PROCEEDINGS OF SPIE

SPIEDigitalLibrary.org/conference-proceedings-of-spie

Shallow U-Net deep learning approach for phase retrieval in propagation-based phase-contrast Imaging

Samuel Li, Matthew French, Konstantin Pavlov, Heyang Thomas Li

Samuel Z. Li, Matthew G. French, Konstantin M. Pavlov, Heyang Thomas Li, "Shallow U-Net deep learning approach for phase retrieval in propagation-based phase-contrast Imaging," Proc. SPIE 12242, Developments in X-Ray Tomography XIV, 122421Q (14 October 2022); doi: 10.1117/12.2644579

SPIE.

Event: SPIE Optical Engineering + Applications, 2022, San Diego, California, United States

Shallow U-Net deep learning approach for phase retrieval in propagation-based phase-contrast imaging

Samuel Z. Li[a], Matthew G. French[b], Konstantin M. Pavlov[b c d], and Heyang Thomas Li[b]

[a]Princeton University, Princeton, NJ, USA
[b]University of Canterbury, Christchurch, New Zealand
[c]Monash University, Victoria, Australia
[d]University of New England, Armidale, Australia

ABSTRACT

X-Ray Computed Tomography (CT) has revolutionised modern medical imaging. However, X-Ray CT imaging requires patients to be exposed to radiation, which can increase the risk of cancer. Therefore there exists an aim to reduce radiation doses for CT imaging without sacrificing image accuracy. This research combines phase retrieval with the ShallowU-Net CNN method to achieve the aim. This paper shows that a significant change in existing machine learning neural network algorithms could improve the X-ray phase retrieval in propagation-based phase-contrast imaging. This paper applies deep learning methods, through a variant of the existing U-Net architecture, named ShallowU-Net, to show that it is possible to perform two distance X-ray phase retrieval on composite materials by predicting a portion of the required data. ShallowU-Net is faster in training and in deployment. This method also performs data stretching and pre-processing, to reduce the numerical instability of the U-Net algorithm thereby improving the phase retrieval images.

Keywords: Deep Learning, Phase Retrieval, Shallow U-Net, X-Ray Projection, Phase Contrast

1. INTRODUCTION

Seeing through solid objects such as flesh and bone was made possible by German physicist Wilhelm Conrad Röntgen's discovery and observation of X-rays in 1895.[1] Since then X-ray technology has progressed by leaps and bounds, especially in medical use such as in X-ray computed tomography scans, or CT for short. However despite undeniable advances in medical practice that CT has brought forth, there exists the risk of an increased possibility of cancer due to X-ray radiation exposure.[2] Thus it is clearly worthwhile to investigate ways to reduce radiation doses from CT scans without compromising effectiveness. This is achievable, for instance, by employing X-ray phase-contrast imaging (PCI).[3] However, PCI requires some forms of phase retrieval to produce high-quality reconstructed images. And given the recent surge in the amount of available data, computer processing power, and development of algorithms, it is only natural to turn to machine learning in our search for solutions.[4] The following section will present the physical concepts for X-ray phase-contrast CT imaging,[5] in particular the phase retrieval problem using the Transport of Intensity Equation (TIE),[6] and then describe necessary background for deep learning in order to construct a possible improvement on phase retrieval.

1.1 Phase Retrieval

Since many aspects of physics are concerned with the flow of information, the inverse problem[7] is a common theme in physics. Before considering the inverse problem, consider the forward problem. In the forward problem, there exists a cause and problem is to determine the effect. On the other hand, for the inverse problem, there exists an effect and the goals is to infer the cause. Consider the concrete example of X-ray imaging. The forward problem gives an object and the task is to determine the image for the given object. The associated inverse problem is therefore to determine features of the object, given an image. The difficulty in the inverse problem is the mapping is not necessarily bijective and the inverse problem may be ill-defined.

Further author information: (Send correspondence to Samuel Z. Li)
Samuel Z. Li: E-mail: zechenl@princeton.edu

Developments in X-Ray Tomography XIV, edited by Bert Müller, Ge Wang, Proc. of SPIE Vol. 12242, 122421Q
 · 0277-786X · doi: 10.1117/12.2644579

A problem arises when imaging near transparent, minimally absorbing objects such as soft biological tissue which have little measurable effect on X-ray intensity. However, such objects have measurable effects on X-ray phase so the goal of phase contrast imaging is to visualise the phase shift of an X-ray propagating through an object. The problem of phase retrieval is due to the high frequency of wave field oscillations as X-rays propagate. As a result, existing state of the art detector technology is too slow to detect the phase of an X-ray wave field and one can only measure the time averaged intensity image produced by phase contrast imaging. Since this image is a function of both the phase and intensity of the input wave field, the question becomes: Given phase contrast images of the input wave field, how can one retrieve the phase of the input wave field?

Two Distance Phase Retrieval Formalism

Define a 3D Cartesian coordinate system and let X-rays propagate through an object in the positive z direction. Suppose the X-rays exit the object at plane $z = z_0$, and suppose intensity is measured at planes $z = z_1$ and $z = z_2$, where $z_0 < z_1 < z_2$. Then the intensity at the exit plane, $z = z_0$ is approximated by:

$$I(x, y, z = z_0) \approx I(x, y, z = z_1) - (z_1 - z_0) \cdot \frac{I(x, y, z = z_2) - I(x, y, z = z_1)}{z_2 - z_1} \tag{1}$$

Hence, by the linear finite difference approximation given in Li's thesis,[8] the accumulated phase shift is approximated by (*cf.*[9]):

$$\psi(x, y, z = z_0) \approx -k\nabla_\perp^{-2} \left(\nabla_\perp \cdot \left\{ \frac{1}{I(x, y, z = z_0)} \nabla_\perp \left[\nabla_\perp^{-2} \frac{\partial I(x, y, z = z_0)}{\partial z} \right] \right\} \right) \tag{2}$$

Please note that this equation is derived under some assumptions discussed in detail by Schmalz *et al.*[10] Thus it is evidently possible perform phase retrieval by measuring X-ray intensity at two parallel planes. However, one must make two measurements at every angle to perform phase retrieval for CT, which introduces more instances of X-ray scattering, and higher radiation doses in the case of medical use.

1.2 Machine Learning

Artificial Neural Networks (ANNs) were first introduced in 1943 by Warren S. McCulloch et al.,[11] as a attempt to create a mathematical algorithm that make computations similar to the way biological nervous systems were understood to at the time. Due to the lack of computational power and lack of training data ANNs did not demonstrate their state of the art ability to automate pattern recognition's task until recent decades. With the recent advances in GPU and the availability of large training data sets ANNs have become the driving force behind the recent resurgence in research of artificial intelligence and machine learning.

Put simply ANNs makes a prediction from some input data by passing it through a series of layers containing nodes. The nodes act to transform the input in most cases by applying a non-linear function called the activation function to input data. The output is connected to every node in the successive layer if the network is densely connected. The outputs act as the connections between nodes where the linear sum of the inputs represent the strength of the connection between a node and the previous layer. After the data has been passed through the entire network the connections between each layer are adjusted using a gradient based algorithm called back-propagation[12] that reduces the difference between the predicted result and the ground truth. The network 'learns' by passing through enough data so that the network converges to a distribution of connections such that the function representative of the patterns in the input data. ANNs have been already used in phase retrieval in propagation-based imaging in context of electron imaging.[13]

Convolutional Neural Networks (CNNs) are a popular ANN architecture for visual pattern recognition tasks such as image segmentation, image classificaiton and object detection. Before CNNs, studies of visual pattern recognition were done mostly in biology. The most notably being the discovery made by Huble & Wiesel in 1959 of cells in the visual cortex responsible for light detection in receptive fields.[14] This discovery inspired the predecessor of the CNN which was a ANN called the "recognitron"[15] proposed by Fukushima et al. in 1982. However, the first to lay down the frame work for the modern CNN was LeCun *et al.* was in his paper entitled "Gradient-Based Learning Applied to Document Recognition".[16] In this paper a mutli layered ANN called

LeNet-5[16] was trained with back-propagation[12] so that it could classify hand written digits.[16] Later, in 2012 Krizhevsky *et al.* improved on the previous methods by proposing a version of LeNet-5[16] with a deeper structure called AlexNet[17] that won the ILSVRC-2012 competition and became the state of the art method for visual pattern recognition. During the past decade there has been a resurgence in the use of CNNs for visual pattern recognition and researchers continued to introduce new CNNs architectures that out perform there predecessor, such as VGGNet,[18] GoogleNet,[19] ZFNet,[20‘] and ResNet.[21]

Every CNN architecture such as the LeNet-5[16] architecture consists of nodes which contain a convolutional layer, pooling layer and fully-connected layer. Convolutional layers contain several kernals which are used to produce different feature maps from input data which are representative of high level features such as lines and curves. A feature map is produced by convolving a learned kernel with every location and corresponding region of the input data and then applying a element-wise non-linear activation function to the convolved result. The activation functions introduced by the convolutional layer adds non-linearities into the CNN so that non-linear features can be extracted. The typical activation functions used are either sigmoid, tanh[22] or ReLU.[23] The convolutional layer produces output to the next layer as multiple feature maps corresponding to each kernal in the layer. Each feature value or sometimes referred to as the neuron, is connected to the section of the input data that it was produced from. This is referred to as the neurons receptive field.

The pooling layer of a CNN is responsible for detecting the dominant features which are shift-invariant. This is achieved by applying a pooling operation to reduce the resolution of the feature maps. There are two typical types of pooling operations called max pooling[24] and average pooling.[25] Max pooling produces a matrix containing the maximum values of the feature maps and while does the same but takes the average. Max pooling is most commonly used as it increases the performance of a CNN be reducing the noise in feature maps. While average pooling simply reduces the dimensions of feature maps.

A CNN will typically consist of multiply convolutional and pooling layers before global semantic information is extracted and predictions are made via a fully connected layer. The fully connected layer "flattens" the feature maps into a column vector of neurons which is typically passed through either a ANN called a feed-forward neural network (FNN), a softmax operator[26] or a support vector machine (SVM)[27] to make predictions.

An appropriate function called the loss is used to relate the difference in the predicted result and the ground truth result to the global parameters of the CNN, such as connection between layers and bias. The connections of the CNN are then adjusted via back-propagation[12] to optimize for minimal loss.

Convolutional neural networks were proven to be the state of the art technique for whole-image recognition classification in 2012,[17] however the idea has recently been extended to the classification of structures in an image via deep convolution neural network architectures.[28] In 2014, Long *et al.*[29] were the first to propose a CNN that out performed the state of the art techniques for image segmentation tasks. They developed a method of replacing the fully connected layer of previous CNN architectures with a convolution layer that output a classification map of the input image. Simply put, a classification map is the result of assigning classes to individual pixels to enclose objects or regions of the input image. The network is trained using a per-pixel loss function which is optimised via back-propagation[12] to minimize the number of pixels in the classification map residing outside their mask.

In a typical FCN information about the low-level features of the input image is lost during the extraction of high level features through multiple convolutional and pooling operations. After recent strides made by researcher on the information loss problem a FCN architecture entitled U-Net[30] was proposed. U-Net is a hierarchical structure made of a down sampling layer and a down up sampling layer. The downwards path follows the methods of typical CNN architectures to output a multiple channel feature map encoding the low level features of the input image. The upwards path acts to halve the number of channels in the feature maps by applying a single single kernel of smaller dimension then the kernels in the downwards path. The result is concatenated with it's corresponding cropped feature map from the downwards path.

2. METHOD

2.1 Architecture

In order to show that, given a properly trained neural network model, it is possible to predict $I(x, y, z = z_1)$ using $I(x, y, z = z_2)$ for any angle of measurement, this method takes simulated data of X-ray intensity after passing through an object, measured at 1 mm and 0 mm propagation distances, from 180 different angles (projections), each 1 degree apart (see for details[31]). That is, this method uses 180 instances (projections) of $I(x, y, z = 1\ mm)$ and the corresponding 180 instances of $(x, y, z = 0\ mm)$. Of these 180 pairs, this method will use 170 projections to train a neural network model, ShallowU-Net, and use the remaining 10 pairs to test the predications. To perform the testing, this method will feed each of the 10 measurements taken at 1 mm propagation distance into ShallowU-Net, which gives 10 predictions for the corresponding measurements taken at 0 mm propagation distance. Then, this method will evaluate the predicted measurements against the actual measurements to determine the viability of the predictions, which, if successful, will show that it is possible to predict $I(x, y, z = z_1)$ using $I(x, y, z = z_2)$ for any angle of measurement given a properly trained neural network model.

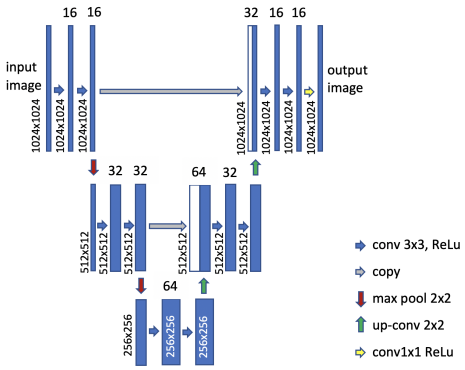


Figure 1: Visual resprentation of ShallowU-Net architcture

This section will introduce the neural network used for this project, a variant of the traditional U-Net architecture called ShallowU-Net. The model takes inputs of shape (1024, 1024, 1), which represents the resolution of the image data used, and the only a singular colour channel is required since only one value is of interest, that is the X-ray intensity. Like traditional U-Net, ShallowU-Net consists of a contracting path and an expansive path. The convolutions, max poolings, and up-convolutions are similarly defined. However, ShallowU-Net includes only two downsampling steps, and two upsampling steps as opposed the four downsampling steps and four upsampling steps found in traditional U-Net. This brings the number of total parameters down to 116,753 which is a significantly smaller amount than a traditional U-Net, hence ShallowU-Net is much faster to train. Even though ShallowU-Net has fewer layers, the model retains accuracy, since the features that X-ray phase retrieval are concerned with, such as the phase contrast edge, are found within a small range of pixels. ShallowU-Net also implements a ReLu activation function in the final layer, as opposed to the traditionally used sigmoid, since ReLu is more suited for pixel wise image regression, instead of the segmentation task that traditional U-Net is used for. In the following sections, it is evident that ShallowU-Net performs very successfully.

2.2 Data

This method will use data created by Baillie *et al.*[31] by simulating X-ray intensity measurements at two distances after passing through a simulated object. The object consists of "a collection of 12 non-overlapping homogeneous spheres of radius" 1×10^{-4} m, "one in each of the eight corners of the bounding cube and one on each face excluding the top and bottom", as shown in figure 2. The voxel size is $1 \times 1 \times 1\ \mu m^3$. The chosen propagation distances are 0 mm and 1 mm and for each propagation distance, 180 projections of X-ray intensity are simulated using the XTRACT software,[32] each differing by a rotation angle of 1 degree, so there are a total of 360 images.

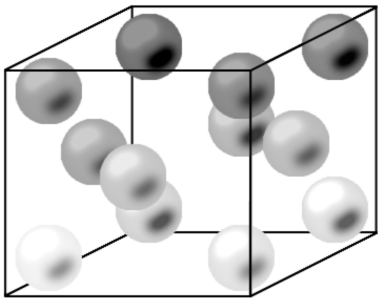


Figure 2: 12 non-overlapping homogeneous spheres

The 180 images measured at 1 mm propagation distance are then loaded into a numpy array of shape (180, 1024, 1024, 1), which will now be referred to as X_data. The images measured at 0 mm propagation distance are loaded into a similar array, called Y_data. For the purpose of testing, slice X_data into two arrays: X_train, which contains the first 170 images and has shape (170, 1024, 1024, 1), and X_test, with shape (10, 1024, 1024), which contains the last 10 images.

2.3 Stretching data

It is evident that X_train has a very small range. The maximum element in X_train is 1.08, and the minimum element is 0.92, which gives a range of 0.16. The results section will show the importance of stretching the data but for now it will suffice to explain the method. To stretch X_train, perform a component wise operation. That is, for each element x in the tensor X_train, $f(x) = (x - 0.9) \times 10$. Thus the stretched data is stored in X_train_stretch, which has a maximum element of value 1.78, and minimum of 0.17, which gives a range of 1.61. This is much larger than the original. The goal is to train the model and make predictions using stretched data and then perform the inverse stretch function on the predictions.

2.4 Training

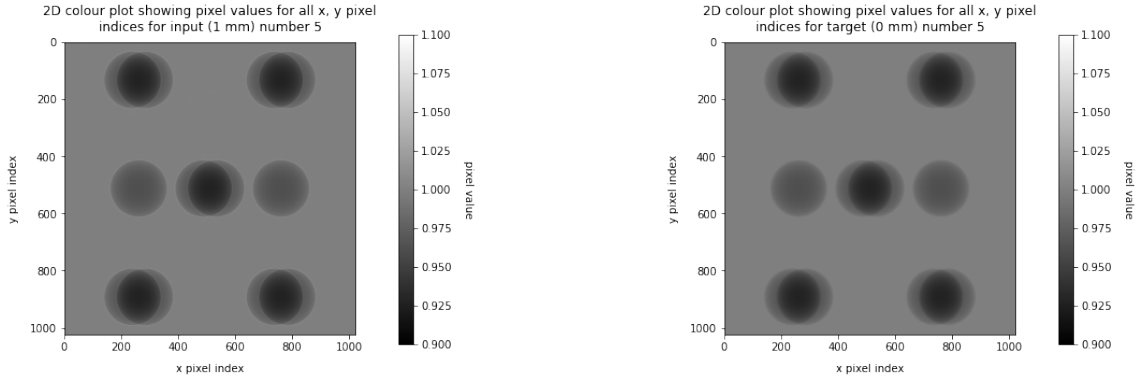
Perhaps the most cumbersome aspect of machine learning, training a neural network is almost always slow due to the large amount of computations and data involved. Here, ShallowU-Net's small number of parameters came as an advantage; A NVIDIA® GeForce® RTX 2070 SUPER GPU is used for the training, which went through 25 epochs for 170 images in a matter of minutes. Chose a 0.1 validations split, which is 153 training pairs, and 17 validation pairs. Set an early stopping callback on validation loss with a patience of 5. This means that if 5 epochs pass and the validations loss does not improve, the training will stop. Also set TensorBoard to track the performance of the model.

3. PREDICTIONS

Having processed the data and trained our model, this section will analyse the predictions that are made. This section will compare colour maps and 1D line plots of predictions make by ShallowU-Net, a traditional U-Net with a VGG-19 backbone, and the homogeneous transport of intensity equation (TIE-Hom).[9]

3.1 Input and output pairs

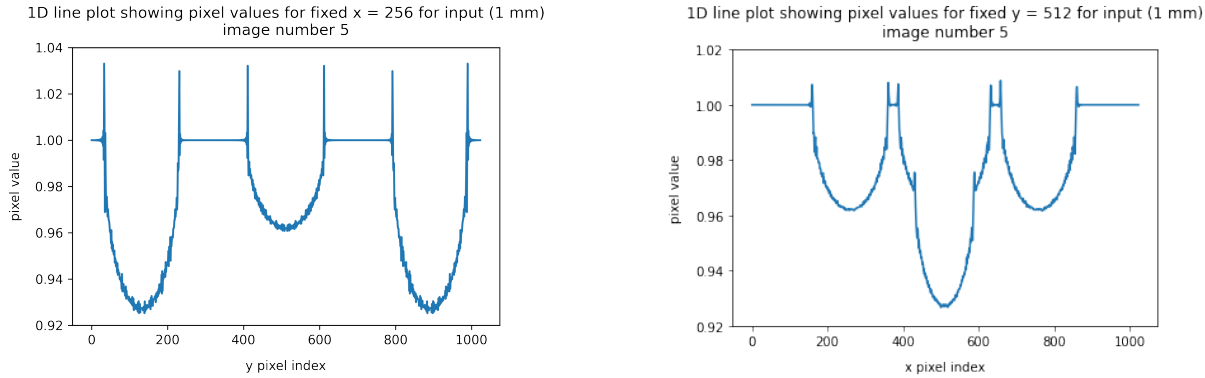
In order to make predictions, input test images measured at 1 mm propagation distance, which the model has not seen before, are loaded into the model. Then we will compare the predictions to the actual corresponding images measured at 0 mm propagation distance. Figure 3a shows a particular image from the 10 images that are inputted into the model for prediction. Note that the colour bar has been fixed for a minimum value of 0.9, and a maximum value of 1.1, allows easy comparisons of similarities and differences between the various images. Figure 4b shows a 1D line plot of Figure 3a fixed at $y = 512$. That is, take Figure 3a as a 1024×1024 array,



(a) Input (b) Output

Figure 3: (a) shows a colour map measured at 1 mm propagation distance, where the horizontal and vertical axis corresponds to 1 μm and (b) shows the corresponding colour map measured at 0 mm propagation distance, where the horizontal and vertical axis corresponds to 1 μm

and take the 513^{th} row of that array. Call this 'line'. For every element in 'line', plot its value against its index to give a 1D line plot. For the sake of comparison the minimum value of the 1D line plot is fixed at 0.91, and the maximum value is fixed at 1.02. Figure 4a shows a 1D line plot of Figure 3a fixed at $x = 256$ in a similar fashion. Note the 'halo' on the periphery of the discs in Figure 3a, and correspondingly, the 'heartbeat shape' spike at the local maxima on Figures 4b and 4a. These show the 'phase contrast edges' characteristic of phase contrast X-ray images at 1 mm propagation distance.

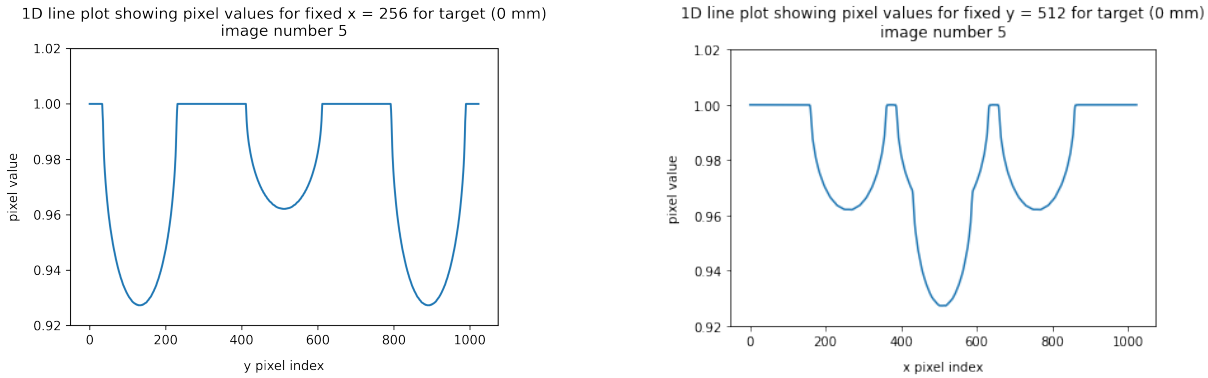


(a) Input (b) Output

Figure 4: (a) shows a 1D line plot, fixed at $x = 256$, for the input image, where the horizontal axis corresponds to 1 μm and (b) shows a 1D line plot, fixed at $y = 512$, for the input image where the horizontal axis corresponds to 1 μm

3.2 Expected output

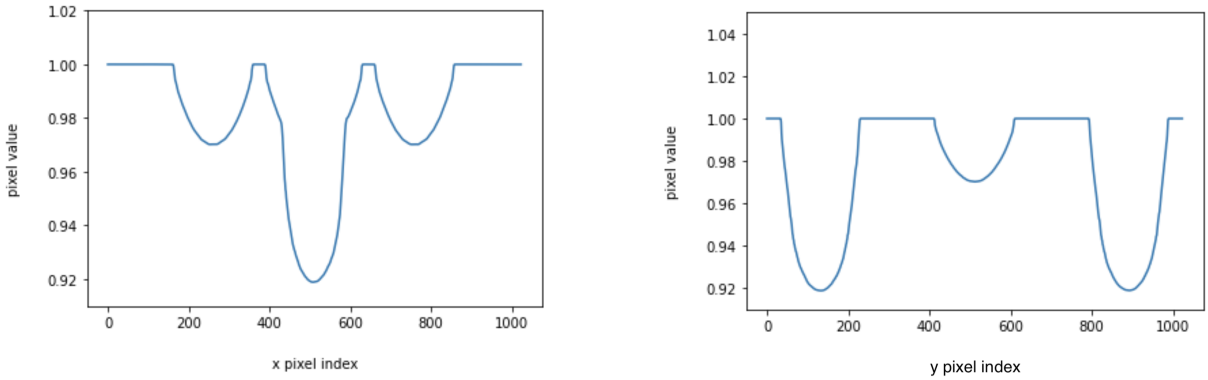
Before considering predictions, first look at the actual corresponding images measured at 0 mm propagation distance. This is the target prediction. Figure 3b shows the image measured at 0 mm corresponding to figure 3a. Likewise, figures 5b and 5a show the 1D line plots for 0 mm measurements, corresponding to figures 4b and 4a, respectively. Again, the minimum and maximum values are fixed to the same values as the figures in the previous section for the sake of comparison. Note the absence of a 'halo' on any periphery of the discs in figure 3b, and correspondingly, the lack of any 'heartbeat shape' spikes on figures 5b and 5a. These show an absence of 'phase contrast edges', which characteristic for phase contrast X-ray images at 0 mm propagation distance. Thus the accuracy of predictions are determined by how closely they resemble the figures shown in this section.



(a) Input (b) Output

Figure 5: (a) shows 1D line plots of slices sampled from the unstretched predicted image. Subplots (a) shows a 1D line plot, fixed at $y = 512$, for image prediction from ShallowU-Net trained on unstretched data, where the horizontal axis corresponds to 1 μm and (b) shows a 1D line plot, fixed at $x = 256$, for image prediction from ShallowU-Net trained on unstretched data, where the horizontal axis corresponds to 1 μm

3.3 U-Net with VGG-19 backbone



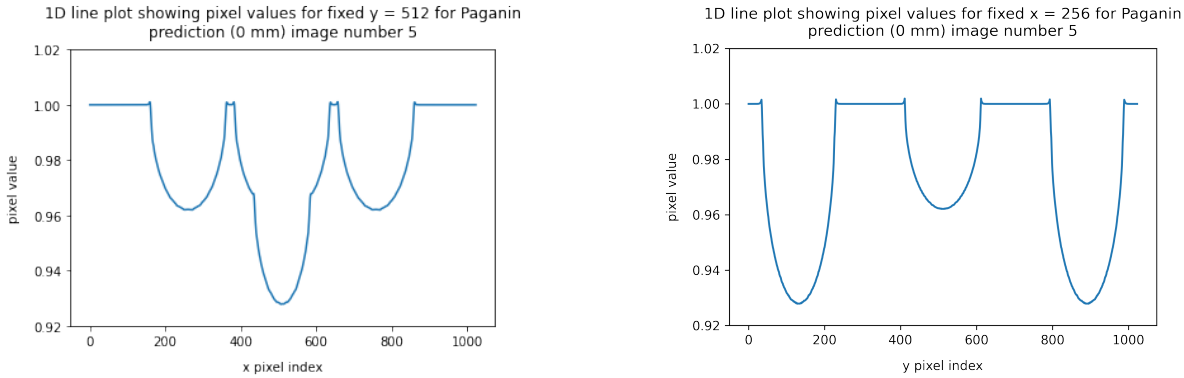
(a) $y = 512$ (b) $x = 256$

Figure 6: shows 1D line plots of slices sampled from the VGG predicted image. Subplot (a) shows a 1D line plot, fixed at $y = 512$, for the prediction by U-Net with VGG-19 backbone trained on stretched data, where the horizontal axis corresponds to 1 μm and (b) shows a 1D line plot, fixed at $x = 256$, for the prediction by U-Net with VGG-19 backbone trained on stretched data, where the horizontal axis corresponds to 1 μm

Before considering predictions made by ShallowU-Net, it is of interest first to examine the performance of an existing architecture. Figure 10c shows the prediction made by U-Net with a VGG-19 backbone trained on stretched data. Fixing the range of the colour bar as done previously, it is evident that the colours are indeed very similar to that of the actual image shown in figure 3b, and a promising absence of 'halos'. However, some interesting gradients not present in the actual image that will become more apparent when looking at the 1D line plots. Figures 6a and 6b show the 1D line plots of the VGG-19 predictions, there exists no 'heartbeat shapes' at the local maxima corresponding to the phase contrast edge. However, comparing to the actual plots in figures 5b and 5a, it's evident that, although the ranges match closely enough, the shape of the curves are rather different, giving rise to the interesting gradient seen in figure 10c.

3.4 Homogeneous transport of intensity equation (TIE-Hom)

Paganin's TIE-Hom algorithm Removes 100 percent of the phase contrast artifacts (halo patterns and spikes) only for single material objects. It requires knowledge of the material or parameter tuning.

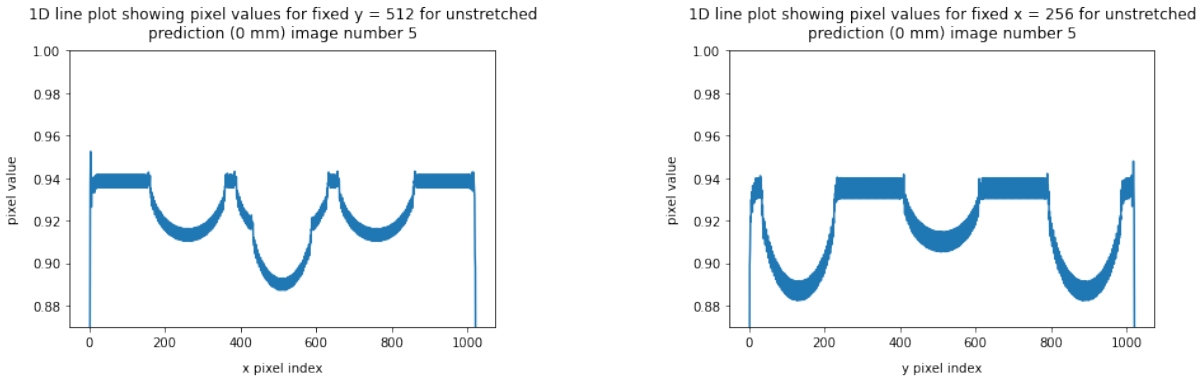


(a) $y = 512$ (b) $x = 256$

Figure 7: (a) shows a 1D line plot, fixed at $y = 512$, for the prediction by TIE-Hom, where the horizontal axis corresponds to 1 μm and (b) shows a 1D line plot, fixed at $x = 256$, for the prediction by TIE-Hom, where the horizontal axis corresponds to 1 μm

3.5 ShallowU-Net trained on unstretched data

Having seen the results from the existing architecture VGG-19, consider the predictions made by ShallowU-Net trained on unstretched data. Figure 10b shows what ShallowU-Net trained on unstretched data predicts an image measured at 0 mm propagation distance to look like, given an image measured at 1 mm. The colour bar range is fixed like before, and by comparing to the actual image shown in figure 3b, it is evident that the predicted image is quite different in colour. The 1D line plots shown in figures 8a and 8b are also very different from the 1D line plot of the actual images shown in figures 5b and 5a, respectively, so different in fact that it's not meaningful to fix the range on these line plots. However, it can be said that the overall shape has been predicted and the phase contrast edge is indeed smaller, albeit somewhat hidden by the noise. This gives motivation to perform the data stretching operation.



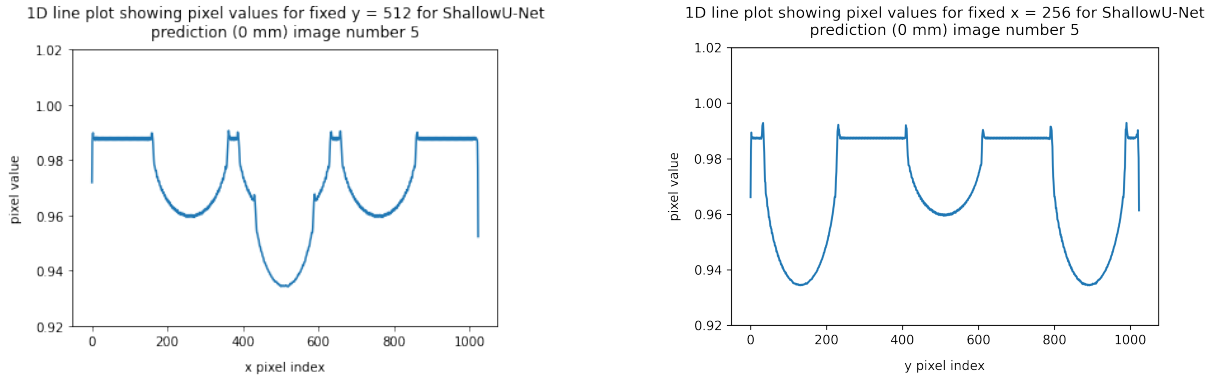
(a) $y = 512$ (b) $x = 256$

Figure 8: shows 1D line plots of slices sampled from the unstretched predicted image. Subplot (a) shows a 1D line plot, fixed at $y = 512$, for image prediction from ShallowU-Net trained on unstretched data, where the horizontal axis corresponds to 1 μm and (b) shows a 1D line plot, fixed at $x = 256$, for image prediction from ShallowU-Net trained on unstretched data, where the horizontal axis corresponds to 1 μm

3.6 ShallowU-Net trained on stretched data

Having seen sub-optimal predictions from ShallowU-Net without data stretching, let us examine the predictions made by ShallowU-Net trained on stretched data. Looking at the predicted image in figure 10a, it is immediately obvious that its colour is very similar to that of figure 3b, which is desired. Also there is little evidence of any

'halos' around the edges of the discs in figure 10a at the phase contrast edge, as desired for an image measured at 0 mm propagation distance. Turning to the the 1D line plots shown in figures 9a and 9b, it is once more apparent that the range reflects that of the plots of the actual image shown in figures 5b and 5a respectively. Also note that the 'heartbeat shapes' at the local maxima are significantly reduced, as desired for an image measured at 0 mm propagation distance. Furthermore, it is of particular interest that the shape of the predicted plot closely resembles the shape of the actual plot. In general, the results from ShallowU-Net trained on stretched data is very desirable.



(a) $y = 512$ (b) $x = 256$

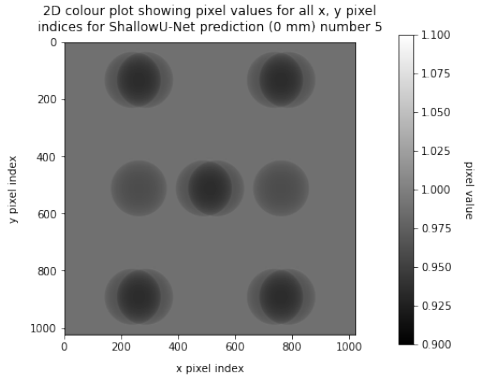
Figure 9: shows 1D line plots of slices sampled from the stretched predicted image. Subplots (a) shows a 1D line plot, fixed at $y = 512$, for image prediction from ShallowU-Net trained on stretched data, where the horizontal axis corresponds to 1 μm and (b) shows a 1D line plot, fixed at $x = 256$, for image prediction from ShallowU-Net trained on stretched data, where the horizontal axis correspond to 1 μm

4. CONCLUSION

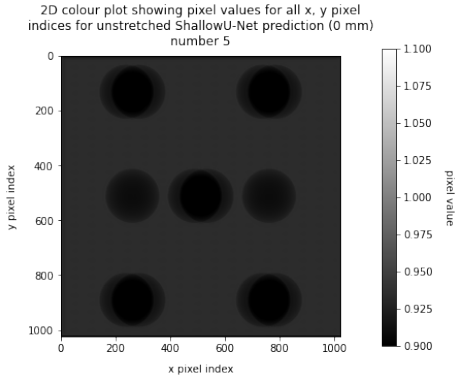
This paper has introduced the workings of phase contrast X-ray CT imaging and the difficulties associated with phase retrieval. Furthermore, this paper has given motivation for pursing this problem through the context of radiation doses associated with medical CT, and formed a postulate for a solution involving the application of deep learning to phase retrieval: to use deep learning to predict X-ray intensity measurements at some propagation distance, given another X-ray intensity measurement at some other propagation distance. In the process of attempting to make such predictions, a custom neural network called ShallowU-Net was built, which has fewer layers, fewer parameters, and is significantly quicker to train than a traditional U-Net as well as many existing architectures such as VGG-19, all while retaining high levels of accuracy. A model such as ShallowU-Net allows a large degree of flexibility with respect to tuning and is highly customisable since the quick training time means that the model can be trained even without a GPU.

Using deep learning methods, the results show that it is indeed possible to make predictions of X-ray intensity measurements at some propagation distance, given another X-ray intensity measurement at some other propagation distance, with significant levels of accuracy. Having justified such a possibility, there could be major advancements in X-ray CT imaging. Say, for the sake of example, that 2 measurements (say at 1 mm and 0 mm propagation distances), at 180 different angles are needed to perform an X-ray CT scan. That's 360 images in total. The findings give rise the possibility of taking taking 180 measurements at 1 mm, 20 measurements at 0 mm, and using that data to train a neural network, say ShallowU-Net. Then ShallowU-Net can predict the remaining 160 images at 0 mm propagation distance. Thus there exists enough data to perform phase retrieval using the two intensity phase retrieval method, but only 200 images were taken instead of the original 360. Assuming X-ray radiation exposure is proportional to the number of images taken, a patient undergoing a CT scan could receive almost half the original amount of radiation required, which provided enormous health benefits such as smaller chances of cancer. Clearly, this is definitely an area of research that deserves further exploration.

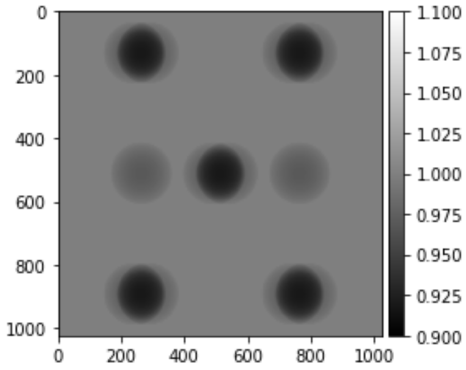
APPENDIX A. PREDICTIONS COLOUR MAPS



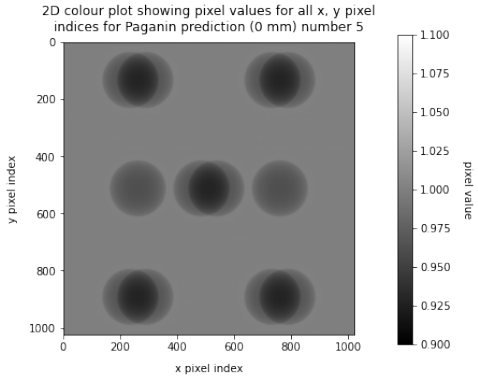
(a) Input



(b) Output



(c) Input



(d) Output

Figure 10: colour maps showing predictions made by four models, where the horizontal and vertical axis corresponds to 1 μm. (a) shows predictions made by ShallowU-Net trained on stretched data. (b) shows predictions made by ShallowU-Net trained on unstretched data. (c) shows predictions made by VGG-19 trained on stretched data. (d) shows predictions made by TIE-Hom.

ACKNOWLEDGMENTS

We acknowledge the funding and support from MARS Bioimaging Ltd. We acknowledge the funding and support from The Council on Science and Technology (CST).

REFERENCES

[1] Roentgen, W. C., "On a new kind of rays," *Nature* **53**, 274–276 (1896).

[2] C., L. E., "Radiation risk from medical imaging," *Mayo Clinic proceedings* (Dec 2010).

[3] Tavakoli Taba, S., Arhatari, B. D., Nesterets, Y. I., Gadomkar, Z., Mayo, S. C., Thompson, D., Fox, J., Kumar, B., Prodanovic, Z., Hausermann, D., Maksimenko, A., Hall, C., Dimmock, M., Pavlov, K. M., Lockie, D., Gity, M., Peele, A., Quiney, H. M., Lewis, S., Gureyev, T. E., and Brennan, P. C., "Propagation-based phase-contrast ct of the breast demonstrates higher quality than conventional absorption-based ct even at lower radiation dose," *Academic Radiology* **28**, e20 – e26 (2021).

[4] Chollet, F., [*Deep Learning with Python*], Manning Publications Co., USA, 1st ed. (2017).

[5] Paganin, D. M. and Pelliccia, D., [*Chapter Two - X-ray phase-contrast imaging: a broad overview of some fundamentals*], vol. 218, 63–158, Elsevier (2021).

[6] Teague, M. R., "Deterministic phase retrieval: a green's function solution," *J. Opt. Soc. Am.* **73**(11), 1434–1441 (1983).

[7] Sabatier, P. C., "Past and future of inverse problems," *Journal of Mathematical Physics* **41**(6), 4082–4124 (2000).

[8] Li, H. T., Kingston, A. M., Myers, G. R., Beeching, L., and Sheppard, A. P., "Linear iterative near-field phase retrieval (lipr) for dual-energy x-ray imaging and material discrimination," *J. Opt. Soc. Am. A* **35**, A30–A39 (Jan 2018).

[9] Paganin, D. and Nugent, K. A., "Noninterferometric phase imaging with partially coherent light," *Phys. Rev. Lett.* **80**(12), 2586–2589 (1998).

[10] Schmalz, J. A., Gureyev, T. E., Paganin, D. M., and Pavlov, K. M., "Phase retrieval using radiation and matter-wave fields: Validity of teague's method for solution of the transport-of-intensity equation," *Physical Review A* **84**(2) (2011).

[11] McCulloch, W. S. and Pitts, W., "A logical calculus of the ideas immanent in nervous activity," *The Bulletin of Mathematical Biophysics* **5**(4), 115–133 (1943).

[12] Hecht-Nielsen, "Theory of the backpropagation neural network," in [*International 1989 Joint Conference on Neural Networks*], 593–605 vol.1 (1989).

[13] Kemp, Z. D. C., "Propagation based phase retrieval of simulated intensity measurements using artificial neural networks," *Journal of Optics* **20**(4), 045606 (2018).

[14] Hubel, D. H. and Wiesel, T. N., "Receptive fields and functional architecture of monkey striate cortex.," *The Journal of physiology* (1968).

[15] Fukushima K., M. S., "Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition.," *Lecture Notes in Biomathematics, Springer, Berlin, Heidelberg.* (1982).

[16] LeCun, Y., Boser, B., Denker, J., Henderson, D., Howard, R., Hubbard, W., and Jackel, L., "Handwritten digit recognition with a back-propagation network," in [*Advances in Neural Information Processing Systems*], Touretzky, D., ed., **2**, Morgan-Kaufmann (1989).

[17] Krizhevsky, A., Sutskever, I., and Hinton, G. E., "Imagenet classification with deep convolutional neural networks," in [*Advances in Neural Information Processing Systems*], Pereira, F., Burges, C. J. C., Bottou, L., and Weinberger, K. Q., eds., **25**, Curran Associates, Inc. (2012).

[18] Simonyan, K. and Zisserman, A., "Very deep convolutional networks for large-scale image recognition," *arXiv 1409.1556* (09 2014).

[19] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S. E., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A., "Going deeper with convolutions," *CoRR* **abs/1409.4842** (2014).

[20] Zeiler, M. D. and Fergus, R., "Visualizing and understanding convolutional networks," *CoRR* **abs/1311.2901** (2013).

[21] He, K., Zhang, X., Ren, S., and Sun, J., "Deep residual learning for image recognition," *CoRR* **abs/1512.03385** (2015).

[22] "Efficient backprop.," *Neural Networks: Tricks of the Trade. Lecture Notes in Computer Science, Springer, Berlin, Heidelberg.* (2012).

[23] Nair, V. and Hinton, G. E., "Rectified linear units improve restricted boltzmann machines.," in [*ICML*], Fürnkranz, J. and Joachims, T., eds., 807–814, Omnipress (2010).

[24] Boureau, Y.-L., Ponce, J., and Lecun, Y., "A theoretical analysis of feature pooling in visual recognition," in [*27TH INTERNATIONAL CONFERENCE ON MACHINE LEARNING, HAIFA, ISRAEL*], (2010).

[25] Wang, T., Wu, D. J., Coates, A., and Ng, A. Y., "End-to-end text recognition with convolutional neural networks," in [*Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)*], 3304–3308 (2012).

[26] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M. S., Berg, A. C., and Li, F., "Imagenet large scale visual recognition challenge," *CoRR* **abs/1409.0575** (2014).

[27] Wijnhoven, R. and de With, P., "Fast training of object detection using stochastic gradient descent," in [*Pattern Recognition, International Conference on*], 424–427, IEEE Computer Society, Los Alamitos, CA, USA (aug 2010).

[28] Simonyan, K. and Zisserman, A., "Very deep convolutional networks for large-scale image recognition," *CoRR* **abs/1409.1556** (2014).
[29] Long, J., Shelhamer, E., and Darrell, T., "Fully convolutional networks for semantic segmentation," *ICLR* (2014).
[30] Ronneberger, O., Fischer, P., and Brox, T., "U-net: Convolutional networks for biomedical image segmentation," *CoRR* **abs/1505.04597** (2015).
[31] Baillie, T., Gureyev, T., Schmalz, J., and Pavlov, K., "Phase-contrast x-ray tomography using teague's method," *Optics Express* **20**, 16913–16925 (07 2012).
[32] Gureyev, T. E., Nesterets, Y., Ternovski, D., Thompson, D., Wilkins, S. W., Stevenson, A. W., Sakellariou, A., and Taylor, J. A., "Toolbox for advanced x-ray image processing," in [*Advances in Computational Methods for X-Ray Optics II*], **8141**, 81410B–81410B–14.