Bayesian Discrete Tomography from a few number of projections

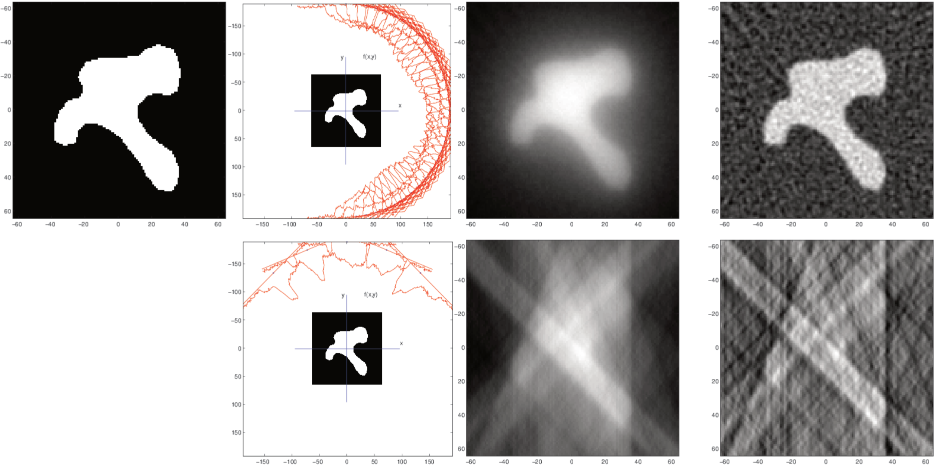
Ali Mohammad-Djafari
Laboratoire des Signaux et Systèmes (L2S)
UMR8506 CNRS-CentraleSupélec-UNIV PARIS SUD
SUPELEC, 91192 Gif-sur-Yvette, France
http://lss.centralesupelec.fr
Email: djafari@lss.supelec.fr
http://djafari.free.fr
http://publicationslist.org/djafari

Invited talk at
Workshop on Discrete Tomography, Polytechnico de Milan, Italy

Contents

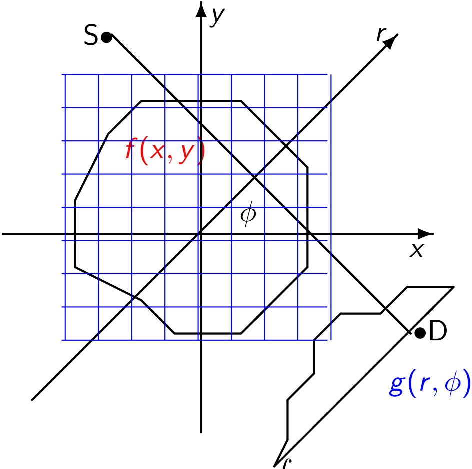
1. Limited angle Tomography
2. Basic Bayesian approach
3. Two main steps:
   - Choosing appropriate Prior model
   - Do the computational efficiently
4. Hierarchical prior modelling
   - Sparsity enforcing models through Student-t and IGSM
   - Gauss-Markov-Potts models
5. Computational tools: JMAP, Gibbs Sampling MCMC, VBA
6. Case study: Image Reconstruction with only two projections
7. Implementation issues
   - Main GPU implementation steps: Forward and Back Projections
   - Multi-Resolution implementation
8. Conclusions

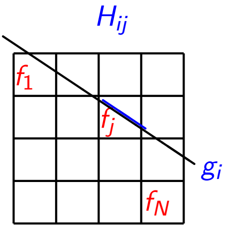
Limited angle Tomography: Limitations of analytical methods



Original   Data   Backprojection   Filtered Backprojection

Algebraic methods: Discretization





$$f(x,y) = \sum_j f_j \, b_j(x,y)$$

$$b_j(x,y) = \begin{cases} 1 & \text{if } (x,y) \in \text{ pixel } j \\ 0 & \text{else} \end{cases}$$

$$g(r,\phi) = \int_L f(x,y) \, \mathrm{d}l$$

$$g_i = \sum_{j=1}^{N} H_{ij} \, f_j + \epsilon_i$$

$$\mathbf{g}_k = \mathbf{H}_k \mathbf{f} + \boldsymbol{\epsilon}_k, k = 1, \cdots, K \longrightarrow \mathbf{g} = \mathbf{H}\mathbf{f} + \boldsymbol{\epsilon}$$

$\mathbf{g}_k$ projection at angle ϕ_k, $\mathbf{g}$ all the projections.

Algebraic methods

$$\mathbf{g} = \begin{bmatrix} \mathbf{g}_1 \\ \vdots \\ \mathbf{g}_K \end{bmatrix}, \mathbf{H} = \begin{bmatrix} \mathbf{H}_1 \\ \vdots \\ \mathbf{H}_K \end{bmatrix} \rightarrow \mathbf{g}_k = \mathbf{H}_k \mathbf{f} + \epsilon_k \rightarrow \mathbf{g} = \sum_k \mathbf{H}_k \mathbf{f} + \epsilon_k = \mathbf{H}\mathbf{f} + \epsilon$$

- $\mathbf{H}$ is huge dimensional: 2D: $10^6 \times 10^6$, 3D: $10^9 \times 10^9$.
- $\mathbf{H}\mathbf{f}$ corresponds to forward projection
- $\mathbf{H}^t\mathbf{g}$ corresponds to Back projection (BP)
- $\mathbf{H}$ may not be invertible and even not square
- $\mathbf{H}$ is, in general, ill-conditioned
- In limited angle tomography $\mathbf{H}$ is under determined, si the problem has infinite number of solutions
- Minimum Norm Solution

$$\widehat{\mathbf{f}} = \mathbf{H}^t(\mathbf{H}\mathbf{H}^t)^{-1}\mathbf{g} = \sum_k \mathbf{H}_k^t(\mathbf{H}_k\mathbf{H}_k^t)^{-1}\mathbf{g}_k$$

can be interpreted as the Filtered Back Projection solution.

Algebraic methods

- Minimum Norm Solution:
  minimize $\|\mathbf{f}\|_2^2$ s.t. $\mathbf{H}\mathbf{f} = \mathbf{g} \longrightarrow \widehat{\mathbf{f}} = \mathbf{H}^t(\mathbf{H}\mathbf{H}^t)^{-1}\mathbf{g}$
- Least square Solution:
  $\widehat{\mathbf{f}} = \arg\min_{\mathbf{f}} \left\{J(\mathbf{f}) = \|\mathbf{g} - \mathbf{H}\mathbf{f}\|^2\right\} \rightarrow \widehat{\mathbf{f}} = (\mathbf{H}^t\mathbf{H})^{-1}\mathbf{H}^t\mathbf{g}$
- Quadratic Regularization:

$$J(\mathbf{f}) = \|\mathbf{g} - \mathbf{H}\mathbf{f}\|^2 + \lambda\|\mathbf{f}\|_2^2 \longrightarrow \widehat{\mathbf{f}} = (\mathbf{H}^t\mathbf{H} + \lambda\mathbf{I})^{-1}\mathbf{H}^t\mathbf{g}$$

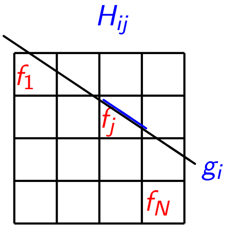
- L1 Regularization: $J(\mathbf{f}) = \|\mathbf{g} - \mathbf{H}\mathbf{f}\|^2 + \lambda\|\mathbf{f}\|_1$
- Lpq Regularization: $J(\mathbf{f}) = \|\mathbf{g} - \mathbf{H}\mathbf{f}\|_p^p + \lambda\|\mathbf{D}\mathbf{f}\|_q^q$
- More general Regularization:

$$J(\mathbf{f}) = \sum_i \phi(g_i - [\mathbf{H}\mathbf{f}]_i) + \lambda \sum_j \psi([\mathbf{D}\mathbf{f}]_j)$$

$$J(\mathbf{f}) = \Delta_1(\mathbf{g}, \mathbf{H}\mathbf{f}) + \lambda\Delta_2(\mathbf{f}, \mathbf{f}_0)$$

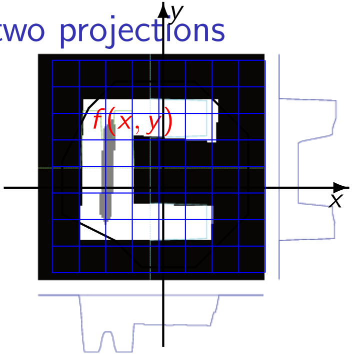
with Δ_1 and Δ_2 any distances (L2, L2, ..) or divergence (KL)

Computed Tomography with only two projections



$$g(r,\phi) = \int_L f(x,y)\,\mathrm{d}l$$
$$f(x,y) = \sum_j f_j\, b_j(x,y)$$
$$b_j(x,y) = \begin{cases} 1 & \text{if } (x,y) \in \text{ pixel } j \\ 0 & \text{else} \end{cases}$$

$$g_i = \sum_{j=1}^{N} H_{ij}\, f_j + \epsilon_i$$
$$\mathbf{g} = \mathbf{H}\mathbf{f} + \epsilon$$



Case study: Reconstruction from 2 projections

$g_1(x) = \int f(x,y)\,\mathrm{d}y$,
$g_2(y) = \int f(x,y)\,\mathrm{d}x$

Very ill-posed inverse problem

$f(x,y) = g_1(x)\, g_2(y)\, \Omega(x,y)$

$\Omega(x,y)$ is a Copula:

$\int \Omega(x,y)\,\mathrm{d}x = 1$
$\int \Omega(x,y)\,\mathrm{d}y = 1$

Simple example

1	3	4
2	4	6
3	7	

?	?	4
?	?	6
3	7	

f_1	f_3	g_3
f_2	f_4	g_4
g_1	g_2	

1	-1	0
-1	1	0
0	0	

-1	1	0
1	-1	0
0	0	

$$\begin{bmatrix} g_1 \\ g_2 \\ g_3 \\ g_4 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix}$$

f_1	f_4	f_7	g_4
f_2	f_5	f_8	g_5
f_3	f_6	f_9	g_6
g_1	g_2	g_3	

- $\mathbf{H}\mathbf{f} = \mathbf{g} \longrightarrow \widehat{\mathbf{f}} = \mathbf{H}^{-1}\mathbf{g}$ if $\mathbf{H}$ invertible.
- $\mathbf{H}$ is rank deficient: $\text{rank}(\mathbf{H}) = 3$
- Problem has infinite number of solutions.
- How to find all those solutions ?
- Which one is the good one? Needs prior information.
- To find an unique solution, one needs either more data or prior information.

Prior information or constraints

- Positivity: $f_j > 0$ or $f_j \in \mathrm{R}^+$
- Boundedness: $1 > f_j) > 0$ or $f_j \in [0, 1]$
- Smoothness: f_j depends on the neighborhoods.
- Sparsity: many f_j are zeros.
- Sparsity in a transform domain: $\mathbf{f} = \mathbf{D}\mathbf{z}$ and many z_j are zeros.
- Discrete valued (DV): $f_j \in \{0, 1, ..., K\}$
- Binary valued (BV): $f_j \in \{0, 1\}$
- Compactness: $f(\mathbf{r})$ is non zero in one or few non-overlapping compact regions
- Combination of the above mentioned constraints
- Main mathematical questions:
  - Which combination results to unique solution ?
  - How to apply them ?

Deterministic approaches

- Iterative methods: SIRT, ART, Quadratic or L1 regularization, Bloc Coordinate Descent, Multiplicative ART,...
- Criteria: $J(\mathbf{f}) = \|\mathbf{g} - \mathbf{H}\mathbf{f}\|^2 + \lambda\|\mathbf{D}\mathbf{f}\|^2$
- Gradient based algorithms: $\nabla J(\mathbf{f}) = -2\mathbf{H}^t(\mathbf{g} - \mathbf{H}\mathbf{f}) + 2\lambda\mathbf{D}^t\mathbf{D}\mathbf{f}$
- Simplest algorithm:

$$\widehat{\mathbf{f}}^{(k+1)} = \widehat{\mathbf{f}}^{(k)} + \alpha^{(k)}\left[\mathbf{H}^t(\mathbf{g} - \mathbf{H}\widehat{\mathbf{f}}^{(k)}) + 2\lambda\mathbf{D}^t\mathbf{D}\widehat{\mathbf{f}}^{(k)}\right]$$

- More criteria:

$$J(\mathbf{f}) = \sum_i \phi(g_i - [\mathbf{H}\mathbf{f}]_i) + \lambda \sum_j \psi([\mathbf{D}\mathbf{f}]_j)$$

  with $\phi(t)$ and $\psi(t) = \{t^2, |t|, |t|^p, ...\}$ or

$$J(\mathbf{f}) = \Delta_1(\mathbf{g}, \mathbf{H}\mathbf{f}) + \lambda\Delta_2(\mathbf{f}, \mathbf{f}_0)$$

- Imposing constraints in each iteration (example: DART)
- Mathematical studies of uniqueness and convergence of these algorithms are necessary

Bayesian estimation approach

$$\mathcal{M}: \qquad \mathbf{g} = \mathbf{H}\mathbf{f} + \boldsymbol{\epsilon}$$

- Observation model $\mathcal{M}$ + Hypothesis on the noise $\boldsymbol{\epsilon} \longrightarrow$
$$p(\mathbf{g}|\mathbf{f};\mathcal{M}) = p_{\boldsymbol{\epsilon}}(\mathbf{g} - \mathbf{H}\mathbf{f})$$
- A priori information $\quad p(\mathbf{f}|\mathcal{M})$
- Bayes : $$p(\mathbf{f}|\mathbf{g};\mathcal{M}) = \frac{p(\mathbf{g}|\mathbf{f};\mathcal{M})\, p(\mathbf{f}|\mathcal{M})}{p(\mathbf{g}|\mathcal{M})}$$
- Maximum A Posteriori (MAP) :

$$\begin{aligned}\widehat{\mathbf{f}} &= \arg\max_{\mathbf{f}} \{p(\mathbf{f}|\mathbf{g})\} = \arg\max_{\mathbf{f}} \{p(\mathbf{g}|\mathbf{f})\ p(\mathbf{f})\} \\ &= \arg\min_{\mathbf{f}} \{J(\mathbf{f}) = -\ln p(\mathbf{g}|\mathbf{f}) - \ln p(\mathbf{f})\}\end{aligned}$$

- Link with Regularization:

$$\widehat{\mathbf{f}} = \arg\min_{\mathbf{f}} \{J(\mathbf{f}) = \Delta_1(\mathbf{g}, \mathbf{H}\mathbf{f}) + \lambda\mathcal{R}(\mathbf{f})\}$$

with $\Delta_1(\mathbf{g}, \mathbf{H}\mathbf{f}) = -\ln p(\mathbf{g}|\mathbf{f})$ and $\lambda\mathcal{R}(\mathbf{f}) = -\ln p(\mathbf{f})$

Case of linear models and Gaussian priors

$$\mathbf{g} = \mathbf{H}\mathbf{f} + \boldsymbol{\epsilon}$$

- Prior knowledge on the noise:

$$\boldsymbol{\epsilon} \sim \mathcal{N}(0, {v_\epsilon}^2\mathbf{I}) \rightarrow p(\mathbf{g}|\mathbf{f}) \propto \exp\left[-\frac{1}{2{v_\epsilon}^2}\|\mathbf{g} - \mathbf{H}\mathbf{f}\|^2\right]$$

- Prior knowledge on $\mathbf{f}$:

$$\mathbf{f} \sim \mathcal{N}(0, v_f^2(\mathbf{D}'\mathbf{D})^{-1}) \rightarrow p(\mathbf{f}) \propto \exp\left[-\frac{1}{2v_f^2}\|\mathbf{D}\mathbf{f}\|^2\right]$$

- A posteriori:

$$p(\mathbf{f}|\mathbf{g}) \propto \exp\left[-\frac{1}{2{v_\epsilon}^2}\|\mathbf{g} - \mathbf{H}\mathbf{f}\|^2 - \frac{1}{2v_f^2}\|\mathbf{D}\mathbf{f}\|^2\right]$$

- MAP : $\widehat{\mathbf{f}} = \arg\max_{\mathbf{f}}\{p(\mathbf{f}|\mathbf{g})\} = \arg\min_{\mathbf{f}}\{J(\mathbf{f})\}$

  with $J(\mathbf{f}) = \|\mathbf{g} - \mathbf{H}\mathbf{f}\|^2 + \lambda\|\mathbf{D}\mathbf{f}\|^2, \qquad \lambda = \frac{{v_\epsilon}^2}{v_f^2}$

- Advantage : characterization of the solution

$$p(\mathbf{f}|\mathbf{g}) = \mathcal{N}(\widehat{\mathbf{f}}, \widehat{\boldsymbol{\Sigma}}) \quad \text{with}$$

$$\widehat{\mathbf{f}} = \left(\mathbf{H}'\mathbf{H} + \lambda\mathbf{D}'\mathbf{D}\right)^{-1}\mathbf{H}'\mathbf{g}, \quad \widehat{\boldsymbol{\Sigma}} = v_\epsilon\left(\mathbf{H}'\mathbf{H} + \lambda\mathbf{D}'\mathbf{D}\right)^{-1}$$

MAP estimation with other priors:

$$\widehat{\mathbf{f}} = \arg\min_{\mathbf{f}} \{J(\mathbf{f})\} \quad \text{with} \quad J(\mathbf{f}) = \frac{1}{v_\epsilon}\|\mathbf{g} - \mathbf{H}\mathbf{f}\|^2 + \Omega(\mathbf{f})$$

Separable priors:

- Gaussian:
  $p(f_j) \propto \exp\left[-\alpha|f_j|^2\right] \longrightarrow \Omega(\mathbf{f}) = \alpha\sum_j |f_j|^2 = \|\mathbf{f}\|_2^2$
- Generalized Gaussian:
  $p(f_j) \propto \exp\left[-\alpha|f_j|^p\right],\ 1 < p < 2 \rightarrow \Omega(\mathbf{f}) = \alpha\sum_j |f_j|^p = \|\mathbf{f}\|_p^p$
- Gamma: $p(f_j) \propto f_j^\alpha \exp\left[-\beta f_j\right] \longrightarrow \Omega(\mathbf{f}) = \alpha\sum_j \ln f_j + \beta f_j$
- Beta:
  $p(f_j) \propto f_j^\alpha (1-f_j)^\beta \longrightarrow \Omega(\mathbf{f}) = \alpha\sum_j \ln f_j + \beta\sum_j \ln(1-f_j)$

Markovian models:

$$p(f_j|\mathbf{f}) \propto \exp\left[-\alpha\sum_{i\in N_j}\phi(f_j,f_i)\right] \longrightarrow \quad \Omega(\mathbf{f}) = \alpha\sum_j\sum_{i\in N_j}\phi(f_j,f_i),$$

Sparsity enforcing models

- 3 classes of models: 1- Generalized Gaussian, 2- Mixture models and 3- Heavy tailed (Cauchy and Student-t)
- Student-t model

$$\mathcal{S}t(f|\nu) \propto \exp\left[-\frac{\nu+1}{2}\log\left(1+f^2/\nu\right)\right]$$

- Infinite Gausian Scaled Mixture (IGSM) equivalence

$$\mathcal{S}t(f|\nu) \propto \int_0^\infty \mathcal{N}(f|,0,1/z)\,\mathcal{G}(z|\alpha,\beta)\,\mathrm{d}z, \quad \text{with } \alpha=\beta=\nu/2$$

$$\left\{\begin{array}{lll} p(\mathbf{f}|\mathbf{z}) & = \prod_j p(f_j|z_j) = \prod_j \mathcal{N}(f_j|0,1/z_j) \propto \exp\left[-\frac{1}{2}\sum_j z_j f_j^2\right] \\ p(\mathbf{z}|\alpha,\beta) & = \prod_j \mathcal{G}(z_j|\alpha,\beta) \propto \prod_j z_j^{(\alpha-1)} \exp\left[-\beta z_j\right] \\ & \propto \exp\left[\sum_j (\alpha-1)\ln z_j - \beta z_j\right] \\ p(\mathbf{f},\mathbf{z}|\alpha,\beta) & \propto \exp\left[-\frac{1}{2}\sum_j z_j f_j^2 + (\alpha-1)\ln z_j - \beta z_j\right] \end{array}\right.$$

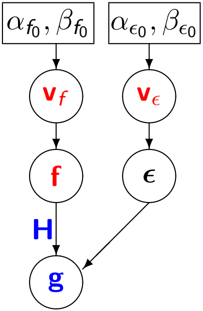
Non stationary noise and sparsity enforcing model

– Non stationary noise:

$$\mathbf{g} = \mathbf{Hf} + \boldsymbol{\epsilon}, \;\; \epsilon_i \sim \mathcal{N}(\epsilon_i|0, v_{\epsilon_i}) \rightarrow \boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{\epsilon}|0, \mathbf{V}_\epsilon = \mathrm{diag}\,[v_{\epsilon 1}, \cdots, v_{\epsilon M}])$$

– Student-t prior model and its equivalent IGSM :

$$f_j|v_{f_j} \sim \mathcal{N}(f_j|0, v_{f_j}) \text{ and } v_{f_j} \sim \mathcal{IG}(v_{f_j}|\alpha_{f_0}, \beta_{f_0}) \rightarrow f_j \sim \mathcal{S}t(f_j|\alpha_{f_0}, \beta_{f_0})$$



$$\begin{cases} p(\mathbf{g}|\mathbf{f}, \mathbf{v}_\epsilon) = \mathcal{N}(\mathbf{g}|\mathbf{Hf}, \mathbf{V}_\epsilon), & \mathbf{V}_\epsilon = \mathrm{diag}\,[\mathbf{v}_\epsilon] \\ p(\mathbf{f}|\mathbf{v}_f) = \mathcal{N}(\mathbf{g}|0, \mathbf{V}_f), & \mathbf{V}_f = \mathrm{diag}\,[\mathbf{v}_f] \end{cases}$$

$$\begin{cases} p(\mathbf{v}_\epsilon) = \prod_i \mathcal{IG}(v_{\epsilon_i}|\alpha_{\epsilon 0}, \beta_{\epsilon 0}) \\ p(\mathbf{v}_f) = \prod_i \mathcal{IG}(v_{f_j}|\alpha_{f_0}, \beta_{f_0}) \end{cases}$$

$$p(\mathbf{f}, \mathbf{v}_\epsilon, \mathbf{v}_f|\mathbf{g}) \propto p(\mathbf{g}|\mathbf{f}, \mathbf{v}_\epsilon)\, p(\mathbf{f}|\mathbf{v}_f)\, p(\mathbf{v}_\epsilon)\, p(\mathbf{v}_f)$$

– JMAP: $(\widehat{\mathbf{f}}, \hat{\mathbf{v}}_\epsilon, \hat{\mathbf{v}}_f) = \arg\max_{(\mathbf{f}, \mathbf{v}_\epsilon, \mathbf{v}_f)} \{p(\mathbf{f}, \mathbf{v}_\epsilon, \mathbf{v}_f|\mathbf{g})\}$

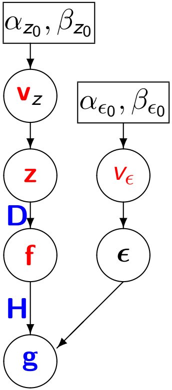
– VBA: Approximate $p(\mathbf{f}, \mathbf{v}_\epsilon, \mathbf{v}_f|\mathbf{g})$
by $q_1(\mathbf{f})\, q_2(\mathbf{v}_\epsilon)\, q_3(\mathbf{v}_f)$

Sparse model in a Transform domain 1

$\mathbf{g} = \mathbf{H}\mathbf{f} + \epsilon, \quad \mathbf{f} = \mathbf{D}\mathbf{z}, \quad \mathbf{z}$ sparse

$$\begin{cases} p(\mathbf{g}|\mathbf{z}, v_\epsilon) = \mathcal{N}(\mathbf{g}|\mathbf{H}\mathbf{D}\mathbf{f}, v_\epsilon \mathbf{I}) \\ p(\mathbf{z}|\mathbf{v}_z) = \mathcal{N}(\mathbf{z}|0, \mathbf{V}_z), \qquad \mathbf{V}_z = \operatorname{diag}[\mathbf{v}_z] \end{cases}$$

$$\begin{cases} p(v_\epsilon) = \mathcal{IG}(v_\epsilon|\alpha_{\epsilon_0}, \beta_{\epsilon_0}) \\ p(\mathbf{v}_z) = \prod_j \mathcal{IG}(v_{z_j}|\alpha_{z_0}, \beta_{z_0}) \end{cases}$$



$$p(\mathbf{z}, v_\epsilon, \mathbf{v}_z, v_\xi|\mathbf{g}) \propto p(\mathbf{g}|\mathbf{z}, v_\epsilon)\, p(\mathbf{z}|\mathbf{v}_z)\, p(v_\epsilon)\, p(\mathbf{v}_z)\, p(v_\xi)$$

– JMAP:

$$(\widehat{\mathbf{z}}, \hat{v}_\epsilon, \widehat{\mathbf{v}}_z) = \underset{(\mathbf{z}, v_\epsilon, \mathbf{v}_z)}{\arg\max} \{p(\mathbf{z}, v_\epsilon, \mathbf{v}_z|\mathbf{g})\}$$

Alternate optimization:

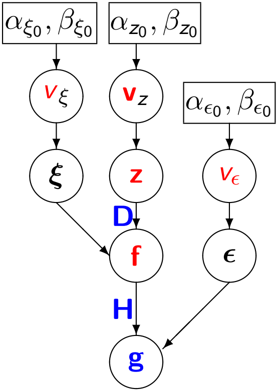
$$\begin{cases} \widehat{\mathbf{z}} = \arg\min_{\mathbf{z}} \{J(\mathbf{z})\} \text{ with:} \\ J(\mathbf{z}) = \frac{1}{2\hat{v}_\epsilon}\|\mathbf{g} - \mathbf{H}\mathbf{D}\mathbf{z}\|^2 + \|\mathbf{V}_z{}^{-1/2}\mathbf{z}\|^2 \\ \widehat{v}_{z_j} = \frac{\beta_{z_0} + \widehat{z}_j^2}{\alpha_{z_0} + 1/2} \\ \hat{v}_\epsilon = \frac{\beta_{\epsilon_0} + \|\mathbf{g} - \mathbf{H}\mathbf{D}\widehat{\mathbf{z}}\|^2}{\alpha_{\epsilon_0} + M/2} \end{cases}$$

– VBA: Approximate

$p(\mathbf{z}, v_\epsilon, \mathbf{v}_z, v_\xi|\mathbf{g})$ by $q_1(\mathbf{z})\, q_2(v_\epsilon)\, q_3(\mathbf{v}_z)$

Alternate optimization.

Sparse model in a Transform domain 2



$$\mathbf{g} = \mathbf{H}\mathbf{f} + \boldsymbol{\epsilon}, \quad \mathbf{f} = \mathbf{D}\mathbf{z} + \boldsymbol{\xi}, \quad \mathbf{z} \text{ sparse}$$

$$\begin{cases} p(\mathbf{g}|\mathbf{f}, v_\epsilon) = \mathcal{N}(\mathbf{g}|\mathbf{H}\mathbf{f}, v_\epsilon \mathbf{I}) \\ p(\mathbf{f}|\mathbf{z}) = \mathcal{N}(\mathbf{f}|\mathbf{D}\mathbf{z}, v_\xi \mathbf{I}), \\ p(\mathbf{z}|\mathbf{v}_z) = \mathcal{N}(\mathbf{z}|0, \mathbf{V}_z), \qquad \mathbf{V}_z = \text{diag}\,[\mathbf{v}_z] \end{cases}$$

$$\begin{cases} p(v_\epsilon) = \mathcal{IG}(v_\epsilon|\alpha_{\epsilon_0}, \beta_{\epsilon_0}) \\ p(\mathbf{v}_z) = \prod_i \mathcal{IG}(v_{z_j}|\alpha_{z_0}, \beta_{z_0}) \\ p(v_\xi) = \mathcal{IG}(v_\xi|\alpha_{\xi_0}, \beta_{\xi_0}) \end{cases}$$

$$p(\mathbf{f}, \mathbf{z}, v_\epsilon, \mathbf{v}_z, v_\xi|\mathbf{g}) \propto p(\mathbf{g}|\mathbf{f}, \mathbf{v}_\epsilon)\, p(\mathbf{f}|\mathbf{z}_f)\, p(\mathbf{z}|\mathbf{v}_z)$$
$$p(v_\epsilon)\, p(\mathbf{v}_z)\, p(v_\xi)$$

– JMAP:

$$(\widehat{\mathbf{f}}, \widehat{\mathbf{z}}, \hat{v}_\epsilon, \widehat{\mathbf{v}}_z, \widehat{v}_\xi) = \underset{(\mathbf{f}, \mathbf{z}, v_\epsilon, \mathbf{v}_z, v_\xi)}{\arg\max} \left\{p(\mathbf{f}, \mathbf{z}, v_\epsilon, \mathbf{v}_z, v_\xi|\mathbf{g})\right\}$$

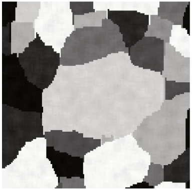
Alternate optimization.

– VBA: Approximate

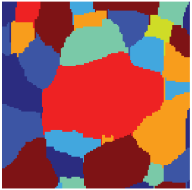
$p(\mathbf{f}, \mathbf{z}, v_\epsilon, \mathbf{v}_z, v_\xi|\mathbf{g})$ by $q_1(\mathbf{f})\, q_2(\mathbf{z})\, q_3(v_\epsilon)\, q_4(\mathbf{v}_z)\, q_5(v_\xi)$

Alternate optimization.

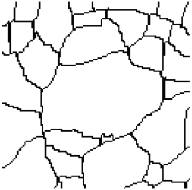
Gauss-Markov-Potts prior models for images



$f(\mathbf{r})$



$z(\mathbf{r})$



$c(\mathbf{r}) = 1 - \delta(z(\mathbf{r}) - z(\mathbf{r}'))$

$$p(f(\mathbf{r})|z(\mathbf{r}) = k, m_k, v_k) = \mathcal{N}(m_k, v_k)$$

$$p(f(\mathbf{r})) = \sum_k P(z(\mathbf{r}) = k)\,\mathcal{N}(m_k, v_k) \quad \text{Mixture of Gaussians}$$

- Separable iid hidden variables: $p(\mathbf{z}) = \prod_{\mathbf{r}} p(z(\mathbf{r}))$
- Markovian hidden variables: $p(\mathbf{z})$ Potts-Markov:

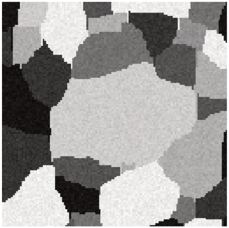
$$p(z(\mathbf{r})|z(\mathbf{r}'), \mathbf{r}' \in \mathcal{V}(\mathbf{r})) \propto \exp\left[\gamma \sum_{\mathbf{r}' \in \mathcal{V}(\mathbf{r})} \delta(z(\mathbf{r}) - z(\mathbf{r}'))\right]$$

$$p(\mathbf{z}) \propto \exp\left[\gamma \sum_{\mathbf{r} \in \mathcal{R}} \sum_{\mathbf{r}' \in \mathcal{V}(\mathbf{r})} \delta(z(\mathbf{r}) - z(\mathbf{r}'))\right]$$

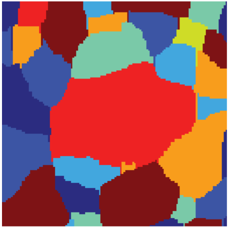
Four different cases

To each pixel of the image is associated 2 variables $f(\mathbf{r})$ and $z(\mathbf{r})$

- $\mathbf{f}|\mathbf{z}$ Gaussian iid, $\mathbf{z}$ iid :
  Mixture of Gaussians
- $\mathbf{f}|\mathbf{z}$ Gauss-Markov, $\mathbf{z}$ iid :
  Mixture of Gauss-Markov
- $\mathbf{f}|\mathbf{z}$ Gaussian iid, $\mathbf{z}$ Potts-Markov :
  Mixture of Independent Gaussians
  (MIG with Hidden Potts)
- $\mathbf{f}|\mathbf{z}$ Markov, $\mathbf{z}$ Potts-Markov :
  Mixture of Gauss-Markov
  (MGM with hidden Potts)

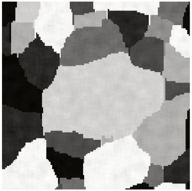


$f(\mathbf{r})$

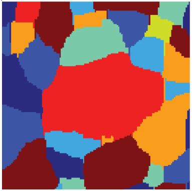


$z(\mathbf{r})$

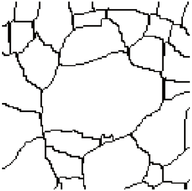
Gauss-Markov-Potts prior models for images



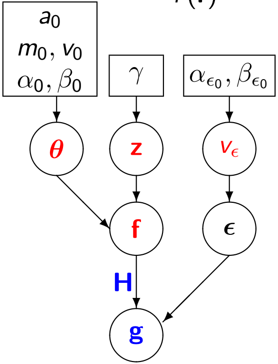
$f(\mathbf{r})$



$z(\mathbf{r})$



$c(\mathbf{r}) = 1 - \delta(z(\mathbf{r}) - z(\mathbf{r}'))$



$$\begin{cases} \mathbf{g} = \mathbf{H}\mathbf{f} + \epsilon \\ p(\mathbf{g}|\mathbf{f}, v_\epsilon) = \mathcal{N}(\mathbf{g}|\mathbf{H}\mathbf{f}, v_\epsilon \mathbf{I}) \\ p(v_\epsilon) = \mathcal{IG}(v_\epsilon|\alpha_{\epsilon_0}, \beta_{\epsilon_0}) \end{cases}$$

$$\begin{cases} p(f(\mathbf{r})|z(\mathbf{r}) = k, m_k, v_k) = \mathcal{N}(f(\mathbf{r})|m_k, v_k) \\ p(\mathbf{f}|\mathbf{z}, \boldsymbol{\theta}) = \sum_k \prod_{\mathbf{r} \in \mathcal{R}_k} a_k \mathcal{N}(f(\mathbf{r})|m_k, v_k), \\ \qquad \boldsymbol{\theta} = \{(a_k, m_k, v_k), k = 1, \cdots, K\} \\ p(\boldsymbol{\theta}) = \mathcal{D}(\mathbf{a}|a_0)\mathcal{N}(\mathbf{a}|m_0, v0)\mathcal{IG}(\mathbf{v}|\alpha_0, \beta_0) \\ p(\mathbf{z}|\gamma) \propto \exp\left[\gamma \sum_{\mathbf{r}} \sum_{\mathbf{r}' \in \mathcal{N}(\mathbf{r})} \delta(z(\mathbf{r}) - z(\mathbf{r}'))\right] \quad \text{Potts MRF} \end{cases}$$

$$p(\mathbf{f}, \mathbf{z}, \boldsymbol{\theta}|\mathbf{g}) \propto p(\mathbf{g}|\mathbf{f}, v_\epsilon)\, p(\mathbf{f}|\mathbf{z}, \boldsymbol{\theta})\, p(\mathbf{z}|\gamma)$$

MCMC: Gibbs Sampling

VBA: Alternate optimization.

Bayesian Computation and Algorithms

- Joint posterior probability law of all the unknowns $\mathbf{f}, \mathbf{z}, \boldsymbol{\theta}$

$$p(\mathbf{f}, \mathbf{z}, \boldsymbol{\theta}|\mathbf{g}) \propto p(\mathbf{g}|\mathbf{f}, \boldsymbol{\theta}_1)\; p(\mathbf{f}|\mathbf{z}, \boldsymbol{\theta}_2)\; p(\mathbf{z}|\boldsymbol{\theta}_3)\; p(\boldsymbol{\theta})$$

- Often, the expression of $p(\mathbf{f}, \mathbf{z}, \boldsymbol{\theta}|\mathbf{g})$ is complex.
- Its optimization (for Joint MAP) or
  its marginalization or integration (for Marginal MAP or PM)
  is not easy
- Two main techniques:
  - MCMC:
    Needs the expressions of the conditionals
    $p(\mathbf{f}|\mathbf{z}, \boldsymbol{\theta}, \mathbf{g})$, $p(\mathbf{z}|\mathbf{f}, \boldsymbol{\theta}, \mathbf{g})$, and $p(\boldsymbol{\theta}|\mathbf{f}, \mathbf{z}, \mathbf{g})$
  - VBA: Approximate $p(\mathbf{f}, \mathbf{z}, \boldsymbol{\theta}|\mathbf{g})$ by a separable one

$$q(\mathbf{f}, \mathbf{z}, \boldsymbol{\theta}|\mathbf{g}) = q_1(\mathbf{f})\; q_2(\mathbf{z})\; q_3(\boldsymbol{\theta})$$

    and do any computations with these separable ones.

MCMC based algorithm

$$p(\mathbf{f}, \mathbf{z}, \boldsymbol{\theta} | \mathbf{g}) \propto p(\mathbf{g} | \mathbf{f}, \mathbf{z}, \boldsymbol{\theta}_1) \, p(\mathbf{f} | \mathbf{z}, \boldsymbol{\theta}_2) \, p(\mathbf{z} | \boldsymbol{\theta}_3) \, p(\boldsymbol{\theta})$$

General Gibbs sampling scheme:

$$\widehat{\mathbf{f}} \sim p(\mathbf{f} | \widehat{\mathbf{z}}, \widehat{\boldsymbol{\theta}}, \mathbf{g}) \longrightarrow \widehat{\mathbf{z}} \sim p(\mathbf{z} | \widehat{\mathbf{f}}, \widehat{\boldsymbol{\theta}}, \mathbf{g}) \longrightarrow \widehat{\boldsymbol{\theta}} \sim (\boldsymbol{\theta} | \widehat{\mathbf{f}}, \widehat{\mathbf{z}}, \mathbf{g})$$

- Generate samples $\mathbf{f}$ using $p(\mathbf{f} | \widehat{\mathbf{z}}, \widehat{\boldsymbol{\theta}}, \mathbf{g}) \propto p(\mathbf{g} | \mathbf{f}, \boldsymbol{\theta}) \, p(\mathbf{f} | \widehat{\mathbf{z}}, \widehat{\boldsymbol{\theta}})$
  When Gaussian, can be done via optimization of a quadratic criterion.
- Generate samples $\mathbf{z}$ using $p(\mathbf{z} | \widehat{\mathbf{f}}, \widehat{\boldsymbol{\theta}}, \mathbf{g}) \propto p(\mathbf{g} | \widehat{\mathbf{f}}, \widehat{\mathbf{z}}, \widehat{\boldsymbol{\theta}}) \, p(\mathbf{z})$
  Often needs sampling (hidden discrete variable)
- Generate samples $\boldsymbol{\theta}$ using
  $p(\boldsymbol{\theta} | \widehat{\mathbf{f}}, \widehat{\mathbf{z}}, \mathbf{g}) \propto p(\mathbf{g} | \widehat{\mathbf{f}}, \sigma_\epsilon^2 \mathbf{I}) \, p(\widehat{\mathbf{f}} | \widehat{\mathbf{z}}, (m_k, v_k)) \, p(\boldsymbol{\theta})$
  Use of Conjugate priors $\longrightarrow$ analytical expressions.
- After convergence use samples to compute means and variances.

Application in CT: Reconstruction from 2 projections

$\mathbf{g}|\mathbf{f}$

$\mathbf{g} = \mathbf{H}\mathbf{f} + \epsilon$

$\mathbf{g}|\mathbf{f} \sim \mathcal{N}(\mathbf{H}\mathbf{f}, \sigma_\epsilon^2 \mathbf{I})$

Gaussian

$\mathbf{f}|\mathbf{z}$

iid Gaussian

or

Gauss-Markov

$\mathbf{z}$

iid

or

Potts

$\mathbf{c}$

$q(\mathbf{r}) \in \{0, 1\}$

$1 - \delta(z(\mathbf{r}) - z(\mathbf{r}'))$

binary

$$p(\mathbf{f}, \mathbf{z}, \boldsymbol{\theta}|\mathbf{g}) \propto p(\mathbf{g}|\mathbf{f}, \boldsymbol{\theta}_1)\, p(\mathbf{f}|\mathbf{z}, \boldsymbol{\theta}_2)\, p(\mathbf{z}|\boldsymbol{\theta}_3)\, p(\boldsymbol{\theta})$$

Proposed algorithms

$$p(\mathbf{f},\mathbf{z},\boldsymbol{\theta}|\mathbf{g}) \propto p(\mathbf{g}|\mathbf{f},\boldsymbol{\theta}_1)\, p(\mathbf{f}|\mathbf{z},\boldsymbol{\theta}_2)\, p(\mathbf{z}|\boldsymbol{\theta}_3)\, p(\boldsymbol{\theta})$$

- MCMC based general scheme:

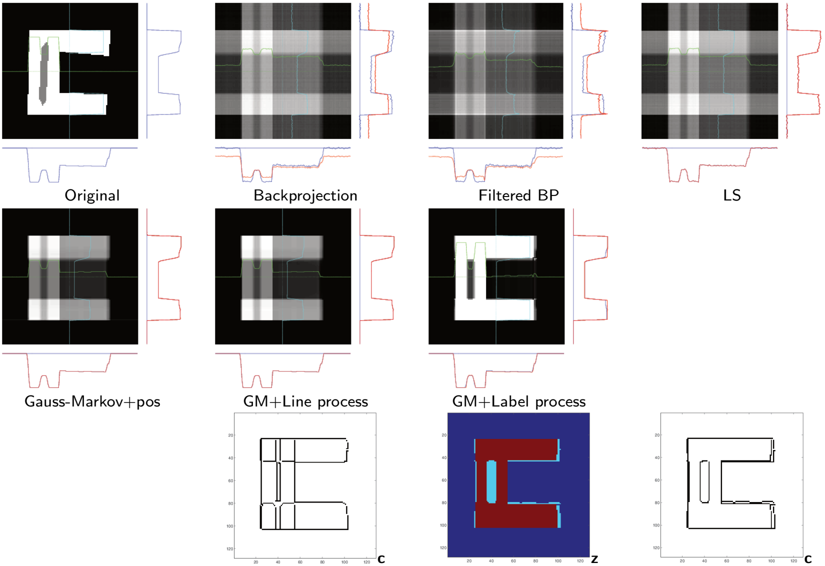
$$\widehat{\mathbf{f}} \sim p(\mathbf{f}|\widehat{\mathbf{z}},\widehat{\boldsymbol{\theta}},\mathbf{g}) \longrightarrow \widehat{\mathbf{z}} \sim p(\mathbf{z}|\widehat{\mathbf{f}},\widehat{\boldsymbol{\theta}},\mathbf{g}) \longrightarrow \widehat{\boldsymbol{\theta}} \sim (\boldsymbol{\theta}|\widehat{\mathbf{f}},\widehat{\mathbf{z}},\mathbf{g})$$

Iterative algorithme:

- Estimate $\mathbf{f}$ using $p(\mathbf{f}|\widehat{\mathbf{z}},\widehat{\boldsymbol{\theta}},\mathbf{g}) \propto p(\mathbf{g}|\mathbf{f},\boldsymbol{\theta})\, p(\mathbf{f}|\widehat{\mathbf{z}},\widehat{\boldsymbol{\theta}})$
  Needs optimization of a quadratic criterion.
- Estimate $\mathbf{z}$ using $p(\mathbf{z}|\widehat{\mathbf{f}},\widehat{\boldsymbol{\theta}},\mathbf{g}) \propto p(\mathbf{g}|\widehat{\mathbf{f}},\widehat{\mathbf{z}},\widehat{\boldsymbol{\theta}})\, p(\mathbf{z})$
  Needs sampling of a Potts Markov field.
- Estimate $\boldsymbol{\theta}$ using
  $p(\boldsymbol{\theta}|\widehat{\mathbf{f}},\widehat{\mathbf{z}},\mathbf{g}) \propto p(\mathbf{g}|\widehat{\mathbf{f}},\sigma_\epsilon^2\mathbf{I})\, p(\widehat{\mathbf{f}}|\widehat{\mathbf{z}},(m_k,v_k))\, p(\boldsymbol{\theta})$
  Conjugate priors $\longrightarrow$ analytical expressions.

- Variational Bayesian Approximation
  - Approximate $p(\mathbf{f},\mathbf{z},\boldsymbol{\theta}|\mathbf{g})$ by $q_1(\mathbf{f})\, q_2(\mathbf{z})\, q_3(\boldsymbol{\theta})$

Results with two projections



Implementation issues

- In almost all the algorithms, the step of computation of $\widehat{\mathbf{f}}$ needs an optimization algorithm.
- The criterion to optimize is often in the form of

$$J(\mathbf{f}) = \|\mathbf{g} - \mathbf{H}\mathbf{f}\|^2 + \lambda\|\mathbf{D}\mathbf{f}\|^2$$

- Very often, we use the gradient based algorithms which need to compute

$$\nabla J(\mathbf{f}) = -2\mathbf{H}^t(\mathbf{g} - \mathbf{H}\mathbf{f}) + 2\lambda\mathbf{D}^t\mathbf{D}\mathbf{f}$$

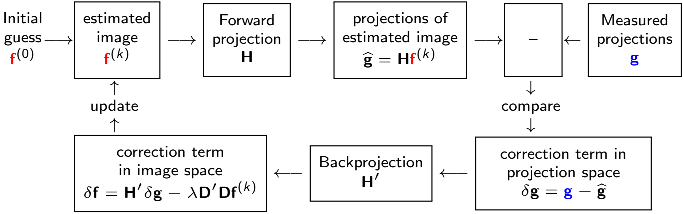
- So, for the simplest case, in each step, we have

$$\widehat{\mathbf{f}}^{(k+1)} = \widehat{\mathbf{f}}^{(k)} + \alpha^{(k)}\left[\mathbf{H}^t(\mathbf{g} - \mathbf{H}\widehat{\mathbf{f}}^{(k)}) + 2\lambda\mathbf{D}^t\mathbf{D}\widehat{\mathbf{f}}^{(k)}\right]$$

Gradient based algorithms

$$\widehat{\mathbf{f}}^{(k+1)} = \widehat{\mathbf{f}}^{(k)} + \alpha \left[\mathbf{H}' \left(\mathbf{g} - \mathbf{H}\widehat{\mathbf{f}}^{(k)}\right) - \lambda \mathbf{D}'\mathbf{D}\widehat{\mathbf{f}}^{(k)}\right]$$

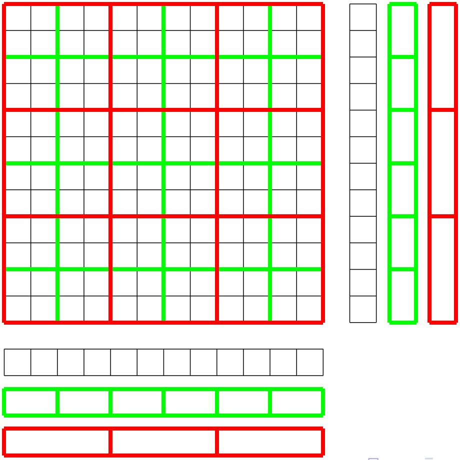
1. Compute $\widehat{\mathbf{g}} = \mathbf{H}\widehat{\mathbf{f}}$ (Forward projection)
2. Compute $\delta\mathbf{g} = \mathbf{g} - \widehat{\mathbf{g}}$ (Error or residual)
3. Compute $\delta\mathbf{f}_1 = \mathbf{H}'\delta\mathbf{g}$ (Backprojection of error)
4. Compute $\delta\mathbf{f}_2 = -\mathbf{D}'\mathbf{D}\widehat{\mathbf{f}}$ (Correction due to regularization)
5. Update $\widehat{\mathbf{f}}^{(k+1)} = \widehat{\mathbf{f}}^{(k)} + [\delta\mathbf{f}_1 + \delta\mathbf{f}_2]$



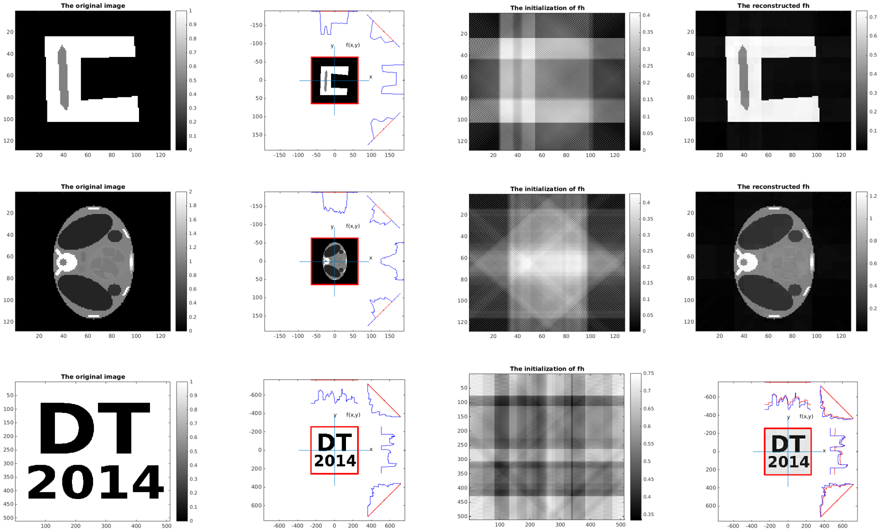
- Steps 1 and 3 need great computational cost and have been implemented on GPU.

Multi-Resolution Implementation

$$\begin{array}{lllc} \text{Sacle 1:} & \text{black} & \mathbf{g}^{(1)} = \mathbf{H}^{(1)}\mathbf{f}^{(1)} & (\; N \times N \;) \\ \text{Sacle 2:} & \text{green} & \mathbf{g}^{(2)} = \mathbf{H}^{(2)}\mathbf{f}^{(2)} & (N/2 \times N/2) \\ \text{Sacle 3:} & \text{red} & \mathbf{g}^{(3)} = \mathbf{H}^{(3)}\mathbf{f}^{(3)} & (N/4 \times N/4) \end{array}$$



Results with 4 projection



Original Projections Initialization Final result

Conclusions

- Limited angle Computed Tomography is a very ill-posed Inverse problem
- Analytical methods have many limitations
- Algebraic methods push further these limitations
- Deterministic Regularization methods push still further the limitations of ill-conditioning.
- Probabilistic and in particular the Bayesian approach has many potentials
- Hierarchical prior model with hidden variables are very powerful tools for Bayesian approach to inverse problems.
- Gauss-Markov-Potts models for images incorporating hidden regions and contours
- Main Bayesian computation tools: JMAP, MCMC and VBA
- Application in different imaging system (X ray CT, Microwaves, PET, Ultrasound, Optical Diffusion Tomography (ODT), Acoustic source localization,...)
- Current Projects: Efficient implementation in 2D and 3D cases