Discrete calculus, inverse problems and optimisation in imaging

Hugues Talbot

Université Paris-Est – ESIEE

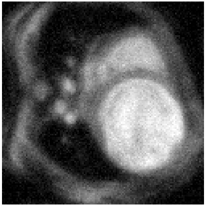
March 26, 2018

Outline of the lecture

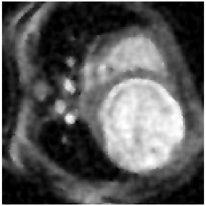
Section 1

Inverse problems in imaging

Motivation: inverse problems in imaging



$\longrightarrow$



- Images we observe are nearly always blurred, noisy, projected versions of some "reality".
- We wish to dispel the fog of acquisition by removing all the artefacts as much as possible to observe the "real" data.
- This is an *inverse* problem.

Maximum Likelihood

- We want to estimate some statistical parameter θ on the basis of some observation x. If f is the sampling distribution, $f(x|\theta)$ is the probability of x when the population parameter is θ. The function

$$\theta \mapsto f(x|\theta)$$

is the *likelihood*. The Maximum Likelihood estimate is

$$\hat{\theta}_{ML}(x) = \underset{\theta}{\operatorname{argmax}}\ f(x|\theta)$$

- E.g, if we have a linear operator H (in matrix form) and Gaussian deviates, then

$$\underset{x}{\operatorname{argmax}}\ f(x) = -\|Hx - y\|_2^2 = -x^\top H^\top Hx + 2y^\top Hx - y^\top y$$

is a quadratic form with a unique maximum, provided by

$$\nabla f(x) = -2H^\top Hx + 2H^\top y = 0 \rightarrow \theta = (H^\top H)^{-1} H^\top y$$

Strengths and drawbacks of MLE

- When possible, MLE is fast and effective. Many imaging operators have a MLE interpretation:
  - Gaussian smoothing ;
  - Wiener filtering ;
  - Filtered back projection for tomography ;
  - Principal component analysis . . .
- However these require a very descriptive model (with few degrees of freedom) and a lot of data, typically unsuitable for images because we do not have a suitable model for natural images.
- When we do not have all these hypotheses, sometimes the Bayesian Maximum A Posteriori approach can be used instead.

Maximum A Posteriori

- If we assume that we know a *prior* distribution g over θ, i.e. some *a-priori* information. Following Bayesian statistics, we can treat θ as a random variable and compute the *posterior* distribution of θ:

$$\theta \mapsto f(\theta|x) = \frac{f(x|\theta)g(\theta)}{\int_{\vartheta\in\Theta} f(x|\vartheta)g(\vartheta)d\vartheta}$$

  (i.e. the Bayes theorem).
- Then the Maximum a Posteriori is the estimate

$$\hat{\theta}_{MAP}(x) = \underset{\theta}{\operatorname{argmax}}\ f(\theta|x) = \underset{\theta}{\operatorname{argmax}}\ f(x|\theta)g(\theta)$$

- MAP is a *regularization* of ML.

Markov Random Fields

So far this is statistics theory. What is the link between MAP and imaging ? We need an imaging model.

- A Markov Random Field is a model made of a set of "sites" (a.k.a. pixels) $S = \{s_1, \ldots, s_n\}$, a set of random variables $y = \{y_1, \ldots, y_n\}$ associated with each pixel, and a set of neighbours $\mathcal{N}_{1,\ldots,n}$ at each pixel location.
- $\mathcal{N}_p$ describes the neighborhood at pixel p.
- Obeys the *Markov condition*, i.e.

$$\Pr(y_p | y_{S \setminus p}) = \Pr(y_p | \mathcal{N}_p)$$

I.e.: the probability of a pixel p depends only on its immediate neighbours.

Formulating the MAP of an MRF

Now let us express a MAP formulation for an MRF

- Given a set of observables $\mathbf{x} = \{x_1, \ldots, x_n\}$,
- We derive a MAP

$$\hat{y} = \underset{y_{1\ldots n}}{\operatorname{argmax}} \; \Pr(y_{1\ldots n}|\mathbf{x}) \tag{1}$$

$$= \underset{y_{1\ldots n}}{\operatorname{argmax}} \prod_{n=1}^{n} \Pr(x_n|y_n) \Pr(y_{1\ldots n}) \tag{2}$$

$$= \underset{y_{1\ldots n}}{\operatorname{argmax}} \sum_{n=1}^{n} \log[\Pr(x_n|y_n)] + \log[\Pr(y_{1\ldots n})] \tag{3}$$

$$= \underset{y_{1\ldots n}}{\operatorname{argmin}} \sum_{p=1}^{n} U_p(y_p) + \sum_{u\in\mathcal{N}_p} P_{u,p}(y_u, y_p) \tag{4}$$

(Geman & Geman, PAMI 1984).

Solving the MAP-MRF formulation

- This last sum is an *energy* contains *unary* terms $U_p(y_p)$ and *pairwise* terms $P_{u,p}(y_u, y_p)$.
- We now have an optimization problem. Depending on the expression of the probability functions, can solve it by i: statistical means, e.g. EM, ii: physical analogies, e.g. simulated annealing or iii: via linear/convex optimization techniques.
- With some restrictions, *graph cuts* are able to optimize these energies.

MRF and Graph Cuts

For instance, consider the binary *segmentation* problem. With unary weights the above can be written:

$$\operatorname{argmin} \hat{E}(G) = \sum_{v_i \in V} w_i(V_i) + \lambda \sum_{e_{ij} \in \vec{E}} w_{ij} \delta_{V_i \neq V_j} \tag{5}$$

- V_i is 1 if $v_i \in V_s$ and 0 if $v_i \in V_t$, i.e. it is 1 if pixel i belongs to the partition containing s and 0 otherwise.
- $\delta_{V_i \neq V_j}$ is 1 if the corresponding e_{ij} is on the cut, and 0 otherwise.
- The first sum contains the pairwise terms, and sums the cost of the cut in the image plane. The second sum contains the unary terms, and adds the cost of a pixel to belong to either the partition containing s or the partition containing t.

Illustration

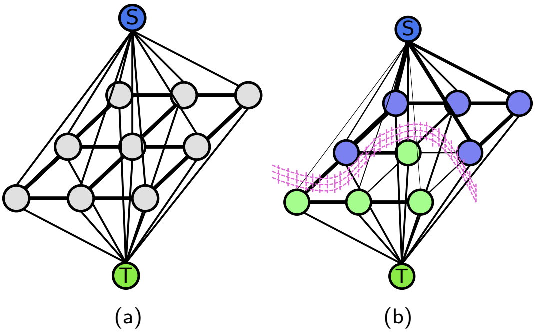
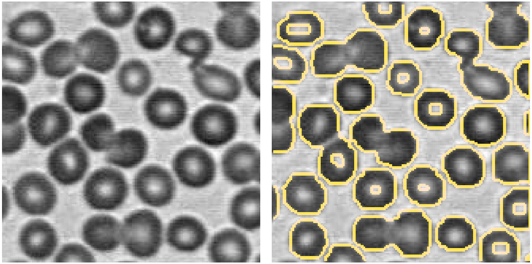


Figure: Segmentation with unary weights. In this case weighted edges link the source and the sink to all the pixels in the image (a). The min-cut is a surface separating s from t (b). Some strong edge weights can ensure the surface crosses the pixel plane, enforcing topology constraints.

Segmentation example



(a) (b)

Figure: Binary segmentation with unary weights and no markers

(Boykov-Jolly segmentation model, ICCV 2001).

Image restoration and graph cuts

- GC are able to optimize some MRF energies exactly (globally) in the binary case
- More generally, *submodular* (e.g. discrete-convex) energies can be at least locally optimized using graph cuts
- Using various constructions, e.g. Ishikawa PAMI 2003, it is possible to map restoration (denoisng) problems to GC.
- Many GC optimization approaches have been invented to solve the corresponding energies: α-expansions, $\alpha - \beta$ moves, convex moves, etc (Veksler 1999). They were essentially known before in other communities (Murota 2003).
- More recent approaches are able to optimize the same kind of energies using different techniques: Belief propagation, Primal-dual Tree-Reweighted, etc (Kolmogorov PAMI 2006).

Graph-based energies

These formulation are very useful but suffer from the purely discrete graph framework

- Formulations and solutions are not isotropic (grid bias)
- Graph based formulation can be resource-intensive (memory and speed)
- They are hard to parallelize
- Hard to incorporate extra constraints and projection/linear operators.

Section 2

Concepts in optimisation

Introduction

- Mathematical optimization is a domain of applied mathematics relevant to many areas including statistics, mechanics, signal and image processing.
- Generalizes many well known techniques such as least squares, linear programming, convex programming, integer programming, combinatorial optimization and others.
- In this talk we will overview both the continuous and discrete formulations.
- We follow the notations of Boyd & Vandeberghe [**?**].

General form

Cost function and constraints

An optimization problem generally has the following form

$$\begin{aligned} \text{minimize } & f_0(x) \\ \text{subject to } & f_i(x) \leq b_i, i = 1, \ldots, m \end{aligned} \quad (6)$$

$x = (x_1, \ldots, x_n)$ is a vector of $\mathbb{R}^n$ called the *optimization variable* of the problem; $f_0 : \mathbb{R}^n \to \mathbb{R}$ is the *cost function* functional; the $f_i : \mathbb{R}^n \to \mathbb{R}$ are the *constraints* and the b_i are the *bounds* (or limits).

A vector $x^\star$ is is *optimal*, or is a solution to the problem, if it has the smallest objective value among all vectors that satisfy the constraints.

Types of optimization problems

- The type of the variables, the cost function and the constraints determine the type of problems we are dealing with.
- Optimization problems, in their most general form, are usually unsolvable in practice. NP-complete problems (traveling salesperson, subset-sum, etc) can classically be put in this form and so can many NP-hard problems.
- Some mathematical regularity is necessary to be able to find a solution: for example, linearity or convexity in all the functions.
- Requiring integer solutions usually, but not always, makes things much harder: Diophantine vs linear equations for instance.

Resolution of optimisation problems

The resolution of an optimisation problem depends on its form. In order of complexity, we can solve optimisation problems:

- In closed form solution (some regression problems)
- If convex: by some iterative descent-like method, yielding a global optimum. Note: may work in the non-differentiable case.
- If non-convex, but regular in some other way (differentiable, quasi-convex, ...): iterative descent-like, converging to a local optimum (or a critical point).
- If combinatorial, usually NP-hard, some exceptions: transport problems (graph cuts, transshipment problems).
- If all else fails: brute force, meta-heuristics.

Example closed form: least-squares

Least squares with no constraints

$$\text{minimize } f_0(x) = \|Ax - b\|_2^2 = \sum_{i=1}^{k} a^\intercal x_i - b_i \tag{7}$$

The system is quadratic, so convex and differentiable. The solution to (7) is unique and reduces to the linear equation

$$(A^\intercal A)x = A^\intercal b. \quad \text{(normal equation)} \tag{8}$$

The analytical solution is $x = (A^\intercal A)^{-1} A^\intercal b$, however $A^\intercal A$ should never be calculated, much less the inverse, for numerical reasons.

Regularization: Tikhonov

Even with something as simple as least-squares, if A is ill-conditioned, the solution will be very sensitive to noise, e.g. in the example of deconvolution or tomography. One solution is to use regularization.

Ill-posed least-squares problems

The simplest regularization strategy is due to Tikhonov [**?**].

$$\text{minimize } f_0(x) = \|Ax - b\|_2^2 + \|\Gamma x\|_2^2, \tag{9}$$

where Γ is a well-chosen operator, e.g. λI or ∇x or a wavelet operator. The solution is given analytically by

$$x = (A^\intercal A + \Gamma^\intercal \Gamma)^{-1} A^\intercal b \tag{10}$$

Example iterative: linear programming

Linear programming with constraints

$$\begin{aligned} &\text{minimize } c^\intercal x \\ &\text{subject to } a_i^\intercal x \leq b_i; i = 1, \ldots, n \end{aligned} \tag{11}$$

- No analytical solution.
- Well established family of algorithms: the Simplexe (Dantzig 1948) ; interior-point (Karmarkar 1984)
- Not always easy to recognize. Important for compressive sensing.

Duality in the LP case

Primal / Dual linear programs

Primal

$$\begin{aligned} &\text{minimize } c^\intercal x \\ &\text{subject to } a_i^\intercal x \leq b_i; i = 1, \ldots, n \end{aligned} \tag{12}$$

Dual

$$\begin{aligned} &\text{maximize } b^\intercal y \\ &\text{subject to } a_i x \geq c_i; i = 1, \ldots, m \end{aligned} \tag{13}$$

- A primal/dual pair of LP problems can be obtained by transposing the constraint matrix and swapping cost function and constraint bounds.
- The primal and dual optima, if they exist, are the same, and can be easily deducted from each other.

Duality in convex optimization

- The same concept of duality applies in convex optimization
- Duality allows one to swap constraints for terms in the objective function
- Two concepts of duality : Lagrange and Fenchel. Both are equivalent.

Lagrange duality

Primal form

$$\begin{aligned} \text{min. } & f_0(x) \\ \text{subject to } & f_i(x) \leq 0, i \in [1, m] \\ & h_i(x) = 0, i \in [1, p] \end{aligned} \tag{14}$$

Dual form

$$\begin{aligned} \text{max. } & g(\lambda, \nu) = \inf_{x \in \mathcal{D}} L_{x,\lambda,\nu} = \left(f_0(x) + \sum_{i=1}^{m} \lambda_i f_i(x) + \sum_{i=1}^{p} \nu_i h_i(x) \right) \\ \text{subject to } & \lambda \geq 0 \end{aligned} \tag{15}$$

Notes on Lagrange duality

- $g(\lambda, \nu)$ is always concave ;
- if p^* is an optimal solution for (14), then $\forall \lambda \geq 0, \forall \nu, g(\lambda, \nu) \leq p^*$
- if d^* is the optimal solution for (15), then $d^* \leq p^*$ (weak duality)
- if (14) is convex, then $d^* = p^*$ (strong duality). (Note: this means the h_i are linear). The reverse is not true.
- Various interesting interpretations, in particular saddle-point (min-max) optimisation, leading to efficient algorithms.
- Complementary slackness ;
- KKT conditions.

Fenchel conjugate

Definition

Let $f : \mathbb{R}^n \to \mathbb{R}$, the function $f^* : \mathbb{R}^n \to \mathbb{R}$ is defined as:

$$f^*(y) = \inf_{x \in \operatorname{dom} f} y^\intercal x - f(x) \tag{16}$$

is the *conjugate* of f. It is always convex.

Example

If $\|.\|$ is a norm on $\mathbb{R}^n$ and its dual norm $\|.\|_*$, the conjugate of $f(x) = \|x\|$ is

$$f^*(y) = \begin{cases} 0 & \|y\|_* \leq 1 \\ \infty & \text{otherwise} \end{cases}, \tag{17}$$

i.e. $f^*(y) = \iota_{\|y\|_* \leq 1}$.

Link between Lagrange duality and Fenchel conjugate

Unconstrained problem

$$\text{minimize } f_0(Ax+b). \tag{18}$$

Its Lagrangian dual is the constant p^*, not very interesting or useful.

Related problem

$$\begin{aligned} \text{minimize } & f_0(y) \\ \text{subject to } & Ax+b=y, \end{aligned} \tag{19}$$

its dual is

$$\begin{aligned} \text{maximize } & b^\intercal \nu - f_0^*(\nu) \\ \text{subject to } & A^\intercal \nu = 0 \end{aligned} \tag{20}$$

Algorithms

Problem

Minimize the function $f \in \Gamma_0(\mathbb{R}^n)on\mathbb{R}^n$

- if f has a β-Lipschitz gradient with $\beta \in]0, +\infty[$,

$$\forall l \in \mathbb{N}, x_{l+1} = x_l + \gamma_l \nabla f(x_l), (\text{ Explicit step }) \tag{21}$$

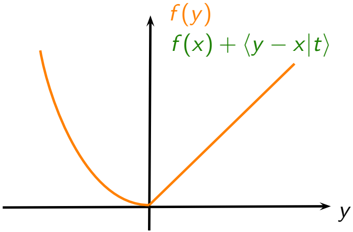
  with $0 < \inf_{l \in \mathbb{N}} \gamma_l$ and $\sup_{l \in \mathbb{N}} \gamma_l < 2\beta^{-1}$.

- If f is not differentiable, replace the gradient with the *subgradient*

$$\partial f = \{t \in \mathbb{R}^n, \forall y \in \mathbb{R}^n, f(y) \geq f(x) + t^\intercal (y - x)\} \tag{22}$$

  $t \in \partial f(x)$: subgradient at $x \in \mathbb{R}^n$, $\partial f : \mathbb{R}^n \rightarrow 2^{\mathbb{R}^n}$.

Illustration subgradient



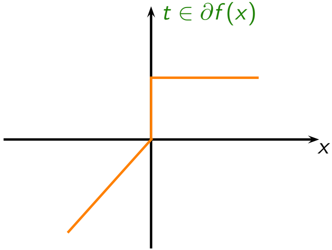
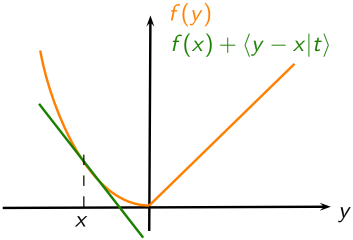


Illustration subgradient



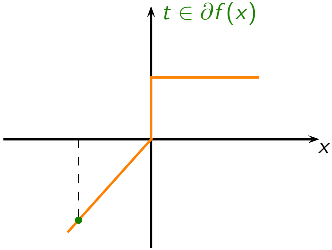


Illustration subgradient

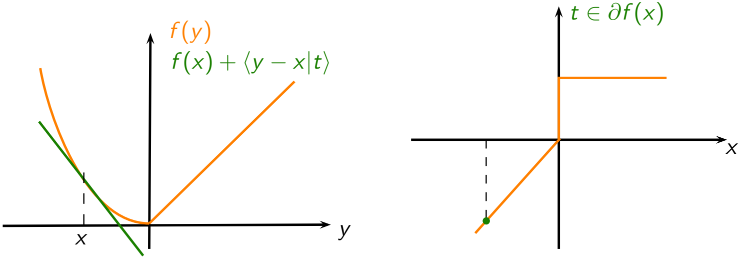


Illustration subgradient

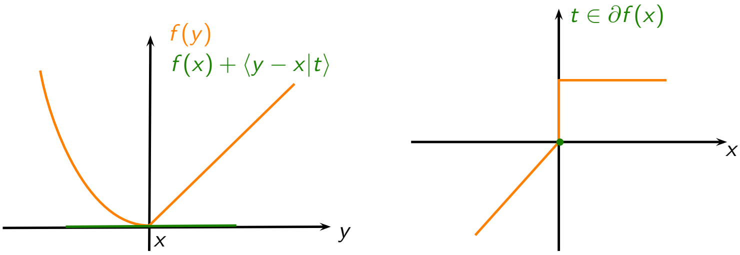
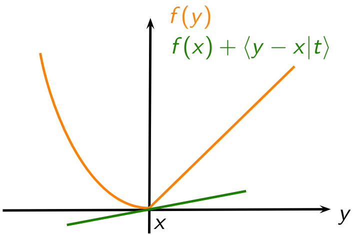


Illustration subgradient



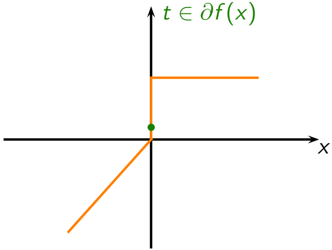


Illustration subgradient

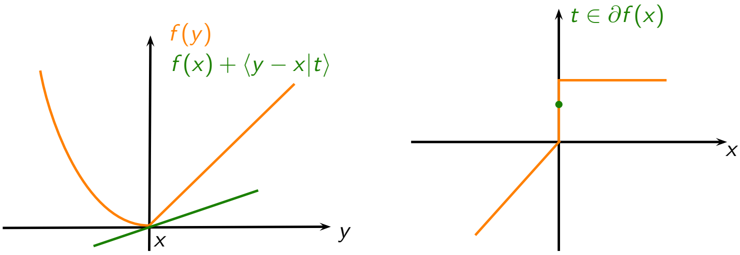


Illustration subgradient

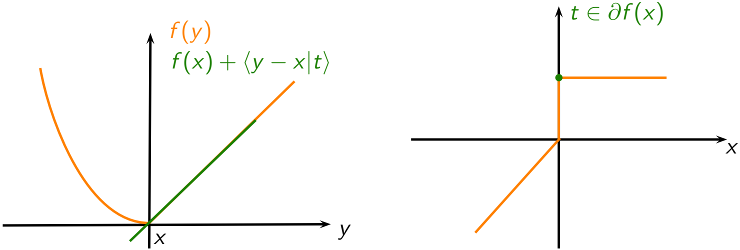
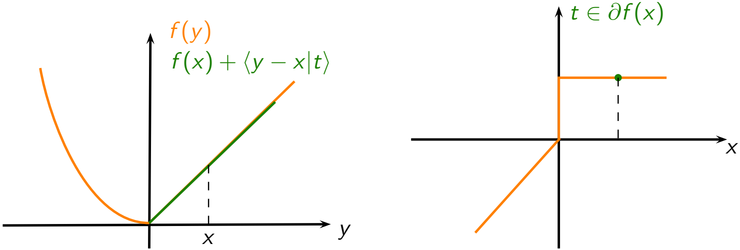


Illustration subgradient



Examples of subgradients

- if f is differentiable at $x \in \mathbb{R}^n$, then $\partial f(x) = \{\nabla f(x)\}$
- if $f = |.|$, then

$$\forall x \in \mathbb{R}, \partial f(x) = \begin{cases} \{\mathrm{sign}(x)\} & \text{if } x \neq 0 \\ \text{[-1,+1]} & \text{if } x = 0 \end{cases} \tag{23}$$

Subgradient algorithm [Shor, 1979]

Explicit form

$$\forall l \in \mathbb{N}, x_{l+1} = x_l - \gamma_l t_l; t_l \in \partial f(x_l), \tag{24}$$

where $(\forall l \in \mathbb{N}), \gamma_l \in]0, +\infty[, \sum_0^{+\infty} \gamma_l^2 < +\infty$ and $\sum_0^{+\infty} \gamma_l = +\infty$.

Implicit form

$$\begin{aligned} \forall l \in \mathbb{N}, & x_{l+1} = x_l - \gamma_l t_l', t_l' \in \partial f(x_{l+1}) \\ \Leftrightarrow & x_l - x_{l+1} \in \gamma_l \partial f(x_{l+1}) \end{aligned} \tag{25}$$

Origins of the proximity operator

Property

Let $\phi \in \Gamma_0(\mathbb{R}^n), \forall x \in \mathbb{R}^n$, there exists a unique vector $\hat{x} \in \mathbb{R}^n$ such that $x - \hat{x} \in \partial\phi(\hat{x})$

- let $\hat{x} = \mathrm{prox}_\phi(x)$
- $\mathrm{prox}_\phi(x) : \mathbb{R}^n \to \mathbb{R}^n$: proximity operator.

Proximal point algorithm

$$\begin{aligned} &\forall l \in \mathbb{N},\ x_l - x_{l+1} \in \gamma_l \partial f(x_{l+1}) \\ &\Leftrightarrow x_{l+1} = \mathrm{prox}_{\gamma_l} f(x_l) \end{aligned} \tag{26}$$

Alternate definition of the prox

Property

Let $f \in \Gamma_0(\mathbb{R}^n)$. For all $x \in \mathbb{R}^n$, $\mathrm{prox}_f(x)$ is the only minimizer of

$$y \mapsto f(y) + \frac{1}{2}\|x - y\|_2^2. \tag{27}$$

The definitions are equivalent

$$\begin{aligned}
\mathrm{prox}_f(x) &= \underset{y}{\mathrm{argmin}} f(y) + \frac{1}{2}\|x - y\|_2^2 \\
&\Leftrightarrow 0 \in \partial\{f(y) + \frac{1}{2}\|x - y\|_2^2\|\} \\
&\Leftrightarrow 0 \in \partial f(y) - x + y \\
&\Leftrightarrow \exists\ \hat{x}, x - \hat{x} \in \partial f(\hat{x})
\end{aligned} \tag{28}$$

Examples of prox

- if $f(x) = |x|, \mathrm{prox}_f(x) = \begin{cases} x+1 & x \leq -1 \\ 0 & x \in [-1,+1] \\ x-1 & x \geq 1 \end{cases}$

  This is soft-thresholding, very popular in wavelet analysis, also see Lasso algorithm in statistics.
- if $f = \iota(\chi)$, χ convex set, and ι the indicator function

  $\iota_\chi(x) = \begin{cases} 0 \ \forall x \in \chi, \\ +\infty \text{ otherwise} \end{cases}$ $\mathrm{prox}_f(x) =$ projection onto convex set χ.

Forward-backward algorithm

Optimisation problem

We seek to minimize the functional $f+g$ on $\mathbb{R}^n$, assuming that g has a β-Lipschitz gradient.

Forward-backward algorithm

$$\forall \ell \in \mathbb{N},\ x_{\ell+1} = x_\ell - \gamma_\ell(t'_\ell + \nabla g(x_\ell)), t'_\ell \in \partial f(x_{\ell+1}) \tag{29}$$

$$\Leftrightarrow x_{\ell+1} = \mathrm{prox}_{\gamma_\ell f}(x_\ell - \gamma_\ell \nabla g(x_\ell)) \tag{30}$$

Section 3

Formulations in imaging

Continuous image restoration model

- We suppose there exists some unknown image $\overline{\boldsymbol{x}} \in \mathbb{R}^N$.
- However we do observe some data $\boldsymbol{y} \in \mathbb{R}^Q$ via some linear operator H, which is corrupted by some noise:

$$\boldsymbol{y} = \boldsymbol{H}\overline{\boldsymbol{x}} + \boldsymbol{u}, \qquad \boldsymbol{H} \in \mathbb{R}^{Q \times N}$$

$\overline{\boldsymbol{x}}$

$\boldsymbol{y}$

Recovery

- We seek to recover a good approximation $\hat{\boldsymbol{x}}$ of $\overline{\boldsymbol{x}}$ from $\boldsymbol{H}$ and $\boldsymbol{y}$.
- H can be:
  - Model for camera, including defocus and motion blur
  - MRI, PET,
  - X-Ray tomography
  - ...
- $\boldsymbol{u}$ often modeled by Additive White Gaussian Noise, but can be Poisson, Poisson Gauss, Rician, etc.

Simplest case: least squares:

$$\hat{\boldsymbol{x}} = \operatorname{argmin}_{\boldsymbol{x}} \|\boldsymbol{H}\boldsymbol{x} - \boldsymbol{y}\|_2^2$$

analytical, simple, effective, but not robust to outliers.

Recovery

- We seek to recover a good approximation $\hat{\boldsymbol{x}}$ of $\overline{\boldsymbol{x}}$ from $\boldsymbol{H}$ and $\boldsymbol{y}$.
- H can be:
  - Model for camera, including defocus and motion blur
  - MRI, PET,
  - X-Ray tomography
  - ...
- $\boldsymbol{u}$ often modeled by Additive White Gaussian Noise, but can be Poisson, Poisson Gauss, Rician, etc.

Tikhonov regularization:

$$\hat{\boldsymbol{x}} = \mathrm{argmin}_{\boldsymbol{x}} \|\boldsymbol{x}\|_2^2 + \lambda \|\boldsymbol{H}\boldsymbol{x} - \boldsymbol{y}\|_2^2$$

reflect the *prior* assumption that we want to avoid large $\boldsymbol{x}$. Also analytical and more robust but not sparse.

Recovery

- We seek to recover a good approximation $\hat{\boldsymbol{x}}$ of $\overline{\boldsymbol{x}}$ from $\boldsymbol{H}$ and $\boldsymbol{y}$.
- H can be:
  - Model for camera, including defocus and motion blur
  - MRI, PET,
  - X-Ray tomography
  - ...
- $\boldsymbol{u}$ often modeled by Additive White Gaussian Noise, but can be Poisson, Poisson Gauss, Rician, etc.

Enforced sparsity:

$$\hat{\boldsymbol{x}} = \operatorname{argmin}_{\boldsymbol{x}} \|\boldsymbol{x}\|_0 + \lambda \|\boldsymbol{H}\boldsymbol{x} - \boldsymbol{y}\|_2$$

If we know $\boldsymbol{x}$ to be sparse (many zero elements) in some space (e.g. Wavelets). Highly non-convex.

Recovery

- We seek to recover a good approximation $\hat{\boldsymbol{x}}$ of $\overline{\boldsymbol{x}}$ from $\boldsymbol{H}$ and $\boldsymbol{y}$.
- H can be:
  - Model for camera, including defocus and motion blur
  - MRI, PET,
  - X-Ray tomography
  - ...
- $\boldsymbol{u}$ often modeled by Additive White Gaussian Noise, but can be Poisson, Poisson Gauss, Rician, etc.

Compressive sensing:

$$\hat{\boldsymbol{x}} = \operatorname{argmin}_{\boldsymbol{x}} \|\boldsymbol{x}\|_1 + \lambda \|\boldsymbol{H}\boldsymbol{x} - \boldsymbol{y}\|_2$$

If we know $\boldsymbol{x}$ to be sparse (many zero elements) in some space (e.g. Wavelets). Smallest convex approximation of the ℓ_0 pseudo-norm.

Formal context

Penalized optimization problem

Find

$$\min_{\boldsymbol{x}\in\mathbb{R}^N} \big(F(\boldsymbol{x}) = \Phi(\boldsymbol{H}\boldsymbol{x}-\boldsymbol{y}) + \lambda R(\boldsymbol{x})\big),$$

Φ $\rightsquigarrow$ Fidelity to data term, related to noise

R $\rightsquigarrow$ Regularization term, related to some *a priori* assumptions

λ $\rightsquigarrow$ Regularization weight

Here, $\boldsymbol{x}$ is **sparse** in a dictionary $\mathcal{V}$ of analysis vectors in $\mathbb{R}^N$

$$\boxed{F_0(\boldsymbol{x}) = \Phi(\boldsymbol{H}\boldsymbol{x}-\boldsymbol{y}) + \lambda\,\ell_0(\boldsymbol{V}\boldsymbol{x})}$$

Formal context

Penalized optimization problem

Find

$$\min_{\boldsymbol{x}\in\mathbb{R}^N} \big(F(\boldsymbol{x}) = \Phi(\boldsymbol{H}\boldsymbol{x}-\boldsymbol{y}) + \lambda R(\boldsymbol{x})\big),$$

- Φ $\rightsquigarrow$ Fidelity to data term, related to noise
- R $\rightsquigarrow$ Regularization term, related to some *a priori* assumptions
- λ $\rightsquigarrow$ Regularization weight

Here, $\boldsymbol{x}$ is **sparse** in a dictionary $\mathcal{V}$ of analysis vectors in $\mathbb{R}^N$

$$\boxed{F_\delta(\boldsymbol{x}) = \Phi(\boldsymbol{H}\boldsymbol{x}-\boldsymbol{y}) + \lambda \sum_{c=1}^{C} \psi_\delta(\boldsymbol{V}_c^\top \boldsymbol{x})}$$

where ψ_δ is a differentiable, non-convex approximation of the ℓ_0 norm.

Benefits and drawbacks of the continuous approach

- pros
  - flexible theory (not just denoising; deblurring, tomography, MRI reconstruction, etc)
  - large library of algorithms, many more than in the discrete case
  - isotropic
  - convergence proofs and characterization of solutions.
- cons
  - non-explicit discretization
  - non-flexible structure
  - deriving projections operators sometimes inefficient or impossible
  - conditions for convergence.

Discrete and continuous approaches

Both the previous discrete and continuous formulation have a MAP interpretation.

- Total Variation (TV) minimization: good regularization tool
- Weighted TV : penalization of the gradient leading to improved results

Our contribution

- General combinatorial formulation of the dual TV problem : easily suitable to various graphs
- Generic constraint in the dual problem : more flexible penalization of the gradient $\rightarrow$ sharper results

Outline

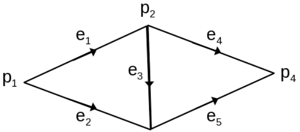
1. Generalization of TV models
2. Parallel Proximal Algorithm as an efficient solver
3. Results

Section 4

Discrete calculus

Discrete formulation on graphs - notations

Graph of N vertices, M edges



Incidence matrix $A \in \mathbb{R}^{M \times N}$

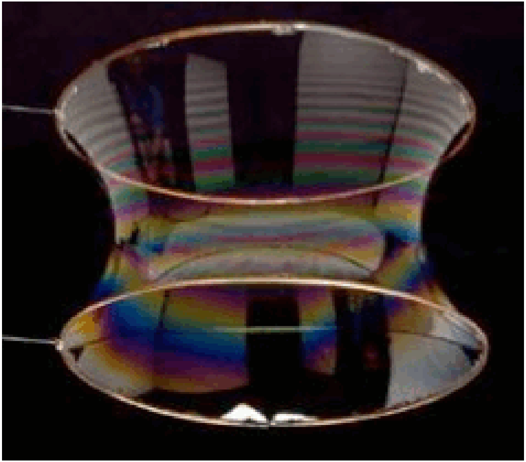
$$A = \begin{array}{c|cccc|} & p_1 & p_2 & p_3 & p_4 \\ \hline e_1 & -1 & 1 & 0 & 0 \\ e_2 & -1 & 0 & 1 & 0 \\ e_3 & 0 & -1 & 1 & 0 \\ e_4 & 0 & -1 & 0 & 1 \\ e_4 & 0 & 0 & -1 & 1 \end{array}$$

- A gradient operator
- $A^\top$ divergence operator
- allows general formulation of problems on arbitrary graphs

For more details: L. Grady and J.R. Polimeni,

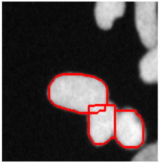
"Discrete Calculus: Applied Analysis on Graphs for Computational Science", Springer, 2010.

Minimal surfaces

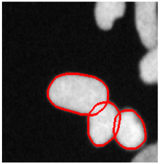


Motivation

- In the continuum: Minimal cut (surface in 3D) is dual of continuous maximum flow [Strang 1983]
- In the classic discrete case min-cut (= "Graph cuts")/ max flow duality but grid bias in the solution
- Recent trend: employ a spatially *continuous* maximum flow to produce solutions with no grid bias



Max Flow (Graph Cuts)



Continuous Max Flow [Appleton-Talbot 2006]

Motivation

- [Appleton-Talbot 2006, generalized by Unger-Pock-Bishof 2008] Fastest known continuous max-flow algorithm has **no stopping criteria** and **no converge proof**.

Our contribution: Combinatorial Continuous Maximum Flow

- a new discrete isotropic formulation
- **avoids blockiness artifacts**
- is **proved to converge**, is **fast**
- **generalizes to arbitrary graphs**

[In SIAM Journal on Imaging Sciences, 2011]

Combinatorial Continuous Maximum Flow (CCMF)

- Incidence matrix of a graph noted A

Continuous MaxFlow	Combinatorial formulation	MaxFlow, GraphCuts
$\max_{\vec{F}} \quad \vec{F}_{st}$ s.t. $\nabla \cdot \vec{F} = 0,\ \|\|\vec{F}\|\| \leq g.$	$\max_{F} \quad F_{st}$ s.t. $A^T F = 0,\ \|A^T\|F^2 \leq g^2$	$\max_{F} \quad F_{st}$ s.t. $A^T F = 0,\ \|F\| \leq g$
	g defined on nodes	g defined on edges

- CCMF : convex problem
- Resolution by an interior point method.

Combinatorial Continuous Maximum Flow (CCMF)

- Incidence matrix of a graph noted A

Continuous MaxFlow

$$\max_{\vec{F}} \quad \vec{F}_{st}$$
$$\text{s.t.} \quad \nabla \cdot \vec{F} = 0,$$
$$||\vec{F}|| \leq g.$$

Combinatorial formulation

$$\max_{F} \quad F_{st}$$
$$\text{s.t.} \quad A^T F = 0,$$
$$|A^T| F^2 \leq g^2$$

g defined on nodes

MaxFlow, GraphCuts

$$\max_{F} \quad F_{st}$$
$$\text{s.t.} \quad A^T F = 0,$$
$$|F| \leq g$$

g defined on edges

- CCMF : convex problem
- Resolution by an interior point method.

Combinatorial Continuous Maximum Flow (CCMF)

- Incidence matrix of a graph noted A

Continuous MaxFlow

$$\max_{\overrightarrow{F}} \quad F_{st}$$
$$\text{s.t.} \quad \nabla \cdot \overrightarrow{F} = 0,$$
$$||\overrightarrow{F}|| \leq g.$$

Combinatorial formulation

$$\max_{F} \quad F_{st}$$
$$\text{s.t.} \quad A^T F = 0,$$
$$|A^T| F^2 \leq g^2$$

g defined on nodes

MaxFlow, GraphCuts

$$\max_{F} \quad F_{st}$$
$$\text{s.t.} \quad A^T F = 0,$$
$$|F| \leq g$$

g defined on edges

- CCMF : convex problem
- Resolution by an interior point method.

Combinatorial Continuous Maximum Flow (CCMF)

- Incidence matrix of a graph noted A

Continuous MaxFlow

$$\max_{\vec{F}} \quad F_{st}$$
$$\text{s.t.} \quad \nabla \cdot \vec{F} = 0, \quad ||\vec{F}|| \leq g.$$

Combinatorial formulation

$$\max_{F} \quad F_{st}$$
$$\text{s.t.} \quad A^T F = 0, \quad |A^T| F^2 \leq g^2$$

g defined on nodes

MaxFlow, GraphCuts

$$\max_{F} \quad F_{st}$$
$$\text{s.t.} \quad A^T F = 0, \quad |F| \leq g$$

g defined on edges

- CCMF : convex problem
- Resolution by an interior point method.

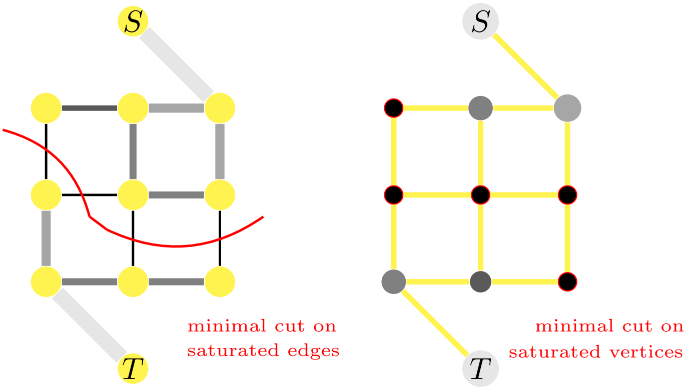
Combinatorial Continuous Maximum Flow (CCMF)

- Incidence matrix of a graph noted A

Continuous MaxFlow	Combinatorial formulation	MaxFlow, GraphCuts
$\max_{\vec{F}} \quad F_{st}$ s.t. $\quad \nabla \cdot \vec{F} = 0,$ $\lVert\vec{F}\rVert \leq g.$	$\max_{F} \quad F_{st}$ s.t. $\quad A^T F = 0,$ $\lvert A^T\rvert F^2 \leq g^2$ g defined on nodes	$\max_{F} \quad F_{st}$ s.t. $\quad A^T F = 0,$ $\lvert F\rvert \leq g$ g defined on edges

- CCMF : convex problem
- Resolution by an interior point method.

Graph Cuts vs CCMF



Scale of weight intensity :

CCMF dual problem

- The dual of the CCMF problem is

$$\min_{\lambda\geq 0,\nu} \sum_{v_i\in V} \underbrace{\lambda_i g_i^2}_{\text{weighted cut}} + \underbrace{\frac{1}{4}\sum_{e_{ij}\in E\setminus\{s,t\}} \frac{(\nu_i-\nu_j)^2}{\lambda_i+\lambda_j}}_{\text{smoothness term}} + \underbrace{\frac{1}{4}\frac{(\nu_s-\nu_t-1)^2}{\lambda_s+\lambda_t}}_{\text{source-sink enforcement}}$$

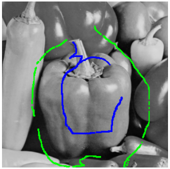
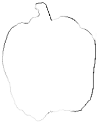


Image
with seeds



λ

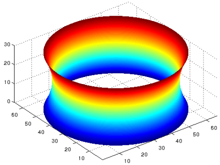
ν

Threshold
of ν at $.5$

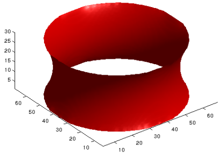
Minimal surfaces

Catenoid test problem:

- source constituted by two full circles
- sink by the remaining boundary of the image, constant metric g



analytic minimal surface

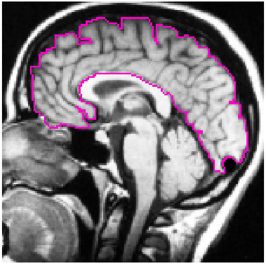


CCMF result isosurface of ν

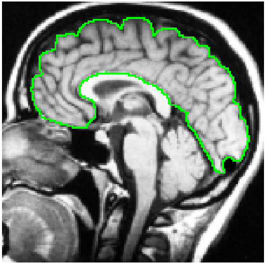
Root Mean Square Error between the surfaces : 0.75

(Appleton-Talbot error : 1.98)

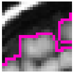
Comparison with Graph cuts



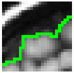
Graph cuts result



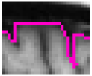
CCMF result



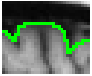
GC



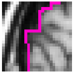
CCMF



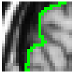
GC



CCMF

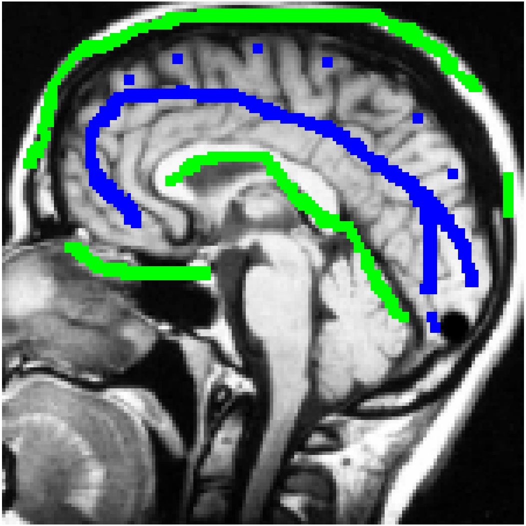


GC

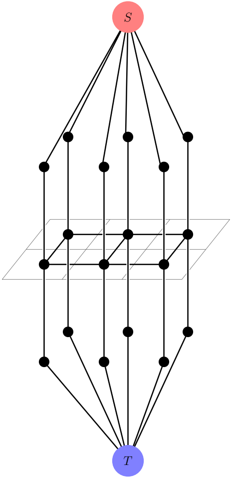


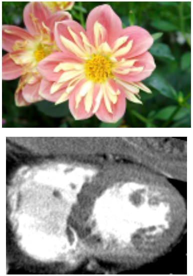
CCMF

Convergence

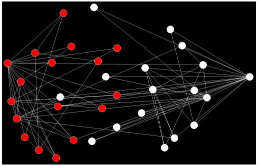


Genericity of the method



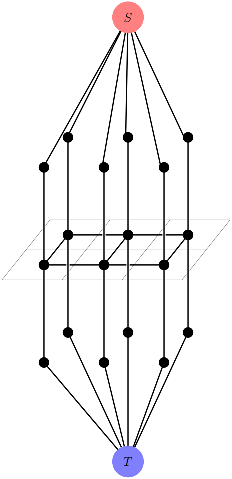


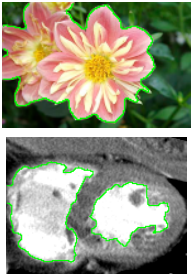
Unseeded segmentation



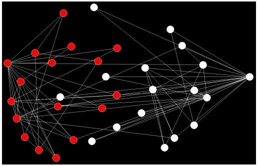
Classification

Genericity of the method





Unseeded segmentation



Classification

Total variation regularization

- Given an original image f
- Deduce a restored image u

Weighted anisotropic TV model [Gilboa and Osher 2007]

$$\min_u \underbrace{\int \left(\int w_{x,y} (u_y - u_x)^2 dy \right)^{1/2} dx}_{\text{regularization} \quad R(u)} + \underbrace{\frac{1}{2\lambda} \int (u_x - f_x)^2 dx}_{\text{data fidelity} \quad \Phi(u)}$$

where

- $\lambda \in]0, +\infty[$ regularization parameter

Equivalent dual formulation

Weighted anisotropic TV model [Gilboa and Osher 2007]

$$\min_u \int \left(\int w_{x,y}(u_y - u_x)^2 dy \right)^{1/2} dx + \Phi(u)$$

is equivalent [Chan, Golub, Mulet 1999] to the min-max problem

$$\min_u \max_{||p||_\infty \leq 1} \int \int w_{x,y}^{1/2}(u_y - u_x) p_{x,y} dx dy + \Phi(u)$$

with p a projection vector field.

Main idea

- p was introduced in practice to compute a faster solution
- constraining p can promote better results

Segmentation

- Same model as denoising, with a labeled fidelity term
- Same regularisation. This includes very widespread models such as watershed, region growing, minimal curves and surfaces, geodesic active contours, and more.

Deblurring, tomography

- Deblurring / tomography simply composes a linear term within the fidelity.
- Same model for regularization as before
- Possible to do very advanced applications: local tomography, angular integration tomography, dual image deblurring, etc.
- Also applicable with wavelets, etc. Any linear operator can serve.

Discrete formulations of TV and its dual

Let $u \in \mathbb{R}^N$ be the restored image.
[Bougleux *et al.* 2007]

$$\min_u \sum_{i=1}^{n} \Big(\sum_{j \in N_i} w_{i,j} (u_j - u_i)^2 \Big)^{1/2} + \Phi(u)$$

where $N_i = \{ j \in \{1, \ldots, n\} \mid e_{i,j} \in E \}$.

We introduce the following combinatorial formulation for the primal dual problem

$$\min_u \max_{\|p\|_\infty \leq 1,\; p \in \mathbb{R}^M} p^\top ((Au) \cdot \sqrt{w}) + \Phi(u)$$

Dual constrained TV based formulation

Constraining the projection vector

- Introducing the projection vector $F \in \mathbb{R}^M = p \cdot \sqrt{w}$
- Constraining F to belong to a convex set C

$$\min_{u \in \mathbb{R}^N} \underbrace{\sup_{F \in C} \quad F^\top (Au)}_{\text{regularization}} + \underbrace{\frac{1}{2\lambda} \|u - f\|_2^2}_{\text{data fidelity}}$$

- $C = \cap_{i=1}^{m-1} C_i \neq \varnothing$ where $C_1, \ldots, C_{m-1}$ closed convex sets of $\mathbb{R}^M$.
- Given $g \in \mathbb{R}^N$, $\theta_i \in \mathbb{R}^M$, $\alpha \geq 1$,
  $C_i = \{F \in \mathbb{R}^M \mid \|\theta_i \cdot F\|_\alpha \leq g_i\}$.

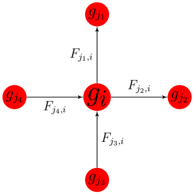
Dual constrained TV based formulation

$$\min_{u\in\mathbb{R}^N} \underbrace{\sup_{F\in C} \quad F^\top(Au)}_{\text{regularization}} + \underbrace{\frac{1}{2\lambda}\|u-f\|_2^2}_{\text{data fidelity}}$$

- $C = \cap_{i=1}^{m-1} C_i, \quad C_i = \{F \in \mathbb{R}^M \mid \|\theta_i \cdot F\|_\alpha \leq g_i\}$, $\alpha \geq 1$.

Example adapted to image denoising

- $g_i \in \mathbb{R}^N$ weight on vertex i, inversely function of the gradient of f at node i.
- Flat area : weak gradient $\rightarrow$ strong g_i $\rightarrow$ strong $F_{i,j}$ $\rightarrow$ weak local variations of u.
- Contours : strong gradient $\rightarrow$ weak g_i $\rightarrow$ weak $F_{i,j}$ $\rightarrow$ large local variations of u allowed.



$$C_i = \{F \in \mathbb{R}^M \mid \sqrt{\sum_{j\in N_i} F_{j,i}^2} \leq g_i$$

Illustration of constraining flow

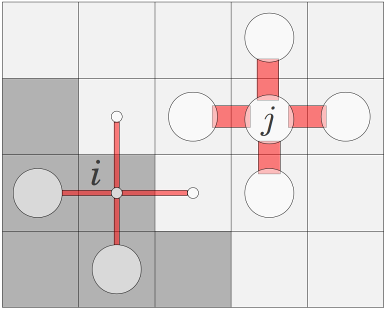
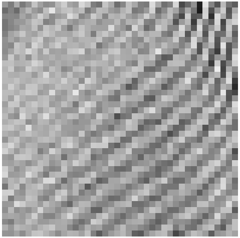
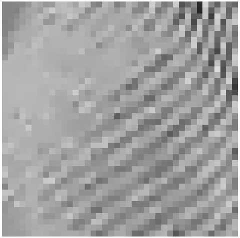


Illustration of constraining flow.

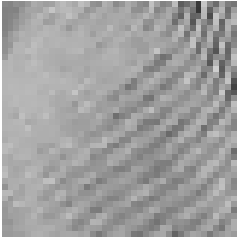
Sharper results



Noisy image



DCTV



Weighted TV

Extension of our DCTV based formulation

$$\min_{u \in \mathbb{R}^N} \underbrace{\sup_{F \in C} \quad F^\top (Au)}_{\text{regularization}} + \underbrace{\frac{1}{2\lambda}\|u - f\|_2^2}_{\text{data fidelity}}$$

- $f \in \mathbb{R}^Q$, observed image
- $u \in \mathbb{R}^N$, restored image
- $F \in \mathbb{R}^M$, dual solution : projection vector

Extension of our DCTV based formulation

$$\min_{u \in \mathbb{R}^N} \underbrace{\sup_{F \in C} \quad F^\top (Au)}_{\text{regularization}} + \underbrace{\frac{1}{2\lambda} \| H u - f \|_2^2}_{\text{data fidelity}}$$

- $f \in \mathbb{R}^Q$, observed image
- $u \in \mathbb{R}^N$, restored image
- $F \in \mathbb{R}^M$, dual solution : projection vector
- $H \in \mathbb{R}^{Q \times N}$, degradation matrix

Extension of our DCTV based formulation

$$\min_{u\in\mathbb{R}^N} \underbrace{\sup_{F\in C} \quad F^\top(Au)}_{\text{regularization}} + \underbrace{\frac{1}{2\lambda}\|Hu - f\|_2^2 + \frac{\eta}{2}\|Ku\|^2}_{\text{data fidelity}}$$

- $f \in \mathbb{R}^Q$, observed image
- $u \in \mathbb{R}^N$, restored image
- $F \in \mathbb{R}^M$, dual solution : projection vector
- $H \in \mathbb{R}^{Q\times N}$, degradation matrix
- $K \in \mathbb{R}^{N\times N}$: projection onto $\mathrm{Ker}\, H$, $\eta \geq 0$

Extension of our DCTV based formulation

$$\min_{u\in\mathbb{R}^N} \underbrace{\sup_{F\in C} \quad F^\top(Au)}_{\text{regularization}} + \underbrace{\frac{1}{2}(Hu-f)^\top \Lambda^{-1}(Hu-f) + \frac{\eta}{2}\|Ku\|^2}_{\text{data fidelity}}$$

- $f \in \mathbb{R}^Q$, observed image
- $u \in \mathbb{R}^N$, restored image
- $F \in \mathbb{R}^M$, dual solution : projection vector

- $H \in \mathbb{R}^{Q\times N}$, degradation matrix
- $K \in \mathbb{R}^{N\times N}$, projection onto $\mathrm{Ker}\, H$, $\eta \geq 0$
- $\Lambda \in \mathbb{R}^{Q\times Q}$, matrix of weights, positive definite

Primal formulation

$$\min_{u\in\mathbb{R}^N} \quad \underbrace{\sigma_C(Au)}_{\text{regularization}} \quad + \underbrace{\frac{1}{2}(Hu-f)^\top\Lambda^{-1}(Hu-f)+\frac{\eta}{2}\|Ku\|^2}_{\text{data fidelity}}$$

- $C=\cap_{i=1}^{m-1}C_i \neq \varnothing$ where $C_1,\dots,C_{m-1}$ closed convex sets of $\mathbb{R}^M$.
- σ_C support function of the convex set C

$$\sigma_C\colon \mathbb{R}^M \to]-\infty,+\infty]\colon a \mapsto \sup_{F\in C} F^\top a.$$

Dual problem

- The problem admits a unique solution $\widehat{u}$.
- Fenchel-Rockafellar dual problem:

$$\min_{F \in \mathbb{R}^M} \quad \sum_{i=1}^{m-1} \underbrace{\iota_{C_i}(F)}_{f_i(F)} + f_m(F)$$

  where ι_C is the indicator function of the convex C (equal to 0 inside C and $+\infty$ outside),
  $f_m \colon F \mapsto \frac{1}{2} F^\top A \Gamma A^\top F - F^\top A \Gamma H^\top \Lambda^{-1} f$,
  and $\Gamma \ = \ (H^\top \Lambda^{-1} H + \eta K)^{-1}$.
- If $\widehat{F}$ is a solution to the dual problem,

$$\widehat{u} = \Gamma \left(H^\top \Lambda^{-1} f - A^\top \widehat{F} \right).$$

Families of algorithms in continuous optimization

- Contour-based algorithms
- Snakes
- Level sets
- Region-based algorithms
- Primal only algorithms
- Primal-dual algorithms

Parallel ProXimal Algorithm (PPXA) for DCTV [?]

$\gamma > 0, \nu \in]0, 2[$.

Repeat until convergence

$$
\left\lfloor
\begin{array}{l}
\text{For (in parallel) } r = 1, \ldots, s+1 \\
\quad \left\lfloor \; \pi_r = \begin{cases} \mathsf{P}_{C_r}(y_r) & \text{if } r \le s \\ (\gamma A \Gamma A^\top + I)^{-1}(\gamma A \Gamma H^\top \Lambda^{-1} f + y_{s+1}) & \text{otherwise} \end{cases} \right. \\
z = \frac{2}{s+1}(\pi_1 + \cdots + \pi_{s+1}) - F \\
\text{For (in parallel) } r = 1, \ldots, s+1 \\
\quad \left\lfloor \; y_r = y_r + \nu(z - p_r) \right. \\
F = F + \frac{\nu}{2}(z - F)
\end{array}
\right.
$$

Parallel ProXimal Algorithm (PPXA) for DCTV [?]

$\gamma > 0, \nu \in]0, 2[$.

Repeat until convergence

$$
\left\lfloor
\begin{array}{l}
\text{For (in parallel) } r = 1, \ldots, s+1 \\
\quad \left\lfloor \pi_r = \begin{cases} \mathsf{P}_{C_r}(y_r) & \text{if } r \leq s \\ (\gamma A \Gamma A^\top + I)^{-1}(\gamma A \Gamma H^\top \Lambda^{-1} f + y_{s+1}) & \text{otherwise} \end{cases} \right. \\
z = \frac{2}{s+1}(\pi_1 + \cdots + \pi_{s+1}) - F \\
\text{For (in parallel) } r = 1, \ldots, s+1 \\
\quad \left\lfloor y_r = y_r + \nu(z - p_r) \right. \\
F = F + \frac{\nu}{2}(z - F)
\end{array}
\right.
$$

- Simple projections onto hyperspheres

Parallel ProXimal Algorithm (PPXA) for DCTV [?]

$\gamma > 0, \nu \in]0, 2[$.

Repeat until convergence

$$
\left\lfloor
\begin{array}{l}
\text{For (in parallel) } r = 1, \ldots, s+1 \\
\quad \left\lfloor \pi_r = \begin{cases} \mathsf{P}_{C_r}(y_r) & \text{if } r \le s \\ (\gamma A \Gamma A^\top + I)^{-1}(\gamma A \Gamma H^\top \Lambda^{-1} f + y_{s+1}) & \text{otherwise} \end{cases} \right. \\
z = \frac{2}{s+1}(\pi_1 + \cdots + \pi_{s+1}) - F \\
\text{For (in parallel) } r = 1, \ldots, s+1 \\
\quad \left\lfloor y_r = y_r + \nu(z - p_r) \right. \\
F = F + \frac{\nu}{2}(z - F)
\end{array}
\right.
$$

- Linear system resolution

Quantitative perfomances

- Speed : competitive with the most efficient algorithm for optimizing weighted TV
- Denoising a 512×512 image
  - with an Alternated Direction of Multiplier Method: 0.4 seconds
  - with the Parallel Proximal Algorithm: 0.7 seconds
- Quantitative denoising experiments on standard images show improvements of SNR (from 0.2 to 0.5 dB) for images corrupted with Gaussian noise of variance σ^2 from 5 to 25.

Results in image denoising

Original image

Noisy SNR=10.1dB

Weighted TV SNR=13.4dB

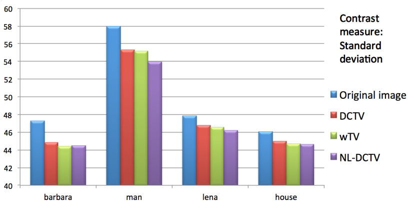
DCTV SNR=13.8dB

Results in image denoising

Weighted TV SNR=13.4dB

DCTV SNR=13.8dB

Comparison with more standard TV



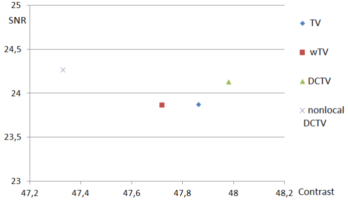
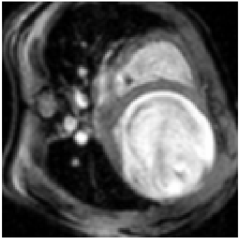
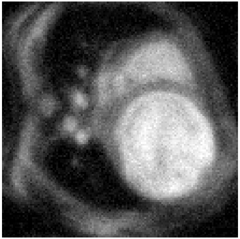


Figure: Left hand side: Standard deviation of each test image compared with the standard deviation of the denoising results, averaged results with $(\sigma^2 = 5, 10, 15, 20, 25, 50)$. Right hand side: mean SNR over the experiments,

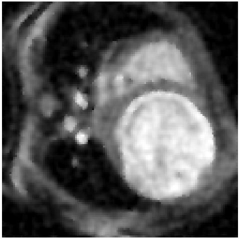
Image denoising and deconvolution



Original image



Noisy, blurred image SNR=12.3dB



DCTV result SNR=17.2dB

Image fusion

Original image

Noisy

SNR=7.2dB

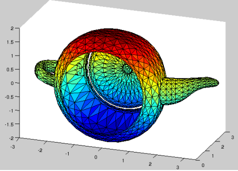
blurry

SNR=11.6dB

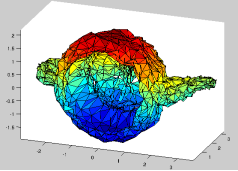
DCTV

SNR=16.3dB

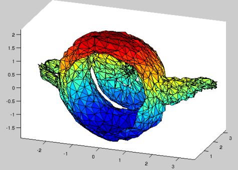
Mesh denoising



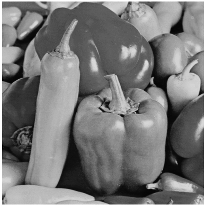
Original
mesh



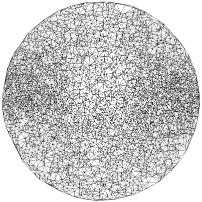
Noisy
mesh



DCTV regularization
on spatial coordinates

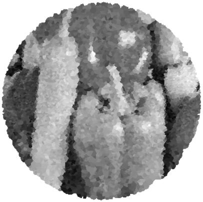


(a) Original image

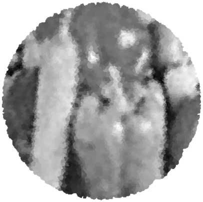


(b) Bottlenosed dolphin structure

(c) Sampled image



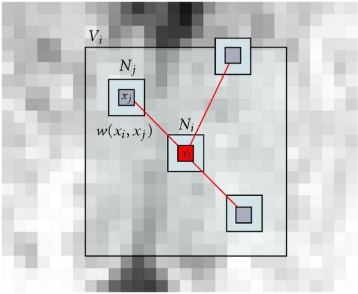
(d) Noisy sampled SNR = 22.1 dB



(e) Taubin filtered result [?] SNR = 19.4 dB

(f) DCTV result ($\lambda = 0.5$) SNR = 23.3 dB

Non-local regularization



(a) Nonlocal graph (figure P. Coupé, [?])

Figure: Example of Non-Local Graph.

Original image

Noisy PSNR=28.1dB

Nonlocal DCTV PSNR=35 dB

Section 5

Non-convex optimisation

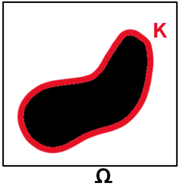
Mumford-Shah functional [?]

We wish to minimize the following energy :

$$\mathcal{MS}(K,u) = \underbrace{\int_{\Omega\backslash K} |u-g|^2 \, \mathrm{d}x}_{\text{fidelity}} + \alpha \underbrace{\int_{\Omega\backslash K} |\nabla u|^2 \, \mathrm{d}x}_{\text{regularization}} + \lambda \underbrace{\mathcal{H}^1(K\cap\Omega)}_{\text{perimeter}}$$

avec :

- Ω the image domaine
- g a given image (e.g. $g \in \mathrm{L}^\infty(\Omega)$)
- u a simplification of g ($u \in \mathrm{H}^1(\Omega\backslash K)$)
- K set of contours



Relaxation

Relaxation in SBV

$$\mathcal{MS}(u) = \alpha \int_\Omega |u-g|^2 \, \mathrm{d}x + \int_\Omega |\nabla u|^2 \, \mathrm{d}x + \lambda \, \mathcal{H}^1(\mathcal{J}_u) \tag{31}$$

Ambrosio-Tortorelli formulation [**?**]

$$\mathrm{AT}_\varepsilon(u,v) = \int_\Omega \alpha|u-g|^2 + v^2|\nabla u|^2 + \lambda\varepsilon|\nabla v|^2 + \frac{\lambda}{4\varepsilon}|1-v|^2 \, \mathrm{d}x$$

if $u, v \in W^{1,2}(\Omega)$ and $0 \leq v \leq 1$.

A bit more Discrete Calculus

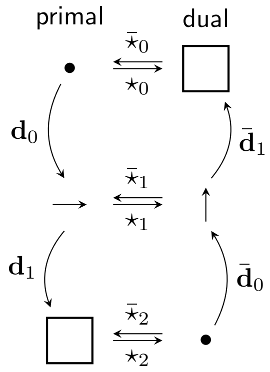


Figure: DEC operators

Formulation in DEC

We define u and g on faces and v on vertices and edges. Fonctions u and g are 2-forms since they represent the gray levels of each pixel.

U2V0

$$\mathrm{AT}_{\epsilon}^{2,0}(u,v) = \alpha\langle u-g, u-g\rangle_2 + \langle \mathbf{M}_{01}v, \bar{\star}\bar{\mathbf{d}}_0\star u\rangle_1^2 + \lambda\varepsilon\langle \mathbf{d}_0 v, \mathbf{d}_0 v\rangle_1 + \frac{\lambda}{4\varepsilon}\langle 1-v, 1-v\rangle_0.$$

U0V1

$$\begin{aligned}\mathrm{AT}_{\epsilon}^{0,1}(u,v) =& \alpha\langle u-g, u-g\rangle_0 + \langle v, \mathbf{d}_0 u\rangle_1 \langle v, \mathbf{d}_0 u\rangle_1 \\ &+ \lambda\epsilon\langle (\mathbf{d}_1 + \bar{\star}\bar{\mathbf{d}}_1\star)v, (\mathbf{d}_1 + \bar{\star}\bar{\mathbf{d}}_1\star)v\rangle_1 \\ &+ \frac{\lambda}{4\epsilon}\langle 1-v, 1-v\rangle_1.\end{aligned}$$

Restoration

Restoration

Non-convex optimization

- The current frontier.
- Many interesting applications thought to be very hard to solve: blind deblurring
- Many current methods extend to the Non-Convex case
- Generally only a local minimum is reached, but this might be OK. The miimum might be of high quality : stochastic optimization.
- For instance: see results achieved by deep-learning methods.

ℓ_2-ℓ_0 regularization functions

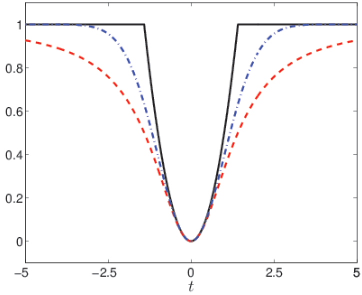
We consider the following class of potential functions:

1. $(\forall \delta \in (0, +\infty))$ ψ_δ is differentiable.
2. $(\forall \delta \in (0, +\infty))$ $\lim_{t \to \infty} \psi_\delta(t) = 1$.
3. $(\forall \delta \in (0, +\infty))$ $\psi_\delta(t) = \mathcal{O}(t^2)$ for small t.

Examples:

– – – $\psi_\delta(t) = \frac{t^2}{2\delta^2 + t^2}$

– · –· $\psi_\delta(t) = 1 - \exp(-\frac{t^2}{2\delta^2})$



Majorize-Minimize principle *[Hunter04]*

Objective: Find $\hat{\boldsymbol{x}} \in \operatorname{Arg\,min}_{\boldsymbol{x}} F_\delta(\boldsymbol{x})$

For all $\boldsymbol{x}'$, let $Q(., \boldsymbol{x}')$ a *tangent majorant* of F_δ at $\boldsymbol{x}'$ i.e.,

$$Q(\boldsymbol{x}, \boldsymbol{x}') \geqslant F_\delta(\boldsymbol{x}), \quad \forall \boldsymbol{x},$$
$$Q(\boldsymbol{x}', \boldsymbol{x}') = F_\delta(\boldsymbol{x}')$$

MM algorithm:

$$\forall j \in \{0, \ldots, J\},$$

$$\boldsymbol{x}^{j+1} \in \operatorname{Arg\,min}_{\boldsymbol{x}} Q(\boldsymbol{x}, \boldsymbol{x}^j)$$

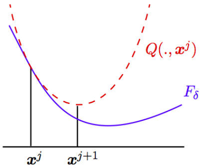
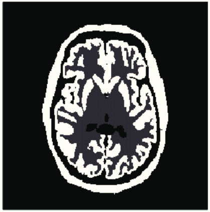
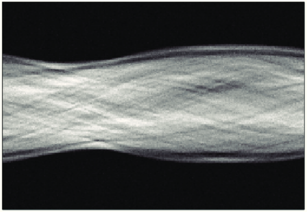


Image reconstruction



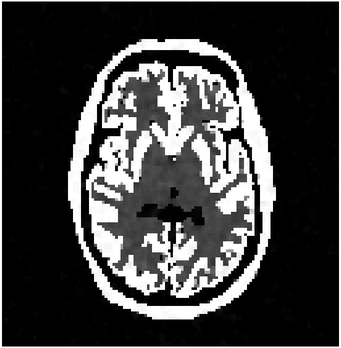
Original image $\overline{\boldsymbol{x}}$
128×128



Noisy sinogram $\boldsymbol{y}$
SNR$=25$ dB

- $\boldsymbol{y} = \boldsymbol{H}\bar{\boldsymbol{x}} + \boldsymbol{u}$ with $\begin{cases} \boldsymbol{H} & \text{Radon projection matrix} \\ \boldsymbol{u} & \text{Gaussian noise} \end{cases}$
- $\hat{\boldsymbol{x}} \in \operatorname{Arg\,min}_{\boldsymbol{x}} \left(\frac{1}{2}\|\boldsymbol{H}\boldsymbol{x} - \boldsymbol{y}\|^2 + \lambda \sum_c \psi_\delta(\boldsymbol{V}_c^\top \boldsymbol{x})\right)$
- Non convex penalty / convex penalty

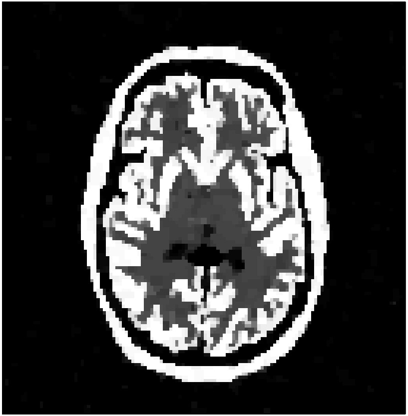
Results: Non convex penalty



Reconstructed image
SNR $= 20.4$ dB

MM-MG algorithm:
Convergence in 134 s

Results: Convex penalty



Reconstructed image
SNR $= 18.4$ dB

MM-MG algorithm:
Convergence in 60 s

Section 6

Conclusion

Conclusion

- Optimization is a very powerful, general methodology
- We've drawn a panorama of interesting methodologies in image processing
  - Extension of TV models via dual formulations
  - Many applications in inverse problems including segmentation
  - Proposed algorithm efficiently solves convex and non-convex problems
  - Application to arbitrary graphs
- Generally optimization problems are unsolvable without some regularity assumptions. There exist a trade-off between the generality of a framework and the efficiency of associated algorithms.
- On to new things: hierarchies of partitions.