A fast method for simultaneous reconstruction and segmentation in X-ray CT application

Yiqiu Dong\textsuperscript{a}, Chunlin Wu\textsuperscript{b} and Shi Yan\textsuperscript{a,b}

\textsuperscript{a}Department of Applied Mathematics and Computer Science, Technical University of Denmark, Lyngby, Denmark; \textsuperscript{b}School of Mathematical Science, Nankai University, Tianjin, China

\textbf{ABSTRACT}

In this paper, we propose a new method to solve the minimization problem in a simultaneous reconstruction and segmentation (SRS) model for X-ray computed tomography (CT). The SRS model uses Bayes' rule and the maximum a posteriori (MAP) estimate on the hidden Markov measure field model (HMMFM). The original method [Romanov M, Dahl AB, Dong Y, Hansen PC. Simultaneous tomographic reconstruction and segmentation with class priors. Inverse Problems Sci Eng. 2016;24(8):1432–1453] includes a subproblem with logarithmic-summation (log-sum) term, which is non-separable to the classification index. This subproblem was solved by Frank–Wolfe algorithm, which is very time consuming especially when dealing with large-scale CT problems. The starting point of this paper is the commutativity of log-sum operations, where the log-sum problem could be transformed into a sum-log problem by introducing an auxiliary variable. The corresponding sum-log problem for the SRS model is separable. By applying the primal-dual algorithm, the sum-log problem turns into several easy-to-solve convex subproblems. In addition, we introduce an improved model by adding Tikhonov regularization on the SRS model, and give some convergence results for the proposed methods. Experimental results demonstrate that the proposed methods produce comparable results compared with the original SRS method with much less CPU time.

\textbf{ARTICLE HISTORY}

Received 13 November 2019
Accepted 23 October 2021

\textbf{KEYWORDS}

Simultaneous reconstruction and segmentation; inverse problems; X-ray CT; alternating minimization method; hidden Markov measure field models

\textbf{MATHEMATICS SUBJECT CLASSIFICATIONS}

65F22; 65K10

\textbf{1. Introduction}

X-ray computed tomography (CT) is an important application of inverse problems, which reconstructs the attenuation coefficients of an object from the damping of X-rays. Different materials have different attenuation coefficients, thus, we are able to see the interior of an object. X-ray CT is widely applied in a lot of fields, and many reconstruction methods have been proposed and applied in the industry, such as the filtered back projection algorithm [1] and algebraic reconstruction techniques [2,3].

After the reconstruction, it is common to apply the image segmentation technique for distinguishing some interested regions. For example, in medical applications we may want...
to separate bones and water/soft tissues, and in industrial applications we may want to separate plastics and metals [4]. A danger to do the reconstruction firstly and segmentation secondly is the error propagation issue, i.e. reconstruction errors propagate in an undesirable way to the segmentation. To avoid this, two types of methods were proposed. One is discrete tomographic methods, e.g. in [5,6], where only a small number of intensity values are assigned to the reconstruction in order to obtain the segmentation directly. In these methods, the reconstruction results are not in the standard intensity range. The other type is simultaneous reconstruction and segmentation (SRS) methods, e.g. in [7–10]. In the SRS methods, the reconstruction and segmentation results are included in a single variational model and be solved together. Due to this simultaneous procedure, segmentation results in fact potentially play a regularization role and further improve the reconstruction.

As far as we know, the first SRS method in CT was proposed in [8], based on the Mumford–Shah level-set approach [11]. In [12], another SRS method according to the hidden Markov measure field model (HMMFM) [9] was proposed, where the segmentation result is obtained through a probability map. To fuse SPECT/CT data together, an SRS model is proposed in [13]. An SRS method based on the Potts model [14] was proposed in [15]. The authors in [16] proposed an SRS model for CT with shadowed data. In order to segment the objects that have different patterns during CT reconstruction, in [17] the dictionary learning technique was introduced into the SRS methods. In addition, many other techniques were incorporated in the SRS methods, such as the graph cut method [18,19], the Bregman distance [20], and the Gamma-mixture prior [21]. SRS could also be used for metal artefact reduction, see [22]. Here, we briefly review the closely related methods to our work in the following.

Following the idea from [9], in [7] the means and variances of the segmentation classes were used as prior and a new SRS method was proposed. Numerical results show that this SRS method can significantly improve the accuracy of reconstruction and segmentation. Further, this method was extended by withdrawing the prior information on the variance, see [10]. In [7], by using Bayes’ rule and the maximum a posteriori (MAP) estimation with proper prior information on the means and variances of the classes, the authors proposed the following variational model:

\[
\min_{x, \delta} E_0(x, \delta) = \lambda_n ||Ax - b||_2^2 + \lambda_c \sum_{k=1}^{K} R(\delta_k) \\
- \sum_{j=1}^{N} \ln \left[ \sum_{k=1}^{K} \frac{\delta_{jk}}{\sqrt{2\pi} \sigma_k} \exp \left( -\frac{(x_j - \mu_k)^2}{2\sigma_k^2} \right) \right],
\]

where \( A \in \mathbb{R}^{M \times N} \) is the system matrix of the CT scanner, \( x \in \mathbb{R}^N \) is the image of attenuation coefficients, and \( b \in \mathbb{R}^M \) contains all measurements. Here, we assume that the image consists of \( K \) classes, and define \( \delta = (\delta_1, \ldots, \delta_K) = (\delta_{jk}) \in \mathbb{R}^{N \times K} \) with \( \delta_{jk} \) as the probability of pixel \( x_j \) belonging to the \( k \)th class. Then we have the constraint, \( \sum_{k=1}^{K} \delta_{jk} = 1 \) and \( \delta_{jk} \geq 0 \) for all \( j \) and \( k \). Furthermore, \( \mu_k \) and \( \sigma_k \) are the mean and standard deviation of the \( k \)th class, respectively. In addition, \( \lambda_n, \lambda_c > 0 \) are the regularization parameters and \( R(\cdot) \) denotes the regularization term. In [7], Tikhonov regularization [23] and total variation (TV) regularization [24] have been tested. Because the logarithmic operator and summation operator are non-separable in the last term of (1) (log-sum term), the constrained
minimization problem in (1) is very difficult to solve. When solving the subproblem with respect to \( x \) in [7], the authors have to use an approximation of the log-sum term. Moreover, the algorithm is time consuming, which limits its applications to large-scale CT problems.

In this paper, we propose a new method to deal with model (1). Inspired by the work in [25,26], the log-sum term can be transformed into a sum-log term by introducing an auxiliary variable. The transformed problem leads to some subproblems that are much easier to solve, and we could apply more advanced algorithms to solve them. Comparing with the method proposed in [7], the new method is much more efficient, which makes it possible to be applied on large-scale CT problems. In addition, we provide some convergence results on the new algorithm.

The rest of the paper is organized as follows. In Section 2, the method in [7] is briefly reviewed. In Section 3, by transforming the log-sum term, we introduce a new method that can solve the minimization problem in model (1) without simplification. Furthermore, an improved model is proposed and some convergence results are given. In Section 4, numerical results are presented. Finally, conclusions are drawn in Section 5.

1.1. Contributions

The main contributions of this work are following:

- By transforming the log-sum term, we introduce a modification of model (1), which is much easier to solve.
- We apply more advanced algorithm to solve the subproblems in the new model, which is much more efficient.
- By introducing a smoothing regularization term on \( x \), we propose an improved model, which provides better numerical results.
- We test the new methods on several simulated data, and show that the new methods can handle larger scaled CT problems comparing with [7].
- We provide some theoretical convergence results on the new methods.

2. Review the method in [7]

In [7], the authors use the alternating minimization method [27] to solve model (1) as follows:

\[
\begin{align*}
x^{n+1} &= \arg \min_x E_0(x, \delta^n), \\
\delta^{n+1} &= \arg \min_\delta E_0(x^{n+1}, \delta).
\end{align*}
\]

The last term in (1) is a log-sum term, which is non-separable to the classification index \( k \), thus it is very difficult to minimize the sub-problem according to \( x \). In order to solve the sub-problem (2), i.e.

\[
\min_x \lambda_n \|Ax - b\|_2^2 - \sum_{j=1}^N \ln \left( \sum_{k=1}^K \frac{\delta_{jk}^n}{\sqrt{2\pi}\sigma_k} \exp \left( -\frac{(x_j - \mu_k)^2}{2\sigma_k^2} \right) \right),
\]
the authors applied the two-step approximation to simplify the log-sum term. Denote
\[ p(x_j|\delta_j, \mu, \sigma) = \sum_{k=1}^{K} \frac{\delta_{jk}}{\sqrt{2\pi} \sigma_k} \exp \left( -\frac{(x_j - \mu_k)^2}{2\sigma_k^2} \right). \] (4)

In the first step, \( p(x_j|\delta_j, \mu, \sigma) \) is approximated by a ‘flat’ Gaussian distribution
\[ \hat{p}(x_j|\delta_j, \mu, \sigma) = \frac{1}{\sqrt{2\pi} \hat{\sigma}_j} \exp \left( -\frac{(x_j - \hat{\mu}_j)^2}{2\hat{\sigma}_j^2} \right), \] (5)

where
\[ \hat{\mu}_j = \sum_{k=1}^{K} \delta_{jk} \mu_k \quad \text{and} \quad \hat{\sigma}_j^2 = \sum_{k=1}^{K} \delta_{jk}(\sigma_k^2 + \mu_k^2) - \hat{\mu}_j^2. \]
After a few iterations on (2) and (3), when the segmentation \( \delta \) is good enough, a ‘sharp’ Gaussian distribution is used to approximate \( p(x_j|\delta_j, \mu, \sigma) \) and further improve the reconstruction result, i.e.
\[ p(x_j|\delta_j, \mu, \sigma) \approx \tilde{p}(x_j|\delta_j, \mu, \sigma) = \frac{1}{\sqrt{2\pi} \sigma_k} \exp \left( -\frac{(x_j - \mu_k)^2}{2\sigma_k^2} \right), \] (6)
where \( k_j = \arg \max_k \delta_{jk} \).

Clearly, after using a single Gaussian distribution to approximate the sum of several Gaussian distribution in \( p(x_j|\delta_j, \mu, \sigma) \), the summation operator of the log-sum in model (1) is eliminated, and the reconstruction \( x \) is obtained by solving a simple least-squares problem.

The sub-problem (3) is
\[ \min_{\delta} \lambda_c \sum_{k=1}^{K} R(\delta_k) - \sum_{j=1}^{N} \ln \left[ \sum_{k=1}^{K} \frac{\delta_{jk}}{\sqrt{2\pi} \sigma_k} \exp \left( -\frac{(x_j^n - \mu_k)^2}{2\sigma_k^2} \right) \right], \]
with the constraint \( \sum_{k=1}^{K} \delta_{jk} = 1 \) and \( \delta_{jk} \geq 0 \) for all \( j \) and \( k \). Because of the constraints and the non-differentiable TV term, a general constrained gradient descent method such as the Frank–Wolfe algorithm [28] could be applied here. Due to the singularity of the TV term, it is well known that the gradient descent algorithms are very inefficient [29].

Obviously, solving the minimization problem in (1) is challenging due to the log-sum term. In this paper, we will take advantage of the commutativity of the log-sum operations studied in [25,26], and transform model (1) into several easy-to-solve convex sub-problems. Besides, we could provide some convergence results that were missing in [7].

### 3. Our method and convergence results

In this section, we propose a new method that solves model (1) and provide some convergence results of the new method.
3.1. The proposed method

Due to the log-sum term in (1), the sub-problem with respect to \( x \) is non-convex and very difficult to solve. Inspired by the transformation of the log-sum operations introduced in [25,26], we can transform the log-sum term into a much simpler form. To do so, we would need the following proposition.

**Proposition 3.1:** (Commutativity of the log-sum operations [25,26]). Given \( f_k > 0, k = 1, 2 \cdots, K \), we have

\[
- \ln \sum_{k=1}^{K} f_k = \min_{\phi \in A} \left\{ - \sum_{k=1}^{K} \phi_k \ln f_k + \sum_{k=1}^{K} \phi_k \ln \phi_k \right\},
\]

where \( \phi = (\phi_1, \phi_2, \ldots, \phi_K) \), and

\[
A = \left\{ \xi | \sum_{k=1}^{K} \xi_k = 1, \xi_k \in (0, 1) \text{ for all } k \right\}.
\]

According to the log-sum term in (1), we define

\[
f_{jk}(x_j, \delta_{jk}) = \frac{\delta_{jk}}{\sqrt{2\pi} \sigma_k} \exp \left( - \frac{(x_j - \mu_k)^2}{2\sigma_k^2} \right). \tag{7}
\]

By applying Proposition 3.1 on (7), we transform the minimization problem (1) into a new optimization problem with respect to the variables \((x, \delta, \phi)\):

\[
\min_{x, \delta \in \mathcal{B}, \phi \in \mathcal{B}} \left\{ E(x, \delta, \phi) = \lambda_n ||Ax - b||^2_2 + \lambda_c \sum_{k=1}^{K} R(\delta_k) + \sum_{j=1}^{N} \left[ - \sum_{k=1}^{K} \phi_{jk} \ln f_{jk}(x_j, \delta_{jk}) + \sum_{k=1}^{K} \phi_{jk} \ln \phi_{jk} \right] \right\}, \tag{8}
\]

where \( \phi \in \mathbb{R}^{N \times K} \) and \( \mathcal{B} = \{ \xi \in \mathbb{R}^{N \times K} | \sum_{k=1}^{K} \xi_{jk} = 1 \text{ and } \xi_{jk} \in (0, 1) \text{ for all } j, k \} \). Note that here we let \( \delta_{jk} \in (0, 1) \) to avoid \( f_{jk}(x_j, \delta_{jk}) = 0 \).

Since \( \delta \) corresponds to the segmentation results, we assume that it is piecewise constant according to each class. Thus, in our method we use TV regularization on each column of \( \delta \), i.e.

\[
R_{TV}(\delta_k) = \sum_{j=1}^{N} \| [\nabla \delta]_{jk} \|_2,
\]

where \( \nabla = \begin{bmatrix} \nabla_1 \\ \nabla_2 \end{bmatrix} \), and \( \nabla_1, \nabla_2 : \mathbb{R}^N \rightarrow \mathbb{R}^N \) denote the discrete derivative operators along the horizontal and vertical directions, respectively. Here, we use the forward finite difference scheme with the Neumann boundary condition. In addition, \([v]_{jk} = \begin{bmatrix} c_{v_{jk}} \\ v_{(j-N),k} \end{bmatrix}\) with \( v \in \mathbb{R}^{2N \times K} \).

In order to solve the minimization problem (8), we apply the alternating minimization method [27], i.e. we solve the sub-problems with respect to \( x, \delta \) and \( \phi \) alternately. The detailed algorithm is given in Algorithm 1.
Algorithm 1 Algorithm for solving the minimization problem in (8)

1. Set $\lambda_n$ and $\lambda_c$, and initialize $n = 0, x^0 = 0, \delta^0$ and $\phi^0$ with $\delta^0_{j,k} = \phi^0_{j,k} = \frac{1}{K}$ for all $j$ and $k$.
2. Update $x^{n+1}$, by

$$x^{n+1} = \arg\min_x E(x, \delta^n, \phi^n)$$

$$= \arg\min_x \left\{ \lambda_n \|Ax - b\|^2 + \sum_{j=1}^{N} \sum_{k=1}^{K} \left[ \frac{\phi^n_{jk}}{2\sigma_k^2} (x_j - \mu_k)^2 \right] \right\}. \quad (9)$$

3. Update $\delta^{n+1}$, by

$$\delta^{n+1} = \arg\min_{\delta \in B} E(x^{n+1}, \delta, \phi^n)$$

$$= \arg\min_{\delta \in B} \left\{ \lambda_c \sum_{k=1}^{K} R_{TV}(\delta_k) + \sum_{j=1}^{N} \sum_{k=1}^{K} (-\phi^n_{jk} \ln \delta_{jk}) \right\}. \quad (10)$$

4. Update $\phi^{n+1}$, by

$$\phi^{n+1} = \arg\min_{\phi \in B} E(x^{n+1}, \delta^{n+1}, \phi)$$

$$= \arg\min_{\phi \in B} \left\{ \sum_{j=1}^{N} \left[ -\sum_{k=1}^{K} \phi_{jk} \ln f_{jk}(x^{n+1}_j, \delta^{n+1}_{jk}) + \sum_{k=1}^{K} \phi_{jk} \ln \phi_{jk} \right] \right\}. \quad (11)$$

5. If $\frac{\|x^{n+1} - x^n\|_2}{\|x^n\|_2} < 10^{-4}$, then stop. Otherwise, let $n = n + 1$, and go to 2.

Note that all three sub-problems (9), (10) and (11) are convex, but the whole energy function in (8) is non-convex. Therefore, the numerical results would depend on the initialization and the numerical algorithms. In Algorithm 1, since we do not have any prior information on the segmentation, we simply set $\delta^0_{j,k} = \phi^0_{j,k} = \frac{1}{K}$ for all $j$ and $k$ as in [7]. But if there is any prior information on $\delta$, it should be taken into account in order to further improve the results. Due to the convexity of the subproblem (9), the initialization of $x$ would not make any difference.

3.1.1. The $x$ sub-problem and the $\phi$ sub-problem

The $x$ sub-problem and the $\phi$ sub-problem are easy to solve and are briefly described here.

In Algorithm 1, the energy function in sub-problem (9) on $x$ is quadratic, so it can be efficiently solved by using the CGLS method [30].

Since sub-problem (11) is separable, we can solve $\phi_{j,k}$ element-wise. According to the first-order optimality condition together with the Lagrange multiplier technique, we obtain
the closed-form solution:

\[
\phi_{jk}^{n+1} = \frac{f_{jk}(x_j^{n+1}, \delta_{jk}^{n+1})}{\sum_{l=1}^{K} f_{jl}(x_j^{n+1}, \delta_{jl}^{n+1})}.
\] (12)

In the following subsection, we will focus on solving \( \delta \) sub-problem (10).

### 3.1.2. The \( \delta \) sub-problem

The difficulties in solving the minimization problem in (10) are the non-differentiable term \( RTV(\delta_k) \), nonlinear and nonquadratic term \( \ln \delta_{jk} \), and the constraint \( \delta \in B \). For ease of readability, we remove the iteration number \( n \) in Algorithm 1 in this section, and denote the iteration number in the solver of the \( \delta \) sub-problem as \( m \). We adopt the primal–dual method [31] to solve this sub-problem.

Firstly, we rewrite the \( \delta \) sub-problem into the following form:

\[
\min_{\delta, u, v} J(u, v) + G(\delta), \quad \text{s.t.} \quad u = \delta, \ v = \nabla \delta, \tag{13}
\]

where \( u \in \mathbb{R}^{N \times K} \) and \( v \in \mathbb{R}^{2N \times K} \) are two auxiliary variables, and

\[
J(u, v) = -\sum_{j=1}^{N} \sum_{k=1}^{K} \phi_{jk} \ln u_{jk} + \lambda_c \sum_{j=1}^{N} \sum_{k=1}^{K} \|[v]_{jk}\|_2,
\]

\[
G = I_{B} = \begin{cases} 0 & \delta \in B, \\ +\infty & \text{else.} \end{cases}
\]

Define \( y = [u \ v] \in \mathbb{R}^{3N \times K} \) and \( L = [\frac{\partial}{\partial \delta}] : \mathbb{R}^{N \times K} \rightarrow \mathbb{R}^{3N \times K} \), where \( I : \mathbb{R}^{N \times K} \rightarrow \mathbb{R}^{N \times K} \) denotes the identity operator. Applying the standard primal–dual method, we need to solve the following sub-problems:

\[
\begin{cases}
\dot{y}^{m+1} = (I + \tau_1 \partial J^*)^{-1} (y^{m} + \tau_1 L \delta^{m}), \\
\dot{\delta}^{m+1} = (I + \tau_2 \partial G)^{-1} (\delta^{m} - \tau_2 L^* y^{m+1}), \\
\dot{\delta}^{m+1} = \delta^{m+1} + \theta (\delta^{m+1} - \delta^{m}).
\end{cases}
\]

where \( J^*, L^* \) are the convex conjugates of \( J \) and \( L \), respectively.

To solve the \( y \) sub-problem, we use the celebrated Moreau's identity,

\[
(I + \tau_1 \partial J^*)^{-1}(y^*) + \tau_1 \left(1 + \frac{1}{\tau_1} \partial J\right)^{-1} \left(\frac{y^*}{\tau_1}\right) = y^*,
\]

then obtain the update of \( y^{m+1} \) pointwisely as

\[
\dot{u}_{jk}^{m+1} = \frac{u_{jk}^{m} + \sqrt{(u_{jk}^{m})^2 + 4\tau_1 \phi_{jk}}}{2\tau_1}, \tag{14}
\]

\[
[y]^{m+1}_{jk} = \max\{\|[v]^{m}_{jk}\|_2 - \lambda_c, 0\}\frac{\|[v]^{m}_{jk}\|_2}{\tau_1 \|[v]^{m}_{jk}\|_2}[v]^{m}_{jk}, \tag{15}
\]

for all \( j = 1, 2, \ldots, N \), and \( k = 1, 2, \ldots, K \).
The $\delta^m$ sub-problem is a projection on a simplex defined by $B$, which can be written as
\[
\delta^{m+1} = \text{Proj}_B(\delta^m - \tau_2 L^* y^{m+1}),
\] (16)
and we apply the method proposed in [32] to solve it.

**Algorithm 2** Primal--dual method for solving the $\delta$ sub-problem

1. Set $\hat{\delta}^0 = \delta^n$, $y^0 = L \hat{\delta}^0$.
2. Update $y^{m+1}$ by (14) and (15).
3. Update $\delta^{m+1}$ (16).
4. Update $\hat{\delta}^{m+1} = \delta^{m+1} + \theta (\delta^{m+1} - \delta^m)$.
5. If $\|\delta^{m+1} - \delta^m\|_2 < 10^{-3}$, then stop. Otherwise let $m = m + 1$ and go to 2.

### 3.2. The improved model

If we look back to model (8), we can see that the only obvious regularization term in the model is $R_{TV}(\delta_k)$. In [7], due to the log-sum term in (1), the segmentation information represented in $\delta$ is utilized to regularize the reconstruction, and there were not any smoothing requirements on $x$. While in the proposed new model (9), $x$ is updated by using the unregularized $\phi$, which brings a risk: the isolated points on unregularized $\phi$ might lead to isolated points in the reconstruction result $x$. This phenomenon is presented in Figure 3 in our numerical results. In order to further improve the reconstruction, we add Tikhonov regularization on $x$, and the modified model (8) is given as follows:

\[
\begin{align*}
\min_{x, \delta, \phi \in B} \quad & F(x, \delta, \phi) = \lambda_n \|Ax - b\|_2^2 + \lambda_\gamma \|\nabla x\|_2^2 + \lambda_c \sum_{k=1}^{K} R_{TV}(\delta_k) \\
& \quad + \sum_{j=1}^{N} \left[ - \sum_{k=1}^{K} \phi_{jk} \ln f_{jk}(x_j, \delta_{jk}) + \sum_{k=1}^{K} \phi_{jk} \ln \phi_{jk} \right],
\end{align*}
\] (17)

where $\lambda_j > 0$ is a regularization parameter. Here we choose Tikhonov regularization on the reconstruction $x$ based on three reasons: (1) we need a smoothing regularization in order to reduce the artefacts from the measurement noise. (2) Tikhonov regularization is quadratic, and it is easy to solve. (3) The reconstruction problem is underdetermined, so we need regularization to ensure the uniqueness of the solution.

We list the detailed algorithm in Algorithm 3. The only difference between Algorithm 1 and Algorithm 3 is the added Tikhonov regularization term in sub-problem (18) with respect to $x$. With Tikhonov regularization the energy function is still quadratic, we can apply the CGLS method [30] to solve (18) efficiently.

### 3.3. Convergence results

The energy functions in both models (8) and (17) are nonconvex, so it is very challenging to prove the proposed algorithms converging to a global minimizer. In this section, we
Algorithm 3 Algorithm for solving the minimization problem in (17)

1. Set $\lambda_n$ and $\lambda_c$, and initialize $x^0 = 0$, $\delta^0$ and $\phi^0$ with all $\delta_{j,k}^0 = \phi_{j,k}^0 = \frac{1}{K}$.

2. Update $x^{n+1}$, by

$$x^{n+1} = \arg\min_x F(x, \delta^n, \phi^n) = \arg\min_x \left\{ \lambda_n||Ax - b||^2_2 + \sum_{j=1}^N \sum_{k=1}^K \left[ \frac{\phi_{jk}^n}{2\sigma_k^2} (x_j - \mu_k)^2 \right] + \lambda_t ||\nabla x||^2_2 \right\}. \quad (18)$$

3. Update $\delta^{n+1}$, by

$$\delta^{n+1} = \arg\min_{\delta \in B} F(x^{n+1}, \delta, \phi^n) = \arg\min_{\delta \in B} \left\{ \lambda_c \sum_{k=1}^K R_{TV}(\delta_k) + \sum_{j=1}^N \sum_{k=1}^K (-\phi_{jk}^n \ln \delta_{jk}) \right\}. \quad (19)$$

4. Update $\phi^{n+1}$, by

$$\phi^{n+1} = \arg\min_{\phi \in B} F(x^{n+1}, \delta^{n+1}, \phi)$$

$$& = \arg\min_{\phi \in B} \left\{ \sum_{j=1}^N \left[ -\sum_{k=1}^K \phi_{jk} \ln f_{jk}(x_j^{n+1}, \delta_{j,k}^{n+1}) + \phi_{jk} \ln \phi_{jk} \right] \right\}. \quad (20)$$

5. If $\frac{||x^{n+1} - x^n||_2}{||x^n||_2} < 10^{-4}$, stop. Or let $n = n + 1$, goto 2.

will show that the proposed algorithms generate a sequence converging to a coordinate-wise minimum point. Note that the model (8) could be seen as a special case of (17) if we choose $\lambda_t = 0$. Thus, we only give the convergence results to Algorithm 3 for solving the minimization problem in (17).

**Proposition 3.2:** For the sequence $\{(x^n, \delta^n, \phi^n)\}$ generated in Algorithm 3, every cluster point is a coordinatewise minimum point of $F(x, \delta, \phi)$.

**Proof:** We use Theorem 5.1 in [33] to give our convergence result. For self-contained, we list Theorem 5.1 in the appendix. For more details, please see [33]. Here, we verify that our algorithm satisfies the assumptions of the theorem.

According to (17), we set

$$F_0(x, \delta, \phi) = \sum_{j=1}^N \left[ -\sum_{k=1}^K \phi_{jk} \ln f_{jk}(x_j, \delta_{j,k}) + \sum_{k=1}^K \phi_{jk} \ln \phi_{jk} \right],$$

$$F_1(x) = \lambda_n||Ax - b||^2_2 + \lambda_t ||\nabla x||^2_2,$$
\[ F_2(\delta) = \begin{cases} \lambda_c \sum_{k=1}^{K} R_{TV}(\delta_k) & \text{if } \sum_{k=1}^{K} \delta_{jk} = 1, \delta_{jk} > 0, \\ \infty & \text{otherwise,} \end{cases} \]

\[ F_3(\phi) = \begin{cases} 0 & \text{if } \sum_{k=1}^{K} \delta_{jk} = 1, \delta_{jk} > 0, \\ \infty & \text{otherwise.} \end{cases} \]

We have the following statements:

- The essentially cyclic rule (Definition A.2) is satisfied, according to Algorithm 3.
- Assumption B1 (in Definition A.3) is satisfied, due to that \( F_0 \) is continuous in its domain.
- Assumption B2 (in Definition A.3) is satisfied, because the right hand of (18) (19) (20) are strictly convex, by the following facts:
  - (18) is quadratic to \( x \) and the second term is strictly convex.
  - TV norm is convex in (19).
  - The second term in (19) and the second term in (20) are separable, and each of the components is strictly convex.
  - The first term in (20) is linear.
  - The feasible sets of \( F_1, F_2, F_3 \) are convex.
- Assumption B3 (in Definition A.3) is satisfied, because \( F_0 \) is continuous on its domain and \( F_1, F_2, F_3 \) are lsc.
- Assumption C2 (in Definition A.3) is satisfied, because the domain of \( F_0 \) is \( \mathbb{R}^N \times B \times B \).

By applying Theorem A.1 in Appendix, i.e. Theorem 5.1 in [33], we know that for the sequence \( \{(x^n, \delta^n, \phi^n)\} \) generated by Algorithm 3, every cluster point is a coordinatewise minimum point of \( F \).

4. Numerical experiments

In this section, we present some numerical experiment results to demonstrate the performance of our methods. We compare them with the one introduced in [7]. All numerical tests are done on a linux server equipped with CPU 2.30Hz and MATLAB R2020a.

4.1. Experimental settings

In our numerical experiments, the phantoms are generated from the AIR Tools package [3] with the command `phantomgallery`, and the system matrix of the CT scanner is obtained by calling `parallel_tomo`. Without special mention, the size of phantoms is 64-by-64, the number of pixels in the detector is 91, and the projection angles are from \( 6^\circ \) to \( 180^\circ \) with the equal space \( 6^\circ \). Then, the underdetermined rate in our inverse problem, \( Ax = b \) is 0.667. In our tests, we assume that the measurements are corrupted by additive white Gaussian noise with mean 0 and standard deviation \( \varepsilon \| A\bar{x} \|_2 \), where \( \varepsilon \) gives the noise level and \( \bar{x} \) denotes the true attenuation coefficients.

When we solve the \( x \) sub-problem, we set the maximum iteration number as 100 and the stopping rule as

\[ \frac{\| x^{n+1} - x^n \|_2}{\| x^n \|_2} \leq 10^{-4}, \]
When we solve the $\delta$ sub-problem, the parameter is set to $\tau_1 = \frac{1}{8}$, $\tau_2 = \frac{1}{8}$, $\theta = 0$. We set the maximum iteration number as 50 and the stopping rule as

$$\frac{||\delta^{m+1} - \delta^m||_2}{||\delta^m||_2} < 10^{-3}.$$ 

In order to illustrate the performance of the methods, we define the reconstruction error and segmentation error as

$$\text{recerr} = \frac{||x - \bar{x}||_2}{||x||_2}, \quad \text{segerr} = \frac{1}{N} \sum_{j \in \Omega} I(l_j - \hat{l}_j^p),$$

where $x$ is the reconstruction result, $I(\cdot)$ is the Dirac function, $l_j$ is the label of pixel $j$ in the

| Table 1. Comparison on a piecewise constant phantom |
|---------------------------------------------------|
|                                                   |
| Method in [7]          | 0.085 | 0.035 | 653        |
| Method by solving (8) | 0.096 | 0.025 | 116        |
| Method by solving (17)| 0.069 | 0.025 | 112        |

**Figure 1.** Comparison of reconstruction and segmentation results with respect to the best case on a piecewise constant phantom. Row 1: reconstruction results; row 2: segmentation results; and row 3: zoomed segmentation results in the square.
segmentation result, which is given by
\[ l_j = \arg \max_k \delta_{jk}, \tag{22} \]
and \( l_j^* \) denotes the true label for pixel \( j \). The regularization parameters in our method as well as in the method proposed in [7] are chosen as the ones which give the smallest \( \text{rec}_{err} + \text{seg}_{err} \) value.

### 4.2. Comparison on a piecewise constant phantom

We first compare our methods by solving (8) and (17) with the one proposed in [7] on an 8-class piecewise constant phantom. The noise level \( \epsilon \) is set as 0.05, and the prior information on the mean and standard deviation for each class is \( \mu_k = \frac{k-1}{7} \) and \( \sigma_k = 0.1 \) for \( k = 1, \ldots, 8 \). In our methods, we set the parameters as \( \lambda_n = 0.1, \lambda_c = 1, \gamma_1 = 1, \gamma_2 = 2, \lambda_t = 5 \).

In Table 1, we list the reconstruction error, segmentation error and CPU times (in second) for all three methods. All results in Table 1 are the average of running on 50 different noise realizations. It can be seen that the methods by solving (8) and (17) one can achieve almost the same segmentation results, which are better than the one from [7]. Comparing reconstruction error, we can see that the method by solving (8) gives the largest error,

![Figure 2. Comparison of reconstruction and segmentation results with respect to the worst case on a piecewise constant phantom. Row 1: reconstruction results and row 2: segmentation results.](image-url)
Figure 3. Comparison of $x$, $\delta$, $\phi$ in (8), (17) with $x$, $\delta$ in [7]. Row 1: the reconstructed $x$; row 2: the segmentation from $\delta$; and row 3: the segmentation from $\phi$.

which is due to not enough regularization on the reconstruction. By adding Tikhonov regularization in the model (17), we are able to obtain comparable results as the simplified model in [7]. In addition, both our methods cost similar CPU time, which is less than the one in [7] with a factor $\frac{1}{6}$.

In order to compare the results visually, in Figures 1 and 2 we show the reconstruction and segmentation results with respect to the best and worst case in these 50 tests, i.e. the results with the smallest and the largest $rec_{err} + seg_{err}$ values, respectively. It is clear that our methods provide more accurate segmentation results, which can be seen from the dark grey class in the middle square region, see the third row for zooming images. For the reconstruction results, the method in [7] gives similar results as the one from the model (17).

Here, we show the phenomenon described in Section 3.2 in Figure 3, by visually comparing the difference on $x$, $\delta$, $\phi$ in (8), (17) with $x$, $\delta$ in [7]. We treat $\phi$ as a segmentation using (22), and show it in the third row of Figure 3. Note that in the method proposed in [7], there is no variable $\phi$. As we can see, in the results from (8) the isolated points in $\phi$ marked by squares lead to isolated points in the reconstructions. After adding regularization on $x$ in (17), the number of isolated points is obviously reduced, and we can clearly see the smoothness in the reconstruction. This indicates that the regularization term is helpful for improving the reconstruction. According to the reconstruction from [7], although there is no regularization on $x$, the reconstruction is still rather smooth without isolated points. It means that the simplification of the model in [7] potentially gives smoothing regularization on the reconstruction through the segmentation $\delta$. 

Method [1]  |  Method (8)  |  Method (17)
Table 2. Comparison on a smooth phantom.

| Method                | $r_{err}$ | $s_{err}$ | CPU time (in second) |
|-----------------------|-----------|-----------|----------------------|
| Method in [7]         | 0.203     | 0.166     | 251                  |
| Method by solving (8) | 0.211     | 0.175     | 29                   |
| Method by solving (17)| 0.194     | 0.173     | 27                   |

Figure 4. Comparison of reconstruction and segmentation results with respect to the best case on a smooth phantom. Row 1: reconstruction results and row 2: segmentation results.

4.3. Comparison on a smooth phantom

In this experiment, we compare the methods on a smooth phantom with three different classes, which is much more difficult to segment compared with piecewise constant phantoms. The noise level is $\varepsilon = 0.01$, and the prior information on the mean and standard deviation for three classes are $\mu_1 = 0.16$, $\mu_2 = 0.24$, $\mu_3 = 0.565$ and $\sigma_k = 0.05$, $k = 1, 2, 3$. Other parameters are set as $\lambda_n = 123$, $\lambda_c = 0.55$, $\gamma_1 = 0.6$, $\gamma_2 = 0.6$, and $\lambda_t = 35$.

In Table 2, we list the averages of the reconstruction errors, segmentation errors and CPU times on 50 experiment tests with different noise realizations. We can see that our methods by solving (8) and (17) achieve similar segmentation results, and the method proposed in [7] provides the smallest segmentation error. Comparing the reconstruction results, it turns out that our method with model (17) gives the best reconstruction results, followed by [7]. It is obvious that Tikhonov regularization in the model (17) plays an important role, which reduces the reconstruction error significantly. In addition, our methods cost only $\frac{1}{7}$ CPU time compared with the method with the simplified model.

In Figures 4 and 5 we show the reconstruction and segmentation results from three methods under the best case and the worst case. The three methods generate comparable results. As we can see, there are no big differences between the results of the three methods.
Figure 5. Comparison of reconstruction and segmentation results with respect to the worst case on a smooth phantom. Row 1: reconstruction results and row 2: segmentation results.

Table 3. Comparison of CPU time (in second) on a piecewise phantom.

| Phantom resolution | 64 × 64 | 128 × 128 | 256 × 256 | 512 × 512 |
|--------------------|---------|-----------|-----------|-----------|
| Projection angle    | 6° : 6° : 180° | 3° : 3° : 180° | 1.5° : 1.5° : 180° | 0.75° : 0.75° : 180° |
| Method in [7]       | 653     | 1194      | 4357      | –         |
| Method with (8)     | 116     | 679       | 2292      | 14506     |
| Method with (17)    | 112     | 620       | 1972      | 14596     |

4.4. Comparison of CPU times on different resolutions

In this section, we compare our methods with the method in [7] with respect to CPU times. To do so, we generate the piecewise constant phantom with four different resolutions, then adjust the number of the projection angles such that the underdetermined rate is still kept as 0.667. The test results are the averages on three different noise realizations.

In Table 3, we list the CPU times for all three methods. Note that in the case of 512-by-512 phantom, we could not apply the method in [7] due to heavy computational cost and limited memory. It is obvious that both our methods cost much less CPU times than the one in [7]. When the resolution increases, our methods with the models (8) and (17) utilize more and more computing time compared to the one in [7], and the ratio of CPU times by our methods are around $\frac{1}{6}$, $\frac{1}{2}$, $\frac{1}{2}$, respectively.

5. Conclusion

In this paper, we propose a new method for simultaneous reconstruction and segmentation in X-ray CT application. By using the commutativity of log-sum operations, the original model proposed in [7] could be solved more efficiently. Although one more variable is introduced, the energy function becomes separable and each sub-problem is convex and easy-to-solve. We also show some convergence results, and numerically discover the role of
the simplification steps in [7]. Numerical results show that the proposed methods provide comparable reconstruction and segmentation results with much less CPU time. Further validating our methods on real data will be a very important direction for future research.

**Acknowledgments**

We thank Hans Martin Kjer from Technical University of Denmark for providing us the codes for the method introduced in [7].

**Disclosure statement**

No potential conflict of interest was reported by the authors.

**Funding**

This work was supported by National Natural Science Foundation of China[11531013,11871035], Villum Fonden[25893] and Recruitment Program of Global Young Expert.

**References**

[1] Kuchment P. The radon transform and medical imaging, Vol. 85. Philadelphia: SIAM; 2014.
[2] Kak A, Slaney M. Principles of computerized tomographic imaging. New York: IEEE Press; 1988.
[3] Hansen P, Saxild-Hansen M. AIR tools-a MATLAB package of algebraic iterative reconstruction methods. J Comput Appl Math. 2012;236(8):2167–2178.
[4] Herman GT, Kuba A. Advances in discrete tomography and its applications. Boston, MA: Birkhauser; 2007.
[5] Batenburg K, Sijbers J. DART: a practical reconstruction algorithm for discrete tomography. IEEE Trans Image Process. 2011;20(9):2542–2553.
[6] Varga L, Balázs P, Nagy A. An energy minimization reconstruction algorithm for multivalued discrete tomography. In CompIMAGE, 2012. p. 179–185.
[7] Romanov M, Dahl AB, Dong Y, Hansen PC. Simultaneous tomographic reconstruction and segmentation with class priors. Inverse Probl Sci Eng. 2016;24(8):1432–1453.
[8] Ramlau R, Ring W. A Mumford–Shah level-set approach for the inversion and segmentation of X-ray tomography data. J Comput Phys. 2007;221(2):539–557.
[9] Marroquín J, Santana E, Botello S. Hidden Markov measure field models for image segmentation. IEEE Trans Pattern Anal Mach Intell. 2003;25(11):1380–1387.
[10] Kjer HM, Dong Y, Hansen PC. User-friendly simultaneous tomographic reconstruction and segmentation with class priors. In: Lauze F, Dong Y, Dahl AB, editors. Scale Space and Variational Methods in Computer Vision; Cham. Springer International Publishing; 2017. p. 260–270.
[11] Mumford D, Shah J. Optimal approximations by piecewise smooth functions and associated variational problems. Commun Pure Appl Math. 1989;42(5):577–685.
[12] de Sompele DV, Brady M. Simultaneous reconstruction and segmentation algorithm for positron emission tomography and transmission tomography. In: Biomedical Imaging: From Nano to Macro, 2008. ISBI 2008. 5th IEEE International Symposium. Silver Spring (MD): IEEE; 2008. p. 1035–1038
[13] Ramlau EKR, Ring W. A Mumford–Shah level-set approach for the inversion and segmentation of spect/ct data. Inverse Problems Imaging. 2011;5(1):137–166.
[14] Potts R. Some generalized order-disorder transformations. In Mathematical Proceedings of the Cambridge Philosophical Society; Vol. 48; Cambridge: Cambridge University Press; 1952. p. 106–109
[15] Storath M, Weinmann A, Friel J, Unser M. Joint image reconstruction and segmentation using the Potts model. Inverse Probl. 2015;31(2):025003.

[16] Lauze F, Quéau Y, Plenge E. Simultaneous reconstruction and segmentation of CT scans with shadowed data. In: Lauze F, Dong Y, Dahl AB, editors. Scale Space and Variational Methods in Computer Vision. Cham: Springer International Publishing; 2017. p. 308–319

[17] Dong Y, Hansen PC, Kjer HM. Joint CT reconstruction and segmentation with discriminative dictionary learning. IEEE Trans Comput Imaging. 2018;4(4):528–536.

[18] Kolmogorov V, Zabin R. What energy functions can be minimized via graph cuts? IEEE Trans Pattern Anal Mach Intell. 2004;26(2):147–159.

[19] Maeda SI, Fukuda W, Kanemura A, Ishii S. Maximum a posteriori x-ray computed tomography using graph cuts. In: Journal of Physics: Conference Series; Vol. 233; IOP Publishing. 2010.

[20] Corona V, Benning M, Ehrhardt MJ, Gladden LF, Mair R, Reci A, Sederman AJ, Reichelt S, Schönlieb CB. Enhancing joint reconstruction and segmentation with non-convex Bregman iteration. Inverse Probl. 2019;35(5):055001.

[21] Hsiao T, Rangarajan A, Gindi G. Joint-map bayesian tomographic reconstruction with a gamma-mixture prior. IEEE Trans Image Process. 2002;11(12):1466–1477.

[22] Lemmens C, Faul D, Nuyts J. Suppression of metal artifacts in CT using a reconstruction procedure that combines map and projection completion. IEEE Trans Med Imaging. 2008;28(2):250–260.

[23] Tikhonov AN. On the stability of inverse problems. Dokl Akad Nauk SSSR; 1943; 39: 195–198.

[24] Rudin L, Osher S, Fatemi E. Nonlinear total variation based noise removal algorithms. Physica D: Nonlinear Phenomena. 1992;60(1):259–268.

[25] Teboulle M. A unified continuous optimization framework for center-based clustering methods. J Mach Learn Res. 2007;8:65–102.

[26] Liu J, Tai XC, Huang H, Huan Z. A weighted dictionary learning model for denoising images corrupted by mixed noise. IEEE Trans Image Process. 2013;22(3):1108–1120.

[27] Csiszár I, Tusnády G. Information geometry and alternating minimization procedures. Stat Decis. 1984;1:205–237.

[28] Bertsekas DP. Nonlinear programming, 2nd ed. Belmont (MA): Athena Scientific; 1999.

[29] Vogel C, Oman M. Iterative methods for total variation denoising. SIAM J Sci Comput. 1996;17(1):227–238.

[30] Bjorck A. Numerical methods for least squares problems, Vol. 51. Philadelphia: SIAM; 1996.

[31] Chambolle A, Pock T. A first-order primal-dual algorithm for convex problems with applications to imaging. J Math Imaging Vis. 2011;40(1):120–145.

[32] Chen Y, Ye X. Projection onto a simplex. arXiv preprint arXiv:110160812011.

[33] Tseng P. Convergence of a block coordinate descent method for nondifferentiable minimization. J Optim Theory Appl. 2001;109(3):475–494.

Appendix

All the following definitions and theorem come from [33].

Let \( f \) have the following form:

\[
\min f(x_1, \ldots, x_n) = f_0(x_1, \ldots, x_n) + \sum_{k=1}^{N} f_k(x_k),
\]

where \( f_0 : \mathbb{R}^{n_1+\ldots+n_N} \to \mathbb{R} \cup \{\infty\}, f_k : \mathbb{R}^{n_k} \to \mathbb{R} \cup \{\infty\}. \) \( f \) is proper.

**Definition A.1:** Block coordinate descent (BCD) method:

- **Initialization.** Choose any \( x^0 = (x_{1}^0, \ldots, x_{N}^0) \in domf \).
- **Iteration** \( r + 1 \) with \( r \geq 0 \). Given \( x^r = (x_{1}^r, \ldots, x_{N}^r) \in domf \), choose an index \( s \in \{1, \ldots, N\} \) and compute a new iterate

\[
x^{r+1} = (x_{1}^{r+1}, \ldots, x_{s}^{r+1}, \ldots, x_{N}^{r+1}) \in domf,
\]
satisfying
\[ x_{r+1}^s \in \arg \min_{x_s} f(x_1^r, \ldots, x_{r-1}, x_s, x_{r+1}, \ldots, x_N^r), \]
\[ x_j^{r+1} = x_j^r, \forall j \neq s. \]

**Definition A.2 (Essentially cyclic rule):** There exists a constant \( T \geq N \) such that every index \( s \in \{1, \ldots, N\} \) is chosen at least once between the \( r \)th iteration and the \((r + T - 1)\)th iteration, for all \( r \).

**Definition A.3:** More assumptions about \( f, f_0, \ldots, f_N \).

- (B1) \( f_0 \) is continuous on \( \text{dom} f_0 \).
- (B2) For each \( k \in \{1, \ldots, N\} \) and \( (x_j)_{j \neq k} \), the function \( x_k \to f(x_1, \ldots, x_N) \) is quasiconvex and hemivariate.
- (B3) \( f_0, f_1, \ldots, f_N \) are lsc.
- (C1) \( \text{dom} f_0 \) is open and \( f_0 \) tends to \( \infty \) at every boundary point of \( \text{dom} f_0 \).
- (C2) \( \text{dom} f_0 = Y_1 \times \cdots \times Y_N \), for some \( Y_k \subseteq \mathbb{R}^{n_k}, k = 1, \ldots, N \).
- \( h \) is quasiconvex if
  \[ h(x + \lambda d) \leq \max\{h(x), h(x + d)\}, \forall x, d, \lambda \in [0, 1]. \]
- \( h \) is hemivariate if \( h \) is not constant on any line segment belonging to \( \text{dom} h \).
- \( z \) is a coordinatewise minimum point of \( f \), if \( z \in \text{dom} f \) and
  \[ f(z + (0, \ldots, d_k, \ldots, 0)) \geq f(z), \forall d_k \in \mathbb{R}^{n_k}, \forall k. \]
- if \( h \) is strictly convex, then \( h \) is quasiconvex and hemivariate.

**Theorem A.1 (Theorem 5.1 in [33]):** Suppose that \( f, f_0, \ldots, f_N \) satisfy Assumptions B1, B2, B3 and that \( f_0 \) satisfies either Assumption C1 or C2. Also, assume that the sequence \( \{x^r = (x_1^r, \ldots, x_N^r)\}_{r=0,1,\ldots} \) generated by the BCD method using the essentially cyclic rule is defined. Then, either \( \{f(x^r)\} \to -\infty \), or else every cluster point \( z = (z_1, \ldots, z_N) \) is a coordinatewise minimum point of \( f \).