# INCORPORATING HANDCRAFTED FILTERS IN CONVOLUTIONAL ANALYSIS OPERATOR LEARNING FOR ILL-POSED INVERSE PROBLEMS

*Caroline Crockett*, David Hong*, Il Yong Chun**, and Jeffrey A. Fessler**

*Department of EECS, University of Michigan, Ann Arbor, Michigan, USA
**Department of EE, University of Hawai'i–Mānoa, Honolulu, Hawai'i, USA

## ABSTRACT

Convolutional analysis operator learning (CAOL) enables the unsupervised training of convolutional sparsifying autoencoders, taking advantage of large datasets to obtain high quality filters. In previous works, using CAOL within model-based image reconstruction (MBIR) for ill-posed inverse problems significantly improved image reconstruction accuracy over existing MBIR using non-trained regularizers and generalized better than existing non-MBIR deep neural network approaches. This paper modifies the CAOL Procrustes filter update to allow some filters to be handcrafted. Doing so makes it possible to incorporate domain knowledge in the learning process and accelerates CAOL by learning fewer filters. We apply the proposed generalization of CAOL to MBIR for sparse-view CT. Numerical experiments show that *1)* handcrafting discrete cosine transform filters can trade-off training time and reconstruction quality and *2)* handcrafting filters based on finite differences can speed up training without sacrificing reconstruction quality.

## 1. INTRODUCTION

New models and algorithms for image reconstruction attempt to reduce computation time and/or form higher quality images. For computed tomography (CT) image reconstruction, the goal is to help doctors diagnose patients while minimizing patient exposure to radiation [1]. To achieve lower radiation doses in CT, one often undersamples the data, making the inverse problem ill-posed. Thus, one must make assumptions about the properties of the images (a prior) to reconstruct them. Analytic models are commonly based on total variation, *e.g.*, [2, 3], or sparsity in the wavelet transform, *e.g.*, [4].

Learned priors can lead to better reconstruction performance than analytic models [5, 6], at a cost of increased training time and decreased generality. Learned priors have become increasingly popular due to greater availability of computational resources and large, labeled training datasets [7, 8].

Among prior learning approaches, convolutional dictionary learning (CDL) methods achieve lower redundancy in sparse representation and are therefore more memory efficient than synthesis patch-based dictionary learning methods, making it possible to train from larger data-sets [8,9]. A combination of a "blind" CDL model and total variation (TV) penalty was successfully applied to sparse-view CT model-based image reconstruction (MBIR) [10]. However, the model has large computational costs because it optimizes both a convolutional dictionary and corresponding sparse representations; benefits of the CDL model itself – without TV regularization – are yet unknown in sparse-view CT MBIR. An alternative prior learning approach is convolutional analysis operator learning (CAOL). CAOL is more amenable to large datasets than CDL [5], and has theoretical benefits from using more training samples [11]. In ill-posed inverse problems like sparse-view CT, applying learned convolutional analysis operators to MBIR has yielded significantly more accurate image reconstruction than existing MBIR with non-trained regularizers, *e.g.*, edge-preserving regularizers [6], and better generalization (and explainability) than existing non-MBIR deep neural network approaches [12].

This paper extends the CAOL framework to incorporate handcrafted filters. The generalization allows system designers to integrate domain-specific knowledge with adaption to the training data. We investigate how the number of handcrafted filters impacts training time and CT image reconstruction quality. Our numerical experiments show how handcrafting general purpose filters can trade-off between training time and CT reconstruction quality, and how handcrafting a few filters using domain-specific knowledge can lead to shorter training times while maintaining reconstruction quality.

## 2. FILTER LEARNING WITH CAOL

CAOL [5] learns $K$ filters, $\mathbf{h}_1, \dots \mathbf{h}_K \in \mathbb{C}^K$ from $N$ training samples. Unlike synthesis models, CAOL learns filters that sparsify, rather than synthesize, the signals. The cost function for learning these filters is [5]:

$$\hat{\mathbf{H}} = \underset{\mathbf{H}}{\operatorname{argmin}} \min_{\mathbf{Z}} \sum_{n=1}^{N} \sum_{k=1}^{K} \frac{1}{2} \|\mathbf{h}_k \circledast \mathbf{x}_n - \mathbf{z}_{k,n}\|_2^2 + \lambda \|\mathbf{z}_{k,n}\|_0 \quad \text{s.t.} \quad \mathbf{H}\mathbf{H}' = \frac{1}{K}\boldsymbol{I}, \tag{1}$$

This work was supported in part by NIH grants R01 EB023618 and U01 EB018753. Author contacts: {cecroc,dahong,fessler}@umich.edu, iychun@hawaii.edu

where $\circledast$ is circular convolution, $\mathbf{x}_n \in \mathbb{C}^M$ is the $n$th training sample, $\mathbf{H} = [\mathbf{h}_1, \ldots, \mathbf{h}_K]$ is a matrix of the filters, $\mathbf{z}_{k,n} \in \mathbb{C}^M$ is a sparse code, and $\lambda$ is a regularization parameter. The tight-frame constraint in (1) encourages filter diversity.

We briefly review the block coordinate descent (BCD) method for optimizing (1). BCD alternates between minimizing with respect to $\mathbf{z}_{k,n}$ and with respect to $\mathbf{H}$.

The sparse code update is separable, yielding:

$$\mathbf{z}_{k,n}^{(i+1)} = \underset{\mathbf{z}_{k,n}}{\operatorname{argmin}} \frac{1}{2}\|\mathbf{h}_k^{(i)} \circledast \mathbf{x}_n - \mathbf{z}_{k,n}\|_2^2 + \alpha\|\mathbf{z}_{k,n}\|_0 \qquad (2)$$

$$= \tau(\mathbf{h}_k^{(i)} \circledast \mathbf{x}_n, \sqrt{2\alpha}), \text{ where } \tau(y,\alpha) = \begin{cases} y & |y| > \alpha \\ 0 & \text{else} \end{cases}$$

is the element-wise hard thresholding operator.

Letting $\mathbf{X}_n$ denote the data matrix for which $\mathbf{X}_n\mathbf{h}_k$ is equivalent to $\mathbf{x}_n \circledast \mathbf{h}_k$, the filter update can be written in matrix form [11]:

$$\underset{\mathbf{H}}{\operatorname{argmin}} \left\|\mathbf{X}\mathbf{H} - \mathbf{Z}^{(i+1)}\right\|_{\mathrm{F}}^2, \quad \text{s.t. } \mathbf{H}\mathbf{H}' = \frac{1}{K}\boldsymbol{I}, \text{ where}$$

$$\mathbf{X} = \begin{bmatrix} \mathbf{X}_1 \\ \vdots \\ \mathbf{X}_N \end{bmatrix} \text{ and } \mathbf{Z}^{(i+1)} = \begin{bmatrix} \mathbf{z}_{1,1}^{(i+1)} \ldots \mathbf{z}_{K,1}^{(i+1)} \\ \vdots \\ \mathbf{z}_{1,N}^{(i+1)} \ldots \mathbf{z}_{K,N}^{(i+1)} \end{bmatrix}. \quad (3)$$

Defining $\mathbf{Q} = \sqrt{K}\mathbf{H}'$, $\mathbf{B} = \mathbf{Z}^{(i+1)'}$, and $\mathbf{A} = \frac{1}{\sqrt{K}}\mathbf{X}'$ yields the standard Procrustes problem [13]:

$$\hat{\mathbf{Q}} = \underset{\mathbf{Q}}{\operatorname{argmin}} \|\mathbf{B} - \mathbf{Q}\mathbf{A}\|_{\mathrm{F}}^2, \quad \text{s.t. } \mathbf{Q}'\mathbf{Q} = \boldsymbol{I} \qquad (4)$$

$$= \mathbf{U}\mathbf{V}', \quad \text{where } \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}' = \text{svd}(\mathbf{B}\mathbf{A}').$$

Therefore, the update equation for the filters is

$$\mathbf{H}^{(i+1)} = \frac{1}{\sqrt{K}}\mathbf{U}\mathbf{V}', \text{ where } \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}' = \text{svd}(\mathbf{X}'\mathbf{Z}^{(i+1)}). \quad (5)$$

The CAOL algorithm alternates between the sparse code update (2) and the filter update (5), where memory efficient implementations form $\mathbf{X}'\mathbf{Z}^{(i+1)}$ incrementally to avoid storing all the sparse codes. See [5] for details about initialization, stopping criteria, and a generalization to non-square $\mathbf{H}$.

## 3. INCORPORATING HANDCRAFTED FILTERS

This section proposes CAOL with handcrafted filters (CAOL-HF): a modification to CAOL that constrains the first $P$ filters to be handcrafted ("predefined") and learns the remaining $L = K - P$ filters from training data.

### 3.1. Modification of the Procrustes problem

Using the standard Procrustes variables, we assume the initialization $\mathbf{Q}^{(0)}$ has the scaled handcrafted ("predefined") filters in the first $P$ rows and satisfies the tight frame constraint in (1). To incorporate handcrafted filters, we include an additional constraint in the filter update:

$$\underset{\mathbf{Q}_L}{\operatorname{argmin}} \|\mathbf{B} - \mathbf{Q}\mathbf{A}\|_{\mathrm{F}}^2, \quad \text{s.t. } \mathbf{Q}'\mathbf{Q} = \boldsymbol{I} \text{ and } \mathbf{Q} = \begin{bmatrix} \mathbf{Q}_P \\ \mathbf{Q}_L \end{bmatrix}, \qquad (6)$$

where $\mathbf{Q}_P \in \mathbb{C}^{P\times K}$ contains the handcrafted filters and $\mathbf{Q}_L \in \mathbb{C}^{L\times K}$ contains learned filters.

The tight-frame constraint in (1) forces $\mathbf{Q}$ to be a unitary matrix. Thus, $\boldsymbol{I}_K = \mathbf{Q}'\mathbf{Q} = \mathbf{Q}\mathbf{Q}'$ implies $\mathbf{Q}_P\mathbf{Q}_L' = \mathbf{0}$ (Condition (i)) and $\mathbf{Q}_L\mathbf{Q}_L' = \boldsymbol{I}_L$ (Condition (ii)).

We now introduce a change of variables:

$$\mathbf{W} = \mathbf{Q}_L(\mathbf{Q}^{(0)})' \iff \mathbf{Q}_L = \mathbf{W}\mathbf{Q}^{(0)}. \qquad (7)$$

We use $\mathbf{Q}^{(0)}$ to define $\mathbf{W}$, but any matrix satisfying the tight frame condition and containing the handcrafted filters in the first $P$ rows works. By Condition (i) and definition of $\mathbf{Q}^{(0)}$,

$$\mathbf{W} = [\mathbf{W}_P, \mathbf{W}_L] = [\mathbf{Q}_L\mathbf{Q}_P', \mathbf{Q}_L(\mathbf{Q}_L^{(0)})']$$
$$= [\mathbf{0}_L, \mathbf{Q}_L(\mathbf{Q}_L^{(0)})'].$$

Furthermore, in terms of $\mathbf{W}$, Condition (ii) becomes

$$\boldsymbol{I}_L = \mathbf{Q}_L\mathbf{Q}_L' = (\mathbf{W}\mathbf{Q}^{(0)})(\mathbf{W}\mathbf{Q}^{(0)})'$$
$$= \mathbf{W}\mathbf{W}' = \mathbf{W}_P\mathbf{W}_P' + \mathbf{W}_L\mathbf{W}_L'.$$

Therefore, in terms of our new variable, the two conditions require that $\mathbf{W}_P = \mathbf{0}$ and $\mathbf{W}_L \in \mathbb{C}^{L\times L}$ be unitary.

Applying these two conditions, the minimization in terms of $\mathbf{W}_L$ is the orthogonal Procrustes problem:

$$\hat{\mathbf{W}}_L = \underset{\mathbf{W}_L}{\operatorname{argmin}} \left\|\mathbf{B}_L - \mathbf{W}_L\mathbf{Q}_L^{(0)}\mathbf{A}\right\|_{\mathrm{F}}^2, \quad \text{s.t. } \mathbf{W}_L'\mathbf{W}_L = \boldsymbol{I}_L$$
$$= \mathbf{U}\mathbf{V}', \quad \text{where } \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}' = \text{svd}(\mathbf{B}_L(\mathbf{Q}_L^{(0)}\mathbf{A})'),$$

and $\mathbf{B}_L$ contains the last $L$ rows of $\mathbf{B}$. Substituting for $\mathbf{Q}$, the final expression for the minimizer to (6) is:

$$\hat{\mathbf{Q}}_L = \mathbf{W}_L\mathbf{Q}_L^{(0)} = \mathbf{U}\mathbf{V}'\mathbf{Q}_L^{(0)}, \quad \text{where} \qquad (8)$$
$$\mathbf{U}\boldsymbol{\Sigma}\mathbf{V}' = \text{svd}(\mathbf{B}_L\mathbf{A}'(\mathbf{Q}_L^{(0)})').$$

Finally, substituting for the original CAOL variables, the modified filter update equation is:

$$\mathbf{H}_L^{(i+1)} = \mathbf{H}_L^{(0)}\mathbf{U}\mathbf{V}', \quad \text{where} \qquad (9)$$
$$\mathbf{U}\boldsymbol{\Sigma}\mathbf{V}' = \text{svd}((\mathbf{H}_L^{(0)})'\mathbf{X}'\mathbf{Z}_L^{(i+1)}),$$

where $\mathbf{Z}_L$ contains the last $L$ columns of $\mathbf{Z}$ that correspond to the sparse codes of the learned filters. In this form, one can verify that the learned filters are constrained to be in the range of $\mathbf{H}_L^{(0)}$. Alg. 1 summarizes CAOL-HF.

### 3.2. Computational benefit

One can use an accumulator to store only one sparse code at a time since the $l$th column of $\mathbf{X}'\mathbf{Z}_L^{(i+1)}$ is $\sum_{n=1}^N \mathbf{X}_n'\mathbf{z}_{P+l,n}$, for $l = 1\ldots, L$. Assuming one uses an accumulator in both implementations, CAOL-HF and CAOL have the same mem-

**Algorithm 1** CAOL with handcrafted filters.

1: **procedure** CAOL-HF($\mathbf{H}_P$, tol, $I_{\max}$)
2: $i = 0$
3: $\mathbf{H}_L^{(0)} = (1/\sqrt{K})\text{null}(\mathbf{H}_P')$
4: **while** $i < I_{\max}$ and $\frac{\left\|\mathbf{H}_L^{(i)} - \mathbf{H}_L^{(i-1)}\right\|}{\left\|\mathbf{H}_L^{(i-1)}\right\|} > \text{tol}$ **do**
5: **for** $k = (P+1) : K, n = 1 : N$ **do**
6: $\mathbf{z}_{k,n}^{(i+1)} = \tau.(\mathbf{h}_k^{(i)} \circledast \mathbf{x}_n, \alpha)$ ▷ From (2)
7: **end for**
8: $\mathbf{U}\mathbf{\Sigma}\mathbf{V}' = \text{svd}((\mathbf{H}_L^{(0)})'\mathbf{X}'\mathbf{Z}_L^{(i+1)})$
9: $\mathbf{H}_L^{(i+1)} = \mathbf{H}_L^{(0)}\mathbf{U}\mathbf{V}'$ ▷ From (9)
10: $i = i + 1$
11: **end while**
12: **return** $\mathbf{H}_L^{(i)}$
13: **end procedure**

ory complexity: $O(\min(M, KL))$, where $L = K$ for CAOL and typically $KL \ll M$.

When $K \leq MNL$ (which holds for large datasets), the per-iteration computational cost of CAOL-HF is smaller than the $O(MNK^2)$ required by CAOL. CAOL-HF avoids $O(MNPK)$ operations by not calculating $\mathbf{z}_{k,n}$ for $k \leq P$ (Alg. 1 line 6) and again when evaluating $\mathbf{X}'\mathbf{Z}_L$ (Alg. 1 line 8). Thus, the time complexity of each CAOL-HF BCD iteration is $O(MNLK)$. Section 4.2 empirically examines the number of iterations required to reach convergence.

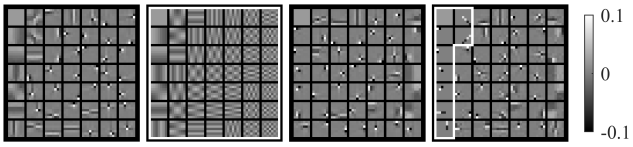
## 4. APPLICATION TO SPARSE-VIEW CT

This section examines the effect of the number of handcrafted filters on training time and sparse-view CT reconstruction quality. All CT images are presented in modified Hounsfield units (HU), where air is 0 HU and water is 1000 HU. Training code is available at [14].

### 4.1. Training setup and results

Training learns $\hat{\mathbf{H}}$ via Alg. 1 from high quality CT images. We used $N = 10$ XCAT phantom $512 \times 512$ slices [15] spaced by five slices (3.125mm) and normalized to [0,1] (see Fig. 1 for example slices). We set $K = 7 \times 7$, $\lambda = 5 \cdot 10^{-4}$ (selected by visually comparing to the filters presented in [6]), a convergence tolerance of $10^{-6}$, and 2000 maximum iterations.

To examine the effect of handcrafted filters, we used two sets of filters. First, we used the 2D DCT, ordered from low to high frequency. We learned filters for $P \in \{0, 1, 3, 6, \ldots, 43, 46, 49\}$, which is equivalent to all filters up to the $i$th anti-diagonal in the usual DCT arrangement. Second, we used extended finite difference (EFD) filters (*e.g.*, [1, -1] and [1, 1, -1, -1]). To initialize, we replaced the first nine DCT filters with our EFD filters and applied the Gram-Schmidt procedure to obtain an orthogonal matrix. We learned filters for $P \in \{0, 1, 5, 9\}$. Fig. 2 shows four example $\mathbf{H}$'s as a grid of filters arranged in column-major order.




(a) DCT, P=0 (b) DCT, P=49 (c) EFD, P=0 (d) EFD, P=9

**Fig. 2**: Example filters for the DCT (a-b) and EFD (c-d) cases. (a) and (c) show the case of all-learned filters ($P = 0$) while (b) and (d) show the case of the maximum number of handcrafted filters ($P = 49$ for DCT and $P = 9$ for EFD). Handcrafted filters are outlined by white borders.

Fig. 3 shows the number of iterations and the time per iteration versus $P$. The time per iteration decreases linearly with $P$ as discussed in Section 3.2. The number of iterations to convergence is less predictable, but the overall trend is that the number of iterations decreases as $P$ increases.

### 4.2. CT reconstruction formulation

Reconstruction recovers a linear attenuation coefficient image $\hat{\mathbf{x}} \in \mathbb{R}^X$ from a post-log measurement $\mathbf{y} \in \mathbb{R}^Y$ [16, 17] by optimizing [5]:

$$\hat{\mathbf{x}} = \underset{\mathbf{x} \geq 0}{\text{argmin}} \frac{1}{2} \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_{\mathbf{W}}^2 + \gamma \min_{\mathbf{Z}} \sum_{k=1}^{K} \frac{1}{2} \|\mathbf{h}_k \circledast \mathbf{x} - \mathbf{z}_k\|_2^2 + \alpha \|\psi \odot \mathbf{z}_k\|_0 . \quad (10)$$

Here, $\odot$ is the Hadamard product, $\mathbf{A} \in \mathbb{R}^{Y \times X}$ is the system matrix that captures CT physics [18]; $\mathbf{W} \in \mathbb{R}^{Y \times Y}$ is a diagonal matrix with $\mathbf{W}_{i,i} = \frac{\rho_i^2}{\rho_i + \sigma^2}$ based on a Poisson-Gaussian model for the pre-log measurements $\rho$ with electronic readout noise variance $\sigma^2$ [16, 17]; $\gamma$ and $\alpha$ are regularization parameters; and $\psi$ is a binary mask that is one only inside the circle inscribing $\mathbf{x}$ [17,19]. To rapidly solve (10) while guaranteeing convergence to a critical point, we applied the block proximal extrapolated gradient method using a majorizer [6].

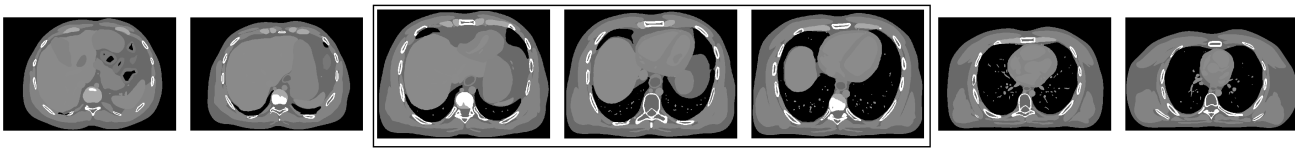


**Fig. 1**: Example training and testing images, arranged by slice order in the phantom (abdominal slices are toward the left, chest slices are toward the right; display window is $[800, 1200]$ HU). Left images: testing images 1 and 2 from the abdominal region. Center box: images from the beginning, middle, and end of the training dataset. Right images: testing images 3 and 4 from the chest region.

### 4.3. Reconstruction setup and results

To simulate sparse-view CT sinograms, we used 4 XCAT phantom $840 \times 840$ slices [15] in 1/mm units with pixel size 0.4883mm, using 888 detectors and 123 views (out of a possible 984 views, yielding a 87.5% reduction in radiation), an incident intensity of $1 \cdot 10^5$, and added noise with $\sigma^2 = 25$. The reconstructed image $\hat{\mathbf{x}}$ is $420 \times 420$ with a pixel size of 0.9766mm. The test images are separated from the training slices by between 18.75 and 53.125mm.

To minimize (10), we used $\gamma = 13 \cdot 10^6$ (suggested in [6]), $\alpha = 5 \cdot 10^{-9}$ (based on a rough grid search), a convergence tolerance of $10^{-6}$, and 5000 maximum iterations. We initialized with the conventional filtered back-projection (FBP) image with a Hamming window. We evaluated the quality of reconstructed images against the true image using root mean square error (RMSE) inside a region of interest defined by $\psi$. Fig. 3 plots the RMSE versus $P$ and Fig. 4 shows example reconstructed images.

For comparison, Fig. 3 reports the RMSE of images reconstructed using a total variation (TV) regularizer with corner rounding. We implemented TV using (10) by replacing $\alpha$ with $\alpha_k$, having $\mathbf{h}_1$ and $\mathbf{h}_2$ take vertical and horizontal differences and satisfy the tight-frame condition (1), and setting $\alpha_k = 0$ for $k \geq 3$. Based on a rough grid search, we chose $\alpha_1 = \alpha_2 = 10^{-7}$ and $\beta = 10^8$.

The DCT filter results (Fig. 3) suggest a trade-off: as $P$ increases, both the iterations to convergence and reconstruction quality tend to decrease (though neither is monotonic). This trend is more noticeable for test images 3 and 4, where the RMSE increases by an average of 5.01 when comparing $P = 0$ to $P = 49$. In comparison, test images 1 and 2 have an average RMSE increase of only 0.85. The EFD filters had lower RMSEs than the DCT filters and, unlike the DCT filters, the RMSE decreases as $P$ increases.

Both the DCT (for small $P$) and EFD filters improve on the TV regularizer for images 3-4 but not for images 1-2. We hypothesize that learned filters led to lower RMSEs for images 3-4 because those images have more high-contrast regions, similar to the majority of our training dataset. If we learn filters on slices similar to test images 1-2, we may outperform TV for these low-contrast images.

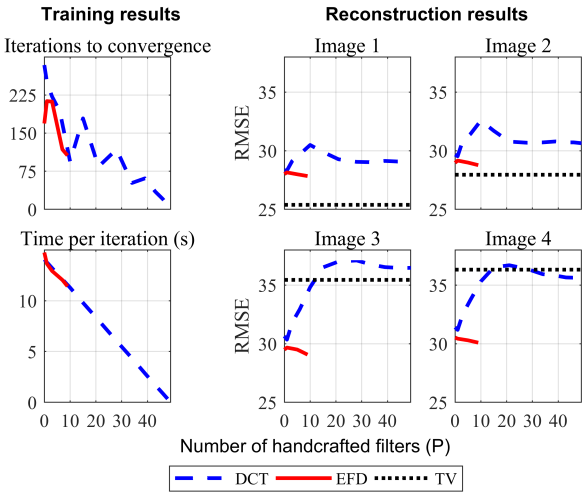



**Fig. 3**: (Left) Number of CAOL-HF iterations required to reach convergence and time per iteration versus the number of handcrafted filters for DCT and EFD filters. (Right) RMSE of the reconstructed CT test images versus the number of handcrafted filters, $P$. TV results are plotted for comparison, but do not vary with $P$.

## 5. CONCLUSION

This paper examined how incorporating handcrafted filters into CAOL affects training time and CT reconstruction quality. Our proposed algorithm, CAOL-HF, has lower per iteration time complexity as the number of handcrafted filters increases, though the overall time complexity is hard to analyze due to the varying number of iterations to convergence. We hypothesize that handcrafting well-designed filters generally leads to fewer iterations, though proving this remains future work. For reconstruction quality, we observed a decrease in quality when handcrafting DCT filters but a slight increase in quality when handcrafting EFD filters that are more appropriate for CT. Future work should consider how to design/learn filters for both high and low contrast CT slices.

Although our experiments are specific to sparse-view CT, the ideas transfer to other signal processing tasks. The presented modification of the Procrustes problem could be used in more domains to understand the trade-off between learning and handcrafting as well as to decrease training time while possibly maintaining or improving reconstruction quality.

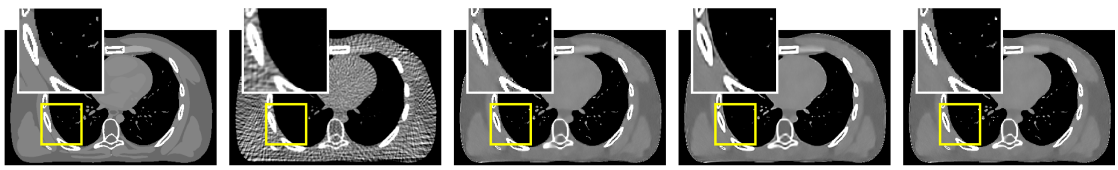



**Fig. 4**: Reconstruction results for test image 3 (display window is $[800, 1200]$ HU).

## 6. REFERENCES


[1] D. J. Brenner and E. J. Hall, "Computed tomography-An increasing source of radiation exposure," *New England journal of Medicine*, vol. 357, no. 22, pp. 2277–84, Nov. 2007. [Online]. Available: https://doi.org/10.1056/NEJMra072149

[2] M. Li, C. Zhang, C. Peng, Y. Guan, P. Xu, M. Sun, and J. Zheng, "Smoothed $\ell_0$ norm regularization for sparse-view X-ray CT reconstruction," *J BioMed Research Intl*, vol. 2016, pp. 1–12, 2016. [Online]. Available: http://doi.org/10.1155/2016/2180457

[3] 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, pp. 5949–67, aug 2011. [Online]. Available: https://doi.org/10.1088/0031-9155/56/18/011

[4] S. G. Mallat, "A theory for multiresolution signal decomposition: the wavelet representation," *IEEE Trans. Patt. Anal. Mach. Int.*, vol. 11, no. 7, pp. 674–89, Jul. 1989. [Online]. Available: https://doi.org/10.1109/34.192463

[5] I. Y. Chun and J. A. Fessler, "Convolutional analysis operator learning: acceleration and convergence," *IEEE Trans. Im. Proc.* (to appear), Oct. 2019. [Online]. Available: https://doi.org/10.1109/TIP.2019.2937734

[6] ——, "Convolutional analysis operator learning: Application to sparse-view CT," in *Proc., IEEE Asilomar Conf. on Signals, Systems, and Comp.*, Pacific Grove, CA, Oct. 2018, pp. 1631–5. [Online]. Available: https://doi.org/10.1109/ACSSC.2018.8645500

[7] B. Sahiner, A. Pezeshk, L. M. Hadjiiski, X. Wang, K. Drukker, K. Cha, R. Summers, and M. L. Giger, "Deep learning in medical imaging and radiation therapy," *Medical Physics*, Nov. 2018. [Online]. Available: https://doi.org/10.1002/mp.13264

[8] I. Y. Chun and J. A. Fessler, "Convolutional dictionary learning: acceleration and convergence," *IEEE Trans. Im. Proc.*, vol. 27, no. 4, pp. 1697–712, Apr. 2018. [Online]. Available: https://doi.org/10.1109/TIP.2017.2761545

[9] ——, "Convergent convolutional dictionary learning using adaptive contrast enhancement (CDL-ACE): Application of CDL to image denoising," in *Proc. Sampling Theory and Appl.*, Tallinn, Estonia, Jul. 2017, pp. 460–464. [Online]. Available: https://doi.org/10.1109/SAMPTA.2017.8024378

[10] P. Bao, W. Xia, K. Yang, J. Zhou, and Y. Zhang, "Sparse-view CT reconstruction via convolutional sparse coding," in *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*, Venice, Italy, Apr. 2019, pp. 1446–1449. [Online]. Available: https://doi.org/10.1109/ISBI.2019.8759260

[11] I. Y. Chun, D. Hong, B. Adcock, and J. A. Fessler, "Convolutional analysis operator learning: Dependence on training data," *IEEE Signal Processing Letters*, vol. 26, no. 8, pp. 1137–1141, Aug. 2019. [Online]. Available: https://doi.org/10.1109/LSP.2019.2921446

[12] X. Zheng, I. Y. Chun, Z. Li, Y. Long, and J. A. Fessler, "Sparse-view X-ray CT reconstruction using $\ell_1$ prior with learned transform," submitted, Feb. 2019. [Online]. Available: http://arxiv.org/abs/1711.00905

[13] P. H. Schonemann, "A generalized solution of the orthogonal procrustes problem," *Psychometrika*, vol. 31, no. 1, pp. 1–10, Mar. 1966.

[14] D. Hong, C. Crockett, and I. Y. Chun, "Convolutional operator learning (for Julia)," [GitHub repository] https://github.com/dahong67/ConvolutionalOperatorLearning.jl/, 2019.

[15] W. P. Segars, G. Sturgeon, S. Mendonca, J. Grimes, and B. M. W. Tsui, "4D XCAT phantom for multimodality imaging research," *Med. Phys.*, vol. 37, no. 9, pp. 4902–15, Aug. 2010. [Online]. Available: https://doi.org/10.1118/1.3480985

[16] I. Y. Chun and T. M. Talavage, "Efficient compressed sensing statistical X-ray/CT reconstruction from fewer measurements," in *Proc. Intl. Mtg. on Fully 3D Image Recon. in Rad. and Nuc. Med*, Lake Tahoe, CA, Jun. 2013, pp. 30–3.

[17] I. Y. Chun, X. Zheng, Y. Long, and J. A. Fessler, "Sparse-view X-ray CT reconstruction using $\ell_1$ regularization with learned sparsifying transform," in *Proc. Intl. Mtg. on Fully 3D Image Recon. in Rad. and Nuc. Med*, Xi'an, China, Jun. 2017, pp. 115–9. [Online]. Available: https://doi.org/10.12059/Fully3D.2017-11-3109002

[18] Y. Long, J. A. Fessler, and J. M. Balter, "3D forward and back-projection for X-ray CT using separable footprints," *IEEE Trans. Med. Imag.*, vol. 29, no. 11, pp. 1839–50, Nov. 2010. [Online]. Available: https://doi.org/10.1109/TMI.2010.2050898

[19] J. A. Fessler and W. L. Rogers, "Spatial resolution properties of penalized-likelihood image reconstruction methods: Space-invariant tomographs," *IEEE Trans. Im. Proc.*, vol. 5, no. 9, pp. 1346–58, Sep. 1996. [Online]. Available: https://doi.org/10.1109/83.535846