Compressed Sensing and Adaptive Graph Total Variation for Tomographic Reconstructions

Faisal Mahmood*, Nauman Shahid*, Ulf Skoglund† and Pierre Vandergheynst†

Abstract—Compressed Sensing (CS) and Total Variation (TV)-based iterative image reconstruction algorithms have received increased attention recently. This is due to the ability of such methods to reconstruct from limited and noisy data. Local TV methods fail to preserve texture details and fine structures, which are tedious for the method to distinguish from noise. In many cases local methods also create additional artifacts due to over smoothing. Non-Local Total Variation (NLTV) has been increasingly used for medical imaging applications. However, it is not updated in every iteration of the algorithm, has a high computational complexity and depends on the scale of pairwise parameters. In this work we propose using Adaptive Graph-based TV in combination with CS (ACSGT). Similar to NLTV our proposed method goes beyond spatial similarity between different regions of an image being reconstructed by establishing a connection between similar regions in the image regardless of spatial distance. However, it is computationally much more efficient and scalable when compared to NLTV due to the use of approximate nearest neighbor search algorithm. Moreover, our method is adaptive, *i.e*, it involves updating the graph prior every iteration making the connection between similar regions stronger. Since TV is a special case of graph TV the proposed method can be seen as a generalization of CS and TV methods. We test our proposed algorithm by reconstructing a variety of different phantoms from limited and corrupted data and observe that we achieve a better result with ACSGT in every case.

Index Terms—Tomography, Total Variation, Graphs, Iterative Image Reconstruction, Non-local Total Variation, Non-local image processing, Compressive Sensing, Non-local denoising, Non-local Regularization

I. INTRODUCTION

TOMOGRAPHIC modalities such as electron tomography (ET) and computed tomography (CT) suffer from low-dose and missing data constraints which lead to noisy and erroneous reconstructions. In the case of ET, biological samples are rotated in a transmission electron microscope (TEM) and can not be exposed to high levels of electron dose because it leads to the degradation of the sample [1, 2].

Where as, in the case of CT low-dose has been a clinical objective to prevent the patient from over exposure to ionizing radiation [3–5]. One way to reduce the dose is by reducing the number of projections or views collected during tomographic imaging. However, this leads to missing data, which results in reconstructions effected by noise.

A variety of image reconstruction algorithms have been employed over the years which can be divided into two categories 1) *Analytical* [6–8] 2) *Iterative* [9–14]. Analytical algorithms, which are based on the Fourier slice theorem such as filtered back-projection (FBP) are computationally more efficient and have been extensively used in a clinical setting [10][15]. Iterative image reconstruction (IIR) methods on the other hand handle noise and missing data relatively better than analytical methods and also have the ability to incorporate *a priori* information. However, such methods are computationally inefficient and often require training parameters, which are impractical in a clinical setting. Initial IIR methods were algebraic in nature such as ART [16], SIRT [17] and their variants [18–21]. Statistical approaches to IIRs are often based on the *Ordered Subset* (OS) technique [9, 22–24].

Another era of algorithms for tomographic reconstructions uses Compressed sensing (CS). However, CS alone is not sufficient to cater for the sparsity of the Gradient Magnitude Image (GMI), therefore, it is used along with Total Variation (TV) in CT and ET [25–30]. We refer to the joint CS and TV setup as CSTV in the sequel. The authors of [31] use CSTV for ET and show that the reconstruction error is better than other iterative algorithms when reconstructing from limited data. A comparison of statistical iterative reconstruction methods with CSTV has been provided in [29]. An analysis of the CSTV method for reconstructing needle shaped biological specimens is presented in [32]. CSTV has also been used in photo-acoustic tomography as presented by the authors of [33]. A concise review of the iterative reconstruction algorithms, including TV regularization methods is presented in [15].

A more advanced type of TV regularizer, known as the non-local TV (NLTV) [34] has been shown to be much more efficient for inverse problems, such as denoising, inpainting and super-resolution [35, 36]. In contrast to the simple TV which takes into account the similarity of a region with only its neighboring regions, NLTV is designed to overcome this limitation by associating a similarity measure of every region of an image / sample with all the regions. This will be explained in more detail in the upcoming sections of this paper. As an application of NLTV, the authors of [37] presented a reconstruction method for Magnetic Resonance Imaging (MRI) via CS. NLTV has been further explored in a spectral

CT setting in [38, 39].

A primary short-coming of NLTV based methods used in CT, ET and MRI settings is that the similarity matrix constructed in the beginning from the initial estimate of the sample is not updated throughout the algorithm, for example [37]. The authors of [38] construct a similarity matrix from the solution of a TV based minimization method and then keep it fixed throughout the algorithm. A primary reason for keeping it fixed is that the NLTV based method suffers from a high cost of associating a similarity measure between every pair of regions in an image / sample. For an image of the size $n \times n$, NLTV costs n^4 and is computationally cumbersome for high resolution applications. Furthermore, it does not make sense for every region to be connected to all other regions. Therefore, NLTV requires a threshold parameter which, based on the euclidean distance can decide if the similarity is strong enough to be non-zero. However, there is no appropriate method to fix this parameter and it depends on the scale of pairwise distances. Although, the results obtained by the above methods are state-of-the-art, we believe that the final reconstruction would be more faithful to the original one if one updates the similarity matrix from the simultaneously reconstructed sample regularly throughout the algorithm.

Contributions: In this paper we propose Compressed Sensing and Adaptive Graph Total Variation (ACSGT) as a method for simultaneous reconstruction and denoising of tomographic data. Our method is a more sophisticated, faster and scalable form of NLTV and enjoys a computational complexity which is much less as compared to NLTV. We promote the use of $\mathcal{K}$-nearest neighbor graphs in Graph Total Variation (GTV), where $\mathcal{K}$ is fixed and unlike NLTV, does not depend on the scale of the pairwise distances. The graph is constructed by using an approximate nearest neighbor search algorithm (FLANN) [40]. Furthermore, our method requires updating the GTV prior in every iteration by constructing a new graph from simultaneously reconstructed sample. Due to a significant computational cost reduction in ACSGT as compared to NLTV, we can afford to update the graph in every iteration. To the best of our knowledge this is the first work which uses adaptive GTV and CS based joint method for the application of CT.

II. INTRODUCTION TO GRAPHS

A graph is represented as a tuple $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \mathcal{W}\}$ where $\mathcal{V}$ is a set of vertices, $\mathcal{E}$ a set of edges, and $\mathcal{W} : \mathcal{V} \times \mathcal{V} \to \mathbb{R}_+$ a weight function. We assume that the vertices are indexed from $1, \ldots, |\mathcal{V}|$. The weight matrix W is assumed to be non-negative, symmetric, and with a zero diagonal. Each entry of the weight matrix $W \in \mathbb{R}_+^{|\mathcal{V}| \times |\mathcal{V}|}$ corresponds to the weight of the edge connecting the corresponding vertices: $W_{i,j} = \mathcal{W}(v_i, v_j)$ and if there is no edge between two vertices, the weight is set to 0. A node v_i connected to v_j is denoted by $i \leftrightarrow j$. For a vertex $v_i \in \mathcal{V}$, the degree $d(i)$ is defined as the sum of the weights of incident edges: $d(i) = \sum_{j \leftrightarrow i} W_{i,j}$. We define a graph signal as a function $s : \mathcal{V} \to \mathbb{R}$ which assigns a value to each vertex in the graph. It is convenient to consider a signal s as a vector of size $|\mathcal{V}|$ with the ith component representing the signal value at the ith vertex.

We define two types of signals on graphs, the one which resides on the vertices and the other one on edges. For a signal s residing on the vertices of graph $\mathcal{G}$, the gradient $\nabla_{\mathcal{G}} : \mathbb{R}^{|\mathcal{V}|} \to \mathbb{R}^{|\mathcal{E}|}$ is defined as

$$\nabla_{\mathcal{G}} s(i,j) = \sqrt{W(i,j)} \left(\frac{s(j)}{\sqrt{d(j)}} - \frac{s(i)}{\sqrt{d(i)}} \right),$$

where we consider only the pair $\{i, j\}$ when $i \leftrightarrow j$. For a signal f residing on the graph edges, the adjoint of the gradient $\nabla_{\mathcal{G}}^* : \mathbb{R}^{|\mathcal{E}|} \to \mathbb{R}^{|\mathcal{V}|}$, called divergence can be written as

$$\nabla_{\mathcal{G}}^* f(i) = \sum_{i \leftrightarrow j} \sqrt{W(i,j)} \left(\frac{1}{\sqrt{d(i)}} f(i,i) - \frac{1}{\sqrt{d(j)}} f(i,j) \right).$$

The Laplacian corresponds to the second order derivative and its definition arises from $Ls := \nabla_{\mathcal{G}}^* \nabla_{\mathcal{G}} s$. Let D be the diagonal degree matrix with diagonal entries $D_{ii} = d(i)$, then another interpretation of the Laplacian is the difference of the weight matrix W from the degree matrix D, thus $L = D - W$, which is referred to as combinatorial Laplacian.

A graph $\mathcal{G}$ can be constructed from a dataset in several ways. For example, for a signal $Y \in \mathbb{R}^{p \times n}$, the vertices v_i of the graph $\mathcal{G}$ can correspond to the data samples $y_i \in \mathbb{R}^p$. For the purpose of this work we are interested in a standard $\mathcal{K}$-nearest neighbors strategy for the graph construction. The first step of this strategy consists of searching the closest neighbors for all the samples using Euclidean distances. Thus, each y_i is connected to its $\mathcal{K}$ nearest neighbors y_j, resulting in $|\mathcal{E}| \approx \mathcal{K}n$ number of connections. The second step consists of computing the graph weight matrix W using one of the several commonly used strategies. The most commonly used scheme is based on the Gaussian kernel:

$$W_{ij} = \begin{cases} \exp \left(- \frac{\|(y_i - y_j)\|_2^2}{\sigma^2} \right) & \text{if } y_j \text{ is connected to } y_i \\ 0 & \text{otherwise.} \end{cases} \quad (1)$$

How can the notion of graphs be used in signal processing? A primary example is the denoising operation, where one can solve an inverse problem by regularizing with a *graph tikhonov operator*. For a noisy signal $y \in \mathbb{R}^n$ and a graph Laplacian L constructed between the entries of y, the graph tikhonov regularization on the optimization variable x (the clean signal) is defined as [41]:

$$x^\top L x = \frac{1}{2} \sum_i \sum_j W_{ij} (x_i - x_j)^2.$$

Clearly, the strongly connected samples in y_i, y_j have a higher W_{ij}, therefore, the resulting x_i, x_j resemble each other more. This smoothness prior can be used to denoise a signal y with an irregular structure by solving the optimization problem of the following form:

$$\min_x \|y - x\|_2^2 + \gamma x^\top L x$$

GTV is another graph based regularizer which has been frequently used in many applications. We discuss this regularizer in detail in the next section.

III. PROPOSED METHOD: COMPRESSED SENSING & ADAPTIVE GRAPH TOTAL VARIATION

Our proposed method involves simultaneous denoising and reconstruction of tomographic projections and constitutes the following important components:

1) Compressed sensing based sparse reconstruction.
2) Adaptive Graph total variation regularizer for improved denoised reconstruction.

A. Optimization Problem

We first present the optimization problem under consideration and then study each of the terms of the objective function in detail. Let $S \in \Re^{p \times q}$ be the sinogram corresponding to the projections of the sample $X \in \Re^{n \times n}$ being imaged, where p is the number of rays passing through X and q is the number of angular variations at which X has been imaged. Let $b \in \Re^{pq}$ be the vectorized measurements or projections ($b = vec(S)$), where $vec(\cdot)$ denotes the vectorization operation and $A \in \Re^{pq \times n^2}$ be the sparse projection operator. Then, the goal in a typical CT or ET based reconstruction method is to recover the vectorized sample $x = vec(X)$ from the projections b. A highly under-determined inverse problem needs to be solved for this type of reconstruction, which is even more tedious if the projections b are corrupted with noise and measurement errors. To circumvent the problem of noisy projections, we propose

$$\min_x \|Ax - b\|_2^2 + \lambda \|\Phi^*(x)\|_1 + \gamma \|x\|_{gtv}, \qquad (2)$$

where Φ is the wavelet operator and $\Phi^*(x)$, where $*$ represents the adjoint operation, denotes the wavelet transform of x and x_{gtv} denotes the total variation of x w.r.t graph $\mathcal{G}$. The first two terms of the objective function above comprise the compressed sensing based sparse reconstruction part of our method and the second term, to which we refer as the *graph total variation* (GTV) regularizer acts as an additional prior for denoising and smoothing the reconstructed sample. Now, we explain these two components in detail.

1) Compressed Sensing: Compressed Sensing (CS), introduced in the seminal papers of Donoho [42] and Candes et. al [43] guarantees the recovery of a data sample from highly undersampled measurements if the sample to be recovered can be sparsely represented in a basis. Let $x \in \mathbb{R}^{n^2}$ be the vectorized CT sample under consideration and $\Phi \in \mathbb{R}^{n^2 \times n^2}$ be the fourier basis, whose columns contain the fourier atoms. Our proposed model, as described earlier, uses wavelets, however for the ease of description we choose to explain compressed sensing with a Fourier matrix instead. For the case of Fourier matrix, $\Phi^*(x) = \Phi^\top x$ and one can sparsely represent the sample x in Φ as:

$$x = \sum_i \Phi_i c_i = \Phi c,$$

where $\|c\|_0 \ll n^2$, i.,e the number of non-zeros in c is much less than n. The sparse coefficients c are also referred to as the *transform codes*. Given this assumption, it is possible to recover the unknown x from highly undersampled measurements

or projections $b \in \mathbb{R}^{pq}$, where $pq < n^2$. The projections or the measurements b can be obtained by applying a projection or sampling operator $A \in \mathbb{R}^{pq \times n^2}$ to the sample x. Thus, the projections b can be given as:

$$b = Ax = A\Phi c$$

Given b, it is possible to recover x by solving the following sparse recovery problem

$$\min_x \|\Phi^T x\|_1 \quad \text{s.t: } Ax = b,$$

where Φ^T denotes the forward transform (Fourier transform if Φ is a Fourier matrix). In case the projections or measurements b are contaminated with noise, one can solve the following recovery problem:

$$\min_x \|Ax - b\|_2^2 + \lambda \|\Phi^T x\|_1,$$

where λ is the regularization parameter which provides a trade-off between the sparsity and noise tolerance.

For tomographic applications, the projection matrix A is a line integral computed by rotating the sample x at different angles and the projections b are typically corrupted by Poisson noise. Therefore, CS is not only used as a reconstruction algorithm but also provides robustness to noise. However, in case of high fraction of noise, CS fails to recover the sample efficiently, therefore, we propose to add another regularization term to our setup as explained below.

2) Graph Total Variation Regularization: The graph total variation (GTV), denoted as $\|x\|_{gtv}$, in eq. (2), like the standard TV has two types 1) anisotropic and 2) isotropic. The former involves the sum of the gradients of nodes (entries in x) w.r.t $\mathcal{G}$ and the later involves the sum of the l_2 norms of the gradients at each node of $\mathcal{G}$. Throughout this paper we use the former formulation as it has an intuitive interpretation, thus

$$\|x\|_{gtv} = \|\nabla_{\mathcal{G}}(x)\|_1 = \sum_i \|\nabla_{\mathcal{G}} x_i\|_1$$
$$= \sum_i \sum_{j \in \mathcal{N}_i} \sqrt{W_{ij}} \|x_i - x_j\|_1, \qquad (3)$$

where the second sum runs over all the neighbors of i, denoted by $\mathcal{N}_i$. The above expression clearly states that GTV involves the minimization of the sum of the gradients of the signals on the nodes of the graphs. In our case, we assume that the elements of the vector x lie on the nodes of the graph $\mathcal{G}$ which are connected with the edges whose weights are W_{ij}. Thus, the minimization of the GTV would ensure that x_i and x_j possess similar values if W_{ij} is high and dissimilar values if W_{ij} is small or zero. As compared to the standard TV, the structure of the sample x is taken into account for the reconstruction purpose. It is a well known fact that l_1 norm promotes sparsity, so the GTV can also be viewed as a regularization which promotes sparse graph gradients. This directly corresponds to enforcing a smoothness of the signal x w.r.t graph $\mathcal{G}$.

The proposed method with GTV can be seen as a generalization of the compressed sensing and total variation based method studied in [29]. While, the standard TV minimizes the gradients of the signal x w.r.t its spatial neighbors only, the

GTV does so in a region which is not restricted only to the neighbors of the elements in x. Thus, the standard TV can be viewed as a specific case of the GTV, where the graph $\mathcal{G}_{grid}$ is a grid graph. In a grid graph $\mathcal{G}_{grid}$ of a sample x, the pixels are only connected to its spatial neighbors (upper, lower, left and right) via unity weights.

Using eq. (3) in eq. (2), our proposed model can be written as:

$$\min_{c} \|Ax - b\|_2^2 + \lambda \|\Phi^*(x)\|_1 + \gamma \|\nabla_{\mathcal{G}} x\|_1 \quad (4)$$

Now using $x = \Phi(c)$ we get

$$\min_{c} \|A\Phi(c) - b\|_2^2 + \lambda \|c\|_1 + \gamma \|\nabla_{\mathcal{G}}(\Phi c)\|_1 \quad (5)$$

From the above equation it is obvious that our proposed model promotes simultaneous sparsity of the transform coefficients c in the Wavelet domain Φ and the sparsity of the gradients in the graph domain. Hence, the model can be viewed as promoting a doubly sparse structure in wavelet and graph domains respectively.

B. Graph Construction for Total Variation

An important step for our method is to construct a graph $\mathcal{G}$ for TV regularization. In contrast to standard TV, which can be directly used as a prior for regularization, GTV needs a graph $\mathcal{G}$ to start with. Ideally, $\mathcal{G}$ should be representative of the reconstructed sample x, however, this is unknown before the reconstruction. To cater this problem, we propose to construct $\mathcal{G}$ from an initial naive estimate of the sample x_{fbp} using filtered back projection (FBP) method.

We propose to construct a graph between the patches of x_{fbp} instead of pixels. As the sample to be reconstructed is an image, and the graph is being constructed from the noisy x_{fbp}, to obtain robustness to noise, it makes more sense to construct the graph from the patches rather than the pixels of x_{fbp}. In the first step $x_{fbp} \in \mathbb{R}^{n \times n}$ is divided into n^2 overlapping patches. Let s_i be the patch of size $l \times l$ centered at the ith pixel of x_{fbp} and assume that all patches are vectorized, $i.e$, $s_i \in \Re^{l^2}$ (Figure 1). In the second step the search for the closest neighbors for all vectorized patches is performed using the Euclidean distance metric. Each s_i is connected to its $\mathcal{K} = 10$ nearest neighbors s_j, resulting in $|\mathcal{E}|$ number of connections. In the third step the graph weight matrix W is computed using the Gaussian kernel weighting scheme (eq. (1)), for which the parameter σ is set experimentally as the average distance of the connected samples. Finally, the combinatorial Laplacian is computed.

Note that the computation of the weight matrix W for graph $\mathcal{G}$ costs $\mathcal{O}(n^4)$. For small n^2, we can use the above strategy directly. Although, the computation of W is expensive, it should be noted that with sufficiently small n^2, the graph can still be computed in the order of a few seconds. For big or high dimensional samples, $i.e$, large n^2, we can use a similar strategy but the computations can be made efficient ($\mathcal{O}(n^2 \log n^2)$) using the FLANN library (Fast Library for Approximate Nearest Neighbors searches in high dimensional spaces) [40]. However, the quality of the graphs constructed
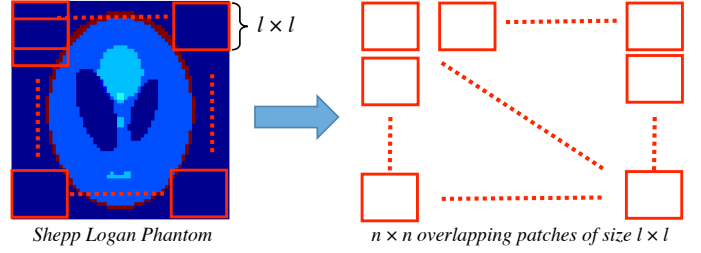

Shepp Logan Phantom   *n × n overlapping patches of size l × l*

Fig. 1: For the construction of patch graph, $x_{fbp} \in \mathbb{R}^{n \times n}$ is divide into n^2 overlapping patches of size $l \times l$ each.

using this strategy is slightly lower due to the approximate nearest neighbor search method. We describe the pros and cons of FLANN later in this manuscript.

C. Adaptive Graph Total Variation Regularization

The above description refers only to the non-adaptive part, where the graph $\mathcal{G}$ is fixed. It is important to point out that the initial estimate of the graph $\mathcal{G}$, obtained via the filtered back projection x_{fbp} is not very faithful to the final solution x. As x is being refined in every iteration, it is natural to update the graph $\mathcal{G}$ as well in every iteration. This simultaneous update of the graph $\mathcal{G}$ corresponds to the adaptive part of the proposed algorithm and its significance will be explained in detail in Section V of the paper.

IV. OPTIMIZATION SOLUTION

We make use of proximal splitting methods to solve problem (4). The specialty of such methods is that they can resolve a tedious complex problem into smaller and relatively trivial subproblems which are solved using proximal operators. The proximal operator of a function λh is defined as follows:

$$\text{prox}_{\lambda h}(y) = \operatorname*{argmin}_{x} \frac{1}{2}\|x - y\|_2^2 + \lambda h(x).$$

More detailed information about such methods can be found in [44, 45].

A. Forward-backward based primal dual

We cast our problem in the form and use the Forward-backward based primal dual method to solve it.

$$\operatorname*{argmin}_{x} f(x) + g(Ax) + h(B(x)). \quad (6)$$

The first term of (2), $f : \mathbb{R}^{n^2} \to \mathbb{R}$ is a convex differentiable function defined as $f(x) = \|Ax - b\|_2^2$. This function has a β-Lipschitz continuous gradient

$$\nabla_f(x) = 2A^\top(Ax - b).$$

Note that $\beta = 2\|A\|_2$ where $\|A\|_2$ is the spectral norm (or maximum eigenvalue) of A. The constant β has important implications in deciding the time step in the iterative optimization methods.

The second term of (6), $g : \mathbb{R}^{|\mathcal{E}|n} \to \mathbb{R}$, where $|\mathcal{E}|$ denotes the cardinality of $\mathcal{E}$ the set of edges in $\mathcal{G}$, is a convex function defined as $g(D) = \gamma_1 \|D\|_1$. The proximal operator of function g is

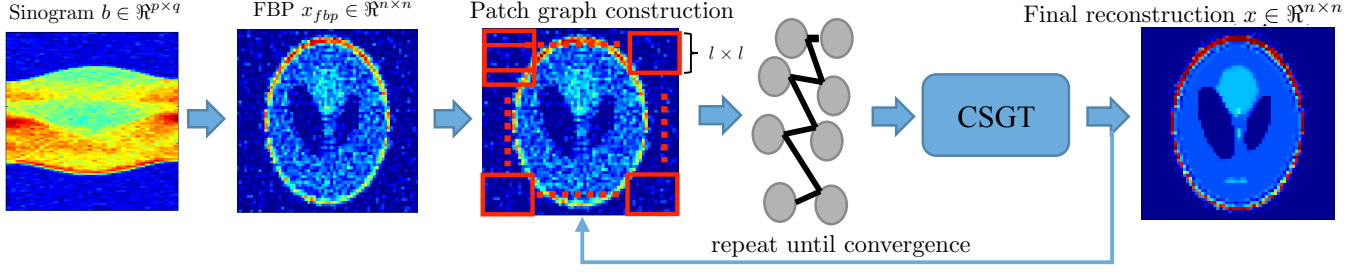$$\text{prox}_{\lambda g}(D) = \text{sgn}(D) \circ \max(|D| - \lambda\gamma_1, 0), \quad (7)$$

Fig. 2: The complete methodology for ACSGT. The input sinogram / projections $b \in \mathbb{R}^{p \times q}$ is first used to obtained a filtered back projection (FBP) $x_{fbp} \in \mathbb{R}^{n \times n}$. It is then used to construct the initial patch graph $\mathcal{G}$ to be used by the CSGT method. The output of CSGT is used to refine / reconstruct the graph and this process is repeated until convergence.

where $\circ$ denotes the Hadamard product. For our case the linear operator A in (6), is ∇_G.

The proximal operator of the function h is the ℓ_1 soft-thresholding given by the elementwise operations.

$$\text{prox}_{\gamma h}(B(x)) = \text{sgn}(B(x)) \circ \max(|B(x)| - \lambda, 0), \quad (8)$$

where $B = \Phi^*$ denotes the adjoint wavelet operator and $B(x)$ denotes the wavelet transform of x.

B. Algorithm

Using these tools, we can use the forward backward based primal dual approach presented in [44] to define Algorithm 1 where τ_1, τ_2, τ_3 are convergence parameters ϵ the stopping tolerance and I, J the maximum number of iterations. δ is a very small number to avoid a possible division by 0. Since we use Unlocbox [46] for solving the optimization algorithm, the convergence parameters τ_1, τ_2, τ_3 are set automatically according to the specified β. U_j corresponds to the primal and V_j to the dual variable in Algorithm 1.

More specifically, Algorithm 1 is based on a forward-backward approach [47]. It combines a gradient descent step (forward step) with a computation step involving a proximity operator (step 1a in Algorithm 1). Note that the gradient in this step is w.r.t. the differentiable function f, to which the result of the application of adjoint operator of g, i.e, $\nabla_{\mathcal{G}}^*$ is added. Then, the proximity step corresponds to the application of the proximal operator of h, which is an element wise soft-thresholding, on this result. This computation corresponds to a kind of subgradient step performed in an implicit (or backward) manner [45]. A deeper justification of this terminology is provided by the theory of monotone operators [48].

C. Computational Complexity

Complexity of Graph Construction: As mentioned earlier we use the Fast Approximate Neartest Neighbors search algorithm (FLANN) [40]. The computational complexity of the FLANN algorithm for n^2 patches of size l^2 each and fixed K is $\mathcal{O}(n^2 \log(n^2))$. Note that l^2 and $\mathcal{K}$ do not appear in the complexity because they are constants. Furthermore, n^2 is the size of the sample under consideration so the computational complexity is much lower as compared to the NLTV [35] based methods.

Algorithm Complexity: Let J denote the number of iterations for the algorithm (the for loop in Algorithm 1) to

---

Algorithm 1 Forward-backward primal dual for ACSGT

$x_0 = x_{fbp}$
1. INPUT: $U_0 = x_0$, $V_0 = \nabla_{\mathcal{G}} x_0$, $\epsilon > 0$
for $j = 0, \dots J - 1$ **do**
    a. $P_j = \text{prox}_{\tau_3 h} \left(\Phi^*(U_j) - \tau_3 \Phi^* \left(\nabla_f(U_j) + \nabla_{\mathcal{G}}^* V_j \right) \right)$
    b. $T_j = V_j + \tau_2 \nabla_{\mathcal{G}}(2P_j - U_j)$
    c. $Q_j = T_j - \tau_2 \text{prox}_{\frac{1}{\tau_2} g} \left(\frac{1}{\tau_2} T_j \right)$
    d. $(U_{j+1}, V_{j+1}) = (U_j, V_j) + \tau_1 \left((P_j, Q_j) - (U_j, V_j) \right)$
    if $\frac{\|U_{j+1} - U_j\|_F^2}{\|U_j\|_F^2 + \delta} < \epsilon$ and $\frac{\|V_{j+1} - V_j\|_F^2}{\|V_j\|_F^2 + \delta} < \epsilon$ **then**
        BREAK
    end if
end for
2. $x = U_j + 1$
3. Construct patch graph $\mathcal{G}$ from x
Repeat steps 1 to 3 for I iterations
OUTPUT: $x = U_j + 1$

---

converge, ad I the number of outer iterations (step 4 of Algorithm 1), then the computational cost of our algorithm is $\mathcal{O}((J|\mathcal{E}|I)$, where $|\mathcal{E}|$ denotes the number of non-zeros edges in the graph $\mathcal{G}$. For a $\mathcal{K}$-nearest neighbors graph $|\mathcal{E}| \approx \mathcal{K}n^2$ so the computational complexity of our algorithm is linear in the size of the data sample n^2, i.e $\mathcal{O}((J\mathcal{K}n^2 I)$.

Overall Complexity: The complexity of our algorithm is $\mathcal{O}((J\mathcal{K}n^2 I)$ and the graph $\mathcal{G}$ is $\mathcal{O}(n^2 \log(n^2))$. The graph $\mathcal{G}$ needs to be updated once in every outer iteration of the algorithm I, thus the overall complexity of the proposed ACSGT method is $\mathcal{O}(I(J\mathcal{K}n^2 + n^2 \log(n^2)))$.

V. WORKING EXPLANATION OF ACSGT

We present a simple example to motivate the use of ACSGT rather than simple CSGD and CSTV. Clearly, the compressed sensing part of all these methods is responsible for retrieving the sample x from the projections b. Thus, our comparison study is focused on the two regularizers, i.e, Adaptive Graph Total Variation (AGTV) and Total Variation (TV). Our two step exposition is described below:

1) CSGD is better than CSTV.
2) Adaptive Graph Total Variation (ACGST) is better than CSGD.

Consider the example of a Shepp-Logan Phantom as shown in top leftmost plot of Fig. 4. The goal is to recover this
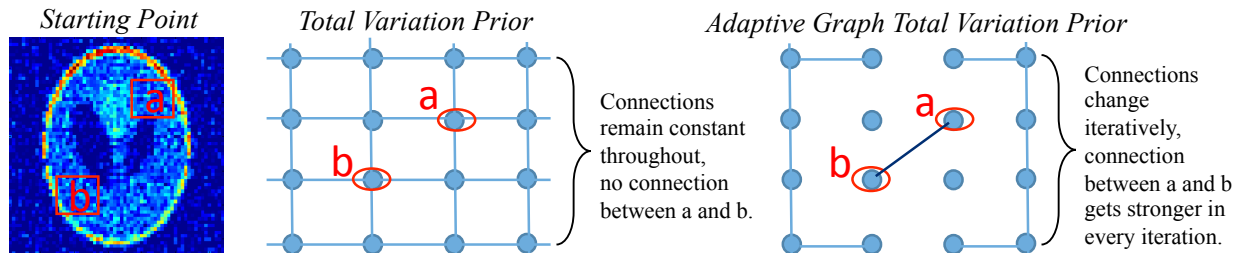
Fig. 3: A comparison of the Total Variation (TV) and Adaptive Graph Total Variation (AGTV) priors for the methods CSGT and ACSGT. The TV prior does not connect patches 'a' and 'b' which possess structural similarity, whereas the GTV prior connects them because the $\mathcal{K}$-nearest neighbor graph is not restricted to spatial neighbors only. Furthermore, this connection keeps getting stronger due to iterative removal of noise and graph updates in every iteration.

phantom from its noisy projections so that the recovered sample is faithful to its original clean version. The CSTV method requires a total variation prior to recover the sample while the CSGD method requires a graph total variation prior for the recovery. Both methods need an initial estimate for the construction of this prior, therefore, for the ease of demonstration we use the filtered back projection (FBP) as an initial estimate of the sample. Recall that our proposed method decomposes the FBP into $n \times n$ patches of size $l \times l$ each. Let (i, j) denote the (horizontal, vertical) position of the center of each patch then:

- For the total variation, each patch $s_{i,j}$ is connected to its spatial neighbors only, *i.e*, $s_{i+1,j}, s_{i-1,j}, s_{i,j+1}, s_{i,j-1}$, as shown in Fig. 3. These connections are fixed throughout the algorithm.
- For the graph total variation, each patch $s_{i,j}$ is only connected to the patches which are among the $\mathcal{K}$ nearest neighbors. Note that unlike TV the connected patches can be spatially far from each other.

Now let us take the example of two patches 'a' and 'b' as labeled in the FBP of Fig. 3. Comparing with the clean phantom in Fig. 4 it is obvious that these patches should possess the same texture at the end of the reconstruction algorithm. Therefore, an intelligent regularizer should take into account the inherent similarity between these patches. To explain the difference between the TV and GTV priors we use a point model as shown in Fig. 3, where each point corresponds to a patch in the FBP. Since 'a' and 'b' are not spatially co-located, the total variation prior does not establish any connection between these patches. Thus, TV fails to exploit the similarity between these patches throughout the algorithm. This leads to slightly different textures for the two patches, as shown in the 3rd row of Fig. 4.

Now consider the case of GTV. Even though the intial estimate of graph $\mathcal{G}$ is obtained from the noisy estimate of sample, *i.e*, the FBP, patches 'a' and 'b' still possess enough structural resemblance to be connected together by an edge (even if it is weak) in the graph. Now, if the graph is kept fixed which is the case of CSGD, one still obtains a better result as compared to CSTV, as shown in the 4th row of Fig. 4. This is due to the fact that the important connections are established by the graph $\mathcal{G}$ and similarity of patches is not restricted to spatially co-located patches only. This is also obvious from the intensity profile analysis in the 4th row of Fig. 4. Finally, we discuss the case of ACSGT, where the graph $\mathcal{G}$ is updated

in every iteration of the algorithm. Obviously, every iteration of the algorithm leads to a cleaner sample and updating the graph $\mathcal{G}$ is only going to make the connection between the patches 'a' and 'b' stronger. This leads to significantly better result than CSTV and CSGD as shown in Fig. 4. Note that the patches 'a' and 'b' possess almost the same structure at the end of ACSGT.

VI. EXPERIMENTAL RESULTS

Experiments were performed using two open source tool-boxes, GSPBox [49] for the graph construction and UN-LocBox for the convex optimization part [46]. These toolboxes provide fast and efficient general purpose algorithms, with an automatic tuning of many implicit non-model parameters, such as step size for iterative optimization algorithms. They are suitable for solving a broad range of problems and can be used as a black-box for computations.

To test the performance of our ACSGT method, we perform reconstructions for many different types of phantoms from different number of projections with varying levels of Poisson noise. Throughout this section, we report the reconstruction results of various phantoms in terms of the ℓ_2 reconstruction error. We compare the performance of ACSGT with many state-of-the-art iterative and convex optimization based algorithms, which include FBP, ART (Kaczmarz), SIRT (Cimmino), CS, CSTV and CSGT.

Each of these methods has its own model parameters, which need to be set or tuned in an appropriate manner. ART (Kaczmarz) and SIRT (Cimmino) were performed using FBP as *a priori*. The stopping criteria for ART and SIRT was set to 100 iterations and the relaxation parameter (η) was tuned to achieve the best result. For the CS method, the reconstruction was performed for uniformly spaced values of λ in the range $(0, 10)$ and the best λ was selected based on the minimum ℓ_2 reconstruction for the phantoms. For CSTV, the reconstruction was performed for sparsity parameter $\lambda \in (0, 1)$ and TV parameter $\gamma \in (0, 10)$ and the parameters for the best result were selected. For the graph based reconstruction (CSGT, ACSGT) a graph prior $\mathcal{G}$ was generated by dividing the result from FBP into patches as explained in Section III-B. For example, for a Shepp-Logan phantom of size 64×64, the graph was constructed by dividing it into $64 \times 64 = 4096$ overlapping patches of size 3×3. A graph $\mathcal{G}$ was constructed between the 4096 patches with 15 nearest neighbors ($\mathcal{K} = 15$) and σ for the weight matrix (Section II) was set to the average
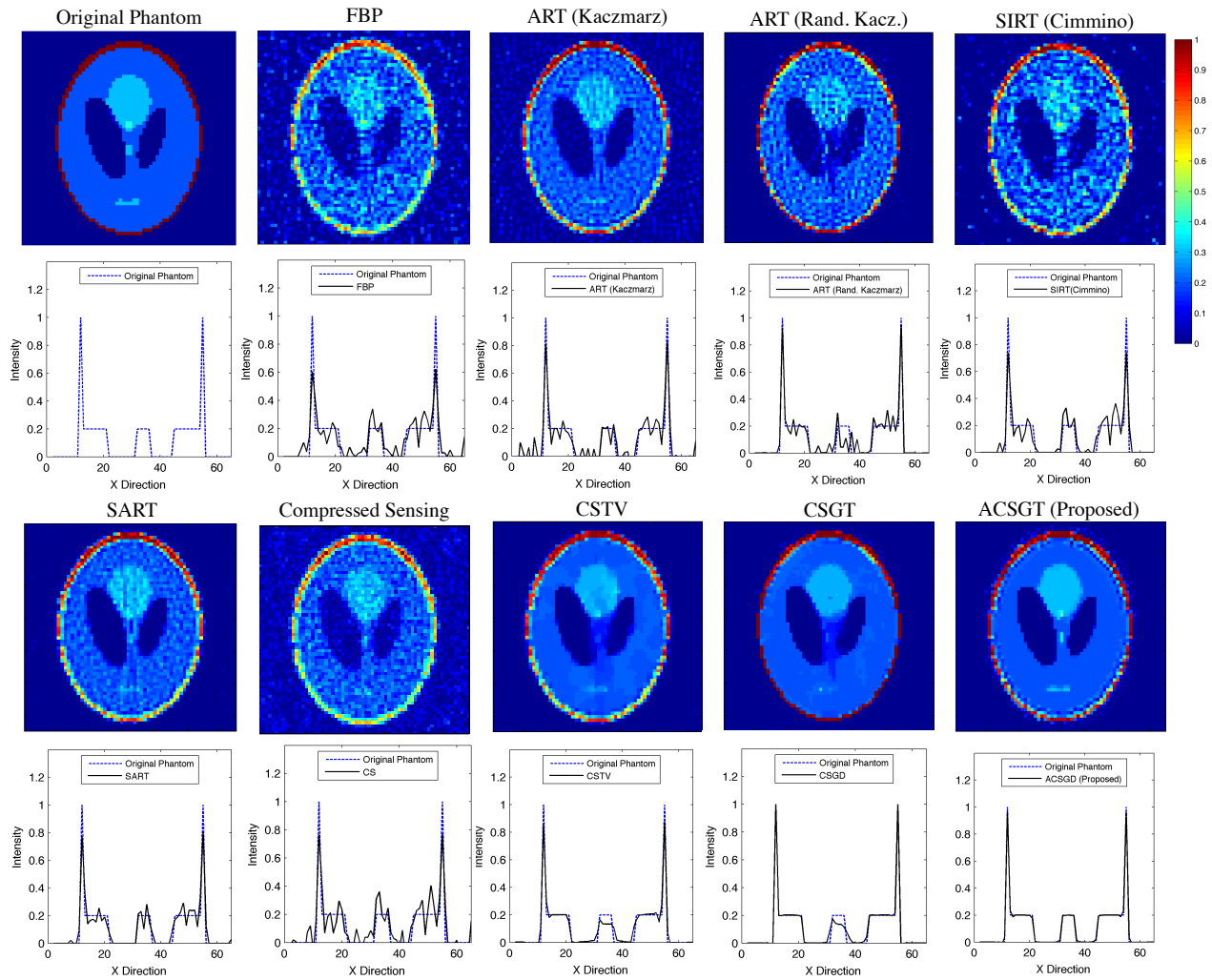
Fig. 4: Comparative analysis of reconstructing Shepp-Logan using various reconstruction methods. The sinogram of a 64×64 Shepp-Logan phantom corrupted with 10% Poission noise was reconstructed using FBP (Linearly interpolated, Cropped Ram-Lak filter); ART (Kaczmarz/Randomized Kaczmarz, Relaxation Parameter (η) = 0.25, Prior: FBP, Stopping Criteria = 100 iterations); SIRT (Cimmino/SART, (η) = 0.25, Prior: FBP, Stopping Criteria = 100 iterations); CS (500 Iterations, Prior: FBP); CSTV (λ = 0.5, γ = 0.1, Prior: FBP, Stopping Criteria = 100 iterations); CSGD (λ = 0.5, γ = 0.2, Prior: Patch Graph from FBP, Stopping Criteria = 100 iterations); ACSGT (λ = 0.5, γ = 1, Prior: Patch Graph from FBP updated every iteration, I and J in Algorithm 1 set to 30). ACSGT clearly gives a better internsity profile as compared to all other methods while preserving the edges.
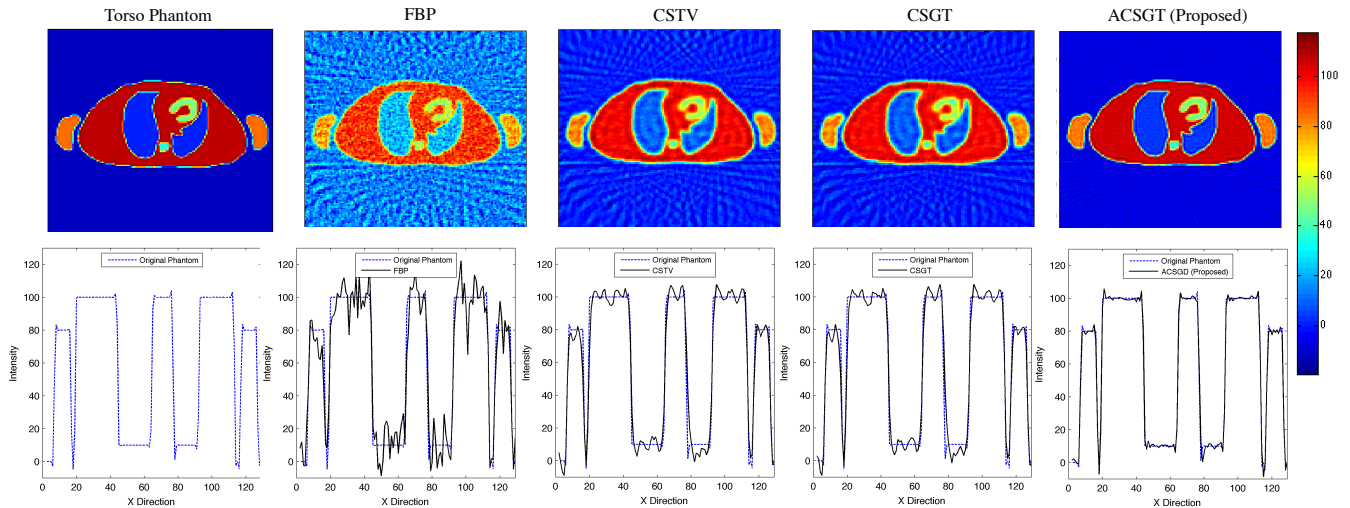

Fig. 5: Comparative analysis of reconstructing a Torso phantom using various reconstruction methods. The sinogram of a 128×128 Torso phantom corrupted with 5% Gaussian Random noise was reconstructed using FBP (Linearly interpolated, Cropped Ram-Lak filter); CSTV (λ = 0.5, γ = 0.1, Prior: FBP, Stopping Criteria = 100 iterations); CSGD (λ = 0.5, γ = 0.2, Prior: Patch Graph from FBP, Stopping Criteria = 100 iterations); ACSGT (λ = 0.5, γ = 0.1, Prior: Patch Graph from FBP updated every iteration, I and J in Algorithm 1 set to 30).
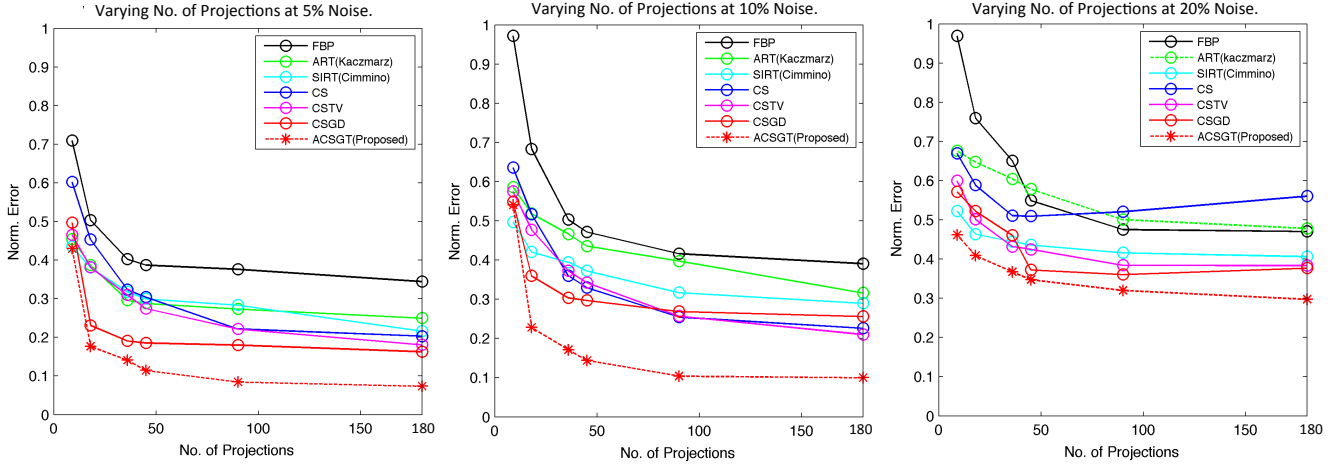
Fig. 6: Comparative analysis of reconstructing a Shepp-Logan phantom using various reconstruction methods at 5% and 10% Poisson noise. FBP (Linearly interpolated, Cropped Ram-Lak filter); ART (Kaczmarz/Randomized Kaczmarz, Relaxation Parameter (η) = 0.25, Prior: FBP, Stopping Criteria = 100 iterations); SIRT (Cimmino/SART, (η) = 0.25, Prior: FBP, Stopping Criteria = 100 iterations); CS (500 Iterations, Prior: FBP); CSTV ($\lambda = 0.5$, $\gamma = 0.1$, Prior: FBP, Stopping Criteria = 100 iterations); CSGD ($\lambda = 0.5$, $\gamma = 0.2$, Prior: Patch Graph from FBP, Stopping Criteria = 100 iterations); ACSGT ($\lambda = 0.5$, $\gamma = 1$, Prior: Patch Graph from FBP updated every iteration, I and J in Algorithm 1 set to 30).

Fig. 7: A small experiment corresponding to the reconstruction of a 32×32 Shepp-Logan phantom from 36 projections $b \in \mathbb{R}^{36}$ using the pre-tuned parameters $\lambda = 0.1, \gamma = 5$ for different values of $\mathcal{K}$ ranging from 5 to 50. The results clearly show that the reconstruction is quite robust to the choice of $\mathcal{K}$, with a small error variation.
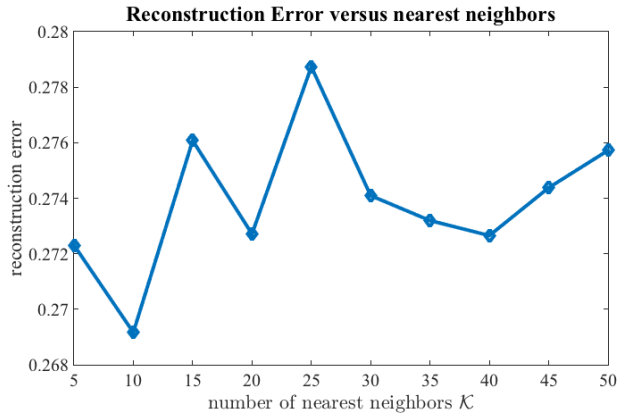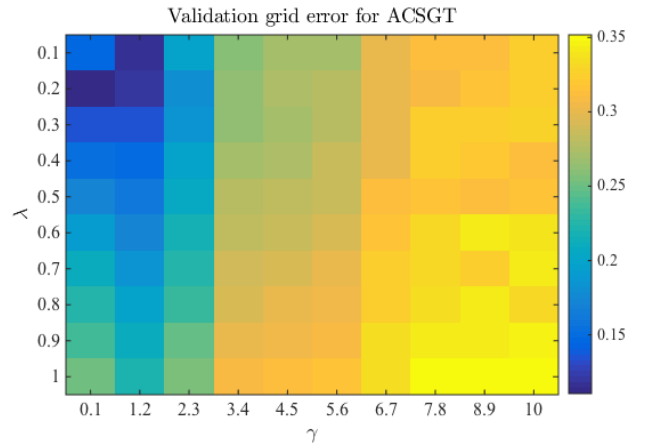

Fig. 8: A small experiment corresponding to the reconstruction of a 32×32 Shepp-Logan phantom from 36 projections $b \in \mathbb{R}^{36}$ using the full parameter gird $\lambda = (0.1, 1), \gamma = (0.1, 10)$ for a fixed value of $\mathcal{K} = 15$. The minimum clustering error occurs at $\lambda = 0.2, \gamma = 0.1$. The results clearly show that the reconstruction error increases smoothly with the parameters.

distance of the 15-nearest neighbors. For Algorithm 1, we set $I = J = 50$ and the convergence parameters τ_1, τ_2, τ_3 were set automatically by UNLocBox. It is worth mentioning here that our GTV based adaptive graph regularization is a faster method of implementing NLTV by using $\mathcal{K}$-nearest neighbors graph. Thus the GTV and NLTV based regularization are equivalent in performance. Therefore, we did not include comparisons with the NLTV based method.

To explain the performance of our model in detail we reconstructed a 64×64 Shepp-Logan [50] phantom from 36 erroneous projections. A sinogram S was built by projecting the phantom using Radon transform and 36 equally spaced projections were collected from 0 to 180 degrees. The sinogram was then corrupted with 10% Poission noise. Figure 6 provides a detailed comparison of the reconstruction of Shepp-Logan phantom via various algorithms along with the intensity profiles plotted underneath each of the reconstructions. It can

be clearly seen that ACSGT performs better than CSGT and CSTV. It is possible to appreciate this visually as the phantom obtained via ACSGT is very similar to the original phantom. Furthermore, a comparison of the intensity profiles of the two phantoms also reveals the same fact. The next best result is obtained by CSGT. Algorithmically, the only difference between CSGT and ACSGT is the regular graph update step in the latter, which tends to make the final reconstruction more faithful to the original phantom. CSTV also obtains a reasonable reconstruction, though worse than ACSGT. CS alone however, has a poor performance. This is not surprising, as for the tomography applications, CS has been mostly used in combination with TV, as it alone does not preserve the GMI. A similar experimental setup was repeated by reconstructing a 128×128 Torso phantom from 36 erroneous projections

corrupted with 5% Gaussian normalized noise (Figure 5).

A graphical comparison for the reconstruction of Shepp-Logan using various reconstruction methods at varying number of projections and noise levels has been given in Figure 6. From this figure it is clear that ACSGT method generally performs better as compared to many other state-of-the-art methods in the reconstruction task and follows a similar trend with respect to the number of projections. It is also interesting to note that the performance of ACSGT saturates after 90 projections for each of the three cases, i.e, the reconstruction error does not improve if the number of projections are increased. Furthermore, for each of the three noise cases one can observe that the drop in the reconstruction error from 50 to 90 projections is not significant. Although, the same observation can be made about CSGT, the error is a always higher than ACSGT. All the other methods, perform far worse than ACSGT. These tables clearly, lead to the conclusion that ACSGT is a step towards getting very fine reconstructions from a very small number of projections, via a scalable method.

Our model has two hyper-parameters, λ for tuning the sparsity of CS based reconstruction and γ to tune the amount of smoothing and denoising in the reconstruction. While, these are model hyper-parameters and need tuning, the graph parameter $\mathcal{K}$, *i.e*, the number of nearest neighbors is quite easy to set for our application. This is shown in Fig. 7 where we perform a small experiment corresponding to the reconstruction of a 32×32 Shepp-Logan phantom from 36 projections $b \in \mathbb{R}^{36}$ using the pre-tuned parameters $\lambda = 0.1, \gamma = 5$ for different values of $\mathcal{K}$ ranging from 5 to 50. The results clearly show that the reconstruction is quite robust to the choice of $\mathcal{K}$, with a small error variation. Thus, $\mathcal{K}$ is easy to set for our application. As the complexity of our proposed algorithm scales with the number of edges $|\mathcal{E}|$ in the graph $\mathcal{G}$ and $|\mathcal{E}| \approx \mathcal{K}n^2$, it is recommended to set $\mathcal{K}$ as small as possible. However, a very small $\mathcal{K}$ might lead to many disconnected components in the graph $\mathcal{G}$. On the other hand, a very large $\mathcal{K}$ might increase the time required for the algorithm to converge and reduce the computational advantage we have over the NLTV method. Therefore, we choose to set $\mathcal{K} = 15$ for our experiments.

In order to show the variation of reconstruction error with (λ, γ) grid, we perform another experiment for the reconstruction of the Shepp-Logan phantom of size 32×32 from 36 projections. For this experiment we keep $\mathcal{K} = 15$ and perform the reconstruction for every pair of parameter values in the tuple (λ, γ), where $\lambda \in (0.1, 1)$ and $\gamma \in (0.1, 10)$. The reconstruction error grid is shown in Fig. 8. The minimum error 0.11 occurs at $\lambda = 0.2, \gamma = 0.1$. It is also interesting to note that the error increases gradually with an increase in the parameter values.

VII. SHORTCOMINGS & FUTURE DIRECTIONS

The proposed ACSGT method has proven to produce much better reconstructions as compared to the state-of-the-art CSTV method. Although, the proposed method is computationally far less cumbersome than NLTV, it still suffers from a few problems which we discuss in this section.

The computational complexity of the proposed method is $\mathcal{O}(I(J\mathcal{K}n^2 + n^2 \log(n^2)))$. As already presented in Algorithm 1, the method requires a double loop, the outer with I iterations and the inner with J iterations. For our experiments we set $I = J = 50$. The main computational burden is offered by the graph construction, which needs to be performed every J iterations. Thus, the method still suffers from a high complexity because of the double loop and regular graph updates. The complexity of graph construction can be reduced by using a parallel implementation of FLANN which is provided by the authors [40]. The degree of parallelism can be increased at the cost of increasing approximation in the estimation of nearest neighbors. As a result of this the graph $\mathcal{G}$ will be different every time the FLANN algorithm is run. However, this does not effect the quality of the graph and for tomographic applications, negligible loss in the performance was observed. It is obviously of interest to reduce the number of inner iterations J and the complexity of the operations in the for loop. Our future work will therefore focus on introducing some approximations in the proposed algorithm to make it faster.

Tuning the hyperparameters is another short-coming of the proposed method. It is reasonable to set the number of $\mathcal{K}$-nearest neighbors to 10 or 15, however, the sparsity parameter λ and the GTV parameter γ need to be tuned properly and are not known beforehand. The results of the validation experiment from Fig.8 show that the error increases gradually with the parameter values. Our future work will thus also focus on finding smart methods to set these parameters automatically for specific tomographic applications.

VIII. CONCLUSIONS

Similar to NLTV our proposed method (ACSGT) goes beyond spatial similarity between different regions of an image being reconstructed by establishing a connection between similar regions in the image regardless of spatial distance. However, our approach is much more scalable and computationally efficient because it uses the approximate nearest neighbor search algorithm for graph construction, making it much more likely to be adapted in a clinical setting. Beyond NLTV, our proposed approach is adaptive. The non-local graph prior is updated every iteration making the connection between similar regions stronger. Thus improving the overall reconstruction quality as demonstrated by experiments. Since TV is a special case of graph TV the proposed method can be seen as a generalization of CS and TV methods and can promote future application specific studies for using CS for tomographic reconstruction from limited data.

REFERENCES

[1] J. Frank, *Electron tomography: methods for three-dimensional visualization of structures in the cell*. Springer Science & Business Media, 2008.

[2] A. Leis, M. Beck, M. Gruska, C. Best, R. Hegerl, and J. Leis, "Cryo-electron tomography of biological specimens," *IEEE Signal Processing Magazine*, vol. 23, no. 3, pp. 95–103, May 2006.

[3] A. Berrington de González, "Projected Cancer Risks From Computed Tomographic Scans Performed in the United States in 2007," *Archives of Internal Medicine*, vol. 169, no. 22, p. 2071, Dec. 2009.

[4] D. J. Brenner and E. J. Hall, "Computed tomography—an increasing source of radiation exposure," *N Engl J Med*, vol. 357, pp. 2277–84, 2007.

[5] M. S. Pearce, J. A. Salotti, M. P. Little, K. McHugh, C. Lee, K. P. Kim, N. L. Howe, C. M. Ronckers, P. Rajaraman, A. W. Craft *et al.*, "Radiation exposure from ct scans in childhood and subsequent risk of leukaemia and brain tumours: a retrospective cohort study," *The Lancet*, vol. 380, no. 9840, pp. 499–505, 2012.

[6] F. Natterer, *The mathematics of computerized tomography*. Siam, 1986, vol. 32.

[7] E. T. Quinto, U. Skoglund, and O. Öktem, "Electron lambda-tomography," *Proceedings of the National Academy of Sciences*, vol. 106, no. 51, pp. 21 842–21 847, 2009.

[8] J. Hsieh, "Computed tomography: principles, design, artifacts, and recent advances." SPIE Bellingham, WA, 2009.

[9] J. A. Fessler, "Statistical image reconstruction methods for transmission tomography," *Handbook of medical imaging*, vol. 2, pp. 1–70, 2000.

[10] Y. Censor, "Finite series-expansion reconstruction methods," *Proceedings of the IEEE*, vol. 71, no. 3, pp. 409–419, 1983.

[11] ——, "Row action methods for huge and sparse systems and their applications," *SIAM review*, vol. 23, no. 4, pp. 444–466, 1981.

[12] J. Qi and R. M. Leahy, "Iterative reconstruction techniques in emission computed tomography," *Physics in medicine and biology*, vol. 51, no. 15, p. R541, 2006.

[13] U. Skoglund, L.-G. Öfverstedt, R. M. Burnett, and G. Bricogne, "Maximum-Entropy Three-Dimensional Reconstruction with Deconvolution of the Contrast Transfer Function: A Test Application with Adenovirus," *Journal of Structural Biology*, vol. 117, no. 3, pp. 173–188, Nov. 1996.

[14] H. Rullgård, O. Öktem, and U. Skoglund, "A componentwise iterated relative entropy regularization method with updated prior and regularization parameter," *Inverse Problems*, vol. 23, no. 5, pp. 2121–2139, Oct. 2007.

[15] M. Beister, D. Kolditz, and W. A. Kalender, "Iterative reconstruction methods in X-ray CT," *Physica Medica*, vol. 28, no. 2, pp. 94–108, Apr. 2012.

[16] R. Gordon, R. Bender, and G. T. Herman, "Algebraic Reconstruction Techniques (ART) for three-dimensional electron microscopy and X-ray photography," *Journal of Theoretical Biology*, vol. 29, no. 3, pp. 471–481, Dec. 1970.

[17] G. Cimmino and C. N. delle Ricerche, *Calcolo approssimato per le soluzioni dei sistemi di equazioni lineari*. Istituto per le applicazioni del calcolo, 1938.

[18] P. C. Hansen and M. Saxild-Hansen, "AIR tools—a MATLAB package of algebraic iterative reconstruction methods," *Journal of Computational and Applied Mathematics*, vol. 236, no. 8, pp. 2167–2178, 2012.

[19] L. Landweber, "An iteration formula for Fredholm integral equations of the first kind," *American journal of mathematics*, vol. 73, no. 3, pp. 615–624, 1951.

[20] A. Brandt, "Algebraic multigrid theory: The symmetric case," *Applied mathematics and computation*, vol. 19, no. 1, pp. 23–56, 1986.

[21] T. Strohmer and R. Vershynin, "A randomized kaczmarz algorithm with exponential convergence," *Journal of Fourier Analysis and Applications*, vol. 15, no. 2, pp. 262–278, 2009.

[22] H. M. Hudson and R. S. Larkin, "Accelerated image reconstruction using ordered subsets of projection data," *IEEE transactions on medical imaging*, vol. 13, no. 4, pp. 601–609, 1994.

[23] H. Erdogan and J. A. Fessler, "Ordered subsets algorithms for transmission tomography," *Physics in medicine and biology*, vol. 44, no. 11, p. 2835, 1999.

[24] H. Nien and J. A. Fessler, "Fast x-ray ct image reconstruction using a linearized augmented lagrangian method with ordered subsets," *IEEE transactions on medical imaging*, vol. 34, no. 2, pp. 388–399, 2015.

[25] C. G. Graff and E. Y. Sidky, "Compressive sensing in medical imaging," *Applied optics*, vol. 54, no. 8, pp. C23–C44, 2015.

[26] G.-H. Chen, J. Tang, and S. Leng, "Prior image constrained compressed sensing (piccs): a method to accurately reconstruct dynamic ct images from highly undersampled projection data sets," *Medical physics*, vol. 35, no. 2, pp. 660–663, 2008.

[27] J. Song, Q. H. Liu, G. A. Johnson, and C. T. Badea, "Sparseness prior based iterative image reconstruction for retrospectively gated cardiac micro-ct," *Medical physics*, vol. 34, no. 11, pp. 4476–4483, 2007.

[28] L. Ritschl, F. Bergner, C. Fleischmann, and M. Kachelrieß, "Improved total variation-based ct image reconstruction applied to clinical data," *Physics in medicine and biology*, vol. 56, no. 6, p. 1545, 2011.

[29] J. Tang, B. E. Nett, and G.-H. Chen, "Performance comparison between total variation (TV)-based compressed sensing and statistical iterative reconstruction algorithms," *Physics in Medicine and Biology*, vol. 54, no. 19, pp. 5781–5804, Oct. 2009.

[30] Z. Tian, X. Jia, K. Yuan, T. Pan, and S. B. Jiang, "Low-dose ct reconstruction via edge-preserving total variation regularization," *Physics in medicine and biology*, vol. 56, no. 18, p. 5949, 2011.

[31] R. Leary, Z. Saghi, P. A. Midgley, and D. J. Holland, "Compressed sensing electron tomography," *Ultramicroscopy*, vol. 131, pp. 70–91, Aug. 2013.

[32] Z. Saghi, G. Divitini, B. Winter, R. Leary, E. Spiecker, C. Ducati, and P. A. Midgley, "Compressed sensing electron tomography of needle-shaped biological specimens – Potential for improved reconstruction fidelity with reduced dose," *Ultramicroscopy*, vol. 160, pp. 230–238, Jan. 2016.

[33] J. Provost and F. Lesage, "The Application of Compressed Sensing for Photo-Acoustic Tomography," *IEEE Transactions on Medical Imaging*, vol. 28, no. 4, pp. 585–594, Apr. 2009.

[34] Y. Lou, X. Zhang, S. Osher, and A. Bertozzi, "Image recovery via nonlocal operators," *Journal of Scientific Computing*, vol. 42, no. 2, pp. 185–197, 2010.

[35] G. Peyré, S. Bougleux, and L. Cohen, "Non-local regularization of inverse problems," in *European Conference on Computer Vision*. Springer, 2008, pp. 57–68.

[36] G. Gilboa and S. Osher, "Nonlocal operators with applications to image processing," *Multiscale Modeling & Simulation*, vol. 7, no. 3, pp. 1005–1028, 2008.

[37] J. Huang and F. Yang, "Compressed magnetic resonance imaging based on wavelet sparsity and nonlocal total variation," in *2012 9th IEEE International Symposium on Biomedical Imaging (ISBI)*. IEEE, 2012, pp. 968–971.

[38] J. Liu, H. Ding, S. Molloi, X. Zhang, and H. Gao, "Ticmr: Total image constrained material reconstruction via nonlocal total variation regularization for spectral ct," 2016.

[39] X. Jia, Y. Lou, B. Dong, Z. Tian, and S. Jiang, "4d computed tomography reconstruction from few-projection data via temporal non-local regularization," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2010, pp. 143–150.

[40] M. Muja and D. G. Lowe, "Scalable nearest neighbor algorithms for high dimensional data," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 11, pp. 2227–2240, 2014.

[41] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The Emerging Field of Signal Processing on Graphs: Extending High-Dimensional Data Analysis to Networks and Other Irregular Domains," *arXiv preprint arXiv:1211.0053*, 2012.

[42] D. L. Donoho, "Compressed sensing," *IEEE Transactions on information theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[43] E. J. Candès and M. B. Wakin, "An introduction to compressive sampling," *IEEE signal processing magazine*, vol. 25, no. 2, pp. 21–30, 2008.

[44] N. Komodakis and J.-C. Pesquet, "Playing with duality: An overview of recent primal? dual approaches for solving large-scale optimization problems," *IEEE Signal Processing Magazine*, vol. 32, no. 6, pp. 31–54, 2015.

[45] P. L. Combettes and J.-C. Pesquet, "Proximal splitting methods in signal processing," in *Fixed-point algorithms for inverse problems in science and engineering*. Springer, 2011, pp. 185–212.

[46] N. Perraudin, D. Shuman, G. Puy, and P. Vandergheynst, "Unlocbox a matlab convex optimization toolbox using proximal splitting methods," *arXiv preprint arXiv:1402.0779*, 2014.

[47] P. L. Combettes and V. R. Wajs, "Signal recovery by proximal forward-backward splitting," *Multiscale Modeling & Simulation*, vol. 4, no. 4, pp. 1168–1200, 2005.

[48] H. H. Bauschke and P. L. Combettes, *Convex analysis and monotone operator theory in Hilbert spaces*. Springer Science & Business Media, 2011.

[49] N. Perraudin, J. Paratte, D. Shuman, V. Kalofolias, P. Vandergheynst, and D. K. Hammond, "Gspbox: A toolbox for signal processing on graphs," *arXiv preprint arXiv:1408.5781*, 2014.

[50] L. A. Shepp and B. F. Logan, "The Fourier reconstruction of a head section," *IEEE Transactions on Nuclear Science*, vol. 21, no. 3, pp. 21–43, Jun. 1974.