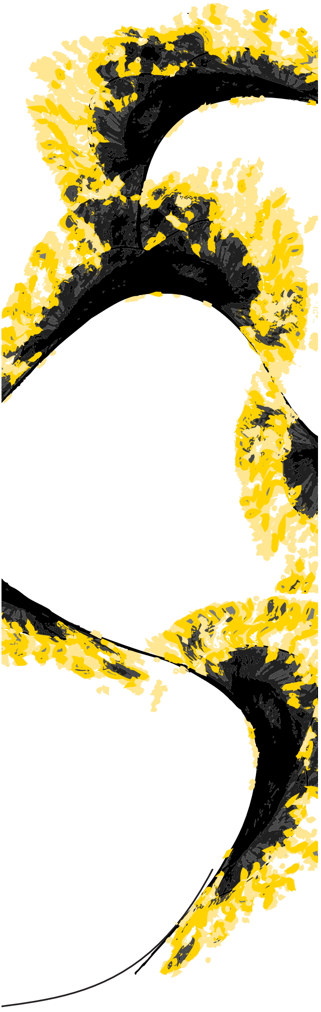
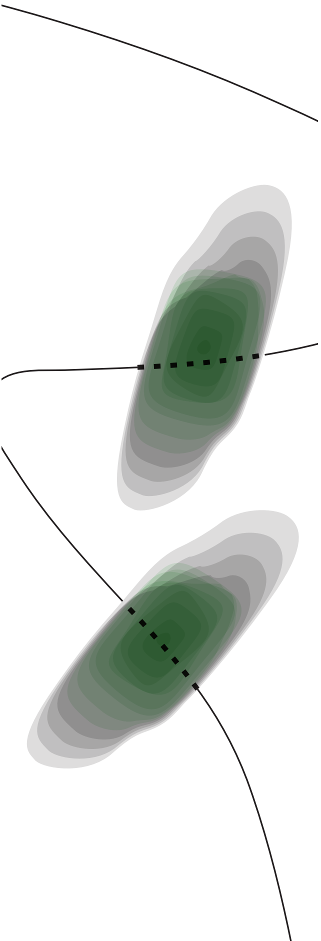


IDENTIFICATION OF PELVIC FLOOR FEATURES IN ULTRASOUND IMAGES USING AN UNSUPERVISED MACHINE LEARNING ALGORITHM

T.J. (Twan) Bongers

MSC ASSIGNMENT



Committee:
prof. dr. ir. C.H. Slump
F. Limbeek-van den Noort, MSc
prof. dr. ir. R.N.J. Veldhuis

September, 2019

038RaM2019
Robotics and Mechatronics
EEMathCS
University of Twente
P.O. Box 217
7500 AE Enschede
The Netherlands

Towards unsupervised learning for classification of ultrasound images

Twan Bongers

September 5, 2019

Summary

This research is focusing on the analyses of higher-dimensional data of the pelvic floor. Using an ultrasound probe in transperineal direction, it captures the pelvic floor and creates an accurate representation of the levator ani muscles where the pelvic floor is located. During the capturing of this video, the women were asked to perform three movements of contraction: contracted, Valsalva, and rest. These videos are captured when the women are 12 and 36 weeks pregnant and 6 months after delivery.

The aim is to create an unsupervised neural network, which reduces the high dimensional input data to two dimensions. This is done according to the sub-research questions: *"How can we use unsupervised learning to get the most relevant features using ultrasound images?"* and *"Do the most prominent features contain contain clinically relevant information?"*. Combining both sub-research questions, results in the main-research question: *"How do we obtain clinically relevant information from pelvic floor ultrasound images using unsupervised deep learning?"*

First, a 2D analysis is performed to validate the principle of using an unsupervised neural network. Form this neural network, the clinically relevant information of the pelvic floor is obtained by comparing the latent spaces of multiple women. The latent space is reduced by performing a PCA and t-SNE analysis to create two parameters describing an input image. These two parameters are plot on a scatterplot.
Second, the 3D input data is applied to an unsupervised neural network, which is based on the 2D network, to create a scatterplot of these input images.
The neural network creates clusters of similar frames. These frames are clustered together because these have similar input features. The most prominent input features are automatically detected by the network, and plotted in the scatterplot.

The research questions can be answered by analyzing the different states of contraction made by the women. It is possible to find a correlation between the different states of contraction and the frames within a single video. These different states of contraction can be used in further research to analyze the maximum contraction or Valsalva state.
The distinction between different time frames is difficult. This is due to the fact that the neural network is trained unsupervised and focuses on the most prominent features. The most prominent features are not clinically relevant. The variance between the women is more significant than the variance over time.

Acknowledgements

The period of research almost took a full year in combination with my Individual Assignment. During this year, I spent a lot of time with my daily supervisor Frieda. I would like to thank her for all the time, curious questions and feedback to my delivered work. The conversations in the "Coffee corner" with a cup of tea are always interesting and instructive.

A special thanks to both other members of my graduation committee; Kees Slump and Raymond Veldhuis. Both are investing a lot of time in me to ask curious questions about my report. These questions and feedback are increasing my writing skills and professional attitude.

Finally, I would like to thank the Robotics and Mechatronics (RaM) chair at the University of Twente for hosting me and their unlimited support of coffee. A special thanks for Jolanda, as she is always taking care of her students if they are her children.

Twan Bongers
Enschede, 03 September 2019

Acronyms

AI Artificial Intelligence. 6

CNN Convolutional Neural Networks. 10, 11, 24

IA Individual Assignment. 15–17

LAM Levator Ani Muscles. 3, 4

MAE Mean Absolute Error. 8

ML Machine Learning. 6

MRI Magnetic Resonance Imaging. 1, 3

MSE Mean Squared Error. 8, 24

NN Neural Network. 6, 24

PCA Princial Component Analysis. 12, 15, 17, 29, 33, 34

POP Pelvic Organic Prolapse. 3

PPM Poboperineal Muscle. 4

PRM Puborectal Muscle. 4, 5, 16, 25, 26, 39

ReLu Rectified Linear unit. 7

RGB Red Green Blue. 11

t-SNE t-Distributed Stochastic Neighbor Embedding. 12, 13, 15, 17, 29, 30, 33, 34

UMCU University Medical Centre Utrecht. 1

US Ultrasound. 3

Contents

1 Introduction

Childbirth is one of the most important and beautiful moments of life and this moment should be memorized as one of the happiest moment in life. However, the consequences of a childbirth can be disastrous. As a result of the childbirth, an organic prolapse or incontinence can continue to exist throughout life.

1.1 Problem description

The pelvic floor muscles are important for the support of the pelvic and abdominal organs. When a pregnant woman is going through childbirth, the risks of getting injured are present. During delivery, the baby's head passes through these muscles causing them to stretch up three times the original length [Lien et al., 2004], which may traumatize the muscles. This stretching is the main reason why one-fifth to one-third of all women will experience pelvic floor problems like pelvic organ prolapse and urinary or fecal incontinence during the rest of their lives [Dietz and Lanzarone, 2005, MacLennan et al., 2000].
After delivery, women with a vaginal delivery as well as a cesarean section are at higher risk for having incontinence. This suggests that the pelvic floor muscles and organs change during the period of pregnancy [Alperin et al., 2015, Dietz et al., 2004]. Women with vaginal childbirth have a higher chance of urinary or fecal incontinence compared to women with a cesarean section or women without ever being pregnant [MacLennan et al., 2000].
Using 3D/4D transperineal ultrasound, it is possible to visualize the muscles in the pelvic floor. The muscle and its movements can be recorded on a video. Currently, the muscle is analyzed by selecting 2D slices in some specific frames of those volume movies. 2D ultrasound images give a good estimation of the current state of the pelvic floor muscles, where all important sections are distinguishable [Grob et al., 2016]. Research about 3D transperineal ultrasound is not far advanced; this is remarkable because ultrasound is not harmful to patients or embryo[Lyons et al., 1988], and these images contain a large amount of clinically relevant information. Using an ultrasound imaging machine as well as using Magnetic Resonance Imaging (MRI) avulsions can be visualized using an extensive analysis [Otcenasek et al., 2007], where an avulsion can be described as an injury in which a body structure is torn of by either trauma or surgery.

1.2 Relevance

Looking to societal benefits instead of individual benefit, the prevention of avulsions and its consequences could be enormous. As within the Netherlands, the costs of health care is rising every year [Zorginstituut Nederland, 2018]. The capability of better detection and diagnosis will likely improve the quality of the therapy and will reduce the costs of unsuccessful therapies or wrong treatments. Improving the prevention of avulsions so that the negative consequences of childbirth can be reduced.

Besides the reduction of costs, the quality of life improves. For example, women with incontinence are currently very dependent on quick accessibility to toilets. When this is not possible for these women, they are limited in their movements. This improvement of quality will result in a happier society.

1.3 Context

This research is part of a research project led by the academic hospital: University Medical Centre Utrecht (UMCU). The goal of this research is to define new methods to find clinical relevant information of the pelvic floor automatically. These found results should help the doctor by making the correct decision of the current status of the pelvic floor. Based on this recommendation, and currently available information, allows the doctor to choose the right treatment.

The available dataset consists of transperineal ultrasound videos from pregnant women from 12 weeks, 36 weeks pregnant, and six months after delivery. Transperineal ultrasound is not harmful to patients and thus, it is possible to create a large dataset of ultrasound images from 258 women. These videos consist of grayscale 3D images. The women in the video are applying three different states of contraction: contracted, rest, and Valsalva. Taking a slice through the 3D frame, the doctor currently tries to estimate the status of the pelvic floor muscle. By designing a computer model to distinguish between different features within the pelvic floor, we can assist doctors for the detection of any avulsion.

1.4 Goal

Within the research of this master thesis, the goal is to analyze hidden features in ultrasound images using an unsupervised autoencoder neural network. These networks are built to encode images into a lower dimension data, latent space, from which this latent space can be decoded into a reconstructed ultrasound image. This latent space is a combination of features from the input image such that the output as accurate as possible can be reconstructed. During training the network is optimized to reconstruct the image as accurately as possible and thus the maximal amount of relevant information is stored in the lower dimensional latent space.
This latent space can be extracted from the neural network for the detection of any possible damage to the pelvic floor. Projected to a lower dimension, the detection of avulsion clusters can be done by currently known labels, but also similarly by the detection of new unique clusters. These new clusters can give new insights into different aspects from the pelvic floor. The found clusters can be analyzed by looking at the ultrasound images and find the relation between the clustered images.
This can be done according to the following research question:

How do we obtain clinically relevant information from pelvic floor ultrasound images using unsupervised deep learning?

Besides the research question, several sub-questions are made:

- *How can we use unsupervised learning to get the most relevant features using ultrasound images?*
- *Do the most prominent features contain clinically relevant information?*

1.5 Report outline

In chapter 2, relevant background information about the pelvic floor, different neural network characteristics and dimension reduction methods are described. This background information is used in chapter 3. A neural network is going to be trained to encode the Ultrasound input images into a latent space. This latent space describes every input image as a series of numbers within the n-dimensional vector. These series of numbers are analyzed using different dimension reduction methods where the visibility of contracted, rest, and Valsalva is examined. The current different states of contraction are used because the most variation within the images is captured within these actions.

In chapter 4, the 2D dataset is replaced by a 3D dataset. A similar approach as used for the 2D data is used for the 3D data. The three dimensional data is transformed into a latent space representation. The dimensions of the input images are reduces so that these can be plotted on a scatter plot. In this scatter plot, the clustering and distance between the frames is analyzed.
Chapters 5, 6, and 7 describe the discussion, conclusion, and recommendation of this research.

2 Background

This chapter contains relevant background information about the project. The pelvic floor and the usage of ultrasound are described in section 2.1. This section is followed by the description of artificial intelligence, section 2.2, and neural networks, section 2.3. From this neural network, obtaining a latent space representation, which is elaborated in section 2.4. Finally, the latent space is reduced in dimension, which is described by the dimension reduction methods as described in section 2.5.

2.1 Pelvic floor

The Levator Ani Muscles (LAM) is a muscle group that supports the lower abdomen. These muscles provide control over the emptying of the bladder and rectum. Damage or weakening of the LAM means that the internal organs, bladder, bowel, and uterus, are not fully supported. This can result in difficulties in controlling the release of urine, feces, or flatus [Hoyte and Damaser, 2016].

The pelvic floor muscle has three passages, the urethra, vagina, and anus, that to pass through the muscle. These muscles wrap firmly around these holes to help keep them shut and in place. When the pelvic floor muscles are contracted, this lifts the internal organs, and tightens the sphincters and opens the urethra, vagina, and anus. Relaxing the pelvic floor muscles allow passage of urine and feces.

When this support has become weaker or has some damage as a consequence of child-birth, the supported organs can prolapse and lead to a Pelvic Organic Prolapse (POP) [DeLancey et al., 2003].

2.1.1 Anatomy

This section describes the anatomy of the LAM, which then can be visualized by ultrasound as described in the next section. The anatomy of the LAM is described using a schematic view, see figure 2.1. Using methods such as Magnetic Resonance Imaging (MRI) or Ultrasound (US), the lower part of the body is visualized. Besides the techniques described before, physicians can enter the rectum or vagina to feel the structure and create an estimation of the condition of the pelvic floor.

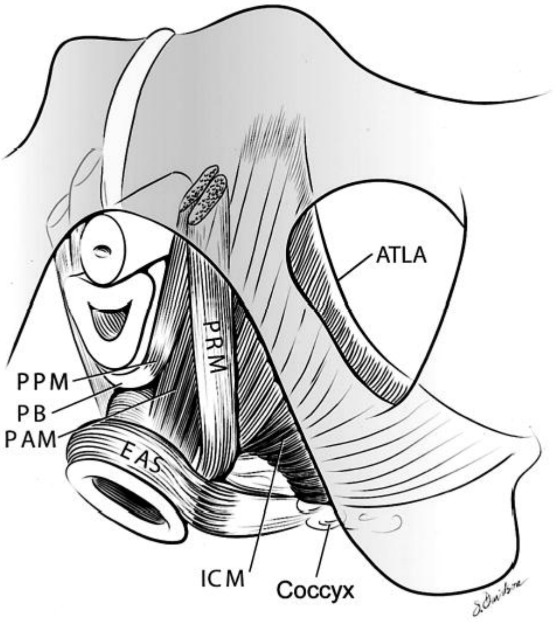


Figure 2.1: Schematic view of the levator ani muscles from below after the vulvar structures and perineal membrane has been removed showing the arcus tendineus of the levator ani (ATLA); external anal sphincter (EAS); puboanal muscle (PAM); perineal body (PB) uniting the 2 ends of the puboperineal muscle (PPM); iliococcygeal muscle (ICM); puborectal muscle (PRM). Note that the urethra and vagina have been transected just above the hymenal ring. [Kearney et al., 2004]

Currently, research is focusing on the Puborectal Muscle (PRM) and the Poboperineal Muscle (PPM). The schematic view of the pelvic floor as described in figure 2.1 gives a short explanation of the position and abbreviation of the LAM [Hoyte and Damaser, 2016]. The PRM and PPM muscles form a sling behind the rectum and vagina. The muscles can be contracted to create an elevation of the anus, perineal body, and vagina.

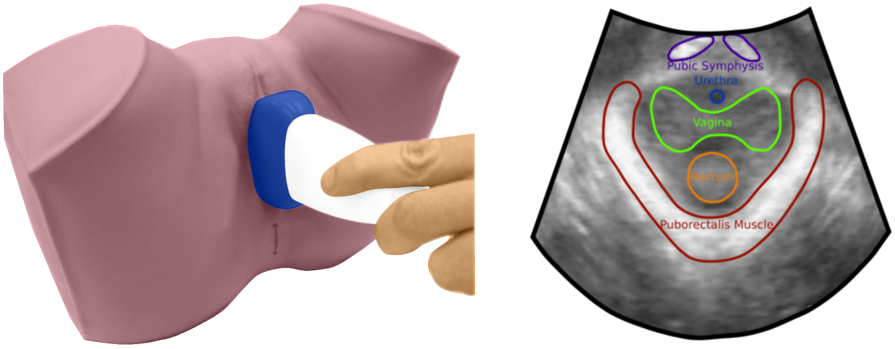
Within the available dataset, three muscle movements, contraction, rest, and Valsalva, are captured during the ultrasound video. These three states are achieved by tightening different muscles in the pelvic floor, and this can be captured with ultrasound. The muscle movements can give information about the pelvic floor and its muscles.
A correct pelvic floor muscle contraction has been described as an inward lift and squeezes around the urethra with resultant urethral closure, stabilization, and resistance downward movement [Bø et al., 2001, Delancey and Ashton-Miller, 2004]. Contraction can be obtained by contracting the puborectal muscle, in a similar way of holding up your feces. This is because the PRM is located around the anus and will tighten the anus. A Valsalva maneuver can be defined as the maximal strain effort, with forced expiration against a closed glottis that resulted in depression of the bladder base, which can be observed by using ultrasound [Thompson et al., 2006]. Valsalva can be obtained by increasing the pressure in the abdomen by holding your breath and pressing. The rest state can be defined as the state when a woman applies no strain or contraction in the pelvic floor, which is the normal state of the pelvic floor. Valsalva and contraction can be compared to the rest state to see the muscle movements.

2.1.2 Transperineal ultrasound

To visualize the specific components of the LAM, we use a transperineal ultrasound probe. The ultrasound images are taken by applying the ultrasound probe in a transperineal direction to the bottom of a woman, as shown in figure 2.2(a). The ultrasound probe generates a 3D image of the pelvic floor, from this high dimensional image, a 2D slice is taken to create a representation, as shown in figure 2.2(b). This 2D slice is taken so that the PRM is exactly in line with the

angle of the plane. Choosing the correct angle maximizes the visibility of the PRM and shows most of the PRM on one 2D image.
Besides the PRM, the pelvis is partly visualized by the pubic symphysis. Bones have a high reflection rate and thus appear white on ultrasound images. Underneath the pubic symphysis, the Urethra, vagina, and rectum are assigned.

Ultrasound devices can create a video of 3D pelvic floor images over time. Within this video, the functionality of the muscle can be visualized by inducing a contracting, Valsalva, or relaxing movement. Since the pelvic floor muscles have a higher collagen concentration than normal muscles they appear bright on ultrasound [Tuttle et al., 2014], as seen on figure 2.2 (b).

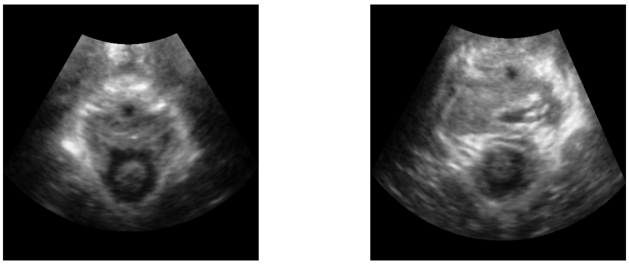


(a) (b)

Figure 2.2: (a) Describes the positioning of the ultrasound probe in Transperineal direction for capturing ultrasound images. (b) The 2D black and white ultrasound image with corresponding section description of pubic symphysis, urethra, vagina, rectum, and puborectalis muscle.[Noort et al., 2018]

2.1.3 Avulsion

An avulsion is a medical term which explains the tearing off of body tissue (muscle or tendon). When a muscle or tendon is partly torn off, it is described as a partly avulsion. When there is no connection between both the ends of the muscle or tendon and its support, then it can be described as a complete avulsion.
The avulsion which most common [Dietz, 2013] in the pelvic floor after a child-birth is the tear-off of the puborectalis muscle as shown in red on figure 2.2(b). This muscle has to stretch up three times its original length during delivery.



(a) (b)

Figure 2.3: (a) Transperineal ultrasound before the delivery (b) Transperineal ultrasound after the delivery, with an avulsion at the left side of the ultrasound image

Image 2.3 shows an example of an ultrasound image of a patient. The left ultrasound image (a) shows a woman with an intact pelvic floor and before delivery. The intact pelvic floor is characterized by the V-shaped puborectalis muscle around the anus. This V-shape muscle looks symmetric and has no interrupting section.

The right ultrasound image shows the ultrasound image of the same patient six months after delivery. From this picture can be seen that the V-shape is still intact underneath the anus, but the muscle shows tear at the upper left part of the V-shape. This tear indicates that the muscle has torn at this part and is (partly) detached from the bone. This detaching is visible by the difference between the left and right side of the puborectalis muscle.

2.2 Artificial Intelligence

In the year 1955, John McCarthy first describes the term Artificial Intelligence (AI) as *"the science and engineering of making intelligent machines"* [McCarthy, 2007]. This statement can be elaborated into more detailed fields, one of which is Machine Learning (ML).

ML is a great sub-field of AI and is dealing with the field of study that let computers achieve the ability to learn without being explicitly programmed. The machine learning principals are in line with the learning of humans. For example, when a child learns to identify objects, we do not tell them an algorithm but rather give them multiple examples of that object. Its brain will automatically identify the features and learns to identify that specific object. In this situation, the brain of the child can be referred to the machine learning algorithm that has been developed.

2.3 Neural network

Deep learning is a subset of ML and based on a Neural Network (NN). Where these Neural networks are based on the collection of connected nodes called neurons; these nodes are loosely modeling the neurons in a biological brain. Each connection, synapse in the biological brain, can transmit a signal from one neuron to another affected by a weighting factor for that specific connection. These neurons can be stacked together into one layer. A neuron in the next layer receives multiple input signals from the previous layer and processes the data by doing a weighted sum. This addition again can be sent to the next layer by applying an activation function. The final layer in a network can be described as the output layer. By comparing the output layer with labels, the accuracy of the predicted output and the real output can be calculated. The difference between real output data and predicted data is used for backpropagation. Backpropagation is a method of computing gradients which is used for updating the weights between nodes. Different functions can calculate the gradient, and such functions are called optimizers. An elaboration of the most important parts about a neural network is given below.

Node

Neurons in the biological brain are loosely defined as nodes in a neural network. The billions of connected neurons in the brain are forming a neural network. Each neuron in this network has the basic structure of multiple input values, which are weighted. When a certain threshold is reached, the neuron will 'fire'. This firing means that the output of that neuron will be propagated throughout the outputs of that specific neuron.

This propagation of neurons in the human brain is mimicked with a neural network. The inputs of that node ($\boldsymbol{\xi}$) are multiplied with a specific weight ($\mathbf{w}$) to perform an addition of $\phi = \xi_0 w_0 + \xi_1 w_1 + \xi_2 w_2 + \cdots + \xi_n w_n$ with an constant value of $\xi_0 = 1$. This constant value represents the threshold of every node in order to 'fire' its output. This is schematically shown in figure 2.4.

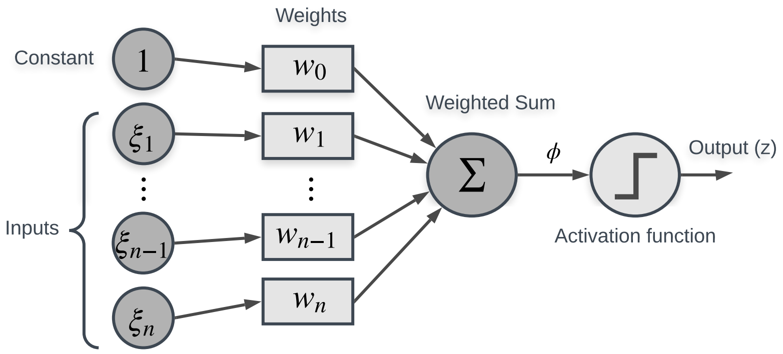


Figure 2.4: A simplified representation of a single node from a neural network.

Figure 2.4 can be described mathematically as the weighted sum over $\boldsymbol{\xi}$ and $\boldsymbol{w}$:

$$\phi = \sum_{i=0}^{n} (w_i \cdot \xi_i) \tag{2.1}$$

Only a small selection of possible activation function, loss function, and optimizers are given below. This is done because there are too many possibilities to elaborate them all. Therefore, a small selection of the most commonly used is given.

Activation functions

Activation functions ($f(\phi)$) are defining the output (z) of a given neuron. The output is defined by a weighted sum over all input values, which is results in a single number. This number is passed to the next neuron. The output of this weighted sum can become very large from $-\infty$ to $+\infty$. To avoid numerical overload of the output of a single neuron, the output is passed through a sigmoid function

$$z = f(\phi) = \frac{1}{1+e^{-\phi}}, \tag{2.2}$$

where the output value of the sigmoid can be in the range from 0 to 1. Towards either the 0 or 1 the numbers tend to respond very less to changes in ξ, this results into a low gradient. Low gradients raise the problem of the vanishing gradient, meaning that the gradient is becoming very low at the first couple of layers within the network.

To avoid the vanishing gradient problem, a Rectified Linear unit (ReLu) can be introduced. This ReLu is a non-linear activation function which uses the maximum value above 0 and zero below 0.

$$f(\phi) = \begin{cases} 0 & \text{for } \phi \leq 0 \\ \phi & \text{for } \phi > 0 \end{cases} \tag{2.3}$$

This formula can be written in short by $f(\phi) = \max(0, \phi)$, with a specified range from 0 to $+\infty$. One problem of the ReLu is that the derivative of any value below zero is 0, this means that when the output is zero, no updates are performed. When these weights are not updated, and this results in dead nodes, to avoid this problem, a leaky ReLu can be used. This leaky ReLu has small linear scalar α in front of the negative values.

$$f(\phi) = \begin{cases} \alpha\phi & \text{for } \phi \leq 0 \\ \phi & \text{for } \phi > 0 \end{cases} \tag{2.4}$$

This α could be chosen to create a small derivative for the negative outputs, to perform an update if the output was negative. This results into the mathematical expression: $f(\phi) = \max(\alpha\phi, \phi)$.

Loss function

The loss function defines the difference between the labels (ℓ_i) and the predicted data ($\wp_i$). This difference can be calculated by subtracting the target value minus the predicted value, where the absolute of this subtraction can be taken to find the Mean Absolute Error (MAE). To calculate the average magnitude of errors in a set of predictions, without considering their directions, all the differences between the target and label are summed and divided by the total number of outputs (no). The Mean Absolute Error (MAE) can be calculated by:

$$MAE = \frac{\sum_{i=1}^{no} \left|\ell_i - \wp_i\right|}{no} \tag{2.5}$$

The mean average error results in a linear error of predicted value.
When the sum of errors is replaced with the weighted sum of squared errors Mean Squared Error (MSE).

$$MSE = \frac{\sum_{i=1}^{no} \left(\ell_i - \wp_i\right)^2}{no}, \tag{2.6}$$

the predictions which are far away from the actual targets are penalized more massively in comparison to the less deviated predictions. This results in a more significant update for predictions which are further away from the correct labyel, due to the square of the MSE.

A small number of the available loss functions has been described. The loss function is chosen in combination with the activation function, this is because some loss functions are optimized for a specific activation function.

Optimizer

Optimizers are used to minimize the loss function, which is dependent on the differences between the target values and the predicted values. The learnable parameters ($\boldsymbol{\theta}$) such as the weights and biases are updated using the optimizer. This is done by minimizing the network's training loss. Minimizing the error can be done by calculating the loss from the target and the predicted values. This loss can be used to converge to the model's (local) minimum, which can be done as follows for the gradient descent:

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta \cdot \nabla J(\boldsymbol{\theta}). \tag{2.7}$$

Gradient descent, equation (2.7), is the formula for updating the learnable parameters, where η is defined as the learning rate, ∇J($\boldsymbol{\theta}$) is the gradient of the loss function J($\boldsymbol{\theta}$) with respect to the learnable parameters ($\boldsymbol{\theta}$).

The gradient descent, as described above, calculates the gradient of the whole dataset and will only perform one update per batch. However, this method can be slow and will not always converge to the global minimum, because every iteration of all the data points are taken into account. When updating the network after each training example (what is called stochastic gradient descent), the technique is converging for larger datasets quicker and has more chance to find the global minimum. This happens because the parameters updates have high variance and cause the loss function to fluctuate to different intensities. However, this fluctuations comes at the cost of converging to an almost minimum and will keep overshooting.

To avoid this problem, mini-batches can be used to avoid the high fluctuations in updating the learnable parameters. This method is called the (mini) batch gradient descent.

To speedup the convergence, a so-called moment can be introduced. This moment uses the acceleration of the previous gradients to soften the oscillations in irrelevant directions. The updating of the new learnable parameters is done by taking a factor (γ) for the previous update and perform the batch stochastic gradient descent:

$$\mathrm{V_t} = \gamma \mathrm{V_{t-1}} + \eta \nabla \mathrm{J}(\boldsymbol{\theta}) \tag{2.8}$$

Where the learnable parameters are updated ($\boldsymbol{\theta}_{t+1}$) by subtracting the gradient descent (V_t) From the learable parameters, $\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - V_t$.

To improve the moment of learning even better, the Adaptive Moment Estimation or shortly 'Adam' optimizer [Kingma and Ba, 2014] can be used. The Adam optimizer adapts its learning rate for each parameter in addition to an exponentially decaying average of past squared gradients, which is similar to moment. The Adam optimizer uses the first and second moment, where the first moment consists of the mean (m_t) and the second of the uncentered variance of the gradients (v_t). The β_1 and β_2 can be chosen to set the amplification of the first or second moments.

$$\begin{aligned} \hat{m}_t &= \frac{m_t}{1-\beta_1^t} \\ \hat{v}_t &= \frac{v_t}{1-\beta_2^t} \end{aligned} \tag{2.9}$$

Equation (2.9) is used to update the learnable parameters:

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \frac{\eta}{\sqrt{\hat{v}_t} + \epsilon} \hat{m}_t \tag{2.10}$$

For updating the weights, the learning rate η is divided by the square root of the second moments plus a small number ϵ to avoid a division by zero.

2.3.1 Unsupervised learning

In comparison with supervised learning, unsupervised learning is a neural network that learns from training data that does not need to be labeled. Instead of responding to feedback obtained by the labels, unsupervised learning identifies commonalities in the dataset and reacts to the presence or absence of such commonalities.

An advantage of unsupervised learning is that in medical applications, there is mostly an abundance of data, but a lack of labeled training data. When unsupervised learning is used, there is no validation set needed, so more data can be used for training the neural network. The labeling of medical data is hard and should be done by a specialist to create reliable labels.

2.3.2 Convolutional neural network

Machine learning algorithms, in particular, convolutional neural networks, have become of great importance within the classification and segmentation of images. The first neural networks were mostly used for the classification, and AlexNet was the breakthrough of the neural networks [Krizhevsky et al., 2012]. AlexNet was the first neural network which was able to win the ImageNet competition in 2012, other classical machine learning and computer vision programs were beaten significantly. Within this competition, the goal of the network is to indicate what objects are pictured. The network with the lowest error rate wins the competition. Humans can achieve an error of 3.6%. The amount of correct classified objects describes the

performance of the network. AlexNet was able to achieve an error rate of 15.3% compared to the 26.2% achieved by the second-best entry [Krizhevsky et al., 2012]. Where the AlexNet was a relatively small network with only eight layers.
The increment of layers will take more time to compute the weights of the networks but increases the performance of the classification [Simonyan and Zisserman, 2014]. Increasing the depth is outperforming more complex recognition pipelines. This confirms the importance of depth in the visual representation of neural networks. In 2014, Google participated in the ImageNet competition [Szegedy et al., 2015] with an error rate of 6.67%, which is close to the human error of 3.6%. Google has optimized the AlexNet by creating a network of 22 layers deep and drastically reduced the number of parameters by using small convolutional layers and skipping parts of the network.
He et al. [2016] uses a similar network with many parameters, and he introduced a novel architecture with skip connections. These skip connections let the ResNet be able to train a neural network with 152 layers while keeping the complexity. This network has been applied to the ImageNet competition, where the beats the human performance of 3.6% by 0.03%. Resnet shows that deeper networks are still able to be trained and that these deeper networks will outperform smaller networks.

Convolutional Neural Networks (CNN) are deep neural networks which consist of an input and output layer, as well as multiple hidden layers. Between those layers, a filter (kernel) can be defined which slides over the complete image (convolution), as schematically shown in figure 2.5. Along this sweep over the picture, every slide (number of moved indices) the filter applies the dot product between the kernel and the overlaying input parameters of the previous layer. This product results in one new value which is stored in the next layer. The number of products defines the dimensions of the new layer. The depth (number of channels) of one layer can increase according to the structure. The creator of the network defines the structure of a network before training. Multiple layers can be places after each other to create the desired shape of the network.
Different techniques like pooling can be used to apply a condition to the next layer. Pooling looks at a specific kernel and keeps the highest or average value and passes this value to the next layer. The last layer can exist of a fully connected layer, which connects all nodes of the previous layer to the next layer. These nodes are used for classification networks, where the output is defined according to known labels of the images.

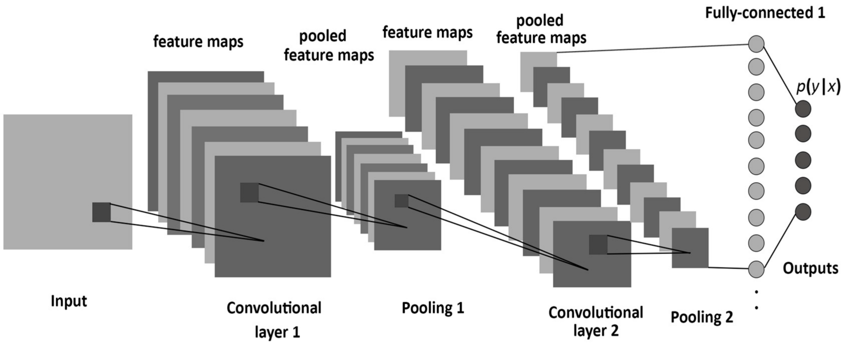


Figure 2.5: A simplified representation of a convolutional neural network for image classification [Albelwi and Mahmood, 2017].

2.3.3 Autoencoder neural network

Convolutional autoencoders are a type of convolutional neural networks used to learn efficient data coding [Hinton and Salakhutdinov, 2006]. This network aims to learn an input-output relation for a set of data, for example, dimensionality reduction, segmentation, or noise reduction [Vincent et al., 2008]. This input-output relation is used by encoding an input image to

a lower-dimensional latent space by applying several convolutions. Along with the encoding side, a decoding side is learned, where the autoencoder tries to generate from the latent space representation as close as possible to its desired output. This can, for example, be the input or a segmented image.
Figure 2.6 gives an example of an autoencoder with dimensionality reduction. By transforming input layers to a multidimensional matrix and transform it back to its original shape. The decoding part is mostly mirrored from the encoded part to reduce the variety within the model and increase insight to network.

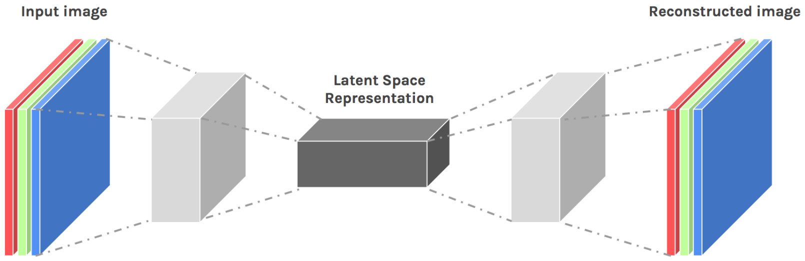


Figure 2.6: Example of an autoencoder network with a RGB input, RGB output, hidden layers and a latent space representation in the middle layer [Despois, 2017]
.

The Red Green Blue (RGB) input, RGB output and hidden layers (grey) are visible in the figure 2.6. The network contains a three-channel RGB input, a three-channel RGB output, and two hidden layers which encode the dimensions of an image input to multidimensional hidden layers. Between the layers, a convolution or pooling can be applied, which results in changing the size of every layer.

Unsupervised neural networks can be learned to create a segmentation of specified objects. One of the most popular autoencoder networks is made by Ronneberger et al. [2015] and is called U-NET. U-NET is one example of creating a segmented output image from an input image. The architecture consists of a contracting path to capture the context and symmetric expanding path that enables precise localization of objects. The network uses 572×572 input data which is fed through several linear convolutional layers and max pool layers to decode the data into a small latent space representation. This representation is decoded into the original image shape to create a segmentation. The shape of this U-NET is graphically similar to the shape of an U. Between the encoding and decoding, layers are interconnected, which introduces an additional flow of information. These deep convolutional neural networks have made breakthroughs in various fields such as; image, video, and text processing. Currently, most CNN are becoming deeper and deeper to create better performance [Gu et al., 2018]. This comes at the cost of needing a larger dataset and massive computational power for training. For manually collecting this amount of data requires enormous amounts of human efforts, the exploration of unsupervised learning is desirable. Unsupervised learning reduces the amount of required human classification.
Besides the capabilities of convolutional neural networks, the fundamental theory of CNN is still underdeveloped. Current CNN models work very well for a variety of applications. However, we sometimes do not know why and how it works essentially [Gu et al., 2018].

2.4 Latent space representation

The autoencoder network creates a latent space which is used for reconstructing the desired output. By training an unsupervised autoencoder network, we minimize the error between the network's input and output. If the output and input are almost identical and the loss function is low, the latent space must contain all information needed to reconstruct the output image. This information is extracted from the network by making the network such that the data passes through a single width, height vector, and a specified depth (D) ($1 \times 1 \times D$). This vector can be used to find any low-level correlation that represents a similar correlation in input pictures. This correlation of vectors can be used to identify labels and define clusters within the obtained data.

The latent space is obtained in the middle of the hidden layers from the convolutional neural network, as indicated in figure 2.6. For all pictures, the vector representation is extracted from the network, and the vector can be viewed as a high dimensional feature representation of the image. To reduce the number of features in these vectors, we can apply a dimensionality reduction method.

2.5 Dimensionality reduction

There are a few dimensionality reduction methods; in this study, we only focused on PCA and t-SNE. Dimension reduction methods are used on high dimensional (matrix) data to reduce the order of the data. This reduced data can be analyzed to find any use full clusters or patterns in a lower dimension.

2.5.1 Principal Component Analysis

Princial Component Analysis (PCA) is a dimensional reduction method, as elaborated in the book Bishop [2006], that maps linearly a high dimensional dataset into a lower-dimensional dataset. For a given input matrix (X) where $\mathbf{x}$ is defined as a vector of one latent space representation and $\mathbf{y}$ is defined as a specific feature over multiple latent spaces. For $\mathbf{x}$ being one latent space representation of n dimensions where $\mathbf{x} = (x_1\ x_2\ \ldots\ x_n)$ is the vector representation of one image.

$$X = \begin{pmatrix} x_1 & x_2 & \cdots & x_n \\ x_1 & x_2 & \cdots & x_n \\ \vdots & \vdots & \ddots & \vdots \\ x_1 & x_2 & \cdots & x_n \end{pmatrix}, = \begin{pmatrix} y_1 & y_1 & \cdots & y_1 \\ y_2 & y_2 & \cdots & y_2 \\ \vdots & \vdots & \ddots & \vdots \\ y_n & y_n & \cdots & y_n \end{pmatrix} \tag{2.11}$$

Principal component analysis converts a set of observation of possibly correlated variables into a set of linearly uncorrelated variables called principal components. These individual principal components are orthogonal to each other and can be distinguished by using the eigenvalues and eigenvectors of the covariance matrix of the specific input matrix. The indexes of the covariance matrix

$$\mathbf{S} = \begin{pmatrix} s_{11} & s_{12} & s_{13} & \cdots & s_{1k} \\ s_{21} & s_{22} & s_{23} & \cdots & s_{2k} \\ s_{31} & s_{32} & s_{33} & \cdots & s_{3k} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ s_{j1} & s_{j2} & s_{j3} & \cdots & s_{jk} \end{pmatrix}, \tag{2.12}$$

can be calculated using formula (2.13). From the input matrix x_i is defined as the indices of $\mathbf{x}$, y_i is defined as the indices of $\mathbf{y}$ and n the length of $\mathbf{x}$.

$$s_{jk} = \frac{1}{n-1} \sum_{i=1}^{n} \left(x_i - \overline{x}\right)\left(y_i - \overline{y}\right) \tag{2.13}$$

The mean of **x** and **y**, denoted as $(\overline{x}, \overline{y})$, can be calculated by:

$$\overline{x} = \frac{1}{n}\sum_{i=1}^{n} x_i, \quad \overline{y} = \frac{1}{n}\sum_{i=1}^{n} y_i \tag{2.14}$$

From this covariance matrix $S^{M\times M}$, the eigenvalues λ_i and corresponding eigenvectors $\mathbf{u}_i$ are calculated. The eigenvectors with the highest eigenvalues capture the largest variation in the data. The chosen eigenvectors are used for defining the new base of the lower dimensional plot. Preferably, the eigenvectors with the highest eigenvalues are used because then the most information is displayed in the lower dimension.

$$S\boldsymbol{u}_i = \lambda_i \boldsymbol{u}_i \quad i = \{1, 2 \ldots M\} \tag{2.15}$$

u_i equals the eigenvector and corresponding eigenvalue λ_i. These can be calculated by solving equation:

$$|S - \lambda_i I| = 0 \tag{2.16}$$

$\boldsymbol{u}_i$ is a column of U ($U = \boldsymbol{u}_1\ \boldsymbol{u}_2\ \ldots\ \boldsymbol{u}_M$) where $U^T U = I$ and the eigenvectors can be calculated by substituting the eigenvalues at the diagonal of the matrix $\bigwedge$:

$$SU = U\bigwedge. \tag{2.17}$$

The PCA analysis can be done by using the input data, which projects the input onto the specific PCA components by multiplying the input matrix (X) to that specific principal components. The projected to the new dimensions (**t**) can be calculated by

$$\mathbf{t}_i = X \cdot \mathbf{u}_i. \tag{2.18}$$

The percentage of variance (VAR) captured within a specific component of the lower dimensional plot can be calculated as follows, where λ is the eigenvalue:

$$VAR(S) = \frac{\lambda_i}{\sum_{i=1}^{n} \lambda_i} \tag{2.19}$$

2.5.2 T-SNE

t-Distributed Stochastic Neighbor Embedding (t-SNE) [Maaten and Hinton, 2008] is a visualization technique to present high-dimensional data by giving each data point a location in a 2D map. The t-SNE technique works by converting high-dimensional Euclidean distances between data points into conditional probabilities that represent similarities. The transformation of the higher dimensional space into the lower dimensional space is non-linear. The similarity of datapoint x_j to datapoint x_i is the conditional probability $p_{j|i}$, that x_i would pick x_j as its neighbor. The proportion to their probability density, $p_{j|i}$, under a Gaussian which is centered at x_i is for nearby data points is relatively high.

The conditional probability $p_{j|i}$ is given by

$$p_{j|i} = \frac{\exp\left(-\left\|\mathbf{x}_i - \mathbf{x}_j\right\|^2 / 2\sigma_i^2\right)}{\sum_{k\neq i} \exp\left(-\left\|\mathbf{x}_i - \mathbf{x}_k\right\|^2 / 2\sigma_i^2\right)}, \tag{2.20}$$

where σ_i is the variance of the Gaussian that is centered on datapoint x_i.

Where the similarity of the data points in the higher dimensional space is picked in proportion to their probability density under a Gaussian centered:

$$p_{ij} = \frac{p_{j|i} + p_{i|j}}{2n} \tag{2.21}$$

Secondly, the similarities and proportions in the higher dimensional space are mimicked on the lower-dimensional space. In the low-dimensional space, the distance is converted into probabilities using a Student's t-distribution.

The similarities on the lower dimensional (y_i and y_j) space are defined as:

$$q_{ij} = \frac{\left(1 + \|\mathbf{y}_i - \mathbf{y}_j\|^2\right)^{-1}}{\sum_{k \neq l} \left(1 + \|\mathbf{y}_k - \mathbf{y}_l\|^2\right)^{-1}} \tag{2.22}$$

Where y_k and y_l are different points in the dataset. A Student t-distribution is used to measure the similarities in the lower dimensional space. The t-distribution creates a map's representation of joint probabilities almost invariant to changes in the scale of the map for points far apart. The relation of these points is iteratively optimized by minimizing the Kullback-Leibler divergence. This results in:

$$KL(P\|Q) = \sum_{i \neq j} p_{ij} \log \frac{p_{ij}}{q_{ij}} \tag{2.23}$$

3 2D ultrasound images

The 2D ultrasound images of the pelvic floor, as elaborated in chapter 2.1, can be analyzed by using an autoencoder neural network. This was done during the prior research performed in the Individual Assignment (IA) [Bongers, 2019].

3.1 Introduction

This prior research was performed by using an autoencoder neural network which can encode input images 512 × 512 into a latent space of 512 values. This latent space is then decoded to reconstruct the original input image. The latent space has a factor of 512 fewer parameters than the input image. This reduction results in data compression of 99.8%.
From the network, the high dimension latent space (512 dimensions) is extracted and subsequently reduced in two dimensions. This further reduction is applied to create a two dimensional scatter plot. The scatter plot is examined for clustering, which can be introduced by the dimension reduction method PCA or t-SNE.

The input images are transformed into a two dimensional scatter plot, using the dimension reduction t-SNE. Figure 3.1, shows the reduction of each of these input images to one datapoint based on two values. Within this figure, the plot shows the t-SNE reduction of the data points by labeling the different states of contraction.

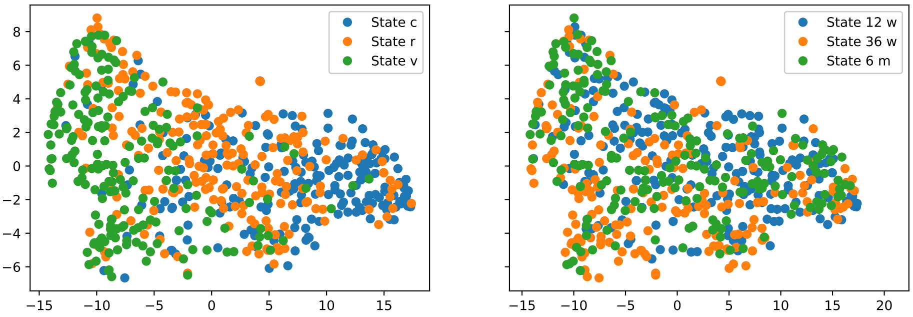


Figure 3.1: 2D scatter-plot of prior research by using the labels of contraction ’c’, rest ’r’ and Valsalva ’v’ and the time labels of 12 weeks, 36 weeks pregnant and 6 months after delivery. [Bongers, 2019]

From figure 3.1 it can be seen that the contraction ’c’ and the Valsalva ’v’ are distinguishable. Besides the contraction and the Valsalva, the rest state is in between. This means that the rest pictures has similarities with either contraction or Valsalva. While looking at the different recording moments during pregnancy, there is no clear clustering visible.

The clustering, found in the prior research, was based on the most prominent feature of the input images. This feature was based on the shape of the input image, instead clinically relevant information visible on the ultrasound image.
The colored dots within this plot are representing three different labels: contraction, rest, and Valsalva. The dots can be replaced with the real output images to see the effect of the clustering, to validate the 2D scatter plot. An example of a scatter plot where the dots are replaced by images is shown figure 3.2.

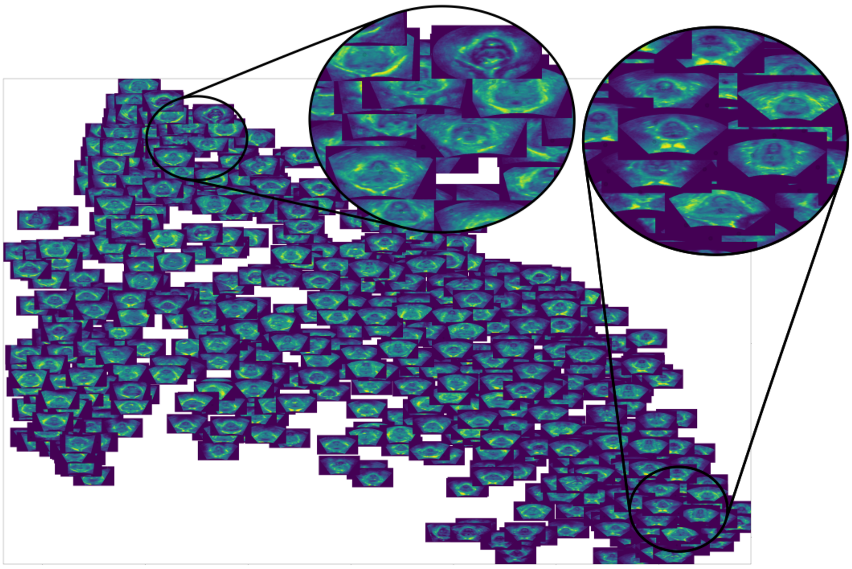


Figure 3.2: 2D scatter-plot of prior research by visualizing the bias caused by the input images of this network. From the two highlighted parts can be seen that the US shape is different.

From this image, it can be seen that the clustering was based on the shape of the ultrasound image, instead of clinically relevant information. The size of the 2D ultrasound image changes because the slides are taken from a 3D input image and changing the angle to obtain the 2D slice according to the position of the PRM. By varying the angle of the 2D slice through the 3D input, the shape of the result changes. This change of shape is the most prominent feature and is visible on the scatter images above. The angle of the PRM is different when women apply a contraction or Valsalva movement. This angle results in a different slice within the 3D ultrasound box. If this 3D matrix is sliced differently, the shape of the ultrasound varies, which explains the clustering.
Therefore we continue with evaluating the ultrasound images by performing an eigenanalysis. This method is visualizing the different variations between the multiple principal components. From these analyses, we can analyze the most significant variation over every principal component between the different images.

This analysis has further been elaborated by performing an eigenanalysis in combination with a mask to latent space representation. From this eigenanalysis, we can see what the specific variation of every principal component is describing. Where finally, the results of this 2D analysis are presented and concluded.

3.2 Method

To make the clustering and network less sensitive to the ultrasound shape. The ultrasound shape can be defined as the ultrasound captured by the device, as shown in figure 2.2, without considering the black background.
At first, the network is trained by only using the ultrasound area in calculating the loss funciton without considering the background. Second, the difference within the principal components is analyzed by applying an eigenvector analysis.

3.2.1 Neural Network

The network, as used in the IA is changed, to be able to extract the latent space and to import new vectors into the latent space. The network of the IA uses skip links between the encoding and decoding. These skip links are increasing the quality of reconstruction, but are resulting in an information flow which does not pass the latent space. To avoid a reduction in information flow, skip links are omitted in future networks.

This results in the neural network, as shown in figure 3.3.

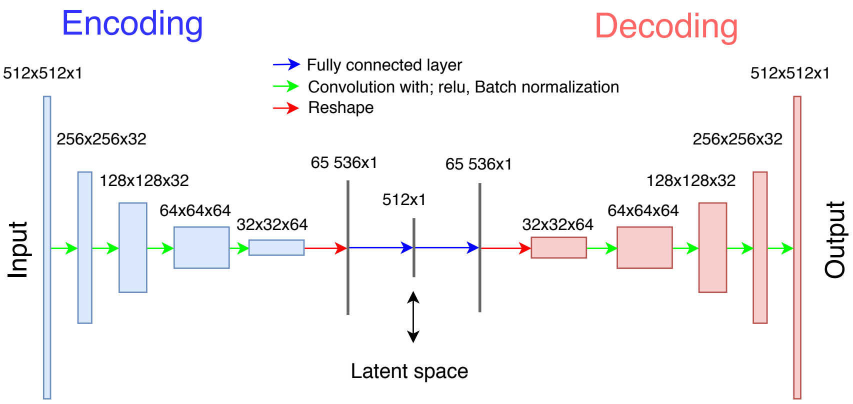


Figure 3.3: Autoencoder neural network for extracting and passing the latent space to the network, where the network consists of 8 convolutional layers and 2 fully connected layers.

The neural network's input exists of 512×512 parameters, which are reduced in size to a similar latent space of 512 parameters as used in the IA. The network consists of 4 convolutional encoding layers, which are used to find a combination of features within the images. This partial independent combination of features is used in a fully connected layer, which connects these. The output of this fully connected layer is represented as a vector representation. The decoding part of the network is similar to the encoding part of the network.

Besides the architecture of the network, the output layer of the network is kept linear. Where the network uses mean square error and Adam optimizer for backpropagation.

3.2.2 Dimension reduction of latent space

The latent space is again further reduced in size by applying two dimension reduction methods; PCA and t-SNE. After applying the dimension reduction methods, the image exists of a datapoint described by two values, so that this can be plotted in a 2D scatter plot. These data points can be represented by plotting one data point on the x-axis and one on the y-axis. This creates a scatter plot from different images.
When the 2D input images are reduced to a 2 dimensional scatter plot, the input data has been reduced by a percentage of

$$\rho = (1 - \frac{2}{512 \cdot 512}) \cdot 100\% = (1 - 2^{-17}) \cdot 100\% = 99.9992[\%], \tag{3.1}$$

where ρ describes the reduction of the latent space.

3.2.3 Mask

A mask for updating the loss function in the neural network is done to create more focus on the image instead of the background. This mask is multiplied with the loss function before applying the backpropagation through the network. The mask creates a 1 for pixel values in the US image (pixels generated by the ultrasound device) and assigning a 0 to the background pixels. The result of this mask is shown in figure 3.4, where the left figure shows the original ultrasound image and the right image the generated mask.

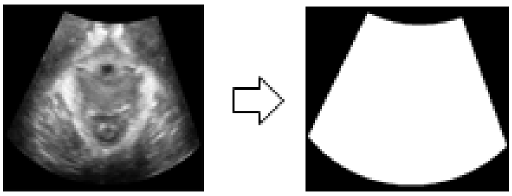


Figure 3.4: Creating a mask for updating the loss function in the neural network.

3.2.4 Eigenvector analysis

An eigenvector analysis can be performed to visualize each of the different principal components. An eigenanalysis is part of the active shape modeling as done by Cootes et al. [1995]. This eigenanalysis shows for every single principal component the variation of that specific component. This method can be used to see which features of the input image are captured in which principal component.
The eigenanalysis is performed on the latent space within the network. The latent space is adapted according to the eigenvector analysis. After that, the network decodes the latent space, which results in the variation of images.

Applying an eigenvector analysis explains the variation captured by one single principal component, this is done by calculating the mean value over all latent spaces

$$\overline{\boldsymbol{x}} = \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{x}_i, \tag{3.2}$$

where N is indicating the number of latent spaces.
U describes all the principal components as vectors presented in a matrix $U = (\mathbf{u}_1\ \mathbf{u}_2\ \dots\ \mathbf{u}_k)$. Where the different principal components can be described by $k = (1,\ 2, \dots N)$. Where the principal components are ordered by the weight of its eigenvalue.

The eigenvalues are used to create the variance over one particular principal component.

$$-m\sqrt{\lambda_k} \leq b_k \leq m\sqrt{\lambda_k} \tag{3.3}$$

Since the variance of the training set can be shown as λ, the suitable limits are typically plus/minus three times the standard deviation of the mean. Where m can be varied from $m = [-3, -2, -1, 0, 1, 2, 3]$

$\boldsymbol{n}$ is a vector of which the k^{th} elements is substituted by b_k. The substituted elements are the principal components over which are varied. This is done by taking a variance (n) to that specific principal component and multiply this by its eigenvector and add the mean value.

$$\mathbf{PC} = \overline{\mathbf{x}} + \boldsymbol{n} \cdot U, \tag{3.4}$$

The Principal Component (***PC***) are explaining the variance over that particular principal component.

3.3 Results

Applying both methods to the results as obtained from the prior research, the results are shown in the following paragraphs. First giving the results of applying a mask and second giving the results of the eigenanalysis.

Using the mask, the clustering improved by eliminating the focus to the shape of the image, which can be seen in figure 3.5. The mask increases the scattering of the different labels (contraction-rest-Valsalva) as found previously. This indicates that the network is less sensitive to the shape of the ultrasound images, but loses the capability to cluster.

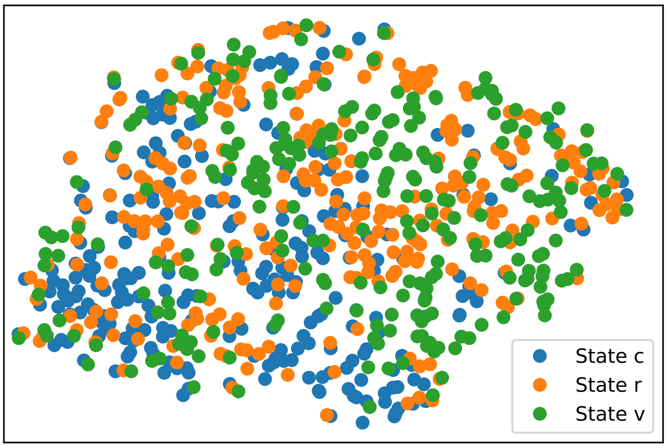


Figure 3.5: 2D scatter-plot of prior research by using the mask to update the loss function. The labels of contraction (state 0), rest (state 1) and valsalva (state 2)

To visualize the differences within the principal components, figure 3.6 visualizes the different variation of every principal component. In this figure, the top row is describing the different values of σ, and the different columns are describing which principal component has been used. For example, the first principal component is described as PC1, and the second principal component is described as PC2 and so-on.
Graphically speaking, every m component represents a single column in figure 3.6. The middle column and the first row represents the homogeneous image, which has the most commonality with all of the pictures.

The second row describes the variation of the first principal component deriving from the general image. The first column represents: minus three times the standard deviation, the second row: minus two times the standard deviation and so-on.
The third row describes the variation of the second principal component with the variance of m as previously described. The structure of the rows 3 to 10 are similar as previously described.

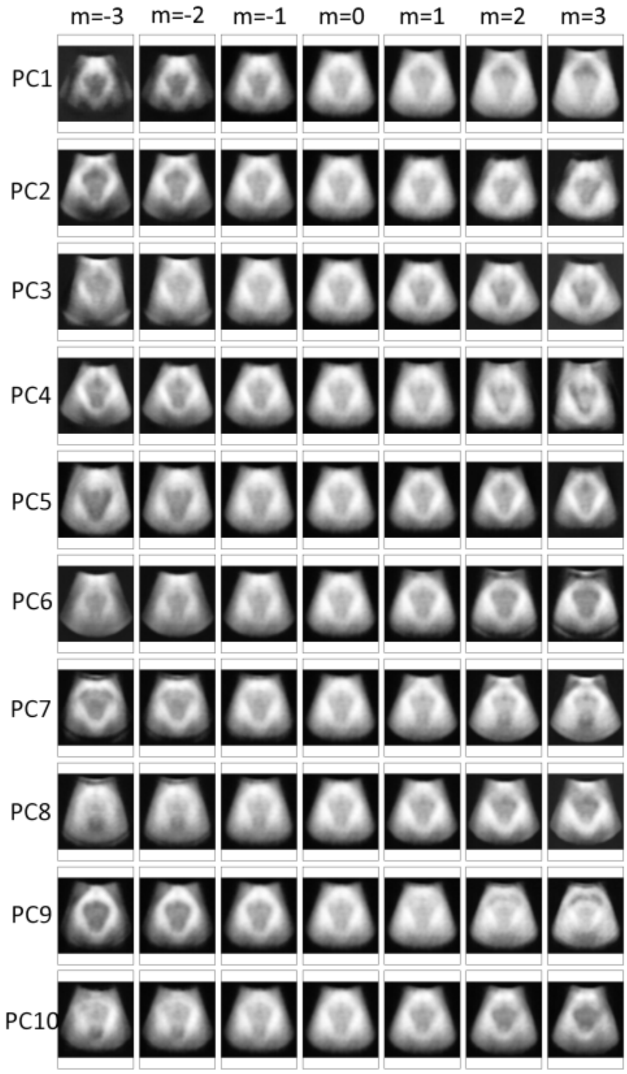


Figure 3.6: The analysis of the eigenvectors by applying plus-minus three times the standard deviation to the homogeneous image.

The variation of every component can be seen, looking at the different principal components, as shown in figure 3.6. This can be seen that by varying the *m* over a specific principal component, the image changes. This change in image, can be due to the change in ultrasound shape or the change due to the change in PRM muscle visible on the US.

3.4 Conclusion

From the 2D analysis, we conclude that the neural network can find the most prominent features of the input ultrasound images, but the network was not able to find clinically relevant features. The most prominent feature which was available within the pictures consists of the variation of ultrasound image shapes. This difference in shape is because the images are taken from a 3D box where the puborectalis muscle was aligned with the 2D image. This is in line with the results as obtained from the eigenanalysis. Most of the principal components are visualizing the difference in shape while extracting the clinically relevant features is not possible. Therefore, the scatterplot corresponds with the selected angle of the orientation of the 2D plane within the 3D ultrasound image.

From this, we can conclude that the most prominent feature of the 2D image is the ultrasound image shapes instead of the puborectalis muscle. To avoid clustering based on the shape of the ultrasound image, we have to use the 3D data, which does not vary in shape.

4 3D ultrasound images

Having analyzed the 2D dataset as used in chapter 3, the research is extended with a 3D analysis.

4.1 Introduction

For this 3D analysis, higher-dimensional images are used to analyze the different states. To get rid of the US shape variance, as found in the previous chapter, the 3D images are used, because these do not suffer from shape invariance.
The method and analysis are both split into two different parts, first frame analysis and an all frames analysis. By analyzing all frames, the variation between the different frames of one patient and between patients can be compared. When looking at the first images of all patients, the difference between the patients can be analyzed.

First frame

Clustering the frames of the first input images of the video are explaining the differences between patients in the time frame. The different time frames can result in clusters of a specific time frame.
If this clustering of time frames is visible, the network can find clinical features which are relating to the change of pelvic floor due to the presence or absence of a baby. This indicates that the network can detect clinically relevant features in the input images.

All frames

In comparison to analyzing the first frame, all frames should cluster for that specific patient with a small variety, because all frames within a video have many similarities. If this clustering happens, the different frames can be analyzed, and the different states of contraction (contracted, rest and Valsalva) should be visible. The variety within this clustering can be analyzed in the higher dimensional data by looking at the Euclidean distance between the frames. While at the lower dimension, the movement of that frame with respect to another frame can be analyzed.

4.2 Method

4.2.1 Image pre-processing

Because all the input images have a slightly different size, the input is reshaped and normalized. This happens such that the pre-determined frame is selected from the center of the original picture. From the center of each image, we take a $192 \times 192 \times 96$ box and resample this box to a $128 \times 128 \times 64$ image. The size of the box is chosen so that these are a power of 2^n ($2^7 \times 2^7 \times 2^6$). Besides the resampling, the images are normalized from a range of 0 to 255 to an input range from 0 to 1. The data has been normalized because this improves the reconstruction quality of a neural network.

4.2.2 Image selection

The videos can be analyzed by using one specific frame from each video or using all frames from that video. Both methods are tested separately because the assumption is made that all first frames are starting in rest, which makes the comparison between the different patient possible. When all frames are used, the different states between one specific patient can be distinguished. This distinction can be used to validate the assumption from the first frames.

First frame

The first frames exist for all patients over the three different time phases. In total, 291 patients have been recorded on video. We should expect 873 videos of the 291 women in 3 timesteps (12 weeks, 36 weeks pregnant, and 6 months after delivery). Although this number is too large because some women have missed one or more of the recording moments. Therefore, we have a data set of 765 videos, of which the first image can be used.

The first frame images are divided into two groups, training set (733 images) and validation set (32 images). The training group is used for training the network and adapting the weights to optimize the input-output relation. The validation set is never used for updating the weights and thus will be independent of the weights. The validation set is used to determine the generalization capability of the network. If the network has reached an optimum, the error validation set increases. If this happens, the network stops training after validating the last 50 validation epochs. Training the network is done by using stochastic batch gradient descent and applying the mean square error loss function in combination with the Adam optimizer. The batch size of the network is defined as 16 frames.

All frames

Using all frames of the video increases the amount of data because each video consists of approximately 60 frames. When using all frames of all women, the number of input images are approximately 46 000 3D images. By using a selection of randomly chosen patients, the amount of data is reduced. This data is chosen randomly throughout the data set to give an objective representation of the data set.
100 videos are randomly selected to create a dataset which is split by half to train and half to test. This test set is used for determining the Euclidean distance between the different frames. This training set consists of 3284 frames from 50 different patients. This set is split into a training set of 2784 and a validation set of 500 images. For training this network, a similar approach as described before is used. Training of the network uses stochastic batch gradient descent with a batch size of 16. This in combination with the MSE loss function and the Adam optimizer.

4.2.3 Neural network

The Neural Network (NN) is created by adapting the NN created for the 2D input images. This similar approach is made because the 3D data has four times more input data than the 2D data. The 2D data has a width and height of 512, this results into $512^2 = 262\ 144$ input parameters. The 3D input data has a width and height of 128 and a depth of 64 this results into $128^2 \cdot 64 = 1$ 048 576 parameters, which is a factor 4 times larger as the 2D input data.
The training time of the 3D network is 32 times longer than the 2D network. Because of the long computational time for the 3D network, the 2D network is trained to validate the hyperparameters. This validation network is shown in appendix A and is describing the most prominent techniques and hyperparameters.

Network architecture

The network, as shown in figure 4.1, contains a similar structure as the 2D NN. Although the size of the input image has changed, the deep NN layers are adapted to the additional input dimension. This results in a 4D hidden layer which is reshaped into a vector representation of 16 304 parameters.
The latent space consists of 2048 parameters; this is four times as much as the validation network because the number of input parameters is also quadrupled. The encoding part of the network reduces the number of inputs from 1 048 576 to 2048; this means that the dimension reduction is 99.8%, which is similar as for the 2D CNN.

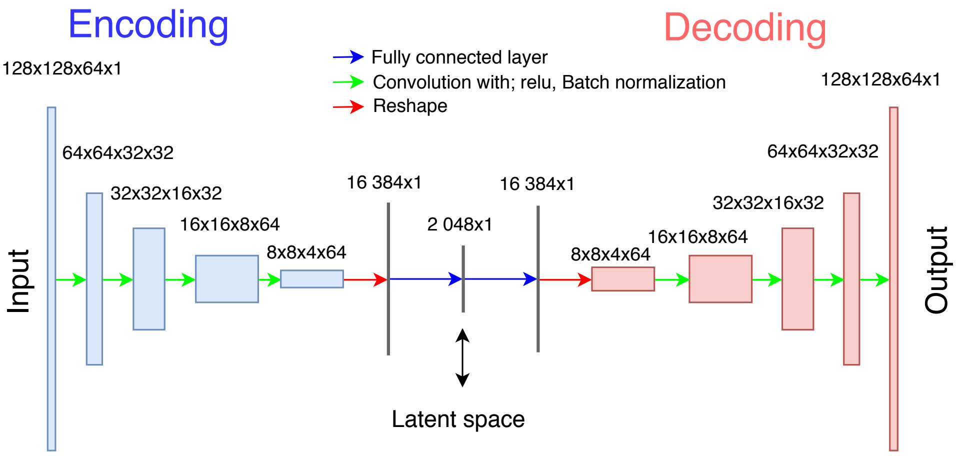


Figure 4.1: Graphic presentation of the shape of the neural network for encoding an input image of size 128 × 128 × 64 to a latent space of 2048 variables.

The graphical representation of the network can be split into three parts. Firstly, the encoding part of the network, this part reduces the input dimension to a desired latent space. Secondly, the latent space representation can be either extracted or entered the network or both. Lastly, the decoder part reconstructs the latent space to the original input shape in an identical form as the encoding part.

The different arrows within the figure are indicating the operations which are applied to the data. The operation applied by an arrow is indicated by its color. The boxes are indicating the corresponding dimensions of the data. The convolutional layers are using a stride of 2 with a kernel of [3 × 3 × 3].

4.2.4 Visualization

The 3D data should be visualized to visualize the PRM in a higher dimension. This check can be done in several ways, but the 3D data is harder to inspect visually. Therefore we use two methods: The first is plotting the bright values in the ultrasound image in a plot since these areas are mostly the muscle areas.

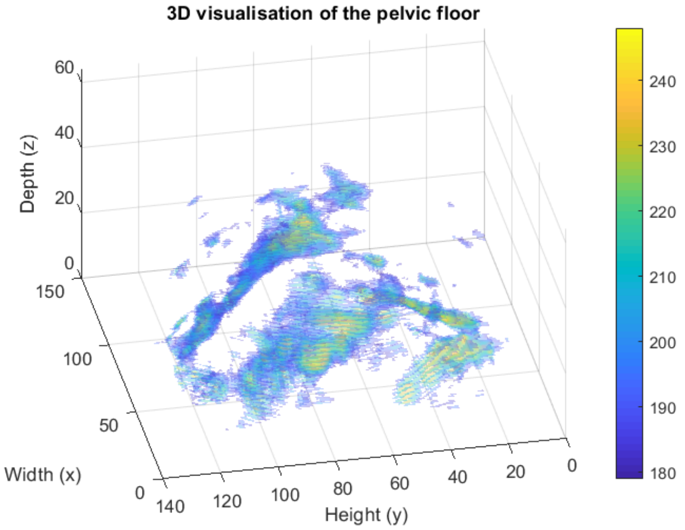


Figure 4.2: 3D plot of the pelvic floor, with a non-normalized voxel threshold of 178 in the range from 0 to 255

The PRM is highlighted in figure 4.2, where only the higher voxel values are displayed.

Secondly, we inspect the 3 center slices of the x,y, and z view of the 3D image. Selecting the mid-plane which is in line with 2 of the 3 axes, as shown in figure 4.3, results in a 2D slice of the pelvic floor. Over the remaining axis can be sliced to see the characteristics in that specific direction. This slice gives the information within this specific plane, while only a small part of the 3D image is shown.

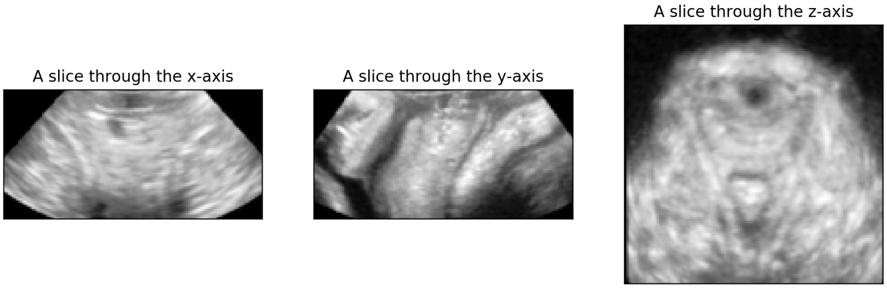

A slice through the x-axis
A slice through the y-axis
A slice through the z-axis

Figure 4.3: Visualizing the 3D plot by slicing the plot in all mid planes over x, y and z.

Figure 4.3 shows the slice through the midplanes of the x, y, and z dimension. The midplanes are chosen because these are displaying clinical relevant information and are the most consistent over all the different patients. This consistency captures the small variations within the women concerning the pelvic floor.
Slice through the z-axis are looking most familiar with the shown data set of chapter 2 but the PRM is less visible. This is because these images are taken in the same plane while rotating around the y-axis. This introduces a different size shape of the images and creates better visibility of the PRM.

4.2.5 Dimension reduction of latent space

The latent space is further reduced in size by applying the dimension reduction methods PCA and t-SNE. The reduced data consists of two dimensions describing a single input image.
When the 3D input images are reduced to a 2 dimensional scatter plot, the input data has been reduced by a percentage of

$$\rho = (1 - \frac{2}{128 \cdot 128 \cdot 64}) \cdot 100\% = (1 - 2^{-19}) \cdot 100\% = 99.9998[\%]. \tag{4.1}$$

Euclidean distance

This reduction results in the sub-clustering of the different states of contraction within the video. The difference between the states of contraction can be found by looking at the frames which are describing the highest amount of displacement within the Euclidean distance. This high displacement is describing the transition frame between the different states of contraction. The Euclidean distance is calculated by taking the squared sum of all individual components and apply the square root to this cumulated value.

The Euclidean distance between points $\boldsymbol{p} = (p_1, p_2)$ and $\boldsymbol{q} = (q_1, q_2)$

$$d(\mathbf{p}, \mathbf{q}) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2}, \tag{4.2}$$

is the euclidean length between those points in two dimensions. Where, q_1 and q_2 are the two data points describing a specific frame in the two dimensional representation. Similarly to the variables p_1 and p_2.

The euclidean can also be calculated over all the dimensions of the latent space (N), where $\boldsymbol{p} = (p_1, p_2, \ldots, p_N)$ and $\boldsymbol{q} = (q_1, q_2, \ldots, p_N)$:

$$d(\mathbf{p},\mathbf{q}) = \sqrt{\sum_{i=1}^{N} \left(q_i - p_i\right)^2} \tag{4.3}$$

where the distance between all points is calculated.

4.3 Analysis

The latent spaces, as obtained from the autoencoder neural network, are examined by labeling the different frames of a patient. The frames are labeled according to the three different states of contraction, assigning a transition state between the states of contraction, and the undefinable frames are also labeled.

Table 4.1: Explanation of the assigned labels to different time frames.

Name	Definition
Rest	Rest is defined as the state where no muscles are applying a contraction.
Contraction	Contraction is defined as the state where the PRM is contracting towards the rectum. This is visualized on the orthogonal section over the y-axis.
Valsalva	Valsalva is defined as the state where pressure is put on the vulva and pushed towards the bottom of the rectum. This can also be visualized on the orthogonal section over the y-axis.
In between	The states between the rest - contraction - Valsalva where the greatest movement is visible is defined as "In between".
Undefined	When a woman is applying a state which is not in line with rest, contraction or Valsalva, these states are defined as undefined.

The recording of the ultrasound is started when the patients are at rest. Then they are asked to perform a contraction movement and a Valsalva movement sequential after each other. Between those states, contraction, and Valsalva, the patient is returning to the rest state to apply the Valsalva. This will lead to the following sequence of states:

Rest - In Between - Contraction - In Between - Rest - In Between - Valsalva - In Between - Rest

4.4 Results

4.4.1 Network result parameters

The network parameters are the parameters which can be extracted during training by analyzing the training loss and validation loss. These losses are defining the performance of the trained network. These parameters are giving the network's capability to reconstruct the input-output relation.

First frame analysis

The network is trained using only the first frame as input parameters. The training of the network results in a training error of 0.00299 with a validation error of 0.00533. From these errors, the average error can be calculated to the average pixel error by taking the square root to the loss value. This can be done because the loss function is defined as the mean squared error.
The average pixel error (e) of the validation loss reached after 101 epochs is:

$$e = \sqrt{0.00533} = 0.0730 \tag{4.4}$$

This pixel value error is the average error of the pixels from the input image, where the input images have a range from 0 to 1. This error is similar to an error of 7.3%.

All frames analysis

Similar to the results of the first frame, the validation loss and the training loss are 0.00104 and 0.000886. This is achieved within 491 epochs and results in an average pixel error of

$$e = \sqrt{0.00104} = 0.0322 \tag{4.5}$$

in the range of 0 to 1. This results in a pixel error of 3.2%.

4.4.2 Visualizing results

First frame analysis

The first frames of all input videos are encoded to the latent space representation. This first frame can be analyzed to see the differences between the women.

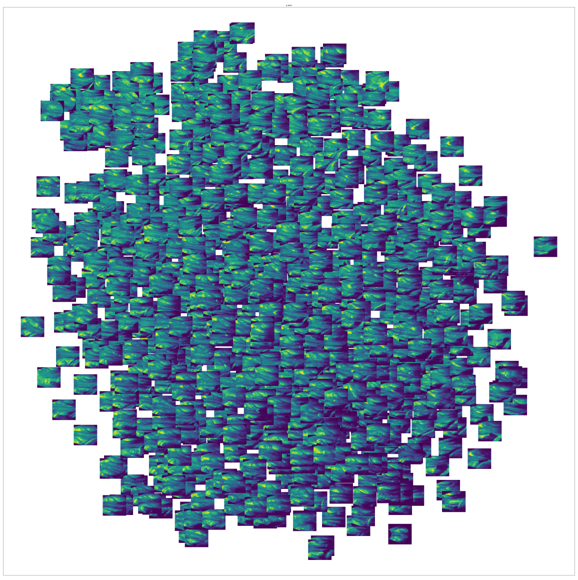


Figure 4.4: The ultrasound images plotted over the t-SNE satterplot, where the sectional x,y and z planes are shown.

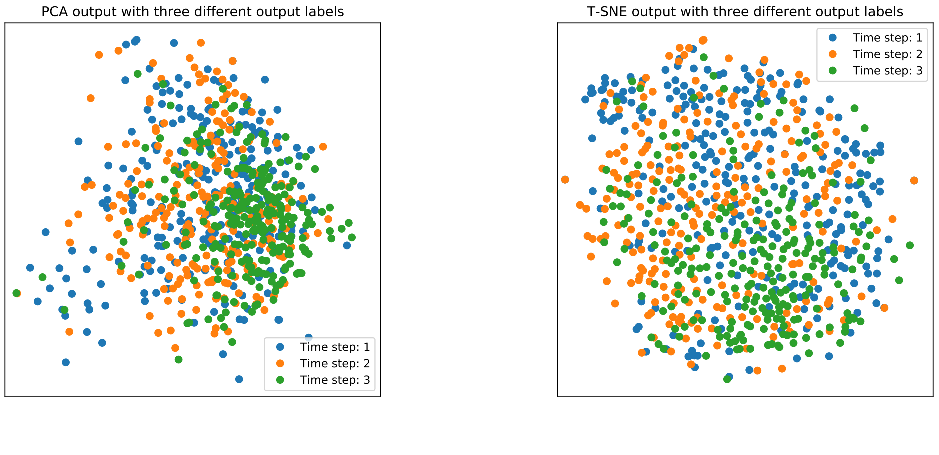
The different pictures as shown in figure 4.4 are displaying the cross-sectional visualization, as described in figure 4.3. This scattering of images shows the distribution of the most prominent feature. The most prominent feature is indicating to be the gradient in intensity from the upper left to the bottom right. Besides this gradient, no clusters are visible by looking at the figure, and the ultrasound images are equally distributed over the images.

4.4.3 Dimension reduction of latent space

The latent space representation is extracted from the network to analyze the hidden features within this vector. The vector analysis is done on both datasets; the first frame of all women and all frames of a smaller group woman.

First frame analysis

The first frame of the ultrasound videos is analyzed by extracting the vector analysis from the neural network. This vector analysis has been reduced in size by applying PCA and t-SNE. The results of the reduced size representation are shown in figure 4.5.



(a)

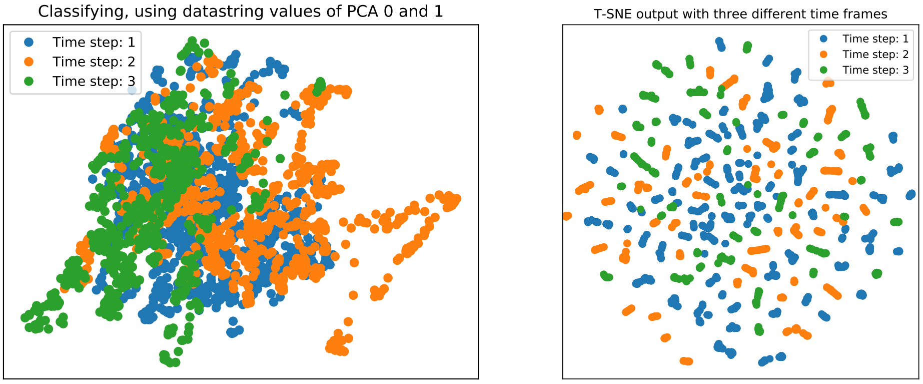
(b)

Figure 4.5: (a) PCA analysis and (b) t-SNE analysis of the latent space from the first frames. In which the three time frame labels are indicated.

From figure 4.5 (a), it can be seen that clustering is hard using the PCA algorithm. While analyzing figure 4.5 (b) the rest, contraction, and Valsalva seems to be closer related to the plot using the t-SNE algorithm. The first time frame, 12 weeks pregnant, is slightly more visible in the upper right corner. The second time frame, 36 weeks pregnant, is more visible in the left part of the point cloud. 6 Months after delivery is more present at the bottom of the scatter plot.

50 patients all frames analysis

PCA and t-SNE are applied to the encoded latent space of 50 random patients. The results of this PCA and t-SNE analysis are shown in figure 4.6.



(a) (b)

Figure 4.6: (a) PCA analysis and (b) t-SNE analysis of the latent space from all input frames. In which the three time frame labels are indicated.

From figure 4.6 (a), it can be seen that the points are oriented in lines. This is because the patients are having similarities in their frames over time while applying a different state of contraction. This similarity in frames can be seen while the distance between the different frames can vary. A low euclidean distance between the images indicates that the different states have a similar state of contraction in the video.

When the woman applies a single state of contraction, the images are having more commonalities and are closely related to each other, which can be seen in the figure. These frames are closer to each other and covering less distance. To calculate the euclidean distance between all the frames of the same patient, the distances between the states of contraction should be higher.

Figure 4.6 (b) shows the latent space clustering of multiple input frames clustered by the t-SNE algorithm. From this picture can be seen that there are many small clusters of frames. This means that there are a lot of small clusters which have a highly similar latent space. These latent spaces are clustered together because these input images are close together and thus be in a similar state of contraction.

Tracking of the frames

The frames are followed over time to see the different states of contraction in the video. Movement of a picture over the principal components is visible as described above. The movement of patient 4 is chosen to see the displacement of the frames over the lower dimension.

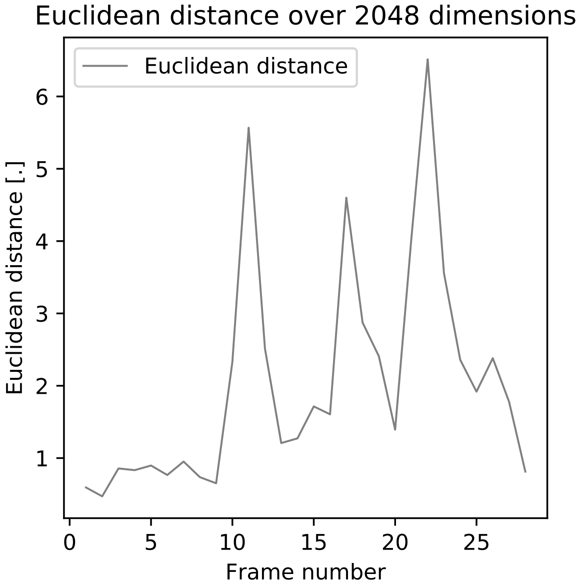


Figure 4.7: Euclidean distance of patient 4 over all 2048 dimensions of the latent space.

From this patient 4, as shown in figure 4.7, it can be seen that the euclidean distance between the multiple frames is different. A low euclidean distance indicates that these frames are close together. When the Euclidean distance is larger, the frames are further apart. By looking at the graph of figure 4.7 there are three spikes visible. These spikes are indicating a higher transition between the frames. This is indicating the transition between rest, contraction, and Valsalva. The distance of the euclidean distance is calculated for each frame with respect to the next frame. The euclidean distance, mentioned on the vertical axis of figure 4.7 does not have a physical meaning because it is the movement of a point in a 2048 dimensional latent space without physical meaning.

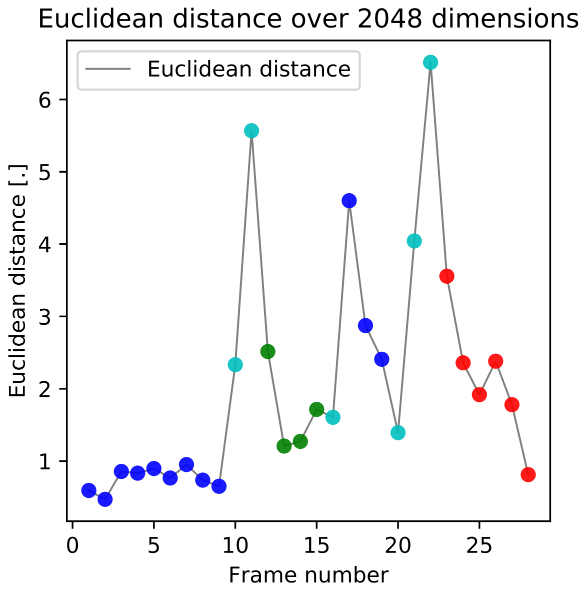
4.4.4 Analysis of results

By using the labels defined in the previous analysis section 4.3, the euclidean distance can be labeled. This labeling indicates the different states of contraction, as well as the transition frames.

All frames analysis

Looking at the results obtained in figure 4.7, the peaks in this graph are indicating the transition frames between the different states. This can be verified by looking through the video and analyzing the different states. Figure 4.8 describes the frames at the time-frames between the different transition frames.

Figure 4.8 shows the Euclidean distance obtained by the latent space. At every frame, a colored dot is plotted, which indicates the state of contraction of that frame. These dots are obtained by manually going through the dataset and assigning a label to each frame.



(a)

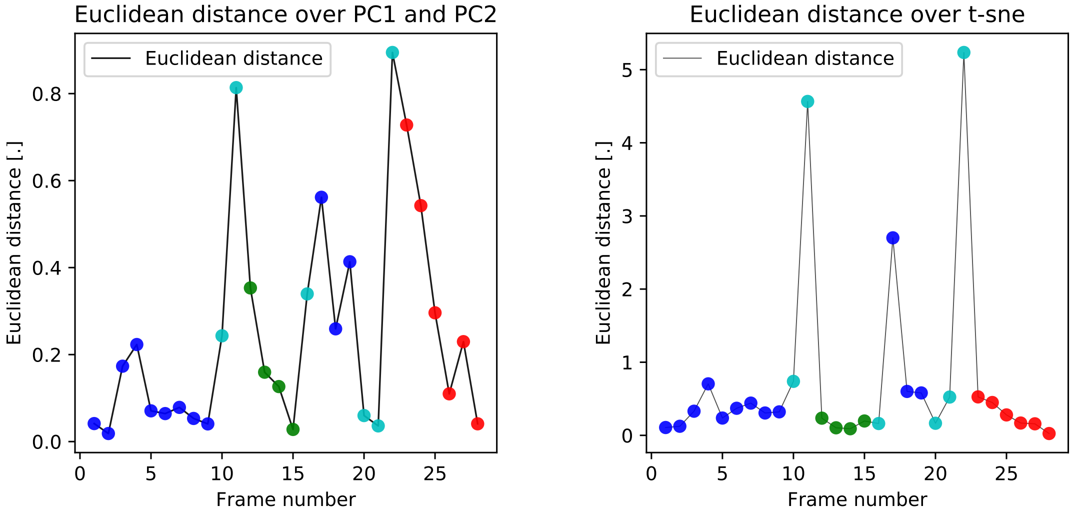
Colour	Definition
●	Rest
●	Contraction
●	Valsalva
●	In between
●	Undefined

(b)

Figure 4.8: (a) The labeled frames with (b) assigning the corresponding frame definition, together with the euclidean distance over the 2048 dimensions of the latent space.

From figure 4.8 can be seen that the patient is starting in the rest state. After this first state of contraction a transition to contraction has been made, where the Euclidean distance is peaking in combination with the frames as indicated in between.
As we go from the contraction state to the rest state, the in between state is one frame ahead of the euclidean distance. From this rest state, the transition to the Valsalva states is made.

Instead of using the euclidean distance of the high dimensional latent spaces, the euclidean distance after the dimension reduction methods can be used. This results in the labeling of the different frames, as shown in figure 4.9.



(a)

(b)

Colour	Definition
●	Rest
●	Contraction
●	Valsalva
●	In between
●	Undefined

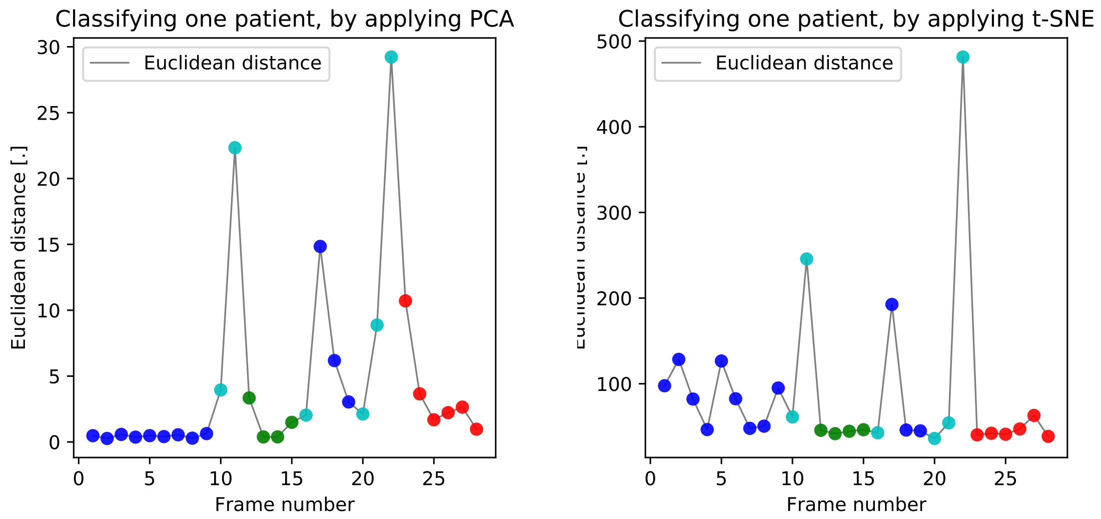
Figure 4.9: The labeled frames of (a) the PCA analysis and (b) the t-SNE analysis by applying these dimensionality reduction to the total dataset and calculate the euclidean distance of one patient in this new dimension. The coloured dots are assigning the corresponding state of contraction to every frame.

From the figures 4.9 (a) and (b) can be seen that the t-SNE algorithm is more focusing on clusters, because this algorithm creates three single frame spikes. The PCA algorithm also finds three spikes, but these consists of more frames than the t-SNE algorithm.

Dimension reduction over one patient

Both previous examples have taken the whole dataset into account. This whole covariance of the total dataset is taken into account. If only one patient is taken, the variance becomes different and also the scattering on the lower dimensions.
To see the difference between the frames of one patient, all frames of that single patient can be taken into account. The displacement of this patient is shown for both the PCA and t-SNE analysis.

The frames as classified by the classifier are scattered and not corresponding with the plot, as shown in figure 4.10.



(a) (b)

Colour	**Definition**
●	Rest
●	Contraction
●	Valsalva
●	In between
●	Undefined

Figure 4.10: The labeled frames of (a) the PCA analysis and (b) the t-SNE analysis by applying these dimensionality reduction to one patient and calculate the euclidean distance of this patient in this new dimension. The coloured dots are assigning the corresponding state of contraction to every frame.

When looking at figure 4.10 (a), three spikes are more visible compared to the PCA analysis, as shown in figure 4.9 (a). When looking at the t-SNE analysis, figure 4.10 (b), the t-SNE algorithm has a significant higher euclidean distance which is indicating that the algorithm is more certain about the clustering of the latent space. However, the relative average euclidean distance increases when this is applied to one patient.

4.5 Conclusion

Looking at the results obtained from the first frame analysis, the network does not show any clear clustering. The transition from one label to another is too scattered to create good clusters.

Looking at the results of multiple frames from one patient, the change over time shows a displacement in the lower dimensions. This displacement can be represented by calculating the euclidean distance. This euclidean is showing some perspective to analyze the different states of contraction within a video. Looking at the different frames of one patient, some more significant changes are occurring. These changes are correlating with the different states of contraction. This indicates that these clusters of low euclidean distances are describing a state-specific state of contraction. However, analyzing more patients is needed in order to confirm this finding.

Comparing the PCA algorithm by applying the algorithm to the full dataset and one patient, results in smaller spikes when applied to one patient. When comparing the euclidean distances obtained by the t-SNE algorithm, the application to the total dataset creates a better visualization. Comparing both dimension reduction methods and using both datasets, the t-SNE algorithm applied to the total dataset creates the best results.

5 Discussion

In chapter 3, we looked at the possibilities of unsupervised learning using medical input images, which results in clustering the most prominent features. These features were based on the US input shape of the images. These features are not clinically relevant, but it proofs that the network can detect features. To avoid dependency based on the 2D shape of the US image, 3D images are used. In chapter 4, we examined the 3D US images by analyzing two different types of input images, using the first or all frames of a patient. Using only the first frame, there is not a clear distinction between the different time frames. However, by looking at all frames, this is resulting in clusters within the videos. These clusters are looking promising because these seem to correlate with the contraction of the pelvic floor.

5.1 Relevant literature

In the literature, most labeling principles are based on supervised neural networks. Finding comparable literature is difficult; more research has been done about semi-supervised neural networks. These networks are combining the supervised network with an unsupervised autoencoder network. [Wang et al., 2019, Guo et al., 2017]

Semi-supervised neural network analysis, where a neural network is trained according to an autoencoder network together with a convolutional classification network. This forces the network to cluster according to the labels used for the classification network. This might help in the future to create a network which is more focusing on relevant clusters instead of the most prominent. This helps to create strong results of clusters, although, the dataset needs to be labeled for using semi-supervised neural networks.

5.2 Critial assesment

This section explains the strong and weak points of the research. First, the strong points are elaborated whereafter the weak points are elaborated.

The data reduction achieved by the network and dimensionality reduction methods is significant. By only using 2^{-19}% of the parameters, the differences between the patients can still be visualized.

Without using any labeling information, clusters can be created in a scatterplot. These clusters can be created by looking at the scatterplot and examining the most prominent features, or by looking at multiple frames of a specific patient. The most prominent feature proofs that clustering of this feature is possible; however, this feature can be irrelevant. When multiple frames are clustered, the frames have many similarities, and the network is able to find the differences between these frames. Furthermore, the neural network is able to use unlabeled input data and create clusters based on the most prominent feature. These clusters can be used to identify these clusters and label the corresponding images.

The network is trained on a large dataset, which is essential when using machine learning algorithms. A large dataset generates more trust in the images, and a wider variety of features caused by patients can be captured.

Having explained the strong points, some weaker points of the research is the complexity of the dataset. The used medical input dataset is containing more complex features than expected. This dataset has a lot of prominent features, such as US size, which an autoencoder network can amplify. To reduce this effect of the most prominent features, the input dataset should be as homogeneous as possible without significant differences over the input images. Similarly to the complexity of the dataset, the used data is homogeneous distributed, which means that input images are based on three-time moments, where both the first and second are healthy women.

Only the third time frame some women may have an avulsion, while most of the women did not have any physical complains yet. It is desirable to have a more evenly distributed ratio between healthy women and patients with an avulsion to create a classification network.

Besides the input data, the network's architecture is kept similar to the network architecture as used for the 2D images. This is done to avoid a long computational time, but this network structure might not be optimal for the 3D images. Extensive analysis has to be performed what the influences are of a convolutional layer and how many may be positioned after each other.

6 Conclusion

Answering the first sub-research question: "How can we use unsupervised learning to get the most relevant features using ultrasound images?" An unsupervised neural network is extracting the most prominent features. An autoencoder network can encode these features to a latent space representation. This latent space representation is reduced to two parameters, which can be plotted on a scatterplot. Unsupervised learning uses all features as available in the input dataset. The unsupervised network automatically defines the features used by the network. The network uses the most prominent features for the reconstruction of the output image. It was found that the most prominent features are the shape and size-dependent features.

However, the network was able to find the most prominent features of the input images. The variation of this most prominent feature are visible in the lower-dimension after applying a dimension reduction method.

Reflecting on the second sub-research question: "Do the most prominent features contain clinically relevant information?" By analyzing the 2D dataset, the most prominent features are based on the size and shape of the ultrasound image. The size and shape are the most prominent features in this dataset. However, if we look at the 3D dataset, the multiple frame analysis shows clustering of similar frames. The difference between multiple frames of a single patient can describe the change of the most prominent feature. Because all frames of a single patient contain many similarities, the most prominent changing feature is the variance in contraction. This variance in contraction is displayed by calculating the euclidean distance to amplify the different clusters.

Taking both sub-research questions into account, the main research question *"How do we obtain clinically relevant information from pelvic floor ultrasound images using unsupervised deep learning?"*, can be answered as follows:
The results of this research are not able to create a distinction between clinically relevant parameters. But, the multi-frame analysis is showing a promising result to track the motion of a muscle. This tracking of a muscle can be usefull in future research to determine the capabilities or state of a muscle.
Because the network uses the most prominent feature of the input dataset, the network might work to a less homogeneous dataset. This dataset, where the most prominent features are clinically relevant, can be better to distinguish between clinical features.

7 Future work

We want to detect clinically relevant features from the 4D input dataset. The designed architecture was able to find the most prominent features of the input dataset, but these were not clinically relevant. By analyzing the difference between multiple frames, it is possible to create a distinction between the frames.
This research has given more insight into the capability to use neural networks, in particular, unsupervised neural networks, to extract clinically relevant features in the input space.
To further develop this analysis, some recommendations are given to improve quality.

- Create a recurrent neural network to analyze all different input images of a single video at once. The variety of the latent will increase because it uses all the different input images. However, this should be investigated to avoid the last images having the most influence on the outcome of the latent space. When these last images are having a significant influence on the latent space, the Valsalva is greater represented than other states of contraction.

- Creating a supervised CNN to classify the 3D images of the video by assigning the different states of contraction. For this type of classification, the labels should be assigned to each of the individual frames. This network can be trained to determine the state of contraction within the videos automatically.

- Creating a dataset with more avulsions would be beneficial because machine learning uses an extensive amount of data to be able to create a right prediction of labels. The current dataset has a limited amount of labeled avulsions because the avulsions are only visible in the third time frame of some patients.

- Create a semi-supervised network, which updates the weights according to the output images as well as using the labels of contraction. This forces the network to cluster according to the labels. This happens because the latent space is updated by the reconstruction of the output and by the classified label.

- Currently there is a new dataset being under development within the UMCU, consisting of different videos. These videos are sorted by the state of contraction (contracted, Valsalva); this is giving more insight into labeling, and these videos do not have to be labeled anymore.

- Use more input parameters such as BMI, length, and weight to see whether these input parameters are effecting the classification. These input parameters can be used next to the input images to be less dependent on the US images.

- Use a segmentation autoencoder neural network to examine the latent space representation of this network. This autoencoder requires segmentation labels, in comparison to an unsupervised autoencoder which uses no labels. The latent space of a segmentation image is more focusing on the shape of the segmentation and is displaying the most significant variance in segmentation. When the segmentation is made of healthy PRM and PRM with avulsion, then the most prominent feature is the avulsion.

Bibliography

S. Albelwi and A. Mahmood. A framework for designing the architectures of deep convolutional neural networks. *Entropy*, 19(6):242, 2017.

M. Alperin, D. M. Lawley, M. C. Esparza, and R. L. Lieber. Pregnancy-induced adaptations in the intrinsic structure of rat pelvic floor muscles. *American journal of obstetrics and gynecology*, 213(2):191–e1, 2015.

C. M. Bishop. *Pattern recognition and machine learning*. Springer, 2006.

K. Bø, F. Lilleås, T. Talseth, and H. Hedland. Dynamic MRI of the pelvic floor muscles in an upright sitting position. *Neurourology and Urodynamics: Official Journal of the International Continence Society*, 20(2):167–174, 2001.

T. Bongers. Unsupervised learning for pelvic floor ultrasound images, 01 2019.

T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham. Active shape models-their training and application. *Computer vision and image understanding*, 61(1):38–59, 1995.

J. O. Delancey and J. A. Ashton-Miller. Pathophysiology of adult urinary incontinence. *Gastroenterology*, 126:S23–S32, 2004.

J. O. DeLancey, R. Kearney, Q. Chou, S. Speights, and S. Binno. The appearance of levator ani muscle abnormalities in magnetic resonance images after vaginal delivery. *Obstetrics & Gynecology*, 101(1):46–53, 2003.

J. Despois. Autoencoders - deep learning bits number1, 2017. URL https://hackernoon.com/autoencoders-deep-learning-bits-1-11731e200694.

H. P. Dietz. Pelvic floor trauma in childbirth. *Australian and New Zealand Journal of Obstetrics and Gynaecology*, 53(3):220–230, 2013.

H. P. Dietz and V. Lanzarone. Levator trauma after vaginal delivery. *Obstetrics & Gynecology*, 106(4):707–712, 2005.

H. P. Dietz, A. Eldridge, M. Grace, and B. Clarke. Does pregnancy affect pelvic organ mobility? *Australian and New Zealand journal of obstetrics and gynaecology*, 44(6):517–520, 2004.

A. T. Grob, M. I. Withagen, M. K. van de Waarsenburg, K. J. Schweitzer, and C. H. van der Vaart. Changes in the mean echogenicity and area of the puborectalis muscle during pregnancy and postpartum. *International urogynecology journal*, 27(6):895–901, 2016.

J. Gu, Z. Wang, J. Kuen, L. Ma, A. Shahroudy, B. Shuai, T. Liu, X. Wang, G. Wang, J. Cai, et al. Recent advances in convolutional neural networks. *Pattern Recognition*, 77:354–377, 2018.

X. Guo, X. Liu, E. Zhu, and J. Yin. Deep clustering with convolutional autoencoders. In *International Conference on Neural Information Processing*, pages 373–382. Springer, 2017.

K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

G. E. Hinton and R. R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *science*, 313(5786):504–507, 2006.

L. Hoyte and M. Damaser. *Biomechanics of the Female Pelvic Floor*, chapter 2 - Pelvic Floor Anatomy and Pathology, pages 13–51. Elsevier inc., 2016.

R. Kearney, R. Sawhney, and J. O. DeLancey. Levator ani muscle anatomy evaluated by origin-insertion pairs. *Obstetrics and gynecology*, 104(1):168, 2004.

D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

K.-C. Lien, B. Mooney, J. O. DeLancey, and J. A. Ashton-Miller. Levator ani muscle stretch induced by simulated vaginal birth. *Obstetrics and gynecology*, 103(1):31, 2004.

E. A. Lyons, C. Dyke, M. Toms, and M. Cheang. In utero exposure to diagnostic ultrasound: a 6-year follow-up. *Radiology*, 166(3):687–690, 1988.

L. v. d. Maaten and G. Hinton. Visualizing data using t-SNE. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

A. H. MacLennan, A. W. Taylor, D. H. Wilson, and D. Wilson. The prevalence of pelvic floor disorders and their relationship to gender, age, parity and mode of delivery. *BJOG: An International Journal of Obstetrics & Gynaecology*, 107(12):1460–1470, 2000.

J. McCarthy. What is artificial intelligence, 2007. URL http://www-formal.stanford.edu/jmc/whatisai.pdf.

F. v. d. Noort, M. v. Stralen, and K. Slump. Automatic analysis of the pelvic floor muscles on 3D ultrasound. 2018.

M. Otcenasek, L. Krofta, V. Baca, R. Grill, E. Kucera, H. Herman, I. Vasicka, J. Drahonovsky, and J. Feyereisl. Bilateral avulsion of the puborectal muscle: magnetic resonance imaging-based three-dimensional reconstruction and comparison with a model of a healthy nulliparous woman. *Ultrasound in Obstetrics and Gynecology: The Official Journal of the International Society of Ultrasound in Obstetrics and Gynecology*, 29(6):692–696, 2007.

O. Ronneberger, P. Fischer, and T. 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.

K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

J. A. Thompson, P. B. O'Sullivan, N. K. Briffa, and P. Neumann. Differences in muscle activation patterns during pelvic floor muscle contraction and valsalva manouevre. *Neurourology and urodynamics*, 25(2):148–155, 2006.

L. J. Tuttle, O. T. Nguyen, M. S. Cook, M. Alperin, S. B. Shah, S. R. Ward, and R. L. Lieber. Architectural design of the pelvic floor is consistent with muscle functional subspecialization. *International urogynecology journal*, 25(2):205–212, 2014.

P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning*, pages 1096–1103. ACM, 2008.

W. Wang, D. Yang, F. Chen, Y. Pang, S. Huang, and Y. Ge. Clustering with orthogonal autoencoder. *IEEE Access*, 7:62421–62432, 2019.

Zorginstituut Nederland. Zorgcijfersdatabank, 2018. URL https://www.zorgcijfersdatabank.nl/.

A Appendix 1

A.1 Validation network

The 3D neural network requires a long time to be trained, so a similar 2D network is trained to verify the input-output relation as well as the hyper parameters.
The two dimensional network is used for encoding the ultrasound images to a latent space representation of 512 variables. From this latent space, the network is decoded into the original input shape of 512 × 512 parameters.

The network consists of an encoding part which encodes the images from the high dimensional data 512 × 512 into the lower dimensional latent space. This

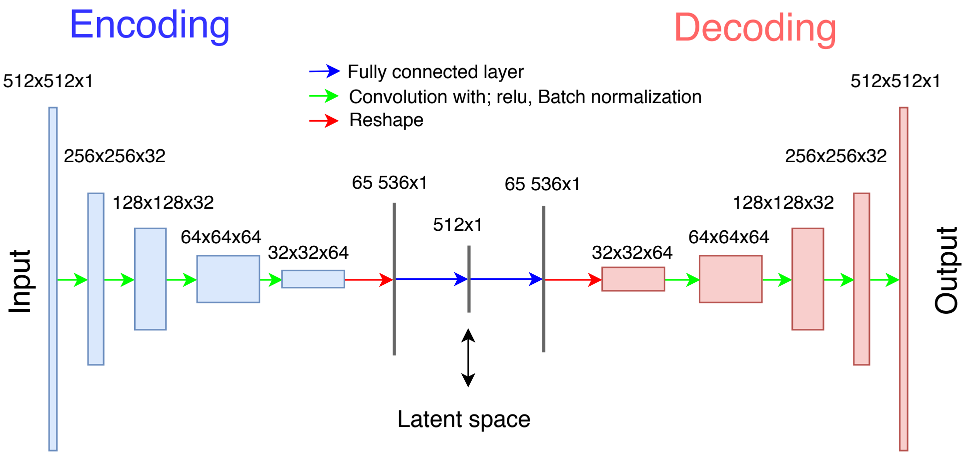


Figure A.1: Graphic presentation of the 2D validation neural network for decoding an input image of size 512 × 512 to a latent space of 512 variables.

The autoencoder is shown in figure A.1 consists of 10 layers from which are 8 convolutional layers and 2 are fully connected layer. The network uses first four convolutional layers to extract features from the input image. Convolutional layers are able to find a combination of features which will be passed to a specific channel. These channels are describing the depth of the picture. Multiple convolutions in sequential other can create a non-linear combination of features. This non-linear combination is extracted and fed through a fully connected layer. This layer connects the different combination of features to a vector based latent space. From this latent space, the network is decoded, similar to the encoding part.
This is done because it reduces the complexity of the network as well as being able to reduce the number of variations over the network.

Because the network is symmetric, the latent error should be half of the error obtained at the output layer. This increases the network's capability to create an accurate latent space representation.