Bayesian 3D X-ray computed tomography image reconstruction with a scaled Gaussian mixture prior model
Li Wang, Nicolas Gac, Ali Mohammad-Djafari

To cite this version:
Li Wang, Nicolas Gac, Ali Mohammad-Djafari. Bayesian 3D X-ray computed tomography image reconstruction with a scaled Gaussian mixture prior model. 34th International Workshop on Bayesian Inference and Maximum Entropy Methods in Science and Engineering (MaxEnt’14), Sep 2014, Amboise, France. pp.556-563, 10.1063/1.4906022. hal-01338706

HAL Id: hal-01338706
https://hal-centralesupelec.archives-ouvertes.fr/hal-01338706
Submitted on 29 Jun 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Bayesian 3D X-ray Computed Tomography image reconstruction with a Scaled Gaussian Mixture prior model

Li WANG\(^1\), Nicolas GAC and Ali MOHAMMAD-DJAFARI

Laboratoire des Signaux et Systèmes 3, Rue Joliot-Curie 91192 Gif sur Yvette

Abstract. In order to improve quality of 3D X-ray tomography reconstruction for Non Destructive Testing (NDT), we investigate in this paper hierarchical Bayesian methods. In NDT, useful prior information on the volume like the limited number of materials or the presence of homogeneous area can be included in the iterative reconstruction algorithms. In hierarchical Bayesian methods, not only the volume is estimated thanks to the prior model of the volume but also the hyper parameters of this prior. This additional complexity in the reconstruction methods when applied to large volumes (from \(512^3\) to \(8192^3\) voxels) results in an increasing computational cost. To reduce it, the hierarchical Bayesian methods investigated in this paper lead to an algorithm acceleration by Variational Bayesian Approximation (VBA) \([1]\) and hardware acceleration thanks to projection and back-projection operators paralleled on many core processors like GPU \([2]\).

In this paper, we will consider a Student-t prior on the gradient of the image implemented in a hierarchical way \([3, 4, 1]\). Operators \(H\) (forward or projection) and \(H^t\) (adjoint or back-projection) implanted in multi-GPU \([2]\) have been used in this study. Different methods will be evaluated on synthetic volume "Shepp and Logan" in terms of quality and time of reconstruction. We used several simple regularizations of order 1 and order 2. Other prior models also exists \([5]\). Sometimes for a discrete image, we can do the segmentation and reconstruction at the same time, then the reconstruction can be done with less projections.

Keywords: Computed Tomography, Limited projections, Non Destructive Testing (NDT), Hierarchical Model, Bayesian JMAP, Variational Bayesian Approximation (VBA), Gaussian, Mixture of Gaussians (MoG) and Student-t prior models

INTRODUCTION OF COMPUTED TOMOGRAPHY

Computed Tomography

X-ray computed tomography (X-ray CT) is a technology that uses computer-processed X-ray to produce tomographic images of specific areas of a scanned object, allowing the users to see what is inside it without cutting it. Digital geometry processing is used to generate a three-dimensional image of the inside of an object from a large series of two-dimensional radiographic images taken around a single axis of rotation. X-ray tomographic image reconstruction consist of determining an object function from its projections. The main forward problem of the tomography is the Radon transform.

When considering a practical problem, a discretization is necessary. If we discretize...
\( f(x,y) \) into pixels and put all the pixels in a vector \( f \) and put all the data \( g(\phi,r) \) of different angles \( \phi \) in a vector \( g \), we then obtain:

\[
g = Hf + \varepsilon
\]  

(1)

where \( \varepsilon \) represents the error, and \( H \) is the projection operator in which the element \( H_{ij} \) represents the length of the ray \( i \) in the pixel \( j \).

**BAYESIAN APPROACH**

From this point, the main objective is to infer on \( f \) given the data \( g \) assuming the forward model \( g = Hf + \varepsilon \). By being Bayesian, we mean to use the Bayes rule:

\[
p(f \mid g) = \frac{p(g \mid f)p(f)}{p(g)} \propto p(g \mid f)p(f)
\]  

(2)

to obtain what is called the posterior law \( p(f \mid g) \) from the likelihood \( p(g \mid f) \) and the prior \( p(f) \). To be able to use the Bayesian approach, first we need to assign \( p(g \mid f) \) and \( p(f) \). Then, we can obtain the expression of the posterior law. Finally, we can infer on \( f \) using this posterior law.

*Markov model.* In the Markov model, the value of \( f_j \) has a relationship with the values of its neighbours. For example, in the case of 1D, \( f_j = F(f_{j1}, f_{j2}, \cdots) \), and in an image, the value of a pixel depends on the values of its neighbours pixels.

When the value of \( f_j \) depends only on the values of the neighbours of distance 1, the model is called to have order 1, and for a Gaussian model we have:

\[
p(f_j \mid f_{j-1}, \sigma^2) = \mathcal{N}(f_j \mid f_{j-1}, \sigma^2) \propto \exp \left[ -\frac{1}{2} \frac{(f_j - f_{j-1})^2}{\sigma_j^2} \right]
\]  

(3)

From this we can write:

\[
p(f) \propto \exp \left\{ -\gamma \| \mathbf{D}f \|_j^\beta \right\}
\]

where \( \mathbf{D}f \) represents a suitable defined gradient of the image, \( \gamma \) is a scale parameter and \( \beta \) a shape parameter. For \( \beta = 2 \) a Gauss-Markov model is obtained.

For the noise term \( \varepsilon \) we choose a Gaussia prior law. This leads to

\[
p(g \mid f) \propto \exp \left\{ -\frac{1}{2\sigma_\varepsilon^2} \|g - Hf\|^2 \right\}
\]

and so

\[
p(f \mid g, \sigma_f^2, \sigma_\varepsilon^2) \propto \exp \left\{ -\frac{1}{2\sigma_\varepsilon^2} J(f) \right\}
\]

with \( J(f) = \|g - Hf\|^2 + \lambda \| \mathbf{D}f \|_\beta^\beta \), where \( \lambda \) is the parameter of the regularization term. When \( \beta = 2 \) (Gauss-Markov case), we got the analytical solution:

\[
\hat{f} = (H' H + \lambda \mathbf{D}' \mathbf{D})^{-1} H' g
\]
We have considered different prior laws with Markov model: the Gaussian law, the Generalized Gaussian law, the Cauchy law and the Huber law.

**UNSUPERVISED BAYESIAN**

In previous section, there are parameters $\sigma^2_e$ and $\sigma^2_j$ as well as $\beta$ which have to be assigned. These are called hyper parameters. For a practical applications we need to estimate them also. In the Bayesian approach this can be done via the joint posterior:

$$p(f, \theta | g, \theta_0) = \frac{p(g | f, \theta_1)p(f | \theta_2)p(\theta | \theta_0)}{p(g | \theta_0)}$$  \hspace{1cm} (4)

where $\theta = [\theta_1, \theta_2]$ and $\theta_0$ is a parameter of $\theta_1$ and $\theta_2$.

In this paper we set $\beta = 2$ but we want to estimate $\theta_1 = v_e = \sigma^2_e$ and $\theta_2 = v_f = \sigma^2_f$. As $\theta_1$ and $\theta_2$ are variances, we use a conjugate prior for them. Here we choose the inverse Gamma law as the conjugate prior:

$$\begin{cases}
    p(v_e | \alpha_e, \beta_e) = IG(v_e | \alpha_e, \beta_e) \\
    p(v_f | \alpha_f, \beta_f) = IG(v_f | \alpha_f, \beta_f)
\end{cases}$$  \hspace{1cm} (5)

One way to estimate the unknowns of our model is to compute the Joint Maximum A Posteriori (JMAP) [6].

$$\left(\hat{f}, \hat{\theta}\right) = \arg \max_{f, \theta} p(f, \theta | g, \theta_0)$$  \hspace{1cm} (6)

In the case where the only hyper parameters are $v_e$ and $v_f$, we can apply the following iterative algorithm:

$$\begin{cases}
    \hat{f} = \arg \max_f p(\hat{f}, v_e, v_f | g, \theta_0) = \left(H' H + \frac{v_e}{v_f} D' D\right)^{-1} H' g \\
    \hat{v}_e = \frac{1}{2} \frac{\|g - Hf\|^2 + \beta_e}{\alpha_e + \frac{\|f\|^2}{2} + 1} \\
    \hat{v}_f = \frac{\beta_f + \frac{1}{2} \|Df\|^2}{\alpha_f + \frac{\|f\|^2}{2} + 1}
\end{cases}$$  \hspace{1cm} (7)

**JMAP AND VBA WITH STUDENT-T PRIOR**

In this section, we will consider the hierarchical model which uses a Student-t distribution for modeling the distribution of sparse signals or images.

For an image in which most parts are homogeneous, the gradient of the image is sparse. To enforce sparsity, we propose to use a heavy tailed prior law, for example, the Generalized Gaussian law and the Student-t law. Here we propose to use the Student-t distribution. For the Student-t prior law, we have the property:

$$St(f_j | \nu, \tau) = \int_{0}^{\infty} \mathcal{N}\left(f_j | 0, \frac{1}{z_j}\right) \mathcal{G}\left(z_j | \frac{\nu}{2}, \frac{\nu \tau^2}{2}\right) dz_j$$
where the $z_j$ is a hidden variable which represents the inverse variance of $f_j$ [4, 1].

This property of Infinite Gaussian Mixture gives the possibility to propose the following hierarchical model:

\[
\begin{align*}
  p(f_j | z_j) &= \mathcal{N}(f_j | f_{j-1}, \frac{v_j}{z_j}) \\
p(z_j | \alpha_{z_j}, \beta_{z_j}) &= \mathcal{G}(z_j | \alpha_{z_j}, \beta_{z_j})
\end{align*}
\]

**JMAP.** With the hierarchical model, we can obtain the expression of the joint a posterior:

\[
p(f, Z, \theta | g) \propto p(g | f, v_e) p(f | Z, v_f) p(Z | \alpha_Z, \beta_Z) p(v_e) p(v_f)
\]

where $p(f | Z, v_f) = \prod_j p(f_j | z_j, v_f)$ and $p(Z) = \prod_j p(z_j)$ and for $p(v_e)$ and $p(v_f)$ we use Inverse Gamma law which is given in (6).

An alternating optimisation of this JMAP criterion results to the following algorithm:

\[
\begin{align*}
  f^{(k+1)} &= \left( H'H + \frac{v_e^{(k)}}{v_f^{(k)}} Z^{(k)} \right)^{-1} H'g \\
  z_j^{(k+1)} &= \frac{\alpha_j - \frac{1}{z_j}}{\beta_j + \left( \frac{f_j^{(k+1)}}{z_j} \right)^2} \\
  v_e^{(k+1)} &= \frac{1}{\alpha_e + \frac{1}{2}} + \beta_e \\
  v_f^{(k+1)} &= \frac{1}{\alpha_f + \frac{1}{2}} + \beta_f
\end{align*}
\]

(8)

**VBA.** As we will see, the main inconvenience of the JMAP approach is that we are summarizing the joint posterior law $p(f, \theta | g)$ by its mode only. Also, for obtaining this mode, in general, an iterative alternating optimization is used, where at each iteration only the values of the estimates at previous iterations are used without accounting for their corresponding uncertainties.

In the VBA approach, the main idea here is to approximate $p(f, \theta | g)$ by a separable law $q(f, \theta) = q_1(f)q_2(\theta) = \prod_j q_{1j}(f_j)q_{2}(\theta)$ which can then be used for inferring on $f$ or $\theta$. The main criterion used is the Kullback-Leibler divergence[7, 8, 3]. As we will see, $f$ depends on $q_2(\theta)$ and $q_2(\theta)$ depends on $q_1(f)$, thus accounting for uncertainties in both steps.

We assume that $f_j | \mu_j, v_{fj}, z_j \sim \mathcal{N}(f_j | \mu_j, \frac{\sigma_j}{z_j})$, $z_j | \alpha_{zj}, \beta_{zj} \sim \mathcal{G}(z_j | \alpha_{zj}, \beta_{zj})$, $v_e | \alpha_e, \beta_e \sim \mathcal{G}(v_e | \alpha_e, \beta_e)$ and $v_f | \alpha_f, \beta_f \sim \mathcal{G}(v_f | \alpha_f, \beta_f)$. And we are now considering all the hyper parameters, parameters and unknowns. The alternate optimization
of VBA is then given by:

\[
\begin{align*}
\lambda_j^{(k+1)} &= \frac{\alpha_j^{(k)}}{\beta_j^{(k)}} \frac{\alpha_j^{(k)}}{\beta_j^{(k)}} \frac{1}{1 + \lambda_j^{(k+1)}} \\
\mu_j^{(k+1)} &= -\frac{1}{1 + \lambda_j^{(k+1)}} \\
\sigma_j^{(k+1)} &= \frac{1}{\alpha_j^{(k)}} \frac{1}{\beta_j^{(k)}} \frac{1}{1 + \lambda_j^{(k+1)}} \\
\alpha_{z_j}^{(k+1)} &= \alpha_{z_j}^{(k)} + \frac{1}{2} \mu_j^{(k+1)} \frac{\alpha_j^{(k)}}{\beta_j^{(k)}} \\
\beta_{z_j}^{(k+1)} &= \beta_{z_j}^{(k)} + \frac{1}{2} \mu_j^{(k+1)} \frac{\alpha_j^{(k)}}{\beta_j^{(k)}} \\
\tilde{\alpha}_{j}^{(k+1)} &= \tilde{\alpha}_{j}^{(k)} + \frac{1}{2} \sum_{j=1}^{N} \left( \frac{\alpha_j^{(k)}}{\beta_j^{(k)}} \right)^2 + \tilde{\sigma}_{j}^{(k+1)} \\
\tilde{\beta}_{j}^{(k+1)} &= \tilde{\beta}_{j}^{(k)} + \frac{1}{2} \sum_{j=1}^{N} \left( \frac{\alpha_j^{(k)}}{\beta_j^{(k)}} \right)^2 + \tilde{\sigma}_{j}^{(k+1)} \\
\end{align*}
\]

where \( N \) is the length of the vector \( f \), \( M \) is the length of the vector \( g \), and the definition of \( \mathcal{A} \) and \( \mathcal{B} \) are:

\[
\begin{align*}
\mathcal{A} &= (H' H)_{jj} \\
\mathcal{B} &= - (H' g)_j + (H' H \mu)_j - (H' H)_{jj} \mu_j
\end{align*}
\]

Here we need to determine the diagonal elements of the matrix \( H' H \) to optimize the unknowns.

**IMPLEMENTATION**

In the forward model \( g = H f + \epsilon \), the vector \( f \) correspond to the object, \( H \) is projection operator and \( g \) represents the projections. Here in our simulation, we use the synthetic volume "Shepp and Logan" with size 256 \( \times \) 256 \( \times \) 256 voxels. In this report, the projection has been done in 256 angles, and under each angle the screen of detectors will receive an image of size 256 \( \times \) 256 pixels. The matrix \( H \) corresponds to the matrix of projection, of which the most component are zero. In the problem 3D, we don’t know the matrix \( H \), but the projection \( H f \) and back-projection \( H' g \) can be accessed. The term \( \epsilon \) represents the noise which follows a Gaussian law. In the algorithm, the most costly parts of the computation are operations on \( H f \) (projection) and the \( H' g \) (back-projection). These two operations are implemented using a GPU.
NUMERICAL RESULTS

In the numerical implementation part, we have compared the JMAP method with the results of different reconstruction methods: simple back-projection, filtered back-projection, least square method and Bayes method (MAP) with different regularizations (Simple Gaussian, Generalized Gaussian, Cauchy and Huber). The middle slice after reconstruction of different methods (after 200 iterations) is shown in FIGURE 1, and we can compare the relative error of different methods, where \( \delta_f = \frac{\|f - \hat{f}\|^2}{\|f\|^2} \) and \( \delta_g = \frac{\|g - \hat{g}\|^2}{\|g\|^2} \).

With the theoretical projection \( H \) and back-projection \( H' \), the object obtained has a large error, and we can not distinguish the details in the image. With the Feldkamp back-projection, we can obtain an image which is already clear enough to distinguish the different materials, but the boundaries are blurred. When we apply the optimisation with different prior models, we can see that the boundaries are easier to distinguish and different materials are clearly separated. In the JMAP method, the boundaries of different materials are more clear than the others methods.

A more accurate compare is showed in FIGURE 2, the error of the JMAP method is smaller than the other non-hierarchical methods.

We can see that the method JMAP has the lowest error among the methods that we considered. With the JMAP method, the borders of two different materials are more distinct.

The VBA method for the big data 3D problem is more complicated than the JMAP
FIGURE 2. The criterion of different method of reconstruction.

FIGURE 3. The criterion of different method of reconstruction.
method. The calculate of the diagonals elements of the matrix $H'HH$ for the case 3D is very difficult because of the unknown huge size matrix $H$ and the time needed for the projection and back-projection. Thus we have compared the VBA method and other methods for a case 1D where the $f$ is of size $N = 29$, $g$ is of size $M = 96$ and the matrix $H$ is known. The relative error of different methods in this case is showed in figure 3. From the compare of different method we can see that the VBA method works better than the others.

CONCLUSIONS

In this paper, JMAP method and VBA method are proposed for doing Bayesian computations for inverse problems where a hierarchical prior modeling is used for the unknowns. A Student-t prior model, which can be written via hidden variables, is considered, and it gives the model a hierarchical structure. In comparing the different reconstruction models, we can say that the method JMAP, which considers a hierarchical problem, has a better property than the other methods. The VBA method also considers the hierarchical problem. The difference is that the VBA method will take into account not only the unknown parameters, but also the uncertainties of the unknowns. The main problem of VBA is to calculate the diagonal elements of the matrix $H'HH$. For a cube of $512 \times 512 \times 512$ voxels, it takes more than 10 days to calculate the diagonals elements. The next step is to optimise the coding of projection and back-projection to reduce the time of calculation and compare these two methods.

REFERENCES