SEMANTIC SEGMENTATION OF X-RAY IMAGES WITH DEEP NEURAL NETWORKS

Nicolai Berthou Thomassen, S123522

The Danish Agricultural Agency and The Technical University of Denmark

ABSTRACT

Semantic segmentation in medical image analysis is crucial for identifying and classifying regions of interest within complex datasets, such as X-ray images. This study explores two distinct approaches to semantic segmentation: a Dense Neural Network (DNN) with three fully connected layers and approximately 550,000 parameters, and a Unet with over 7.7 million parameters, employing a convolutional approach on 256 × 256 pixel patches.

Both models exhibit high training accuracy exceeding 90%, but their performance on unseen data reveals challenges, particularly in addressing correlated datapatches. The DNN shows susceptibility to overfitting, reaching 75% accuracy on unseen data compared to 90% in training, while the Unet achieves 95% accuracy on unseen data compared to an impressive 98.8% in training. Visual interpretations reveal nuances, with DNN capturing trends but struggling with fine details, while Unet, though visually adept, faces challenges in small patch interpretation.

Proposed improvements include implementing validation loss monitoring, refining the DNN architecture to mitigate overfitting, and addressing overlapping training data and convolutional parameters for the Unet. Data augmentation, such as blurring, zooming, and contrast adjustments, is recommended to reduce data correlation. Ethical considerations in medical imaging underscore the need for transparency, fairness, and addressing biases embedded in training data.

In conclusion, the comparative analysis provides insights into the strengths and limitations of the DNN and Unet architectures, serving as a foundation for further exploration and refinement. Recommendations include technical enhancements and a heightened focus on ethical considerations in deploying these models in medical contexts.

Index Terms— Deep Neural Network, Classificaiton, Unet

1. INTRODUCTION

Rapid progress in X-ray physics and better capabilities of X-ray synchrotron sources, has made analyzing tomographic X-ray datasets relevant across scientific, medical, and industrial sectors. Conventional methods of manually segmenting images post data collection is error-prone and time consuming, which in turn makes results be characterized by uncertainties and subjectivity. Automating the segmentation process is therefore imperative to match the speed of acquisition and ensure timely insights in scientific and industrial applications. The challenge presented in this project addresses leveraging deep neural networks for automated segmentation of ptychographic X-ray images, thereby eliminating human involvement and expediting the analysis process significantly.

The training dataset consists of real-world raw X-ray images and corresponding manually labelled datasets. Examples of the data can be seen in Figure 1. It is a visually easy task to solve, and simple methods such as thresholding or KNN clustering will do well in the task.

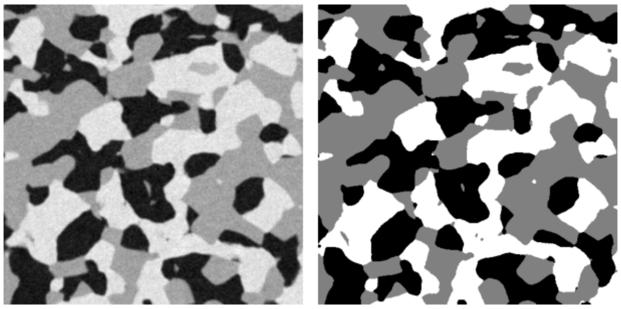

Fig. 1: 501 by 501 image data (Left) and corresponding image label (Right).

The objective of this project is however to develop and train a deep neural network using established architectures commonly deployed in various computer vision tasks. Specifically the scope of the project is to train a minimalistic network utilizing only dense layers and comparing the output-accuarcy to a network using a UNet architecture.

2. DATA SUB-SETTING

The dataset consists of 500 image and corresponding label- sets, which in Deep Learning is a scarce dataset. To enhance the number of individual, although not uncorrelated, training-samples, the data is prepared for the training. To prepare the images for training, the data and labels are preprocessed and transformed into patches of mutable image shape. Augmentations such as random horizontal and vertical flips as well as random rotations are applied to the patches, effectively strengthening the models generalisation capabilities. The augmentations and preparations are created using the Torch dataset object.

The dataset is split into training and testing sets using random split, with the testing set comprising 20% of the overall data. A subset is furthermore held out for validation purposes. Finally, a DataLoader instance is created for both the training and testing sets, which serves a facilitating role in the training and evaluation of the neural network model for semantic segmentation. The routine I created provides a comprehensive foundation for handling the image data preprocessing, and will be used in both versions of the semantic segmentation DNNs described below. Examples of the training patches can be seen in Figure 2.

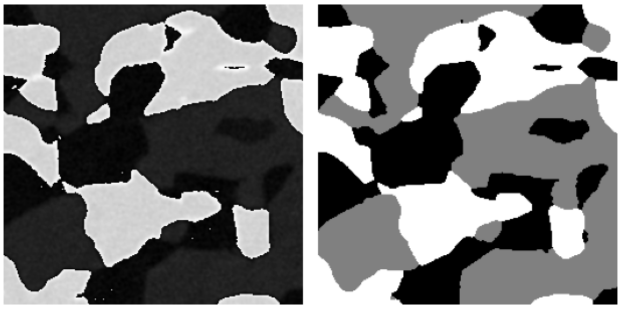

Fig. 2: Example of a 32 by 32 training data patch (Left) and corresponding training data label (Right).

3. SIMPLE LINEAR MODEL

The most primitive model that is used in the process of semantic segmentation is a DNN comprised of three fully connected layers, a visual representation can be found in Figure 3. In this version the input layer takes a flattened version of the $\mathbf{n} \times \mathbf{n}$ patch and densely connects 128 nodes. Hereafter a batch-normalisation is applied and finally a rectified linear unit activation function. The first hidden layer forwards the 128 nodes and produces another 128 nodes. Similar to the input layer, this also has batch-normalisation applied and rectified linear unit applied as activation. Finally the 128 nodes from the first hidden layer is parsed onto the second hidden layer which densely connects to the corresponding $\mathbf{n} \times \mathbf{n}$ output. The output is reshaped to the original image view for training. Cross entropy is used as loss function as the problem is multiclass and Adam optimization is used with $L2$ regularisation. The inputs have had advanced parameter-initialisation applied to them, thus Xavier weight initialisation and constant bias initialisation is used for both of the dense layers. For training patches of shape $\mathbf{32} \times \mathbf{32}$ we achieve $\approx \mathbf{90}\%$ accuracy, see Figure 4, but it is prone to over-fitting as the patches stem from the same dataset.

To visualize and measure how well it actually performs, a full validation image is predicted, see Figure 5. It is clear, that general trends are picked up, but the immediate visual interpretation is that the dark class and white class are overrepresented. The actual accuracy is measured to 74.9%

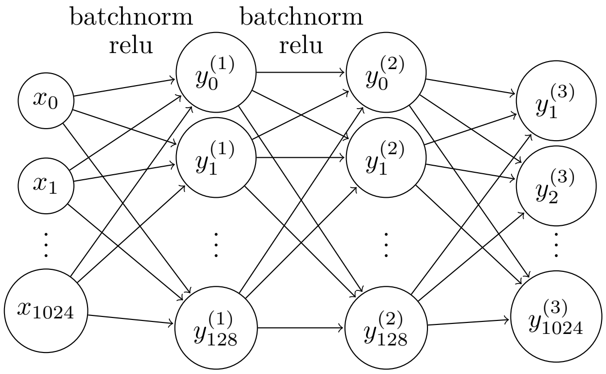

Fig. 3: Network graph of the dense Neural Network with 128 nodes in first and second hidden layer.

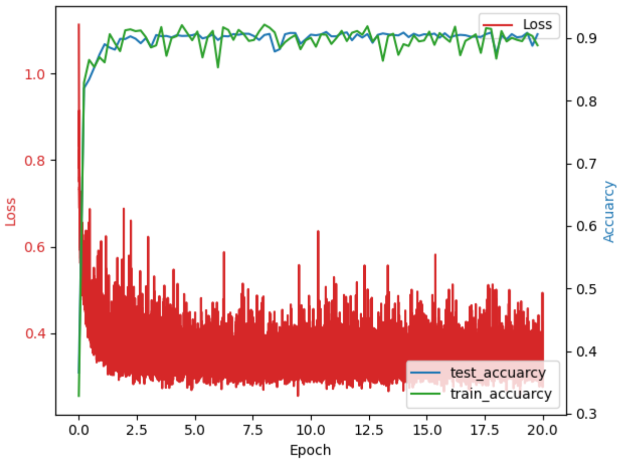

Fig. 4: 32 by 32 patch used in training the simple network yielding an accuarcy of around 90%.

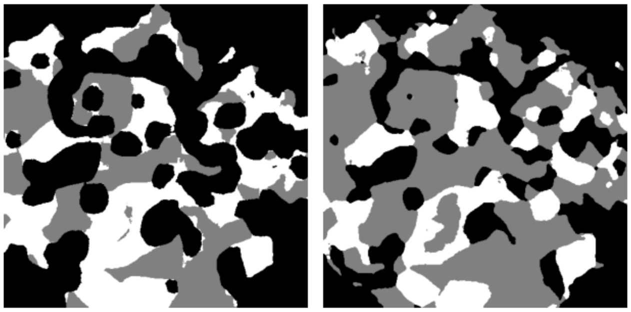

Fig. 5: Classification of entire validation image reached accuarcy of 74.9% (left preediction, right label)

4. COMPLEX UNET MODEL

An intuitive approach to the problem of classifying said x-ray images is the Unet architecture. The Unet has an established reputation for achieving high accuracy in semantic segmentation tasks. The Unet I have developed takes in the 1-channel data in patches of 256×256 pixels. These patches are put through a convolution block that creates 32 feature-maps using a stride of 1 and a 3 by 3 kernel. The output is then normalized with regular batch-normalization and fed to a Rectified Linear unit activation function. This process is repeated, on the 32-channel-output producing another 32 feature-maps, then a batch-normalization and finally ReLU. The product is then stored for the decoder and fed to an encoding block which performs max-pooling using a stride of 2 and a kernel of 2 by 2, which effectively reduces the shape of the product to half i.e 128×128 pixels. This entire process is repeated on the encoded product, except that the amount of feature-maps created is doubled with each iteration. The Unet has a minimum shape of 16×16 pixels and 512 feature-maps in the bottom of the U.

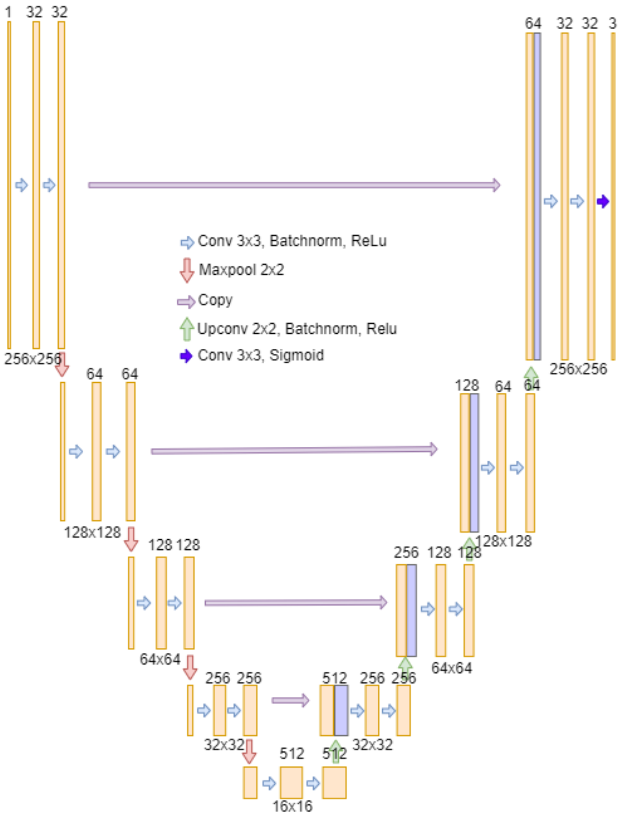

Fig. 6: The architecture of the Unet model in the project.

In the bottom of the U, the encoding happens once again, but in the very end a decoding step takes place. The decoder effectively doubles the amount of pixels in each direction and halves the amount of feature-maps. For the bottom of the U with input $512 \times 16 \times 16$ the output of the decoding is $256 \times 32 \times 32$. This product is then normalized with regular batch-normalization and fed through a ReLu-activation function.

The product is then concatenated with the output from the encoding layer that has the same pixel-dimension, which in turn doubles the amount of feature-maps. The output is then pushed through the convolution block as in the encoding step. This is done repeatedly until the shape is the same as the original image. The final product is parsed through a convolution layer with a 1×1 kernel, producing 3 feature-maps, which finally has a Sigmoid-activation function applied to it. See Figure 6

To make the model generalize better, the output has a dropout of 0.2 is introduced.

The output of the Unet is trained using a Cross-Entropy-loss-function, and Adam optimizer with l2 regularization introduced. The Unet trains well, as seen in Figure 7, reaching an accuarcy of 98.8 and a test accuarcy of 98.9. While the scores are fairly high, the actual accuracy of the model is lower, as much of the training data is overlapping due to the nature of the original images. In theory only one uncorrelated sample can be produced per original image when the patch size is 256 × 256. This means that the high training accuracy is, to some extent, caused by the fact, that the model already has seen the data.

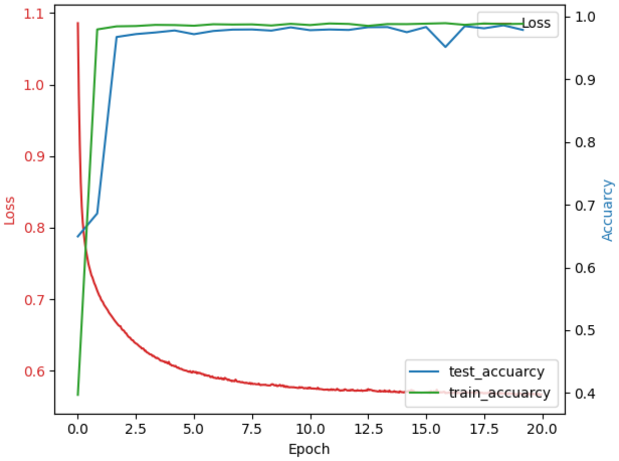

Fig. 7: Training of the Unet.

Looking at the classification of an entire image, Figure 8, it is clear, that the Unet-architecture is able to pick up the small nuances in the image. The small gray patch, south-west of the center of the image, is however a big blob of random pixels. This serves to tell, that the model is not able to effectively understand the nature of a segment, which we with human eyes are capable of. Looking at the image data for the corresponding classification, Figure 9, we see that the gray patch is a little lighter, than the other gray zones, but not light enough for us to have issues with identifying the coherent class.

5. DISCUSSION

In medical image analysis, the task of semantic segmentation plays an important role in identifying and classifying regions of interest within complex datasets, such as X-ray images. In this project two distinct approaches to semantic segmentation, a Dense Neural Network (DNN) and a Unet have been trialled.

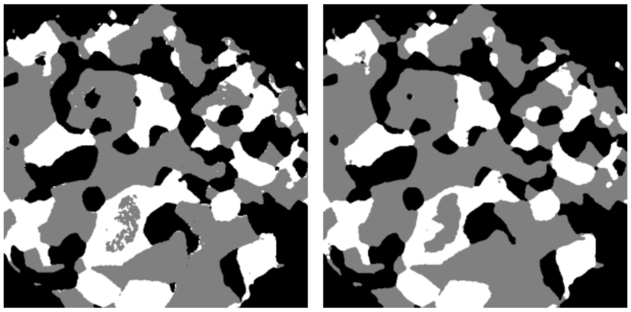

Fig. 8: Semantic segmentation of an entire image reaching an accuracy of 95%.

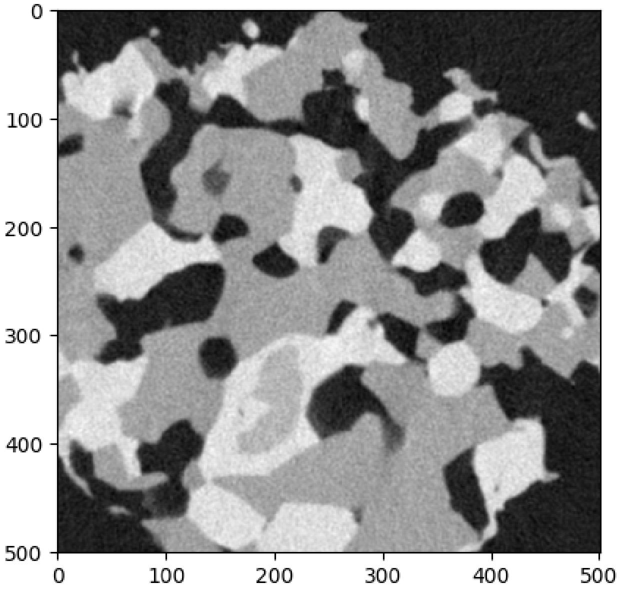

Fig. 9: Corresponding image data

The Dense Neural Network (DNN) is comprised by three fully connected layers and has approximately 550.000 parameters. Each layer contributes to the transformation of a flattened **32** × **32** patch into a segmented output with some success. The Unet having more than 7.7M parameters, employs a convolutional approach by processing data in the format of 256 × 256 pixel patches. The Unet's design contributes to its robust segmentation performance.

The performance of both models, showcases a caveat originating from the correlated data-patches. While the accuarcy of both models exceed 90% in training, both models reach lower accuracies when used on unseen data. The dense net, which reached 90% accuracy in training reaches a mere 75% on the unseen data which underlines, that the model has overfittet the training-data. As for the Unet, an impressive 98.8% training accuracy was reached, but only 95% on the unseen data.

Looking forward, potential improvements and modifications for both models emerge, the immediate suggestion is to implement validation-loss for the training routine. This introduction alone, would assist in understanding when the training of the model starts overfitting, which again leads to a way of detecting when to do a cut-off in the number of epochs. Other suggestions include refining the Dense Neural Network architecture to mitigate overfitting, examples for this could be more hidden layers and more neurons per layer. For the Unet, addressing the issue of overlapping training data as well as the depth and the number of feature-maps produced in each convolution-step could be pivotal. An idea that applies to both models is to introduce more data-augmentation, such as blurring, zooming, cut-out and contrast-adjustments to remove the correlation between each data-sample, obviously a larger dataset is a great as well. The models overfit when the data is correlated or scarce, 500 uncorrelated samples are simply not enough. The models recognise the data it uses per epoch. This is very pronounced in the dense-training, where I generate many more training samples as the small patch-size allows for this, the amount of correlated data fed to the model is thus much larger than that of the Unet. The Unet on the other hand has been fed with overlapping patches to generate enough data for training, which again causes another challenge.

In the context of medical imaging, ethical considerations come to the fore. The discussion emphasizes the need for transparency and fairness in deploying these models, urging the community to carefully consider the impact of model performance on patient outcomes and the potential biases that might be embedded in the training data.

In conclusion, the comparative analysis of the Dense Neural Network and Unet architectures provides a nuanced understanding of their strengths, limitations, and potential paths for improvement. This discussion serves as a stepping stone for further exploration.

6. CONCLUSION

The exploration of semantic segmentation in medical image analysis using Dense Neural Network (DNN) and Unet architectures has shed light on their respective strengths, limitations, and avenues for improvement. The DNN, with its three fully connected layers, demonstrated a commendable performance, albeit exhibiting susceptibility to overfitting. The Unet, boasting a convolutional approach and a more complex architecture, showcased robust segmentation capabilities but also faced challenges in dealing with correlated data.

The performance evaluation revealed that both models achieved high accuracy during training, surpassing 90%, but encountered a significant drop when applied to unseen data. Overfitting, particularly evident in the DNN, underscores the necessity for incorporating techniques such as validation loss monitoring to detect and mitigate overfitting early in the training process.

Future enhancements for both models have been proposed, ranging from refining the DNN architecture by introducing more hidden layers and neurons to addressing issues of overlapping training data and adjusting the depth and feature-maps in each convolution step for the Unet. Additionally, the importance of data augmentation, including techniques such as blurring, zooming, cut-out, and contrast adjustments, is highlighted to mitigate the impact of correlated data and to expand the dataset. The scarcity of data is identified as a significant challenge, emphasizing the need for a larger, diverse dataset in the context of medical imaging.

It is further emphasizing that ethical dimensions of deploying these models in medical contexts is imperative. Transparency, fairness, and a careful examination of potential biases within training data are advocated to ensure responsible use and to minimize any unintended consequences on patient outcomes.

In essence, this comparative analysis provides a nuanced understanding of the DNN and Unet architectures, offering a foundation for further exploration and refinement.

7. REFERENCES

[1] D. S. Bhattiprolu, “Binary semantic segmentation using u-net dataset,” https://github.com/bnsreenu/python_for_image_processing_APEER/blob/master/tutorial118_binary_semantic_segmentation_using_unet.ipynb [Accessed: (2023/12/21)].

[2] S. Segura Lucas, “Segmentation of images of the microstructure of solid oxide cells, segmentering a billeder af mikrostrukturen af fastoxidceller,” 2023.

[3] N. B. Thomassen, “Ipynb file for project,” 20223, https://github.com/NicolaiThomassen/

Semantic-segmentation-of-X-ray-Images-with-Deep-Neural-Networks/
tree/main [Accessed: (2023/12/21)].