Searching Effective Parameters for Low-Dose CT Reconstruction by Ant Colony Optimization

Ziyi Zheng, Eric Papenhausen and Klaus Mueller

***Abstract*— Low-dose Computed Tomography (CT) has been gaining substantial interest, due to the growing concerns with regards to the X-ray dose delivered to the patient. To cope with the limited data collected at 30% of standard radiation, low-dose CT reconstruction algorithms generally require several iterations of forward projection, back-projection and regularization, involving many parameters to achieve desirable quality and speed. The interactions among these parameters can be complex, and thus effective combinations can be difficult to identify for a given scanning scenario. Non-optimized parameters can result in increased reconstruction time and reduced image quality. As a result, the parameter choice is often left to a highly-experienced expert. In this work we focus on an automatic parameter optimization framework. In the pre-computation step of our parameter learning framework, effective parameters are learned with the access of a known gold-standard. As these domain-specific and algorithm-dependent parameters are obtained, they are applied in the similar incoming data. In addition, an edge-enhancement component is introduced and automatically tuned to increase the sharpness in iterative reconstruction. The preliminary results with the non-local mean filter indicate that our parameter optimization scheme can identify effective parameters resulting in sharper results than those results generated by popular iterative reconstructions algorithms.**

Index Terms*—*Iterative reconstruction, low-dose CT, ant colony optimization, non-local mean filter

I. INTRODUCTION

Cone-beam CT has become to be a major imaging technique thanks to its image-fidelity and scan time. The traditional cone-beam CT reconstruction method is the FDK [1] algorithm, which provides high resolution results but requires several hundreds of patient X-ray projections. With the growing concern about the potential risk of X-ray radiation exposure to the human body, low-dose CT has become a significant research topic. Dose reduction usually involves lowering the X-ray energy per projection and/or reducing the total number of projections. Both methods typically suffer from low signal-to-noise ratio (SNR) in the reconstructions. Iterative reconstruction schemes, matched with suitable regularization methods have been shown to cope well with these few-view or high-noise scenarios.

Iterative methods typically offer a diverse set of parameters that allow control over quality and computation speed, often requiring trade-offs. These expert-picked parameters need to be learned from many experiments which are both domain-specific and algorithm-dependent. Thus the hand-tuning of the parameters in iterative algorithm could be very time-consuming. Researchers have proposed methods to set the parameters by monitoring the convergence and updating reconstruction parameters on the fly [2]. In this work we focus on automatic parameter optimization by pre-computation. We propose a two-step method. The first step is the parameter learning step; which uses the known gold-standard. After parameters are trained, they are re-used in similar data with similar scanner settings. Furthermore, our framework can incorporate an interleaved un-sharp mask to avoid over-smoothing on the reconstruction results. We compared our results with those obtained by automatic controlled iterative reconstructions algorithms and our results indicate that the optimized parameters can resolve finer structures as good as or better than previous methods.

Our paper is organized as follows. Section 2 discusses related work. Section 3 describes our framework. Section 4 presents some initial results, and Section 5 concludes the paper.

Ziyi Zheng, Eric Papenhausen and Klaus Mueller are with the Computer Science Department, Stony brook University, Stony Brook, NY 11777 USA. (E-mail: { zizhen, epapenhausen, mueller}@cs.sunysb.edu).

II. RELATED WORKS

Recent research in iterative reconstruction demonstrates a trend of combining forward and back-projection with various regularization methods. The regularization algorithm usually employed is Total Variation Minimization (TVM) as used in the Adaptive-Steepest-Descent-Projection-Onto-Convex-Sets (ASD-POCS) algorithm [2]. On the other hand, de-noising filters, such as the bilateral filter [3] and the non-local means (NLM) filter [4], can also be used in regularization [5][6][7]. Compared to the ASD-POCS algorithm which can determine the TVM parameters on the fly, de-noising filters based reconstruction methods need a special control unit to guide regularization parameter during the convergence. Our framework fulfills this need and provides optimized parameters which can be directly applied on de-noising filter based regularization algorithm.

This work serves as an extension of previous research focused on optimizing bilateral filter-based iterative reconstruction, using genetic algorithms [8]. The ant colony optimization (ACO) [9] algorithm is inherently a path searching engine driven by a large number of artificial ants and is widely applied to solve many optimization problems. In this paper, we utilize the ACO algorithm to find optimal parameters in NLM-based iterative reconstruction. In addition, we improve the current existing regularization pipeline by adding an edge-boosting stage with optimized parameters.

III. APPROACH

A. Iterative Reconstruction with Regularization

We use the OS-SIRT algorithm with GPU-accelerated forward projection and back-projection [8]. The forward projection operator simulates X-ray images at a certain viewing angle φ. The result of this projection is then compared to the acquired image obtained at the same viewing angle. Here, the weight factor w_{il} determines the contribution of v_l to r_i and is given by the interpolation kernel used for sampling the volume. The forward projection is:

$$r_i = \sum_{l=1}^{N} w_{il} \cdot v_l \qquad i = 1,2,...,M \tag{1}$$

where M and N are the number of rays and voxels, respectively. The correction update is computed as:

$$v_j^{(k+1)} = v_j^{(k)} + \lambda \frac{\sum_{p_i \in OS_s} \frac{p_i - r_i}{\sum_{l=1}^{N} w_{il}} w_{ij}}{\sum_{i=1}^{N} w_{ij}} \tag{2}$$

$$r_i = \sum_{l=1}^{N} w_{il} \cdot v_l^{(k)} \tag{3}$$

Here, p_i represents the pixels in the M/S acquired images that form a specific (ordered) subset OS_S where $1 \le s \le S$ and S is the number of subsets. The factor λ is the relaxation factor that scales the corrective update to each voxel. This factor is important in balancing quality and speed and will be optimized by our framework. The factor k is the iteration counter. It will be incremented when all M projections have been processed. In the GPU implementation, only the forward projection uses a ray-driven approach, where each ray is a parallel thread and interpolates the voxels on its path. Conversely, the back-projection uses a voxel-driven approach, where each voxel is a parallel thread and interpolates the 2D correction projections. Thus, the weights used in the projection and the back-projection are slightly different but it has minimum effect on the reconstruction quality [5][8].

We use a 2D NLM filter [4] for regularization. The NLM filter is a non-linear filter that replaces the pixel located at x with the mean of the pixels whose Gaussian neighborhood looks similar to the neighborhood of x:

$$NLM(x) = \frac{\sum_{y \in W} e^{-\frac{\sum_{t \in N} G_a(t) \left| f(x+t) - f(y+t) \right|^2}{h^2}} f(y)}{\sum_{y \in W} e^{-\frac{\sum_{t \in N} G_a(t) \left| f(x+t) - f(y+t) \right|^2}{h^2}}} \tag{4}$$

Here, x, y and t are 2D spatial variables. W is the window centered at x. N is the 2D neighborhood centered at x or y. G_a is a 2D Gaussian kernel with a standard deviation a. The variable h acts as a parameter to control smoohting. Thus, the NLM filter contains several parameters to achieve best quality. For the performance, one observation of NLM filtering is that the computational time spend on NLM does not change as like it does in TVM regularization.

At the last stage of our regularization scheme, the un-sharp masking is used to avoid over-smoothness in iterative reconstruction. The control factor of sharpness is given by later optimizations.

B. Ant Colony Optimization (ACO)

The ant colony optimization (ACO) algorithm is a swarm intelligence method to search for good paths in discrete graphs. Intuitively, it launches a large number of artificial ants searching for best score. Each artificial ant independently moves in the graph and reports the scores it gets. The probability for an ant to choose an edge connecting two nodes is affected by the moving trend of all ants. More specifically, the probability for choosing one choice will increase if a large number of high score-ants choose it.

Iterative CT reconstruction can be modeled as a classic path searching problem. In this discrete graph of nodes and edges, each node represents a unique state of the volume being reconstructed, after a certain number of iterations and tagged with a score that encodes a quality metric. Each edge represents the computation of a correction pass and a regularization pass. The weight on an edge represents the time cost, which in our case is treated uniformly as one. The overall goal is to find the node having the highest score within a given cost. Since we have a uniform cost for NLM filtering, the problem is reduced to find the best score after a fixed number of steps.

We attempt to optimize six parameters including the relaxation factor λ in Equation (2), the h factor, Gaussian blur factor, window size and block size for the NLM filter in equation (4) and the un-sharp masking factor. These parameters can be different per iteration, resulting in an astronomical search space. For example, assuming we allow 100 discretized values for each parameter and allow 10 iterations, the search space will be 10^{120}. This search space is so huge that simple exhaustive search algorithm will fail to find the optimal solution in a reasonable amount of time.

We adopt a greedy heuristic to prune the search tree. This heuristic guides ants with a "best guess" for the path on which the solution lies. The greedy ant system only searches the best parameter setting for a single iteration, then adopts the best setting and moves on to the next iteration. The solution space can then be reduced from 10^{120} to 10^{13}. Assuming the optimal parameters would be similar or slightly adjusted for adjacent iterations, we can improve the search efficiency through pheromone control. The pheromone for the current iteration is re-used in the next iteration, making ants more likely to choose parameters similar to previous best parameters.

In the following, we describe our version of the ant system in detail. The probability for an ant to choose a discretized value j for ith parameter is:

$$P(i,j) = \frac{\tau_{ij}}{\sum_{q=0}^{R_i} \tau_{iq}} \tag{5}$$

where τ_{ij} is the pheromone on value j for ith parameter and R_i is the discrete resolution of the ith parameter. The value of τ_{ij}

is initially set to one; this will allow ants to make purely random decisions. After a small group of ants (in our case it is 10) finishes their moves for all 6 parameters, the pheromone is updated as:

$$\tau_{ij} \leftarrow (1-\rho)\tau_{ij} + \sum_{k=1}^{10} s_k/10 \qquad \forall j \in J \tag{6}$$

where s_k is the normalized score (with $0 < s_k < 1$) of the kth ant, as the amount of new pheromone put on ρ is the pheromone evaporation factor (with $0 < \rho < 1$). The range of pheromone is clamped within $[0, 1]$. The equation (6) updates the pheromone such that later ants will be more likely to follow the path of previous high-score ants. If in the current iteration no ants report better scores than in the previous iteration, we launch another group of ants until better scores are found or the maximum number of ants for one iteration is reached. In the absence of a human observer, s_k is generated by a computer based on a quality metric. In this paper, we use the correlation coefficient (CC) to determine s_k.

IV. RESULTS

In our experiments, we collected two sets of data from a cone-beam CT scanner, a Medtronic O-arm system. We perform the reconstruction algorithms on the central slice only. The detector 1D resolution was 1,024 pixels with pixel-size 0.388mm. The low-dose case was set as 60 evenly-distributed projections and the gold-standard was generated by the FDK algorithm with 360 projections. In the OS-SIRT scheme, we made 40 subsets which contain 1-2 projections. Head phantoms were used to test the parameter optimization algorithm and the training results are shown in Figure 1. The result from 60 iterations of the OS-SIRT algorithm with a constant λ=1 is in panel (a) and the gold-standard is in panel (b). 60 iterations of the ASD-POCS result is displayed in panel (c). We use SART instead of ART in this ASD-POCS implementation to make a fair comparison. The ASD-POCS result is smooth but has some typical cartoon-like structures due to the TVM scheme. On the other hand, our trained results (d) are the more similar to the gold standard (b) and can preserve sharper details than ASD-POCS can (c).

Figure 2 shows the plot of trained parameters through 60 iterations with CC scores. We can see that the CC score stops improving after 25 iterations. Figure 3 shows the plot of relaxation-correction factor λ in equation (2). The optimized parameter suggests starting with a bigger value around 2.0 and gradually decreasing to 0.0. Note the overall decreasing trend contains a certain amount of perturbation and there was very little correction after 40 iterations.

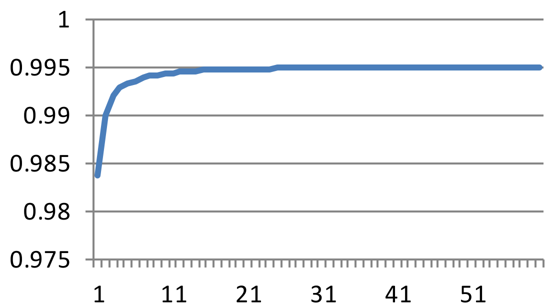
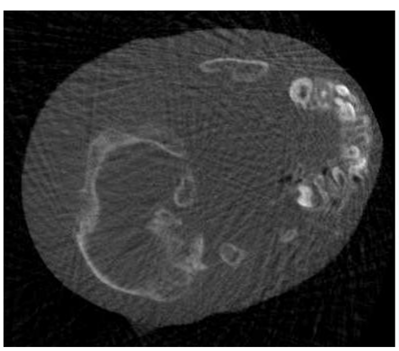
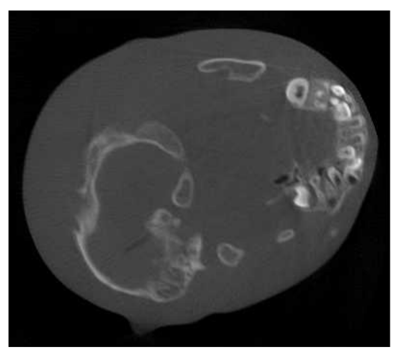


Figure 2. The correlation-coefficient CC through 60 iterations

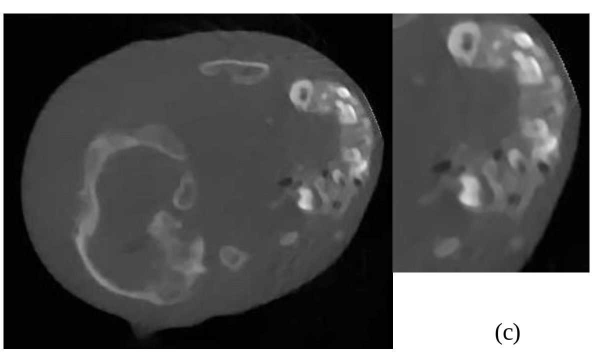
We tested the learned parameter setting on another similar



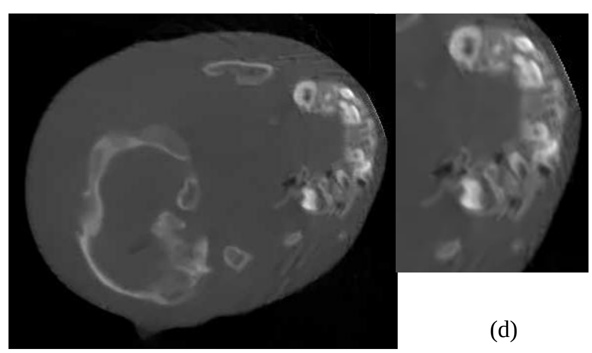
(a)



(b)



(c)



(d)

Figure 1. A central slice of a reconstruction for a training dataset. (a) low-dose OS-SIRT. (b) gold-standard FDK by 360 projections. (c) low-dose ASD-PCOS and (d) optimized low-dose. (a), (c) and (d) use 60 projections and 60 iterations.

head phantom. Now the central slice was shifted to another autonomy region in the head phantom. Figure 4 documents the image quality of the new dataset with (a) 60 iterations of OS-SIRT with a constant $\lambda=1$, (b) 360 projection FDK, (c) 60 iterations of ASD-POCS and (d) 60 iterations of optimized parameters. We observed that even in another scan, the parameters can guide the reconstruction to achieve better visual quality than ASD-POCS.

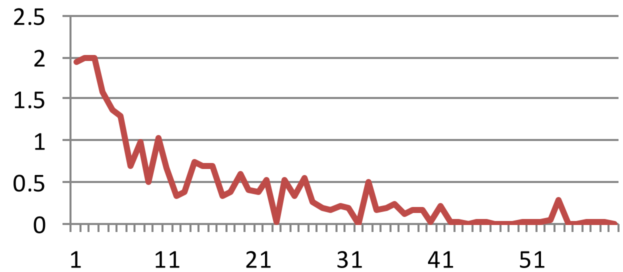


Figure 3. The λ in Equation (2) through 60 iterations

All of our experiments were conducted on an NVIDIA GTX 480 GPU, programmed with CUDA 3.2 runtime API and with an Intel Core 2 Duo CPU @ 2.66GHz. We display the reconstruction results simultaneously during the optimization. In the implementation, we use shared-memory in the GPU to perform prefetching, which enables 4× speedups [6]. Table I shows the configurations and performance of our framework. The iteration versus time does not scale linearly since later iterations will need to launch more ants to improve the score, especially in the case when the scores do not increase after a large number of iterations. We can avoid this over-fitting by detecting whether λ becomes close to zero.

TABLE I
PERFORMANCE OF ANT COLONY OPTIMIZATION

Maximum # of Ants	Iterations	Time(s)
1000	10	92
1000	30	265
1000	50	1.44k

V. CONCLUSIONS

We have devised an efficient framework to optimize various parameters for iterative CT reconstruction using an ant colony optimization algorithm. Our preliminary results show that the learned parameters can be readily applied to similar scans with promising results.

In future work, we would like to employ other perceptual quality metrics. We are also planning to extend the current framework to 3D reconstruction and study parameter optimization on different scanning scenarios, for example, patient size, X-ray tube's voltage and current.

ACKNOWLEDGMENT

We thank Medtronic for providing access to their O-arm scanner and head phantoms.

REFERENCES

[1] L. A. Feldkamp, L. C. Davis, and J. W. Kress, "Practical cone-beam algorithm," J. Opt. Soc. Am. Vol 1, No. A6, 612–619 , 1984

[2] E. Y. Sidky, X. Pan, "Image reconstruction in circular cone-beam computed tomography by constrained, total-variation minimization," Physics in Medicine and Biology, vol. 53, pp. 4777-4807, 2008

[3] C. Tomasi, and R. Manduchi, "Bilateral filtering for gray and color images," IEEE International Conference on Computer Vision, pp. 839-846, 1998.

[4] A. Buades, B. Coll, , J. M. Morel, "A non-local algorithm for image denoising," Computer Vision and Pattern Recognition, pp. 60-65, 2005.

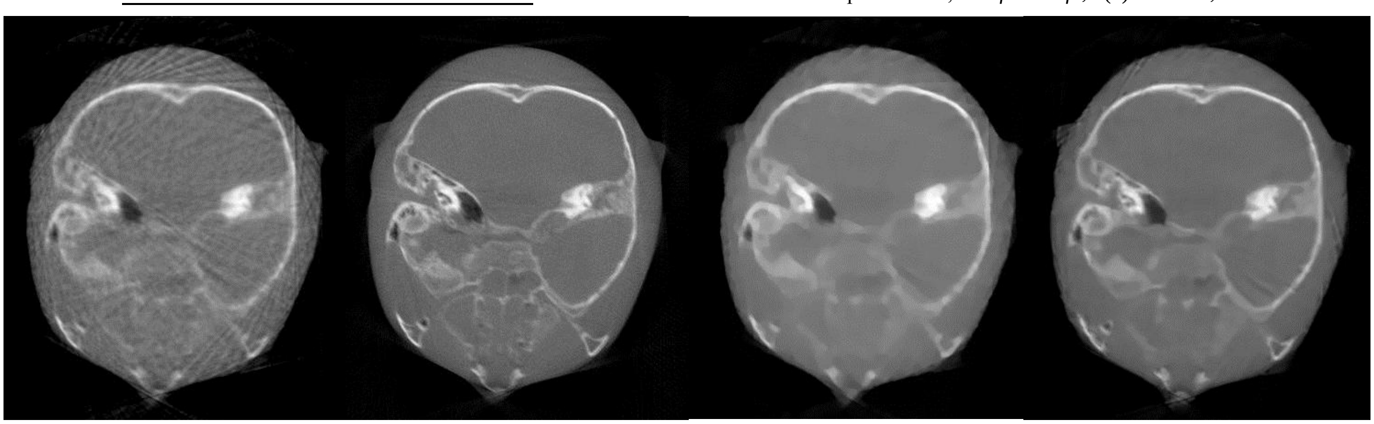
[5] W. Xu and K. Mueller, "A performance-driven study of regularization methods for GPU-accelerated iterative CT," In *Workshop on High Performance Image Reconstruction*, 2009.

[6] Z. Zheng, W. Xu, K. Mueller, "Performance tuning for CUDA-accelerated neighborhood denoising filters," In *Workshop on High Performance Image Reconstruction*, 2011

[7] J. Huang, J. Ma, N. Liu, H. Zhang, Z. Bian, Y. Feng, Q. Feng, W. Chen, "Sparse angular CT reconstruction using non-local means based iterative-correction POCS," *Computers in Biology and Medicine*, 41(4):195-205, 2011

[8] W. Xu and K. Mueller, "Using GPUs to learn effective parameter settings for GPU-accelerated iterative CT reconstruction algorithms", GPU Computing Gems Emerald Edition, Chapter 43, January 26, 2011

[9] M. Dorigo, G. DiCaro, and L. M. Gambardella, "Ant algorithms for discrete optimization," *Artificial Life*, 5(2):137–172,1999.



(a) (b) (c) (d)

Figure 4. A central slice of a reconstruction for a new dataset. (a) low-dose OS-SIRT. (b) gold-standard FDK by 360 projections. (c) low-dose ASD-PCOS and (d) optimized low-dose. (a), (c) and (d) use 60 projections and 60 iterations.