# Efficient Scalable Variational Bayesian Approximation methods for inverse problems

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

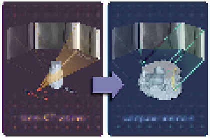
SIAM Uncertainty Quantification UQ16, EPFL, April 4-8, 2016,
Lausanne, Suisse

# Contents

1. Two inverse problems:
   - X ray Computed Tomography: Linear model
   - Microwave or Ultrasound Tomography: Bilinear model
2. Basic and Unsupervised 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 and Variational Bayesian Approximation (VBA)
6. Scalability and implementation issues for Big Data
7. Conclusions

# Computed Tomography: Seeing inside of a body

- $f(x, y)$ a section of a real 3D body $f(x, y, z)$
- $g_\phi(r)$ a line of observed radiography $g_\phi(r, z)$



- Forward model:
  Line integrals or Radon Transform

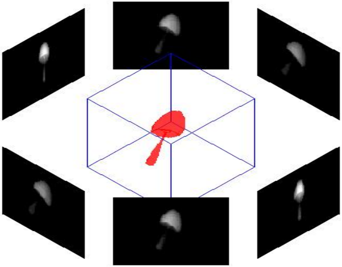
$$
\begin{aligned}
g_\phi(r) &= \int_{L_{r,\phi}} f(x, y)\, \mathrm{d}l + \epsilon_\phi(r) \\
&= \iint f(x, y)\, \delta(r - x\cos\phi - y\sin\phi)\, \mathrm{d}x\, \mathrm{d}y + \epsilon_\phi(r)
\end{aligned}
$$

- Inverse problem: Image reconstruction

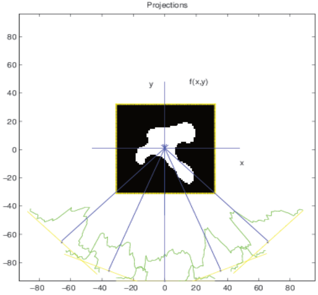
  Given the forward model $\mathcal{H}$ (Radon Transform) and
  a set of data $g_{\phi_i}(r), i = 1, \cdots, M$
  find $f(x, y)$

# 2D and 3D Computed Tomography

3D



2D

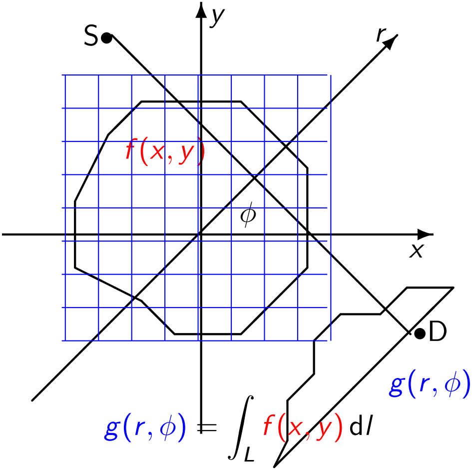


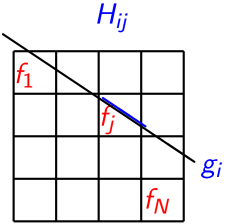

$$g_\phi(r_1,r_2)=\int_{\mathcal{L}_{r_1,r_2,\phi}} f(x,y,z)\,\mathrm{d}l \qquad g_\phi(r)=\int_{\mathcal{L}_{r,\phi}} f(x,y)\,\mathrm{d}l$$

Forward probelm: $f(x,y)$ or $f(x,y,z) \longrightarrow g_\phi(r)$ or $g_\phi(r_1,r_2)$

Inverse problem: $g_\phi(r)$ or $g_\phi(r_1,r_2) \longrightarrow f(x,y)$ or $f(x,y,z)$

# Algebraic methods: Discretization




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

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

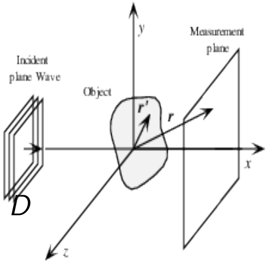
$$b_j(x,y) = \begin{cases} 1 & \text{if } (x,y) \in \text{ pixel } j \\ 0 & \text{else} \end{cases}$$

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

- $\mathbf{H}$ is huge dimensional: 2D: $10^6 \times 10^6$, 3D: $10^9 \times 10^9$.
- $\mathbf{Hf}$ corresponds to forward projection
- $\mathbf{H}^t\mathbf{g}$ corresponds to Back projection (BP)

# Microwave or ultrasound imaging

Measures: diffracted wave by the object $g(\mathbf{r}_i)$
Unknown quantity: $f(\mathbf{r}) = k_0^2(n^2(\mathbf{r}) - 1)$
Intermediate quantity : $\phi(\mathbf{r})$

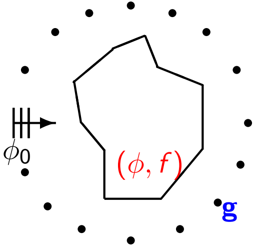



$$g(\mathbf{r}_i) = \iint_D G_m(\mathbf{r}_i, \mathbf{r}')\phi(\mathbf{r}')\, f(\mathbf{r}')\, \mathrm{d}\mathbf{r}', \;\; \mathbf{r}_i \in S$$

$$\phi(\mathbf{r}) = \phi_0(\mathbf{r}) + \iint_D G_o(\mathbf{r}, \mathbf{r}')\phi(\mathbf{r}')\, f(\mathbf{r}')\, \mathrm{d}\mathbf{r}', \;\; \mathbf{r} \in D$$

**Born approximation** $(\phi(\mathbf{r}') \simeq \phi_0(\mathbf{r}'))$ **):**

$$g(\mathbf{r}_i) = \iint_D G_m(\mathbf{r}_i, \mathbf{r}')\phi_0(\mathbf{r}')\, f(\mathbf{r}')\, \mathrm{d}\mathbf{r}', \;\; \mathbf{r}_i \in S$$




**Discretization:**

$$\left\{\begin{array}{l} \mathbf{g} = \mathbf{G}_m \mathbf{F} \boldsymbol{\phi} \\ \boldsymbol{\phi} = \boldsymbol{\phi}_0 + \mathbf{G}_o \mathbf{F} \boldsymbol{\phi} \end{array}\right. \longrightarrow \left\{\begin{array}{l} \mathbf{g} = \mathbf{H}(\mathbf{f}) \\ \text{with} \quad \mathbf{F} = \mathrm{diag}(\mathbf{f}) \\ \mathbf{H}(\mathbf{f}) = \mathbf{G}_m \mathbf{F} (\mathbf{I} - \mathbf{G}_o \mathbf{F})^{-1} \boldsymbol{\phi}_0 \end{array}\right.$$

# Microwave or ultrasound imaging: Bilinear model

Nonlinear model:

$$g(\mathbf{r}_i) = \iint_D G_m(\mathbf{r}_i, \mathbf{r}')\phi(\mathbf{r}')\, f(\mathbf{r}')\, \mathrm{d}\mathbf{r}', \;\; \mathbf{r}_i \in S$$

$$\phi(\mathbf{r}) = \phi_0(\mathbf{r}) + \iint_D G_o(\mathbf{r}, \mathbf{r}')\phi(\mathbf{r}')\, f(\mathbf{r}')\, \mathrm{d}\mathbf{r}', \;\; \mathbf{r} \in D$$

Bilinear model: $w(\mathbf{r}') = \phi(\mathbf{r}')\, f(\mathbf{r}')$

$$g(\mathbf{r}_i) = \iint_D G_m(\mathbf{r}_i, \mathbf{r}')w(\mathbf{r}')\, \mathrm{d}\mathbf{r}', \;\; \mathbf{r}_i \in S$$

$$\phi(\mathbf{r}) = \phi_0(\mathbf{r}) + \iint_D G_o(\mathbf{r}, \mathbf{r}')w(\mathbf{r}')\, \mathrm{d}\mathbf{r}', \;\; \mathbf{r} \in D$$

$$w(\mathbf{r}) = f(\mathbf{r})\phi_0(\mathbf{r}) + \iint_D G_o(\mathbf{r}, \mathbf{r}')w(\mathbf{r}')\, \mathrm{d}\mathbf{r}', \;\; \mathbf{r} \in D$$

**Discretization:** $\mathbf{g} = \mathbf{G}_m \mathbf{w} + \boldsymbol{\epsilon}, \quad \mathbf{w} = \boldsymbol{\phi} \,.\, \mathbf{f}$

- Constrast $\mathbf{f}$ - Field $\phi$: $\quad \boldsymbol{\phi} = \boldsymbol{\phi}_0 + \mathbf{G}_o \mathbf{w} + \boldsymbol{\xi}$
- Constrast $\mathbf{f}$ - Source $\mathbf{w}$ : $\quad \mathbf{w} = \mathbf{f} \,.\, \boldsymbol{\phi}_0 + \mathbf{G}_o \mathbf{w} + \boldsymbol{\xi}$

# Bayesian approach for linear model

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

- Observation model $\mathcal{M}$ + Information on the noise $\boldsymbol{\epsilon}$:
$$p(\mathbf{g}|\mathbf{f}, \theta_1; \mathcal{M}) = p_{\boldsymbol{\epsilon}}(\mathbf{g} - \mathbf{H}\mathbf{f}|\theta_1)$$
- A priori information $\qquad p(\mathbf{f}|\theta_2; \mathcal{M})$
- Basic Bayes :
$$p(\mathbf{f}|\mathbf{g}, \theta_1, \theta_2; \mathcal{M}) = \frac{p(\mathbf{g}|\mathbf{f}, \theta_1; \mathcal{M})\, p(\mathbf{f}|\theta_2; \mathcal{M})}{p(\mathbf{g}|\theta_1, \theta_2; \mathcal{M})}$$
- Unsupervised:
$$p(\mathbf{f}, \boldsymbol{\theta}|\mathbf{g}, \boldsymbol{\alpha}_0) = \frac{p(\mathbf{g}|\mathbf{f}, \theta_1)\, p(\mathbf{f}|\theta_2)\, p(\boldsymbol{\theta}|\boldsymbol{\alpha}_0)}{p(\mathbf{g}|\boldsymbol{\alpha}_0)}, \quad \boldsymbol{\theta} = (\theta_1, \theta_2)$$
- Hierarchical prior models:
$$p(\mathbf{f}, \mathbf{z}, \boldsymbol{\theta}|\mathbf{g}, \boldsymbol{\alpha}_0) = \frac{p(\mathbf{g}|\mathbf{f}, \theta_1)\, p(\mathbf{f}|\mathbf{z}, \theta_2)\, p(\mathbf{z}|\theta_3)\, p(\boldsymbol{\theta}|\boldsymbol{\alpha}_0)}{p(\mathbf{g}|\boldsymbol{\alpha}_0)}, \quad \boldsymbol{\theta} = (\theta_1, \theta_2, \theta_3)$$

# Bayesian approach for bilinear model

$$\mathcal{M}: \quad \mathbf{g} = \mathbf{G}_m \mathbf{w} + \epsilon, \quad \mathbf{w} = \mathbf{f}.\phi_0 + \mathbf{G}_o \mathbf{w} + \boldsymbol{\xi}, \quad \mathbf{w} = \phi \,.\, \mathbf{f}$$

$$\mathcal{M}: \quad \mathbf{g} = \mathbf{G}_m \mathbf{w} + \epsilon, \quad \mathbf{w} = (\mathbf{I} - \mathbf{G}_o)^{-1} (\boldsymbol{\Phi}_0 \mathbf{f} + \boldsymbol{\xi}), \quad \mathbf{w} = \phi \,.\, \mathbf{f}$$

- Basic Bayes:

$$p(\mathbf{f}, \mathbf{w} | \mathbf{g}, \boldsymbol{\theta}) = \frac{p(\mathbf{g}|\mathbf{w}, , \theta_1)\, p(\mathbf{w}|\mathbf{f}, , \theta_2)\, p(\mathbf{f}|, \theta_3)}{p(\mathbf{g}|\boldsymbol{\theta})} \propto p(\mathbf{g}|\mathbf{w}, \theta_1)\, p(\mathbf{w}|\mathbf{f}, \theta_2)\, p(\mathbf{f}|\theta_3)$$

- Unsupervised:

$$p(\mathbf{f}, \mathbf{w}, \boldsymbol{\theta} | \mathbf{g}, \boldsymbol{\alpha}_0) \propto p(\mathbf{g}|\mathbf{w}, \theta_1)\, p(\mathbf{f}|\mathbf{w}, \theta_2) p(\mathbf{f}|\theta_3)\, p(\boldsymbol{\theta}|\alpha_0), \;\; \boldsymbol{\theta} = (\theta_1, \theta_2, \theta_3)$$

- Hierarchical prior models:

$$p(\mathbf{f}, \mathbf{w}, \mathbf{z}, \boldsymbol{\theta} | \mathbf{g}, \boldsymbol{\alpha}_0) \propto p(\mathbf{g}|\mathbf{w}, \theta_1)\, p(\mathbf{w}|\mathbf{f}, \theta_2)\, p(\mathbf{f}|\mathbf{z}, \theta_3)\, p(\mathbf{z}|\theta_4)\, p(\boldsymbol{\theta}|\boldsymbol{\alpha}_0)$$

# Two main steps in Bayesian inference

1- Assigning priors:

- Simple priors $p(\mathbf{f})$: Gaussian, Gamma, Beta, Generalized Gaussian (Laplace), Student-t, ...
- Hierarchical $p(\mathbf{f}|\mathbf{z})\, p(\mathbf{z})$:
  - Finite Mixture models
  - Infinite Gaussian Scaled Mixture IGSM
  - Gauss-Markov-Potts

2- Doing efficiently computations

- JMAP: Alternate optimization
- Marginalization via EM
- MCMC
- Approximate Bayesian Computation (ABC)
- Variational Bayesian Approximation (VBA)

# Sparsity enforcing models

- 3 classes of models:
  1. Generalized Gaussian (Laplace)
  2. Mixture models
  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

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

- Generalization

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

$$\begin{cases} p(\mathbf{f}|\mathbf{z}) & = \prod_j p(f_j|z_j) = \prod_j \mathcal{N}(f_j|0,1/z_j) \propto \exp\left[-\frac{1}{2}\sum_j z_j f_j^2\right] \\ p(\mathbf{z}|\alpha,\beta) & = \prod_j \mathcal{G}(z_j|\alpha,\beta) \propto \prod_j z_j^{(\alpha-1)} \exp\left[-\beta z_j\right] \\ & \propto \exp\left[\sum_j (\alpha-1)\ln z_j - \beta z_j\right] \\ p(\mathbf{f},\mathbf{z}|\alpha,\beta) & \propto \exp\left[-\frac{1}{2}\sum_j z_j f_j^2 + (\alpha-1)\ln z_j - \beta z_j\right] \end{cases} \quad (3)$$

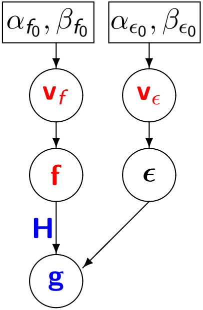
# Non stationary noise and sparsity enforcing model

– Non stationary noise:

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

– Student-t prior model and its equivalent IGSM :

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




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

$$\begin{cases} p(\mathbf{v}_\epsilon) = \prod_i \mathcal{IG}(v_{\epsilon_i}|\alpha_{\epsilon_0}, \beta_{\epsilon_0}) \\ p(\mathbf{v}_f) = \prod_i \mathcal{IG}(v_{f_j}|\alpha_{f_0}, \beta_{f_0}) \end{cases}$$

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

Objective: Infer $(\mathbf{f}, \mathbf{v}_\epsilon, \mathbf{v}_f)$

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

# Variational Bayesian Approximation

Depending on cases, we have to handle $p(\mathbf{f}, \boldsymbol{\theta}|\mathbf{g})$, $p(\mathbf{f}, \mathbf{z}, \boldsymbol{\theta}|\mathbf{g})$, $p(\mathbf{f}, \mathbf{w}, \boldsymbol{\theta}|\mathbf{g})$ or $p(\mathbf{f}, \mathbf{w}, \mathbf{z}, \boldsymbol{\theta}|\mathbf{g})$. Let consider the simplest case:

- Approximate $p(\mathbf{f}, \boldsymbol{\theta}|\mathbf{g})$ by $q(\mathbf{f}, \boldsymbol{\theta}|\mathbf{g}) = q_1(\mathbf{f}|\mathbf{g})\, q_2(\boldsymbol{\theta}|\mathbf{g})$ and then continue computations.
- Criterion $\mathrm{KL}(q(\mathbf{f}, \boldsymbol{\theta}|\mathbf{g}) : p(\mathbf{f}, \boldsymbol{\theta}|\mathbf{g}))$
- $\mathrm{KL}(q : p) = \int\int q \ln q/p = \int\int q_1 q_2 \ln \frac{q_1 q_2}{p} = \int q_1 \ln q_1 + \int q_2 \ln q_2 - \int\int q \ln p = -H(q_1) - H(q_2) - < \ln p >_q$
- Iterative algorithm $q_1 \longrightarrow q_2 \longrightarrow q_1 \longrightarrow q_2, \cdots$

$$\left\{ \begin{array}{ll} q_1(\mathbf{f}) & \propto \exp\left[\left\langle \ln p(\mathbf{g}, \mathbf{f}, \boldsymbol{\theta}; \mathcal{M}) \right\rangle_{q_2(\boldsymbol{\theta})}\right] \\ q_2(\boldsymbol{\theta}) & \propto \exp\left[\left\langle \ln p(\mathbf{g}, \mathbf{f}, \boldsymbol{\theta}; \mathcal{M}) \right\rangle_{q_1(\mathbf{f})}\right] \end{array} \right. \tag{4}$$

$p(\mathbf{f}, \boldsymbol{\theta}|\mathbf{g}) \longrightarrow$ **Variational Bayesian Approximation** $\longrightarrow \widehat{q}_1(\mathbf{f}) \longrightarrow \widehat{\mathbf{f}}$ ; $\longrightarrow \widehat{q}_2(\boldsymbol{\theta}) \longrightarrow \widehat{\boldsymbol{\theta}}$

# JMAP, Marginalization, Sampling and exploration, VBA

- JMAP:

$$\boxed{\begin{array}{c} p(\mathbf{f},\boldsymbol{\theta}|\mathbf{g}) \\ \text{optimization} \end{array}} \begin{array}{l} \longrightarrow \widehat{\mathbf{f}} \\ \longrightarrow \widehat{\boldsymbol{\theta}} \end{array} \quad \begin{array}{c} \text{Alternate} \\ \text{Optimization} \end{array} \quad \left\{ \begin{array}{l} \mathbf{f} = \arg\max_{\mathbf{f}} \{p(\mathbf{f}|\boldsymbol{\theta},\mathbf{g})\} \\ \boldsymbol{\theta} = \arg\max_{\boldsymbol{\theta}} \{p(\boldsymbol{\theta}|\mathbf{f}\mathbf{g})\} \end{array} \right.$$

- Marginalization

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

Joint Posterior — Marginalize over $\mathbf{f}$

- Sampling and Exploration
  - Gibbs sampling: $\mathbf{f} \sim p(\mathbf{f}|\boldsymbol{\theta},\mathbf{g}) \rightarrow \boldsymbol{\theta} \sim p(\boldsymbol{\theta}|\mathbf{f}\mathbf{g})$
  - Other sampling methods: IS, MH, Slice sampling,...
- Variational Bayesian Approximation

$$\boxed{p(\mathbf{f},\boldsymbol{\theta}|\mathbf{g})} \longrightarrow \boxed{\begin{array}{c}\text{Variational}\\ \text{Bayesian}\\ \text{Approximation}\end{array}} \begin{array}{l} \longrightarrow q_1(\mathbf{f}) \longrightarrow \widehat{\mathbf{f}} \\ \\ \longrightarrow q_2(\boldsymbol{\theta}) \longrightarrow \widehat{\boldsymbol{\theta}} \end{array}$$

# Variational Bayesian Approximation

- Approximate $p(\mathbf{f}, \boldsymbol{\theta}|\mathbf{g})$ by $q(\mathbf{f}, \boldsymbol{\theta}) = q_1(\mathbf{f})\, q_2(\boldsymbol{\theta})$ and then use them for any inferences on $\mathbf{f}$ and $\boldsymbol{\theta}$ respectively.
- Criterion $\mathrm{KL}(q(\mathbf{f}, \boldsymbol{\theta}|\mathbf{g}) : p(\mathbf{f}, \boldsymbol{\theta}|\mathbf{g}))$
  $\mathrm{KL}(q : p) = \int\int q \ln \frac{q}{p} = \int\int q_1 q_2 \ln \frac{q_1 q_2}{p}$
- Iterative algorithm $q_1 \longrightarrow q_2 \longrightarrow q_1 \longrightarrow q_2, \cdots$

$$\begin{cases} \widehat{q}_1(\mathbf{f}) & \propto \exp\left[\langle \ln p(\mathbf{g}, \mathbf{f}, \boldsymbol{\theta}; \mathcal{M}) \rangle_{\widehat{q}_2(\boldsymbol{\theta})}\right] \\ \widehat{q}_2(\boldsymbol{\theta}) & \propto \exp\left[\langle \ln p(\mathbf{g}, \mathbf{f}, \boldsymbol{\theta}; \mathcal{M}) \rangle_{\widehat{q}_1(\mathbf{f})}\right] \end{cases} \tag{5}$$

$p(\mathbf{f}, \boldsymbol{\theta}|\mathbf{g}) \longrightarrow$ [Variational Bayesian Approximation] $\longrightarrow q_1(\mathbf{f}) \longrightarrow \widehat{\mathbf{f}}$; $\longrightarrow q_2(\boldsymbol{\theta}) \longrightarrow \widehat{\boldsymbol{\theta}}$

# BVA: Choice of family of laws $q_1$ and $q_2$

- Case 1 : $\longrightarrow$ Joint MAP

$$\left\{\begin{array}{ll}\widehat{q}_1(\mathbf{f}|\widetilde{\mathbf{f}}) & = \delta(\mathbf{f}-\widetilde{\mathbf{f}}) \\ \widehat{q}_2(\boldsymbol{\theta}|\widetilde{\boldsymbol{\theta}}) & = \delta(\boldsymbol{\theta}-\widetilde{\boldsymbol{\theta}})\end{array}\right. \longrightarrow \left\{\begin{array}{l}\widetilde{\mathbf{f}} = \arg\max_{\mathbf{f}} \left\{p(\mathbf{f},\widetilde{\boldsymbol{\theta}}|\mathbf{g};\mathcal{M})\right\} \\ \widetilde{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} \left\{p(\widetilde{\mathbf{f}},\boldsymbol{\theta}|\mathbf{g};\mathcal{M})\right\}\end{array}\right. \quad (6)$$

- Case 2 : $\longrightarrow$ EM

$$\left\{\begin{array}{ll}\widehat{q}_1(\mathbf{f}) & \propto p(\mathbf{f}|\boldsymbol{\theta},\mathbf{g}) \\ \widehat{q}_2(\boldsymbol{\theta}|\widetilde{\boldsymbol{\theta}}) & = \delta(\boldsymbol{\theta}-\widetilde{\boldsymbol{\theta}})\end{array}\right. \longrightarrow \left\{\begin{array}{ll}Q(\boldsymbol{\theta},\widetilde{\boldsymbol{\theta}}) = & \langle \ln p(\mathbf{f},\boldsymbol{\theta}|\mathbf{g};\mathcal{M})\rangle_{q_1(\mathbf{f}|\widetilde{\boldsymbol{\theta}})} \\ \widetilde{\boldsymbol{\theta}} & = \arg\max_{\boldsymbol{\theta}} \left\{Q(\boldsymbol{\theta},\widetilde{\boldsymbol{\theta}})\right\}\end{array}\right. \quad (7)$$

- Appropriate choice for inverse problems

$$\left\{\begin{array}{ll}\widehat{q}_1(\mathbf{f}) & \propto p(\mathbf{f}|\widetilde{\boldsymbol{\theta}},\mathbf{g};\mathcal{M}) \\ \widehat{q}_2(\boldsymbol{\theta}) & \propto p(\boldsymbol{\theta}|\widetilde{\mathbf{f}},\mathbf{g};\mathcal{M})\end{array}\right. \longrightarrow \left\{\begin{array}{l}\text{Accounts for the uncertainties of} \\ \widehat{\boldsymbol{\theta}} \text{ for } \widehat{\mathbf{f}} \text{ and vise versa.}\end{array}\right. \quad (8)$$

Exponential families, Conjugate priors

# JMAP, EM and VBA

JMAP Alternate optimization Algorithm:

$$\theta^{(0)} \longrightarrow \widetilde{\theta} \longrightarrow \boxed{\widetilde{\mathbf{f}} = \arg\max_{\mathbf{f}} \left\{ p(\mathbf{f}, \widetilde{\theta} | \mathbf{g}) \right\}} \longrightarrow \widetilde{\mathbf{f}} \longrightarrow \widehat{\mathbf{f}}$$

$$\uparrow \qquad\qquad\qquad\qquad\qquad \downarrow$$

$$\widehat{\theta} \longleftarrow \widetilde{\theta} \longleftarrow \boxed{\widetilde{\theta} = \arg\max_{\theta} \left\{ p(\widetilde{\mathbf{f}}, \theta | \mathbf{g}) \right\}} \longleftarrow \widetilde{\mathbf{f}}$$

EM:

$$\theta^{(0)} \longrightarrow \widetilde{\theta} \longrightarrow \boxed{q_1(\mathbf{f}) = p(\mathbf{f} | \widetilde{\theta}, \mathbf{g})} \longrightarrow q_1(\mathbf{f}) \longrightarrow \widehat{\mathbf{f}}$$

$$\uparrow \qquad\qquad\qquad\qquad\qquad \downarrow$$

$$\widehat{\theta} \longleftarrow \widetilde{\theta} \longleftarrow \boxed{\begin{array}{c} Q(\theta, \widetilde{\theta}) = \langle \ln p(\mathbf{f}, \theta | \mathbf{g}) \rangle_{q_1(\mathbf{f})} \\ \widetilde{\theta} = \arg\max_{\theta} \left\{ Q(\theta, \widetilde{\theta}) \right\} \end{array}} \longleftarrow q_1(\mathbf{f})$$

VBA:

$$\theta^{(0)} \longrightarrow q_2(\theta) \longrightarrow \boxed{q_1(\mathbf{f}) \propto \exp\left[ \langle \ln p(\mathbf{f}, \theta | \mathbf{g}) \rangle_{q_2(\theta)} \right]} \longrightarrow q_1(\mathbf{f}) \longrightarrow \widehat{\mathbf{f}}$$

$$\uparrow \qquad\qquad\qquad\qquad\qquad \downarrow$$

$$\widehat{\theta} \longleftarrow q_2(\theta) \longleftarrow \boxed{q_2(\theta) \propto \exp\left[ \langle \ln p(\mathbf{f}, \theta | \mathbf{g}) \rangle_{q_1(\mathbf{f})} \right]} \longleftarrow q_1(\mathbf{f})$$

# Variational Bayesian Approximation

Depending on cases, we have to handle $p(\mathbf{f}, \boldsymbol{\theta}|\mathbf{g})$, $p(\mathbf{f}, \mathbf{z}, \boldsymbol{\theta}|\mathbf{g})$, $p(\mathbf{w}, \boldsymbol{\phi}, \boldsymbol{\theta}|\mathbf{g})$ or $p(\mathbf{w}, \boldsymbol{\phi}, \mathbf{z}, \boldsymbol{\theta}|\mathbf{g})$. Let consider the simplest case:

- Approximate $p(\mathbf{f}, \boldsymbol{\theta}|\mathbf{g})$ by $q(\mathbf{f}, \boldsymbol{\theta}|\mathbf{g}) = q_1(\mathbf{f}|\mathbf{g})\, q_2(\boldsymbol{\theta}|\mathbf{g})$ and then continue computations.
- Criterion $\mathrm{KL}(q(\mathbf{f}, \boldsymbol{\theta}|\mathbf{g}) : p(\mathbf{f}, \boldsymbol{\theta}|\mathbf{g}))$
- $\mathrm{KL}(q : p) = \int\int q \ln q/p = \int\int q_1 q_2 \ln \frac{q_1 q_2}{p} = \int q_1 \ln q_1 + \int q_2 \ln q_2 - \int\int q \ln p = -H(q_1) - H(q_2) - < \ln p >_q$
- Iterative algorithm $q_1 \longrightarrow q_2 \longrightarrow q_1 \longrightarrow q_2, \cdots$

$$\left\{ \begin{array}{lll} q_1(\mathbf{f}) & \propto \exp\left[ \langle \ln p(\mathbf{g}, \mathbf{f}, \boldsymbol{\theta}; \mathcal{M}) \rangle_{q_2(\boldsymbol{\theta})} \right] \\ q_2(\boldsymbol{\theta}) & \propto \exp\left[ \langle \ln p(\mathbf{g}, \mathbf{f}, \boldsymbol{\theta}; \mathcal{M}) \rangle_{q_1(\mathbf{f})} \right] \end{array} \right. \tag{9}$$

$p(\mathbf{f}, \boldsymbol{\theta}|\mathbf{g}) \longrightarrow$ Variational Bayesian Approximation $\longrightarrow \widehat{q}_1(\mathbf{f}) \longrightarrow \widehat{\mathbf{f}}$

$\longrightarrow \widehat{q}_2(\boldsymbol{\theta}) \longrightarrow \widehat{\boldsymbol{\theta}}$

# Main questions related to uncertainty quantification (UQ)

We approximate $p(\mathbf{f}, \boldsymbol{\theta}|\mathbf{g})$ by:

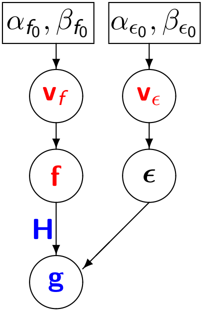
- $q_1(\mathbf{f}) q_2(\boldsymbol{\theta})$, partial separation
- $\prod_j q_{1j}(f_j) \prod_k q_{2k}(\theta_k)$ full separaton
- or a combination

$$\boxed{p(\mathbf{f}, \boldsymbol{\theta}|\mathbf{g})} \rightarrow \boxed{\begin{array}{c}\text{Variational}\\ \text{Bayesian}\\ \text{Approximation}\end{array}} \begin{array}{l} \rightarrow q_1(\mathbf{f}) \rightarrow \mathrm{E}_{q_1}\{\mathbf{f}\}, \mathrm{Var}_{q_1}\{\mathbf{f}\}, \\ \\ \rightarrow q_2(\boldsymbol{\theta}) \rightarrow \mathrm{E}_{q_2}\{\boldsymbol{\theta}\}, \mathrm{Var}_{q_2}\{\boldsymbol{\theta}\} \end{array}$$

Now, the questions arise:

1. $\mathrm{E}_p\{\mathbf{f}\} = \mathrm{E}_{q_1}\{\mathbf{f}\}$ and $\mathrm{E}_p\{\boldsymbol{\theta}\} = \mathrm{E}_{q_1}\{\boldsymbol{\theta}\}$ ?
2. $\mathrm{Var}_p\{\mathbf{f}\} \geq \mathrm{Var}_{q_1}\{\mathbf{f}\}$ and $\mathrm{Var}_p\{\boldsymbol{\theta}\} \geq \mathrm{Var}_{q_1}\{\boldsymbol{\theta}\}$?
3. higher moments ?
4. Other properties ?

# Direct sparsity enforcing model




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

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

$$\boldsymbol{\theta}_1 = \mathbf{v}_\epsilon,\ \boldsymbol{\theta}_2 = \mathbf{v}_f$$

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

$$\begin{cases} p(\mathbf{v}_\epsilon) = \prod_i \mathcal{IG}(v_{\epsilon_i}|\alpha_{\epsilon_0}, \beta_{\epsilon_0}) \\ p(\mathbf{v}_f) = \prod_i \mathcal{IG}(v_{f_j}|\alpha_{f_0}, \beta_{f_0}) \end{cases}$$

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

Objective: Infer $(\mathbf{f}, \mathbf{v}_\epsilon, \mathbf{v}_f)$

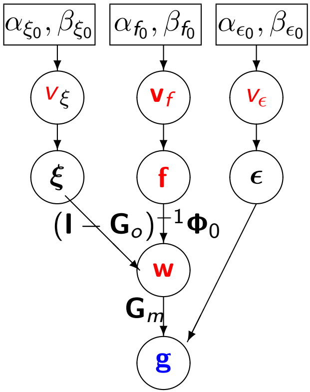
– VBA: Approximate $p(\mathbf{f}, \mathbf{v}_\epsilon, \mathbf{v}_f|\mathbf{g})$
by $q_1(\mathbf{f})\, q_2(\mathbf{v}_\epsilon)\, q_3(\mathbf{v}_f)$

# Direct sparsity enforcing model: Bilinear case

$$\mathbf{g} = \mathbf{G}_m \mathbf{w} + \epsilon,$$
$$\mathbf{w} = (I - \mathbf{G}_0)^{-1}(\boldsymbol{\Phi}_0 \mathbf{f} + \boldsymbol{\xi}), \quad \mathbf{f} \text{ sparse}$$

$$p(\mathbf{f}, \mathbf{w}, \boldsymbol{\theta} | \mathbf{g}) \propto p(\mathbf{g}|\mathbf{w}, \theta_1)\, p(\mathbf{w}|\mathbf{f}, \theta_2)\, p(\mathbf{f}|\boldsymbol{\theta}_3)\, p(\boldsymbol{\theta})$$

$$\theta_1 = v_\epsilon, \; \theta_2 = v_\xi, \; \boldsymbol{\theta}_3 = \mathbf{v}_f$$




$$\begin{cases} p(\mathbf{g}|\mathbf{w}, v_\epsilon) = \mathcal{N}(\mathbf{g}|\mathbf{G}_m\mathbf{w}, v_\epsilon \mathbf{I}) \\ p(\mathbf{w}|\mathbf{f}, v_\xi) = \mathcal{N}(\mathbf{w}|(\mathbf{I} - \mathbf{G}_0)^{-1}\Phi_0 \mathbf{f}, v_\xi \mathbf{I}) \\ p(\mathbf{f}|\mathbf{v}_f) = \mathcal{N}(\mathbf{g}|0, \mathbf{V}_f), \quad \mathbf{V}_f = \operatorname{diag}[\mathbf{v}_f] \end{cases}$$

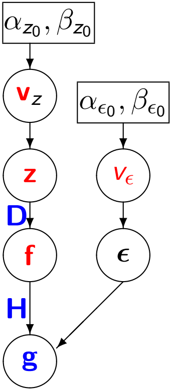
$$\begin{cases} p(v_\epsilon) = \mathcal{IG}(v_{\epsilon_i}|\alpha_{\epsilon_0}, \beta_{\epsilon_0}) \\ p(\mathbf{v}_f) = \prod_i \mathcal{IG}(v_{f_j}|\alpha_{f_0}, \beta_{f_0}) \\ p(v_\xi) = \mathcal{IG}(v_\xi|\alpha_{\xi_0}, \beta_{\xi_0}) \end{cases}$$

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

Objective: Infer $(\mathbf{f}, \mathbf{w}, \mathbf{v}_f, v_\epsilon, v_\xi)$

– VBA: Approximate $p(\mathbf{f}, \mathbf{w}, \mathbf{v}_f, v_\epsilon, v_\xi | \mathbf{g})$
by $q_1(\mathbf{f})\, q_2(\mathbf{w})\, q_3(\mathbf{v}_f)\, q_4(v_\epsilon)\, q_5(v_\xi)$

# Sparse model in a Transform domain 1




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

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

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

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

– JMAP:

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

Alternate optimization:

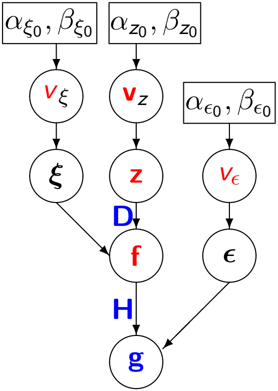
$$\begin{cases} \widehat{\mathbf{z}} = \arg\min_{\mathbf{z}} \{J(\mathbf{z})\} \text{ with:} \\ J(\mathbf{z}) = \frac{1}{2\hat{v}_\epsilon} \|\mathbf{g} - \mathbf{H}\mathbf{D}\mathbf{z}\|^2 + \|\mathbf{V}_z^{-1/2}\mathbf{z}\|^2 \\ \widehat{v}_{z_j} = \frac{\beta_{z_0} + \widehat{z}_j^2}{\alpha_{z_0}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

– 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




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

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

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

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

– JMAP:

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

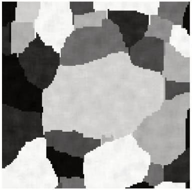
Alternate optimization.

– VBA: Approximate

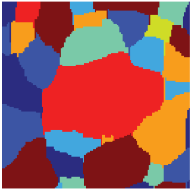
$p(\mathbf{f}, \mathbf{z}, v_\epsilon, \mathbf{v}_z, v_\xi|\mathbf{g})$ by $q_1(\mathbf{f})\, q_2(\mathbf{z})\, q_3(v_\epsilon)\, q_4(\mathbf{v}_z)\, q_5(v_\xi)$

Alternate optimization.

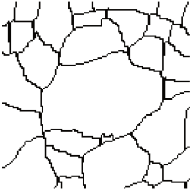
# Gauss-Markov-Potts prior models for images



$f(\mathbf{r})$



$z(\mathbf{r})$



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

$$p(f(\mathbf{r})|z(\mathbf{r}) = k, m_k, v_k) = \mathcal{N}(f(\mathbf{r})|m_k, v_k) \quad (10)$$

$$p(f(\mathbf{r})) = \sum_k P(z(\mathbf{r}) = k)\, \mathcal{N}(f(\mathbf{r})|m_k, v_k) \quad \text{Mixture of Gaussians} \quad (11)$$

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

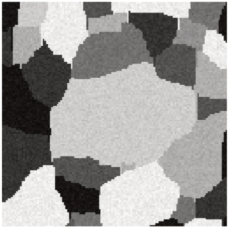
$$p(z(\mathbf{r})|z(\mathbf{r}'), \mathbf{r}' \in \mathcal{V}(\mathbf{r})) \propto \exp\left[\gamma \sum_{\mathbf{r}' \in \mathcal{V}(\mathbf{r})} \delta(z(\mathbf{r}) - z(\mathbf{r}'))\right] \quad (12)$$

$$p(\mathbf{z}) \propto \exp\left[\gamma \sum_{\mathbf{r} \in \mathcal{R}} \sum_{\mathbf{r}' \in \mathcal{V}(\mathbf{r})} \delta(z(\mathbf{r}) - z(\mathbf{r}'))\right] \quad (13)$$

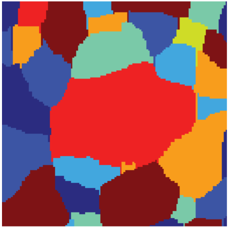
# Four different cases

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)

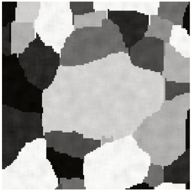


$f(\mathbf{r})$

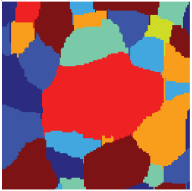


$z(\mathbf{r})$

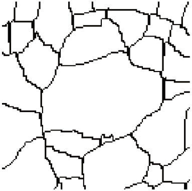
# Gauss-Markov-Potts prior models for images



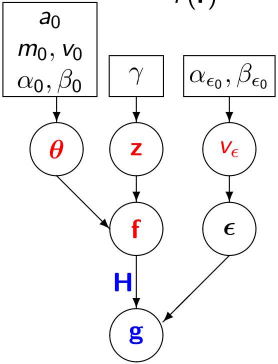
$f(\mathbf{r})$



$z(\mathbf{r})$



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




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

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

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

MCMC: Gibbs Sampling

VBA: Alternate optimization.

(14)

# Bayesian Computation and Algorithms

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

$$p(\mathbf{f}, \mathbf{z}, \boldsymbol{\theta}|\mathbf{g}) \propto p(\mathbf{g}|\mathbf{f}, \boldsymbol{\theta}_1)\ p(\mathbf{f}|\mathbf{z}, \boldsymbol{\theta}_2)\ p(\mathbf{z}|\boldsymbol{\theta}_3)\ p(\boldsymbol{\theta}) \tag{16}$$

- 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

$$q(\mathbf{f}, \mathbf{z}, \boldsymbol{\theta}|\mathbf{g}) = q_1(\mathbf{f})\ q_2(\mathbf{z})\ q_3(\boldsymbol{\theta}) \tag{17}$$

    and do any computations with these separable ones.

# MCMC based algorithm

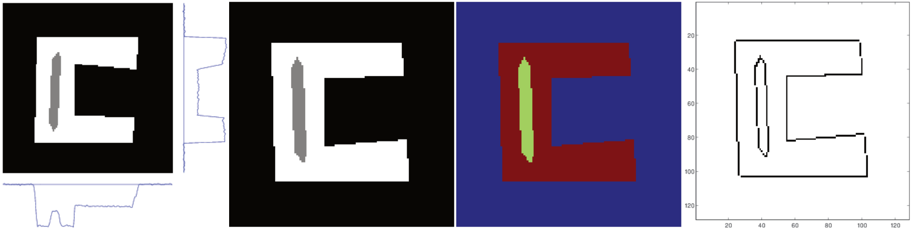
$$p(\mathbf{f}, \mathbf{z}, \boldsymbol{\theta}|\mathbf{g}) \propto p(\mathbf{g}|\mathbf{f}, \mathbf{z}, \boldsymbol{\theta}_1)\, p(\mathbf{f}|\mathbf{z}, \boldsymbol{\theta}_2)\, p(\mathbf{z}|\boldsymbol{\theta}_3)\, p(\boldsymbol{\theta})$$

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




| $\mathbf{g}\vert\mathbf{f}$ | $\mathbf{f}\vert\mathbf{z}$ | $\mathbf{z}$ | $\mathbf{c}$ |
|---|---|---|---|
| $\mathbf{g} = \mathbf{H}\mathbf{f} + \boldsymbol{\epsilon}$ | iid Gaussian | iid | $q(\mathbf{r}) \in \{0,1\}$ |
| $\mathbf{g}\vert\mathbf{f} \sim \mathcal{N}(\mathbf{H}\mathbf{f}, \sigma_\epsilon^2 \mathbf{I})$ | or | or | $1 - \delta(z(\mathbf{r}) - z(\mathbf{r}'))$ |
| Gaussian | Gauss-Markov | Potts | 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:

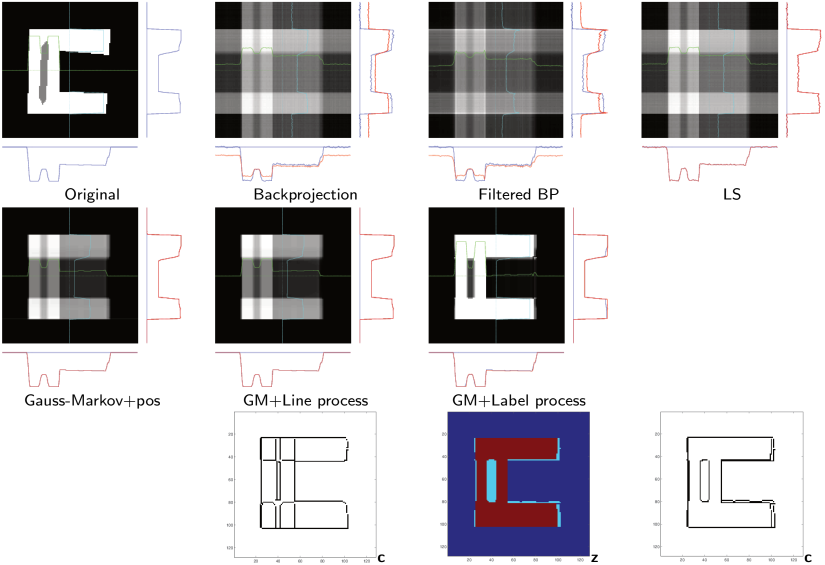
$$\widehat{\mathbf{f}} \sim p(\mathbf{f} | \widehat{\mathbf{z}}, \widehat{\boldsymbol{\theta}}, \mathbf{g}) \longrightarrow \widehat{\mathbf{z}} \sim p(\mathbf{z} | \widehat{\mathbf{f}}, \widehat{\boldsymbol{\theta}}, \mathbf{g}) \longrightarrow \widehat{\boldsymbol{\theta}} \sim (\boldsymbol{\theta}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

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

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

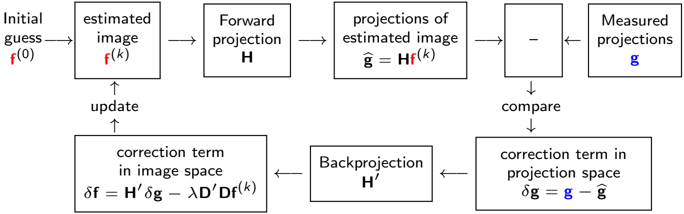
- So, for the simplest case, in each step, we have

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

# 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.

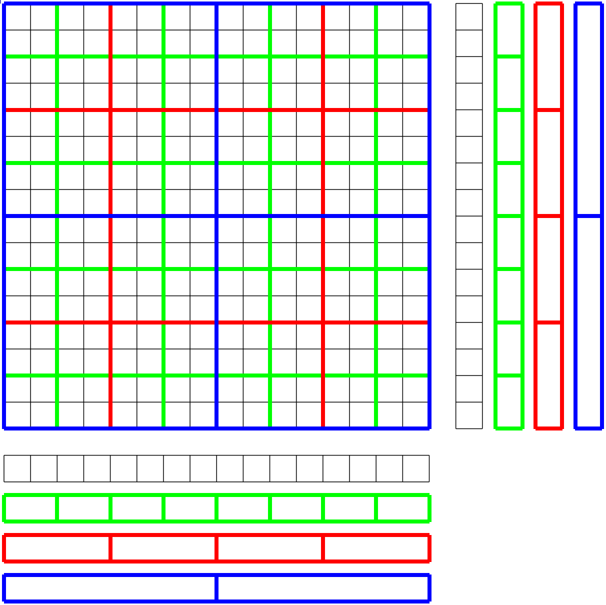
# Multi-Resolution Implementation

Scale 1: black:
$\mathbf{g}^{(1)} = \mathbf{H}^{(1)}\mathbf{f}^{(1)}$
$(\ N\ \times\ N\ )$

Scale 2: green:
$\mathbf{g}^{(2)} = \mathbf{H}^{(2)}\mathbf{f}^{(2)}$
$(N/2 \times N/2)$

Scale 3: red:
$\mathbf{g}^{(3)} = \mathbf{H}^{(3)}\mathbf{f}^{(3)}$
$(N/4 \times N/4)$

Scale 4: blue:
$\mathbf{g}^{(4)} = \mathbf{H}^{(4)}\mathbf{f}^{(4)}$
$(N/8 \times N/8)$



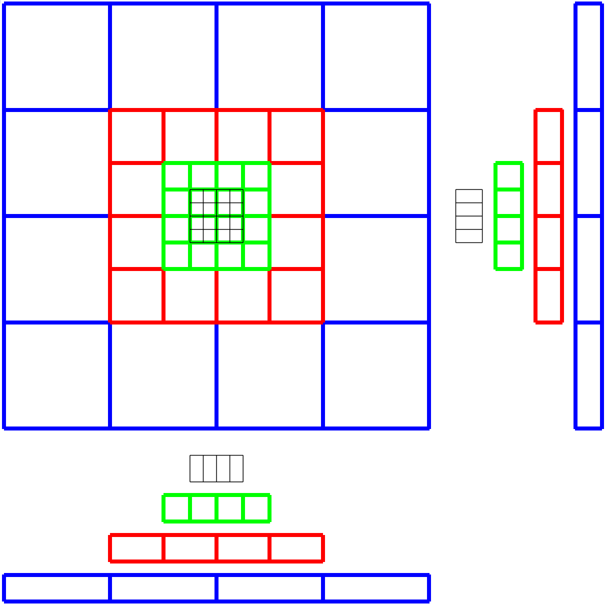
# Super-Resolution Implementation

Scale 1: blue:
$\mathbf{g}^{(1)} = \mathbf{H}^{(1)}\mathbf{f}^{(1)}$
$(\Delta x = 1, \Delta y = 1)$

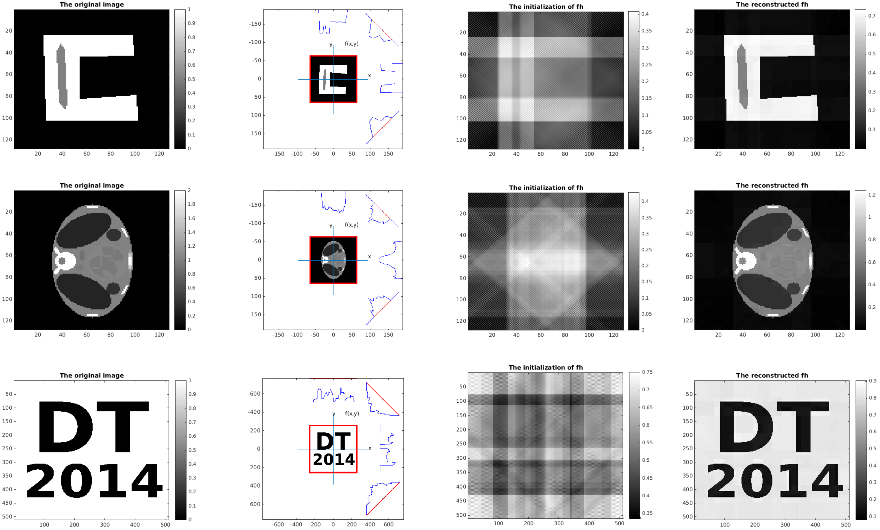
Scale 2: red:
$\mathbf{g}^{(2)} = \mathbf{H}^{(2)}\mathbf{f}^{(2)}$
$(\Delta x = 1/2, \Delta y = 1/2)$

Scale 3: green:
$\mathbf{g}^{(3)} = \mathbf{H}^{(3)}\mathbf{f}^{(3)}$
$(\Delta x = 1/4, \Delta y = 1/4)$

Scale 4: black:
$\mathbf{g}^{(4)} = \mathbf{H}^{(4)}\mathbf{f}^{(4)}$
$(\Delta x = 1/8, \Delta y = 1/8)$



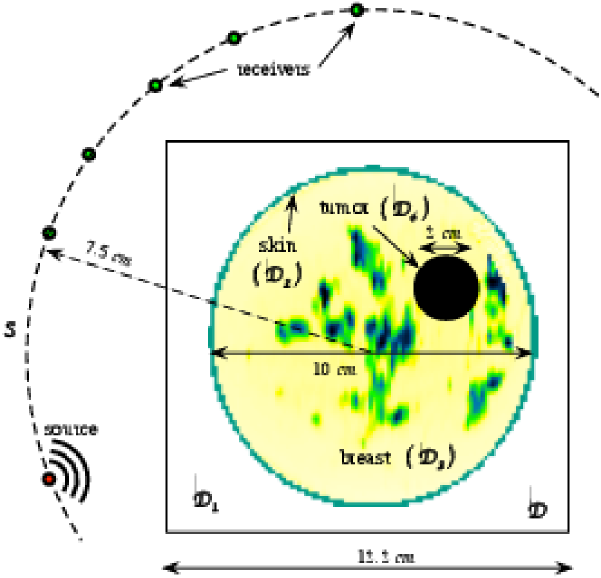
# Limited angle X ray Tomography




Original Projections Initialization Final result

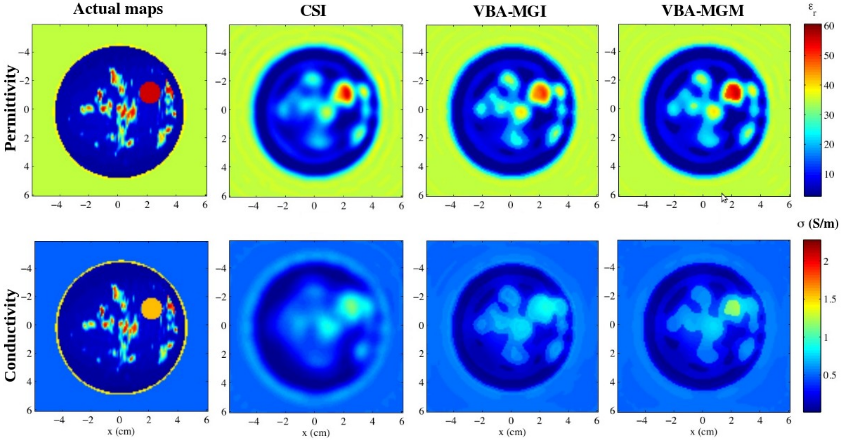
# Microwave Imaging for Breast Cancer detection

(L. Gharsalli et al.)



# Microwave Imaging for Breast Cancer detection

CSI: Contrast Source Inversion, VBA: Variational Bayesian Approach,
MGI: Independent Gaussian mixture, MGM: Gauss-Markov mixture



# Conclusions

- Bayesian approach with Hierarchical prior model with hidden variables are very powerful tools for inverse problems.
- We explored two classes of priors:
  - Generalized Student-t for sparse representation and
  - Gauss-Markov-Potts models for images incorporating hidden regions and contours
- The computational cost of all the sampling methods (MCMC and many others) are too high to be used in practical high dimensional applications.
- We explored VBA tools for effective approximate Bayesian computation.
- Application in different imaging system (3D X ray CT, Microwaves, PET, Ultrasound, Optical Diffusion Tomography (ODT), Acoustic source localization,...)
- Current Projects: Efficient implementation of different forward and adjoint operators as well as Bayesian computations in 2D and 3D cases on GPU