Analysis of Deep Convolutional Neural Network for Inverse Problems in Imaging

Kutay Ugurlu

Contents

I Introduction 2

II Theory 2

II-A Shift-invariant Normal Operator 2
II-B Forward Model of X-Ray CT Problem 2
II-C Direct Inversion 3
II-D Iterative Inversion 4

III Proposed Method: FBPConvNet 5

III-A Neural Network Model 5
III-B Experiments 6
III-C Training the Network 6
III-C1 Data 6
III-C2 Training Procedure 6

IV Results and Discussion 7

IV-A Reconstruction Performance Metrics 7
IV-B Sample Reconstructions 7
IV-C Replicated Experiment and Results 9
IV-C1 Testing on Ellipsoid Dataset 10
IV-C2 Testing on SYNAPSE dataset 10
IV-D Conclusion 10

List of Figures

1 Projection slice theorem's illustration [3] 3
2 Block diagrams [1] for (a) unfolded version [4] of iterative shrinkage method [5] with sparsifying transform W (b) and the corresponding block for CNN (c), where L is the Lipschitz constant, x_0 is the initial estimates,b_i is the learned bias and w_i is the trainable weights. 4
3 Modified U-Net Architecture 5
4 Information flow in the proposed framework 6
5 Reconstruction comparisons for different datasets 7
6 Performance with different number of views 8
7 Training and Validation Losses for 20 epochs 9
8 Reconstructions for different number of views 9
9 Reconstructions generated for SYNAPSE dataset for different number of views 11

List of Tables

I Sample size for Datasets 6
II Results for Biomedical Dataset 8
III Results for Experimental Dataset 8
IV Experiment results for training configuration described in Section IV-C.
* Noise SNR is described in Equation (13) 10

I. Introduction

In this term project report, the study *Analysis of Deep Convolutional Neural Network for Inverse Problems in Imaging* by Jin *et al.* [1] is reviewed, the results are replicated and the experiments are conducted using another dataset from a challenge.

The authors of the paper propose a convolutional neural network architecture mainly for the inverse problem of computerized tomography(CT) problems, while they stress that the proposed method is available for all normal-convolutional inverse problems. The study aims to reconstruct the image from lower views by direct inversion followed by a convolutional neural network, more specifically a U-Net [2] based architecture. Starting from the observation that the normal operator H^*H, where H^* is the adjoint operator of H that satisfies the relation $\langle f, H^*g\rangle = \langle Hf, g\rangle$, that appears as a forward model in a set of inverse problems, the authors investigate the relationship between the CNN models and the iterative optimization models utilized in inverse problems.

II. Theory

A. Shift-invariant Normal Operator

To define a shift-invariant normal operator, the authors made the following definitions for the continuous domain:

1) *Isometry:* An isometry T is a linear operator such that $T^*Tf(x) = f(x)$
2) *Multiplication:* A multiplication M_m is a linear operator such that $M_m f(x) = m(x)f(x)$ where $m(x)$ is a continuous and bounded function.
3) *Convolution:* A convolution H_h is a linear operator such that $H_h f = \mathcal{F}^* M_{\hat{h}} \mathcal{F} f$ where $\mathcal{F}$ is the Fourier transform and $\hat{h}$ is the Fourier transform of h.
4) *Reversible Change of Variables:* A reversible change of variables $\Phi_\psi : L_2(\Omega_1) \to L_2(\Omega_2)$ is a linear operator such that $\Phi_\phi f = f(\phi(.))$ for some $\phi : \Omega_2 \to \Omega_1$ and such that its inverse exists. That is, it is a linear operator that represents the same function via a reversible domain transformation, such as Cartesian to Polar coordinate transformation.

Theorem 1. *If an operator is in the form $H = TM_m{\Phi_\phi}^{-1}\mathcal{F}$, then normal operator H^*H represents a convolution operation with $|detJ_\phi|M_{\Phi_\phi|m|^2}$*

where J_ϕ is the Jacobian matrix.

Proof.

$$H^*H = \mathcal{F}^*({\Phi_\phi}^{-1})^* \underbrace{M_m^* \underbrace{T^*T}_{I} M_m}_{M_{|m|^2}} {\Phi_\phi}^{-1}\mathcal{F} \tag{1}$$

$$H^*H = \mathcal{F}^*({\Phi_\phi}^{-1})M_{|m|}^2{\Phi_\phi}^{-1}\mathcal{F} \tag{2}$$

$$(\Phi_\phi^{-1})^* = |detJ_\phi|\Phi_\phi \tag{3}$$

$$H^*H = \mathcal{F}^*|detJ_\phi|M_{\Phi_\phi|m|}^2\mathcal{F} \tag{4}$$

where the definition 3 is used in Equation (4) where the order of the change of variables and multiplications are exchanged with a change of variables in the multiplying kernel. Equation (4) is a multiplication in the Fourier domain and the inversion following it, hence it is a convolution. □

B. Forward Model of X-Ray CT Problem

The main focus of the paper is reconstructing the X-ray CT image from low-view projections. When we consider the forward model as the 2-dimensional X-Ray Transform $R : L_2()\mathbb{R}^2) \to L_2([0, 2\pi) \times \mathbb{R})$, also known as Radon Transform, we can express it using Fourier Slice Theorem as follows:

$$R = T{\Phi_\phi}^{-1}\mathcal{F} \tag{5}$$

where the coordinate transformation operation is defined as conversion from Cartesian to polar coordinates and T is inverse Fourier transform in terms of 2D Fourier transform's radial variable w. The operation expressed by Eqn. 5 takes the Fourier transform of the image, expresses the resultant signal in polar coordinates ($\Phi^{-1}(\theta, r) = (r\cos\theta, r\sin\theta)$) and takes the inverse Fourier transform. This process is equivalent to the operation of the projection slice theorem illustrated in Figure 1. Theorem 1 states that R^*R is a convolution with $\frac{1}{||w||}$ where w is the frequency variable for the 2D Fourier transform and M_m in Theorem 1 is multiplication by identity.

Proof.

$$R = T{\Phi_\phi}^{-1}\mathcal{F} \tag{6}$$

$$R^*R = \mathcal{F}^*|detJ_\phi|\mathcal{F} \quad (4) \tag{7}$$

$$\tag{8}$$

where $detJ_\Phi$ is $\frac{1}{||r||}$ in the polar space domain and $detJ_\Phi(w)$ is $\frac{1}{||w||}$ in polar frequency domain. □

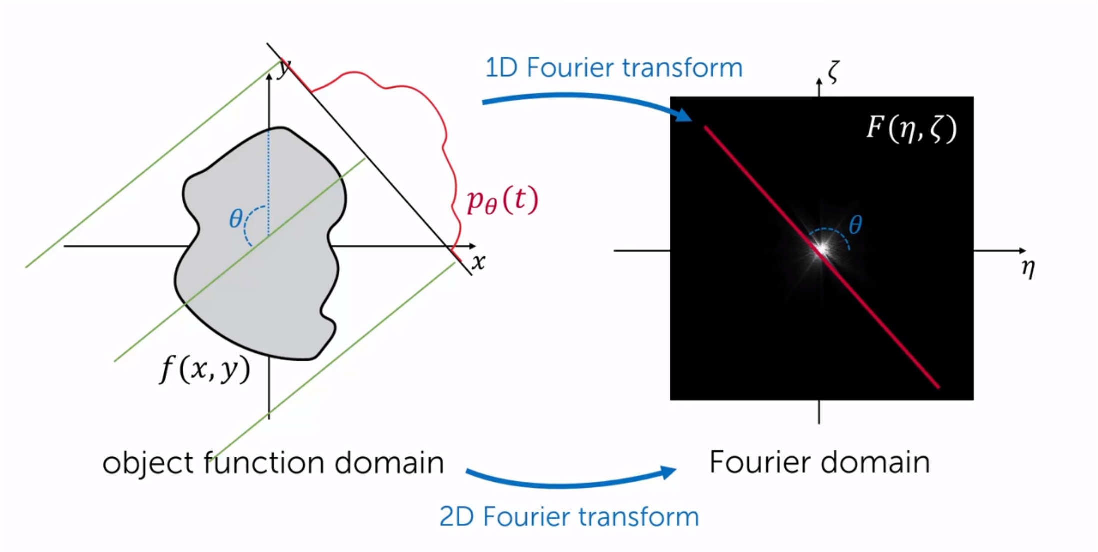


Figure 1: Projection slice theorem's illustration [3]

C. Direct Inversion

One may apply direct inversion methods to solve the inverse problem of tomography, *i.e.*, obtaining the image from measurements. The problem can be formulated as $g = Hf$ where H is a normal convolutional operator. According to [1], the solution can be obtained by direction inversion in two ways as follows:

$$f = W_h H^* g \tag{9}$$

$$f = H^* T M_h T^* g \tag{10}$$

where W_h in Equation (9) is the convolution operator with $\frac{1}{|detJ_\phi|\Phi_\phi|m(w)|^2}$ and M_h is $\frac{1}{|detJ_\phi||m(w)|^2}$. The physical interpretation behind these equations can be considered as follows: The operation in Equation (9) corresponds to the deconvolution in the reconstruction space, whereas Equation (10) corresponds to a filtering operation in the Fourier domain followed by a back-projection if T is Fourier transform, which is called filtered back projection(FBP) where the projections are filtered with $||w||$ in the 2D polar Fourier transform domain.

D. Iterative Inversion

There also exist iterative approaches to solving inverse problems and they are formulated as follows:

$$\underset{a}{\operatorname{argmin}} \, ||y - HWa||_2^2 + \lambda ||a||_1 \tag{11}$$

where $x = Wa$ and a is a sparse representation of x in a domain where x is transformed to by W. Since this formulation does not produce a closed-form solution, the equation is solved iteratively.

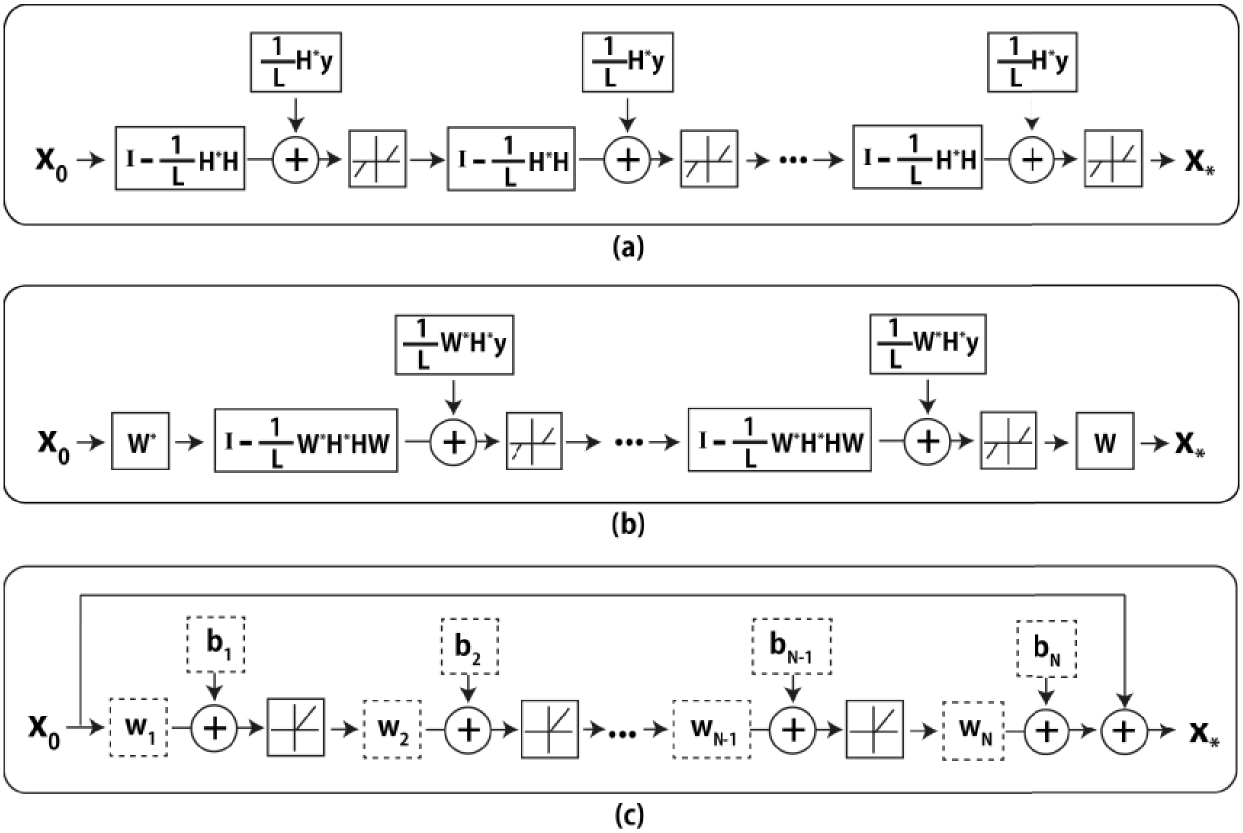


Figure 2: Block diagrams [1] for (a) unfolded version [4] of iterative shrinkage method [5] with sparsifying transform W (b) and the corresponding block for CNN (c), where L is the Lipschitz constant, x_0 is the initial estimates,b_i is the learned bias and w_i is the trainable weights.

In Figure Figure 2, the authors focus on the iterative procedures that utilize sparsifying transforms and investigate the relationship between using the CNN and these methods. In (a) of Figure 2, we observe that in every iteration we encounter the normal operator. In (b), where a sparsifying transformation is used, we again have the normal operator in the unrolled iteration steps. These blocks include filtering by an operator that includes the normal operator along with the sparsifying transformation which usually can be expressed as convolutions, an additive bias per iteration followed by a pointwise nonlinearity.

III. Proposed Method: FBPConvNet

The authors state that the filtering and pointwise linearity operations in the iterative reconstruction methods may suggest that CNNs may be a good fit for the solution of the inverse problems.

The approach simply consists of applying a direct inversion method to the measurements (sinograms in CT case) by Matlab's `iradon` function and feeding the resultant projections to the CNN to train the network to learn the mapping between the back-projected measurements and suitable ground truth images. The authors explain the reason for using a direct inversion method, instead of following an end-to-end neural net training approach between measurements and ground truth images as follows: By projecting the measurements to the image domain, the learning is greatly simplified due to the fact that the network does not have to learn the coordinate transformation between Cartesian and Polar space. Since filtered back projection encapsulates this operation and the physical information about the forward model.

A. Neural Network Model

The design of the proposed network was based on U-Net. The modified architecture is illustrated in Figure 3.

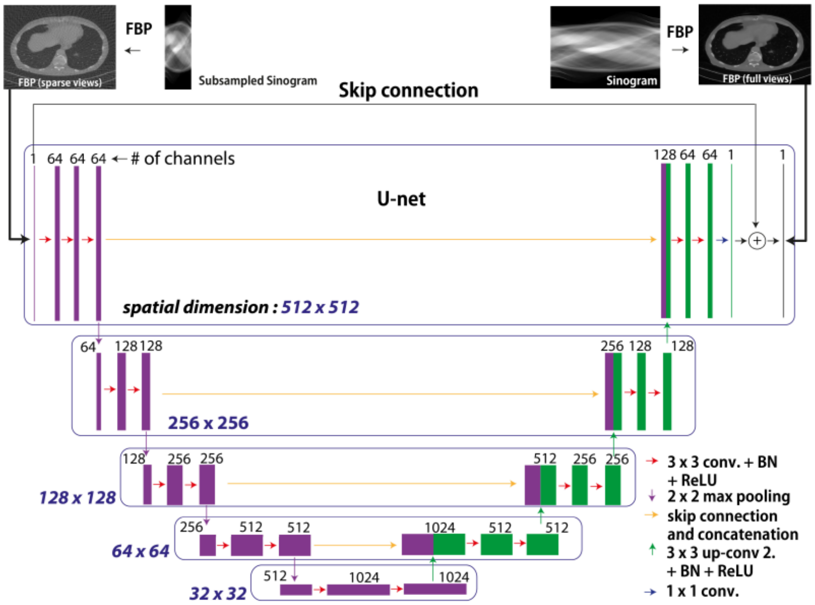


Figure 3: Modified U-Net Architecture

The authors summarized the reasons for basing their architecture on U-Net as follows:

- *Multilevel Decomposition:* The use of pooling kernels with different sizes helps the network effectively invert H^*H. The usage of pooling makes the effective filter sizes in the middle layers larger than that of earlier and later layers. If the normal operator has non-compact support, the different sizes of Fourier transform corresponding to these different sizes provide more levels of decomposition.
- *Multichannel filtering:* The convolutional kernels have the ability to make arbitrary combinations of channels, analogous to wavelet subbands in ISTA or split variables in ADMM, that computes the solution in multiple channels or split variables.
- *Skip Connection for residual learning:* In addition to the original U-Net work, the authors added a skip connection between the input and output, allowing the network to learn the difference between the input and output. These connections have significant improvement in overcoming the gradient vanishing problem.
- *Last Layer:* The last layer of the network is replaced with a 1 × 1 convolution layer since the original U-Net implementation has 2 channels: foreground and background.

B. *Experiments*

Jin *et al.* begin the reconstruction via full-view sinogram from both real and simulated data. Then the reconstruction is performed on the subsampled sinogram. This type of procedure holds an important place in human imaging because lowering the views lowers the radiation dose received by the patient. The full-view filtered back projection data is used as ground truth images, rather than the actual image since the authors point out that this is a more realistic setting in practice since the oracle information will never be available in CT. The results are compared with the filtered back projection and method introduced in [6] that solves Equation (11) via ADMM.

C. *Training the Network*

1) Data: The sample size of the train and test splits of the datasets are given in Table I.

Split Dataset	Training Data	Test Data	Total
Ellipsoid	475	25	500
Biomedical	Scans from subjects except one	Scans from the remaining subject	500
Experimental	327	25	377

Table I: Sample size for Datasets

The ellipsoid dataset comprises ellipses of random intensity, size, and location. For the FBP implementation, MATLAB's `iradon` function was utilized in both the original and replicated study. Sinograms for this dataset are 729 pixels by 1000 views. The biomedical dataset is from the Low-dose Grand challenge competition made by the Mayo clinic and test data is formed by the scan of a subject whose scans were not in the training data. The sinograms are generated by the same approach in the ellipsoid dataset. The experimental dataset comprises 377 sinograms from a real CT dataset collected from an experiment at the Paul Scherrer Institute. Each sinogram is 1493 pixels by 721 views and comes from one z-slice of a single rat brain. The image intensity dynamic range is set between -500 and 500.

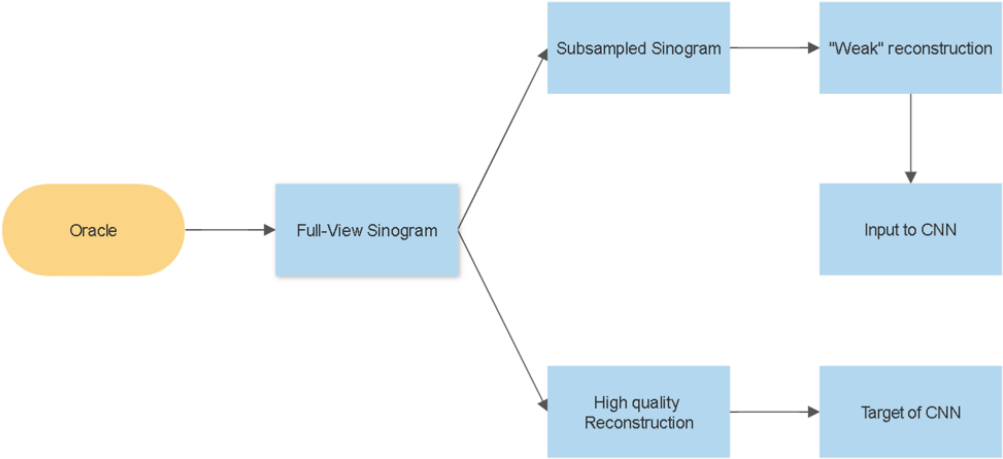


Figure 4: Information flow in the proposed framework

2) Training Procedure: The proposed CNN is trained with pairs of low view and full view FBP images (see Figure 4) in MATLAB R2022(The MathWorks, Inc., Natick, Massachusetts, United States) using MatConvNet [7]. To reduce overfitting, data augmentation is realized by mirroring the images in both vertical and horizontal directions. The training hyperparameters are set as follows:

- Gradient clipping: 10^{-2}.
- Learning rate: [0.01,0.001]
- Loss: MSE (Euclidean Loss)
- Batch size: 1
- Momentum: 0.99
- Optimizer: SGD

The hyperparameters batch size and learning rate are updated in the most recent repository.

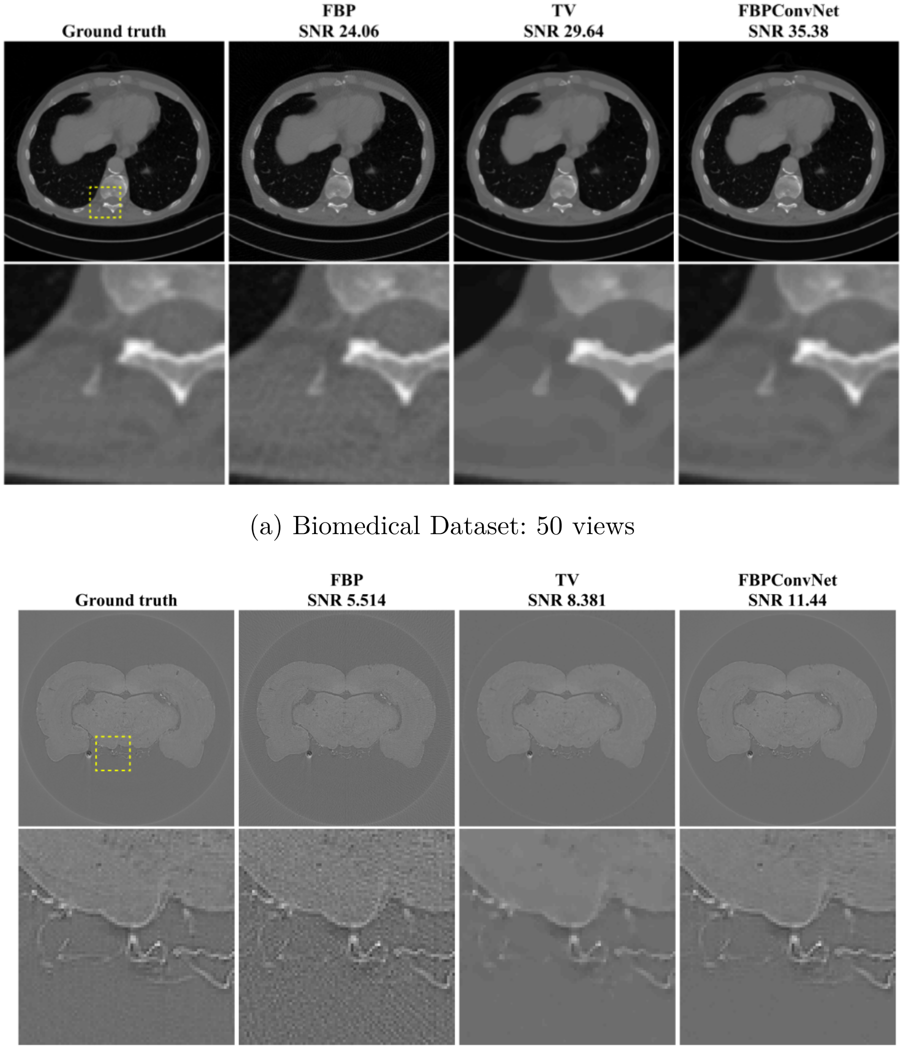
IV. Results and Discussion

A. *Reconstruction Performance Metrics*

The authors use SNR as a quantitative metric. The SNR of the reconstruction $\hat{x}$ with the ground truth image x is given by Equation (12).

$$SNR = \max_{a,b\in\mathbb{R}} \frac{||x||_2}{||x - a\hat{x} + b||_2} \tag{12}$$

B. *Sample Reconstructions*



(a) Biomedical Dataset: 50 views

(b) Experimental Dataset: 145 views

Figure 5: Reconstruction comparisons for different datasets

The results of the experiments showed that the choice of CNN for the set of inverse problems where the forward model is convolution is suitable. In both Figure 5a and Figure 5b the reconstruction performance of the FBPConvNet against back projection can be observed qualitatively. The proposed method yields compelling results in both real and synthetic data. It outperformed the iterative reconstruction method [6]. For a subsampling factor of 7 (143 views), the SNR of the reconstructed image is 35.58dB SNR whereas, for the subsampling factor of 20 (50 views), it was 11.44 dB SNR.

Methods Metrics	FBP	TV [6]	FBPConvNet
SNR for 143 views	24.97	31.92	36.15
SNR for 50 views	13.52	25.2	28.83

Table II: Results for Biomedical Dataset

Methods Metrics	FBP	TV [6]	FBPConvNet
SNR for 143 views	5.38	8.25	11.34
SNR for 50 views	3.29	7.25	8.85

Table III: Results for Experimental Dataset

As we observe in Table III, the performance of the proposed network is lower in the experimental dataset and for lower views. We are able to see this trend in Figure 6 where the network is trained with 143 views and tested with different views. We observe that the SNR of reconstruction drops monotonically with increasing subsampling factors of decreasing the number of views.

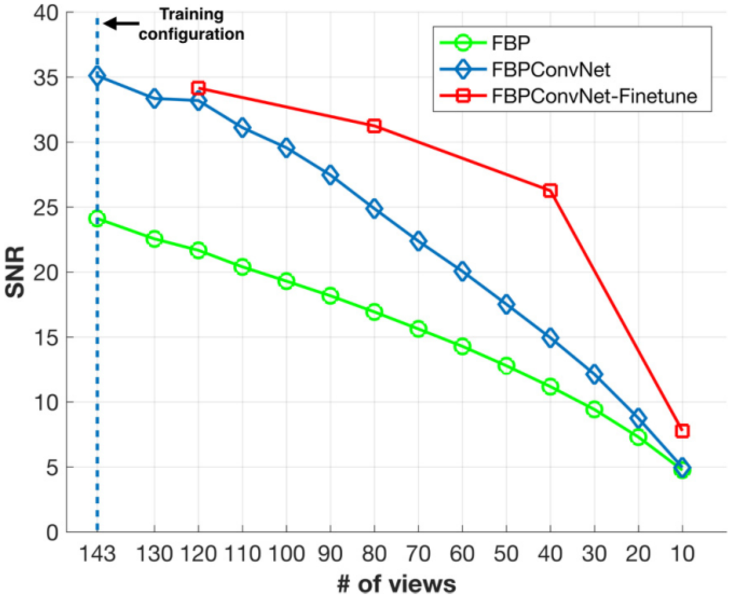


Figure 6: Performance with different number of views

C. Replicated Experiment and Results

Two models are trained for 10 and 20 epochs with accordingly scaled initial learning rates with respect to 100 epochs in the original training setup, using the ellipsoid dataset provided in the original repository of the study, where the input images of the CNN are produced with FBP of the 50 view projections. The better model is used in the evaluation of different testing configurations. It is important to note that the network is undertrained when compared to the training configuration in the original study.

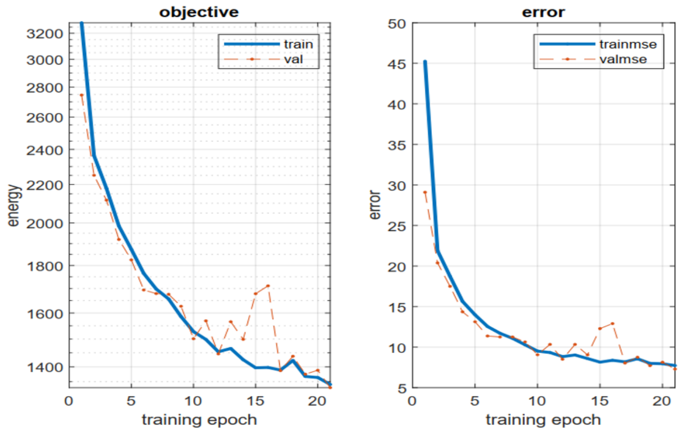
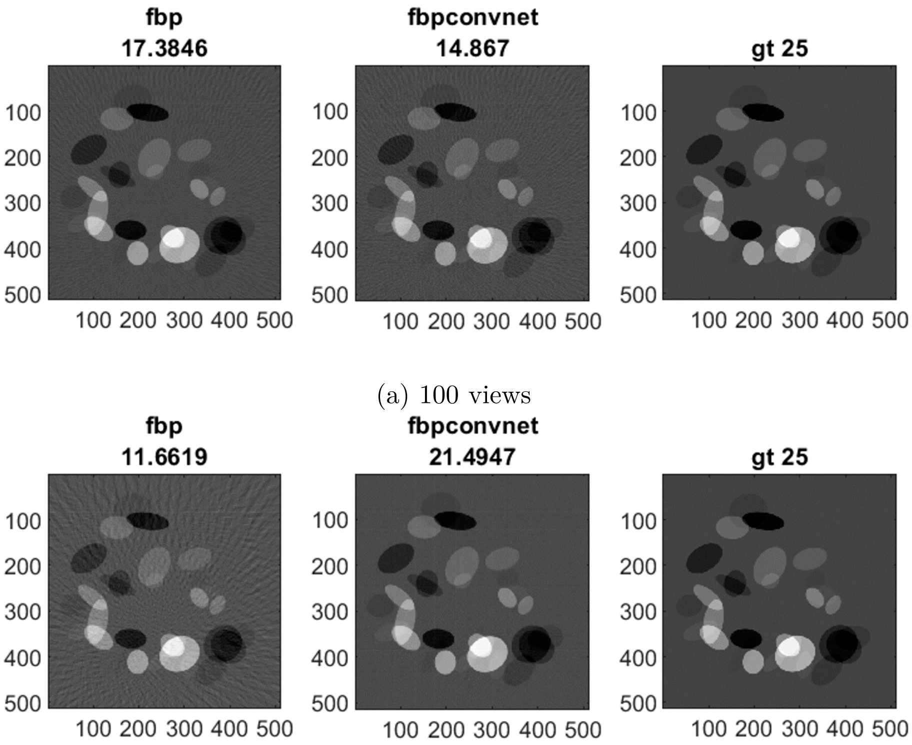


Figure 7: Training and Validation Losses for 20 epochs



(a) 100 views

(b) 50 views

Figure 8: Reconstructions for different number of views

1) Testing on Ellipsoid Dataset: Different sub-datasets are generated with noise and without noise and for a different number of view angles using the ellipsoidal dataset to test the performance of the trained network. For the case illustrated in Figure 8, we observe that FBPConvNet generates outputs with SNRs 14.87 and 21.49 dB respectively.

Experiment Number	Subsampling $\times N$	Noise dB*	FBP SNR(dB)	FBPConvNet SNR(dB)
1	10	-	15.88	11.83
2	10	10	15.88	11.83
3	20	-	10.21	17.48
4	20	10	10.21	17.48
5	25	-	8.67	17.89
6	25	10	8.67	17.89
7	50	-	4.83	8.48
8	50	10	4.83	8.48

Table IV: Experiment results for training configuration described in Section IV-C.
* Noise SNR is described in Equation (13)

$$SNR_{noise} : \frac{P_{signal}}{P_{noise}} = \frac{\sum |signal|^2}{|noise|^2} \tag{13}$$

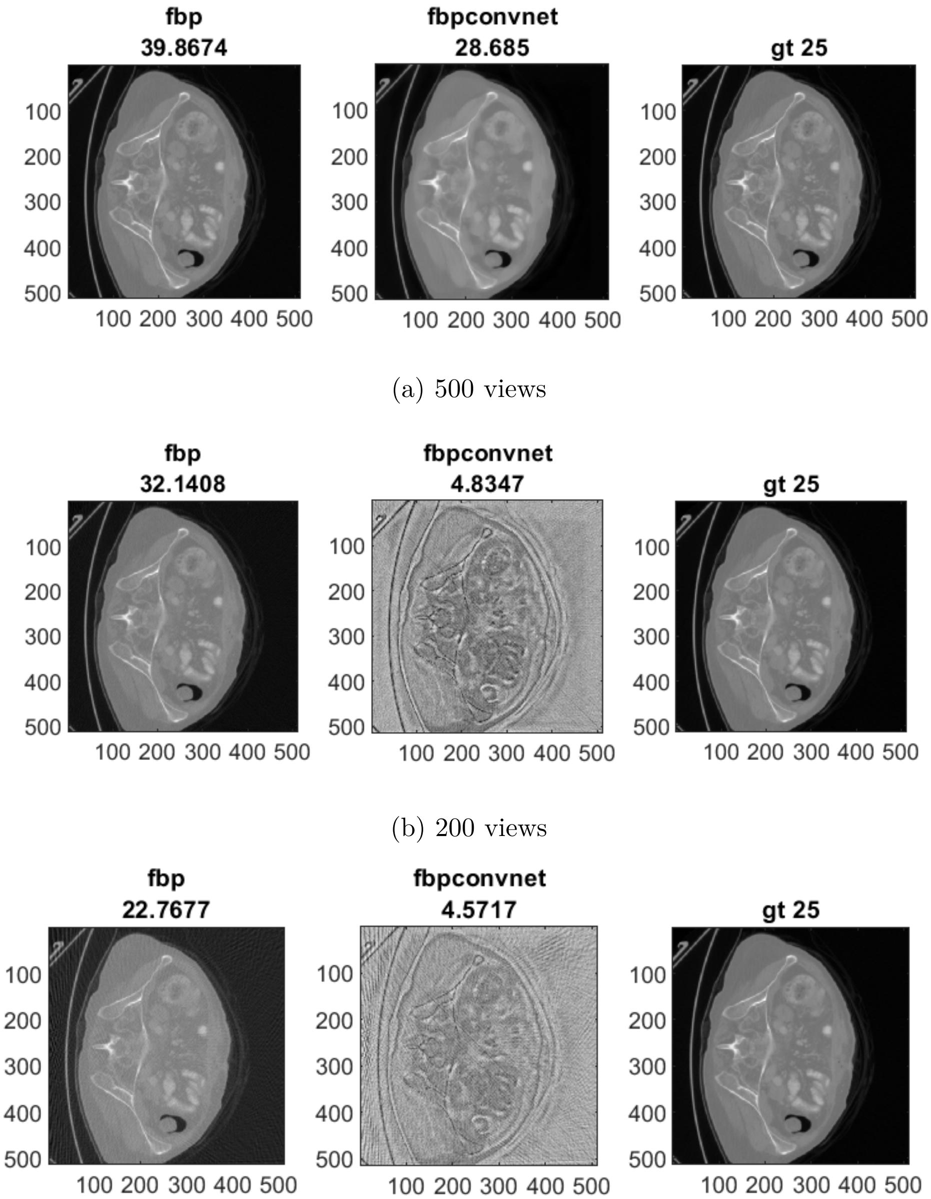
The experiments took 4 seconds per test image on CPU on average and they show that the trained model's performance peaks when the input images' sinogram subsampling matches that of images in the training set. This is why SNR values of reconstructions from FBPConvNet exhibit a non-monotonic behavior, whereas the performance of FBP is decreasing monotonically with decreasing number of projection view angles. Furthermore, it is observed that the added 10 dB SNR noise did not affect the reconstructions in the SNR sense in the given precision, although the network is trained with noise-free images. Compared with the original study, although undertrained, the model showed similar success for 50 views in the ellipsoid dataset.

2) Testing on SYNAPSE dataset: To test the generalization of the trained network on another dataset, the synapse dataset is utilized [8]. This dataset is from MultiAtlas Labelling Challenge and is for segmentation purposes. Since it is open access, images from the dataset are utilized which includes scans that add up to 3779 axial contrast-enhanced CT images.

The reconstructions from Figure 9 show that the trained network does not only learn the convolutional filters that correspond to R^*R for FBP of subsampled sinograms or their inverse. Instead, the model also learned how to remove the artifacts of the ”weak” reconstructions from subsampled sinograms fitting to the prior distribution of the images in the ellipsoid dataset.

D. Conclusion

In this project, the study *Deep Convolutional Neural Network for Inverse Problems in Imaging* is analyzed using the software provided and some of the results are replicated using the provided preprocessed ellipsoid dataset. For the training configuration used to replicate the study, the network improved the images generated by FBP, outperforming it by approximately 9 dB. For other subsampling factors and the same dataset, FBPConvnet output qualitatively acceptable images.



(a) 500 views

(b) 200 views

(c) 100 views

Figure 9: Reconstructions generated for SYNAPSE dataset for different number of views

References

[1] Kyong Hwan Jin et al. "Deep Convolutional Neural Network for Inverse Problems in Imaging". In: *IEEE Transactions on Image Processing* 26.9 (2017), pp. 4509–4522. DOI: 10.1109/TIP.2017.2713099.

[2] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. "U-net: Convolutional networks for biomedical image segmentation". In: *International Conference on Medical image computing and computer-assisted intervention*. Springer. 2015, pp. 234–241.

[3] Thijs Kooi. *Photometric data augmentation in projection radiography*. Apr. 2021. URL: https://medium.com/lunit/photometric-data-augmentation-in-projection-radiography-bed3ae9f55c3.

[4] Karol Gregor and Yann LeCun. "Learning fast approximations of sparse coding". In: *Proceedings of the 27th international conference on international conference on machine learning*. 2010, pp. 399–406.

[5] Ingrid Daubechies, Michel Defrise, and Christine De Mol. "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint". In: *Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences* 57.11 (2004), pp. 1413–1457.

[6] Michael T McCann et al. "Fast 3D reconstruction method for differential phase contrast X-ray CT". In: *Optics express* 24.13 (2016), pp. 14564–14581.

[7] Andrea Vedaldi and Karel Lenc. "Matconvnet: Convolutional neural networks for matlab". In: *Proceedings of the 23rd ACM international conference on Multimedia*. 2015, pp. 689–692.

[8] (Author Name Not Available). "Segmentation Outside the Cranial Vault Challenge". In: (2015). DOI: 10.7303/SYN3193805. URL: https://repo-prod.prod.sagebase.org/repo/v1/doi/locate?id=syn3193805&type=ENTITY.