously extending the analogy to redundant wavelets in an analysis prior (as opposed to orthogonal wavelet synthesis) and taking the upsampling into account would require further work (see Chapter 2 for references to related work).

As explained in Section 5.2.3, we can ease the computations by adding rows and columns to \(\Phi\) (see (5.4)). In this case, one natural choice is to add back the columns that were deleted by the masks \(M_\ell\).
5.3 Multispectral Compressive Imaging by Generalized Inpainting

\( i.e., \) to choose \( R_n = \text{Id} \) and \( R_m = M = (M_1, \ldots, M_{n\lambda}) \) so that\(^3\)

\[
\Phi = \text{diag}_{n\lambda}(U_p).
\]

Therefore, \( \Phi^* \Phi + \mu \text{Id} \) is a separable sparse matrix which is easily precomputed and fast to invert, for example with the conjugate gradients algorithm. Even though the gain is less obvious than in the MSRC case discussed in Section 5.4.3, we found, empirically, that using this trick speeds up ADMM’s convergence compared to the direct use of \( \Phi \) (with \( R_m = \text{Id} \)). Let it be noted that in the case \( m_u = n_u \) and \( m_v = n_v \) (\( i.e., \) a subsampling rate of \( 1/n_{\lambda} \)), we have \( U_p = \text{Id} \) which makes the inversion step as trivial as a scalar multiplication by \((1 + \mu)^{-1}\).

5.3.2 Simulations

For each of the eight CAVE samples (see Section 5.2.3), we measure the performances of the reconstruction as a function of the subsampling rate \( m/n \). Since \( n \) is fixed, we explore four sensor configurations \( m_u \times m_v \) of increasing size, namely \( m_u = m_v \in \{256, 512, 768, 1024\} \), that is \( m/n \in \{\frac{1}{16}, \frac{1}{4}, \frac{9}{16}, 1\} \). Each recovery is conducted for two different FP filters configurations: the mosaic pattern (Figure 5.1b) and the random pattern (Figure 5.1c), which is a random permutation of all the filters on the sensor. To demonstrate the robustness of the method, the experiment is performed once with 40dB (low noise) and once with 20dB (higher noise) of input SNR of additive white gaussian noise on the measurements. On Figure 5.4, the plot shows, for each case, the average over the eight test samples of the Peak Signal-to-Noise Ratio (PSNR) in dB, expressed as a function of the Mean Squared Error (MSE) as \( \text{PSNR} \triangleq -10 \log_{10}(\text{MSE}) \) (since the dynamic range is 1).

One can see that at lower subsampling ratios, the random arrangement outperforms the mosaic sensor. This is particularly visible at high

\(^3\)We used the notation (see nomenclature), \( \text{diag}_n(A) \triangleq \begin{pmatrix} A & \cdots & A \end{pmatrix} \), to denote the block diagonal matrix formed by repeating \( n \) times \( A \) diagonally, without overlap.
input SNR. This indicates that the strong aliasing due to extreme subsampling is somehow mitigated by randomness. When the sampling ratio is 1, i.e., when the Nyquist rate is achieved, mosaic pattern sampling beats random sampling which, at this point, only introduces unnecessary scrambling. Note that both results in that case are above 50dB of PSNR and look visually perfect (not shown).

On Figure 5.5, we display the results for the two chosen samples (see Section 5.2.3) at \( m/n = 1/16 \) and with 40dB of input SNR. We deliberately chose this extreme low sampling rate to expose the most obvious differences between both sensor arrangements. In this figure, we first show the initialization point \( x_{\text{init}} \) as defined by (5.8). Note that in this case \( \Phi = \text{Id} \) so \( x_{\text{init}} = (1 + \tau^2)^{-1} y_{\text{lin}} \). Both arrangements give visually scrambled results, particularly on the outer 470 nm and 620 nm bands where less data-points are available to perform linear interpolation. The spectral error, depicted by the red areas on the spectral curves, is particularly large for pixel (B) on the balloons sample and for pixel (A) on the chart and stuffed toy sample. This is due to the fact that
Figure 5.5 Qualitative comparison between random and mosaic layout for MSVI. The patches correspond to zooms on regions of the cube (see Figure 5.2). The crosses (labelled (A), (B) and (C)) point to pixels whose spectra are represented in the 4th column. The light gray lines are reminders of the groundtruth and the red area represents the error for each curve.
those spatial locations are close to an edge or in a heavily textured region which makes linear interpolation unstable.

The results obtained by the proposed method, denoted \( \hat{x} \), are much more visually accurate. Edges and textures are more defined and one can almost guess the tiny numbers on the leftmost patch of the chart and stuffed toy sample, whereas it is totally impossible to distinguish anything from the linear interpolation result. As predicted by the PSNR curves of Figure 5.4, the random arrangement leads to smaller spectral error areas, in particular for the chart and stuffed toy sample. notice how the edge of the balloons, the horizontal bars of the chart and the stripes on the stuffed toy’s sleeve are well resolved. The mosaic arrangement leads to a “grid” artifact already observed in [Degraux et al., 2015], whereas the random arrangement, as one could expect, leads to seemingly unstructured random noise artifacts.

To mitigate all those effects, it might be preferable to target a higher sampling rate, e.g., \( m/n = 1/4 \). We chose not to display that test case as the artifacts are less obvious (though present), which makes the comparison harder. The presence of input noise also tightens the gap between both cases.

### 5.3.3 Experiments

On Figure 5.6, we present the results of an experiment carried on actual measurements recorded with IMEC’s mosaic sensor on two different test scenes. This imager has a resolution of \( 1024 \times 2048 \) pixels organized in a mosaic of \( 256 \times 512 \) identical \( 4 \times 4 \) macro-pixels; each with \( n_\lambda = 16 \) different FP filters at wavelengths of visible light (as in Figure 5.1b). For this experiment we restricted the measurements to a \( 512 \times 512 \) region (depicted by the white square on the false color preview images on top). The results on Figure 5.6 are compared qualitatively since no groundtruth is available. The subsampling rate \( m/n \), here is \( 1/4 \), i.e., we recover a volume with \( 256 \times 256 \times 16 \) voxels. The naive, super-pixel based, demosaicing method (top row), used in [Geelen et al., 2014],
Figure 5.6 Result of two experiments on real data acquired with IMEC’s sensor. Qualitative comparison between naive demosaicking (nearest neighbor), the initialization (5.8), and the proposed method. Top row, false color groundtruth, uses nearest neighbor. The crosses (labelled (A), (B) and (C)) point to pixels whose spectra are represented in the 4th column.
is clearly the worst. The middle row shows the result of the linear interpolation and Tikhonov regularization initialization method (5.8). Though visual quality improvement is obvious, the same “grid artifact” as in the simulations, appears and is particularly visible on the 551 nm band of the left sample (Leaves). As opposed to the simulations of Figure 5.5, where the sampling rate is more extreme (1/16), this artifact is almost completely removed with the proposed method (bottom row). This result strongly advocates for replacing the naive method with the proposed sparse regularization method if the computational resources allow it. Unfortunately, without the exact filters calibration profiles and the reference endmembers spectra, there is no way of evaluating if the spectral variations are accurately reconstructed. However, in light of the excellent results obtained at identical subsampling rate in simulation (see Figure 5.4) one can reasonably imagine that this is the case.

5.4 Multispectral Compressive Imaging by Out-of-Focus Random Convolution

This section focuses on the second proposed compressive architecture, the Multispectral Random Convolution (MSRC) device. As in the previous section, we start by describing the ideal image formation model. Then, we discuss specific design and implementation aspects linked to important non-idealities, such as attenuation and diffraction, that one can expect to encounter. Finally, we present simulation results that compare qualitatively and quantitatively the ideal case with two more realistic cases, including diffraction.

5.4.1 Image Formation Model

In order to define the MSRC forward model, we give here a description of the optical path, based on ideal, diffraction-free, geometrical optics. This follows the ideas originally introduced by Björklund and Magli [2013] in their panchromatic parallel compressive architecture.
The difference, here, is that we use the FP filtered sensor instead of a panchromatic sensor.

**Continuous model**  For a precise description, it is easier to express the object of interest in a continuous domain, *i.e.*, as the function of three (scalar) variables, $X_0(u, v, \lambda)$, briefly introduced in Section 5.1, instead of the familiar $X_0 \in \mathbb{R}^{nu \times nv \times n\lambda}$. In fact, in order to lighten the notations, we will consider only one spatial dimension, *i.e.*, instead of treating the full 3-D spectral density, $X_0(u, v, \lambda)$, we will use a simplified $X_0(u, \lambda)$. Note that everything is separable and can straightforwardly be extended back to two spatial dimensions by a simple cartesian product (or by viewing $u$ and associated, as position vectors in $\mathbb{R}^2$).

The sensing model is summarized in Figure 5.7 and explained as follows. We let the scene, modeled as an extended, incoherent source of light, be imaged, in focus, on an *image plane* by means of an ideal relay lens. This image plane is where the sensor would be placed in a classical camera. Instead, here, we consider that the (flipped) source $X_0(-u, \lambda)$ is
virtually located in this plane and radiates light in every direction. In particular, it illuminates an aperture plane placed at a distance $d$ along the optical axis $z$. There, we place a coded aperture, i.e., an array of $s_u \times s_v$ square elements that are either opaque or transparent (e.g., at random with equal probability). This optical element is modeled by its transmittance function

$$S(u) = \sum_{i=1}^{s_u} S_i \text{rect}\left(\frac{u}{\Delta_s} - (i - \frac{s_u}{2})\right)$$

with aperture pitch $\Delta_s$ and where $\text{rect}(v) = 1$ if $v \in [0, 1)$ and 0 elsewhere. By default, its $s_u$ symbols are $S_i \in \{0, 1\}$ (see, hereafter, the paragraph on complementary patterns for more on that). The coded aperture is assumed to have negligible effect in the spectral domain. From a ray optics perspective, the coded aperture is illuminated by replicas of the source, $X_0(u'(\vartheta) - u, \lambda)$ flipped and shifted by $u'(\vartheta) = d\tan(\vartheta)$, as relayed by bundles of parallel rays that propagate in the same direction, defined by the angle $\vartheta \in [-\pi/2, \pi/2]$ w.r.t. the optical axis. Thus, immediately after the coded aperture, any shift $u'$ corresponds to a spatially-modulated replica of the source, i.e., $S(u)X_0(u'(\vartheta) - u, \lambda)$ that propagates along the former direction. A lens is then placed in front of the coded aperture and used to focus the modulated light on the sensor. For an ideal thin lens with focal length $f$, all rays sharing the same direction $\vartheta$ converge on its focal plane at $u''(\vartheta) = f\tan(\vartheta)$. Thus, the spectral density observed at a point $u''$ on the focal plane is the sum of all the parallel rays intensity,

$$Y(u''(\vartheta), \lambda) = \int S(u)X_0(u'(\vartheta) - u, \lambda)du$$

$$= [S * X_0](u'(\vartheta), \lambda),$$

or equivalently,

$$Y(u, \lambda) = [S * X_0](\frac{d}{f}u, \lambda),$$
where $\ast$ denotes linear convolution in the spatial domain. We may now fix the sampling grid of the coded aperture by the angles, $\vartheta_i = \arctan((i - m_u) \frac{\Delta s}{d})$, for $i \in [m_u]$. With this, the rays corresponding to the $i^{th}$ shift on the coded aperture before the lens converge at $u''(\vartheta_i) = (i - \frac{m_u}{2}) \frac{\Delta s f}{d}$. This defines the relationship with between $\Delta_s$, $f$, $d$ and the pixel pitch of the imaging sensor, $\Delta_m = \frac{\Delta s f}{d}$. We let the sampling function of the $i^{th}$ detector be $M_i(u, \lambda) = Q_i(\lambda) \text{rect}\left(\frac{u}{\Delta m} - (i - \frac{m_u}{2})\right)$. In this notation, we highlight that the sensor is spectrally-filtered, i.e., we assign a different spectral response $Q_i(\lambda)$ depending on the pixel index $i$ so that each detector only extracts information in a specific band (see Figure 5.1a). For the sake of simplicity, we assume that the spectral responses $Q_i(\lambda) \approx \delta(\lambda - \lambda_i)$ are close to approximating a Dirac delta at each filter’s centre wavelength $\lambda_i$. Thus, the $i^{th}$ measurement, for $i \in [m_u]$, is obtained as

$$y_i = \int \int M_i(u, \lambda)Y(u, \lambda)dud\lambda,$$

forming the discrete measurements vector. If there are $m_u$ sensor pixels, there are $m_u$ recorded shifts of the target on the coded aperture. Assume that one beam of parallel light rays coming from the target covers $n_u$ coded aperture elements. Then the coded aperture must have at least $s_u = n_u + m_u - 1$ elements to cover all recorded angles.

**Complementary patterns** As explained in [Björklund and Magli, 2013], it is possible to manipulate the coded aperture pattern and the measurements vector so that the equivalent symbols of the transmittance function $S(u)$ are $S_i \in \{-1, 1\}$ instead of $S_i \in \{0, 1\}$. This implies the use of a programmable coded aperture (or a fixed, e.g., printed, mask that can be removed easily for an open acquisition). Two solutions, that rely on the linearity of the measurements, are proposed: Either, by using two complementary patterns $S_+(u)$ and $S_-(u)$, where transparent pixels ($S_i = 1$) become opaque ($S_i = 0$) and vice versa, and subtracting the
corresponding measurements vectors,

\[ y_i = y_{+,i} - y_{-,i} = \int \int M_i(u, \lambda)[(S_+ - S_-) * X_0](\frac{d}{d}u, \lambda)dud\lambda, \]

or by subtracting measurement made with a fully transparent aperture, \( S_{on}(u) \), (i.e., for which \( S_i = 1, \forall i \)) from two times the positive pattern measurements, i.e., \( y_i = 2y_{+,i} - y_{on,i} \) with \((2S_+ - S_{on})\). We can envision that this subtraction could be realized on chip at the sensor level, thus eliminating the need to store and manipulate two full frames per snapshot. In the rest, we consider that either of those approaches is adopted and that the equivalent \( S_i \in \{-1, 1\} \) pattern is obtained in a single snapshot.

**Discrete model** Coming back to two discrete spatial dimensions and one discrete spectral dimension, the discrete forward model of the optical processing chain is

\[
Y = \sum_{\ell=1}^{n_\lambda} M_\ell(S \bar{\ast} X_{0,\ell}),
\]

where \( Y \in \mathbb{R}^{m_u \times m_v} \) is the array of recorded measurements; \( M_\ell(\cdot) : \mathbb{R}^{m_u \times m_v} \rightarrow \mathbb{R}^{m_u \times m_v} \) are the mask operators, which correspond to the discrete, 2-D version of the sampling function \( M_i(u, \lambda) \); the mask \( S \in \{-1, 1\}^{s_u \times s_v} \) represents the discrete, 2-D version of the aperture array \( S(u) \); \( X_{0,\ell} \in \mathbb{R}^{n_u \times n_v} \) is the band of index \( \ell \) of the full target cube, \( X_0 \in \mathbb{R}^{n_u \times n_v \times n_\lambda} \); and \( \bar{\ast} \) denotes a discrete 2-D convolution that has been truncated to its valid part\(^4\). The size, \( s_u \times s_v \), of the coded aperture is chosen so that the valid convolution matches the size of the sensor, i.e., \((s_u - n_u + 1, s_v - n_v + 1) = (m_u, m_v)\).

\(^4\)Here, the full convolution is of size \( n_u + s_u - 1 \times n_v + s_v - 1 \) while the so-called valid convolution is the center part, of size \( s_u - n_u + 1 \times s_v - n_v + 1 \). The valid convolution is also the valid part of a circular convolution of size \( s_u \times s_v \).
5.4 Multispectral Compressive Imaging by Out-of-Focus Random Convolution

**Multi-snapshots**  As was the case for the MSVI, since a total of $m_u \times m_v$ measurements is recorded by the sensor, the latter would produce $\frac{m_u m_v}{n_\lambda}$ measurements per band in a single snapshot acquisition. Rather than doing so, we here consider the possibility of partitioning the acquisition of $y \in \mathbb{R}^m$ by taking multiple snapshots, $\{Y_p\}_{p \in [m_S]}$, with $m_S$ different aperture patterns, i.e., $\{S_p\}_{p \in [m_S]}$. Therefore, the total number of measurements becomes $m = m_u m_v m_S$ and for each band, only $\frac{m}{n_\lambda m_S}$ measurements are required at each snapshot. In addition, taking multiple snapshots with different aperture patterns is expected to reduce the correlation between measurements.

5.4.2 Non-idealities and practical considerations

The parallel compressive MSRC scheme entails some additional concerns that must be taken into account when considering an actual implementation. First and foremost, the geometrical optics assumption is unfortunately too simplistic and one must also treat the wave character of light. In this section, we explain the effect of diffraction along with a few other non-idealities.

**Diffraction and Point Spread Function**

As well anticipated by [Björklund and Magli, 2013], the main optical-level limitation of this scheme is the impact of diffraction that inevitably occurs at the coded aperture. When a single square aperture of small width, followed by a lens, as the one placed in front of the FPA in Figure 5.7, is illuminated by a plane wave, Fourier optics [Goodman, 2005, Chapter 4] predicts the formation of a diffraction pattern at the focal plane of the lens. The image formation model of the system must therefore account for the presence of an additional 2-D convolution kernel that describes the diffraction pattern. In other words, using the superposition principle, the effect of diffraction at the coded aperture is, here, modelled as an optical filter (see [Goodman, 2005]), whose impulse
response, or **Point Spread Function (PSF)**, is the diffraction pattern due to a single aperture.

Specifically, using Fourier optics, one arrives at the following expression for the 2-D, wavelength-dependent, diffraction kernel at the focal plane,

\[ H(u, v, \lambda) = a \text{sinc}^2 \left( \frac{u \Delta_s}{\lambda f} \right) \text{sinc}^2 \left( \frac{v \Delta_s}{\lambda f} \right), \]

where \( a > 0 \) is an energy conservation constant depending on \( \lambda, \Delta_s \) and \( f \). The constant is unimportant as we consider the normalized diffraction kernel in the following. This PSF has a low-pass effect that substantially limits the bandwidth of the system. Thus, the effective random convolution operator modulates the scene at a rate smaller than the Nyquist rate of \( x_0 \). This will cause a decrease in the performances of random convolution, as the measurements will be more correlated than the ideal model.

Again, the mathematical expression is separable in the spatial dimension. Thus, for the sake of coherence with the 1-D continuous description, we simplify the diffraction kernel to one spatial dimension as

\[ H(u, \lambda) = a \text{sinc}^2 \left( \frac{u \Delta_s}{\lambda f} \right). \]

Before being sampled by the sensor, the function \( Y(u, \lambda) \) is thus convolved spatially with \( H(u, \lambda) \) as

\[ \tilde{Y}(u, \lambda) = [H * Y](u, \lambda), \]

or, equivalently, defining \( \tilde{H}(u, \lambda) \triangleq \frac{f}{d} H(\frac{f}{d} u, \lambda), \)

\[ \tilde{Y}(u, \lambda) = [\tilde{H} * S * X_0](\frac{d}{f} u, \lambda). \]

From the last expression, one can see that this is equivalent to replacing \( S(u) \) by a **diffracted aperture pattern**, \( \tilde{S}(u, \lambda) \triangleq [\tilde{H} * S](u, \lambda). \)
5.4 Multispectral Compressive Imaging by Out-of-Focus Random Convolution

Discretizing, as before, the exact diffracted pattern would mean sampling the exact expression of \( \tilde{S}(u, \lambda) \) with a rect function. As a simplification, we rather sample \( \tilde{H}(u, \lambda) \) beforehand and we instead perform the convolution in the discrete domain, i.e., we define the spatially discretized (1-D) diffraction kernel, for \( i \in [m_h] \), where \( m_h \) is the arbitrary length of the sampling grid,

\[
h_i(\lambda) \buildrel \Delta \over = b \int \tilde{H}(u, \lambda) \text{rect} \left( \frac{u}{\Delta_s} - (i - \frac{m_h}{2}) \right) \, du
\]

where \( b \) is a normalization factor such that \( \sum_i h_i(\lambda) = 1 \). Note that this is equivalent to sampling \( H(u, \lambda) \) with steps \( \Delta_m \) instead of \( \Delta_s \). We thus have the diffracted symbols of \( \tilde{S}(u', \lambda) \) approximately given by

\[
\tilde{S}_i = \sum_{j=1}^{m_h} S_{i-j} h_j(\lambda).
\]

The 2-D discrete model is then adapted by replacing the aperture pattern \( S \) by a diffracted, wavelength dependent aperture pattern \( \tilde{S}_\ell \). Specifically, let \( H_\ell \in \mathbb{R}^{m_h \times m_h} \) be the 2-D discrete PSF, by straightforwardly extending to 2-D the definition of \( h_i(\lambda_\ell) \), then

\[
\tilde{S}_\ell \buildrel \Delta \over = H_\ell * S.
\]

When several snapshots are taken, we note \( \tilde{S}_{p,\ell} \buildrel \Delta \over = H_\ell * S_p \). Since the size \( m_u \times m_v \) of the focal plane is chosen to match the valid convolution with a coded aperture of size \( s_u \times s_v \), we can safely truncate the diffracted aperture pattern to an effective size of \( s_u \times s_v \) (instead of keeping the full \( s_u + m_h - 1 \times s_v + m_h - 1 \) pattern).

**Sizing example.** The goal of this sizing example is to demonstrate that the diffraction kernel is an important and challenging issue. If we use a straightforward implementation of the scheme depicted on Figure 5.7, without additional optical elements, we can compute the size of the diffraction kernel as a function of the pixel pitches \( \Delta_m \) and \( \Delta_s \) and
the focal length, \( f \), which is constrained by the size of the lens, itself constrained by the size of the coded aperture. Specifically, the diameter \( D_{\text{lens}} \) of the focusing lens must be bigger than the coded aperture, \( i.e., \)

\[
D_{\text{lens}} \geq \sqrt{2} \max \{s_u, s_v\} \Delta s.
\]

Moreover, commercially available lenses cannot have a focal length that is smaller than the lens radius, \( i.e., \)

\[
f \geq D_{\text{lens}}/2.
\]

The width of the diffraction kernel on the sensor can be characterized by the location of its first zeros on either side of the origin, \( i.e., \) when the argument of the \( \text{sinc}^2(\cdot) \) is equal to one. It might be convenient to characterize the width in terms of number of pixels, \( i.e., \)

\[
D_{\text{PSF},\lambda} = 2 \frac{\lambda f}{\Delta_m \Delta s},
\]

so that \( H(0.5 D_{\text{PSF},\lambda \Delta_m}, \lambda) = \tilde{H}(0.5 D_{\text{PSF},\lambda \Delta_s}, \lambda) = 0 \).

We now give the sizing example which correspond to the simulations parameters (see Section 5.4.4), \( i.e., \) \( n_u = n_v = 256, m_u = m_v = 256 \), so that \( s_u = s_v = 511 \). Notice that the largest PSF width corresponds to the longest wavelength, \( i.e., \) in our case, \( \lambda_{\text{max}} = 620\text{nm} \). Let \( \Delta_s = 80\mu\text{m} \) so that the coded aperture is about 41mm wide and we must choose a lens with a diameter of at least 58mm, with focal length \( f \geq 29\text{mm} \), \( e.g., \) we can arbitrarily choose \( f = 40\text{mm} \). All these parameters being fixed, the PSF width on the focal plane (at \( \lambda_{\text{max}} = 620\text{nm} \)) is 620\(\mu\text{m} \) and the number of pixels is entirely determined by the pixel pitch \( \Delta_m \) of the sensor. Note that \( \Delta_m \) also determines the distance \( d \) between the image plane and the coded aperture. With these parameters, the huge diffraction PSF is not compatible with a straightforward use of the standard CMOS technology of \( \Delta_m = 5.5\mu\text{m} \) used in [Geelen et al., 2014]. In that case, we get a width of \( D_{\text{PSF},\lambda_{\text{max}}} = 112 \) pixels, which is obviously impractical.
5.4 Multispectral Compressive Imaging by Out-of-Focus Random Convolution

Instead, we must devise a way to modify the equivalent $\Delta_m$. A simple, naive way of achieving that is to bin pixels together and form bigger macro-pixels. This solution, proposed in [Björklund and Magli, 2013], is obviously wasteful in terms of number of pixels on the sensor and defeats the purpose of the compressive architecture. Another possibility, that would require further investigation, is to find a way to magnify the sensor as viewed from the focusing lens. For the simulations, intended as a proof of concept, we make the assumption that it is somehow possible to effectively magnify 10 or 20 times the standard 5.5$\mu$m CMOS sensor. This leads to a width of respectively $D_{\text{PSF},\lambda_{\text{max}}} = 11$ pixels ($\Delta_m = 55\mu m$) and $D_{\text{PSF},\lambda_{\text{max}}} = 5$ pixels ($\Delta_m = 110\mu m$). The 11 pixels PSF is illustrated on Figure 5.8.

Other practical considerations

We mention here a few other important issues that may occur with the proposed MSRC design. First, the different spectral profiles associated to the FP filters (see Figure 5.1a) must be precisely known in order to compensate for the different attenuation coefficients; either before reconstruction in a pre-processing step, or during the reconstruction, by
including the profiles in the sensing operator, or in post-processing after reconstruction. In this work (i.e., in the simulations), we assume that the filters all have perfect narrow-band profiles with equal gains.

Beside the spectral profiles differences, manufacturing process variability might also introduce unknown gains in the sensor. Whereas this is, arguably, not so much of a problem in the MSVI design, as the artifacts resulting from a corrupted measurement would still be localized; this can potentially have wilder consequences for the MSRC reconstruction, where each corrupted measurement provides corrupted information about the entire scene. The problem of blind gain calibration, for situations where direct gain calibration is not possible, has been tackled, e.g., in [Cambareri and Jacques, 2016].

Note that narrow band filtering considerably decreases the total light intensity throughput of the system. Therefore, particular care must be taken when choosing the other optical components in order to limit further unnecessary light attenuation. For instance, several possible choices exist for the coded aperture. A manufactured mask with physical holes will provide the best light throughput but makes it harder to program several snapshots. A semi-transparent LCD Spatial Light Modulator (SLM) will have imperfectly opaque and imperfectly transparent elements, entailing light losses and light leakage. The same goes with reflective, polarizing, Liquid Crystal on Silicon (LCoS) devices [Nagahara et al., 2010], which imply the use of a polarizing beam splitter (see [Fowles, 1968]), thus considerably decreasing the light throughput. Despite their excellent light transmittance, Digital Micro-mirror Devices (DMD), such as the one used in the single-pixel camera [Romberg et al., 2008], are probably not a good choice for the MSRC architecture because the deflection angle of the mirrors is not precise enough to be used out of focus.

Another important issue is, precisely, the alignment and disposition of the optical elements with respect to each other. A calibration procedure must be provided in order to precisely set, for instance, the
5.4 MULTISPECTRAL COMPRESSIVE IMAGING BY OUT-OF-FOCUS RANDOM CONVOLUTION

Distances \( d \) and \( f \) and the roll angle between the sensor and the coded aperture.

The lenses must also be chosen carefully so as to minimize chromatic and spherical aberrations. When those aberrations occur nonetheless, they must also be subject to calibration and correction.

Finally, it must be noted that the modeled diffraction kernel is, itself, an approximation based on several more or less realistic assumptions, such as the use of a perfect thin lens, the fact that the object is an incoherent plane wave, and so on. To model precisely the true PSF of the system, one can also devise elaborate calibration procedures. For instance, one can use a random aperture or a Modified Uniformly Redundant Array (MURA) aperture (see, e.g., [Brady, 2009]), along with a point-like target light source (or another simple known object) and formulate the PSF estimation problem as a regularized inverse problem. See, for instance, [González et al., 2014; Guérit et al., 2016] for some work on PSF estimation methods with geometrical scene priors in different imaging contexts. One can also expect, in some cases, that the spatial invariance of the PSF may be an unrealistic assumption, in which case a spatially dependent PSF could also be estimated.

5.4.3 Sensing matrix implementation

Based on the discrete model of Sections 5.4.1 and 5.4.2, we can now write the sensing matrix corresponding to the MSRC forward model. Let \( \tilde{S}_{p, \ell} \in \mathbb{R}^{m_u n_v \times m_u n_v} \) be the partial block circulant matrix which defines the valid convolution operator with the diffracted aperture pattern \( \tilde{S}_{p, \ell} \), and let \( M_\ell \in \mathbb{R}^{m_u n_v \times m_u n_v} \) be, as in the MSVI design (see Section 5.3.1), the matrix equivalent to the mask linear operator \( M_\ell(\cdot) \) defined above. First, notice that every band and every snapshot can be processed separately by the submatrices

\[
\Phi_{p, \ell} = M_\ell \tilde{S}_{p, \ell}.
\]
The full sensing matrix, thus reads

\[
\Phi = \begin{pmatrix}
\Phi_{1,1} & \ldots & \Phi_{1,n}\lambda \\
\vdots & \ddots & \vdots \\
\Phi_{mS,1} & \ldots & \Phi_{mS,n}\lambda 
\end{pmatrix}.
\]

This follows from the natural order in which the elements are stacked in the vectorized \( y \in \mathbb{R}^m \) and \( x_0 \in \mathbb{R}^n \). Note that each \( \Phi_{p,\ell} \) is a partial (and partially masked) circulant matrix (i.e., partial random convolution), with entries in \( \{-1, 0, 1\} \), which enjoys good compressed sensing properties as explained in Section 5.1.2 and Chapter 2.

**Sensing matrix factorization**

We now focus on an efficient factorization \( \Phi = R_m \tilde{F} R_n^* \) so as to comply with the requirements of the reconstruction method (see Section 5.2.3). Let \( R_{m_u m_v} \in \mathbb{R}^{m_u m_v \times s_u s_v} \) be the restriction operator that selects the valid part, of size \( m_u \times m_v \), of a circular convolution of size \( s_u \times s_v \). Similarly, let \( R_{n_u n_v}^* \in \mathbb{R}^{s_u s_v \times n_u n_v} \) be the zero-padding operator (transpose of the restriction) that pads the input image so that it matches the size \( s_u \times s_v \) of the circular convolution. Let \( F \in \mathbb{C}^{s_u s_v \times s_u s_v} \) be the 2-D DFT of size \( s_u \times s_v \) and let \( \Sigma_{p,\ell} = \text{diag}(F \text{vec}(\tilde{S}_{p,\ell})) \), i.e., the diagonal matrix formed with the 2-D DFT of the diffracted aperture pattern. With all these ingredients, we can factorize,

\[
\Phi_{p,\ell} = M_{\ell} R_{m_u m_v} F^* \Sigma_{p,\ell} F R_{n_u n_v}^*.
\]

In the light of the factorization of \( \Phi_{p,\ell} \), the factors of the full matrix \( \Phi \) read \( R_n^* = \text{diag}_{n\lambda}(R_{n_u n_v}^*) \), \( R_m = \text{diag}_{mS}(R_{m_u m_v}) \), with

\[
\tilde{R}_{m_u m_v} = (M_1 R_{m_u m_v} \ldots, M_{n\lambda} R_{m_u m_v}),
\]
and, denoting $F_{n} = \text{diag}_{n} (F)$, $F_{m} = \text{diag}_{m} (F)$ and the diagonal matrices $\tilde{\Sigma}_{p} = \text{diag} (\Sigma_{p,1}, \ldots, \Sigma_{p,n})$, $\bar{\Phi} = F_{m} \left( \begin{array}{c} \tilde{\Sigma}_{1} \\ \vdots \\ \tilde{\Sigma}_{m} \end{array} \right) F_{n}^{*}$.

As needed, the advantage of this factorization is that $\bar{\Phi}^{*} \bar{\Phi} + \mu \text{Id}$ is easily invertible. Indeed, since $F_{m}$ and $F_{n}$ are unitary and $\tilde{\Sigma}_{p}$ is diagonal, noting $\tilde{\Sigma}^{2} = \sum_{p=1}^{m} \tilde{\Sigma}_{p}^{2}$, we have,

$$\bar{\Phi}^{*} \bar{\Phi} + \mu \text{Id} = F_{n}^{*} \tilde{\Sigma}^{2} F_{n} + \mu \text{Id} = F_{n}^{*} \left( \tilde{\Sigma}^{2} + \mu \text{Id} \right) F_{n}.$$ 

Therefore, inverting $\bar{\Phi}^{*} \bar{\Phi} + \mu \text{Id}$ is just equivalent to invert the diagonal matrix, $\tilde{\Sigma}^{2} + \mu \text{Id}$. Note that computing $\Phi x$ for some input vector $x \in \mathbb{R}^{n}$ requires to compute $n \lambda$ DFT (using the FFT algorithm) and $m \lambda$ inverse DFT of size $s_{u} \times s_{v}$. Similarly, computing $\Phi^{*} z$ for some input vector $z \in \mathbb{R}^{m}$ requires $m \lambda$ DFT and $n \lambda$ inverse DFT. Comparatively, computing the inverse of $\bar{\Phi}^{*} \bar{\Phi} + \mu \text{Id}$ is therefore cheaper (as soon as $m > 1$) since it only requires $n \lambda$ DFT and $n \lambda$ inverse DFT.

### 5.4.4 Simulations

In this section, we compare the reconstruction results of the MSRC technique, using the method described in Section 5.2.3, with the sensing matrix from Section 5.4.3. Specifically, we evaluate the performances on the selected samples from the CAVE data-set (see Section 5.2.3) for three parameter settings and two noise levels. As mentioned in the sizing example of Section 5.4.2, the size of the FPA and of the target are fixed to $n = 256$, $m = 256$, so that $s = 511$. To vary the sampling rate $m/n$, the number of snapshots is gradually increased as $m \in \{1, 4, 9, 16\}$ so as to match the sampling rates of the MSVI simulations (see Section 5.3.2). Note that simulations (not sown here)
were also performed by increasing the FPA and coded aperture sizes instead of the number of snapshots (thus fixed to $m_S = 1$), similarly to what was done for the MSVI setup. The performances were substantially the same (though slightly inferior, as predicted in Section 5.4.1). One may therefore choose the method that best suits their needs and practical constraints.

On Figure 5.9, the performances are quantitatively compared, in terms of reconstruction PSNR, between an ideal case where diffraction was neglected and two cases where the diffraction kernel was respectively $D_{PSF,\lambda_{max}} = 5$ pixels and $D_{PSF,\lambda_{max}} = 11$ pixels wide (see Section 5.4.2 and Figure 5.8). As expected, the global trend indicates that increasing the size of the PSF substantially decreases the reconstruction quality. This is true under both 20dB and 40dB noise regimes. One particularly interesting case, however, is when the sampling ratio is at its minimum, i.e., $m/n = 1/16$. At that sampling ratio, the reconstruction PSNR of the diffraction-free case is roughly identical (in fact even...
5.5 Final Comparison

In this short section, we compare the simulation results presented above, i.e., the MSVI method (with random layout) against the MSRC architecture. This is intended to demonstrate that, even though MSRC entails some practical complications and potential issues, part of which have not been solved yet (see Section 5.4.2), it can, in a semi-realistic scenario (5 pixels PSF, 40dB of input SNR) give a substantial performance...
Figure 5.10 Qualitative comparison between PSF sizes in a simulated MSRC sensor. See Figure 5.5 for a detailed explanation of the figure layout and see Figure 5.2 for the false color full groundtruths. Reconstruction PSNRS were
5.5 Final Comparison

boost (about 4dB, on average) over inpainting in extreme subsampling situations \((m/n = 1/16)\).

Figure 5.11 repeats some of the curves from Figures 5.4 and 5.9. The first thing to notice is that, in the ideal case, i.e., if one can, somehow, get rid of the diffraction effects, the MSRC device provides (under reasonably low noise) a performance improvement of 1dB (for the Nyquist rate) to 4dB (for the 1/16 subsampling rate) over the MSVI. This, alone, justifies, a posteriori, the present study on the feasibility of the MSRC design.

However, the mere inclusion of a reasonably small diffraction kernel, justified in Section 5.4.2, implies a drastic performance drop, especially at higher sampling rates. As noted above, the main and most interesting exception, highlighted in green, is the \(m/n = 1/16\) subsampling rate where, despite the PSF, the random convolution architecture beats the inpainting methods by 4dB.

Finally, let us also emphasize that high quality optical and electronics components are required so as to avoid falling in a higher noise regime. One can see on Figure 5.11 that the gap between MSVI and the ideal MSRC falls to zero as soon as 20 dB of input noise (SNR) is added to the measurements.

For the qualitative comparison of Figure 5.12, we focus on the case, highlighted in green on Figure 5.11, where MSRC outperforms, on average, better than MSVI, even when diffraction is added. Notice, first, that the spectral accuracy on the selected pixels (in particular, pixel (B) on balloons and pixels (A) and (B) on chart and stuffed toy) is better with both MSRC than with MSVI. Regarding spatial accuracy, the differences are mostly noticeable on the chart and stuffed toy sample. Note the noisy patterns that appear between the stripes on the toy’s sleeve with MSVI reconstruction. It is, nevertheless, worth noting that the spatial high-frequency content is still quite affected by the diffraction kernel. This is, once again, especially visible on the chart patterns and digits in the first column of the chart and stuffed toy sample.
Figure 5.11 Average reconstruction PSNR of the MSVI and MSRC architectures. The diamond markers (orange) indicate the MSVI random layout architecture. The dot markers (red) denote the MSRC architecture without PSF. The triangle markers (black) indicate the MSRC architecture with $D_{\text{PSF}, \lambda_{\text{max}}}$ = 5 pixels. Plain lines (resp. dotted lines) indicate an input SNR on the measurements of 40dB (resp. 20dB). The same color-marker code as on Figures 5.4 and 5.9 has been used.
Figure 5.12 Qualitative comparison between simulated MSVI (random layout), diffraction-free MSRC and MSRC with a 5 pixels PSF. See Figure 5.5 for the details and see Figure 5.2 for the false color full groundtruths.
5.6 Conclusion

In this chapter we proposed and compared two strategies for compressive multispectral imaging: Multispectral Volume Inpainting (MSVI) and Multispectral Random Convolution (MSRC). Both strategies use a sensor with integrated Fabry-Pérot filters to record measurements in narrow bands of the spectrum of light, jointly with the principles of Compressed Sensing (CS). As opposed to most prior work in compressive multispectral/hyperspectral imaging, the proposed methods involve no dispersive elements in the optical path and thus entail no mixing in the spectral domain.

Along with the conceptual optical design, we proposed, for each acquisition device, an accurate discrete linear forward model. The reconstruction procedure, formulated as a regularized inverse problem, was presented in a unified, abstract way, to demonstrate its versatility. The original prior used in the problem is an analysis redundant frame, based on undecimated wavelets and discrete cosine transforms. The analysis-$\ell_1$ minimization problem was then solved efficiently with ADMM, thanks to an efficient factorization of the sensing matrices.

The first architecture, MSVI, formulates the recovery problem as generalized inpainting. It thus integrates the multispectral snapshot sensor in a computational imaging scheme. However, as opposed to the other method, no optical mixing in the spatial domain is performed and MSVI only relies on the spatial and inter-band redundancy to obtain a high-resolution recovery from subsampled measurements when the scenes follow the proposed sparse model prior.

The second architecture, MSRC, uses an out-of-focus coded aperture to provide spatial-domain mixing. Doing so, it implements random convolutions, which, as opposed to mere subsampling, have beneficial mathematical and computational properties allowing higher compression ratios. However, we showed that if not properly sized, it may be heavily impacted by diffraction effects.
Through extensive numerical simulations, we explored different realistic setups. We devised some practical guidelines and highlighted limitations for both methods allowing to proceed towards an informed implementation. MSVI is much simpler than MSRC in terms of implementation and practical issues such as calibration. In terms of performances, in an ideally calibrated and ideally sized (so as to eliminate diffraction) setup with limited noise, MSRC was shown to give better performances, especially in low sampling rate regimes. In other situations (higher noise, higher sampling rate) MSVI seems to be the preferred solution. Especially taking into account of the cost of adding optical elements.

Some perspectives and open questions are discussed in Chapter 6.
CHAPTER 6

CONCLUSIONS

This chapter summarizes the main contributions of this thesis and then discusses a few opened perspectives.

6.1 Summary

Chapter 2 establishes a general background for this thesis. While no contribution is made in that chapter, it introduces key concepts and definitions that are useful throughout the dissertation. It provides classical results and many references that help the reader at placing the contributions of Chapters 3, 4, and 5 in their context.

Chapter 3 studies constrained $\ell_1$-regularized convex minimization for solving sparse inverse problems of the form, $y = \Phi x_0 + w$. Our main theoretical findings provide support recovery guarantees, in high signal-to-noise ratio conditions, when the data fidelity is a compact convex constraint on the residual. The chapter extends results that are well known for the usual Euclidean fidelity, i.e., the $\ell_2$-norm, to more general convex constraints that can, alternatively, be defined by a finite coercive gauge. This includes, among others, $\ell_\alpha$-losses for $\alpha \geq 1$, mixed norms and the nuclear norm. Those results not only characterize exact support stability (sparsistency) but also give an explicit way of predicting the extended support and the sign pattern of any perturbed solution, including in case of support instability. We prove that it is possible to do so by
computing a minimal polar gauge certificate from the knowledge of $x_0$ and $\Phi$. Since they explicitly depend on every instance of $x_0$ and $\Phi$, these support recovery results are of the non-uniform type. They are thus different, in many aspects, from those obtained using tools such as the Restricted Isometry Property; also because they consider support stability, as opposed to stability in $\ell_2$ sense.

The main result, Theorem 5, unifies two very different situations. First, non-smooth losses, including polyhedral losses such as $\ell_1$ or $\ell_\infty$ constraints, entail a specific proof strategy, leading to explicit tight bounds and a closed form solution. They were, thus, treated separately from smooth losses, such as the $\ell_2$-norm, and partially smooth losses, such as mixed norms. Preliminary results, focused on $\ell_1$ and $\ell_\infty$ losses, were presented in [Degraux et al., 2016]. The rest will be submitted for a future journal publication.

Chapter 4 presents a novel unsupervised online convolutional dictionary learning algorithm for multimodal imaging applications. The contributions are multiple. We propose an original two-stage cost function for convolutional sparse coding, which couples an inverse problem regularized by a TV analysis prior, with the problem of finding the best joint-sparse synthesis representation in a convolutional dictionary. In addition to computing the convolutional joint-sparse representation, the novel formulation thus aims at solving the inverse problem associated to indirectly observed data, making it an unsupervised method.

Based on that cost function, we extend an online dictionary learning method, originally published in [Mairal et al., 2010], whose patch-based approach uses directly observed data. The method we propose here uses convolutional atoms which provide several benefits over patches, such as translation invariance and native capability for global optimization. The convolutional approach requires a careful implementation to preserve the computational efficiency. Additionally, we apply joint sparsity of learned convolutional atoms as a way to transfer information be-
6.1 Summary

Between different modalities. The online capability of the algorithm makes it suitable for handling very large datasets or dynamically streamed multimodal data.

In an extensive numerical experiment, we apply the proposed approach to the problem of intensity–depth fusion for depth inpainting. We demonstrate that the proposed method is able to outperform guided filtering [He et al., 2013], and weighted TV [Castorena et al., 2016] on most samples of the considered dataset. The proposed method is able to capture intensity–depth feature correspondences by learning appropriate joint atoms. We also show, in a second experiment, that it has the interesting capability of continuously adapting and improving a specialized convolutional dictionary, which is interpretable and can potentially be re-used for other regularization tasks.

The work presented in this chapter has been accepted at the *IEEE International Conference on Image Processing 2017* and a preprint is available [Degraux et al., 2017]. An extended version of that paper will be submitted for journal publication.

Chapter 5 proposes and studies two acquisition device architectures for MS imaging, based on CS. Both proposed methods use a dedicated snapshot multispectral sensor which integrates Fabry-Pérot spectral filters to record measurements in several narrow bands of the spectrum of light. They involve no dispersive elements in the optical path and maintain a low system-level complexity.

The first proposed imaging technique is related to inpainting and super-resolution. It aims at completing the multispectral volume which was subsampled by spectral filtering at pixel level. The second proposed imaging technique adds a layer of complexity but enables higher compression ratios. By introducing spatial light modulation with an out-of-focus coded aperture before the sensor, it implements random convolutions, which have several benefits in terms of mathematical (RIP guarantees) and computational properties (fast computation with the
This more complex image formation model requires a more accurate design evaluation, but is potentially capable of achieving higher reconstruction quality at lower sampling rate.

In addition to this analysis, we discuss the sparse signal model and the recovery algorithm which provide an accurate reconstruction of the MS cubes with tractable computation time. It relies on the redundancy of multispectral images to obtain a high-resolution, high quality recovery from the recorded measurements. We use the same approach in both cases to solve the associated inverse problem. Specifically, we use an analysis-sparse signal prior in a convex formulation. The convex problem is then solved with an efficient and popular algorithm, known as ADMM.

Through extensive numerical simulations, in which both architectures are compared in terms of achievable recovery performances, we explore various tradeoffs in different realistic setups. We highlight some practical guidelines and limitations for both methods.

Part of the material presented in this chapter has been published in [Degraux et al., 2014, 2015]. The rest will be submitted for a future journal publication.

6.2 Perspectives and open questions

Chapter 3 establishes support stability results under the assumption that the noise, as measured with a bounded coercive gauge, $F(w)$, is small. This deterministic bound on $F(w)$ constitutes the only requirement on the noise model. A first interesting perspective would be to analyze the constants involved in the main theorem in order to establish links between the results and the noise statistics, and also, possibly adding some constraints on $\Phi$, in order to obtain probabilistic non-uniform guarantees. For instance, in the context of compressed sensing, using
the randomness of $\Phi$ may allow to derive sharp sample complexity bounds as a function of the sparsity of $x_0$.

Arbitrarily large noise has been left out of the present study. As noted in the chapter, no other results in the literature provide any insight about support recovery in the large noise regime. In that case, we are nevertheless able to provide bounds on the distance between $x_0$ and the recovered vector. Some effort, as yet unpublished, has been made in this sense, assuming even more general regularizers than the $\ell_1$-norm. This may be the subject of a forthcoming paper.

The concept of sparse support recovery can also be extended to the more general concept of model selection or model consistency, as studied by [Vaiter, 2014; Vaiter et al., 2015, 2014]. The tools developed in that line of work were a large source of inspiration for the results presented here. It is therefore natural to envision a fusion of the class of compact convex constraints, studied here, with model consistency of partly smooth regularizers, replacing the $\ell_1$-norm and support stability. Extending this idea, it might be interesting to generalize even further the class of convex fidelity constraints by considering more general, possibly partly smooth, penalty functions.

Chapter 4 opens a few perspectives. First of all, the convergence of the proposed online convolutional dictionary learning algorithm would be interesting to analyze. Note that empirical convergence observations were omitted for brevity, as the numerical results focused mainly on recovery quality. Theoretical analysis of this novel convolutional online method falls in a very active field of research. As mentioned in Chapter 2, bi-convex alternating minimization with quadratic coupling is a well studied class of problems with a lot of applications (see Section 2.2.5). The stochastic point of view could also allow to develop a proof of convergence and maybe sample complexity bounds, e.g., by adapting the analysis of [Gribonval et al., 2015; Mairal, 2013; Mensch et al., 2017] to the proposed approach. A particularly interesting aspect to study would
be the interaction between the sensing matrices, e.g., the associated number of measurements, and the number of training samples that is necessary to obtain convergence guarantees.

Regarding the numerical performances of the convolutional sparse coding, [Wohlberg, 2016] shows empirical evidence that ADMM performs faster, in some situations, than FISTA. Replacing FISTA by ADMM in Algorithm 4 should not be difficult and might provide a substantial speed improvement.

Another potential direction would be to apply the framework to other multimodal inverse problems with more complex sensing matrices and more modalities. For instance, one can apply the framework to hyperspectral pan-sharpening [Liu and Boufounos, 2012; Wei et al., 2015], or other image fusion tasks [Liu et al., 2016].

Chapter 5 studied and compared two approaches for MS imaging based on spectrally filtered snapshot sensors. As a first general perspective, it would be interesting to consider a more general class of spectrally filtered sensors. For instance, one may wonder what happens when we increase the number of different filters in the filter bank, therefore reducing the number of pixels per band but increasing the number of bands. This study could lead to interesting design tradeoffs and new numerical challenges.

This also opens the question of exploring different spectral filter designs. For example, in the present study, we have briefly mentioned the fact that FP filters’ spectral responses sometimes have more than one narrow peak. This is due to the physics of the interferometers and, in the context of multispectral snapshot imaging, this is usually viewed as a non ideality that one has to avoid at all costs. Instead, one might be able to exploit this multi-mode FP resonance and turn it into an advantage. Rather than providing a monochromatic (narrow-band) measurement, each pixel would provide a superposition of several narrow bands. These profiles would have to be engineered in a controlled way and
precisely calibrated in advance. They could be integrated in the sensing model as a mixing matrix with known entries. If done properly, this could add interesting spectral mixing properties, potentially matching, in some way, the framework of CS theory and random matrices. A by-product advantage of multiplying the number of bands in each filter's response would be that it necessarily increases the total light throughput, therefore increasing the associated signal-to-noise ratio. Let us note that the proposed architectures are not constrained to specifically use the FP filters technology. Any kind of pixel-level filtering can achieve essentially the same capabilities. Depending on the targeted application, it might make sense to use a smaller number of bigger (or more spread) pixels. For instance, using an array of narrow-band, highly sensitive photo-diodes would not only increase the input SNR, but also increase the pixel pitch $\Delta_m$ and therefore decrease the relative size of the PSF in the MSRC architecture.

On that note, an important question remains open on how to mitigate the size of the diffraction PSF, e.g., by making the equivalent $\Delta_m$ bigger. Using a sensor with a low fill-factor is a straightforward workaround. Another idea that needs to be investigated is to use a magnifying lens after the focusing lens, possibly coupled with a light diffuser. If this is a viable solution, this would allow using off-the-shelf 5.5\(\mu\)m snapshot sensors, e.g., from IMEC to actually build an effective prototype.

Building a working prototype of MSRC is a significant challenge. An early attempt, documented in Valerio Cambareri’s thesis [Cambareri, 2015, Chapter 9], was realized in 2014. It was using an LCoS spatial light modulator with a polarizing beam splitter, and IMEC’s mosaic sensor. We discovered afterwards that the poor results were partly due to the huge diffraction kernel whose effect was impossible to cancel without rethinking the design. This experience, however, gave a lot of insight for the present study. Once the question of the PSF is resolved, a natural continuation of this work would be a new attempt at practical implementation. Note that [Cambareri, 2015, Chapter 9] also presents
a PSF estimation method, which could be used as a base in such a new experimental setup to obtain a more accurate PSF than our theoretical diffraction model.

Another interesting issue to address in a hypothetical practical setup would be automatic gain calibration. As mentioned in Section 5.4.2, this problem was studied in [Cambareri and Jacques, 2016].

On a more theoretical note, it would be interesting to derive sample complexity bounds for both the generalized inpainting method, and the random convolution-based method. As mentioned in Chapter 2 and Chapter 5, both approaches lead to sensing matrices that are close to typical and well-studied CS random matrices. In particular, the random layout MSVI is akin to random basis ensembles, while the MSRC is intrinsically connected to random partial block-circulant matrices. However, because of the spectral dimension, and in the case of generalized inpainting, because of the upsampling operator, they are not exactly equivalent to those canonical cases. A precise theoretical study of the fine structure of those random matrices, including the spectral dimension, would be an interesting challenge. Note that the interactions with the proposed analysis sparsity transform also have to be taken into account, especially for the MSVI.

Finally, a more general question related to random convolutions is whether it is possible to modify the probability distribution of the random mask so as to account for local coherence with the sparsity transform. The intent of this idea, that one could call variable density random convolutions, would be to extend the line of work about variable density sampling [Adcock et al., 2015; Jones et al., 2016; Krahmer and Ward, 2014; Puy et al., 2011] to the realm of random convolutions.
Publications


REFERENCES


References


References


(including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 8753(1):31–42.


APPENDIX A

ELEMENTS OF CONVEX OPTIMIZATION

In this appendix, we provide a brief theoretical background on selected topics in convex optimization. A lot of these standard definitions were copied, sometimes verbatim, from the thesis of Samuel Vaiter [2014] (and references therein). For more details, the reader can also refer to books on convex optimization and convex analysis like [Rockafellar, 1970], [Boyd and Vandenberghe, 2004], [Hiriart-Urruty and Lemaréchal, 2001].

Generalities

Convex problems have the desirable property of having a unique closed convex set of global minimizers. Once a stationary point has been found, convexity guarantees that it is a global minimum. Convex problems are a wide and very useful class of optimization problems. A minimization problem is convex if and only if its objective function and, by extension, its feasible set are convex.

Definition 9 (Convex set). A set $C \subseteq \mathbb{R}^n$ is said to be convex if any straight line linking two points in $C$ is entirely in $C$, i.e., for any $x, x' \in C$ and for all $\theta \in [0, 1]$,

$$\theta x + (1 - \theta)x' \in C.$$

Note that the relative interior $\text{ri}$ of a nonempty convex sets $C \subseteq \mathbb{R}^n$ is defined as the set of points in $C$ that are not on the boundary relative
ri(C) ≜ \{ x \in C \mid \forall x' \in C, \exists \theta > 1, \theta x + (1 - \theta)x' \in C \}.

**Definition 10** (Convex function). A function \( f : \mathbb{R}^n \to \bar{\mathbb{R}} \) is said to be convex if any straight line linking two points of it graph is entirely above the graph, i.e., for any \( x, x' \in C \) and for all \( \theta \in [0, 1] \),

\[
f(\theta x + (1 - \theta)x') \leq \theta f(x) + (1 - \theta)f(x').
\]

If taking \( x' \neq x \) always results in the above inequality to be strict, then the function is strictly convex. If moreover, there is a quadratic margin for some \( d > 0 \) such that,

\[
f(\theta x + (1 - \theta)x') \leq \theta f(x) + (1 - \theta)f(x') - \frac{d}{2} \theta(1 - \theta)\|x' - x\|_2^2,
\]

the function is said to be strongly convex.

The functions and constraint sets that we consider in convex optimization are always convex. Moreover, for the optimization methods to work, convex objective functions, with values in the extended reals \( \bar{\mathbb{R}} \), need to be proper, i.e., bounded from below and with a finite value for at least one \( x \), and lower semi-continuous.

**Definition 11** (Lower semicontinuity). A function \( f : \mathbb{R}^n \to \bar{\mathbb{R}} \) is lower semi-continuous (l.s.c.) if for all \( x \in \text{dom}(f) \),

\[
\liminf_{v \to x} f(v) \geq f(x).
\]

This means intuitively that if a \( f \) is discontinuous at \( x \) and forms a step (or a jump) at that point, then point \( (x, f(x)) \) is always included at the lower part of the step. It also means that all the sub-level sets and the epigraph of a convex l.s.c. function \( f \) are a closed convex sets. In the literature, the set of proper convex l.s.c. functions over \( \mathbb{R}^n \) is sometimes denoted \( \Gamma_0(\mathbb{R}^n) \). The fact that the objective function of a convex problem is in \( \Gamma_0(\mathbb{R}^n) \) ensures, amongst other things, that the solution exists and that if it is not unique, at least the convex solution set is closed. Note
Finally that the indicator function $\iota_C$ (2.10) associated to a convex set $C$ is proper convex l.s.c.

**First-order optimality**

In order to establish the conditions of optimality associated to a convex problem, let us first define the *subdifferential*, which generalizes the gradient of a function at points where it is not differentiable.

**Definition 12 (Subdifferential).** Let $\partial$ be the subdifferential operator. The subdifferential $\partial f(x)$ of a convex function $f$ at $x$ is the set

$$\partial f(x) \triangleq \{ u \in \mathbb{R}^n \mid f(x') \geq f(x) + \langle u, x' - x \rangle, \text{ for all } x' \in \text{dom}(f) \}.$$  

It is the set of gradients $u$ of all possible affine subspaces (hyperplanes) of dimension $n$ in $\mathbb{R}^{n+1}$ that intersect the graph of $f$ at point $(x, f(x))$ and are, for all $x' \in \text{dom}(f)$, entirely below $f(x')$. This is why an element of $\partial f(x)$ is called a *subgradient* and when $f$ is differentiable at $x$, the gradient $\nabla f(x)$ is the only subgradient of $f$ at $x$. A simple 1-D example is depicted on Figure A.1.

The following condition is called the first-order optimality condition or simply *optimality condition*. Despite its apparent simplicity, it is a necessary and sufficient condition for a vector to be solution of a convex optimization problem. It generalizes, in a sense, the Karush-Kuhn-Tucker

![Figure A.1](image-url)
(KKT) conditions to non-differentiable (but convex) objective functions. This is precisely what makes convex optimization so powerful.

**Proposition 2** (Optimality condition). A vector $\hat{x}$ is the global minimizer of a convex function $f$ if, and only if, $0 \in \partial f(\hat{x})$.

### Proximal operator

All of the algorithms presented below use the proximal mapping or proximal operator defined as follows.

**Definition 13** (Proximal operator). The proximal operator associated to a convex, lower semi-continuous function $f : \mathbb{R}^n \to \mathbb{R}$ is the mapping $\operatorname{prox}_f : \mathbb{R}^n \to \mathbb{R}^n$ defined by

$$
\operatorname{prox}_f(v) = \arg\min_x f(x) + \frac{1}{2} \|v - x\|^2_2. \quad (A.1)
$$

It is sometimes interpreted as a generalization of the projection on a convex set. Indeed, if $f = \iota_C$, the indicator function of a convex set $C$, then the associated proximal operator is precisely the projector, i.e.,

$$
\operatorname{prox}_{\iota_C}(v) = \pi_C(v) \triangleq \arg\min_{x \in C} \frac{1}{2} \|v - x\|^2_2 \quad (A.2)
$$

It is also interestingly associated to the subdifferential operator $\partial f$ by the relationship

$$
\operatorname{prox}_f(v) = (\operatorname{Id} + \partial f)^{-1}(v) \quad (A.3)
$$

The right-hand side of the equation is called the *resolvent* of the subdifferential and is single valued. For more details on this, see [Parikh and Boyd, 2013] and references therein. Let us finally mention that a lot of functions admit a closed form expression for their proximal operator [Combettes and Pesquet, 2011; Parikh and Boyd, 2013]. One famous example is the $\ell_1$-norm which is associated to the *soft thresholding* or *shrinkage* operator

$$
\operatorname{prox}_{\lambda\|\cdot\|_1}(v) = S_\lambda(v) \triangleq \text{sign}(v) \odot (|v| - \lambda)_+, \quad (A.4)
$$
where \( \odot \) denotes the Hadamard (component-wise) product and \( (\cdot)_+ \) leaves the positive entries intact and sets the negative entries to 0. The scalar soft thresholding is depicted on Figure A.2. For the other functions who do not admit a closed form proximal operator we can solve the problem (A.1) iteratively, for example by using one of the algorithms presented below. An example of how to do it efficiently for the TV-norm can be found in [Beck and Teboulle, 2009b].

**Gauge, polar set, polar gauge**

We give here and under the next subtitle some basic tools which will help the reader understand the contributions presented in Chapter 3 and the associated non-uniform results presented at the end of Section 2.2.3. Let us note, however, that those are not crucial to understand the other chapters.

**Definition 14 (Gauge).** Let \( C \) be a nonempty closed convex set in \( \mathbb{R}^n \) such that \( 0 \in C \). The gauge associated to \( C \) is the non-negative, l.s.c., and sublinear function defined as,

\[
\gamma_C(u) \triangleq \inf \{ \gamma \geq 0 \mid u \in \gamma C \}.
\]

Conversely, \( \gamma_C \) is the gauge associated to \( C \) if, and only if \( \gamma_C \) is positively homogeneous and

\[
C = \{ u \in \mathbb{R}^n \mid \gamma_C(u) \leq 1 \}.
\]
Figure A.3 The gauge associated to $C$ at $u$ is the smallest $\gamma$ such that $u \in \gamma C$.

A gauge is bounded, i.e., $\gamma_C(u) < +\infty$ for all $u \in \mathbb{R}^n$, and continuous if, and only if $0 \in \text{int}(C)$. Note that this implies that $0 \notin \text{bd}(C)$ but also that $\text{int}(C)$ is not empty, i.e., the set does not lie on a smaller dimensional subspace (i.e., is not “flat”). Moreover, a gauge is coercive, i.e.,

$$\lim_{\|u\| \to +\infty} \gamma_C(u) = +\infty,$$

if and only if the set $C$ is compact. For example, if $C$ is an $\ell_\alpha$-ball $B_\alpha \triangleq \{x \in \mathbb{R}^n \mid \|x\|_\alpha \leq 1\}$ with $\alpha \geq 1$, then the associated gauge is the $\ell_\alpha$-norm, i.e., $\gamma_{B_\alpha}(u) = \|u\|_\alpha$. A simple example of gauge is shown on Figure A.3.

**Definition 15 (Polar set).** The polar of a nonempty convex set $C$ is defined by

$$C^\circ \triangleq \{u \mid \langle u, x \rangle \leq 1 \text{ for all } x \in C\}.$$

$C^\circ$ is a closed convex set containing the origin. When $C$ is also closed and contains the origin, then in coincides with its bipolar, i.e., $C^{\circ\circ} = C$. If $C = B_\alpha$ with $\alpha \geq 1$, then the polar set is the dual $\ell_\beta$-ball $B_\beta$, i.e., with $(\alpha, \beta)$ satisfying $\frac{1}{\alpha} + \frac{1}{\beta} = 1$.

**Definition 16 (Polar gauge).** The gauge associated to the polar of a set is defined as

$$\gamma_C^\circ(u) \triangleq \gamma_{C^\circ}(u) = \{\gamma \geq 0 \mid u \in \gamma C^\circ\}.$$

and called the polar gauge.
Finally, the subdifferential of a gauge \( \gamma_C \) at a point \( x \) can always be split in an inequality and an equality implicating the polar gauge \( \gamma_{C^o} \). As Vaiter [2014] (Proposition 2.8, coming from Proposition 3.1.4 in [Hiriart-Urruty and Lemaréchal, 2001] and Proposition 3.6) puts it, it is completely characterized by the face of its polar set \( C^o \) exposed by \( x \), i.e., the intersection of \( C^o \) and the supporting hyperplane \( \{ u \in \mathbb{R}^n \mid \langle u, x \rangle = \gamma_C(x) \} \).

**Proposition 3** (Subdifferential of a gauge). Let \( C \subseteq \mathbb{R}^n \) be a convex set containing 0. Then

\[
\partial \gamma_C(x) = \{ u \in \mathbb{R}^n \mid u \in C^o \text{ and } \langle u, x \rangle = \gamma_C(x) \} = \{ u \in \mathbb{R}^n \mid \gamma_{C^o}(u) \leq 1 \text{ and } \langle u, x \rangle = \gamma_C(x) \}.
\]

Moreover, let \( 0 \in \text{int}(C) \) so that \( \gamma_C \) is bounded and let \( \bar{S}_x = \text{aff}(\partial \gamma_C(x)) \) be the smallest affine subspace in which \( \partial \gamma_C(x) \) lies. Then

\[
\partial \gamma_C(x) = C^o \cap \bar{S}_x.
\]

**Duality and certificates**

**Definition 17** (Convex conjugate). The Legendre-Fenchel conjugate \( f^* \in \Gamma_0(\mathbb{R}^n) \) of a proper convex l.s.c. function \( f \in \Gamma_0(\mathbb{R}^n) \) is

\[
f^*(u) \triangleq \sup_{x \in \text{dom}(f)} \langle u, x \rangle - f(x).
\]

Moreover, the bi-conjugate of \( f \) is itself, i.e., \( f^{**} = f \).

Note that the convex conjugate of the indicator function of a convex set \( C \), i.e., \( \iota_C^*(u) = \sup_{x \in C} \langle u, x \rangle \), is called the **support function** of \( C \). If \( C = B_\alpha \) with \( \alpha \geq 1 \), then the associated support function is the dual \( \ell_\beta \)-norm, i.e.,

\[
\iota_{B_\alpha}^*(u) = \| u \|_\beta \quad \text{and} \quad \| \cdot \|_\alpha^*(u) = \iota_{B_\beta}(u).
\]

More generally, the support function of a closed convex set \( C \) is the polar gauge, i.e., \( \iota_C^* = \gamma_{C^o} = \gamma_{C^0}^o \). The converse is true if \( C \) contains the origin.
The convex conjugate is useful to define the *saddle point problem* associated to \((\mathcal{P}_F^R)\), assuming that \(F \in \Gamma_0(\mathbb{R}^m)\) and \(R \in \Gamma_0(\mathbb{R}^n)\). Since \(F = F^{**}\), we have

\[
F(y - \Phi x) = F^{**}(y - \Phi x) = \sup_{p \in \mathbb{R}^m} \langle y - \Phi x, p \rangle - F^*(p).
\]

Because \(F^*\) is in \(\Gamma_0(\mathbb{R}^m)\), the supremum actually exists and is reached so we can replace \(\sup_{p \in \mathbb{R}^m}\) by \(\max_{p \in \mathbb{R}^m}\) and write the saddle point problem

\[
\min_{x \in \mathbb{R}^n} \max_{p \in \mathbb{R}^m} R(x) + \langle y - \Phi x, p \rangle - F^*(p).
\]

The newly introduced variable \(p \in \mathbb{R}^m\) is called the dual variable and the original variable \(x \in \mathbb{R}^n\) is called the primal variable. This saddle point problem also admits a necessary and sufficient first order optimality condition on the primal-dual pair \((x, p)\) which reads

\[
\begin{pmatrix}
0 \\
y
\end{pmatrix} \in \begin{pmatrix}
\partial R & -\Phi^* \\
\Phi & \partial F^*
\end{pmatrix} \begin{pmatrix}
x \\
p
\end{pmatrix},
\]

i.e., \(\Phi^* p \in \partial R(x)\) and \(y - \Phi x \in \partial F^*(p)\). Note that one has to be careful with this matrix expression since \(\partial R\) and \(\partial F^*\) are set-valued non-linear mappings.

Coming back to \((\mathcal{S}_F^R)\), we can swap the \(\min_{x \in \mathbb{R}^n}\) and \(\max_{p \in \mathbb{R}^m}\) and isolate every term that depends on \(x\)

\[
\max_{p \in \mathbb{R}^m} \left( \min_{x \in \mathbb{R}^n} R(x) - \langle \Phi^* p, x \rangle \right) + \langle p, y \rangle - F^*(p).
\]

Notice that the parenthesis is nothing else than minus the convex conjugate of \(R\) evaluated at \(\Phi^* p\). Therefore, we define the *dual problem* as

\[
\hat{p} \in \text{Argmax}_{p \in \mathbb{R}^m} -R^*(\Phi^* p) + \langle p, y \rangle - F^*(p).
\]

Note that injecting a dual solution \(\hat{p}\) in (A.5) gives a necessary and sufficient condition for \(x\) to be solution to \((\mathcal{P}_F^R)\) and vice-versa. Fur-
thermore, for any pair \((\hat{x}, \hat{p})\) that is solution to (A.5) or equivalently to 
\((\mathcal{S}_F^R)\), \(\hat{x}\) is also solution to \((\mathcal{P}_F^R)\) and \(\hat{p}\) to \((\mathcal{D}_F^R)\).

A particular and very important form of the regularized regression problem \((\mathcal{P}_F^R)\), is when the fidelity is exact, \(i.e.,\) when \(F\) is the indicator function of an equality constraint with noiseless observations \(y_0 \triangleq \Phi x_0\),

\[
\min_{x \in \mathbb{R}^n} R(x) \quad \text{s.t.} \quad y_0 = \Phi x. \tag{P_0^R}
\]

The first order optimality condition states in that case that \(x_0\) is a solution of \((P_0^R)\) if and only if the set

\[
\mathcal{D}_{x_0} \triangleq \{ p \in \mathbb{R}^m \mid \Phi^* p \in \partial R(x_0) \} \tag{A.6}
\]

is not empty, \(i.e.,\) \(\mathcal{D}_{x_0} \neq \emptyset\). This set \(\mathcal{D}_{x_0}\) is often referred to as the set of dual certificates\(^1\) (see for instance [Vaiter et al., 2014]). The certificates are called non-degenerate when the inclusion is strict, \(i.e.,\) \(\Phi^* p \in \text{ri} \partial R(x_0)\).

\(^1\)In the literature, the word certificate sometimes refers to \(\eta = \Phi^* p\) instead of \(p\) which are then called pre-certificates.