UPPSALA
UNIVERSITET

Development and Deployment of a Deep Learning Model for Segmentation of Liver Scans

Project for Antaros Medical AB

Authors: Dag Lindgren, Lowe Lundin, Andreas Wallin
Supervisors: Carl Sjöberg, Camilla Englund, Taro Lagner

Project in Computational Science: Report

February 2020

PROJECT REPORT

Abstract

The aim of the project was to develop and deploy deep learning models and a pipeline, aiding the segmentation of liver MRI-scans by creating automated suggestions. To be useful, the models should generalize well between datasets. This was done using deep learning and the U-Net architecture, building upon the work described in Langner et al[1]. The networks achieved a Dice score of 0.971 for the volume segmentation and a Dice score of 0.933 for the fat-fraction segmentation. The results suggest a strong ability to generalise between datasets produced by different MRI-devices and imaging protocols, if a few patient scans are manually segmented and moved to the training set. For the volume segmentations, just moving one patient scan improves the Dice score to 0.957, from 0.493 when no patient scans were moved. For the fat-fraction segmentations, moving one patient scans improves the Dice score to 0.876, from 0.342, when no patient scans were moved.

1 Introduction

In drug development it is necessary to quantify treatment effects to support decision making. Antaros Medical is a strategic development partner to the pharmaceutical industry that uses medical imaging techniques to help in this process. An important and sometimes time consuming part of image analysis is to create segmentations in the acquired images and it is therefore valuable to facilitate this procedure by automation.

1.1 Project Aim

The aim of the project is to build a deep learning pipeline and train two networks for liver segmentation and calculation of liver volume and fat-fraction from Magnetic Resonance Imaging (MRI) scans. The envisioned end-goal product is one that can be fed with scans from new studies, with different devices and imaging protocols and segment it to a degree of satisfaction high enough for shipping after minor or no revision from a human operator. It is also important for the pipeline to be easily adaptable to new segmentation tasks.

1.2 Fatty liver disease

The liver is an organ found in the abdomen which performs an array of functions that support metabolism, digestion, detoxification and vitamin storage. Fatty liver disease is the name of a condition where fat builds up in the liver, it can be subdivided into two categories, alcoholic and nonalcoholic fatty liver disease. By analysing scans created by medical imaging, one can diagnose and monitor the progression or regression of a disease. This methodology can also be used to conclude if the liver has an abnormal fraction of fat depots or volume, where a fat-fraction of over 5.56% is considered unhealthy, see [2].

1.3 MRI

A commonly used method for medical imaging is MRI. By the use of strong magnetic fields and radio waves, the hydrogen protons in the tissue are excited and then themselves emit radio signals, which can then be captured by sensors. The information is then used in a computer to generate an image.

When creating a volumetric scan, one usually collects 2D images with a given resolution e.g. $2\ \mathrm{mm}$ height and $2\ \mathrm{mm}$ width per pixel and stacks these on top of one another, creating an "image stack". With a given slice thickness, indicating the distance that the pixel represents in the third dimension e.g. $5\ \mathrm{mm}$, this yields a voxel, which in this case is $20\ \mathrm{mm}^3$. Multiplying this value with the number of voxels segmented as liver gives the volume of the liver, which makes it possible to determine the volume of a liver.

By adjusting the imaging protocol of the scanning device, one can excite the hydrogen nuclei of water and fat by using a frequency that is in resonance with their spin precession. The water and fat molecules have a slightly different spin frequency. Utilizing multifrequency measurements and the spin frequency, one can create two different images for water and fat. This is done by looking at how the response from the tissue changes i.e. how the vector sum of the spin changes over time, to find the proportion of fat and water in a given voxel. By combining the two images, one can create a composite image called a fat-fraction image, where each pixel in the image has a value that corresponds to the percentage of fat in that pixel, also known as fat-fraction. The pixel values segmented as liver fat are then used to determine the fat percentage by taking the mean or median value of the segmented pixels belonging to the positive class, i. e. the liver.

2 Deep Learning Background

2.1 Convolutional Neural Networks

A basic structure in an artificial neural network is the fully connected layer, in which weights are associated with every possible combination of nodes between two layers. This becomes a problem when multiple layers are added, as the number of parameters can reach hundreds of millions, which becomes exceedingly expensive to compute. A CNN solves this by assuming that the information needed for making a decision often is spatially close and thus, only takes the weighted sum over nearby inputs. It also assumes that the networks' kernels can be reused for all nodes, hence the number of weights can be drastically reduced. To counteract only one feature being learnt per layer, multiple kernels are applied to the input which creates parallel channels in the output. Consecutive layers can also be stacked to allow the network to find more high-level features. This means that the network, for example, can detect a liver-shaped object. For every layer added, the receptive field, the region of the image that contributes information, grows depending on the kernel size allowing it to find larger patterns to use for segmentation.

CNN layers are commonly mixed with pooling layers, which allows more kernels, that finds additional high-level features without increasing the computational load. This is done by a non linear down-sampling, e.g. taking the maximum value of a 2x2 square and passing that signal to the next layer.

If a network is to be used as a segmentation network, it is necessary to up-sample the most high level features, making a pixel-to-pixel comparison possible between the input and output of the network. This can be done with transposed convolution which works like regular convolution but instead takes one node value and sends it through a kernel of weights to up-sample it.

2.2 U-Net

The U-Net is a popular networks architecture for medical segmentation (cf. [3][4][5]). It consists of two parts, the contracting part (or the encoder/downward part), which extracts high-level features and the expanding part (or the decoder/upward part), decoding the high-level features to be used in segmentation. Both parts consist mainly of convolutional layers, mixed with pooling layers for the contraction and transposed convolution layers for the expansion. What is unique with the U-Net is that it connects corresponding convolutional blocks in the encoder and the decoder via long skip connections (also called bridges). This allows more local information, which would otherwise be lost deeper down, to skip the bottleneck. The result is that the U-Net can use both more high-level features as shapes and local information as texture for the classification of a specific pixel. The implementation of the U-Net as shown in [1] differs from the original in that it uses internal padding, which prevents the kernel size from shrinking the feature maps. This allows us to make the assumption that every pixel, even the border regions can be segmented, hence the output will have the same height and width as the input, which in turn allows us to ignore the cropping in the long skip connections that was used in the original U-Net implementation[6].

2.3 Dropout

Dropout is a regularisation technique, where certain nodes are randomly, completely ignored in training to reduce the chance of a network over-fitting. The idea is that the nodes "take turn" trying to minimize the loss function, which in theory means that the network learns not to over-rely on a few nodes. Dropout could also help to reduce the risk that some layers are trained to compensate for the mistakes of others, so-called "co-adaptation", which typically does not generalise very well. Our U-Net implementation uses dropout in the two deepest convolutional blocks.

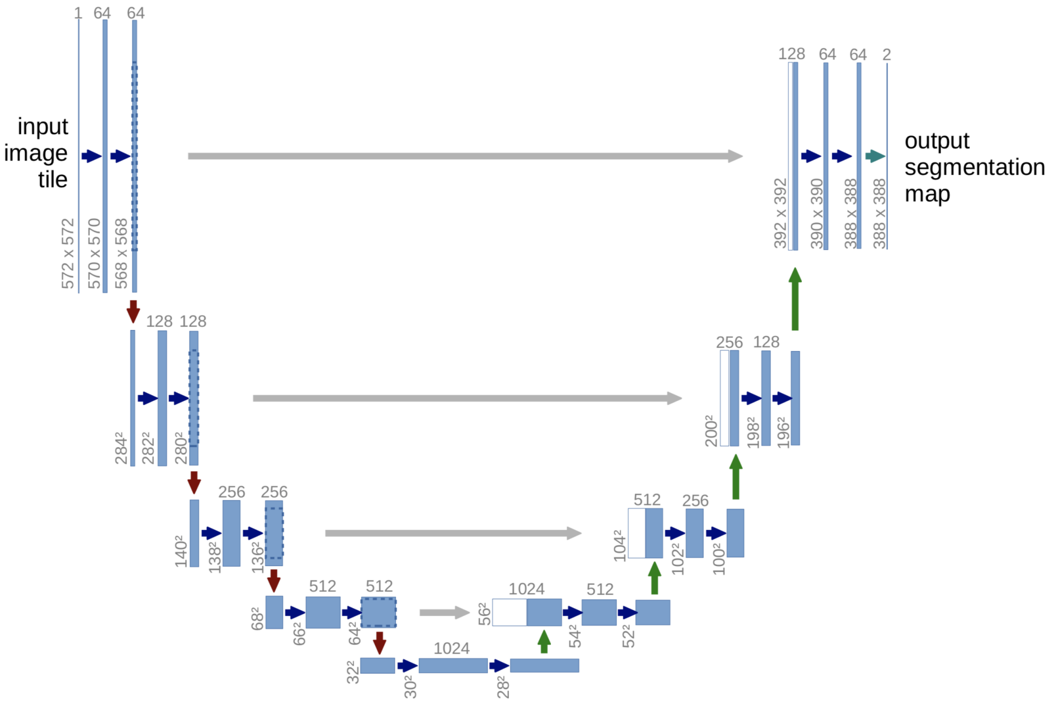

Figure 1: The figure illustrates the U-Net architecture.

2.4 Transfer Learning

One inherent problem in deep learning, and in almost all applications of machine learning, is the reliance on large amounts of data. Without enough data, the millions of parameters in a deep learning architecture can not be properly updated from training and it would be hard to find usable high-level features. Instead, the network would find solutions that over-fit to the training data. Transfer learning solves this problem partially, by using pre-trained weights on tasks not necessarily similar to the one at hand. The pre-trained weights are generally trained on datasets with millions of images e.g. ImageNet, containing around 14 million images in 1000 different categories, giving us networks like the VGG11[7]. The features extracted from such a wide variety of images and classification tasks have the potential to help in segmentation problems. Even if the images are very different from the task at hand, extracting lines and edges and objects of a certain shape, e.g. a circle is useful to most networks. This is not a novel approach and has been explored in previous works[8].

2.5 Volumetric slicer and/or V-net

Often medical imaging techniques yield a 3D volume scan. U-Net, however, has a major limitation to work only with 2D information so that every slice is segmented separately and then needs to be stacked together into a volume. In order not to waste the 3D information, the V-net[9] was developed, which uses three dimensional convolution instead. The results from similar studies to ours, such as [1] and [10] suggest the V-Net is often inferior to the U-Net. Instead we explored a compromise by continuing to segment a 2D slice but instead of only feeding the network the slice to be segmented, the neighbouring slices were also added to the input as extra channels. This method was used successfully by the winning team in Liver Tumor Segmentation Challenge as can be seen in [10].

2.6 Short skip connections

Kaiming et al.[11] showed that among other things that short skip connections could be added to the VGG networks in order to combat the problem of vanishing gradients and to smooth the optimisation problem. Adding short skip connections also makes the network more flexible by allowing it to dynamically choose which convolutional blocks to utilise. As the U-Net is closely related to this family of networks it is easy to add the short skip connections over the convolutional blocks.

2.7 Augmentations

Augmentation is a common technique in deep learning, where the data is manipulated in different ways to create more data out of an otherwise limited data set. For example, an image may be flipped 90 degrees to emulate a new image, meaning the network has more data to train on and could, in theory, be invariant to the orientation of the input. Augmentation serves to make the network less prone to over-fitting, typically implying better generalisation. One augmentation technique used by our pipeline is elastic deformations, as described in [12], where a grid of a fixed amount of nodes is overlaid onto the image and then randomly deformed so that the underlying image is interpolated to follow the grids deformation. This technique has the ability to create unique, plausible data in contrast to many other augmentations which often does not change the shape of the objects in an image.

3 Datasets and Implementation

3.1 Datasets used

The two datasets mainly used were from two different studies. Both were produced with MRI-imaging, described in 1.3, run along the transversal plane. The datasets originated from two different scanners, Philips 1.5T and GE 3T. For the Philips 1.5T scanner the THRIVE sequence was used for Liver Volume, generating 55 slices with a slice thickness of 5 mm, a width and height of 2mm. The Liver fat for the Philips 1.5T (dataset 1) scanner's images had a slice thickness of 6 mm with 34 slices and pixels with 1.95mm height and width. For the GE 3T (dataset 2) scanner the water-fat sequence was used for Liver Volume, generating 28 slices with a slice thickness of 5 mm, a width and height of 1.5mm. The Liver fat for the GE 3T scanner's images had a slice thickness of 10 mm with 25 slices and pixels with 1.72mm height and width. This results in quite different scans, as can be seen in Figure 2. At the start of the project, 11 % of the data was left aside as test data, only to be used when the best architecture and set of hyperparameters were found.

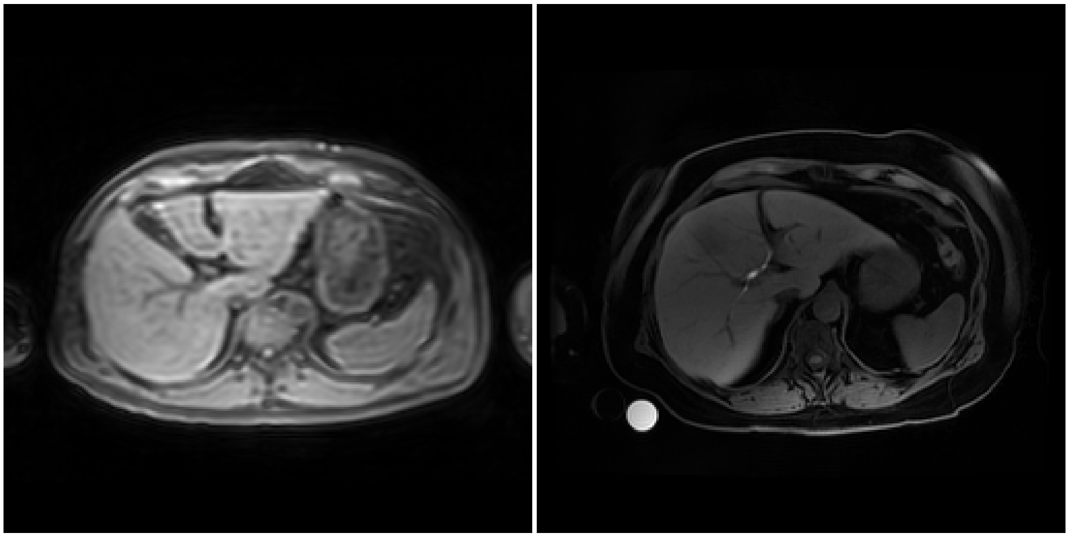

Figure 2: Volume scan from dataset 1 (left) and dataset 2 (right) after harmonising spacing.

It is also important to note that when segmenting liver volume, the entire area which can be considered to be part of the liver is segmented. In contrast to when segmenting the liver fat images, where the human operators are instructed to maintain a certain margin to the border of the liver, to avoid partial volume effects, to ensure that no pixel contain both liver fat and background.

To ensure that the pipeline easily can be applied to different tasks an additional set is used. This set contained CT scans from the liver, abdomen and thigh, where abdomen and thigh had multiple classes. The liver was to be segmented as liver and background. The thigh had fat tissue, connective tissue, muscle tissue and background and lastly the abdomen had visceral fat, subcutaneous fat and background. This dataset only contained one 2D slice per patient, with 2151 patients.

3.2 Data processing

Since the data is volumetric and the U-Net processes 2D images we need to separate the patient volumes into slices. This means that the volumes need to be sliced along a chosen axis. There are three axes to choose from; transversal, which reads a body horizontally from head to toe, coronal, front to back, and sagittal, left to right. All types of orientations have been tried, but transversal has shown to be vastly superior.

For the network to better be able to handle scans from different MRI devices and imaging protocols that differs in properties, such as intensity ranges or resolution, the data can be processed to appear more similar. Hereafter we describe the techniques used in this study.

3.2.1 Resolution discrepancies

The two datasets contained scans taken with different pixel spacing, meaning that the actual distance between two neighbouring pixels was dissimilar in different images. By zooming into or out of the images with first order spline interpolation the pixel spacing could be harmonised. The pooling layers divides the width and height of the following tensor with two and we have four pooling layers, hence the input images side lengths need to be evenly dividable with 16. We chose 256x256 to be the number of pixels in our input images as that was the closest numbers dividable with 16 to the largest images (we chose the largest ones since we rather up-sample then down-sample to avoid losing information). If the image was not 256x256 after zooming it was padded (or cropped) to that size. Zooming was not applied in the third dimension since it resulted in too much information loss. This created some problems for the liver volume network, since adding neighbouring slices as spacing between slices was noticeably different in the two datasets. This resulted in only one neighbouring slice being added on each side when segmenting with the liver volume network compared to four for the fat-fraction network.

3.2.2 Normalisation

To make the intensities of the pixels more aligned, min-max normalisation and a method which we call "clipping" were used on the scans.

Normalisation can be done in three ways, where the maximum and the minimum pixel intensities are picked from either individual slices, the entire image stack per patient or the entire collection of patients are considered. This is referred to as slice-wise, patient-wise and global normalisation. The min-max normalisation, as described in 1, when applied all intensities are confined to an interval from 0 to 1 as can be seen in the following equation,

$$y_i = \frac{x_i - min(x)}{max(x) - min(x)}. \tag{1}$$

Clipping, in this context, is a technique where a certain percentile of the highest or lowest intensity pixels are "clipped" to the value of a pixel just below or above a given threshold, which removes outliers.

3.3 Evaluation metrics

To evaluate the performance of the networks, evaluation metrics are needed. In order to estimate liver volume and liver fat percentage, the following methods were implemented.

For liver volume, the number of segmented pixels were counted per patient. The fat-fraction is a bit more complicated. The value comes from the mean/median pixel value of the fat-fraction scan in the liver class. The discrepancies between the predicted and the ground truth of these values were then calculated for evaluation. The 95% Limit of Agreement was also used and describes where the 95%

boundary of the positive and negative error, based upon the mean error, lies. It indicates how the the errors are biased, towards either under- or over-segmentation. The 95% LoA as described in 2 and 3 can be calculated in the following equations, where μ is the mean value of the error, and σ is the standard deviation of the errors,

$$LoA_{upper} = \mu + 1.96\sigma, \tag{2}$$

$$LoA_{lower} = \mu - 1.96\sigma. \tag{3}$$

Other metrics used, which are also common for segmentation of medical images, were the Sørensen–Dice coefficient (Dice) and Mean Absolute Error (MAE). Although all metrics can be used for evaluation, Dice, as described in 4, is the one most present in papers on the topic, therefore it was primarily considered when evaluating the results. It computes the overlap in the positive class, X of the network's prediction, Y and ground truth as

$$Dice = \frac{2 \mid X \cap Y \mid}{\mid X \mid + \mid Y \mid} = \frac{2TP}{2TP + FP + FN}, \tag{4}$$

where TP means "True Positive", FP means "False Positive" and FN means "False Negative".

3.4 Run setup

The testing of different models and hyperparameters where done in three steps.

- All model changes where tried individually and run with five-fold cross-validation for 75000 iterations and then compared with the baseline U-Net.
- The results were used to determine which improvements were chosen for the second, more rigorous validation step where each combination of these changes were tried. This time with 10-fold cross-validation and for 100000 iterations.
- The best performing network was chosen to process the test data

The results from the last step will be the focus of the results part, while most of the earlier results where excluded in order to limit the extent of the report.

Unless stated otherwise, cross entropy was used as a loss function while Adam, with the implementation described in[13], was used as the optimiser. After the first series of tests, the hyperparameters were set accordingly, learning rate = 0.0001, β_1, which is the exponential decay rate for first moment estimates in Adam was set to 0.9, β_2, which is the exponential decay rate for the second-moment estimates in Adam was set to 0.999. σ, which is the the strength of the elastic deformations was set to 8 and 64 grid points were used for the deformations and finally batch size was set to 1 and dropout was set to 0.5.

All runs in testing and training were performed on three Dell 5820 Precision Tower desktops each equipped with an Intel® Xeon® W-2133 3.6 GHz processor and an Nvidia GeForce RTX 2080 TI graphics card with 12 GB memory.

3.5 Deployment

The best performing liver volume and fat-fraction networks were saved onto a computer acting as server. Other computers on the same network are then able to send raw datasets to the server, where it is preprocessed and fed to the network so that a prediction is made. The preprocessing is then reversed and the file saved in .vtk format. When finished, the server returns the segmented data to the sender. This is enabled by the use of ØMQ, which is an asynchronous messaging library available in Python. The absolute majority of the time it takes to make the inference runs are spent zipping, unzipping and preprocessing the raw data while the inference itself takes very little time.

4 Results

In this section we present the results of changes in the method that actually improved our measures in order to keep the size of the report reasonable.

4.1 Liver volume network

In Table 1 we see that using transfer learning, partial volumetric information with the adjacent slices, and skip connections achieved among the best. It was chosen as the best one since it was the most stable. Figures 3, 4, 5 and 6 show the differences in performance between the best and the base U-Net. An example of a representable segmentation is seen in Figure 7.

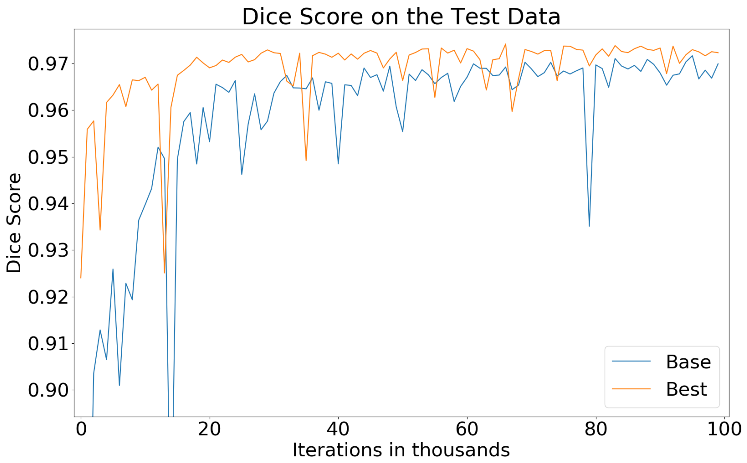

Figure 3: The best model, in orange, using both transfer learning as an encoder and adding one slice above and below as input to the network, compares to the basic implementation of the U-Net, in blue.

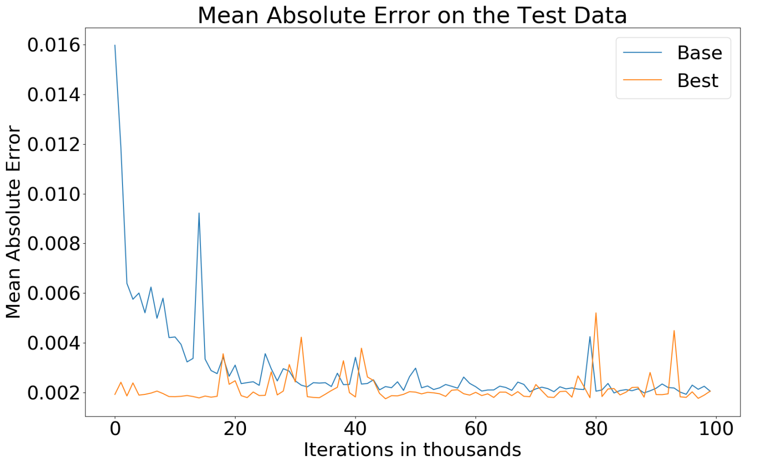

Figure 4: The best model, in orange, using both transfer learning as an encoder and adding one slice above and below as input to the network, compares to the basic implementation of the U-Net, in blue.

For the volume predictions of the base network as seen in Figure 5, the mean value is 1.6% off and the median is 0.9%, while the maximum outlier is 6.5% off. The 95% Limit of Agreement for this network is +4067 and -4867, which means that it is equally biased to making optimistic and pessimistic predictions.

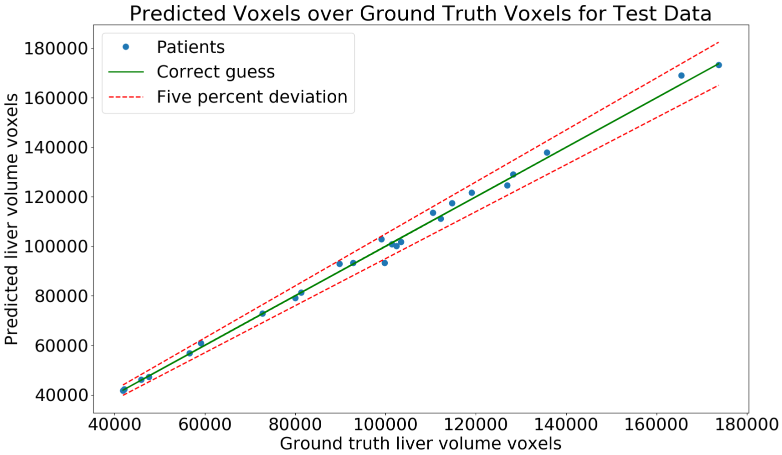

Figure 5: Model prediction of patients liver volume when evaluated on the test data using the base U-Net.

For the volume predictions of the best network as seen in Figure 6, the mean difference is 1.1% and the median is 1.5% off and the maximum outlier is 5.2%. The 95% Limit of Agreement is +5506 and -1815, hence it can be concluded that it is quite biased towards optimistic predictions.

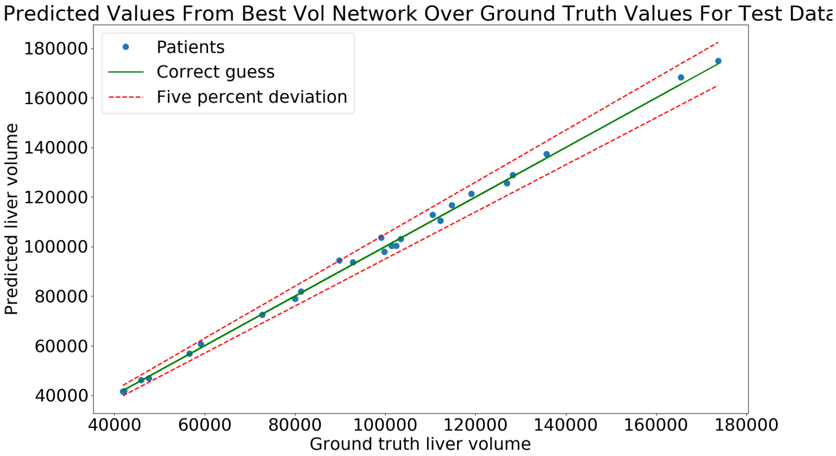

Figure 6: Model predictions of patients liver volume when evaluated on the test data using our best implementation.

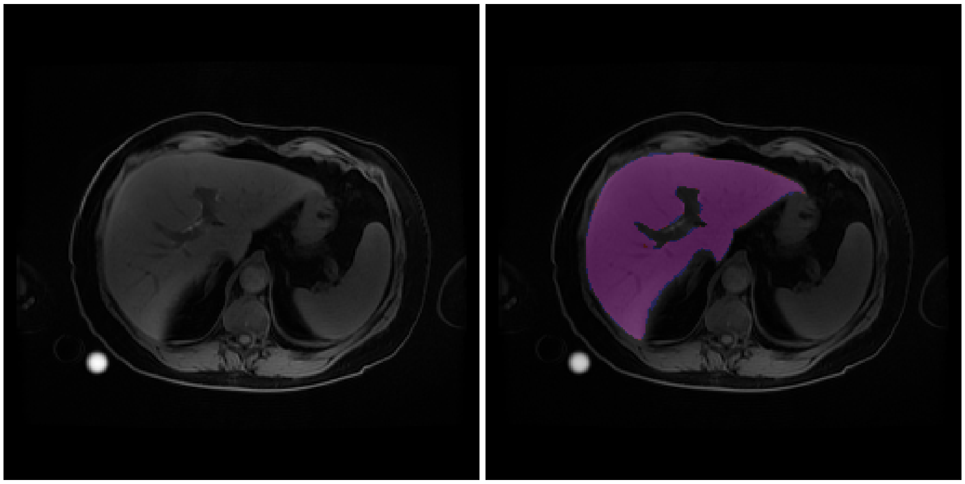

Figure 7: Unsegmented MRI scan on the left and the proposed liver volume segmentation from our model. Where purple is overlap when compared to the ground truth, red is under-segmentation and blue is over-segmentation. Here, a fairly representative segmentation is shown.

4.2 Fat-fraction network

In Table 1 we can see that using all improvements reached the same Dice as leaving short skip connections out, but since the Dice curve achieved more stability we preferred using all improvements. In Figures 8 and 9, the differences in Dice are shown between the optimal and the base U-Net during testing.

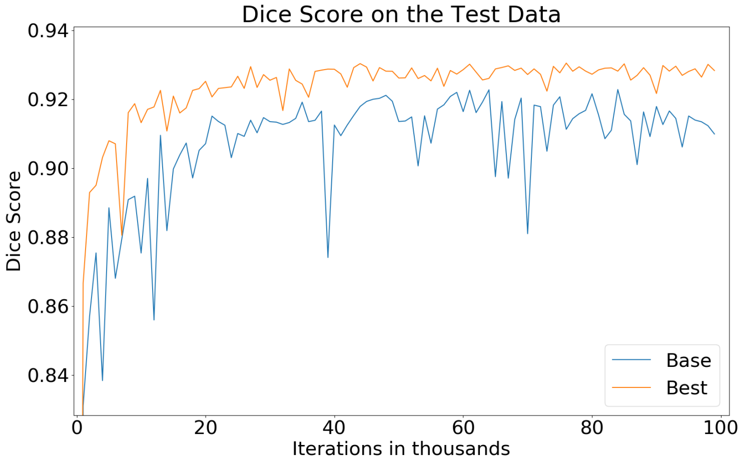

Figure 8: Dice score for the best model used on test data in comparison with the base network, the difference in highest Dice is 0.08, which corresponds to about 10% less error but the best network is also much more stable.

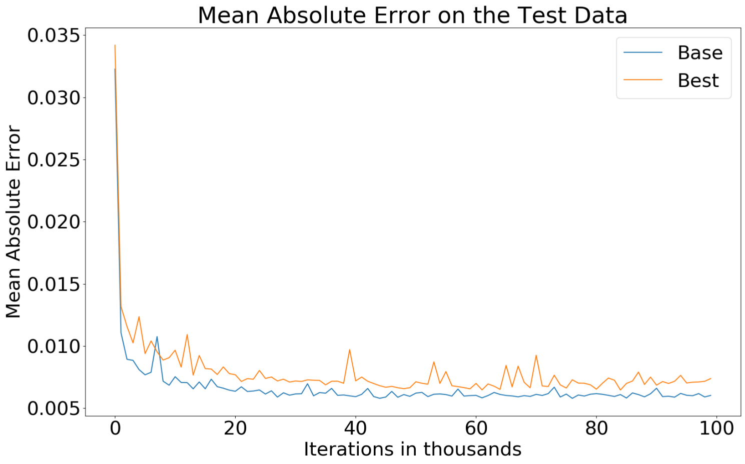

Figure 9: MAE for the best model used on test data in comparison with the base network.

The fat-fraction values for the basic U-Net seen in Figure 10 gave relative mean differences of 1.5% and had three outliers that don't match the ground truth with differences of one percentage point. Important to note is that one patient would have been predicted to not have fatty liver disease, where as ground truth says it has fatty liver disease since the fat-fraction is larger than 5.5%. The Limit of Agreement is +6.521 and -7.284.

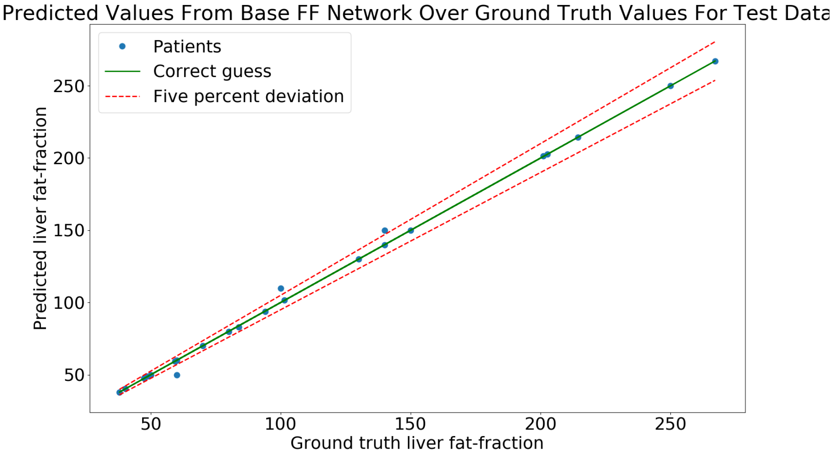

Figure 10: Model predictions of median liver fat-fraction when evaluated on the test data using the base U-Net. Three patients in the test data has their predictions differed by more than 5%.

The fat-fraction values for the best U-Net seen in in Figure 11 gave relative mean differences of 0.4% and one outlier, which had a relative difference of 5%. This patient had a large cyst which was incorrectly segmented, an occurrence previously unseen to the network since it is only present in the test data. The Limit of Agreement is +1.983 and -1.745.

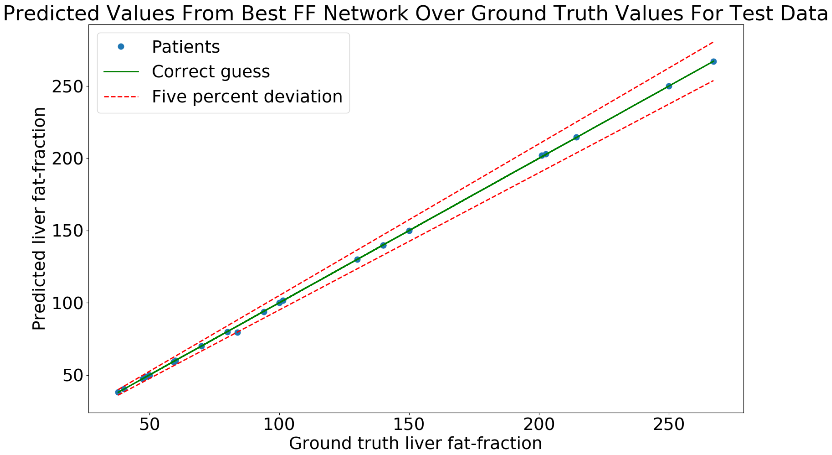

Figure 11: Model predictions of median liver fat-fraction when evaluated on the test data using our best implementation. The median liver fat differed by 5% from the ground truth for only one of the patients.

4.3 Combinations of methods that improved the results

Methods used	Highest Dice, volume	Highest Dice, fat-fraction
Base	0.966	0.922
E	0.965	0.925
V	0.968	0.930
R	0.965	0.923
T	0.968	0.929
E + V	0.966	0.931
E + R	0.966	0.926
E + T	0.969	0.931
V + R	0.967	0.931
V + T	**0.971**	0.931
R + T	0.970	0.928
E + V + R	0.968	0.932
E + V + T	0.970	**0.933**
E + R + T	0.970	0.931
V + R + T	**0.971**	0.931
E + V + R + T	**0.971**	**0.933**

Table 1: Combinations of methods that previously showed improvements. E = Elastic deformations, V = Partial volumetric information, T = Transfer learning, R = short skip connections.

4.4 Generalisation between datasets

Number of patient scans moved	Dice, volume	Dice, fat-fraction
0	0.493	0.342
1	0.957	0.876
3	0.955	0.853
5	0.963	0.895
10	0.968	0.908

Table 2: Dice score as patient scans from dataset 2 are manually segmented and moved to dataset 1, when training on dataset 1 and evaluating on dataset 2.

To test the networks ability to generalise we used one of the datasets for training and one of the datasets for evaluation. We then moved a number of patient scans from the evaluation set to the training set. In Table 2 we see that a network trained on dataset 1 generalises very poorly to dataset 2. However by manually segmenting a few patient scans in dataset 2 and adding them to dataset 1 the Dice scores are vastly improved.

4.5 Other datasets

With minor changes to the data preprocessing our pipeline could create models, which reached a Dice score of 0.968 for liver segmentation, 0.991 for abdomen segmentation and 0.994 for thigh segmentation, visual results can be seen in Figure 12. These comparably high Dice scores might be explained by the number of patients available for training and notably easier classes to separate.

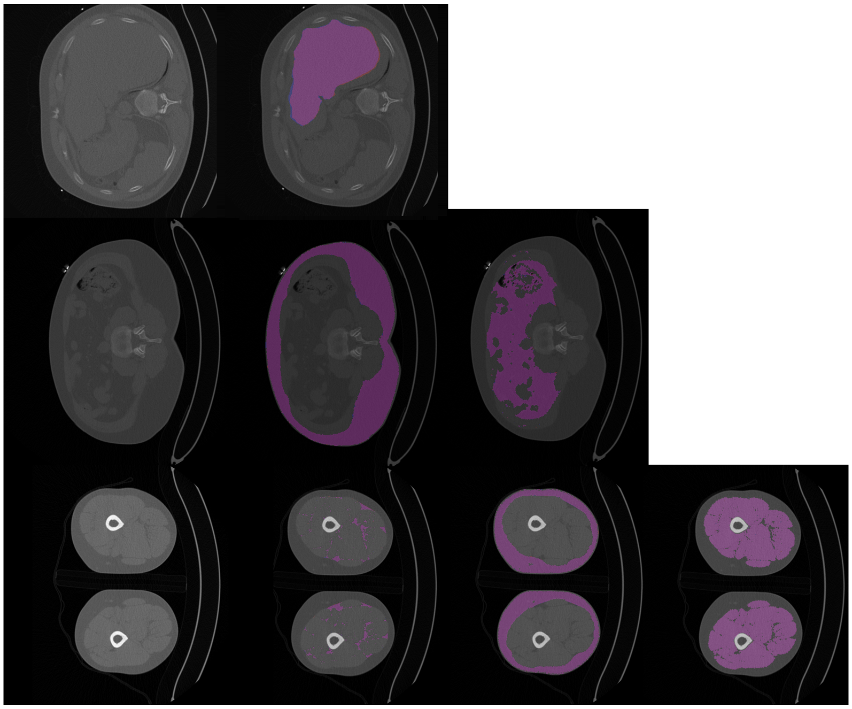

Figure 12: Results on other datasets. Red: under-segmentation, blue: over-segmentation, purple: correct segmentation.

4.6 Normalisation

In Figure 13 we see that normalising the inputs did not improve the results for either the volume network or the fat-fraction network.

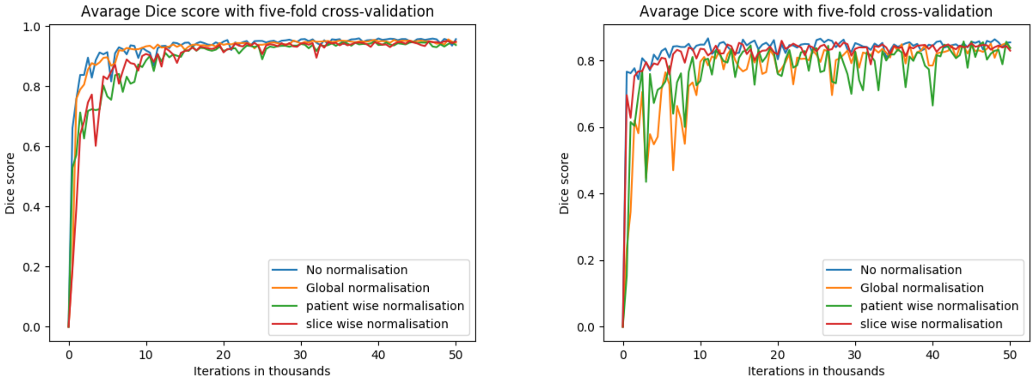

Figure 13: Left: Different normalisations applied to the input data for volume network. Right: Different normalisations applied to input data for the fat-fraction network.

5 Discussion

It is important to note that the ground truth segmentations are subjective. A human operator that segments the same image twice will do it in a slightly different way, which creates an intra-operator error. This, in turn, should create a theoretical max of how well our network can become since what is considered ground truth may vary from slice to slice and the network may never converge since there is no objective truth to converge towards. Antaros Medical keeps statistics of the operators mean difference in predicted liver volume, where the average operator had a 1.7% deviation, while our best network has an average difference of 1.1%. This in turn, is not a perfect comparison, as different datasets were used and the measure is not an actual error measure, but rather a measure of the operators' bias toward over/under segmentation. Despite this, it at least puts our network's performance fairly close to the theoretical optimum.

In Table 1 many combinations have very similar Dice scores and it is therefore hard to draw solid conclusions from these runs. However, it can be seen that transfer learning and the partial volumetric information generally are more important than the short skip connections and the elastic deformations.

The most unexpected result was that normalising the data did not yield any better result for neither the volume nor fat-fraction network. One could speculate that this could be explained by two different datasets being used with very differing qualities and hence different features. It could therefore be necessary that the network easily can tell those datasets apart and this ability could be hampered by normalising the data.

The only data augmentation that helped us was using elastic deformations. One explanation to this could be that we had relatively many patient volumes to train on. This decreased the need for creating additional data.

For a neural network to be useful in practice it needs to be able to generalise well to new data. It would not be particularly useful if 100 patient scans needs to be segmented manually so that a remaining 50 can be segmented automatically. In Table 2 we can see that the volume network generalise very poorly between the two datasets, however it seems that by just moving some (perhaps as few as one) of the validation datasets patients to the training, the ability to generalise is greatly increased. We find this to be very interesting and worthy of future investigation since the ability to generalise to new studies is vital if the networks are to be deployed. The varying Dice scores that can be seen in Table 2. It may seem like moving one patient scan is preferable to moving three, but this can be explained by the fluctuations that can be seen in Figure 11. These fluctuations may increase as there are only a few of the patients that are from the dataset, which is used during evaluation.

Lastly, the results of the other dataset shows that it is easy to adapt the current pipeline to completely different segmentation problems and get segmentations of high quality.

6 Conclusion

This project has produced an easily adaptable deep learning pipeline and two networks that can utilise, among other things, transfer learning and volumetric information. The liver volume network reached a Dice score of 0.971. It predicted 24 out of 25 liver volumes with less than 5% error. The fat-fraction network reached a Dice of 0.933 and for 24 out of 25 patients the network predicted the correct median fat-fraction value. The results also suggest that the ability to generalise can be raised to acceptable levels by manually segmenting one patient scan in a new dataset and adding them to the training set. This improved the volumetric network's Dice score from 0.493 to 0.957. Moving one patient scan raised the fat-fraction network's Dice score from 0.342 to 0.876.

References

[1] Taro Langner, Anders Hedström, Katharina Mörwald, Daniel Weghuber, Anders Forslund, Peter Bergsten, Håkan Ahlström, and Joel Kullberg. Fully convolutional networks for automated segmentation of abdominal adipose tissue depots in multicenter water-fat mri. *Magnetic Resonance in Medicine*, 00:1–10, 2018.

[2] Jeffrey D Browning, Lidia S Szczepaniak, Robert Dobbins, Pamela Nuremberg, Jay D Horton, Jonathan C Cohen, Scott M Grundy, and Helen H Hobbs. Prevalence of hepatic steatosis in an urban population in the united states: impact of ethnicity. *Hepatology*, 40(6):1387–1395, 2004.

[3] Berk Norman, Valentina Pedoia, and Sharmila Majumdar. Use of 2d u-net convolutional neural networks for automated cartilage and meniscus segmentation of knee mr imaging data to determine relaxometry and morphometry. *Radiology*, 288(1):177–185, 2018.

[4] Artem Sevastopolsky. Optic disc and cup segmentation methods for glaucoma detection with modification of u-net convolutional neural network. *Pattern Recognition and Image Analysis*, 27(3):618–624, 2017.

[5] Brahim Ait Skourt, Abdelhamid El Hassani, and Aicha Majda. Lung ct image segmentation using deep neural networks. *Procedia Computer Science*, 127:109–113, 2018.

[6] 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*, pages 234–241. Springer, 2015.

[7] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[8] Vladimir Iglovikov and Alexey Shvets. Ternausnet: U-net with VGG11 encoder pre-trained on imagenet for image segmentation. *CoRR*, abs/1801.05746, 2018.

[9] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. *CoRR*, abs/1606.04797, 2016.

[10] Patrick Bilic, Patrick Ferdinand Christ, Eugene Vorontsov, Grzegorz Chlebus, Hao Chen, Qi Dou, Chi-Wing Fu, Xiao Han, Pheng-Ann Heng, Jürgen Hesser, Samuel Kadoury, Tomasz K. Konopczynski, Miao Le, Chunming Li, Xiaomeng Li, Jana Lipková, John S. Lowengrub, Hans Meine, Jan Hendrik Moltz, Chris Pal, Marie Piraud, Xiaojuan Qi, Jin Qi, Markus Rempfler, Karsten Roth, Andrea Schenk, Anjany Sekuboyina, Ping Zhou, Christian Hülsemeyer, Marcel Beetz, Florian Ettlinger, Felix Grün, Georgios Kaissis, Fabian Lohöfer, Rickmer Braren, Julian Holch, Felix Hofmann, Wieland H. Sommer, Volker Heinemann, Colin Jacobs, Gabriel Efrain Humpire Mamani, Bram van Ginneken, Gabriel Chartrand, An Tang, Michal Drozdzal, Avi Ben-Cohen, Eyal Klang, Michal Marianne Amitai, Eli Konen, Hayit Greenspan, Johan Moreau, Alexandre Hostettler, Luc Soler, Refael Vivanti, Adi Szeskin, Naama Lev-Cohain, Jacob Sosna, Leo Joskowicz, and Bjoern H. Menze. The liver tumor segmentation benchmark (lits). *CoRR*, abs/1901.04056, 2019.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015.

[12] Andreas Östling. Automated kidney segmentation in magnetic resonance imaging using u-net, 2019.

[13] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.