Bayesian Blind Deconvolution

Ali Mohammad-Djafari
Laboratoire des Signaux et Systèmes,
UMR8506 CNRS-SUPELEC-UNIV PARIS SUD 11
SUPELEC, 91192 Gif-sur-Yvette, France
http://lss.supelec.free.fr

Email: djafari@lss.supelec.fr
http://djafari.free.fr

Seminar at Amir Kabir University (Polytechnique de) Tehran,
December 9, 2014
Signal Deconvolution

\[ g(t) = \int f(\tau) h(t - \tau) \, d\tau + \epsilon(t) \]

**Forward:** \( f(t) \)  \( \rightarrow \) \( g(t) = h(t) \ast f(t) + \epsilon(t) \)

**Inverse:** \( \hat{f}(t) \)  \( \leftarrow \) \( g(t) \)
Image Restoration

Forward model: 2D Convolution

\[ g(x, y) = \int \int f(x', y') h(x - x', y - y') \, dx' \, dy' + \epsilon(x, y) \]

Inversion: Deconvolution
Blind Deconvolution

\[ g(t) = h(t) * f(t) + \epsilon(t) \]
\[ g(x, y) = h(x, y) * f(x, y) + \epsilon(x, y) \]

- Convolution: Given \( f \) and \( h \) compute \( g \)
- Identification: Given \( f \) and \( g \) estimate \( h \)
- Deconvolution: Given \( g \) and \( h \) estimate \( f \)
- Blind deconvolution: Given \( g \) estimate both \( h \) and \( f \)

Discretization:

- \( g = h * f + \epsilon \rightarrow g = H f + \epsilon \),
  \( H \) huge dimensional Toeplitz or TBT matrice obtained from the elements of the impulse response \( h(t) \) or the Point Spread Function (PSF) \( h(x, y) \)

- \( g = f * h + \epsilon \rightarrow g = F h + \epsilon \),
  \( F \) huge dimensional Hankel or HBT matrice obtained from the elements of the input signal \( f(t) \) or the image \( f(x, y) \)
### Identification and Deconvolution: Classical methods

<table>
<thead>
<tr>
<th>Deconvolution</th>
<th>Identification</th>
</tr>
</thead>
<tbody>
<tr>
<td>$g(t) = h(t) \ast f(t) + \epsilon(t)$</td>
<td>$g(t) = h(t) \ast f(t) + \epsilon(t)$</td>
</tr>
</tbody>
</table>

#### Wiener filtering

**Deconvolution**

$$\hat{f}(\omega) = \frac{H^*(\omega)}{|H(\omega)|^2 + \frac{S_{ee}(\omega)}{S_{ff}(\omega)}} g(\omega)$$

**Identification**

$$\hat{h}(\omega) = \frac{F^*(\omega)}{|F(\omega)|^2 + \frac{S_{ee}(\omega)}{S_{hh}(\omega)}} g(\omega)$$

#### Regularization

**Deconvolution**

$$g = H f + \epsilon$$

$$J(f) = \|g - H f\|^2 + \lambda_f \|D_f f\|^2$$

$$\nabla J(f) = -2H'(g - H f) + 2\lambda_f D'_f D_f f$$

$$\hat{f} = [H' H + \lambda_f D'_f D_f]^{-1} H' g$$

**Identification**

$$g = F h + \epsilon$$

$$J(h) = \|g - F h\|^2 + \lambda_h \|D_h h\|^2$$

$$\nabla J(h) = -2F'(g - F h) + 2\lambda_h D'_h D_h h$$

$$\hat{h} = [F' F + \lambda_h D'_h D_h]^{-1} F' g$$

#### Circulant approximation

**Deconvolution**

$$\hat{f}(\omega) = \frac{H^*(\omega)}{|H(\omega)|^2 + \lambda_f |D_f(\omega)|^2} g(\omega)$$

**Identification**

$$\hat{h}(\omega) = \frac{|F^*(\omega)|}{|F(\omega)|^2 + \lambda_h |D_h(\omega)|^2} g(\omega)$$
## Deconvolution, Identification: Bayesian approach

<table>
<thead>
<tr>
<th></th>
<th>Deconvolution</th>
<th>Identification</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Forward model</strong></td>
<td>$g = H f + \epsilon$</td>
<td>$g = F h + \epsilon$</td>
</tr>
<tr>
<td><strong>Likelihood</strong></td>
<td>$p(g</td>
<td>f) = \mathcal{N}(g</td>
</tr>
<tr>
<td><strong>A priori</strong></td>
<td>$p(f) = \mathcal{N}(f</td>
<td>0, \Sigma_f)$</td>
</tr>
<tr>
<td><strong>Bayes</strong></td>
<td>$p(f</td>
<td>g) = \frac{p(g</td>
</tr>
</tbody>
</table>

Particular case: $\Sigma_{\epsilon} = \sigma_{\epsilon}^2 I$, $\Sigma_f = \sigma_f^2 (D_f' D_f)^{-1}$, $\Sigma_h = \sigma_h^2 (D_h' D_h)^{-1}$

### Deconvolution

$p(f | g) \propto p(g | f) p(f) \propto \exp \left[ -J(f) \right]$

$J(f) = \| g - H f \|_2^2 + \lambda_f \| D_f f \|_2^2$

$\lambda_f = \sigma_{\epsilon}^2 / \sigma_f^2$

$p(f | g) = \mathcal{N}(f | \hat{f}, \hat{\Sigma}_f)$

$\hat{f} = [H'H + \lambda_f D_f' D_f]^{-1} F'g$

$\hat{\Sigma}_f = [H'H + \lambda_f D_f' D_f]^{-1}$

### Identification

$p(h | g) \propto p(g | h) p(h) \propto \exp \left[ -J(h) \right]$

$J(h) = \| g - F h \|_2^2 + \lambda_h \| D_h h \|_2^2$

$\lambda_h = \sigma_{\epsilon}^2 / \sigma_h^2$

$p(h | g) = \mathcal{N}(h | \hat{h}, \hat{\Sigma}_h)$

$\hat{h} = [F'F + \lambda_h D_h' D_h]^{-1} F'g$

$\hat{\Sigma}_h = [F'F + \lambda_h D_h' D_h]^{-1}$
 Blind Deconvolution: Bayesian approach

\[ g = H f + \epsilon = F h + \epsilon \]

- Joint posterior law:
  \[ p(f, h | g) \propto p(g | f, h) p(f) p(h) \propto \exp[-J(f, h)] \]
  \[ J(f, h) = \| g - H f \|^2 + \lambda_f \| D_f f \|^2 + \lambda_h \| D_h h \|^2 \]

- Joint MAP:
  \[ (\hat{f}, \hat{h}) = \arg \max_{(f, h)} \{ p(f, h | g) \} = \arg \min_{(f, h)} \{ J(f, h) \} \]

- Alternate optimization:
  \[
  \begin{cases}
    \hat{f}^{(k)} = \arg \max_f \{ p(f^{(k)}, h | g) \} = \arg \min_f \{ J(f^{(k)}, h) \} \\
    \hat{h}^{(k)} = \arg \max_h \{ p(f^{(k)}, h | g) \} = \arg \min_h \{ J(f^{(k)}, h) \}
  \end{cases}
  \]
Blind Deconvolution: Variational Bayesian Approximation algorithm

- Joint posterior law:
  \[ p(f, h | g) \propto p(g | f, h) p(f) p(h) \]

- Approximation: \( p(f, h | g) \) by \( q(f, h) = q_1(f) q_2(h) \)

- Criterion of approximation: Kullback-Leibler
  \[
  KL(q|p) = \int q \ln \frac{q}{p} = \int q_1 q_2 \ln \frac{q_1 q_2}{p}
  \]

  \[
  KL(q_1 q_2|p) = \int q_1 \ln q_1 + \int q_2 \ln q_2 - \int q \ln p
  = -\mathcal{H}(q_1) - \mathcal{H}(q_2) + \langle -\ln p((f, h | g)) \rangle_q
  \]

- When the expression of \( q_1 \) and \( q_2 \) are obtained, use them.
Variational Bayesian Approximation algorithm

- Kullback-Leibler criterion

\[ KL(q_1, q_2 | p) = \int q_1 \ln q_1 + \int q_2 \ln q_2 + \int q \ln p \]
\[ = -\mathcal{H}(q_1) - \mathcal{H}(q_2) + \langle -\ln p((f, h | g)) \rangle_q \]

- Free energy

\[ F(q_1, q_2) = -\langle \ln p((f, h | g)) \rangle_{q_1 q_2} \]

- Equivalence between optimization of \( KL(q_1, q_2 | p) \) and \( F(q_1, q_2) \)

- Alternate optimization:

\[ \hat{q}_1 = \arg \min_{q_1} \{ KL(q_1, q_2 | p) \} = \arg \min_{q_1} \{ F(q_1, q_2) \} \]
\[ \hat{q}_2 = \arg \min_{q_2} \{ KL(q_1, q_2 | p) \} = \arg \min_{q_2} \{ F(q_1, q_2) \} \]
Summary of Bayesian methods for Blind Deconvolution

\[ p(f, h|g) = \frac{p(g|f, h)p(f)p(h)}{p(g)} \]

- **JMAP:** 
  \[(\hat{f}, \hat{h}) = \text{arg max}_{(f, h)} \{ p(f, h|g) \} \]

\[
\begin{align*}
\hat{f}^{(k+1)} &= \text{arg max}_f \{ p(f, h^{(k)}|g) \} \\
\hat{h}^{(k+1)} &= \text{arg max}_h \{ p(f^{(k)}, h|g) \}
\end{align*}
\]

- **VBA:** Approximation: \[ p(f, h|g) \text{ by } q(f, h) = q_1(f) q_2(h) \]

\[
\begin{align*}
\hat{q}_1(f) &= \text{arg min}_{q_1} \{ KL(q_1 q_2|p) \} = \text{arg min}_{q_1} \{ F(q_1 q_2) \} \\
\hat{q}_2(h) &= \text{arg min}_{q_2} \{ KL(q_1 q_2|p) \} = \text{arg min}_{q_2} \{ F(q_1 q_2) \}
\end{align*}
\]
Summary of Bayesian methods for Blind Deconvolution

Alternate optimization for JMAP

\[ \hat{h}^{(0)} \rightarrow \hat{h} \rightarrow \hat{f} = \arg \max_f \{ p(f, h^{(k)} | g) \} \rightarrow \hat{f} \]

\[ \hat{h} = \arg \max_h \{ p(f^{(k)}, h | g) \} \leftarrow \hat{f} \]

Alternate optimization for VBA

\[ q_2^{(0)}(h) \rightarrow q_2(h) \rightarrow \hat{q}_1 = \arg \min_{q_2} \{ \text{KL}(q_1 \hat{q}_2 | p) \} \rightarrow \hat{q}_1(f) \rightarrow \hat{h} \]

\[ \hat{h} \leftarrow \hat{q}_2(h) \leftarrow \hat{q}_2 = \arg \min_{q_2} \{ \text{KL}(\hat{q}_1 q_2 | p) \} \leftarrow \hat{q}_1(f) \]
JMAP and VBA with Gaussian priors

**JMAP:**

Initialization: \( h^{(0)} = h_0, \quad H = \text{Convmtx}(h^{(0)}) \)

Iterations:

\[
f^{(k)} = (H'H + \lambda_f I)^{-1} H'g
\]

\[
F = \text{Convmtx}(f^{(k-1)})
\]

\[
h^{(k)} = (F'F + \lambda_h C'hC_h)^{-1} F'g
\]

\[
H = \text{Convmtx}(h^{(k-1)})
\]

**VBA:**

Initialization: \( h^{(0)} = h_0, H = \text{Convmtx}(h^{(0)}), D_f = I, D_h = I \)

Iterations:

\[
f^{(k)} = (H'H + \lambda_f I + \nu\epsilon D'_f D_f)^{-1} H'g
\]

\[
\Sigma_f = \nu\epsilon (H'H + \lambda_f I + \nu\epsilon D'_f D_h)^{-1}
\]

\[
F = \text{Convmtx}(f^{(k-1)}) \quad \text{Tr}\{H\Sigma_f H'\} = \|D'_h f\|^2_2
\]

\[
h^{(k)} = (F'F + \lambda_h C'hC_h + \nu\epsilon D'_h D_h)^{-1} F'g
\]

\[
\Sigma_h = \nu\epsilon (F'F + \lambda_h C'hC_h + D'_h D_h)^{-1}
\]

\[
H = \text{Convmtx}(h^{(k-1)}) \quad \text{Tr}\{F\Sigma_h F'\} = \|D'_f f\|^2_2
\]

A. Mohammad-Djafari, Bayesian Blind Deconvolution, Ami Kabir University, Math. & Computer Science Dept., December 9, 2014, 12/20
JMAP and VBA with Gaussian priors

\( \hat{h}^{(0)} \rightarrow \hat{h} \rightarrow H = \text{Convmtx}(h) \)

\( \hat{f} = (H'H + \lambda_f I)^{-1} H'g \)

\( \hat{h} \leftarrow \hat{f} \)

\( F = \text{Convmtx}(f) \)

\( \hat{h} = (F'F + \lambda_h C'_h C_h)^{-1} F'g \)

\( \hat{f} \rightarrow \hat{f} \)

\( H = \text{Convmtx}(\hat{h}) \)

\( \text{Tr} \{ F \Sigma_h F' \} = \| D_f f \|_2^2 \)

\( \hat{f} = (H'H + \lambda_f I + D'_f D_f)^{-1} H'g \)

VBA:

\( \hat{h}^{(0)} \rightarrow \hat{h} \rightarrow \Sigma_h \rightarrow \hat{f} \rightarrow \hat{f} \)

\( F = \text{Convmtx}(f) \)

\( \text{Tr} \{ H \Sigma_h H' \} = \| D_h h \|_2^2 \)

\( \hat{h} = (F'F + \lambda_h C'_h C_h + D'_h D_h)^{-1} F'g \)

\( \hat{f} \leftarrow \hat{f} \)

\( \hat{f} \leftarrow \hat{f} \)

\( \Sigma_h = v_{\epsilon}(H'H + \lambda_h C'_h C_h + D'_h D_h)^{-1} \)

A. Mohammad-Djafari, Bayesian Blind Deconvolution, Ami Kabir University, Math. & Computer Science Dept., December 9, 2014, 13/20
Student-t prior

- Sparsity enforcing: Heavy tailed priors: Cauchy and Student-t:

\[ p(f) = \prod_j \mathcal{T}(f_j | \nu, v_f) \]

Infinite scaled mixture representation:

\[ \mathcal{T}(f_j | \nu, v_f) = \int_0^\infty \mathcal{N}(f_j | 0, z_j^{-1}) \mathcal{G}(z_j | \nu/2, \nu/2) \, dz_j \]  \hspace{1cm} (1)

- Hierarchical model: \( p(f | z) p(z) \)

\[ p(f | z) = \prod_j \mathcal{N}(f_j | 0, z_j^{-1}v_f), \quad p(z) = \prod_j \mathcal{G}(z_j | \alpha, \beta) \]

- Joint posterior: \( p(h, f, z | g) \propto p(g | h, f) p(f | z) p(z) \)
  - JMAP: \( (\hat{h}, \hat{f}, \hat{z}) = \arg \max_{(h, f, z)} \{ p(h, f, z | g) \} \)
  - VBA: Approximate \( p(h, f, z | g) \) by \( q_1(h) q_2(f) q_3(z) \)
Because we only changed \( p(f) \) by \( p(f|z)p(z) \) where both are separable, the only changes are:

- replace

\[
f^{(k)} = (H'H + \lambda_f I)^{-1}H'g
\]

- by:

\[
f^{(k)} = \left(H'H + \lambda_f \hat{Z}^{(k)}\right)^{-1}H'g
\]

\[
\hat{Z}^{(k)} = \text{diag} \left[ \hat{z}_j^{(k)} \right], \quad \hat{z}_j^{(k)} = \frac{\hat{\beta}_j}{\hat{\alpha}_j}
\]

with

\[
JMAP \quad \begin{cases} 
\hat{\alpha}_j = \alpha + \frac{1}{2} \| f \|_2^2 \\
\text{and} \\
\hat{\beta}_j = \beta + \frac{1}{2} + \| g - h * f \|_2^2 
\end{cases}
\]

\[
VBA \quad \begin{cases} 
\hat{\alpha}_j = \alpha + \frac{1}{2} < \| f \|_2^2 > \\
\text{and} \\
\hat{\beta}_j = \beta + \frac{1}{2} + < \| g - h * f \|_2^2 > 
\end{cases}
\]
Comparison between JMAP and VBA

Alternate optimization for JMAP

\[
\hat{h}(0) \Rightarrow \hat{h} \Rightarrow Z = \text{diag}[\hat{z}] , \quad H = \text{Convmtx}(\hat{h}) \\
\hat{f} = (H' H + \lambda_f Z)^{-1} H' g \\
\]

\[
\hat{z}(0) \Rightarrow \hat{z} \Rightarrow \\
\alpha_j = \alpha + \frac{1}{2} \| f \|^2 \\
\beta_j = \beta + \frac{1}{2} + \| g - h \ast f \|^2 \\
\hat{z}^{(k)}_j = \frac{\hat{\beta}_j}{\hat{\alpha}_j} \\
F = \text{Convmtx}(f) \\
\hat{h} = (F' F + \lambda_h C'_h C_h)^{-1} F' g \\
\]

A. Mohammad-Djafari, Bayesian Blind Deconvolution, Ami Kabir University, Math. & Computer Science Dept., December 9, 2014, 16/20
Comparison between JMAP and VBA

VBA

\[ Z = \text{diag} [\hat{z}] , \quad H = \text{Convmtx}(\hat{h}) \]
\[ \text{Tr} \{ F \Sigma_h F' \} = \| D_h f \|^2 \]
\[ \hat{f} = (H' H + \lambda_f Z + D_f^T D_f)^{-1} H' g \]

\[ \hat{\alpha}_j = \alpha + \frac{1}{2} < \| f \|^2 > \]
\[ \hat{\beta}_j = \beta + \frac{1}{2} + < \| g - h \ast f \|^2 > \]
\[ \hat{\Sigma}_h = \text{diag} [\hat{z}] \]
\[ \text{Tr} \{ H \Sigma_f H' \} = \| D_h h \|^2 \]
\[ \hat{h} = (F' F + \lambda_h C_h' C_h + D_h^T D_h)^{-1} F' g \]
\[ \hat{\Sigma}_h = \nu \epsilon (H' H + \lambda_h C_h' C_h)^{-1} \]
Conclusions

- In this paper, we considered the Blind Image Deconvolution problem in a Bayesian framework.

- We compared two main algorithms: JMAP and VBA giving some detailed insight for each of them for two cases:
  - Gaussian prior for both IRF $h$ and the input signal $f$ and
  - Gaussian prior for the IRF but a Student-t prior for the input signals or images to enhance or to account for possible sparsity structure of the input.

- JMAP: easy and not very costly but uncertainties are not accounted for.

- VBA: is more costly but both uncertainties of $\hat{f}$ and $\hat{h}$ are accounted for, in each iteration, for computing respectively, $\hat{h}$ and $\hat{f}$.
Simulated results

original $f$

PSF $h$

Blurred & noisy $g$

simple deconv $\hat{f}$

Estimated PSF $\hat{h}$

Estimated $\hat{f}$

A. Mohammad-Djafari, Bayesian Blind Deconvolution, Ami Kabir University, Math. & Computer Science Dept., December 9, 2014, 19/20
Questions and Remarks