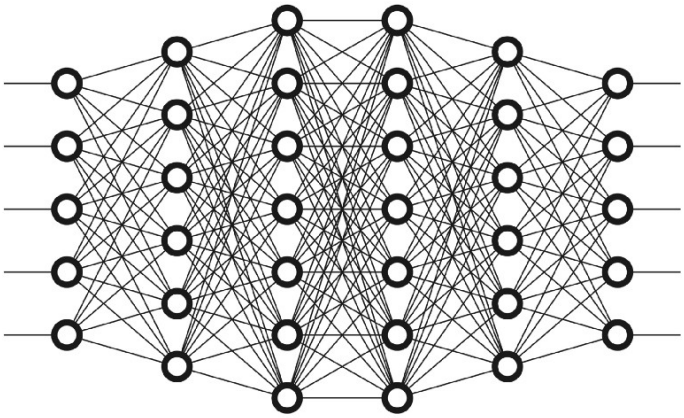
Development of a convolutional neural network for landmark detection of the levator plate

A U-net based structure for MR image segmentation

Kyra de Bree - s2108259



Daily supervisor: Frieda van den Noort
Chair of committee: Anique Bellos-Grob
External member: Frank Simonis

Multi-Modality Medical Imaging (M3I)
University of Twente
06-02-2023 - 20-04-2023

The image on the front page belongs to [1].

Contents

Samenvatting

Veel vrouwen krijgen in hun leven te maken met een verzakking van de bekkenbodemorganen (POP), en uit onderzoek is gebleken dat dit zich mede uit in de vorm van de levator plate (LP). Deze is gedefinieerd als de middellijn waar het diafragma van de pelvis samenkomt, tussen het rectum en de coccyx. Het definieren van deze vorm is echter nog lastig en tijdrovend, dus is het doel van dit onderzoek om de detectie van de anatomische punten met betrekking tot de LP te automatiseren. Er hebben 60 vrouwen deelgenomen aan dit onderzoek, wat in totaal tot 171 bruikbare scans leidde. Het model is gebaseerd op een U-net structuur en er zijn 12 ingetekende coördinaten die allen een anatomisch punt in de bekkenboden representeren. Deze zijn omgezet in Gaussische kernels, hierbij wordt het probleem benaderd als een segmentatieprobleem. De gemiddelde afstand tussen het originele coördinaat en het punt dat door het model is gevonden is 7.0 mm, dit was 6.1 mm voor de punten waaruit de LP bestaat. Het model lijkt geen verschil in werking te hebben tussen de scans van de POP-patienten tegenover niet-patienten, wat het model geschikt zou kunnen maken om een analyse te doen tussen deze groepen. Concluderend kan er gezegd worden dat het gelukt is om de detectie van anatomische punten met betrekking to de LP te automatiseren.

Introduction

About 40 percent of women worldwide will experience pelvic organ prolapse (POP) [2]. Pelvic organ prolapse is when one or more of the organs located in the pelvis slip down from their normal position and bulge into the vagina [3]. Causes of POP include but are not limited to pregnancy and childbirth, going through menopause, being overweight, having long-term constipation and having a hysterectomy [3].

One of the main problems of pelvic organ prolapse is the need for a reliable and consistent staging method. In 1996, a standard system of terminology was introduced to describe, quantify and stage POP [4], and in 2002 POP-Q was introduced into the medical world [5] and it has been the scale of measurement ever since. POP-Q measurements are done during a physical exam and are recorded using a ruler or tape measure while the patient is in supine position[5]. However, it has been shown using magnetic resonance imaging (MRI) that the extent of a prolapse is significantly larger in upright position than in supine position [6]. MRI measurements have also shown that there is a correlation between the LP shape and the severity of POP [7]. Furthermore, there is also a correlation between the LP shape and long term recurrence of prolapse after prolapse repair [8]. To analyse the LP shape, twelve points that represent anatomical landmarks have been hand-drawn in midsagittal MRI slices. This is a time consuming task, and has led to the need for automation. In recent years, deep learning has been introduced in medical imaging to assist with image classification, object detection, image segmentation and more[9]. Since deep learning has shown great results in other landmark detection problems, this raises the question how it can be of assistance in the automation of the analysis of the LP shape.

Theory

3.1 Pelvic organ prolapse

Pelvic organ prolapse is defined by descent of the anterior vaginal wall, posterior vaginal wall, uterus, or vaginal apex into the vagina. Prolapse of pelvic structures can cause a sensation of pelvic pressure or bulging through the vaginal opening and may be associated with urinary incontinence, voiding dysfunction, fecal incontinence, incomplete defecation, and sexual dysfunction [10].

3.2 Clinical picture

Pelvic organs are at risk of descending when the muscles and connective tissues holding them up are weakened, which can happen through childbirth, heavy lifting, a chronic cough or frequent constipation [11]. POP can occur in women of all ages, though it more commonly affects older women.The incidence of women who have POP is expected to increase by 46% by the year 2050 [12]. POP is generally divided in four stages, where a first degree prolapse means the organs have only slipped down a little, and a fourth stage means that more than one centimeter of the vagina or womb is bulging out of the vaginal opening [11]. Diagnosis of POP is done through a physical examination, and treatment is mainly based on the severity of the symptoms. Treatments include strengthening of the pelvic muscles through exercises, inserting a pessary or performing surgery [11].

3.2.1 Pelvic support

The organs of the pelvic floor are supported by the interaction between the levator ani muscles and the connective tissues that attach the uterus and vagina to the pelvic sidewalls [13]. The musculus levator ani is composed of three parts, namely the puborectalis, pubococcygeus and the iliococcygeus muscle, as can be seen in figure 3.1. Posteriorly, the levator ani attaches to the last two segments of the coccyx. The paired muscles of the pelvic diaphragm join to form a raphe and contribute to the anococcygeal ligament. This median raphe between the anus and the coccyx is called the levator plate (LP) and is the shelf on which the pelvic organs rest [14]. Weakness of the levator ani may loosen the sling behind the anorectum and cause the LP to sag down [14].

3.2.2 Levator plate shape

Multiple studies have shown a correlation between the orientation and shape of the LP and the severity of POP [7] [16]. In recent studies, a principal component analysis has been done to compare the shape of the LP between young and old women with prolapse [16]. This used the anatomical landmarks and structures as shown in figure 3.2, and has shown that the LP angle compared to the horizontal reference line (PICS Line) increases with POP. Another studies based on this analysis with nulliparous, parous, and parous post-menoausal women, found that the LP descends with childbirth and menopause [17].

3.3 Deep learning

Deep learning is a method in artificial intelligence where computers are taught to process data like the human brain; learn by manually labeled training examples [18] [19]. This is done by so called artificial neural networks, which are computing systems inspired by the way that biological neurons signal to each other [20]. A neural network typically consists of input nodes, hidden layers, and output nodes. These nodes are also called neurons. The term ”deep” refers to the number of layers in the neural network - any network with more than three layers is considered deep.

3.3.1 Weights and biases

All neurons from one layer are connected to the ones from the layer before and after them, and all these connections have their own weight and bias [20]. This correlation between the input neuron and the output neuron is given by the equation:

$$output = \sum(weight * input) + bias \tag{3.1}$$

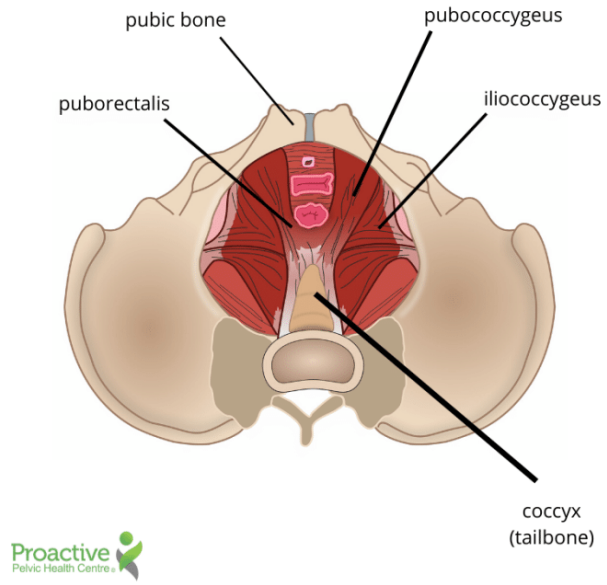


Figure 3.1: The levator ani muscle supports the pelvic floor and consists of three parts: the puborectalis, the pubococcygeus and the iliococcygeus muscle. Posteriorly, the levator ani attaches to the coccyx. The LP is where the muscles of the pelvic diaphragm join between the anus and the coccyx. [15]

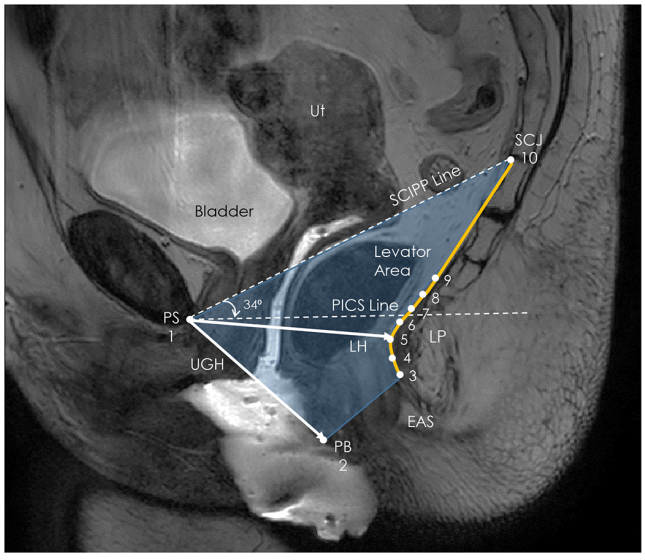


Figure 3.2: A midsaggital MRI slice with twelve anatomical landmarks used in research to analyse the LP shape differences between women. The abbreviations are; PS: pubic symphysis, UGH:urogenital hiatus, PB: perineal body, EAS: external anal sphincter, LP: levator plate, LH: levator hiatus, SCJ: sacrococcygeal joint, SCIPP line: sacrococcygeal to inferior pubic point line, Ut: uterus, PICS line: horizontal reference line. [8]

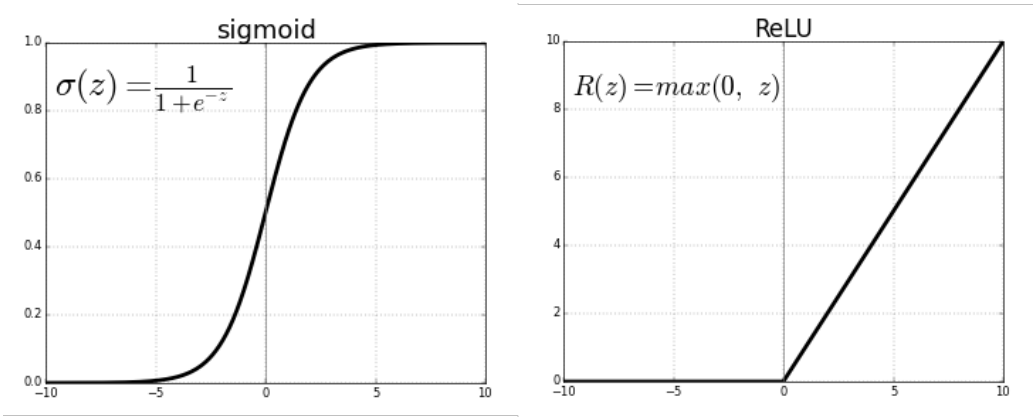


Figure 3.3: Two activation functions commonly used in neural networks, with on the x-axis the neural input and the y-axis the neural output. On the left the sigmoid activation function, on the right the ReLU activation function. [23]

This means that the weights and biases directly influence the output of the system, and are therefore the parameters that get trained by the neural network to have an outcome that is as close as possible to the desired output.

3.3.2 Activation functions

Before a value is passed as an input to a neuron, it is put through a so called activation function to normalize it. That is done to decide whether or not a neuron should be activated or not, mimicking the stimulation of a biological neuron [21]. Different activation functions influence the input values in their own way, below the two most commonly used are discussed.

Sigmoid function

The sigmoid function is used to squish values between zero and one, where an input value of zero gets translated into 0.5. This is presented mathematically in equation 3.2, and visually in figure 3.3. This function is most commonly used when values close to zero are relevant for the model, such as a system with a probability output between zero and one [21].

$$\sigma(z) = \frac{1}{1 + e^{-z}} \tag{3.2}$$

ReLU function

ReLU stands for Rectified Linear Unit. As the name already states, it is a function that rectifies values with a linear component. The ReLU function makes all values equal to or lower than zero equal to zero, and keeps the positive values the same, as can be seen in equation 3.3 and figure 3.3. An advantage of using the ReLU function is that it is only activated for positive neurons, and automatically deactivates neurons with a negative value. This makes the ReLU function a lot more computationally efficient compared to the sigmoid function [22], which also accelerates the gradient descent process as will be discussed later. With the deactivation of some neurons also comes a disadvantage, which has to do with backpropagation. This will also be discussed later on.

$$ReLU(z) = max(0, z) \tag{3.3}$$

3.3.3 Error, loss function and cost function

In learning networks, the error is the difference between the actual output and the desired output [24]. This error can be computed in different ways, examples are the mean absolute error and the mean squared error [25]. The way in which the error is calculated for a single training example is called the loss function. Different loss functions will give different errors for the same prediction, and therefore strongly influence the performance of the model. It depends on the type of task the model has, which loss function is best. Since the loss function is the quantification for one training example, it is still needed to have a measure for the performance of the model as a whole. This is the cost function. Essentially the cost function is a result of all the loss functions over an entire training data set [25].

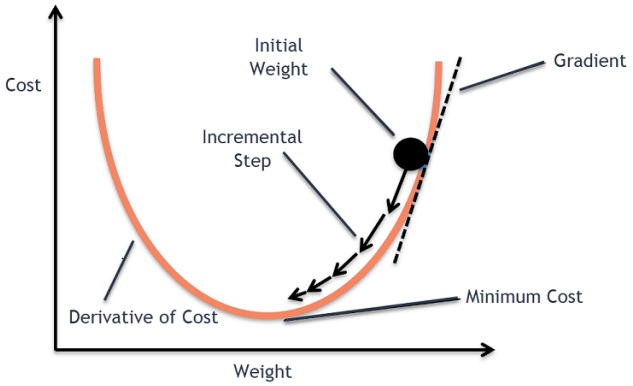


Figure 3.4: Backpropagation uses the gradient: the partial derivative of the loss function with respect to weights an biases. This is used in gradient descent to find the minimum of the cost function. [28]

3.3.4 Training a system: backpropagation and gradient descent

To minimize the cost function, backpropagation is used [26]. Backpropagation calculates the gradient, which is partial derivative of the loss function with respect to the weights and biases. The gradient represents what relative proportions to the change in weights and biases would cause the most rapid decrease to the cost. This is visually represented in figure 3.4. Each training iteration, which is called an epoch [27], will calculate the gradient and take a step towards its most negative value; this is called gradient descent. In a mathematical equation, this is represented as:

$$x_{n+1} = x_n - \alpha \nabla C(x_n) \tag{3.4}$$

where n is the number of epoch, α is the step size and ∇C is the gradient. Gradient descent is the basis for most optimization functions.

3.3.5 Optimisation functions

Optimisation functions, also called optimisation algorithms or optimizers, determine the way the weights and biases have to be adjusted [29]. Optimization algorithms can be divided into two categories: the constant learning rate algorithms and the adaptive learning algorithms. The constant learning rate algorithms have a constant learning rate which has to be chosen and tuned, an example is the Stochastic Gradient Descent. The adaptive learning algorithms use different learning rates for each iteration where the change in learning rate depends on the different parameters during training. Examples of adaptive learning algorithms are Adagrad and Adam, with Adam currently being the most used optimizer.

To come back to the activation functions, it was already mentioned that the sigmoid function squeezes all values between zero and one, and a ReLU function keeps all positive numbers the same. For the gradient descent this means that with a sigmoid function, from a certain number up, it does not matter how positive the value is, it will be translated into a one, and will therefore influence the gradient with the same power. With a ReLU function however, a positive value will stay the same, influencing the gradient descent more strongly if it is more positive. On the other hand, a disadvantage when working with ReLU is that inactivated nodes cannot be backpropagated. This is called the dying ReLU problem [30]. In essence, this means that when a neuron has a value below zero, the neuron gets deactivated and will not be activated again, even when the ideal value would be higher than zero.

3.3.6 Learning rate

Each activation function has a few parameters that are adapted for each model. The most important parameter is the learning rate, which influences the steps with which the weights are adjusted. The learning rate determines which minimum the model will converge to, and in which way [31]. If a learning rate is too small, approaching a minimum is prolonged, and it is likely that the minimum that is found is not the global minimum, but a local one. If a learning rate is too large, it will overshoot and "miss" the minimum it was converging to, taking longer than necessary. A learning rate is adequate when it oversteps local minima but converges at a decent rate. This is illustrated by figure 3.5.

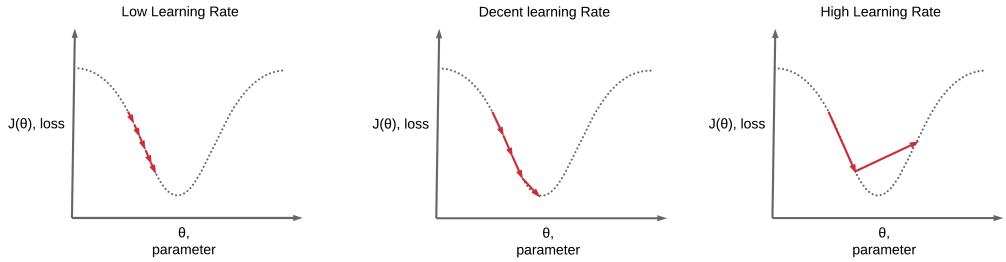


Figure 3.5: The learning rate influences the way the minimum is found. A too low learning rate will take long and converge locally, and a too high learning rate will not converge at all. [32]

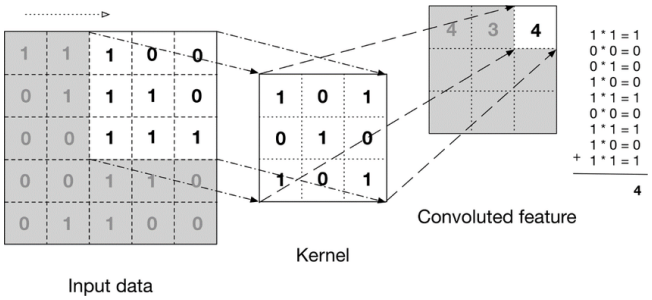


Figure 3.6: A 3 × 3 kernel slides over the input data and convolutes the pixel values with itself to project the result onto the next layer, resulting in a feature map. [36]

3.4 Convolutional neural networks

Neural networks have a large variation in types and applications, and the leading category for image classification is the convolutional neural network (CNN) [33].

3.4.1 Components of a CNN

CNN's mainly consist of three types of layers: convolutional layers, pooling layers and a fully connected layer [34]. The convolutional layer is the main building block of a CNN. In this layer a kernel, also called a filter, slides over the previous layer and convolutes itself with the pixels, creating a feature map. This is illustrated in figure 3.6. The size of a kernel can vary, but is typically 3 × 3 pixels [34]. The function of convolutional layers is to identify local correlations between pixels and extract fundamental features [35], and therefore the values in the kernel are the weights that get trained in a neural network. Pooling layers are also known as downsampling, and its function is to decrease the number of parameters and thereby reduce complexity and improve efficiency. Similarly to a convolutional layer, a pooling filter slides over the pixels and compresses the values onto a smaller map. However, its operation does not require weights, but simply uses the values that were present in the input. The two main types of pooling layers are max pooling and average pooling, which map the maximum value and the average value of the input in the filter, respectively [34]. In a CNN, the output values of a layer may get flattened, which means every pixel value gets represented in a one dimensional array. This array is the input for the fully connected (FC) layer, which connects the pixel values to the possible classifications through neurons.

3.4.2 Fully convolutional network (FCN)

A CNN without FC layer is called a fully convolutional network (FCN). Where a CNN that ends with a fully connected layer has an output of a classification probability, a FCN has an output of a heatmap that represents the probability in pixel values of an image [37]. These heatmaps are also called masks. Because of this, FCN is suitable for image segmentation and has been widely used for medical imaging tasks and organ localization [38]. One of the best known and most employed FCN's is U-net [39].

3.4.3 U-net

U-net [40] is designed for biomedical image segmentation, and has a symmetrical architecture with a contracting path, bridge, expansive path and skip connections, as can be seen in figure 3.7. The contracting part encodes the pixel values in an abstract representation so that with each layer, a different level of cohesion is extracted, creating a feature hierarchy. The bridge connects the contracting path with the expansive path, whereas the

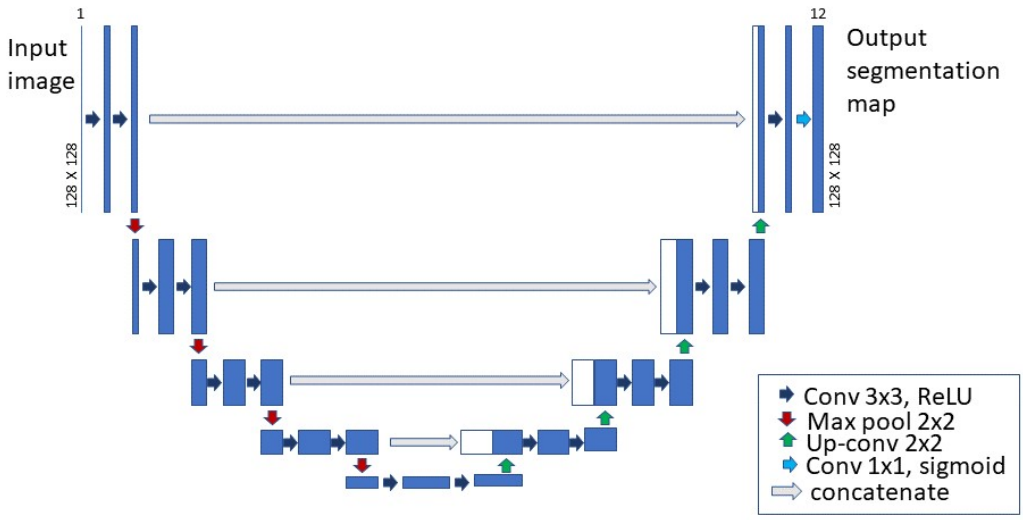


Figure 3.7: U-net structure with on the left its input, on the right its output and in between an architecture of convolutional layers and pooling layers. This image is based on the model of the original publishers of U-net; [40].

expansive path decodes the abstract representation and generates a segmentation mask. The skip connections act as a shortcut and provide additional information to the expansive path, which helps with more precise image segmentation and more efficient backpropagation.

3.5 Evaluation and validation

3.5.1 Train, validation and test data

In order to evaluate and validate a model, its data set is divided into three groups; the training set, the validation set and the test set. The training set is used to train the weights and biases of the network in order to learn the patterns of this data. The validation set is separate from the training set and gets used during training to validate the model performance for every epoch. The test set is a separate set from both of the other data sets and gets used to test and evaluate the model after completing all training [41]. A problem using this distinction between three data sets is that it may cause data with useful information to be excluded from the training, or that the system may be biased [42]. This problem mainly comes into play when there is limited data or when the test set is not large enough [43].

3.5.2 Cross-validation

Cross-validation (CV), also known as K-Fold Cross-validation, is used to overcome this problem. With K-Fold CV the parameter K is introduced, which represents the number of folds that the concatenated training and validation data is divided into [42] [43]. The model is then trained on $K-1$ parts, and uses the last part as its validation set. The training happens for K iteration, and each iteration a different set is the validation data, as is illustrated in figure 3.8.

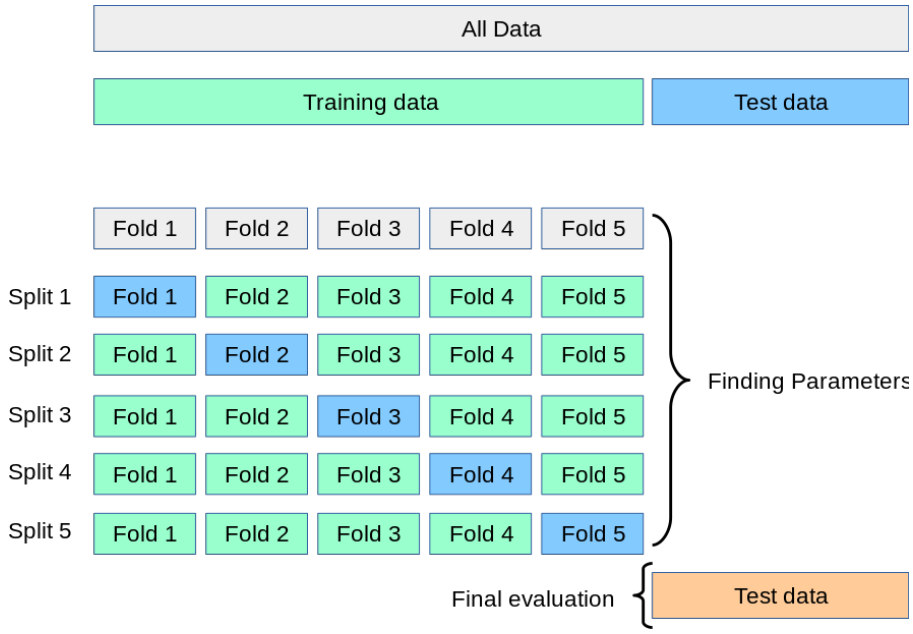


Figure 3.8: K-fold cross validation uses $K-1$ folds for training and the other for validation, while still keeping apart a test set for the final evaluation. In this image, K is 5. [44]

Materials and methods

4.1 Data collection

This study utilizes MRI scans from women who participated in the EPPA study. The participants provided written informed consent and the study was approved by local ethics committees (NL74061.091.20). The study included 15 patients from the gynecology department of Ziekenhuis Groep Twente (ZGT) hospital in Hengelo and 45 control women who were asymptomatic for POP who were recruited via advertisements. The control group consisted of 15 nulliparous women, 15 parous pre-menopausal women (with at least one vaginal delivery), and 15 parous post-menopausal women. Participants had to be at least 18 years old, be able to stand for 20 minutes without assistance and pass the MRI safety checklist. Women who had a jeans size larger than 52 (EU) or 22 (US) were excluded due to limited coil circumference [45].

4.2 Image acquisition

All upright scans were made with a tiltable 0.25T MR scanner (G-Scan; Esaote, Genoa, Italy), angled at 81° to enable a natural standing position. A multi-slice midsagittal 2D T2-weighted fast spin echo (FSE) scan was obtained with the following parameters: 2 ms echo time (TE), 3480 ms repetition time (TR), 1.3 × 1.3 mm^2 reconstructed resolution, 340 × 340 mm^2 field of view (FOV), 192 × 200 matrix size, 5 mm slice thickness, 11 slices, and a total scan time of approximately 2 minutes [46]. In addition, a 3D hybrid contrast enhancement (HYCE) MRI scan was done in the midsagittal position. Each woman was scanned three times in one day, with the first scan between 8 AM and 9 AM, the second scan between 12 PM and 1 PM, and the third scan between 4 PM and 5 PM. This resulted in a data set of 180 scans, from which 9 scans were excluded because some people fainted during the upright acquisition or because they were in the control group but showed a prolapse on the MRI. The final data set has 171 scans from 43 patients; 15 from prolapse patients and 28 from the control group.

4.3 Data labeling

The LP shape analysis of the upright scans was performed using ImageJ software (version 1.53q, LOCI, University of Wisconsin). For each scan twelve points were allocated as defined by Schmidt et al. [8] and as seen in figure 3.2. These have been manually annotated by a single observer and checked by a second observer for agreement. The anatomical landmarks that were assigned are the inferior pubic point of the pubic symphysis (1), the perineal body (2), the most superior point of the external anal sphincter (3), the middle of the puborectalis bundle (5) which is approximately the shortest distance from the pubic symphysis to the LP, the inferior coccyx (9), and the sacrococcygeal joint (10). The remaining points were equal sampling points (4, 6, 7, 8) at approximately half the distance between the above-mentioned anatomic landmarks. The LP is defined as a curved line between points (3) to (10). Point (11) was placed at the anterior and point (12) at the posterior fornix of the vagina. The coordinates of these points were analysed and saved in a .csv file.

4.4 Data preperation

The MRI data was presented in a .tiff file and the slice number and coordinates of the anatomical landmarks were presented in a .csv file. Since the CNN takes square images as input data, both the MRI slices and the mask coordinates had to be translated into PNG's. An example of a MR image together with its mask is presented in figure 4.1. The data was split into a test and train data set, where five subjects were selected for the test data set; one from each control group and two POP patients. This resulted in a train data set of 156 scans and a test data set of 15 scans. The train data set was divided into five for each iteration of the K-fold cross validation, which resulted in either 31 or 32 scans in the validation set.

4.4.1 MR images

The preprocessing of the MR images was done with self-developed code (Python (3.10.10)). The images were cropped to fit the region of interest, since the original images had more information stored than needed and cropping decreases the training time. This resulted in a working area of 250 × 250 voxels, which corresponds to

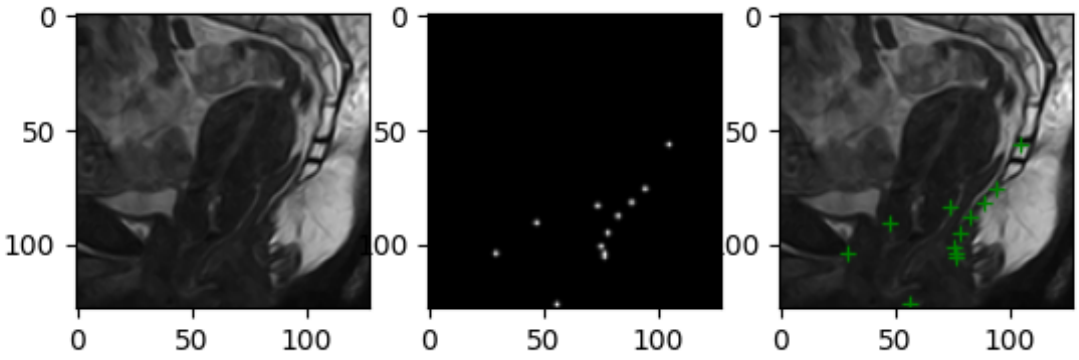


Figure 4.1: From left to right; a midsaggital MRI slice that was selected to do the LP shape analysis on, its mask with the selected coordinates convoluted with a Gaussian kernel, and the overlay of the MRI slice and its mask coordinates.

122.1 mm^2. Each pixel had a value between 0 and 255, as it was stored as an 8-bit integer. To process this, all values were divided by 255 since the model takes values between 0 and 1. It was chosen to resize all images to an array of $128 \times 128 \times 1$, as this seemed a good balance between keeping valuable information of the pictures, but getting an adequate training time.

4.4.2 Mask images

The coordinates provided in the .csv file were given in mm, so had to be scaled to pixels to fit over the MR slice. Each coordinate was convoluted with a Gaussian kernel ($\sigma = 0.5$) in order to formulate the landmark detection as a probability distribution heatmap. All separate points were visualized and saved as masks by the script in appendix B. The masks were cropped, reshaped and resized in the same way as the MR images. The masks were stored into an array of $128 \times 128 \times 12$, with each slice in the last dimension representing an individual point.

4.5 Model development and training

The final model architecture was already presented in figure 3.7. The python script uses Tensorflow (2.7.0) and Keras (2.7.0), and is based on the code of Sreenivas B. [47]. Multiple optimizers were looked into through literature [48] [49] [50] and Adam was selected. The initial learning rate α was varied and examined (figure 4.2). It was selected to be 0.0001 because of the stable descent through the epochs, and the expectancy to converge to the same minimum as the other learning rates. The other parameters of Adam, $\beta 1$, $\beta 2$ and ϵ, were not changed and have a TensorFlow Keras default value of 0.9, 0.999, and 10^{-8} respectively. Mean squared error (MSE) was chosen as the loss function after comparison with the loss functions accuracy and dice loss. The MSE is calculated as:

$$\sum_{i=1}^{D}(mask_i - result_i)^2 \tag{4.1}$$

where $mask_i$ and $result_i$ are the desired and model-found pixel values, respectively.
The model was fit over the training data with a validation split of 20% of the training data and a batch size of 16, and was run for 5000 epochs. It was saved at the epoch with best performance, so the smallest mean squared error. The training was done on the GPU of the University of Twente using JupyterLab. In the end, a threshold was implemented so that all values above 0.999 are converted to ones and all values equal to or below 0.999 are converted to zeros. For visualization, the different layers of the mask were combined and overlayed with the input MR image for comparison.

4.6 Evaluation of the model

4.6.1 Euclidean distance

To compare the model results with the input data, the euclidean distance was calculated. This is defined as the absolute distance between two points, and is mathematically described as:

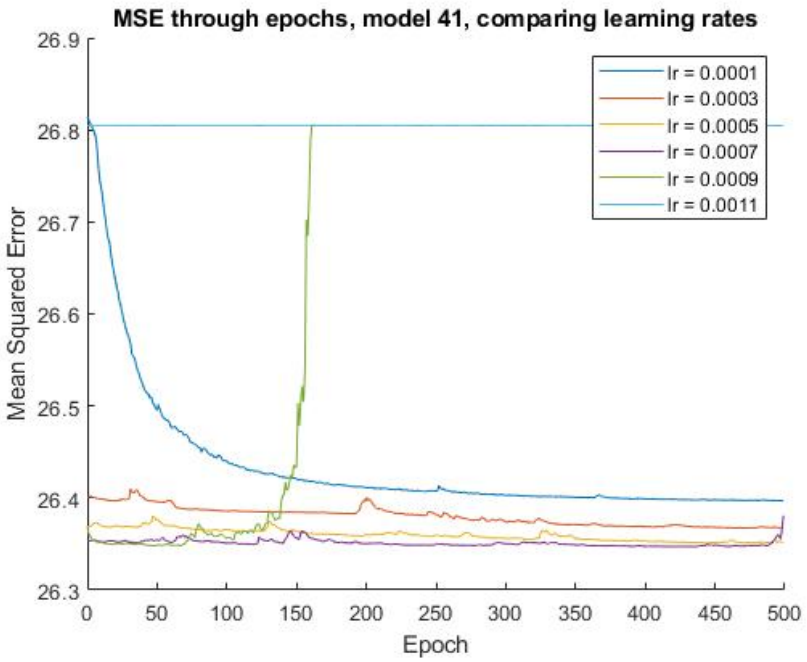


Figure 4.2: Comparing different initial learning rates for optimizer Adam through 500 epochs. For the model it was chosen to use an initial learning rate of 0.0001 (darker blue) because of the stable descent of the graph.

$$d = \sqrt{(x_{mask} - x_{model})^2 + (y_{mask} - y_{model})^2} \tag{4.2}$$

Since each layer of the mask represents its own anatomical landmark, the euclidean distance was calculated for each point individually.

4.6.2 Cross validation

K-fold cross validation was applied as it was described earlier, with a K of 5. For each iteration of the script, the mean squared error was computed, the total number of found points was counted and the euclidean distances were calculated.

Results

All analysis in the results is done on the five subjects in the test data set, which each had 3 scans. This leads to a total of 15 scans, and with 12 points located in each scan, this results in a total of 180 points.

5.1 Visual analysis

To assess whether or not the model works, the different slices of the predicted mask were overlayed and compared to the original mask. This comparison of the raw data can be found in appendix A. In order to provide a better comparison between the coordinates that were found in the different layers of the mask, each layers average coordinate was calculated and plotted together with the original mask coordinates, and plotted over the MR image (figure 5.1). To confirm that the found model points are in the correct slice, each anatomical landmark has been marked with its own colour in figure 5.2, with its assigned euclidean distance in the same colour. In this image, point 11 and 12 were not found by the model.

5.2 Quantitative analysis

5.2.1 Comparison between model folds

The five folds of the model done with K-fold cross validation each resulted in a mean squared error (MSE), average euclidean distance (AED) and number of found points (n). These results are summarized in table 5.1. An analysis of the descent of the MSE through the epochs per model can be found in appendix B. As can be seen in table 5.1, all models together had an average MSE of 6.2 and an AED of 7.0 mm. The maximum possible number of found points is 180 (15 scans with 12 landmarks each), and on average the models found 114.4 points which is 63.6% of the total.

Table 5.1: Comparing the mean squared error (MSE), average euclidean distance (AED) [mm] and the number of points found by each model fold (n).

Fold	MSE	AED [mm]	n (max 180)
Mod 1	6.27339	7.7	97
Mod 2	6.199474	6.3	117
Mod 3	6.137434	8.1	127
Mod 4	6.16159	6.7	105
Mod 5	6.206108	6.1	126
Average	6.195599	7.0	114.4

5.2.2 Comparison between landmarks

The AED and number of found points per anatomical landmark is presented in table 5.2. As can be seen, landmark 1 has the lowest AED (2.0 mm) whereas landmark 11 showed the highest AED (20.7 mm). The landmarks corresponding to the LP (3 to 10) had an AED of 6.1 mm. Additionally, table 5.2 shows a high variance in the number of found points per landmark, so for better comparison this has also been presented in figure 5.3. This figure also presents the variability per model fold for each landmark. The maximum number of found points per landmark is 75 (15 different scans over 5 folds). Anatomical landmarks 2, 11 and 12 were detected in less than 25% of the scans (n = 12 and 9 respectively). The most detected landmarks are 3, 4, 5, 6, 7, 8, and 10, as these were detected in more than 75% of the scans (n = 60, 64, 59, 70, 68, 60 and 63 respectively). Figure 5.3 also shows that model fold 1 did not find any anatomical landmarks 1 and 2. In order to quantify these results further, the euclidean distances per landmark are presented in a boxplot in figure 5.4. This shows a large interquartile range for the landmarks 10 and 11 (15.1 mm and 20.9 mm respectively) as well as a high median for the landmarks 10, 11 and 12 (9.4 mm, 23.4 mm and 9.8 mm respectively).

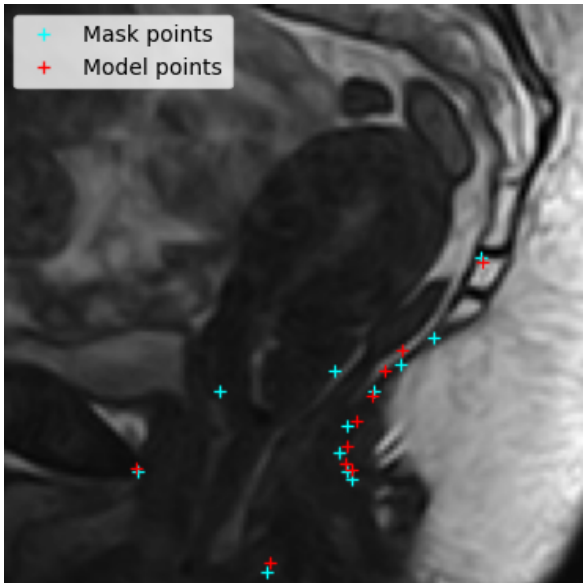


Figure 5.1: The points found by the model (red) and the mask points (blue) overlayed on the MR image.

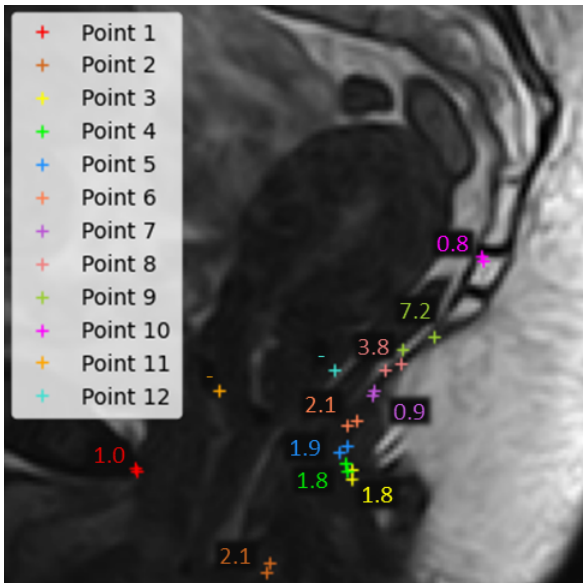


Figure 5.2: Comparison per anatomical landmark (1-12), with the euclidean distance [mm] related to it. Point 11 and 12 were not found by the model and therefore have the label "-".

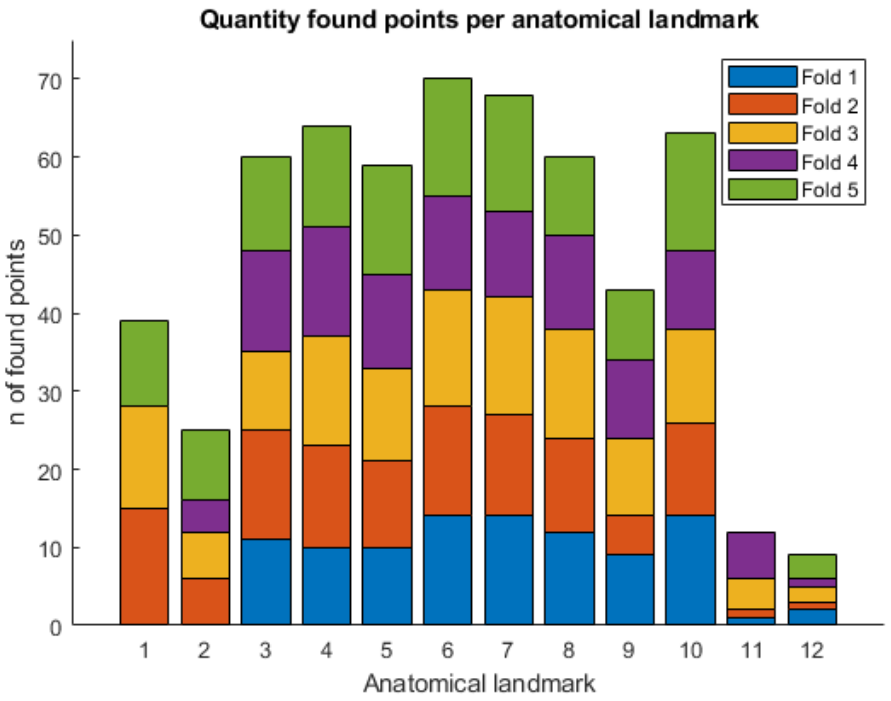


Figure 5.3: Number of found points per anatomical landmark. The maximum possible number is 75 (5 model folds for 15 scans). Model fold 1 did not find any points for landmarks 1 and 2, model 4 did not find any for landmark 1 and model 5 did not find any for landmark 11.

Table 5.2: Comparing the average euclidean distance (AED) and number of found points (n) between the different anatomical landmarks.

Landmark	AED [mm]	n (max 75)
1	2.0	39
2	6.4	25
3	5.3	60
4	6.3	64
5	6.5	59
6	5.7	70
7	5.6	68
8	6.3	60
9	7.9	43
10	11.1	63
11	20.7	12
12	12.8	9

Table 5.3: Comparing the average euclidean distance (AED) and number of points (n) found between the different subjects and their scans.

Subject	AED [mm]	n (max 180)	Scan	Avg ED [mm]	n (max 60)
Control 1 (nonparous)	6.3	92	1	3.0	42
			2	7.4	32
			3	10.4	18
Control 2 (parous, pre)	3.4	136	1	2.5	46
			2	3.7	41
			3	3.7	49
Control 3 (parous, post)	8.0	101	1	7.5	34
			2	5.6	36
			3	10.1	31
Patient 1 (POP)	7.2	129	1	6.3	42
			2	6.4	44
			3	8.0	43
Patient 2 (POP)	11.4	114	1	13.7	39
			2	9.9	37
			3	9.0	38

5.2.3 Comparison between subjects

For a complete evaluation of the model, the AED per scan has been calculated. This can be found in table 5.1, together with the number of found points per scan. This shows a difference in AED the different subjects, but not mainly between the POP patients and the control group. For the number of found points there is also no apparent difference between the control group and the POP patients. The maximum possible number of found points is 60 (5 model folds with 12 anatomical landmarks), and the lowest number of found points is in scan 3 of control 1 (n = 18). Analysis of the boxplot of the euclidean distances (figure 5.5) showed a variance between the different subjects and scans. Control 1 scan 3 has the largest interquartile range of 18.1. Patient 2 shows a high interquartile range for scan 1 and 2 (9.1 and 10.5 respectively) and a high median for all three scans (15.3, 12.0 and 9.6 respectively).

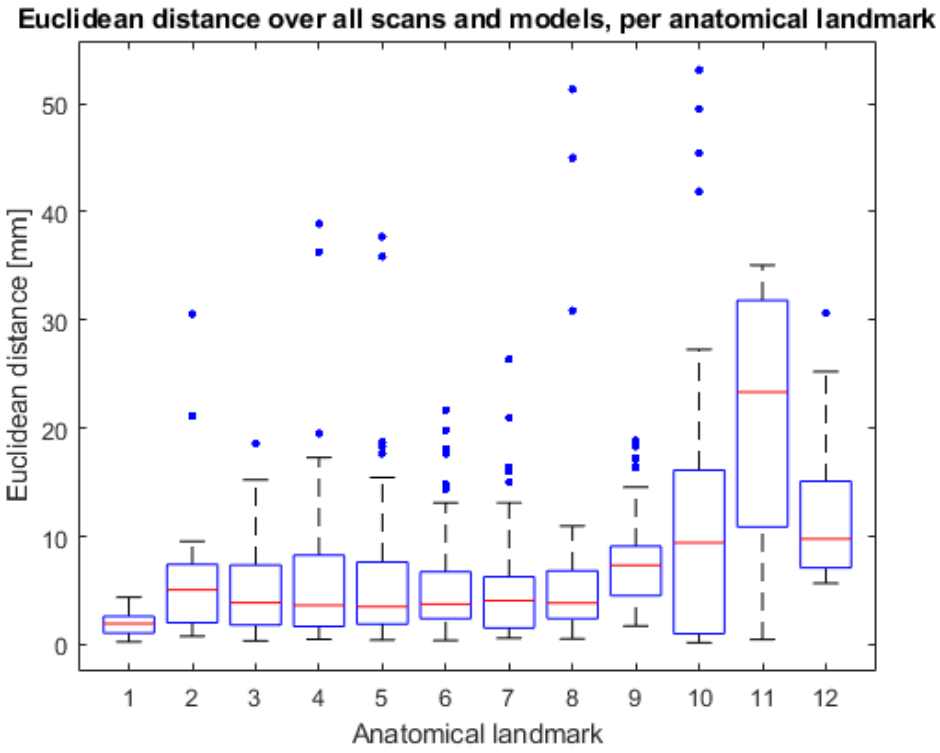


Figure 5.4: Boxplot comparing the euclidean distance between the assigned anatomical landmarks. This shows a large interquartile range for landmarks 10 and 11 (15.1 mm and 20.9 mm respectively) and a high median for the landmarks 10, 11 and 12 (9.4 mm, 23.4 mm and 9.8 mm respectively).

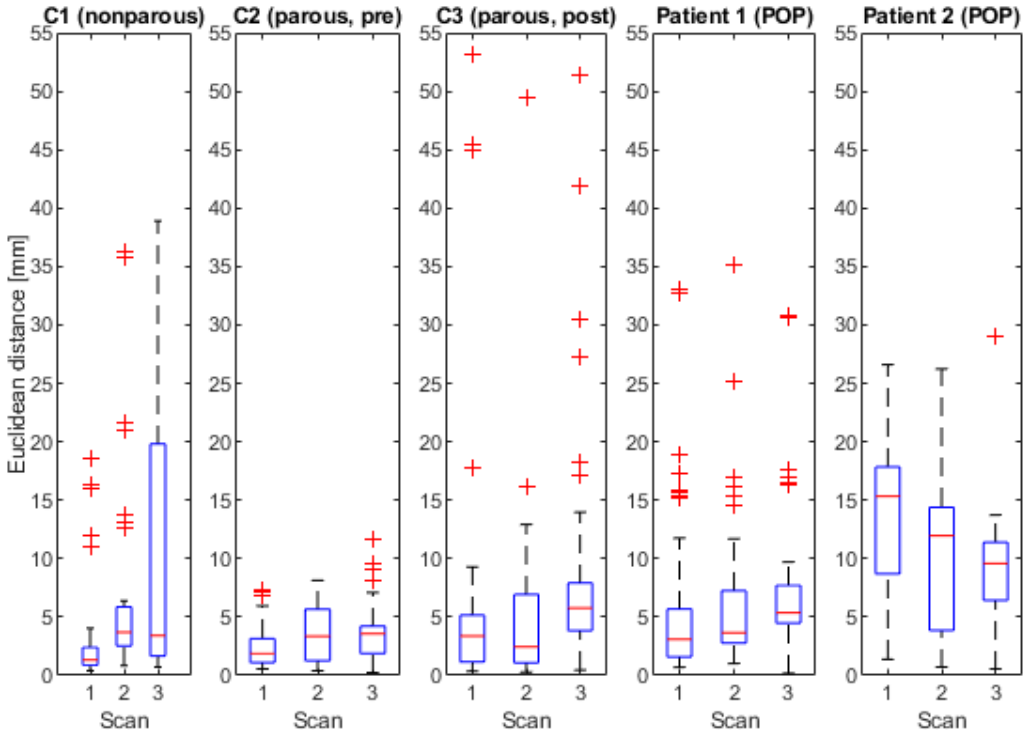


Figure 5.5: Boxplot comparing the different subjects and their scans. A large interquartile range is found in scan 3 from control 1 (18.1) and in scan 1 and 2 from POP patient 2 (9.1 and 10.5 respectively). Patient 2 also shows a high median for all scans (15.3, 12.0 and 9.6 respectively)

Discussion

This study shows a CNN based model for the detection of anatomical landmarks associated with the LP shape. K-fold CV was applied for 5 folds, and showed an average MSE of 6.2, an AED of 7.0 mm and an average found points of 114.4 points per model, which corresponds to 63.6% of all points. The model folds did not perform the same for each landmark, as fold 1 did not detect landmark 1 and 2, fold 4 did also not detect landmark 1 and fold 5 did not detect landmark 11. Over all model folds, landmark 1 and landmarks 3 through 10 were detected in high rates (more than 50% of the scans), with landmark 6 being detected the most (70 times out of 75 scans). Landmark 1 corresponds to the inferior point of the pubic symphysis and landmark 3 through 10 represent the LP. For the analysis of the LP shape with PCA [8] [17], the PICS line (the horizontal line through landmark 1) and landmarks 3 to 10 were used. This means that with the developed model, the anatomical landmarks for the analysis of the LP shape are highly detectable. The results showed different euclidean distances and number of detected points between different scans and patients, but did not show a significant difference between the control group and POP patients. This means that the developed model could be used for both of these groups and could possibly be used to analyse the difference between them.

Previous research has shown landmark detection of medical images based on a neural network for the analysis of CT and X-ray images of the prostate and craniofacial area [51] [52], as well as for MR images of brain structures and cardiac images [53] [51] [54]. Research has also shown that it is possible to detect landmarks by using a heatmap [55] [56] [54]. This study shows results that are in agreement with this, as well as that it shows that landmark detection in the pelvic area related to the LP shape is possible. A few of these researches included a distance error in their findings which could be compared to this research. The study of [53] showed a 3D localisation error of 0.79 mm with a processing field of $256 \times 26 \times 150$ mm^3, and the study of [54] showed an AED of 1.82-3.78 mm with a working field of 400 mm^2. This studies shows a significantly larger AED of 7.0 mm, with a working field of 122.1 mm^2. An explanation for this could be the limited amount of test and training data, since [53] uses 1128 annotated volumes and [54] uses 34 089 training images and 7723 test images, whereas this study only comprised 156 training images and 15 test images. Further data processing of the MSE would be necessary to adequately compare this research to other research, as most of the comparable studies use a root mean squared error or a normalized (root) mean squared error. The MSE is also highly dependent on the working field it is computed from, and it was already mentioned that this differs per studies.

The strength of this research is the application of a CNN to detect landmarks in the pelvic area, which has not been done before. Most importantly, anatomical landmarks 3 to 10 which correspond to the LP, have been detected which could be used to compute its shape. The cross validation method adds to this by showing that the network would work for different train and validation sets, which resulted in comparable results for the MSE, AED and number of found points.

A few limitations may have contributed to inferior performance of the network. Firstly, to reduce the training time and needed processing power, all images were resized to 128×128 voxels, while it originally was 250×250 voxels. This means the resolution was reduced by almost 75%. Another general problem is acquiring enough labeled training data. It was already mentioned before that studies in the same research field have much larger data sets to train and test their models with. It is expected that with more training data, the difference between the model cross validation folds would be minimized as the higher number of data would average out irregular cases. The current differences between the model folds could be related to the difference in training and validation data for each split, as K may have been too small for the data sets to be statistically representative for the broader data set. Furthermore, a larger test data set would make it possible to show the difference between the groups of subjects better, as there were four types of groups and there were only five subjects in the test set. Another limitation is that the data labeling has been done only once, and has not been done by a medical professional. A labeling by a second reader would be required to measure interreader variation, which could average the mask coordinates to a value that is as close as possible to its true value. This could also present the difference between readers for different anatomical landmarks, and place the model results in a broader context. One of the main limitations of this research though, was time. The person developing this model had no prior knowledge of deep learning, neural networks and landmark detection, and the project had to be done within ten weeks. Consequently there was limited time to experiment with some of the parameters of the model, which with more time could have been examined further.

Future research could look into the development of a model with only the landmarks used for PCA, in order to eliminate any influence the other landmarks could have on the results and hopefully learn the correlation

between the points accurately. Furthermore, future research could look into different types of cross validation, to experiment with what type would be the best training and representation for the model. As was already briefly mentioned before, more labeled training data would be desired to experiment with a different number of cross validation splits. It was also mentioned before that more test data would be required to see the correlation between the groups of subjects and the model performance for them. Subsequently, future work could look into applying PCA to the results of the model and look at the difference between the control groups and POP patients, to see if the model is able to correctly determine the shape difference between them. Something else that could be looked into is the preprocessing of image data with image enhancement algorithms, as these may improve the performance of CNN models [57]. Another region that could be explored is the comparison between landmarks as a coordinate on its own compared to landmarks in a segmentation map, as landmarks in a segmentation map may not give the best results. Lastly, a loss function that combines the euclidean distance with the mean squared error could be explored to hopefully reduce the euclidean distance further.

To conclude, the model that is developed in this study shows a U-net based structure for the detection of landmarks related to the LP, that could be used in research around pelvic organ prolapse. Some anatomical landmarks were much better detectable than others, with the landmarks corresponding to the inferior point of the pubic symphysis and the LP being highly detectable with a relatively small euclidean distance. The model appeared to not perform different for POP-patients and non-patients. To improve the model, much more training and test data would be required, but for now the developed model can detect the anatomical landmarks associated with the LP shape and could be the basis for a model that could be implemented clinically.

Bibliography

[1] *Want to know how deep learning works? here's a quick guide for everyone.* Mar. 2020. URL: https://www.freecodecamp.org/news/want-to-know-how-deep-learning-works-heres-a-quick-guide-for-everyone-1aedeca88076/ (visited on 10/04/2023).

[2] Wang B et al. "Global burden and trends of pelvic organ prolapse associated with aging women: An observational trend study from 1990 to 2019". In: *Front Public Health* (2022). DOI: 10.3389/fpubh.2022.975829.

[3] *Overview - Pelvic organ prolapse.* Mar. 2021. URL: https://www.nhs.uk/conditions/pelvic-organ-prolapse/#:~:text=Pelvic%20organ%20prolapse%20is%20when,can%20cause%20pain%20and%20discomfort (visited on 10/04/2023).

[4] Richard C Bump et al. "The standardization of terminology of female pelvic organ prolapse and pelvic floor dysfunction". In: *American journal of obstetrics and gynecology* 175.1 (1996), pp. 10–17.

[5] *Pelvic organ prolapse quantification (pop-Q) system.* URL: https://www.physio-pedia.com/Pelvic_Organ_Prolapse_Quantification_(POP-Q)_System (visited on 10/04/2023).

[6] Anique TM Grob et al. "Underestimation of pelvic organ prolapse in the supine straining position, based on magnetic resonance imaging findings". In: *International Urogynecology Journal* 30 (2019), pp. 1939–1944.

[7] Yvonne Hsu et al. "Levator plate angle in women with pelvic organ prolapse compared to women with normal support using dynamic MR imaging". In: *American journal of obstetrics and gynecology* 194.5 (2006), pp. 1427–1433.

[8] Payton Schmidt et al. "Preoperative level II/III MRI measures predicting long-term prolapse recurrence after native tissue repair". In: *International Urogynecology Journal* 33.1 (2022), pp. 133–141.

[9] Mingyu Kim et al. "Deep learning in medical imaging". In: *Neurospine* 16.4 (2019), p. 657.

[10] Cheryl B Iglesia and Katelyn R Smithling. "Pelvic organ prolapse". In: *American family physician* 96.3 (2017), pp. 179–185.

[11] Idris Akanji Ayantoye. *The Role of Diabetes Mellitus in Collagen Disorder Associated with Pelvic Organ Prolapse.* Illinois Institute of Technology, 2019.

[12] Jennifer M Wu et al. "Forecasting the prevalence of pelvic floor disorders in US Women: 2010 to 2050". In: *Obstetrics & Gynecology* 114.6 (2009), pp. 1278–1283.

[13] John OL DeLancey. "What's new in the functional anatomy of pelvic organ prolapse?" In: *Current opinion in obstetrics & gynecology* 28.5 (2016), p. 420.

[14] Sender Herschorn. "Female pelvic floor anatomy: the pelvic floor, supporting structures, and pelvic organs". In: *Reviews in urology* 6.Suppl 5 (2004), S2.

[15] Angelique Montano-Bresolin. *Levator ani syndrome: One way to name the pain.* Dec. 2022. URL: https://www.proactiveph.com/blog/levator-ani-syndrome-one-way-to-name-the-pain/ (visited on 10/04/2023).

[16] Mary Duarte Thibault et al. "A comparison of MRI-based pelvic floor support measures between young and old women with prolapse". In: *International Urogynecology Journal* (2023), pp. 1–8.

[17] M de Vries and Frieda van den Noort. "Assessment of pelvic organ prolapse extend during the day using PCA, An MRI study". In: (Aug. 2022).

[18] *What is deep learning?: How it works, techniques amp; applications.* URL: https://nl.mathworks.com/discovery/deep-learning.html (visited on 10/04/2023).

[19] *What is deep learning?* URL: https://www.ibm.com/topics/deep-learning (visited on 10/04/2023).

[20] *What are neural networks?* URL: https://www.ibm.com/topics/neural-networks (visited on 10/04/2023).

[21] Sagar Sharma. *Activation functions in neural networks.* Nov. 2022. URL: https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6 (visited on 10/04/2023).

[22] *Activation functions in neural networks.* Feb. 2023. URL: https://www.geeksforgeeks.org/activation-functions-neural-networks/ (visited on 10/04/2023).

[23] Sagar Sharma. *Activation functions in neural networks.* Nov. 2022. URL: https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6 (visited on 10/04/2023).

[24] Shadi Karazi, Mahmoud Moradi, and Khaled Benyounis. "Statistical and numerical approaches for modelling and optimising laser micromachining process-Review". In: *Reference Module in Materials Science and Materials Engineering* (2019).

[25] Stephen Allwright. *Loss function vs cost function, what's the difference?* Dec. 2022. URL: https://stephenallwright.com/loss-function-vs-cost-function/ (visited on 10/04/2023).

[26] Simeon Kostadinov. *Understanding backpropagation algorithm.* Aug. 2019. URL: https://towardsdatascience.com/understanding-backpropagation-algorithm-7bb3aa2f95fd (visited on 10/04/2023).

[27] Baeldung. *Epoch in neural networks.* Mar. 2023. URL: https://www.baeldung.com/cs/epoch-neural-networks (visited on 10/04/2023).

[28] Rekha M. *The ascent of gradient descent.* June 2020. URL: https://blog.clairvoyantsoft.com/the-ascent-of-gradient-descent-23356390836f (visited on 10/04/2023).

[29] Ayush Gupta. *A comprehensive guide on Optimizers in deep learning.* Mar. 2023. URL: https://www.analyticsvidhya.com/blog/2021/10/a-comprehensive-guide-on-deep-learning-optimizers/ (visited on 10/04/2023).

[30] Lu Lu et al. "Dying relu and initialization: Theory and numerical examples". In: *arXiv preprint arXiv:1903.06733* (2019).

[31] Jeremy Jordan. *Setting the learning rate of your neural network.* Mar. 2023. URL: https://www.jeremyjordan.me/nn-learning-rate/ (visited on 10/04/2023).

[32] Ritchie Ng. *Learning rate scheduling.* URL: https://www.deeplearningwizard.com/deep_learning/boosting_models_pytorch/lr_scheduling/ (visited on 10/04/2023).

[33] *Image recognition with deep neural networks and its use cases.* Jan. 2020. URL: https://www.altexsoft.com/blog/image-recognition-neural-networks-use-cases/ (visited on 10/04/2023).

[34] *What are convolutional neural networks?* URL: https://www.ibm.com/topics/convolutional-neural-networks (visited on 10/04/2023).

[35] Sinam Ajitkumar Singh, Takhellambam Gautam Meitei, and Swanirbhar Majumder. "Short PCG classification based on deep learning". In: *Deep Learning Techniques for Biomedical and Health Informatics.* Elsevier, 2020, pp. 141–164.

[36] *Discrete convolution with a 3x3 kernel.* URL: https://www.researchgate.net/figure/Discrete-convolution-with-a-3x3-kernel_fig3_335609766 (visited on 10/04/2023).

[37] *The difference between the CNN and FCN.* URL: https://www.researchgate.net/figure/The-difference-between-the-CNN-and-FCN-the-transforming-of-fully-connected-layers-into_fig15_341403564 (visited on 10/04/2023).

[38] Christian F Baumgartner, Ozan Oktay, and Daniel Rueckert. "Fully convolutional networks in medical imaging: Applications to image enhancement and recognition". In: *Deep Learning and Convolutional Neural Networks for Medical Image Computing: Precision Medicine, High Performance and Large-Scale Datasets* (2017), pp. 159–179.

[39] *What is U-Net?* URL: https://www.educative.io/answers/what-is-u-net (visited on 10/04/2023).

[40] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. "U-net: Convolutional networks for biomedical image segmentation". In: *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*. Springer. 2015, pp. 234–241.

[41] *Train test validation split: How to amp; best practices [2023].* URL: https://www.v7labs.com/blog/train-validation-test-set#h1 (visited on 10/04/2023).

[42] Siladittya Manna. *K-fold cross validation for deep learning using keras.* June 2020. URL: https://medium.com/the-owl/k-fold-cross-validation-in-keras-3ec4a3a00538 (visited on 10/04/2023).

[43] Rebecca Patro. *Cross validation: K Fold vs Monte Carlo.* Feb. 2021. URL: https://towardsdatascience.com/cross-validation-k-fold-vs-monte-carlo-e54df2fc179b (visited on 10/04/2023).

[44] *Cross-validation: Evaluating estimator performance.* URL: https://scikit-learn.org/stable/modules/cross_validation.html (visited on 10/04/2023).

[45] Liselot J De Kruif et al. "Differences in Levator Plate shape in supine and upright position - A magnetic resonance imaging study". In: (2022).

[46] Lisan M Morsinkhof et al. "Pelvic inclination correction system for magnetic resonance imaging analysis of pelvic organ prolapse in upright position". In: *International Urogynecology Journal* 33.10 (2022), pp. 2801–2807.

[47] *Digitalsreeni*. URL: https://www.youtube.com/@DigitalSreeni (visited on 10/04/2023).

[48] Ayush Gupta. *A comprehensive guide on Optimizers in deep learning*. Mar. 2023. URL: https://www.analyticsvidhya.com/blog/2021/10/a-comprehensive-guide-on-deep-learning-optimizers/ (visited on 10/04/2023).

[49] Sanket Doshi. *Various optimization algorithms for training neural network*. Aug. 2020. URL: https://towardsdatascience.com/optimizers-for-training-neural-network-59450d71caf6 (visited on 10/04/2023).

[50] Jason Brownlee. *Gentle introduction to the adam optimization algorithm for deep learning*. Jan. 2021. URL: https://machinelearningmastery.com/adam-optimization-algorithm-for-deep-learning/ (visited on 10/04/2023).

[51] Jun Zhang, Mingxia Liu, and Dinggang Shen. "Detecting anatomical landmarks from limited medical imaging data using two-stage task-oriented deep neural networks". In: *IEEE Transactions on Image Processing* 26.10 (2017), pp. 4753–4764.

[52] Jiahong Qian et al. "CephaNet: An improved faster R-CNN for cephalometric landmark detection". In: *2019 IEEE 16th international symposium on biomedical imaging (ISBI 2019)*. IEEE. 2019, pp. 868–871.

[53] Christine A Edwards et al. "Deepnavnet: Automated landmark localization for neuronavigation". In: *Frontiers in Neuroscience* 15 (2021), p. 670287.

[54] Hui Xue et al. "Landmark detection in cardiac MRI by using a convolutional neural network". In: *Radiology: Artificial Intelligence* 3.5 (2021), e200197.

[55] Antonia Stern et al. "Heatmap-based 2d landmark detection with a varying number of landmarks". In: *Bildverarbeitung für die Medizin 2021: Proceedings, German Workshop on Medical Image Computing, Regensburg, March 7-9, 2021*. Springer. 2021, pp. 22–27.

[56] Jun Wan et al. "Precise Facial Landmark Detection by Reference Heatmap Transformer". In: *IEEE Transactions on Image Processing* (2023).

[57] Xiaoran Chen. *Image enhancement effect on the performance of convolutional neural networks*. 2019.

Appendices

7.1 Appendix A: Raw output image of the model

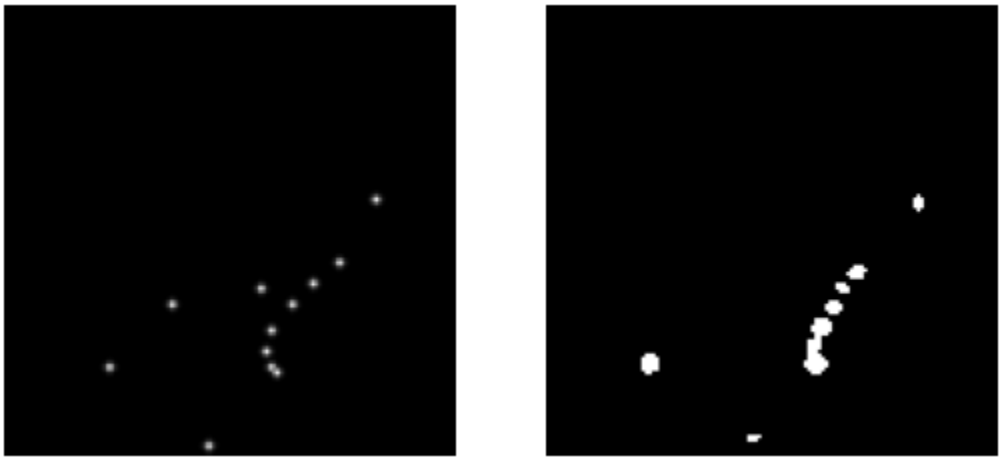


Figure 7.1: A comparison between the expected mask points (left) and the output of the model (right).

7.2 Appendix B: The MSE per model fold, through the epochs

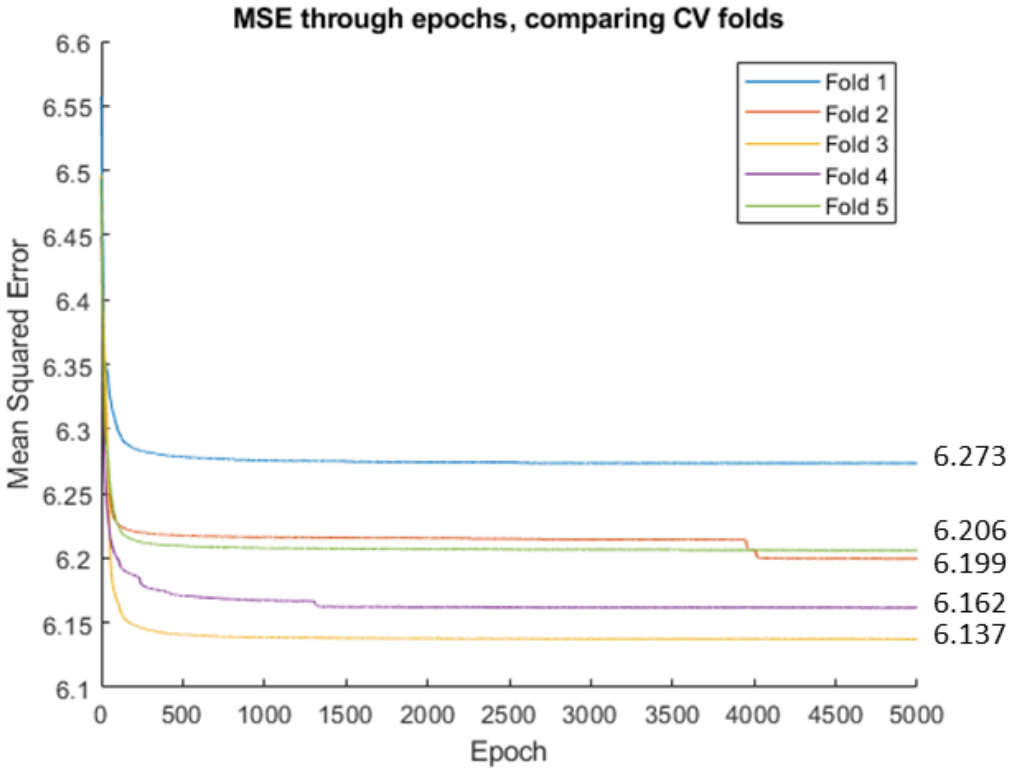


Figure 7.2: Comparing the descent of the MSE for each K-fold cross validation model fold. The final MSE value is displayed on the right of the graph.