Foundations of Smart Sensing
Compressive Sensing

MSc in Statistics for Smart Data - ENSAI

Aline Roumy

December 2020

Outline

About me

Aline Roumy
Researcher at Inria, Rennes
Expertise: compression for video streaming
image/signal processing, information theory, machine learning

Web: http://people.rennes.inria.fr/Aline.Roumy/
email: aline.roumy@inria.fr

Course schedule (tentative)

Compressive sensing (CS): a self-sufficient course with a lot of connections to sparse approximations

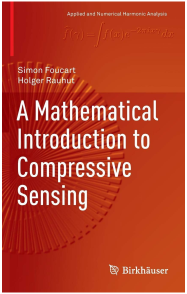
- Dec 2nd: Lecture (CS: intro+ how it works)
- Dec 4th: Lab
- Dec 10th at **9am**: Lab (no course in the afternoon)

Course grading

- **Final Exam:** about lectures 1-9 (C. Elvira, J. Cohen, C. Herzet)
  - written exam (Dec 16th)
  - 2 hours - No document.
- **Project:** about lectures 10-12 (A. Roumy)
  - Part 1- Summary of the course
    (half page of text not including eventual figures)
  - Part 2- Computer lab (Dec 4 and 10th)
    - using Collaborative Jupyter notebook
    - write a **short** report (with jupyter) on the lab activities:
      max 2 pages for the comments (excluding proofs, figures)
    - send the pdf file + code files via email to `aline.roumy@inria.fr`
    - You will get a grade from the evaluation of your report.
  - **deadline Dec. 10th 8pm**
- TO DO:
  - after 1st course: read the course and write the course summary. Get familiar with Collaborative Jupyter notebook
  - after 2nd course: augment/correct the course summary. Add comments in YOUR version of the code.
  - 3rd course: Add comments in YOUR version of the code. Do the final question.

Course material

S. Foucart, H. Rauhut, A mathematical introduction to compressive sensing, Birkhaüser, 2013.

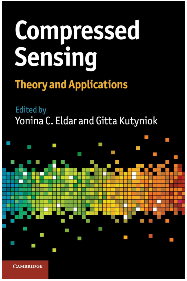


Early and short version:

S. Foucart, Notes on compressed sensing, 2009. (pdf)

Course material

Compressed Sensing: Theory and Applications, Edited by Y.C. Eldar and G. Kutyniok, Cambridge University Press, 2012.



- Chapter 1:
  M.A. Davenport, M.F. Duarte, Y.C. Eldar, G. Kutyniok Introduction to compressed sensing. **(pdf)**
- Short version:
  G. Kutyniok, Theory and Applications of Compressed Sensing, GAMM Mitteilungen 36 (2013), 79-101.

Outline

Part 1 - Why compressive sensing?

What is compressive sensing?

Compressive sensing:
is a novel way to acquire (or sense or sample) and compress data.

Classical = sampling then compression

Compressive sensing = sampling **AND** compression

Several names exist:

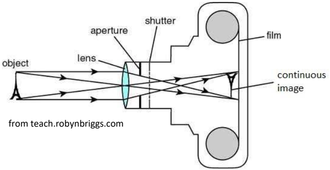
- compressed sensing
- compressed sampling
- compressive sampling
- **compressive sensing**. More accurate. Chosen in this course.
  The one of the reference book.

Part 1 - Why compressive sensing?

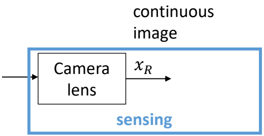
Review of **classical** digital acquisition:
classical=sampling + compression

Film camera

Film camera: records images passing through the camera's lens.

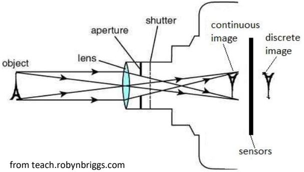


$x_R : [0,1]^2 \to \mathbb{R}^3$



Digital camera

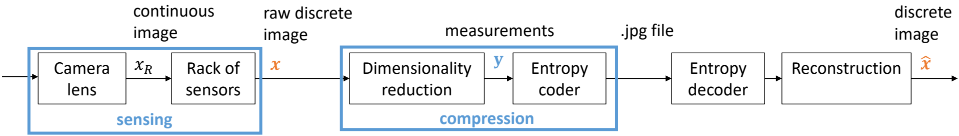
Digital camera: converts an image into **digital** data and **compress** it.



$x_R : [0,1]^2 \to \mathbb{R}^3$

$\mathbf{x} : \{1, N_a\} \times \{1, N_b\} \to \{0{,}255\}^3$

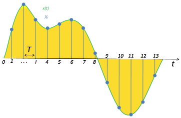
$\mathbf{y} : \{1, M_a\} \times \{1, M_b\} \to \{0{,}255\}^3$



In this course: focus on the signal processing processes i.e. sensing + dimensionality reduction
entropy coder is an invertible process (from samples to bit)

Questions related to Digital camera

Question related to sampling:
is it possible to recover a continuous signal from its sampled (discrete) version?



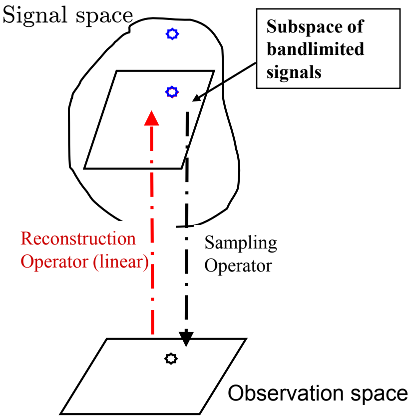
Wikipedia.

cf. course of Clément Elvira

Question related to compression:
is it possible to reduce the size of a discrete image?

Sampling: (1) optimal sampling rate

Nyquist–Shannon sampling theorem: "Exact reconstruction of a continuous-time signal from discrete samples is possible if the signal is bandlimited and the sampling frequency is greater than twice the highest frequency."

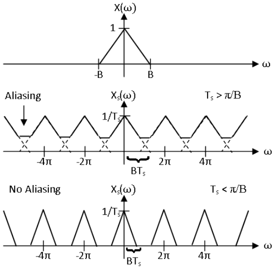


Mike Davies.

Sampling: (2) degradation if “slow” sampling

Sampling below the optimal rate introduces:

(1) aliasing

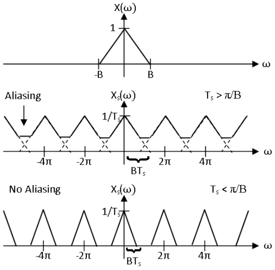


SVI.nl

Sampling: (2) degradation if "slow" sampling

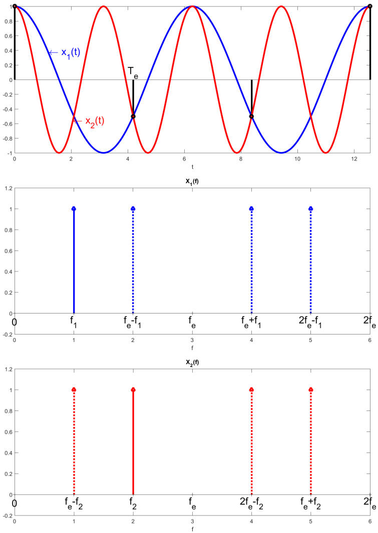
Sampling below the optimal rate introduces:

(1) aliasing



SVI.nl

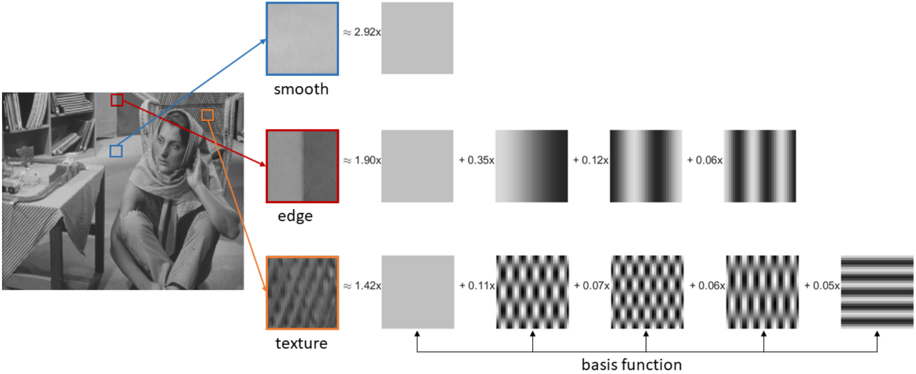
(2) signal ambiguity



Compression: (1) image decomposition principle

1- Split image into blocks of size $N_1 \times N_2$ each

2- Decompose each $N_1 \times N_2$ image block as:



How to choose the basis functions? How to compute the coefficients?

Compression: (2) image decomposition example

with 2D-discrete cosine transform (DCT) (orthogonal basis)

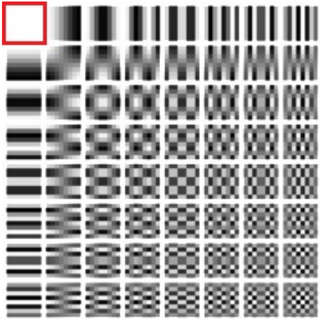
1- Split image into blocks of size $N_1 \times N_2$ each

2- For each $N_1 \times N_2$ image block (x_{n_1,n_2})
compute the $N_1 \times N_2$ block of transformed image (c_{k_1,k_2}) with:

$$c_{k_1,k_2} = \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} x_{n_1,n_2} \underbrace{\cos\left[\frac{\pi}{N_1}\left(n_1 + \frac{1}{2}\right) k_1\right] \cos\left[\frac{\pi}{N_2}\left(n_2 + \frac{1}{2}\right) k_2\right]}_{\Phi_{n_1,n_2}(k_1,k_2)}$$

Example: 8x8 DCT transform
Top-left matrix is $(\Phi_{n_1,n_2}(k_1=0, k_2=0))_{n_1,n_2}$

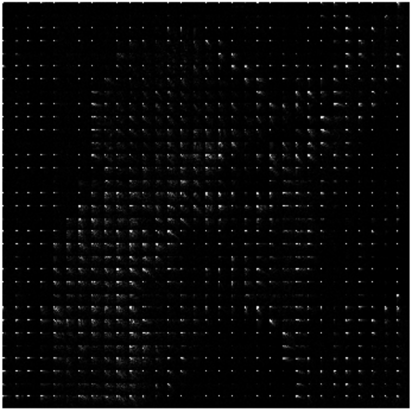
Quiz 1, 2, 3



Compression: (3) image decomposition result

Left: image

Right: discrete cosine transform of image



Key concept: few degrees of freedom in the transform domain

Compression: (4) dimensionality reduction with s-term approximation

1. Dimensionality reduction:

   keep the s coefficients c_s with largest absolute value

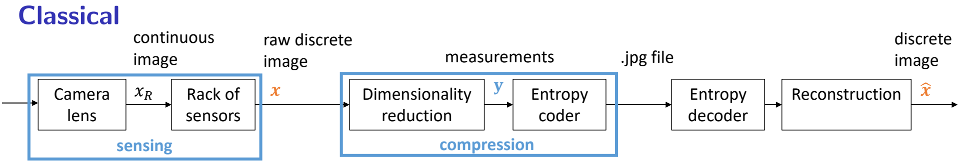
2. Reconstruction: $\hat{x} = \Phi^{-1} c_s$

Left: 1% kept

Right:5% kept

Quiz 4

Summary on classical sensing



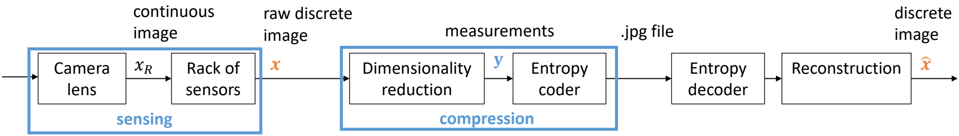
Sampling raw discrete HD video
1920x1080=2.07 M pixels/image
25Hz: images/s,
12(=8+2+2) bits/pixel
$\rightarrow$ 0.6 Gbit/s

Compression
For instance, HEVC (2013)
0.6 Gbit/s $\rightarrow$ 2Mbit/s

compression ratio 300:1!!!

Classical vs compressive sensing

Classical

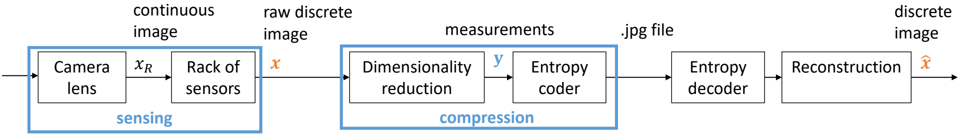


lots of samples,
throw most of the coefficients away

x: $[1, N_a] \times [1, N_b] \to \{0, 255\}^3$
y: $[1, M_a] \times [1, M_b] \to \{0, 255\}^3$
$(M_a M_b \ll N_a N_b)$

Classical vs compressive sensing

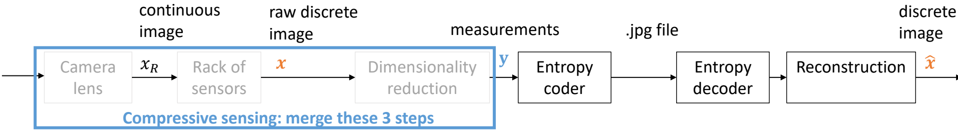
Classical



lots of samples,
throw most of the coefficients away

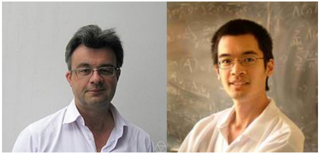
x: $[1, N_a] \times [1, N_b] \to \{0, 255\}^3$
y: $[1, M_a] \times [1, M_b] \to \{0, 255\}^3$
$(M_a M_b \ll N_a N_b)$

Compressive sensing: can we acquire less data in the first place?
and still recover $\hat{x}$?



Can we sample signals at the "Information Rate"?

Yes, we can!



Wikipedia.

E. J. Candes and T. Tao, 2005
"Decoding by linear programming"

Wikipedia.

D. L. Donoho, 2006
"Compressed sensing"

Outline

Part 2 - Maths of compressive sensing - how it works?

Notations (Reminder)

Norms

Definition (l_p-norm)

The l_p-**norm** of $x \in \mathbb{R}^n$, $p > 1$ is defined as

$$||x||_p = \begin{cases} \left(\sum_{i=1}^{n} |x_i|^p\right)^{1/p} & p \in [1, \infty) \\ \max_i |x_i| & p = \infty \end{cases}$$

If $p < 1$, definition still valid, but triangle inequality not satisfied $\Rightarrow$ **quasi-norm**.

Definition (inner product)

$$\langle x, z \rangle = z^T x = \sum_{i=1}^{n} x_i z_i$$

See textbook F.R. for extension to $\mathbb{C}^n$.

Definition (support and l_0-norm)

The **support** of a vector x is the index set of its non-zero entries, i.e.

$$\text{supp}\ (x) = \{j \in [n] : x_j \neq 0\},\ \text{ where } [n] = \{1, 2, ..., n\}$$

The **l_0-norm** of x is defined as

$$||x||_0 = \ \text{card}\ (\ \text{supp}\ (x)\ )$$

$||x||_0$ counts the number of non-zero entries of x.
$||.||_0$ is not even a quasi-norm.

Sparsity definition

Definition (s-sparse)

A signal $x \in \mathbb{R}^n$ is said to be s-sparse if it has at most s non-zero entries, i.e. $||x||_0 \leq s$.

Definition (Σ_s)

We define Σ_s as the set containing all s-sparse signals, i.e. $\Sigma_s = \{x \in \mathbb{R}^n : ||x||_0 \leq s\}$.

Quiz 5

Note 1: Sparsity is a highly nonlinear model (Σ_s is not a linear space)

Note 2: in many practical cases, x is not sparse itself, but it has a sparse representation in some basis Φ. We still say that x is s-sparse, with the understanding that we can write $x = \Phi u$, and $||u||_0 \leq s$.

Approximate sparsity

- A sparse signal can be represented exactly giving the positions and values of its s nonzero components

- Real-world signals are rarely exactly sparse.
  We need to
  - generalize the def: from "sparse" to "compressible" signals,
  - describe the representation error i.e. the error incurred representing just s components of the signal.

Best s-term approximation

The best s-term approximation picks the s components that minimize the representation error

Definition (best s-term approximation)

For $p > 0$, the l_p-error incurred by the best s-term approximation to a vector $x \in \mathbb{R}^n$ is given by

$$\sigma_s(x)_p = \min_{\hat{x} \in \Sigma_s} ||x - \hat{x}||_p$$

- If $x \in \Sigma_s$, then $\sigma_s(x)_p = 0$ for any p.

Compressible signal

Optimal strategy to compute the best s-term approximation:
thresholding

- Reorder the elements of x by decreasing magnitude
- Pick the first s elements, set all others to zero.

Definition (compressible signal)

a signal $x \in \mathbb{R}^n$ is said to be compressible if the error of its best s-term approximation decays quickly in s
i.e. if $\exists C_1, q > 0$ such that $|x_i| \leq C_1 i^{-q}$., when the coefficients have been ordered such that $|x_1| \geq |x_2| ... \geq |x_n|$.

Sparsity support

Suppose $x \in R^n$. Let $S \subset [n]$ and $S^c \subset [n] \backslash S$

- S: sparsity support of x, i.e. the locations of the nonzero coefficients of x
- S^c: set of locations of the 0 coefficients
- S for compressible signal: set of locations of the coefficients belonging to the best s-term approximation of x.

Notation

x_S vector obtained by setting the entries of x indexed by S^c to 0.

M_S matrix obtained by setting the columns of M indexed by S^c to 0.

- Same notation to denote vectors/matrices where the elements/columns have been removed, instead of being set to 0

Outline

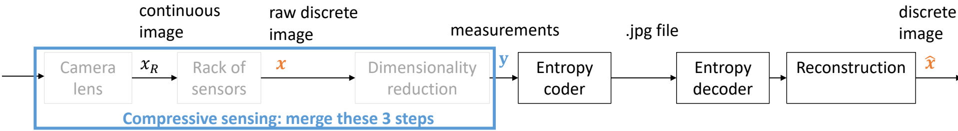
Part 2 - Maths of compressive sensing - how it works?

Problem formulation

Compressive sensing

Goal of Compressive sensing (CS):

- achieve the same reconstruction quality on $\hat{x}$ as the best s-term approximation
- from the measurement y acquired with a nonadaptive encoder.



To achieve this, we need to

1. model the dependency between signal x and measurement y
2. formulate the reconstruction problem

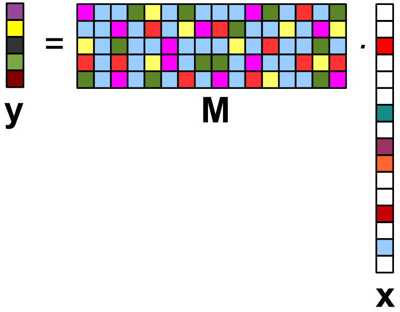
Sensing process model

(Modeling the dependency between signal and measurement)

Let $x \in R^{nx1}$ *be a s-sparse signal to be recovered.*
Let $y \in R^{mx1}$, $m < n$, *be linear measurements of the signal as*

$$y = Mx$$

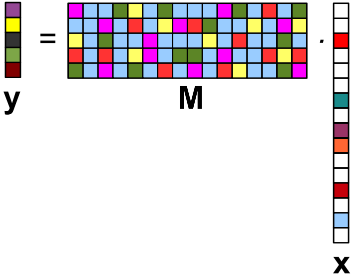
with $M \in R^{mxn}$, *being the sensing matrix.*



Reconstruction: problem formulation

(problem formulation)

Given measurement y, sensing matrix M and the model $y = Mx$, Recover x, s-sparse.

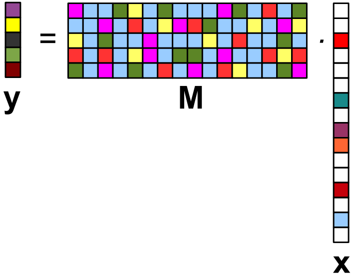


Difficulties?

Reconstruction: problem formulation

(problem formulation)

Given measurement y, sensing matrix M and the model $y = Mx$, Recover x, s-sparse.



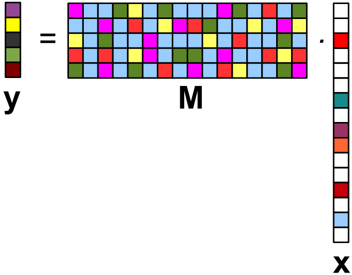
Difficulties?

- Underdetermined system $\Rightarrow$ infinitely many solutions.

Reconstruction: problem formulation

(problem formulation)

Given measurement y, sensing matrix M and the model $y = Mx$, Recover x, s-sparse.



Difficulties?

- Underdetermined system $\Rightarrow$ infinitely many solutions.
- **Idea** exploit the sparsity assumption of x.

Minimum l_0-norm solution

$$\hat{x} = \arg\min_{z \in \mathbb{R}^n} ||z||_0 \text{ subject to } Mz = y$$

Complexity?

- Problem is non-convex
- Problem is NP-hard:
  for a given s, try all possible $\binom{n}{s}$ supports, estimate the s nonzero values of x, check if constraint is satisfied
  $\Rightarrow$ **infeasible** for practical problem sizes

Practical philosophies

$$\hat{x} = \arg\min_{z \in \mathbb{R}^n} ||z||_0 \textit{ subject to } Mz = y$$

Greedy algorithms

Focus on $||x||_0$

Thresholding algorithms

Focus on $y \sim Mx$

Convex relaxation algorithms

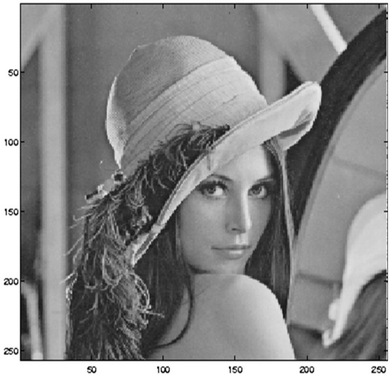
Solve a nicer problem

see course C. Elvira

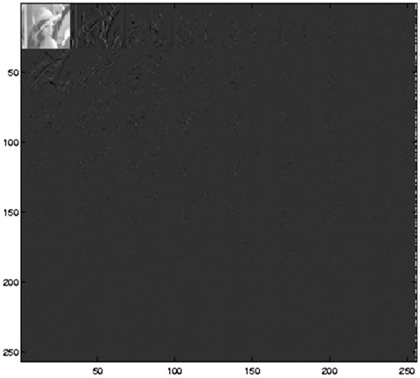
Signal sparse in transform domain

Real signals are rarely directly sparse...

but rather sparse in a transform domain



original image



DCT coefficients of the image
in the transform domain

Signal sparse vs signal sparse in transform domain

x sparse

SENSING
$y = Mx$

RECONSTRUCTION
$\hat{x} = \arg\min_{z \in \mathbb{R}^n} ||z||_1$
subject to $Mz = y$
.

$x = \Phi u$, u sparse

SENSING
$y = Mx$

RECONSTRUCTION
$\hat{u} = \arg\min_{z \in \mathbb{R}^n} ||z||_1$
subject to $M\Phi z = y$
$\hat{x} = \Phi \hat{u}$

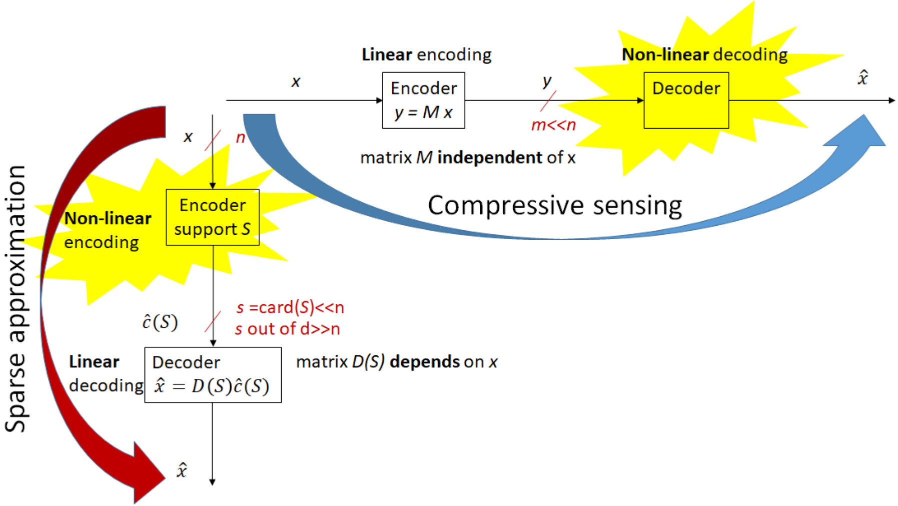
In conclusion: sparse vs sparse in the transform domain

- same sensing
- similar reconstruction problem
- Make sure that $M\Phi$ (and not M) is a “good” sensing matrix

Part 2 - Maths of compressive sensing - how it works?

Compressive sensing vs other schemes

Compressive sensing (CS) vs Sparse approximation (SA)



CS vs SA (con't)

Non-linear solvers:

CS Given y and M, find $\hat{x}$ sparse such that $M\hat{x} \approx y$.

Return $\hat{x}$ with guarantee that

$$||\hat{x} - x|| \quad \text{small}$$

SA Given x and D, find $\hat{c}$ sparse such that $\hat{x} = D\hat{c} \approx x$.

Return $\hat{x}$ with guarantee that

$$||\hat{x} - x|| = ||D(\hat{c} - c)|| \quad \text{small}$$

CS vs SA (con't)

Non-linear solvers:

CS Given y and M, find $\hat{x}$ sparse such that $M\hat{x} \approx y$.

Return $\hat{x}$ with guarantee that

$$||\hat{x} - x|| \quad \text{small}$$

SA Given x and D, find $\hat{c}$ sparse such that $\hat{x} = D\hat{c} \approx x$.

Return $\hat{x}$ with guarantee that

$$||\hat{x} - x|| = ||D(\hat{c} - c)|| \quad \text{small}$$

Same decomposition algorithms

Different criteria

CS vs SA (con't)

Non-linear solvers:

CS Given y and M, find $\hat{x}$ sparse such that $M\hat{x} \approx y$.

Return $\hat{x}$ with guarantee that

$$||\hat{x} - x|| \quad \text{small}$$

SA Given x and D, find $\hat{c}$ sparse such that $\hat{x} = D\hat{c} \approx x$.

Return $\hat{x}$ with guarantee that

$$||\hat{x} - x|| = ||D(\hat{c} - c)|| \quad \text{small}$$

Root-finding algorithm:

CS Given $y = 0$ and f, find $\hat{x}$ such that $y = 0 \approx f(\hat{x})$.

Return $\hat{x}$ with guarantee that

$$||\hat{x} - x|| \quad \text{small}$$

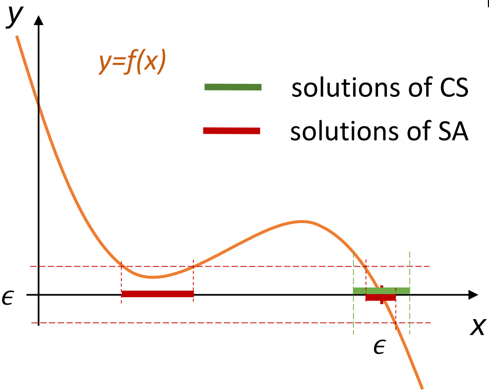
SA Given $y = 0$ and f, find $\hat{x}$ such that $y = 0 \approx \hat{y} = f(\hat{x})$.

Return $\hat{y}$ with guarantee that

$$||f(\hat{x}) - 0|| \quad \text{small}$$

CS: proximity to the true root
SA: proximity to zero in the range of the function

CS vs SA (con't)



Root-finding algorithm:

CS Given $y = 0$ and f, find $\hat{x}$ such that $y = 0 \approx f(\hat{x})$.
Return $\hat{x}$ with guarantee that

$||\hat{x} - x||$ small

SA Given $y = 0$ and f, find $\hat{x}$ such that $y = 0 \approx \hat{y} = f(\hat{x})$.
Return $\hat{y}$ with guarantee that

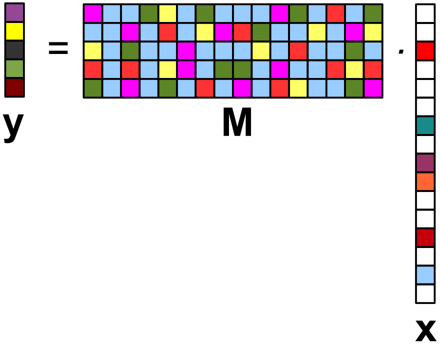
$||f(\hat{x}) - 0||$ small

CS: proximity to the true root
SA: proximity to zero in the range of the function

Part 3 - Compressive sensing - good sensing matrix?

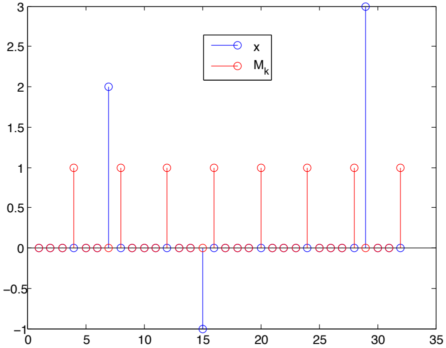
First insights

Sensing process



- How should we choose a "good" matrix M with $m \ll n$?

Sensing matrices that are not good



Vector y is all zero!
→ If x sparse, M must be non-sparse
→ We need M to be different from x

Good sensing matrices

- if A follows a subgaussian distribution with $m \geq c\ s \ln(n/s)$, c=constant,

  [easy construction / easy to verify...]

  then with probability at least $1 - 2e^{-c_0 m}$, c_0= constant

  exact reconstruction under P_1, OMP, IHT...

- Gaussian, Bernoulli (Rademacher entries) matrices ..., subsampled Fourier matrices achieve exact reconstruction.
- the constant c depends on the algorithm and the sensing matrix distribution.

Part 4 - Compressive sensing - what it is good for?

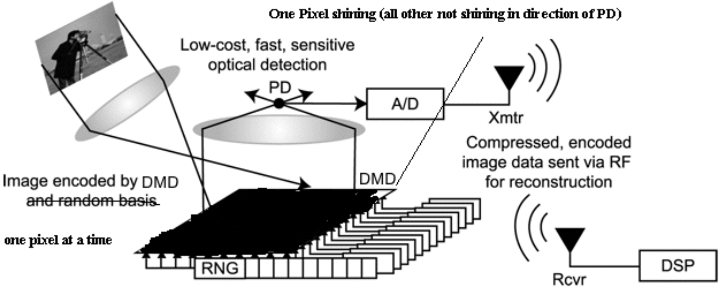
How to spot a compressive sensing system?

Case 1

- Think about systems that use a raster mode for sampling then think of physical ways to perform multiplexing instead
- Once you perform the multiplexing, use compressive sensing solvers to reconstruct signal
- Does it work better or as well with fewer measurements ?

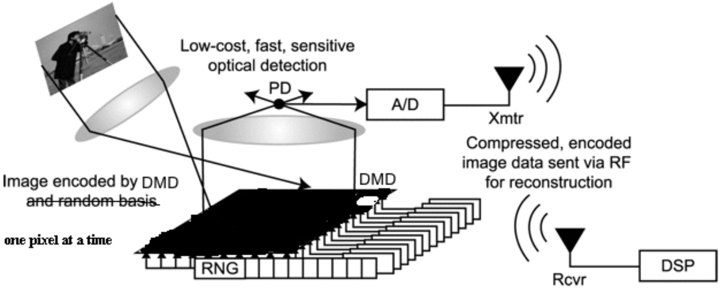
Example: single pixel camera

Classical i=1



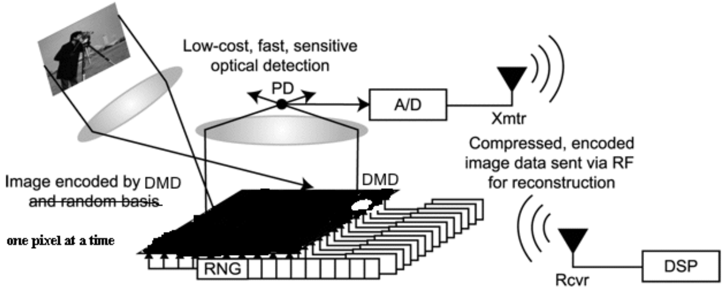
Example: single pixel camera

Classical i=2



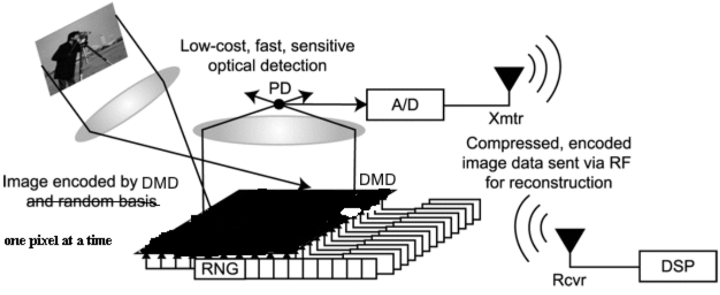
Example: single pixel camera

Classical i=3



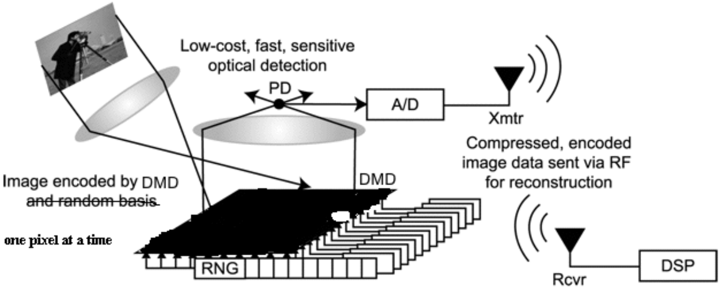
Example: single pixel camera

Classical i=4



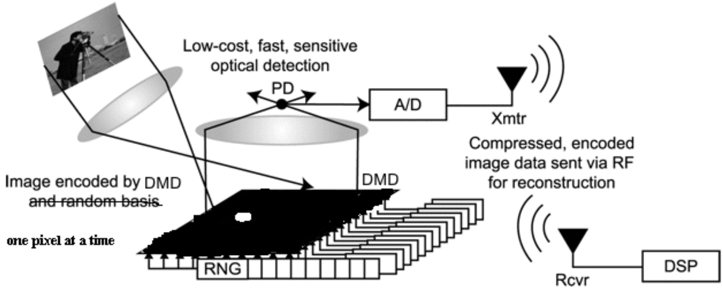
Example: single pixel camera

Classical i=5



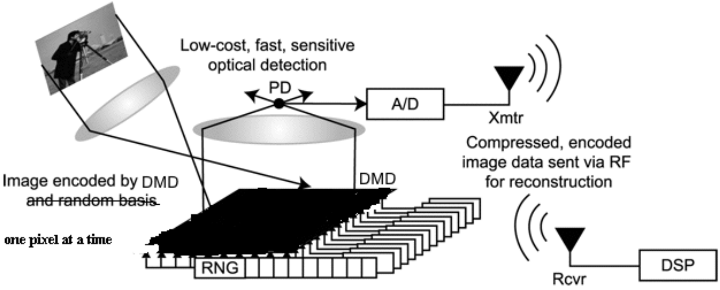
Example: single pixel camera

Classical i=1000



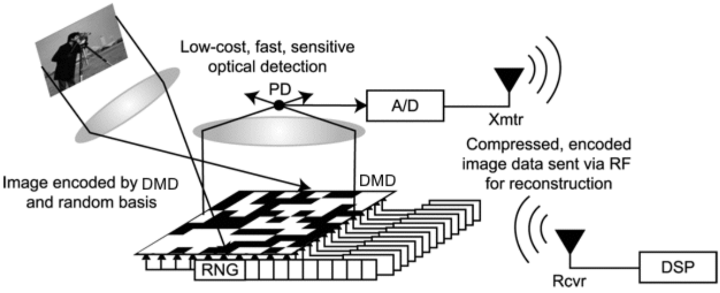
Example: single pixel camera

Classical i=10000000



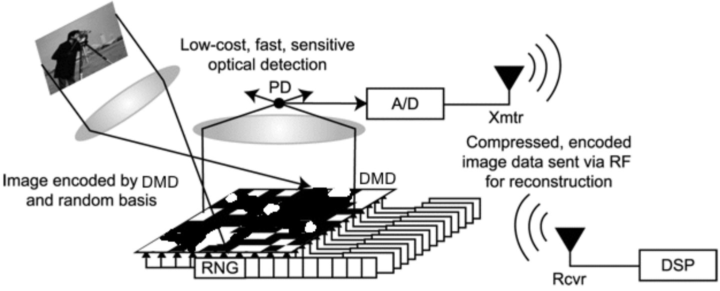
Example: single pixel camera

Compressive sensing $i=1$



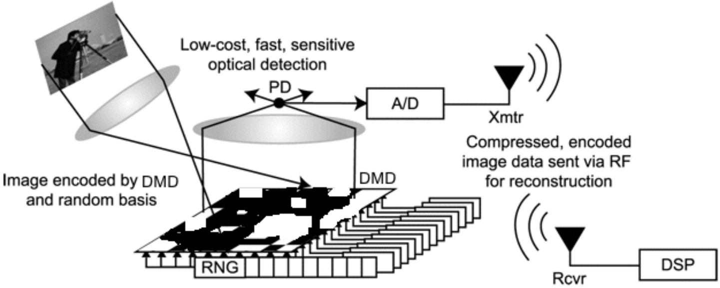
Example: single pixel camera

Compressive sensing i=2



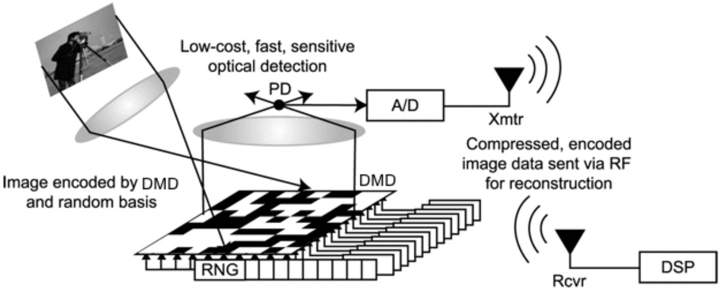
Example: single pixel camera

Compressive sensing i=3



Example: single pixel camera

Compressive sensing



if image is 3-sparse, the sufficient number of measurements scales with 3 and not the size of the image!!!!

How to spot a compressive sensing system?

Case 2

- Look for acquisition schemes that multiplexes a signal already
- Is the signal produced by this system sparse in some basis?
- If yes, subsample the acquisition,
  use compressive sensing solvers to reconstruct signal
- Does it work better or as well with fewer measurements ?

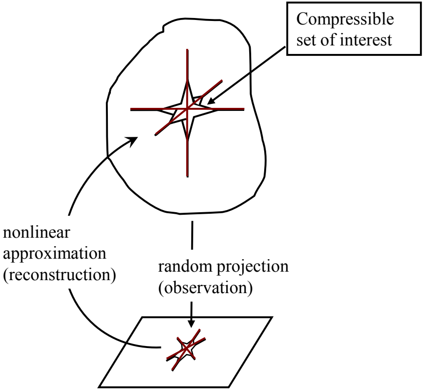
Part 5 - Compressive sensing - summary

Compressive sensing overview

Observe $x \in \mathbb{R}^n$ via m measurements, with $m \ll n$
More precisely, $y = Mx$ where $y \in \mathbb{R}^m$

Assumptions:

- signal approximately s-sparse
- use $m \geq c\ s \log \frac{n}{s}$, c=constant, random linear measurements
- reconstruct by a non linear mapping



Mike Davies.