# Low Dose CT Reconstruction via Edge-preserving Total Variation Regularization

**Zhen Tian**[1,3,5], **Xun Jia**[2,3,5], **Kehong Yuan**[1], **Tinsu Pan**[4], and **Steve B. Jiang**[2,3]

Steve B. Jiang: sbjiang@ucsd.edu

[1]Department of Biomedical Engineering, Graduate School at Shenzhen, Tsinghua University, Shenzhen, Guangdong 518055, China

[2]Department of Radiation Oncology, University of California San Diego, La Jolla, CA 92093, USA

[3]Center for Advanced Radiotherapy Technologies, University of California San Diego, La Jolla, CA 92093, USA

[4]Department of Imaging Physics, University of Texas M. D. Anderson Cancer Center, Houston, Texas 77030, USA

## Abstract

High radiation dose in CT scans increases a lifetime risk of cancer and has become a major clinical concern. Recently, iterative reconstruction algorithms with Total Variation (TV) regularization have been developed to reconstruct CT images from highly undersampled data acquired at low mAs levels in order to reduce the imaging dose. Nonetheless, the low contrast structures tend to be smoothed out by the TV regularization, posing a great challenge for the TV method. To solve this problem, in this work we develop an iterative CT reconstruction algorithm with edge-preserving TV regularization to reconstruct CT images from highly undersampled data obtained at low mAs levels. The CT image is reconstructed by minimizing an energy consisting of an edge-preserving TV norm and a data fidelity term posed by the x-ray projections. The edge-preserving TV term is proposed to preferentially perform smoothing only on non-edge part of the image in order to better preserve the edges, which is realized by introducing a penalty weight to the original total variation norm. During the reconstruction process, the pixels at edges would be gradually identified and given small penalty weight. Our iterative algorithm is implemented on GPU to improve its speed. We test our reconstruction algorithm on a digital NCAT phantom, a physical chest phantom, and a Catphan phantom. Reconstruction results from a conventional FBP algorithm and a TV regularization method without edge preserving penalty are also presented for comparison purpose. The experimental results illustrate that both TV-based algorithm and our edge-preserving TV algorithm outperform the conventional FBP algorithm in suppressing the streaking artifacts and image noise under the low dose context. Our edge-preserving algorithm is superior to the TV-based algorithm in that it can preserve more information of low contrast structures and therefore maintain acceptable spatial resolution.

---

[5]Zhen Tian and Xun Jia have contributed equally to this work and should be considered co-first authors.

## 1. Introduction

X-ray computed tomography (CT) has been extensively used in clinics to provide patient volumetric images for a number of purposes nowadays. However, by its nature, CT scans expose a high x-ray radiation dose to the patient which may result in a non-negligible lifetime risk of cancer (Hall and Brenner, 2008; de Gonzalez *et al.*, 2009; Smith-Bindman *et al.*, 2009). This fact has become a major concern for the clinical applications of CT scans, particularly for pediatric patients, who are more sensitive to radiation and have a longer life expectancy than adults (Brenner *et al.*, 2001; Brody *et al.*, 2007; Chodick *et al.*, 2007). Therefore, it is highly desirable to reduce CT imaging dose while maintaining clinically acceptable image quality.

A simple way to reduce the x-ray dose is to lower mAs levels in CT data acquisition protocols. Nonetheless, this approach will result in an insufficient number of x-ray photons detected at imager and hence elevate the quantum noise level on the sinogram. As a consequence, the quality of the CT images reconstructed from a conventional filtered backprojection (FBP) algorithm (Deans, 1983) will be degraded by the noise-contaminated sinogram data. Another way to reduce imaging dose is to decrease the number of x-ray projections acquired by, *e.g.*, operating the x-ray generator in a high-frequency pulsed model in future with hardware modification. Yet, this will cause serious streaking artifacts in the reconstructed CT images, as the FBP algorithms require that the number of projections should satisfy the Shannon sampling theorem (Jerri, 1977).

Recently, compressed sensing algorithms (Donoho, 2006) have been applied to the CT reconstruction problem. In particular, Total Variation (TV) methods (Rudin *et al.*, 1992) have presented their tremendous power in CT reconstruction with only a few x-ray projections (Sidky *et al.*, 2006; Song *et al.*, 2007; Chen *et al.*, 2008; Sidky and Pan, 2008). In such approaches, an energy function of a TV form is minimized subject to a data fidelity condition posed by the x-ray projections. Since this energy term corresponds to image gradient, minimizing it will effectively remove those high spatial gradient parts such as noise and streaking artifacts in the reconstructed CT images. One disadvantage of this TV approach is its tendency to uniformly penalize the image gradient irrespective of the underlying image structures. As a result, edges, especially those of low contrast regions, are sometimes over smoothed, leading to the smoothed edge to some extent and loss of low contrast information. To resolve this issue, an edge guided compressive sensing reconstruction algorithm has recently been proposed in solving an MRI image reconstruction problem (Guo and Yin, 2010). In such an approach, edges are detected during the reconstruction process and are purposely excluded from being smoothed by the TV norm, by which the reconstruction can be speed up. Though this method can be easily generalized to the CT reconstruction problems, when it comes to the low dose case where insufficient or/and noisy x-ray projections are used in the reconstruction, a high quality edge detection algorithm is required to avoid incorrect detection of those fake edges caused by image noise and streaking artifacts. The sophisticated edge detection algorithms usually pose a high computation burden in the reconstructions. Moreover, they may require fine tuning of some case dependent parameters, making it hard to control their efficacy.

Another disadvantage of the TV-based reconstruction methods are the associated time consuming computation process due to their iterative nature. Generally speaking, the prolonged computational time makes the iterative CT reconstruction approaches prohibitive in many routine clinical applications. Recently, high-performance graphics processing units (GPUs) have been reported to speed up heavy duty computational tasks in medical physics, such as CBCT reconstruction (Xu and Mueller, 2005, 2007; Li *et al.*, 2007; Yan *et al.*, 2008; Jia *et al.*, 2010b), deformable image registration (Sharp *et al.*, 2007; Samant *et al.*, 2008; Gu *et al.*, 2010), dose calculation (Gu *et al.*, 2009; Hissoiny *et al.*, 2009; Jia *et al.*, 2010a) and treatment plan optimization (Men *et al.*, 2009; Men *et al.*, 2010). In principle, high computation efficiency can be expected utilizing GPU in our CT reconstruction problem.

In this work, we generalize the TV-based CT reconstruction algorithm to an edge-preserving TV (EPTV) regularization form to reconstruct CT images under undersampling and/or low mAs situation. In particular, an EPTV norm is designed by introducing a penalty weight to the original TV norm, which enables the algorithm to automatically locate the sharp discontinuities of image intensity and adjust the weight adaptively to adopt the progressively recovered edge information during the reconstruction process. This regularization term automatically ensures that less smoothing is performed on edges to better preserve the edges, as will be seen in below. Our reconstruction algorithm is implemented on GPU to speed up the computation process.

## 2. Methods

### 2.1 Reconstruction Model

CT projection can be mathematically formulated as a linear equation,

$$Pf = y, \quad (1)$$

where *f* is a vector whose entries correspond to the x-ray linear attenuation coefficients at different voxels of the patient image. *P* is a projection matrix in fan-beam geometry. Numerically, we generate its element $p_{i,j}$ by the length of the intersection of the x-ray *i* with the pixel *j*, which can be explicitly computed by Siddon's fast ray tracing algorithm (Siddon, 1985). The vector *y* represents the log-transformed projection data measured on image detectors at various projection angles. A CT reconstruction problem is formulated as to retrieve the unknown vector *f* based on the projection matrix *P* and the observation vector *y*. When it comes to an undersampled problem where inadequate projection data is used to reconstruct the CT image, the problem become underdetermined and there exist infinitely many solutions to the Eq. (1).

As opposed to solve the linear equation directly, the CT image can be reconstructed by minimizing an energy function with a TV regularization term:

$$f = \operatorname{argmin}_f E(f) = \operatorname{argmin}_f \frac{\mu}{2} \|Pf - y\|_2^2 + J(f), s.t. f_i \geq 0 \text{ for } \forall i, \quad (2)$$

where $\|\cdot\|_n$ denotes the $l_n$ vector norm in the imager vector space. In Eq. (2), the first term is known as a data fidelity term, which ensures the consistency between the reconstructed image $f$ and the measurement $y$. The second one is a regularization term, which is chosen to be a TV semi-norm. The introduction of the TV term in this optimization process differentiates those infinite many solutions to the Eq. (1) and picks out the one with desired image properties as the reconstructed image. Specifically, the TV term is defined as

$$J(f) = \|f\|_{TV} = \int \|\nabla f(x)\|_1 \mathrm{d}x, \quad (3)$$

where $\nabla f(x)$ represents the gradient of an image $f$ at a pixel $x$. The TV term has been shown to be robust to remove noise and artifacts in the reconstructed image $f$ (Sidky and Pan, 2008; Jia *et al.*, 2010b). A scalar $\mu$ is introduced to adjust the relative weights between the data fidelity term and the regularization term. This parameter depends on different factors, such as the complexity of the image content and the extent of the consistency between the measured data and the ground truth image. In this paper we choose it manually to yield a good reconstruction quality.

Despite the great success of the TV model in terms of reconstructing high quality CT images, edges around low contrast regions are sometimes oversmoothed and low contrast information is lost as a consequence. To overcome this limitation, we propose an EPTV regularization term by introducing a penalty weight vector $\omega$ in defining the TV term, namely

$$J(f) = \|f\|_{EPTV} = \int \omega(f(x)) \|\nabla f(x)\|_1 \mathrm{d}x. \quad (4)$$

$$\omega(f(x)) = \exp\left[-(\|\nabla f(x)\|_1/\sigma)^2\right]. \quad (5)$$

The parameter $\sigma$ controls the amount of smoothing that we would like to apply to those pixels at edges, especially the low contrast edges, relative to those non-edge pixels. Apparently, the choice of $\sigma$ is of central importance for the algorithm. A large $\sigma$ is not able to differentiate image gradients at different pixels. In such a circumstance, the algorithm becomes essentially the TV method. In contrast, small $\sigma$ tends to give small weights to almost every pixel, making the EPTV norm inefficient in removing noise or streaking artifacts. Since the gradient values changes during the iterative reconstruction process, we propose to adaptively set the value of $\sigma$ according to the histogram of the gradient magnitude, so that a certain percentage pixels have the gradient values larger than $\sigma$. The specific percentage is dependent on the image complexity, that is, the complex images with more structures and edges should have a larger percentage than the simple images with fewer structures. In particular, for the test cases in this paper, the percentage is chosen 90%~95% as we found this choice is a good balance between reconstruction efficiency and image quality. Besides, during the reconstruction process, the pixels at edges would be gradually picked out and given small penalty weight that could be neglected, which makes the EPTV term sparser and speeds up the implementation.

## 2.2 Optimization Approach

Since the cost function defined in this minimization problem is convex, it is sufficient to consider the optimality condition:

$$\mu P^T(Pf-y)+\frac{\delta}{\delta f}\|f\|_{EPTV}=0, \quad (6)$$

where $P^T$ denotes the transposition matrix of $P$. By introducing a scalar parameter $\lambda > 0$ and a vector $v$, we can split the Eq. (6) as

$$\mu P^T(Pf-y)=\lambda(f-v),$$
$$\frac{\delta}{\delta f}\|f\|_{EPTV}=\lambda(v-f). \quad (7)$$

This inspires us that the reconstruction problem can be solved by iteratively performing the following two steps:

$$(P1): v=f-\frac{\mu}{\lambda}P^T(Pf-y),$$
$$(P2): f=\text{argmin}\left(\frac{\lambda}{2}\|f-\nu\|_2^2+\|f\|_{EPTV}\right), \quad (8)$$

The meaning of these two steps is straightforward. In each iteration, we first obtain a trial solution in (P1) constrained by the data fidelity term. This trial solution may contain noise and artifact, since no regularization is performed. Then the subproblem (P2) utilizes the EPTV term to effectively remove those noise and artifacts, improving image quality. This two step iteration is actually the so-called forward-backward splitting algorithm, whose mathematical properties, such as convergence, has been discussed in (Combettes and Wajs, 2005; Hale *et al.*, 2008). Xu and Mueller proposed a similar two-step reconstruction algorithm, where OS-SIRT is used to obtain a trial solution and the image enhancement is achieved by using bilateral filter or TV minimization (Xu and Mueller, 2009b).

The subproblem (P1) in Eq. (8) is a gradient descent update with a step size of $\mu/(2\lambda)$ for a minimization problem $v=\text{argmin}_f\|Pf-y\|_2^2$. The introduction of the parameter $\lambda$ in the algorithm is to ensure a small step size for numerical stability consideration. In this work, we modify the subproblem (P1) into using a conjugate gradient least square (CGLS) method (Hestenes and Stiefel, 1952) to solve the minimization problem instead of the gradient descent update. The skeleton of the CGLS method is presented in Appendix I. We found that the efficiency of the algorithm has been improved with this modification, yet the mathematical proof of its convergence is still needed. The minimization problem (P2) is solved with a simple gradient descent (GD) method in an iterative manner. At each iteration, the gradient direction $g$ is first numerically calculated. An inexact line search is then performed along the negative gradient direction and a step size is determined according to Amijo's rule (Bazaraa *et al.*, 2006). The solution is then updated accordingly. The penalty weight is also updated accordingly at each iteration. Specifically, the gradient direction $g$ is calculated as:

$$g = \lambda(f - v^{(k)}) + \frac{\delta}{\delta f}\|f\|_{EPTV}; \quad (9)$$

$$\frac{\delta}{\delta f}\|f\|_{EPTV} = -\nabla\frac{\omega(f(x))\nabla f(x)}{\|f(x)\|_{TV}} \quad (10)$$

Numerically, we approximate the functional variation of EPTV, $\frac{\delta}{\delta f}\|f\|_{EPTV}$, by a symmetric finite difference scheme to ensure the stability of the algorithm. The detailed derivation and the approximation of the variation of EPTV are described in Appendix II. Moreover, since physically those pixel values are x-ray attenuation coefficients and has to be positive, we have also ensure this condition by a simple truncation of those negative pixel values of the reconstructed images in each iteration. In summary, the EPTV algorithm is implemented as follows.

EPTV Algorithm:

---

Initialize $f^{(0)} = 0$. For $k = 1,2,\ldots$, do the following steps until convergence.

1. Solve (P1): $v^{(k+1)} = \operatorname{argmin}_f \|Pf - y\|_2^2$ with an initial value $f^{(k)}$ using CGLS solver.

2. Solve (P2): $f^{(k+1)} = \operatorname{argmin}_f \frac{\lambda}{2}\|f - v^{(k+1)}\|_2^2 + \|f\|_{EPTV}$ with GD method, and $\omega$ is updated at each iteration in this step,

3. Ensuring image positivity: $f^{(k+1)} = 0$ if $f^{(k+1)} < 0$.

---

The number of iterations used in our algorithm is case dependent, as it varies with the number of projections used for reconstruction, the noise level due to the mAs level used in scanning and so on. Generally, the fewer projections are used, the more obvious streak artifacts are yielded due to undersampling, the more iterations are required to suppress these streaks. Similarly, the lower mAs level is used, the more iterations are needed to remove the noise.

## 2.3 Details in CUDA implementation

Our GPU-based reconstruction code was developed under the Compute Unified Device Architecture (CUDA) programming environment and GPU hardware platform. This platform enables a number of tasks implemented in parallel on different CUDA threads simultaneously, which speeds up the performance of our reconstruction algorithm.

There are two strategies to implement the Step1 in Algorithm1 on GPU according to the size of the projection matrix $P$. For the cases with a small image size and a small amount of projections where the projection matrix $P$ and its transposition matrix $P^T$ are small enough to be stored in GPU memory, the strategy is to pre-calculate the projection matrix and save it and its transposition in GPU as a lookup table. Then CGLS method used in Step 1 becomes a set of matrix and vector operations, such as sparse matrix-vector multiplication, vector addition, scalar-vector operation and so on, which can be efficiently performed on GPU by

adopting the fast GPU sparse matrix-vector multiplication (Bell and Garland, 2008) and CUBLAS Library (NVIDIA, 2008). For the cases where the projection matrix and its transposition are too large to be loaded on GPU, especially when cone-beam geometry is used, the strategy is to directly compute the forms $y = Pf$ and $f = P^T y$ repeatedly whenever needed. The former $y = Pf$ can be easily performed in parallel on GPU by making each thread responsible for one ray line using an improved Siddon's ray tracing algorithm (Siddon, 1985). The latter $f = P^T y$ is actually a backward projection in that it maps the projection $y$ on the detector back to the slice image $f$ by updating its pixel values along all the ray lines. It can be calculated by still using the Siddon's algorithm on GPU with each thread responsible for updating voxels along a ray line. However, this operation would cause a memory writing conflict problem due to the possibility of simultaneously updating a same pixel value by different GPU threads. When this conflict occurs, one thread will have to wait until other threads finish updating. It is this fact that severely limits the maximal utilization of GPU's massive parallel computing power. To solve this issue, a GPU-friendly backward-projection method is developed as follows:

$$\left[ P^T y \right](u, v) = \frac{\Delta u \Delta v}{\Delta \beta} \cdot \sum_\theta \frac{y^\theta(\beta^*)}{l(u^*, v^*)}; \quad (11)$$

where $(u^*, v^*)$ is one pixel on the reconstructed CT slice, and $l(u^*, v^*)$ is the distance from the x-ray source to the pixel $(u^*, v^*)$. $\beta^*$ is the angular coordinate of the rayline connecting the x-ray source and the pixel $(u^*, v^*)$ at the projection angle $\theta$. $y^\theta(\beta^*)$ is the corresponding detector reading. $\Delta u$ and $\Delta v$ are the size of pixels on CT slices, and $\Delta \beta$ is the angular spacing of the detector units. The derivation of Eq. (11) is briefly shown in the Appendix III. By this backward projection method, we can simply get the value of $f$ at a given pixel $(u^*, v^*)$ by summing the projection values $y^\theta(\beta^*)$ over all projection angles $\theta$ after proper geometry correction, which allows us to implement the calculation in parallel on GPU with each thread responsible for one pixel and thus to avoid the memory writing conflict problem. In numerical computation, since we always evaluate $y^\theta(\beta)$ at a set of discrete coordinates and $\beta^*$ does not necessarily coincide with these discrete coordinates, a linear interpolation is performed to obtain $y^\theta(\beta^*)$. In this paper, we adopt the second strategy for all the cases in order to keep the algorithm uniform.

The main operations in the Step 2 are calculating the gradient direction, descending along the direction, and updating the penalty weight *etc.* All of them can be easily implemented in parallel on GPU with each thread computing one entry. The Step 3 can be similarly processed. The flow chart of our reconstruction algorithm is shown in Fig. 1.

## 3. Experimental Results

We tested our reconstruction algorithm on three cases: a digital NURBS-based cardiac-troso (NCAT) phantom in thorax region (Segars, 2002), a physical chest phantom, and a Catphan phantom. In digital phantom experiments, we simulated x-ray projections using Siddon's ray tracing algorithm (Siddon, 1985) in fan-beam geometry with an arc detector of 888 units and a spacing of 1.0239mm. The source to detector distance is 949.075mm and the source to rotation center distance is 541.0mm. All of these parameters mimic a realistic configuration of a GE Lightspeed QX/I CT scanner. In all cases, we simulated/acquired x-ray projection

data along 984 directions equally spaced in a full rotation and a subset of them were used for reconstruction. The size of reconstructed images is 512×512.

### 3.1 Digital phantom experiment

We first tested our EPTV algorithm on an NCAT phantom in thorax region using undersampled x-ray projections. In order to better illustrate how our EPTV algorithm performs, in Fig. 2 we show the penalty weights and the reconstructed image at different iterations during the reconstruction with 40 projections being used. During the evolution, the algorithm automatically picks out the edges and gave them small weight $w$ in the EPTV term to avoid being oversmoothed, as shown in Fig. 2 (2nd row). Accordingly, the structural information is gradually resolved in the reconstructed image, with main structures and obvious artifacts at first and low contrast structures and fewer artifacts later, shown in Fig.2 (3rd row). Note that there are some false edges detected at iteration 5, which are gradually removed with more iterations, leading to visually better reconstructed images.

The reconstructed images using 40 projections with 50 iterations are shown in Fig. 3. We can see that the conventional FBP algorithm is not able to reconstruct the CT images with such few projections and obvious streaking artifacts are observed, which makes the image unacceptable. In contrast, even with such few projections, both the TV method and the EPTV method can still capture most of the structures, leading to visually much better reconstruction results. However, one disadvantage of TV based method, including our EPTV method, is the potential missing of small structures in the reconstructed images and degradation of image resolution due to the very few projections used. For instance, one very small dot-like structure in the right side of the lung below the heart is missing in both Fig. 3(c) and (d).

As for the comparison between the TV method and the EPTV method, the spinal bone structures of low contrast indicated by the arrow in Fig. 3(c) and (d) are hardly resolved by the TV method, while much clearer structures with sharper edges are observed in the image reconstructed by the EPTV method. To better compare the TV method and the EPTV method in detail, we also show the horizontal and vertical intensity profiles going through the center of the reconstructed images. Clearly, large fluctuations at non-edge points are observed in the profiles obtained from the TV method. Besides, the jumps indicated by arrows are broadened and weakened, indicating blurring effects and losing of contrast. On the other hand, the profiles of EPTV are very close to that of the ground truth with sharp jumps at edge points and small fluctuation at non-edge points. All of these clearly demonstrate the advantages of the EPTV algorithm in resolving the low contrast edges over the TV algorithm within the same number of iterations. It should be noted that the TV method can produce very accurate reconstruction results given enough number of iterations. For example, Sidky et al. (Sidky et al., 2006) have shown a very nice match of the profile in their numerical example of their TV algorithm. Yet, the convergence speed of an algorithm is an important factor that determines its practicality. Comparing two iterative reconstruction algorithms, the one that can produce higher image quality with a same amount of computation time is more favorable. In the comparison shown in Fig. 3, we compare the TV method and the EPTV method with the same number of iterations, *i.e.*, 50, which is enough

for EPTV but not for TV. This actually shows the improvement of our EPTV method over the original TV method in terms of convergence speed.

To quantify the reconstruction accuracy of the algorithm, we use relative error as a metric to measure the similarity between the ground truth image and reconstructed image. Besides, since edge information is an important feature of images and human vision is highly sensitive to it, edge cross-correlation coefficient (ECC) (Xu and Mueller, 2009a) is also used as another metric to evaluate the reconstruction accuracy. These metrics are calculated as follows:

$$e = \frac{\|f - f^*\|_2^2}{\|f^*\|_2^2} \quad (12)$$

$$c = \frac{\sum_i (b_i - \bar{b})(b_i^* - \bar{b}^*)}{\sqrt{\sum_i (b_i - \bar{b})^2 \sum_j (b_i^* - \bar{b}^*)^2}} \quad (13)$$

where $f^*$ is the ground truth image and $b$ is a binary image in which 1 indexes the edge points on the reconstructed image and 0 indexes others. The edges are detected using a standard Sobel edge detector (Kanopoulos *et al.*, 1988). $b^*$ is defined in a similar manner for the ground truth image. The over-bar indicates an average of the corresponding quantities over all pixels.

For a fair comparison, the parameter $\mu$ in both the TV and the EPTV methods are carefully tuned manually, so that the best qualities indicated by the smallest error level $e$ are obtained from each of them. The relative error $e$ and the edge correlation coefficient $c$ of the reconstructed images of the digital phantom are shown as functions of number of projections and number of iterations, respectively, in Fig. 4. As expected, the more projections used, the better reconstruction quality will be obtained with smaller relative error and higher edge correlation coefficient. Fig. 4(a) and (b) clearly demonstrate that with the same number of projections, the EPTV method is superior to the TV method with a smaller error and higher edge cross-correlation. In this particular case, 40 projections are sufficient for EPTV reconstruction to clearly resolve the low-contrast structures. The evolution curves of the relative error and the edge correlation during iterations in Fig. 4(c) and (d) show that the EPTV algorithm converges much faster than the TV method. With these curves, we can conclude that the EPTV method outperforms the TV method, in that it leads to smaller relative error, higher correlation, and faster convergence for a given number of projections.

### 3.2 Physical phantom experiments

In this paper, we scanned both a physical chest phantom and a physical Catphan phantom under high dose and low dose protocols. The experiment on the chest phantom was designed to evaluate the tolerance of the EPTV algorithm to image noise due to low imaging dose, while the experiment on the Catphan phantom aimed to evaluate the performance of our algorithm in terms of spatial resolution. The parameter $\mu$ used in the TV method and the EPTV method are adjusted manually so that visually the best image quality is obtained.

In the experiment on the chest phantom, for the high dose protocol, 984 projections with 0.4mAs/projection were acquired and then used to reconstruct the image with the FBP algorithm. In the low dose protocol, the chest phantom was scanned at 0.02mAs/projection and only 200 equally spaced projections are used for reconstruction. The dose in the low dose protocol is about 1% of that in the high dose protocol. The reconstruction results with 30 iterations are shown in Fig. 5. Two different display windows are used to stress the bones and the details in the lung, respectively. We can see that using FBP in such a low dose situation, the image quality deteriorates greatly with obvious streaking artifacts and image noise. In contrast, images reconstructed by TV and EPTV under the same low dose context show better image quality with less noise and streaking artifacts. While comparing TV and EPTV methods, the latter is found to be superior on preserving the edges of fine structures. The bones are much clearer with a hollow in the image reconstructed by EPTV than that in the image reconstructed by TV. Although the cloud-like and low contrast details in the lung region in Fig. 5(b1) are great challenges to be reconstructed for both the TV algorithm and the EPTV algorithm, they are better preserved by the latter. It is worth noticing that under this low dose protocol, image degradation occurs in all of the three algorithms. But the artifacts in the FBP result and those in the TV or the EPTV results are quite different. For FBP, it leads to serious noise and streaks of high spatial frequency. This may lead to a visual impression that the image contains more detailed structures, especially inside the lung as in (b2). On the other hand, it is well known that TV-based algorithms suffer from edge blurring artifacts. To better demonstrate this point, we have also presented in Fig 5(c2)~(c4) the difference images inside the lung area between the reconstruction results and the ground truth image assumed to be the high dose reconstruction results in Fig. (a1). Apparently, these figures indicate that the FBP result is contaminated by a high amount of high spatial frequency noise signal over the entire lung area. On the other hand, the TV based algorithms can suppress this noise contamination considerably, leading to relatively small error level inside the lung area, though a large error around edges, especially around the rib bones, is clearly observed.

To further quantify the reconstruction accuracy, the relative error of the whole image $e_{whole}$ and that of the lung region $e_{lung}$ are calculated are as follows:

$$e_{whole} = \frac{\|f - f^*\|_2^2}{\|f^*\|_2^2}, \quad (14)$$

$$e_{lung} = \frac{\|f_{lung} - f^*_{lung}\|_2^2}{\|f^*_{lung}\|_2^2} \quad (15)$$

where, $f^*$ is the ground truth image shown in Fig. 5 (a1). $f_{lung}$ is the subimage of the lung region. As indicated in Table 1, the EPTV method leads to the smallest error of the whole image among the three algorithms. It also results in the smallest error inside the lung region.

To quantify the image quality of reconstructed images in terms of image noise, the signal to noise ratio (SNR) of a region of interest (ROI) is shown in Table 1. The calculation is as follows:

$$SNR = 10 \times \log_{10} \left[ \frac{E^2\{P\}}{E(\{P - E(P)\}^2)} \right], \quad (16)$$

where $P$ is the set of pixels inside the ROI and $E(\cdot)$ stands for a spatial average over the ROI. It can be observed from Table 2 that when the imaging dose is decreased by 100 times, the SNR of images reconstructed by FBP is lowered from 40.6 to 20.4, indicating the sensitivity of the FBP method to image noise and its inability to handle the low dose situation. In contrast, the SNRs of images reconstructed by TV and EPTV methods are only decreased by a little compared to images reconstructed by FBP with high dose. With these results, we can conclude that our EPTV method not only outperforms the TV method on preserving the low contrast edges, but also inherits the advantage of the TV method on handling the low dose situation and effectively suppressing image noise.

The experimental results of the Catphan phantom with 20 iterations are shown in Fig. 6. The low imaging dose in this case is 8mAs, about 2% of that in the high dose scan, by decreasing the number of projections from 984 to 400 and reducing the mAs/projection from 0.4 to 0.02 mAs/projection. The sixth line pair at which the arrows point clearly demonstrate the different performances of the three reconstruction algorithms in terms of resolution. For the FBP reconstruction at low dose, the sixth line pair can still be clearly resolved as in Fig. 6(f). Compared with the reconstructed image of high dose, the low dose image reconstructed by FBP deteriorates mainly due to image noise, but still maintains good spatial resolution (6.5 lp/cm). On the other hand, although the TV method performs well in suppressing the image noise, the spatial resolution is degraded to 4 lp/cm. The sixth pair of lines is blurred and cannot be visually resolved, see Fig. 6(g). Finally, for the image reconstructed by the EPTV method, it contains less noise and maintains high spatial resolution (6.5 lp/cm), where the lines in the sixth pair can be clearly identified from each other in Fig. 6(h). The results of this test confirm the conclusion that, not only does our EPTV method effectively reduce image noise in low dose CT reconstruction, but also performs better than TV in maintaining acceptable spatial resolution. However, the EPTV algorithm prefer to perform less smoothing on edges, which makes the edges not as smooth as those reconstructed by the TV algorithm. Therefore, the EPTV algorithm maintains higher spatial resolution at the cost of the smoothness of the edges.

### 3.3 Computation efficiency

To speed up our iterative algorithm, we have implemented and tested it on an NVIDIA Tesla C1060 card. For comparison purpose, the algorithm in C language was also implemented on a 2.27 GHz Intel Xeon processor. The computation times per iteration of the EPTV algorithm on CPU and GPU for an axial CT slice are shown in Table. 3, which is case dependent, as it varies with the number of projections ($NP$). The total computation time scales linearly with the number of iterations ($NI$). When comparing $T_{CPU}$ and $T_{GPU}$, it is found that the algorithm is sped up by factors of about 19.6~23.1 by the GPU implementation. We have also timed some key operations in an iteration step of the EPTV algorithm, such as the CGLS solver, GD solver, and the post processing to ensure the image positivity. Within the CGLS solver, different components such as the forward calculation of computing $Pf$ (FC) and the backward calculation of computing $P^T y$ (BC) are also timed.

The detailed results are shown in Table 4. It is found that the most time-consuming step is the CGLS used to solve the (P1) problem, which is linearly related to the number of projections. Since the GD step to solve the (P2) problem and the postprocessing part operate on the reconstructed image directly, their computation time is independent of the number of projections. Although currently the reconstruction speed of the EPTV is still lower than that of the FBP algorithm, it is expected that with other techniques, such as multi-GPU implementation and multi-resolution reconstruction, the computation time will be further shortened.

## 4. Discussion and Conclusions

Two possible ways to reduce the imaging dose of CT scans are to reduce the number of projections and lower the mAs per projection. However, the common FBP reconstruction algorithms result in streaking artifacts and obvious image noises in these circumstances, which can be clearly observed in the experimental results. The iterative algorithm with TV norm as a regularization term has presented its tremendous power in CT reconstruction with undersampled projections acquired at low mAs level. However, the low contrast structures tend to be smoothed out by the TV regularization, posing a great challenge to the TV method. In this work, we propose an edge-preserving TV norm to solve the problem by giving adaptive penalty weights to pixels to suppress smoothing near both the strong edges and low contrast edges, relative to the non-edge pixels. By this adaptive penalization mechanism, our algorithm could automatically pick out the edges gradually during the implementation and give them a small weight and thus preferentially performs smoothing on those non-edge pixels to reduce image noise while keeping those edges unaffected, especially the low contrast edges. Our EPTV approach has been validated on a digital NCAT phantom, a physical chest phantom and also a Catphan phantom. The experimental results have clearly demonstrated the advantage of our approach over the conventional filtered backprojection algorithm and the TV-based reconstruction algorithm under low dose context on effectively suppressing image noise and streaking artifact and also maintaining acceptable signal-to-noise ratio and spatial resolution, which indicates a promising prospect of our low dose CT reconstruction algorithm in clinical applications for a considerable reduction of radiation dose exposed to patients.

To further understand the advantages of our EPTV algorithm over the TV algorithm, it is better to consider how the minimization is performed for the problem (P2) in Eq. (8). This model essentially tend to enhance image quality of an input image $v$ by utilizing a regularization function $J[f]$, which is of a total variation form in the conventional TV method and EPTV term in this paper. It is straightforward to perform the minimization by using a gradient descent approach, or mathematically equivalently to solve a partial differential equation (PDE), $\frac{\partial f}{\partial t} = -\lambda(f-v) - \frac{\partial_J[f]}{\partial f}$, where $t$ is a variable to parameterize the evolution of the solution. In TV case, the variation term on the right hand side is $\frac{\partial_J[f]}{\partial_f} = -\nabla \cdot \left(\frac{1}{|\nabla f|}\nabla f\right)$. This equation is then essentially a diffusion equation with a diffusion constant $\frac{1}{|\nabla f|}$. During the evolution according to the PDE, the high frequency signals, such as noise, can be removed due to the diffusive process. Meanwhile, sharp edges are preserved, as at those sharp edges $|\nabla f| \rightarrow \infty$ and hence the diffusion constant approaches zero. It is essentially for

this reason that TV method can be very effective in terms of removing noise while preserving sharp edges. In a continuous model, apparently sharp edges will be perfectly preserved since the diffusion constant becomes exactly zero at edges. However, in a discrete case as in numerical calculation, the diffusion constant will never be exactly zero due to the use of the finite difference. This means the diffusion process during the optimization will more or less take place at edges no matter how sharp the edge is. This will lead to blurring of the edge to a certain extent, though very small for sharp edges. This is even worse for low contrast edges, as the diffusion is significant. As for our EPTV algorithm, we can perform

the same analysis and obtain $\frac{\partial_J [f]}{\partial_f} = -\nabla \cdot \left(\frac{1}{|\nabla f|} \nabla f\right)$. The diffusion constant now becomes $\frac{w}{|\nabla f|}$, where it is always true that $w < 1$ according to its definition. Therefore, the diffusion at edges is always smaller in EPTV than in TV, leading to better edge preserving. Moreover, during the optimization, the parameter $w$ is adaptively adjusted so that there is always a distinction between edge and non-edge, making the algorithm to preferentially perform smoothing on those non-edge pixels. Besides, note that TV term is a special case of the EPTV term with all $w$ factors equal to unity and it holds that $w < 1$ in the EPTV case, therefore the EPTV term is smaller than the corresponding TV term, which makes the EPTV term a sparser representation than the original TV term.

GPU is employed in our experiments to improve the speed of this iterative algorithm, which is always a big challenge for the clinical applications of this type of algorithms. It is found that the GPU implementation is able to speed up the iterative reconstruction process by factors of 19.6~23.2 and make it possible to reconstruct an image in seconds instead of minutes. Though currently the reconstruction speed still cannot compete with the speed of the conventional FBP, it is hoped that the efficiency can be improved by using multi-GPU and reconstructing multiple CT slices simultaneously in the near future. We would also work on optimizing our algorithm to improve the speed further.

The choice of the parameter $\mu$ is of central importance to the success of this EPTV reconstruction method. In this preliminary study, we manually choose this parameter to adjust the relative weight between the data fidelity term and the regularization term to get the best reconstructed CT image quality. It is found that the optimal value of this parameter is case dependent. In future, we would study on the parameter setting try to find an automatic or semi-automatic way to guarantee a good choice of the parameter.

Lowering mAs level to reduce the patient radiation dose can easily be to done on existing commercial CT scanners. While, the other method proposed in our paper, namely, reducing the number of projections to reduce the dose is not straightforward on currently available commercial CT scanners due to the use of continuous x-ray generation mode. However, technically it is possible to modify the scanners to operate in high-frequency pulsed mode(Kang *et al.*, 2010; Myagkov *et al.*, 2009; Yue *et al.*, 2002), if there is a clinical need (such as the one suggested in this paper).

## Acknowledgments

# References

Bazaraa, M.; Sherali, H.; Shetty, C. Nonlinear programming: theory and algorithms. New York: Wiley; 2006.

Bell, N.; Garland, M. NVIDIA Technical Report NVR-2008-004. NVIDIA Corporation; 2008. Efficient sparse matrix-vector multiplication on CUDA.

Brenner DJ, Elliston CD, Hall EJ, et al. Estimated risks of radiation-induced fatal cancer from pediatric CT. AJR Am J Roentgenol. 2001; 176:289–96. [PubMed: 11159059]

Brody A, Frush D, Huda W, et al. Radiation risk to children from computed tomography. Pediatrics. 2007; 120:677–82. [PubMed: 17766543]

Chen G, Tang J, Leng S. Prior image constrained compressed sensing (PICCS): a method to accurately reconstruct dynamic CT images from highly undersampled projection data sets. Med Phy. 2008; 35:660–3.

Chodick G, Ronckers C, Shalev V, et al. Excess lifetime cancer mortality risk attributable to radiation exposure from computed tomography examinations in children. Isr Med Assoc J. 2007; 9:584–7. [PubMed: 17877063]

Combettes P, Wajs V. Signal recovery by proximal forward-backward splitting. Multiscale Model Simul. 2005; 4:1168–200.

Deans, SR. The Radon transform and some of its applications. New York: Wiley; 1983.

de Gonzalez AB, Mahesh M, Kim KP, et al. Projected Cancer Risks From Computed Tomographic Scans Performed in the United States in 2007. Arch Int Med. 2009; 169:2071–7. [PubMed: 20008689]

Donoho D. Compressed sensing. IEEE Trans Inform Theory. 2006; 52:1289–306.

Gu X, Choi D, Men C, et al. GPU-based ultra-fast dose calculation using a finite size pencil beam model. Phys Med Biol. 2009; 54:6287–97. [PubMed: 19794244]

Gu XJ, Pan H, Liang Y, et al. Implementation and evaluation of various demons deformable image registration algorithms on a GPU. Phys Med Biol. 2010; 55:207–19. [PubMed: 20009197]

Guo, W.; Yin, W. Rice CAAM Report TR10-02. 2010. EdgeCS: Edge Guided Compressive Sensing Reconstruction.

Hale E, Yin W, Zhang Y. Fixed-Point Continuation for l1-Minimization: Methodology and Convergence. SIAM J Optim. 2008; 19:1107–30.

Hall EJ, Brenner DJ. Cancer risks from diagnostic radiology. Br J Radiol. 2008; 81:362–78. [PubMed: 18440940]

Hestenes M, Stiefel E. Methods of conjugate gradients for solving linear systems. J Res Natl Bur Stand. 1952; 49:409–36.

Hissoiny S, Ozell B, Despres P. Fast convolution-superposition dose calculation on graphics hardware. Med Phys. 2009; 36:1998–2005. [PubMed: 19610288]

Jerri A. The Shannon sampling theorem-Its various extensions and applications: A tutorial review. Proc IEEE. 1977; 65:1565–96.

Jia X, Gu X, Sempau J, et al. Development of a GPU-based Monte Carlo dose calculation code for coupled electron-photon transport. Phys Med Biol. 2010a; 55:3077–86. [PubMed: 20463376]

Jia X, Lou Y, Li R, et al. GPU-based fast cone beam CT reconstruction from undersampled and noisy projection data via total variation. Med Phy. 2010b; 37:1757–60.

Kang M, Park C, Lee C, et al. Dual-Energy CT: Clinical Applications in Various Pulmonary Diseases. Radiographics. 2010; 30:685–99. [PubMed: 20462988]

Kanopoulos N, Vasanthavada N, Baker R. Design of an image edge detection filter using the sobel operator. IEEE J solid-state circuits. 1988; 23:358–67.

Li M, Yang H, Koizumi K, et al. Fast cone-beam CT reconstruction using CUDA architecture. Med Imaging Technol. 2007; 25:243–50.

Men C, Gu X, Choi D, et al. GPU-based ultrafast IMRT plan optimization. Phys Med Biol. 2009; 54:6565–73. [PubMed: 19826201]

Men C, Jia X, Jiang S. GPU-based ultra-fast direct aperture optimization for online adaptive radiation therapy. Phys Med Biol. 2010; 55:4309–19. [PubMed: 20647601]

Myagkov B, Shikanov E, Shikanov A. Development and investigation of a pulsed x-ray generator. At Energy. 2009; 106:143–8.

NVIDIA C. CUBLAS Library. Santa Clara, CA: NVIDIA Corporation; 2008.

Rudin L, Osher S, Fatemi E. Nonlinear total variation based noise removal algorithms. Physica D. 1992; 60:259–68.

Samant S, Xia J, Muyan-Ozcelik P, et al. High performance computing for deformable image registration: Towards a new paradigm in adaptive radiotherapy. Med Phys. 2008; 35:3546–53. [PubMed: 18777915]

Segars, W. Development of a new dynamic NURBS-based cardiac-torso (NCAT) phantom. Chapel Hill, NC: University of North Carolina Press; 2002.

Sharp G, Kandasamy N, Singh H, et al. GPU-based streaming architectures for fast cone-beam CT image reconstruction and demons deformable registration. Phys Med Biol. 2007; 52:5771–84. [PubMed: 17881799]

Siddon R. Fast calculation of the exact radiological path of a three-dimensional CT array. Med Phys. 1985; 12:252–5. [PubMed: 4000088]

Sidky E, Kao C, Pan X. Accurate image reconstruction from few-views and limited-angle data in divergent-beam CT. J X-Ray Sci Technol. 2006; 14:119–39.

Sidky E, Pan X. Image reconstruction in circular cone-beam computed tomography by constrained, total-variation minimization. Phys Med Biol. 2008; 53:4777–807. [PubMed: 18701771]

Smith-Bindman R, Lipson J, Marcus R, et al. Radiation dose associated with common computed tomography examinations and the associated lifetime attributable risk of cancer. Arch Intern Med. 2009; 169:2078–86. [PubMed: 20008690]

Song J, Liu Q, Johnson G, et al. Sparseness prior based iterative image reconstruction for retrospectively gated cardiac micro-CT. Med Phys. 2007; 34:4476–83. [PubMed: 18072512]

Xu F, Mueller K. Accelerating popular tomographic reconstruction algorithms on commodity PC graphics hardware. IEEE Trans Nucl Sci. 2005; 52:654–63.

Xu F, Mueller K. Real-time 3D computed tomographic reconstruction using commodity graphics hardware. Phys Med Biol. 2007; 52:3405–19. [PubMed: 17664551]

Xu, W.; Mueller, K. Learning effective parameter settings for iterative CT reconstruction algorithms. Fully 3D Reconstruction in Radiology and Nuclear Medicine Conference; 2009a. p. 251-4.

Xu, W.; Mueller, K. A performance-driven study of regularization methods for gpu-accelerated iterative CT. 2nd High Performance Image Reconstruction Workshop; Beijing. 2009b. p. 20-3.

Yan G, Tian J, Zhu S, et al. Fast cone-beam CT image reconstruction using GPU hardware. J X-Ray Sci Technol. 2008; 16:225–34.

Yue G, Qiu Q, Gao B, et al. Generation of continuous and pulsed diagnostic imaging x-ray radiation using a carbon-nanotube-based field-emission cathode. Appl Phys Lett. 2002; 81:355–7.

# Appendix

## Appendix I. CGLS algorithm

The conjugate gradient least square (CGLS) method used to solve the minimization problem $\|Pf-y\|_2^2$, which is a substitution of the subproblem (P1) in Eq. (8), is implemented as follows.

CGLS Algorithm:

Initialinize: $m = 0$, $u^{(0)} = f^{(k)}$, $r^{(0)} = y - Pu^{(0)}$, $s^{(0)} = P^T r^{(0)}$. Do the Steps 1–5 for M times.

**1**    $a^{(m)} = \|P^T r^{(m)}\|_2^2 / \|Ps^{(m)}\|_2^2$

2     $u^{(m+1)} = u^{(m)} + a^{(m)} s^{(m)}$

3     $r^{(m+1)} = r^{(m)} - a^{(m)} P s^{(m)}$

4     $b^{(m)} = \left\| P^T r^{(m+1)} \right\|_2^2 / \left\| P^T r^{(m)} \right\|_2^2$

5     $s^{(m+1)} = P^T r^{(m+1)} + b^{(m)} s^{(m)}$

6     $v^{(k+1)} = u^{(M)}$

## Appendix II. Derivation of the variation of EPTV and its finite difference scheme

To derive the functional variation of EPTV, $\frac{\delta}{\delta f} \| f \|_{EPTV}$, let us consider the integrations by parts of the EPTV term:

$$
\begin{aligned}
\| f \|_{EPTV} &= \int \omega(f(x)) \| \nabla f(x) \|_1 \, dx \\
&= \int \nabla f(x) \cdot \frac{\omega(f(x)) \nabla f(x)}{\| \nabla f(x) \|_1} \, dx \\
&= f \frac{\omega(f(x)) \nabla f(x)}{\| \nabla f(x) \|_1} \Big|_{\partial \Omega} - \int f \cdot \nabla \frac{\omega(f(x)) \nabla f(x)}{\| \nabla f(x) \|_1} \, dx
\end{aligned} \tag{A1}
$$

where $\Omega$ is the image domain and $\partial \Omega$ is its boundary. Assume zero boundary condition, the first term vanishes. Taking variation of the second term with respect to $f$ leads to Eq. (10). Numerically, we approximate the functional variation of EPTV by a symmetric finite difference scheme:

$$
\frac{\delta}{\delta f_{i,j}} \| f \|_{EPTV} \approx - \left[ \frac{\omega_{i+1/2,j}(f_{i+1,j} - f_{i,j})}{\| f_{i+1/2,i} \|_{TV}} - \frac{\omega_{i-1/2,j}(f_{i,j} - f_{i-1,j})}{\| f_{i-1/2,j} \|_{TV}} \right] - \left[ \frac{\omega_{i,j+1/2}(f_{i,j+1} - f_{i,j})}{\| f_{i,j+1/2} \|_{TV}} - \frac{\omega_{i,j-1/2}(f_{i,j} - f_{i,j-1})}{\| f_{i,j-1/2} \|_{TV}} \right] \tag{A2}
$$

where,

$$
\begin{aligned}
\| f_{i+1/2,j} \|_{TV} &= \left[ (f_{i+1,j} - f_{i,j})^2 + \tfrac{1}{2}(f_{i+1/2,j+1} - f_{i+1/2,j})^2 + \tfrac{1}{2}(f_{i+1/2,j} - f_{i+1/2,j-1})^2 \right]^{1/2}, \\
f_{i+1/2,j} &= \tfrac{1}{2}(f_{i+1,j} - f_{i,j}), \\
\omega_{i+1/2,j} &= \exp \left( - \frac{\| f_{i+1/2,j} \|_{TV}^2}{\sigma^2} \right).
\end{aligned} \tag{A3}
$$

Such an approximation scheme, though cumbersome, ensures the stability of the algorithm.

## Appendix III. Derivation of Eq. (11)

Let $f(.): \mathbf{R}^2 \to \mathbf{R}$ and $y(.): \mathbf{R} \to \mathbf{R}$ be two smooth enough functions in the CT slice domain and in the x-ray projection domain, respectively. The operator $P^{\theta^T}$, being the adjoint operator of the x-ray projection operator $P^\theta$, should satisfy the condition

$$
\langle f, P^{\theta^T} y \rangle = \langle P^\theta f, y \rangle. \tag{A4}
$$

where $\langle .,. \rangle$ denotes the inner product. This condition can be explicitly expressed as

$$\int \mathrm{d}u \mathrm{d}v \, f(u,v)[P^{\theta^T} y](u,v) = \int \mathrm{d}\beta \, [P^\theta f](\beta) y(\beta). \quad \text{(A5)}$$

Now take the functional variation with respect to *f(x)* on both sides of equation (A5) and interchange the order of integral and variation on the right hand side. This yields

$$[P^{\theta^T} y](u,v) = \frac{\delta}{\delta f} \int \mathrm{d}\beta \, [P^\theta f](\beta) y(\beta) = \int \mathrm{d}\beta \, y(\beta) \frac{\delta}{\delta f} [P^\theta f](\beta). \quad \text{(A6)}$$

With help of a delta function, the forward projection can be written as

$$[P^\theta f](\beta) = \int \mathrm{d}l \, f(u,v) \, \delta(u - u_s - l\cos(\beta+\theta)) \delta(v - v_s - l\sin(\beta+\theta)). \quad \text{(A7)}$$

where, $(u_S, v_S)$ is the coordinate of the x-ray source, *l* is the distance between the source and the pixel $(u, v)$ in the CT slice domain. Now substituting (A7) into (A6), we obtain

$$\begin{aligned}
[P^{\theta^T} y](u,v) &= \int \mathrm{d}l \mathrm{d}\beta \, y(\beta) \, \delta(u - u_s - l\cos(\beta+\theta)) \delta(v - v_s - l\sin(\beta+\theta)) \\
&= \int \mathrm{d}l \mathrm{d}\beta \, y(\beta) \, \delta(\beta-\beta^*) \delta(l-l^*) \frac{1}{\left|\frac{\partial(h_1,h_2)}{\partial(\beta,l)}\right|_{\beta^*,l^*}} = \frac{y^\theta(\beta^*)}{l^*}.
\end{aligned} \quad \text{(A8)}$$

$\beta^*$ is the angular coordinate of the rayline connecting the x-ray source and the pixel $(u^*, v^*)$ at the projection angle $\theta$. $y^\theta(\beta^*)$ is the corresponding detector reading. Changing from continuous coordinate to discrete coordinate, the ratio of pixel size $\Delta u \Delta v$ on CT slice to the angular spacing size $\Delta\beta$ of the detector unit is added to the Eq. (A8). Additionally, a summation over projection angles $\theta$ is performed to account for all the x-ray projection images. So we arrive at the Eq. (11). We have tested the accuracy of such defined operator $P^T$ in terms of satisfying condition expressed in Eq. (A4). Numerical experiments indicate that this condition is satisfied with numerical error less than 1%, which is found accurate enough for our iterative CT reconstruction purpose.
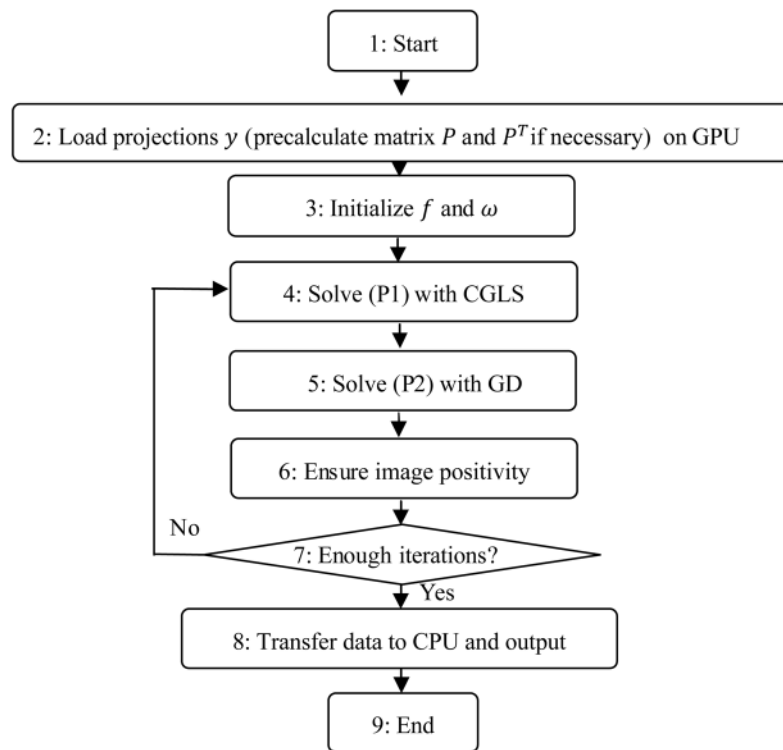
**Figure 1.**
The flow chart of our GPU-based EPTV reconstruction. Blocks 4~6 correspond to the Steps 1~3 of the EPTV Algorithm.
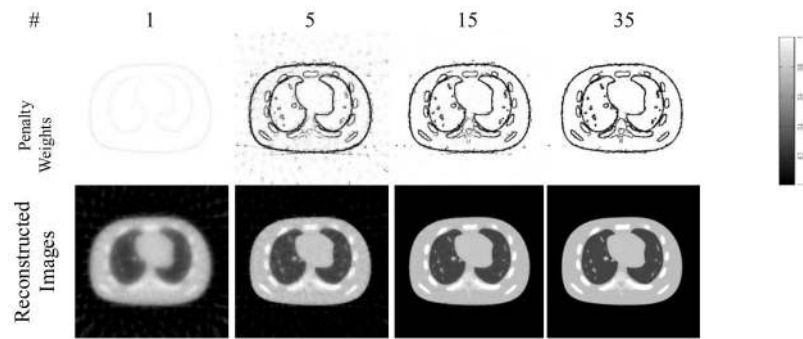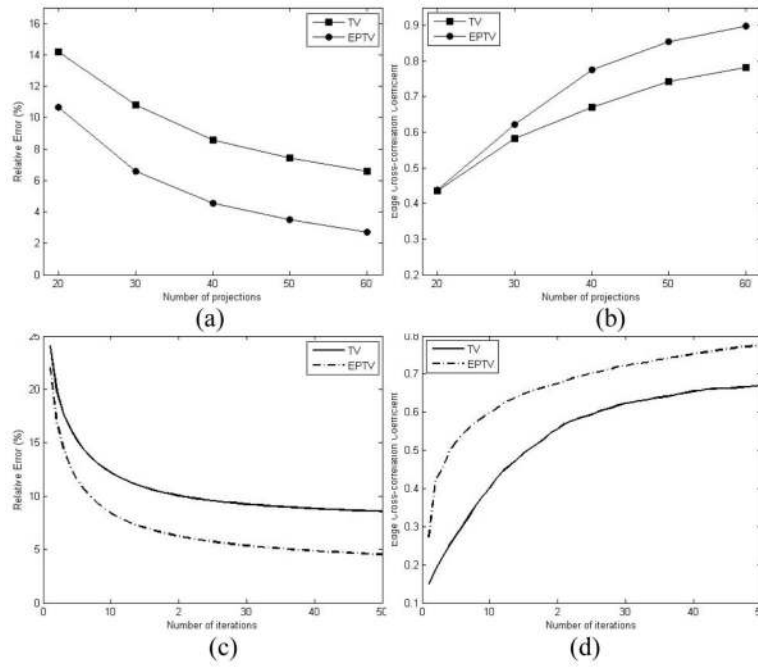
**Figure 2.**
The evolution of the penalty weights, reconstructed image and energy function at different numbers of iteration (top row) during the reconstruction using our EPTV algorithm with 40 projections.
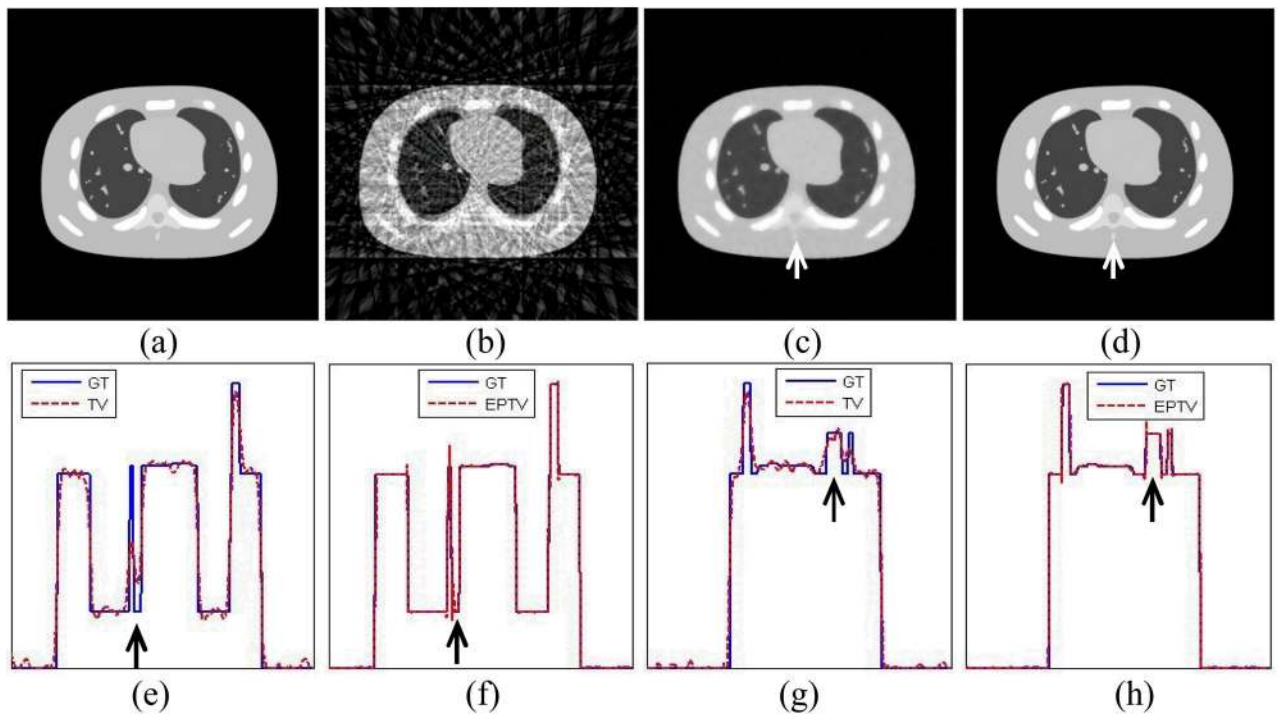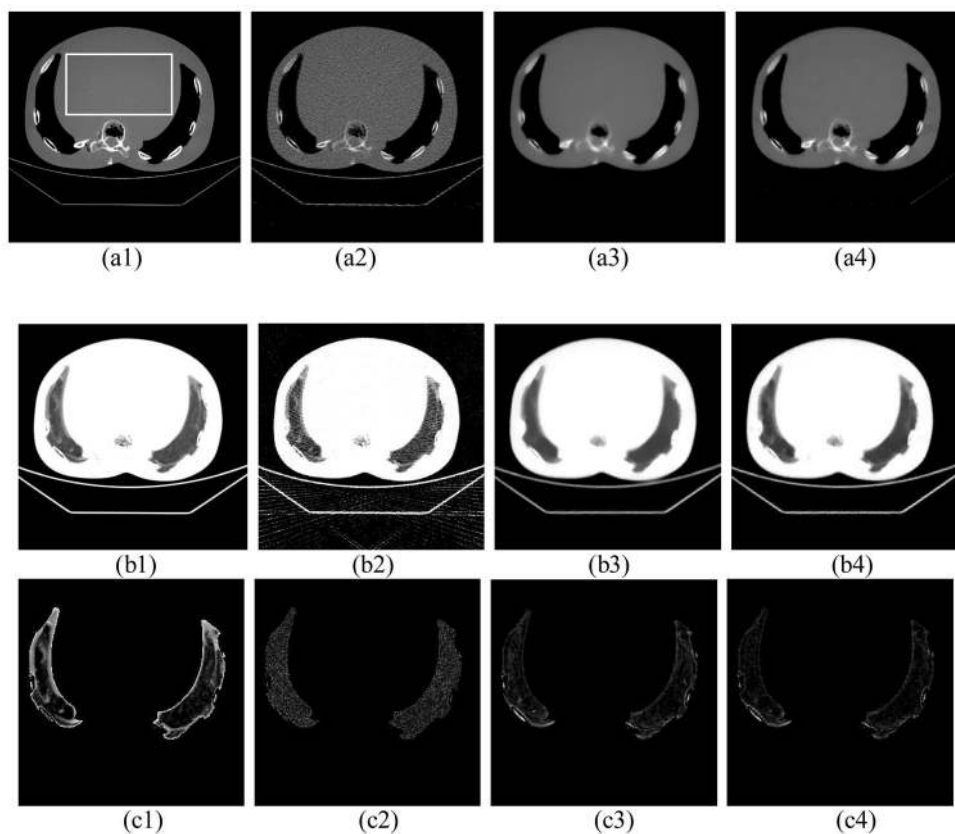
**Figure 3.**
The reconstruction results of NCAT phantom using 40 projections with 50 iterations. (a) is the ground truth image. (b) ~ (d) show the images reconstructed by FBP, TV, EPTV, respectively. (e)(f) depict the horizontal intensity profiles through the center of reconstructed images of TV and EPTV, while (g)(h) are the corresponding vertical intensity profiles. The profiles of ground truth (GT) are also plotted in solid lines for comparison. The arrows indicate the area where EPTV is clearly superior to TV.
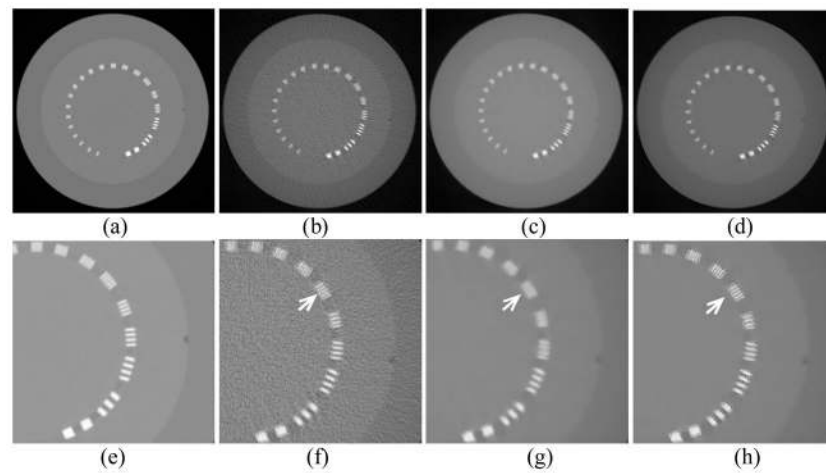
**Figure 4.**
(a)(b) show the relative error and the edge cross-correlation coefficient as functions of the number of x-ray projections at the 50th iteration; (c)(d) show the evolution curve of the relative error and the edge cross correlation during the first 50 iterations using 40 projections.

**Figure 5.**
Reconstruction results of a physical chest phantom with 30 iterations. For the first two rows, (1) shows the image reconstructed by FBP at 400mAs (984 projections × 0.407mAs/projection). (2), (3), and (4) show the images reconstructed by FBP, TV, EPTV, respectively, at 4mAs (200 projections × 0.0203mAs/projection). The CT images are shown with display window [−400 1000] and [−900 10] in the first two rows to show the bones and the details in the dark lung region, respectively. The rectangle indicates an ROI used for SNR calculation. In the third row, (c1) shows the enlarged lung region of (1), (c2)~(c4) show the difference of the lung region between (1) and (2),(3),(4), respectively.

**Figure 6.**
Reconstruction results of the Catphan phantom with 20 iterations. (a) shows the image reconstructed by FBP at 400mAs (984 projections × 0.407mAs/projection). (b), (c), and (d) show the images reconstructed by FBP, TV, EPTV, respectively, at 8mAs (400 projections × 0.0203mAs/projection). (e)–(h) zoom in the corresponding images on the first row. Arrows indicate the area where EPTV is clearly superior to TV.

**Table 1**

Relative errors of the whole image and subimage of lung region

|  | FBP | TV | EPTV |
|---|---|---|---|
| $e_{whole}$ | 0.39 | 0.16 | 0.15 |
| $e_{lung}$ | 0.17 | 0.13 | 0.10 |

**Table 2**

SNR of the ROI in the reconstructed images of the chest phantom.

|  | **FBP** | **FBP** | **TV** | **EPTV** |
|---|---|---|---|---|
| mAs | 400 | 4 | 4 | 4 |
| SNR | 40.6 | 20.4 | 39.9 | 40.0 |

**Table 3**

The computation time per iteration of the EPTV algorithm on CPU and GPU.

|         | NP  | NI | $T_{CPU}$ (s) | $T_{GPU}$ (s) | $T_{CPU}/T_{GPU}$ |
|---------|-----|----|-----|-----|------|
| NCAT    | 40  | 50 | 2.54 | 0.11 | 23.2 |
| Chest   | 200 | 30 | 4.25 | 0.20 | 21.7 |
| Catphan | 400 | 20 | 7.57 | 0.38 | 19.6 |

**Table 4**

The computation time per iteration of each component in the EPTV algorithm. The numbers are in a unit of ms.

|         | NP  | CGLS   | GD   | Post-processing |
|---------|-----|--------|------|-----------------|
| NCAT    | 40  | 13.21  | 4.87 | 0.014           |
| Chest   | 200 | 67.34  | 4.90 | 0.014           |
| Catphan | 400 | 122.83 | 4.90 | 0.014           |