Computational Imaging and Spectroscopy Sparse Coding and inverse problems

Thierry SOREZE

DTU July 2024

DTU Fotonik
Department of Photonics Engineering

Proximal Splitting

Problem formulation

Many problems in image processing can be formulated as the inversion of the system:

$$y = \mathbf{H}x_0 \odot \varepsilon$$

Where $x_0 \in \mathbb{R}^N$, is the signal we seek to recover, $y \in \mathbb{R}^m$ is the vector of corrupted observations. $\odot$ is either $+$ or $\times$, and ε is an unknown noise model. $\mathbf{H} : \mathbb{R}^N \rightarrow \mathbb{R}^m$ is a linear operator, typically ill behaved.

This problem is generally ill-posed, and need to be regularized, by reducing the space of candidate solutions, by adding prior knowledge on the structure of the unknown vector x_0

Proximal Splitting

Problem formulation

Deconvolution: $\mathbf{H}$ is the convolution by a blurring kernel, y lacks the high frequency components of x_0

Inpainting: $\mathbf{H}$ is a pixelwise multiplication by a binary mask

Decoding (Compressed Sensing): $\mathbf{H}$ is an $m \times N$ sensing matrix taking $m \ll N$ measurements at random from the input signal x_0, supposed to be sparse in a dictionary $\mathbf{\Phi}$

Proximal Splitting

Sparsity regularized inverse problems

Here we assume that the solution of our initial problem is sparsely represented in some dictionary $\mathbf{\Phi}$

The problem we which to solve is then cast as the following composite and structured minimization problem (P):

$$\min_{x \in \mathbb{R}^N} D(\mathbf{H}x, y) + \sum_{k=1}^{l} R_k(x)$$

$D: \mathbb{R}^m \times \mathbb{R}^m$ is a function measuring the consistency to the observed data y , and R_k are functions encoding the priors to be imposed on the signal to be recovered, x_0

We consider $D(\cdot, y), \forall y,$ and $R_k \forall k$ to be lower semi continuous convex functions. R_1 is usually a sparsity promotion constraint

Proximal Splitting

Sparsity regularized inverse problems

i. Synthesis Sparsity problems

We seek a sparse set of coefficients α and solution image which is synthesized from these coefficients as $x = \mathbf{\Phi}\alpha$, where $\mathbf{\Phi} \in \mathbb{R}^{N\times T}$. This type of prior is called synthesis sparsity prior.

The ℓ_1 decoder known as **Basis Pursuit** in the literature reads:

$$\min_{\alpha\in\mathbb{R}^T} \|\alpha\|_1 \quad \text{s.t} \quad y = \mathbf{H\Phi}\alpha = \mathbf{F}\alpha$$

This is an instance of (P) in which $l = 1$

$$R_1(x) = \min_{\alpha\in\mathbb{R}^T} \|\alpha\|_1 \quad \text{s.t} \quad x = \mathbf{\Phi}\alpha$$

Proximal Splitting

Sparsity regularized inverse problems

i. Synthesis Sparsity problems

$D(\mathbf{H}\,\cdot, y)$ is the indicator function of the affine subspace $\{x \in \mathbb{R}^N | y = \mathbf{H}x\}$

In presence of noise, the equality constraint must be relaxed to a noise aware variant. (P) becomes Basis Pursuit DeNoising (BPDN) when

$$\min_{\alpha \in \mathbb{R}^T} \frac{1}{2} \|y - \mathbf{F}\alpha\|^2 + \lambda \|\alpha\|_1, \qquad \lambda > 0$$

This setting is also known as Lasso in the literature, in the ℓ_1 constrained form

$$\min_{\alpha \in \mathbb{R}^T} \frac{1}{2} \|y - \mathbf{F}\alpha\|^2 \ \text{s.t} \quad \|\alpha\|_1 \le \rho$$

Proximal Splitting

Sparsity regularized inverse problems

ii. Analysis Sparsity problems

In the analysis sparsity prior framework, we seek a solution image x whose coefficients $\mathbf{\Phi}^{\mathrm{T}}x$ are sparse. The ℓ_1-analysis prior formulation of the previous equations are given by:

$$\min_{x\in\mathbb{R}^N}\left\|\mathbf{\Phi}^{\mathrm{T}}x\right\|_1 \qquad \mathrm{s.t} \quad y=\mathbf{H}x$$

$$\min_{x\in\mathbb{R}^N}\frac{1}{2}\|y-\mathbf{H}x\|^2+\lambda\left\|\mathbf{\Phi}^{\mathrm{T}}x\right\|_1$$

Proximal Splitting

ISTA for BPDN

Algorithm: *Iterative Soft-Thresholding Algorithm*

Init: choose some $\alpha^{(0)}$, $\mu \in \left]\frac{2}{|||\mathbf{H}|||^2|||\Phi|||^2}\right[$, $\tau_t \in [0,\kappa], \kappa = \frac{4-\mu\beta|||\mathbf{H}|||^2|||\Phi|||^2}{2} \in$ $]1,2[, \sum_{t\in\mathbb{N}} \tau_t(\kappa - \tau_t) = +\infty$

For t=0 to Niter -1 **do**

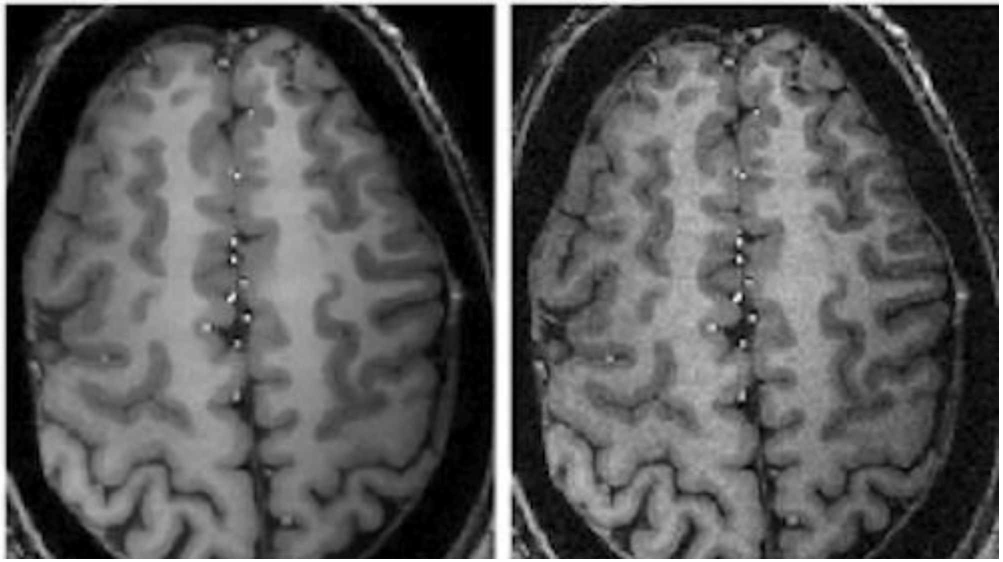
1. Gradient descent: $\alpha^{(t+1/2)} + \mu\mathbf{\Phi}^{\mathrm{T}}\mathbf{H}^{T}\left(y - \mathbf{H\Phi}\alpha^{(t)}\right)$
2. Soft-Thresholding: $\alpha^{(t+1)} = \alpha^{(t)} + \tau_t\left(\mathbf{SoftThresh}_{\mu\lambda}\left(\alpha^{(t+1/2)}\right) - \alpha^{(t)}\right)$

Proximal Splitting

Proximal Splitting framework

Splitting scheme	Objective	Asssumption
Forward-backward	$D(\mathbf{F}\,\cdot, y) + J_1$	J_1 simple, $\nabla D(\mathbf{F}\,\cdot, y)$ Lipschitz
FISTA	$D(\mathbf{F}\,\cdot, y) + J_1$	J_1 simple, $\nabla D(\mathbf{F}\,\cdot, y)$ Lipschitz
Douglas-Rachford	$D(\mathbf{F}\,\cdot, y) + J_1$	J_1 and $D(\mathbf{F}\,\cdot, y)$ simple
ADMM	$D(\mathbf{F}\,\cdot, y) + J_1$	J_1 and $D(\mathbf{F}\,\cdot, y)$ simple, $\mathbf{F}\mathbf{F}^{\mathrm{T}}$ invertible
GFB	$D(\mathbf{F}\,\cdot, y) + \sum_{k=1}^{l} J_k$	J_k simple, $\nabla D(\mathbf{F}\,\cdot, y)$ Lipschitz
Primal-dual	$D(\mathbf{F}\,\cdot, y) + J_1 + \sum_{k=2}^{l} G_k \circ \mathbf{A}_k$	J_1 and G_k simple, $\mathbf{A}_k$ linear, $\nabla D(\mathbf{F}\,\cdot, y)$ Lipschitz

COMPRESSED SENSING



Introduction and motivation

The acquisition of signals relies on the Nyquist-Shannon-Whitaker sampling scheme, which states that the sampling frequency ▯ should be at least $\geq 2 * Bandwich$

However this is restricted to the case of signals of **known band limit**, and implies a high number of measurements, resulting in high storage requirement

Most of the time measurements can be too long, costly, harmful, difficult to acquire, incomplete, or storage greedy

Introduction and motivation

Typically the storage problem is solved by dropping the less significant data after acquisition

All the above problems have led to a new sensing paradigm: Compressed sensing, or compressive sensing/sampling, or simply sparse recovery, allowing compression at the **acquisition time**

Introduction and motivation

Compressed Sensing (CS) is a mathematical theory that allows the recovery of incomplete, missing, undersampled, or more simply sparse, measurements under certain circumstances

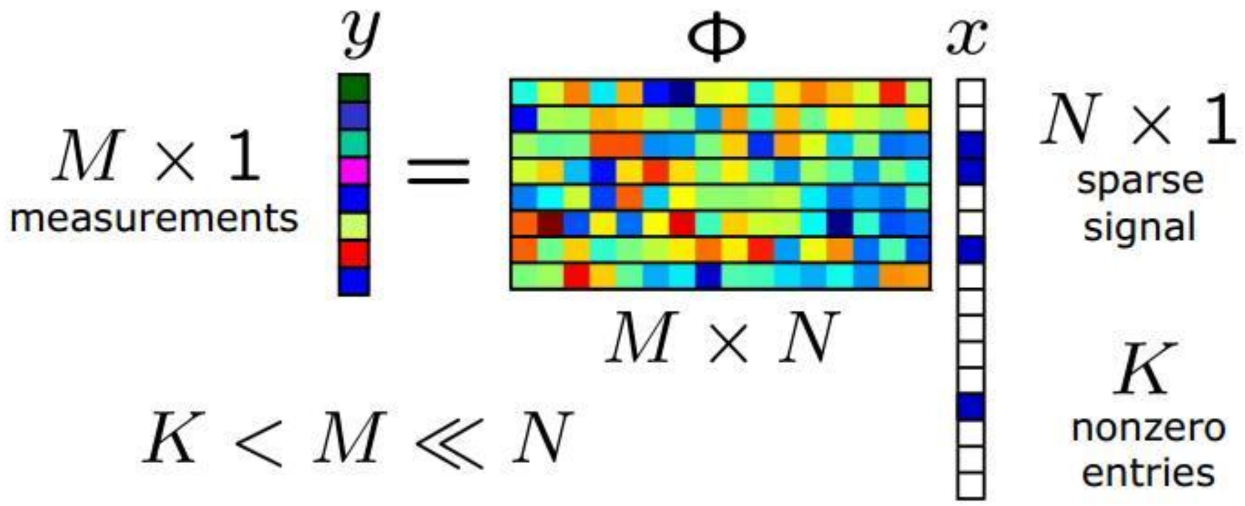
The problem that we wish to solve is the following:

$$y = Ax$$

With $x \in \mathbb{R}^N$ and $y \in \mathbb{R}^M$, $M \ll N$, $A \in \mathbb{R}^{M \times N}$

We assume that the signal of interest can be represented in a sparsifiyng dictionnary by $x = \mathbf{\Phi}\hat{x}$, $\mathbf{\Phi} \in \mathbb{R}^{N \times N}$

Introduction and motivation



It is easy to see that this is an **ill-posed** problem

The key idea of **CS** is to assume that the measured signal is **sparse**

Sparse recovery

Recovery algorithm

We need to solve an under determined system of equations $y = \boldsymbol{A}x$ assuming that x is sparse or compressible

The most intuitive solution to this problem is

$$(P_0) \qquad \min_x \|x\|_0 \quad s.t \;\; y = \boldsymbol{A}x$$

However this is a NP-hard problem, so we replace the ℓ_0-norm by the closest norm, the ℓ_1-norm and rewrite the solution as

$$(P_1) \qquad \min_x \|x\|_1 \quad s.t \;\; y = \boldsymbol{A}x$$

Sparse recovery

Condition for recovery, Spark and null space property

Let define the spark of a matrix $\boldsymbol{A} \in \mathbb{R}^{M \times N}$ as

$$Spark(A) \equiv \min\{k: \mathcal{N}(A) \cap \Sigma_k \neq \{0\}\}, Spark(A) \in [2, m+1]$$

With $\mathcal{N}(A)$ the null space of $\boldsymbol{A}$. Roughly speaking the spark of a matrice $\boldsymbol{A}$ is the minimal number of lineary dependent columns of A

If a solution x of (P_0) satisfies $\|x\|_0 \leq k$ then this solution is unique and $k < Spark(\boldsymbol{A})/2$.

A matrice $\boldsymbol{A}$ has the null space property (NSP) of order k if $\forall h \in \mathcal{N}(A) \setminus \{0\}$ and for all $|\Lambda| \leq k$ $\|1_\Lambda h\|_1 < \frac{1}{2}\|h\|_1$

Sparse recovery

Condition for recovery, mutual coherence

If a solution x of (P_1) satisfies $\|x\|_0 \leq k$ then this solution is unique and $\boldsymbol{A}$ satisfies the NSP of order k

Let $\boldsymbol{A}$ be a $m \times n$ matrix with coefficients $(a_i)_{i=1}^{n}$, then its *mutual coherence* $\mu(A)$ *is defined as*

$$\mu(\boldsymbol{A}) \equiv \max_{i \neq j} \frac{|\langle a_i, a_j \rangle|}{\|a_i\|_2 \|a_j\|_2}, \qquad \mu(\boldsymbol{A}) \in \left[\sqrt{\frac{n-m}{m(n-1)}}, 1\right]$$

We have for any matrix $\boldsymbol{A}$ $Spark(\boldsymbol{A}) \geq 1 + \frac{1}{\mu(A)}$

Let $x \in \mathbb{R}^n \setminus \{0\}$ be a solution of (P_0) satisfying $\|x\|_0 < \frac{1}{2}(1 + \mu(A)^{-1})$, then x is the unique solution of (P_0) and (P_1)

Sparse recovery

Condition for recovery, restricted isometry property

Let $\boldsymbol{A}$ be a $m \times n$ matrix, then $\boldsymbol{A}$ satisfies the *restricted isometry property(RIP)* of order k if $\exists \delta_k \in (0,1)$, or shortly *RIP*$(k, \delta)$, such that

$$(1-\delta_k)\|x\|_2^2 \leq \|\mathbf{A}_k x\|_2^2 \leq (1+\delta_k)\|x\|_2^2 \quad \forall x \ \in \Sigma_k$$

Assume that A is RIP of order $2k$ with $\delta_{2k} < \sqrt{2}-1$, then the solution x^* obeys

$$\|x^* - x\|_1 \leq C \cdot \sigma_k(x)_1$$

And

$$\|x^* - x\|_2 \leq C \cdot \left(\frac{\sigma_k(x)_1}{\sqrt{k}}\right)$$

For some constant C. If x is k-sparse the recovery is **exact**

Sparse recovery

Condition for recovery, noisy measurements

In real applications the measured signal could contain additive noise ε, with $\|\varepsilon\|_2 \leq \epsilon$, we will then solve

$$(P_2) \qquad \min_x \|x\|_1 \quad s.t \quad \|\boldsymbol{A}x - y\|_2 \leq \epsilon$$

And the solution x^* obeys

$$\|x^* - x\|_2 \leq C \cdot \left(\frac{\sigma_k(x)_2}{\sqrt{k}}\right) + C_1 \cdot \epsilon$$

Design of Compressive sampling matrices

Sensing matrice

In CS applications the matrix $\boldsymbol{A}$ has usually the following expression

$$\boldsymbol{A} = \boldsymbol{\Phi}\boldsymbol{\Psi}$$

$\boldsymbol{\Psi}$ is the sensing, or measurement matrix. $\boldsymbol{\Phi}$ is the sparsifying, or sparse representation matrix, thus we can rewrite the problem as follow

$$y = \boldsymbol{\Psi}x = \boldsymbol{\Psi}\widehat{\boldsymbol{\Phi}}\boldsymbol{x}$$

With $x = \boldsymbol{\Phi}\hat{x} = \sum_{\boldsymbol{i=1}} \boldsymbol{\varphi_i}\, \hat{x}_{\boldsymbol{i}}$. We can define the mutual coherence between the sensing matrix and the representation matrix

$$\mu(\boldsymbol{\Phi}, \boldsymbol{\Psi}\ ) \equiv \sqrt{n} \cdot \max_{1 \le k, j \le n} \left| \left\langle \varphi_k, \psi_j \right\rangle \right|, \qquad \mu(\boldsymbol{\Phi}, \boldsymbol{\Psi}\ ) \in [1, \sqrt{n}]$$

Design of Compressive sampling matrices

Random matrices

The lower the mutual coherence the better, as typically we have the following constraint on the minimum number of measurements, m, needed to ensure the recovery

$$m \geq C \cdot \mu^2(\boldsymbol{\Phi}, \boldsymbol{\Psi}) \cdot k \cdot \ln n$$

The incoherence between $\boldsymbol{\Phi}$ and $\boldsymbol{\Psi}$ expresses the fact that $\boldsymbol{\Phi}$ is not sparse in $\boldsymbol{\Psi}$.

Design of Compressive sampling matrices

Random matrices

The mutual coherence is maximally incoherent, i.e. unity, for the spike, dirac, basis associated with the Fourier basis, which corresponds to the regular Shannon-Whitaker sampling scheme

For instance dirac and Fourier basis are maximally incoherent, noiselets are maximally incoherent with wavelets, about $\sqrt{2}$ between noiselets and Haar basis, about 2.2 and 2.9 for Daubechies D4 and D8 wavelets

Design of Compressive sampling matrices

Random matrices

It is well known that a randomly sampled sensing matrix $\boldsymbol{\Psi}$ will be incoherent with any sparsifying basis $\boldsymbol{\Phi}$, thus providing universality

Linear and **incoherent** measurements ensure both low *mutual coherence* and *RIP*, hence their importance in CS

Matrices populated uniformely at random with n orthonomalized vectors have a very low mutual coherence, classical matrices of this type are Gaussian and Bernoulli matrices

Design of Compressive sampling matrices

Structured random matrices

However random sensing matrices have some drawbacks when it comes to real applications.

To overcome this one can subsample randomly an incoherent sensing matrix from an existing structured sensing matrix, leading to a new expression of the CS matrix A

$$\boldsymbol{A} = \boldsymbol{R\Phi\Psi}$$

With $\boldsymbol{R}$ a $m \times n$ matrix that samples randomly $m \times n$ elements from $\boldsymbol{\Psi}$. $\boldsymbol{\Phi}$ and $\boldsymbol{\Psi}$ are $n \times n$ matrices

Examples

Random Fourier ensemble

Fourier matrices offer fast transforms through FFT

$$\boldsymbol{A} = \boldsymbol{R}\boldsymbol{F}\boldsymbol{\Phi}$$

With $\boldsymbol{F} \in \mathbb{R}^{\boldsymbol{N}}$ is the Discrete Fourier transform on $\mathbb{R}^{\boldsymbol{N}}$, $\boldsymbol{R}$ is $m \times n$ matrix that samples randomly $m \times n$ elements from $\boldsymbol{F}$.$\boldsymbol{\Phi}$is a canonical sparsity basis

These matrices are used in MRI, radio interferometry, etc.

The number of measurements needed is of the order of $\mathcal{O}(k \cdot (\ln N)^4)$, some worked applications have reported $\mathcal{O}(k \cdot \ln N)$

Examples

Random Convolution

Convolutions are important in optical engineering as they are used to compute *point spread functions* (PSF), making random convolution ensemble very useful for CS imaging

$$\boldsymbol{A} = (\boldsymbol{R}\boldsymbol{F}^{*}\boldsymbol{\Sigma}\boldsymbol{F})\boldsymbol{\Phi}$$

With $\boldsymbol{\Sigma} \in \mathbb{R}^{\boldsymbol{N}\times\boldsymbol{N}}$ is a complex diagonal matrix, with unit amplitude diagonal element and random phase. Contrary to Fourier ensemble the sensing matrix $(\boldsymbol{R}\boldsymbol{F}^{*}\boldsymbol{\Sigma}\boldsymbol{F})$ is incoherent with any sparse basis $\boldsymbol{\Phi}$

It satisfies a mutual incoherence of $\mu(\boldsymbol{R}\boldsymbol{F}^{*}\boldsymbol{\Sigma}\boldsymbol{F},\boldsymbol{\Phi}) \leq \boldsymbol{2}\sqrt{(\ln \boldsymbol{2}\boldsymbol{N}^{\boldsymbol{2}}/\boldsymbol{\gamma})/\boldsymbol{N}}$, with $0 < \boldsymbol{\gamma} < \boldsymbol{1}$ and requires about $\mathcal{O}(k \cdot (\ln N)^{5})$ measurements

Examples

Separable matrices

In some applications it is required to measure very large multi-dimensional dataset, for instance in hyper spectral imaging. If each dimension are separable, the corresponding sensing matrix lead to kronecker products,

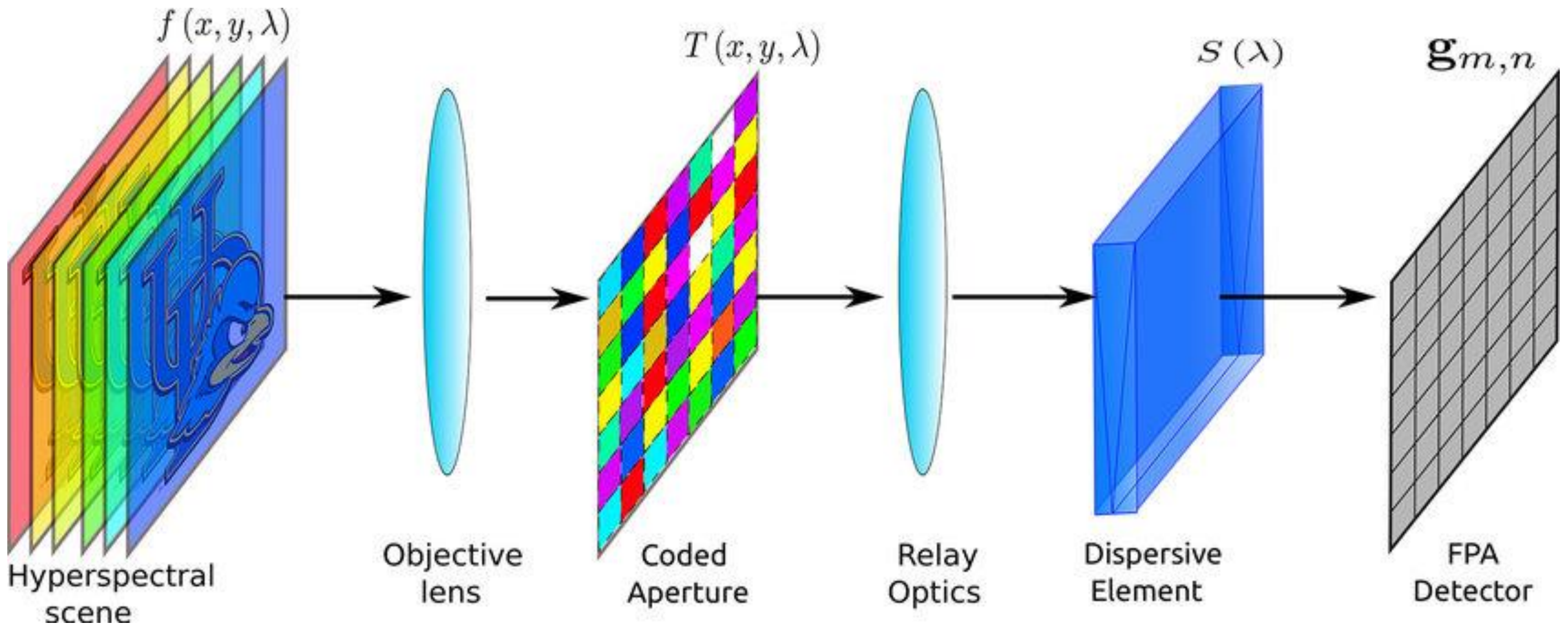
$$\boldsymbol{A}\otimes\boldsymbol{B} = \begin{pmatrix} \boldsymbol{a_{11}B} & \cdots & \boldsymbol{a_{1n}B} \\ \vdots & \ddots & \vdots \\ \boldsymbol{a_{m1}B} & \cdots & \boldsymbol{a_{mn}B} \end{pmatrix}$$

In CS application the sensing matrix for a D-dimensional signal will be then expressed as a Kronecker product of D matrices

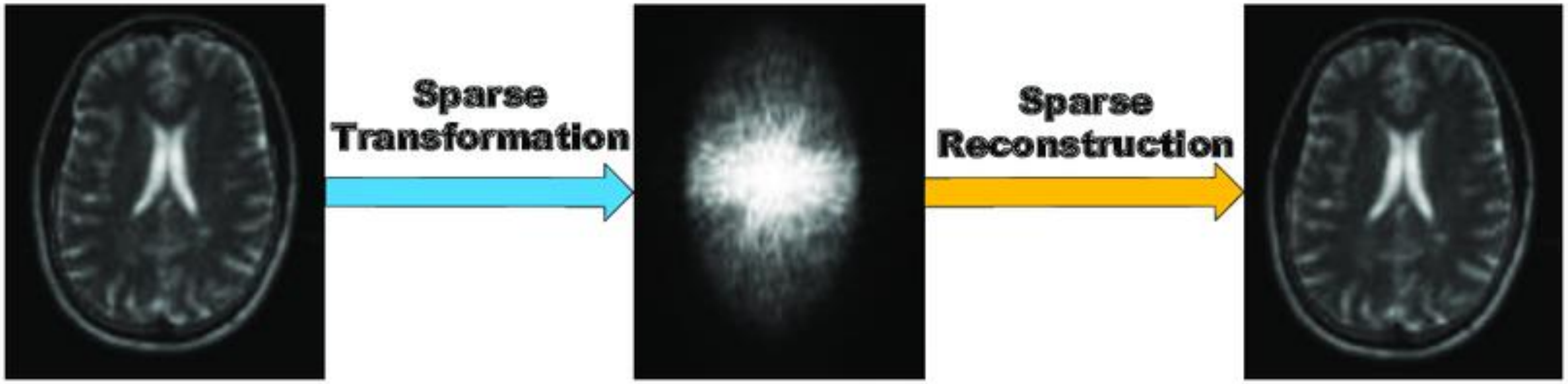
$$\boldsymbol{\Phi}_{Kronecker} = \boldsymbol{\Phi}_1 \otimes \boldsymbol{\Phi}_2 \cdots \otimes \boldsymbol{\Phi}_D$$

Example: Compressive imaging

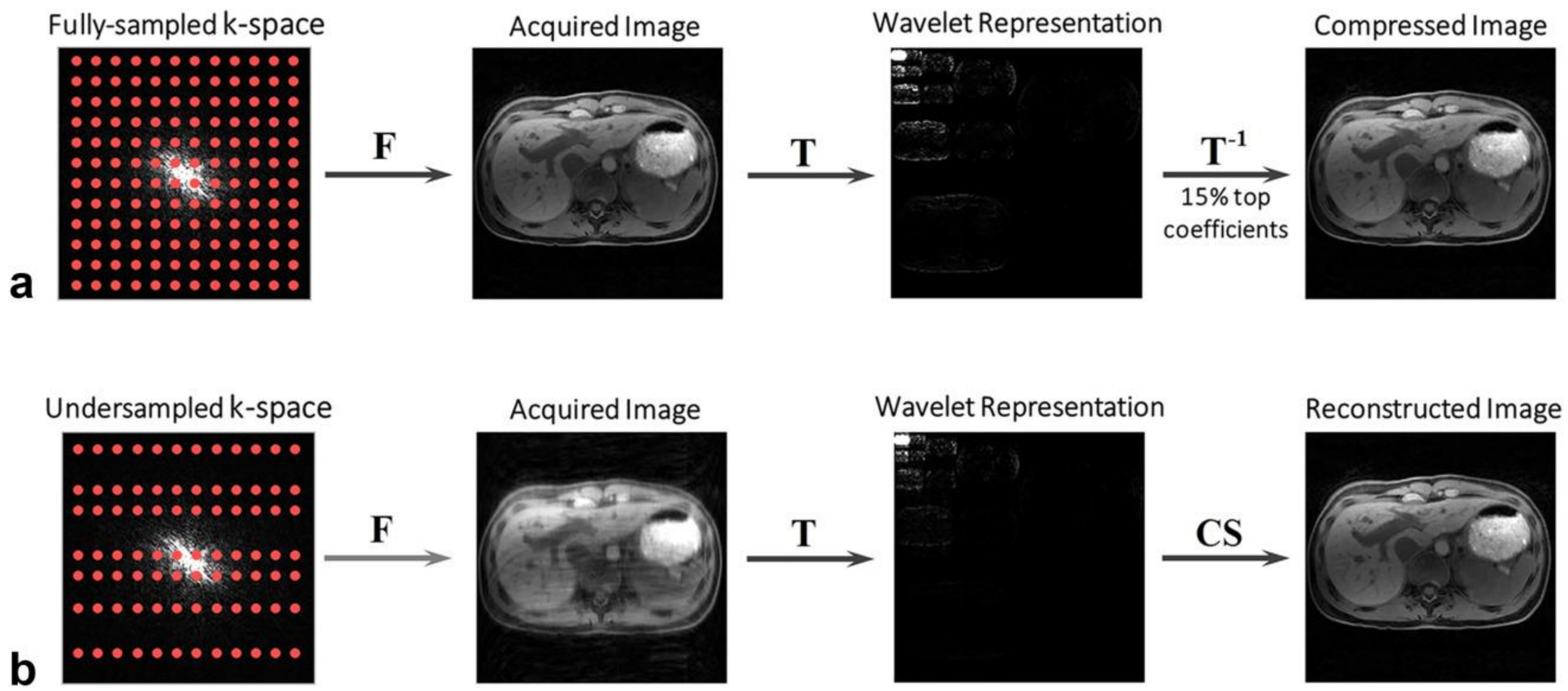
Coded Aperture compreSsed Sensing Imaging (CASSI)



Example: Compressive imaging



Example: Compressive imaging



Hands on: Compressive imaging

Examples

- https://www.kaggle.com/code/brucelangford/an-introduction-to-compressed-sensing
- https://pyunlocbox.readthedocs.io/en/stable/tutorials/compressed_sensing_forward_backward.html
- https://scikit-learn.org/stable/auto_examples/applications/plot_tomography_l1_reconstruction.html

Python tools

- https://pyunlocbox.readthedocs.io/en/stable/index.html
- https://cvxopt.org/
- https://scikit-learn.org/0.15/modules/generated/sklearn.linear_model.Lasso.html
- https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.OrthogonalMatchingPursuit.html