Prediction of Invasive Cervical Spine Surgery Success by a Convolutional Neural Network Algorithm

A Novel Application of Machine Learning in Computer Aided Decision-Making in the Field of Neurosurgery

Leonie Pereboom

Master of Science Thesis

TU Delft Delft University of Technology

Department of Biomedical Engineering

Prediction of Invasive Cervical Spine Surgery Success by a Convolutional Neural Network Algorithm

A Novel Application of Machine Learning in Computer Aided Decision-Making in the Field of Neurosurgery

Master of Science Thesis

Leonie Pereboom

March 25, 2021

Faculty of Mechanical, Maritime and Materials Engineering (3mE) · Delft University of Technology

In collaboration with the Division of Image Processing, Department of Radiology and the Department of Neurosurgery from the Leiden University Medical Center, LUMC

Abstract

Objective: The aim of this research is to build a machine learning model in order to predict the success of invasive surgical treatment on a degenerated cervical spine, based on the baseline X-ray images. The purpose of the results of this research is an application in computer-aided diagnostics and treatment planning in the field of neurosurgery.

Background: Spinal degeneration can be described as the gradual loss of spinal structure and a decreased functioning of the spine over time. Before the diagnosis of spine degeneration can be made, a sagittal X-ray analysis of the spine is very important. With the current ageing population and the relatively high prevalence of neck pain and spinal complaints of approximately 30%, there is a great demand on MRI and X-ray analysis in healthcare. Machine learning techniques, and especially convolutional neural networks, seem promising for the application of X-ray image analysis.

Methodology: This research was preceded by a literature study about cervical degeneration and the implementation of ML in spine research. Of the available machine learning techniques, artificial neural networks show the best classification accuracy when it comes to image classification. Convolutional Neural Networks (CNNs) in particular are applicable for the computer vision classification task of this study. That is why, in consultation with the neurosurgery department of the Leiden University Medical Center (LUMC), it was decided to focus on the development of a Convolutional Neural Network (CNN) to perform the binary classification task.

Results: The final configuration of the convolutional neural network (CNN) consists of four convolutional layers with ReLU activation function and a maximal pooling function. Batch normalization was applied after the first convolutional layer of the model in order to create a more stable training environment. The false positive rate (FPR) is 19% on average and 15% during the best performing run. The ROC curve shows an AUC during the best performing run of 0.91 and 0.86 on average, whereby 1.0 would be a perfect classifier.

Conclusion: It can be concluded that this classification task could be performed by a convolutional neural network (CNN), adapted to the specific classification task. The lowest achieved false positive rate of the model was 15%, which shows a major improvement compared to the current clinical situation at the department of neurosurgery in the Leiden University Medical Center, where about 25% of invasive spinal surgical operations, involving cervical degeneration, are of no benefit to the patient.

Table of Contents

Preface

This thesis is the final assignment for obtaining the Master of Science degree in BioMedical Engineering at the Delft Univeristy of Technology. During this nine-month graduation project, I gained a lot of knowledge and practical experience in the field of machine learning, especially in convolutional neural networks for radiological image analysis. Of course, I could not have conducted this research without a lot of support and help along the way. Therefore, I want to thank the following people.

First I want to thank Dr. Rob Remis, for the thorough coaching and technical support during the graduation process, including my literature study. Despite the COVID-19 circumstances, I could always reach out to you. I am very grateful for the atmosphere of positivity and possibilities that you have created during our collaboration.

I would like to thank Merel de Leeuw den Bouter for the always fast and very helpful feedback, technical sparring sessions and support during the technical programming part of my research.

A special thanks to Caroline Goedmakers, with whom I have worked well together and from whom I have learned a lot in the field of research and publishing. I am grateful for the strong support throughout the process, the positive spirit, and the countless calls I was always able to schedule when I needed it. Your contribution in the clinical field was indispensable to achieve this end product.

Moreover, I want to thank Dr. Carmen Vleggeert-Lankamp for her advice and guidance as a neurosurgeon involved in this research. Finally, I want to thank the department of Neurosurgery and the Division of Image Processing of the Leiden University Medical Center for making this research possible.

Delft, University of Technology — Leonie Pereboom
March 25, 2021

Nomenclature

List of Acronyms

ACD Anterior Cervical Discectomy

AI Artificial Intelligence

ANN Artificial Neural Network

CA Classification Accuracy

CAD Computer Aided Diagnostics

CNN Convolutional Neural Network

CT Computed Tomography

DL Deep Learning

DNN Deep Neural Network

DRG Dorsal Root Ganglia

GPU Graphics Processing Unit

ML Machine Learning

MRI Magnetic Resonance Imaging

NGF Nerve Growth Factor

Chapter 1

Introduction

With a prevalence of approximately 30%, neck pain is in the fourth place of most common causes of physical disability world wide [1]. Neck pain can be categorized as neuropatic or nociceptive, caused by neurological abnormalities or mechanical damage of the cervical spine, respectively [1]. In most cases, acute neck pain is resolved without invasive treatment, as it is caused by accidental mechanical damage of the spine or surrounding tissue. However, in nearly 50% of patients suffering from neck pain, the pain returns or develops a chronic nature, whereby medicinal therapy or invasive treatment is needed [1]. In this chronic situation, there may be cervical degeneration, which has a progressive character and thereby increases over time. This spinal degeneration is often multi-causal and can be defined as reduced functioning of (parts of) the spine over time and a gradual loss of normal structure of the vertebrae, intervertebral discs and surrounding tissue [2]. Patients with neck pain caused by cervical degeneration often also experience neck stiffness, headache, unilateral or bilateral shoulder pain and/or pain in the anterior chest area, numbness in the arm or fingers and a general reduced range of motion of the neck [3]. Cervical degeneration can be correlated to sagittal misalignment, radiculopathy or myelopathy as well [4, 5, 6].

The procedure to diagnose a patient with 'spinal degeneration' consists of a sagittal X-ray analysis of the spine, with an additional analysis of the softer tissues within and surrounding the spine. To view the major structures and tissues of the spine, e.g. the intervertebral discs, nerves, the spinal canal, vertebral end-plates and the facet joints (zygapophysial joint), among others, magnetic resonance imaging (MRI) and computed tomography (CT) can be used [2]. This radiological analysis has to be executed by qualified radiologists. In this research, only X-ray images are analysed.

From those sagittal X-ray images a radiologist or neurosurgeon could determine the degree of spinal degeneration, based on radiological features [7]. Spinal degeneration is an important aspect in the treatment planning, as it determines whether the patient would benefit most from surgical or conservative treatment, and it influences the possible outcomes. Nevertheless, nowadays it is still very difficult for a medical specialist to determine whether the patient should benefit from invasive surgery or not [5]. The timing of the surgery is an important factor, as most non-invasive improvement should occur within four to six months [5], and on the contrary, literature shows significantly better results if the surgery is performed within six months from the time when the first symptoms occurred [8]. Because of this difficult treatment planning aspect, which strongly differs per individual, there is no clear consensus regarding invasive surgery.

Because of the risk and the invasiveness of surgery in the cervical spine area, non-operative treatment and a conservative approach are preferred. This may include physical therapy, manipulation of the

vertebral positioning, medication and timely immobilization of the neck [5]. When this approach is no longer adequate, surgical intervention may be an option. However, the recommended length of conservative treatment is unclear [5]. The specific surgical procedure, and whether this surgery would be feasible, depends on the clinical situation of the patient. Signs or symptoms that induce early surgical intervention are progressive neurological deficiencies, myelopathy (or associated clinical signs), fractures, or other signs of cervical instability, bone lesions or severe degeneration [5]. In this research, data from the NEtherlands Cervical Kinematics (NECK) Trial has been used, provided by the LUMC [9]. At the moment, based on the percentages in the NECK-trial dataset, the percentage of unnecessary surgeries, e.g. without improvement for the patient, is approximately 25%. Preventing these operations from being carried out is of clinical relevance, as this would lead to a reduction in costs and patients would be spared the risk of surgical complications and recovery time.

With the current pressure on healthcare, the ageing population and the relatively high prevalence of neck and spinal abnormalities, there is an increasing demand on MRI and X-ray imaging and analysis [1]. However, radiological analysis is a time consuming task which has to be done carefully, as diagnostic and treatment decisions are based on the outcomes. In addition, there is an inter-observer variability of 30 % on average [10], which leaves room for improvement. Automating parts of the radiological analysis could support the medical specialists to provide a more consistent outcome, with an increased time efficiency [10].

Artificial intelligence (AI) is a rapidly improving field in medical research. Over the last decade, it has become increasingly popular. Within the field of AI, machine learning (ML) algorithms seem to have great potential for computer aided diagnostics (CAD), as it is applicable to classification and regression problems [7]. An ML algorithm "learns", which means that the algorithm can improve through previous experience or provided data. This could give a valid result for data that the ML algorithm has never seen before, without explicitly programming the model [11]. Because the ML application in this research focuses on computer vision, a convolutional neural network (CNN) will be used, as this type of ML model can be used for image analysis and feature extraction [12]. Therefore, the aim of this research is to build a CNN model in order to predict the success of the invasive surgical treatment on a degenerated cervical spine, based on the baseline X-ray images. The resulting ML model algorithm would be a computer-aided diagnostic tool which could aid in the treatment planning in the field of neurosurgery. This leads to the following research question: "How could a convolutional neural network algorithm be implemented in order to predict the success of an invasive cervical spine surgery, based on baseline X-ray images?"

To become familiar with the medical terms and definitions, a short introduction with clinical information about the cervical spine and cervical degeneration will be given in Chapter 2. This part is followed by technical background information of the working principle of the used ML model in Chapter 3. The core of this report will consist of the method (Chapter 4) and results (Chapter 5), including a performance analysis of the machine learning model. Finally, the discussion, conclusion and some recommendations regarding future research will be given (Chapter 6 to 8).

Chapter 2

Clinical Background Information

2-1 Cervical Degeneration

2-1-1 BioMechanical Vertebrae Properties

Anatomical and clinical knowledge of the healthy situation in cervical vertebrae is essential to detect and analyze cervical degeneration. The cervical part of the spine consists of seven vertebral segments, denoted as C1 (cranial) to C7 (caudal) in Figure 2-1. Important anatomical components of the cervical spine can be seen in Figure 2-1 as well. Because of the lack of a numerical scale on radiological images, normative language is used to describe vertebral dimensions in this research. The cervical part of the human spine can be divided in two parts: the upper part, consisting of vertebrae C1 (Atlas) and C2 (Axis), and the lower part consisting of C3 to C7. As can be seen in Figure 2-1, those two parts differ strongly in shape. Therefore, the upper and lower part are often described and analyzed differently or in different process steps [13].

A healthy vertebra consists of a spongious bony core with a more dense bone outer layer, the cortex of the vertebra. This cortex can be seen on an X-ray image as a radiopaque edge of the vertebra [13]. Because of the increased axial forces, the thickness of the cortex slightly increases from C3 to C7, even as the anterior-posterior diameter and the height of the vertebrae [13]. If we focus on the surfaces, most of the vertebral body is concave. However, the superior and inferior surfaces of the cervical vertebrae tend to be saddle-shaped [13].

Due to the gradual and progressive nature of cervical degeneration, it takes some time to go from the asymptomatic onset of the process to the severe stage. During the process of cervical degeneration, changes in the intervertebral discs and the vertebral endplates occur. In the end, the intervertebral discs can seriously deform or even tear, causing a bulging disc or even a hernia [15].

During the degeneration process, forces and stresses on the vertebrae change. In addition, the forces are distributed unevenly across the surface of the vertebra as the mechanical properties of the tissue shift from a healthy state to degenerate tissue [15]. Due to a reduced damping capacity of degenerated discs, vertebrae are subject to wear and tear, especially at the corners of the vertebral bodies. Thereby, the degeneration process leads to remodelling and restructuring of the vertebral bodies to withstand the changing forces and spinal compression. For example, the anterior-posterior diameter of the vertebrae increases and the height of the vertebral bodies decreases [15]. In this remodelling process, osteophytes (bone outgrowth) can develop. Most of the osteophyte remain asymptomatic and occur in 20-30% of the population [13].

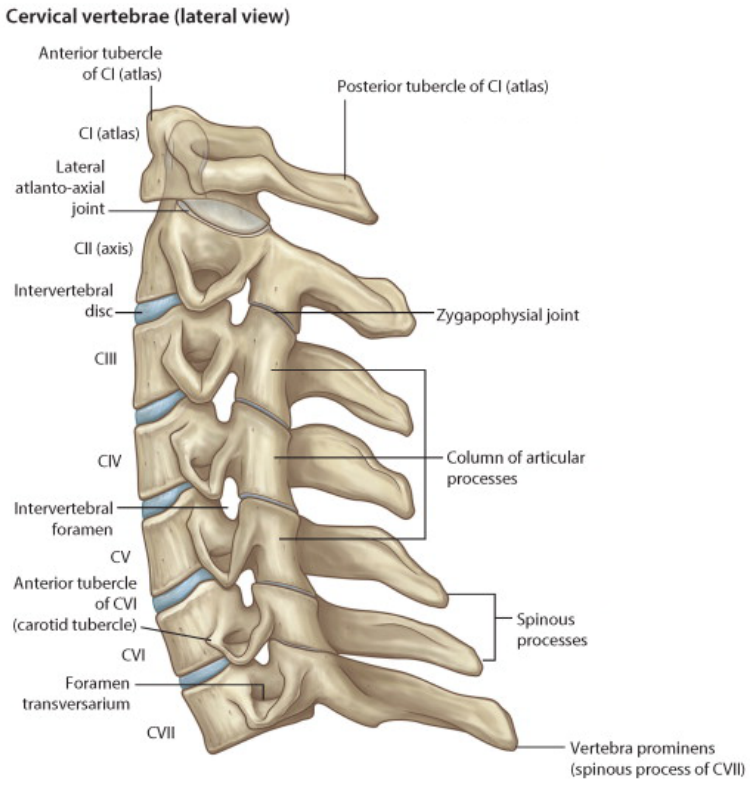


Figure 2-1: Visualization of the anatomical structures of the cervical spine segments in lateral view [14].

2-1-2 BioMechanical Disc Properties

Healthy spinal discs consist of a firm, protective outer layer of tough collagen type I, the annulus fibrosis, and a more viscous inner part, the nucleus pulposus (Figure 2-2) [16] [17]. The nucleus pulposus consists of an irregular network with collagen type II and elastine fibres, which contains proteoglygan aggregates to attract water molecules by osmotic pressure [18]. Because of the osmotic pressure in the discs, the main component of the intervertebral discs is water. The water provides the stiffness of the discs at a moment of impact or during long term compressive loading of the spine [18]. During compressive loading, some water is forced out of the discs due to the hydrostatic pressure, whereby the aggrecan concentration increases and the swelling potential of the disc grows. As a result, the disc is able to resist further compression [18]. In this way, the two layers work together as a shock and compression resistant mechanism in daily movements, and thereby protect the vertebral bones from wear and tear. A systematic visualization, together with the dimensions of a healthy intervertebral disc can be seen in Figure 2-3. However, the sizes may vary slightly by gender, age and individual anatomy. The vertebral endplate is located between each disc and vertebra, which contributes to the flow of nutrients into the discs [17], and is also visible in Figure 2-3.

Over the years, the mechanical properties of the intervertebral discs decrease over time by hydration loss [17]. This is caused by biochemical modification of aggrecan, which results in the loss of proteoglycan and glycosaminoglycans. This process induces the decrease of osmotic pressure of the disc matrix, and thereby increases the dehydratation of the tissue [16]. The ability of the disc to bear spinal compression and dampen outer impact forces is altered, causing small damages in the tissue. The outer annulus fibrosis could develop small tears, whereby the inner, more vulnerable nucleus pulposus moves closer to the edge of the disc. As a result, the disc height decreases and the disc may bulge outwards

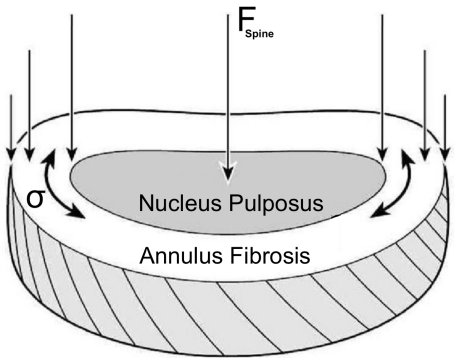


Figure 2-2: Abstract visualization of the intervertebral disc, with the compression force of the spine depicted with F_{Spine} and the tensile stress with σ[18].

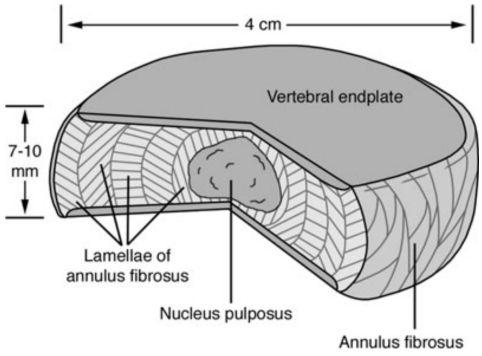


Figure 2-3: Abstract visualization, with a quarter cut-out, of the intervertebral disc. The dimensions of the intervertebral disc are 4 cm wide and 7 to 10 mm thick, as depicted in the illustration [19].

[17]. The most common types of tear on the annulus fibrosis can be seen in Figure 2-4. Over time, tears will develop even more on the outer layer of the disc, which causes more hydration loss, and so on. It is the start of the vicious circle of the degeneration process, as the intevertebral discs are not able to repair and restore the tissue damage [17]. In addition, the intervertebral end plate becomes stiffer and more brittle. Those structural changes of the tissue counteract the diffusion of nutrients to the discs, which could reinforce the degenerative process of the discs even more [17]. The result of this process can be seen on the right hand side of Figure 2-5. The following aspects can be seen in degenerative discs, in different stages of the degenerative process [15]:

1. Reduced height of the intervertebral disc
2. Increased stiffness of the disc: more brittle structure
3. Bulging
   (a) Symmetric
   (b) Asymmetric
4. Herniation
5. Inconsistent signal intensity (SI), dependent on water content and calcification of the disc

Recent studies have shown several important aspects of the composition of degenerated cervical discs, which could relate cervical degeneration to the developed neck pain [3]. The first aspect is an increased

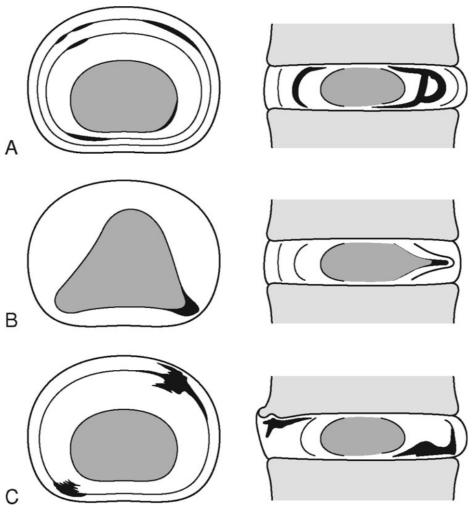


Figure 2-4: The three major categories of intervertebral disc tear in transverse (left) and saggital (right) plane: (A) Delamination or circumferential clefts, (B) Radial cleft, and (C) Peripheral lesion on the edge of the disc. Disrupted disc tissue is black in color and the nucleus pulposus is grey in color [18].

level of inflammatory cytokines, secreted by the affected discs [3]. Those inflammatory cytokines induce an upregulation of nerve growth factor (NGF), mRNA expression and thereby the secretion of NGF proteins in the disc cells [3]. Second, diseased cervical discs contain ingrowth of a large number of free nerve endings, in contrast to healthy intervertebral discs [3]. The third characteristic is the innervation pattern of the C5-C6 discs, which are innervated by the dorsal root ganglia neurons (DRG neurons) [3]. This innervation pattern could be a major aspect in the various pain complaints, because the major part of innervating nerve fibres are afferent sensory. Those nerve fibres are related to pain perception [3]. All those aspects together are remarkable, because in a healthy situation cervical discs are a-neural and contain a minimal concentration of cytokines [3].

Eventually, the degeneration process could lead to a complete tear in the outer annulus fibrosis. This is called a herniated disc (Fig. 2-6) [16]. The inner nucleus pulposis contains the inflammatory cytokines, caused by the degeneration process, which might cause severe pain when affecting the free nerve endings in the annulus fibrosis [17]. Moreover, a cervical foraminal stenosis could occur, in which the nucleus pulposis presses against the nearest nerve. A cervical stenosis might appear as well, in which the spinal cord is compressed [17]. Both stenoses are very painful and should be prevented at all times.

2-1-3 Zygapophysial Joints (Facet Joints)

Healthy Zygapophysial Joint

At every level in the spine (cervical, thoracic and lumbar) there are facet joints, also called zygapophysial joints, on the posterior side and between two vertebrae [20]. Those joints influence the mechanical performance and effect on the total behaviour of the spine. The zygapophysial joint is bilateral. The joint is positioned symmetrically with respect to the mid-saggital plane, and it is located on the motion segment, which is posterolateral relative to the spinal cord center. Moreover, it is a typical diarthrodial type of joint, consisting of bones, soft tissue (cartilage and ligaments), a synovial fluid, and an articular cavity between the joint surfaces. The cartilage surface provides a low friction force during motion of a healthy spine. A visualization of a healthy zygapophysial joint and

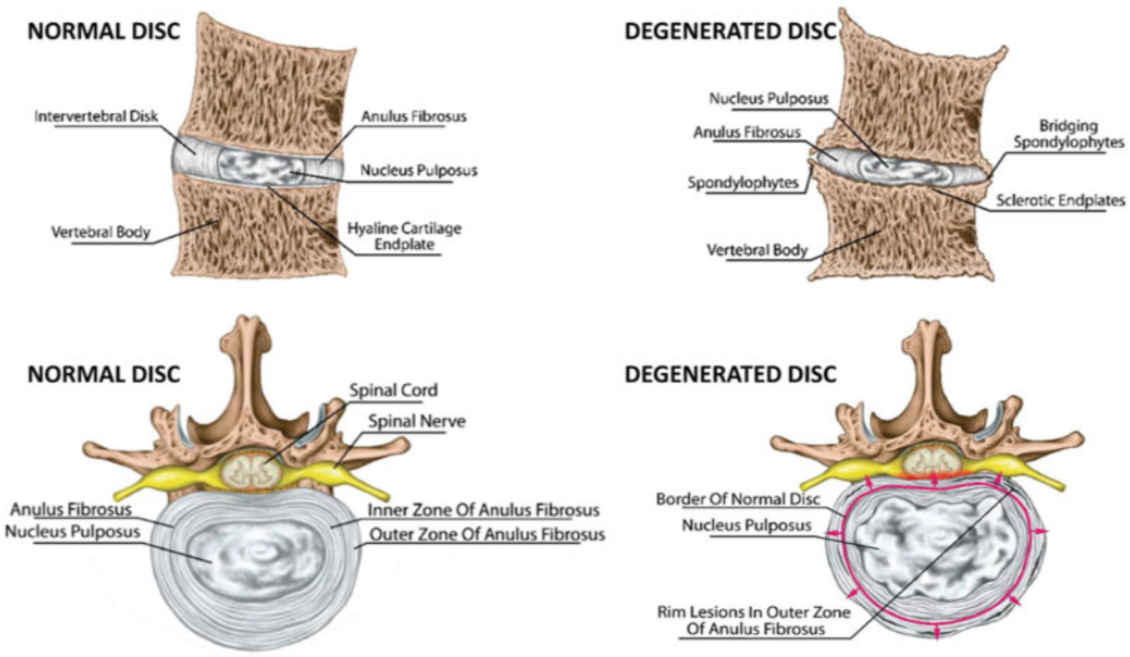


Figure 2-5: Spine and intervertebral disc anatomy in sagittal and cross-sectional views of healthy and degenerate discs [16].

its components can be seen in Figure 2-7. The joint is a key player in the functioning of the spine, as it guides and constrains the motion of the vertebrae, while it facilitates the force transmission through the spine as well during a loading cycle [20]. Hereby, the zygapophysial joint contributes to the overall stability of the spine, which is defined as the ability to maintain its alignment, and thereby protects the neural structures, during loading [20].

Affected Zygapophysial Joint

The cervical degeneration not only affects the vertebrae and discs, but can affect the zygapophysial joint as well. Similar to the abovementioned degeneration process, it is a progressive condition in the joint. During the degenerative process, both structural and cellular changes occur, whereby the mechanical properties of the joint are decreased. The mechanical properties of the intervertebral discs and the zygapophysial joint are strongly correlated, whereby degeneration of the joint will affect the mechanical properties and motion behaviour of the whole cervical spine segment [20]. It works the other way around as well, as disc degeneration can influence the overall degenerative cascades of the spine. The detection of intervertebral disc degeneration and zygapophysial joint degeneration has been reported both independently and combined [20]. There remains debate about the sequence of these events and which part, the joint or disc, is the first affected structure. However, literature has shown that degeneration of zygapophysial joints is generally preceded by degeneration of adjacent discs [20]. Although it cannot be concluded that degeneration of intervertebral discs and zygapophysial joints is always correlated, it can be assumed that those degenerative patterns are related when they co-exist in a spine.

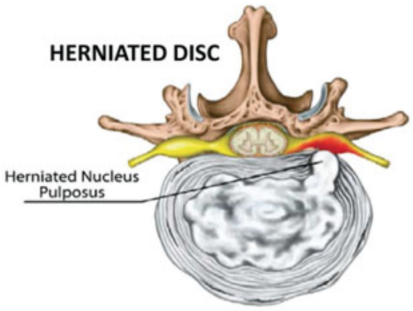


Figure 2-6: Intervertebral disc anatomy of a herniated disc in cross-sectional view [16].

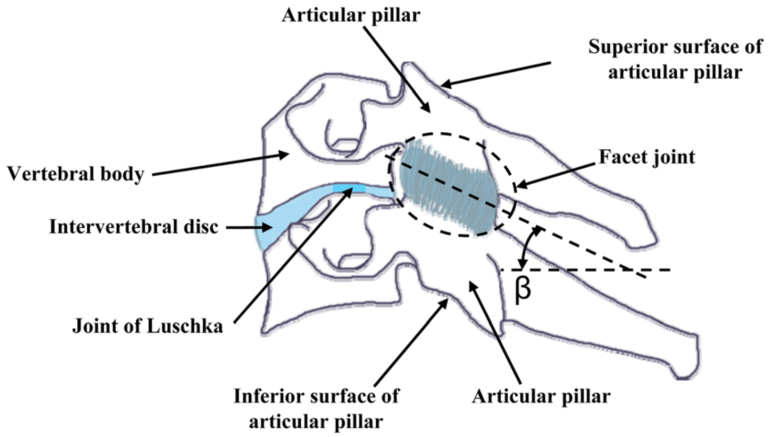


Figure 2-7: Lateral view of a cervical vertebra with the denoted general anatomy, including the zygapophysial joint and the orientation of the joint relative to its angle with the axial plane (β)[20]

2-2 Invasive Surgical Treatment

The surgical procedure which is analysed in this research is anterior cervical discectomy (ACD). The ACD procedure consists of removing the affected disc. This procedure is performed from the front, the anterior side of the body. There are several categories of ACD, depending on the vertebrae stabilisation method:

- Anterior Cervical Discectomy and Fusion (ACDF)
- Anterior Cervical Discectomy with Arthroplasty (ACDA)
- (merely) Anterior Cervical Discectomy (ACD)

Anterior cervical discectomy (ACD) is also called anterior decompression, as the aim of the treatment is to decompress the associated nerve root. The ACD procedure involves of intervertebral disc removal from the anterior side of the neck. After the disc removal, an allograft is placed to restore the height of the intervertebral space, followed by stabilisation of the vertebrae by a syntheses plate. After the recovery process, the vertebrae should be fused and the pain will slowly subside after surgery. An illustration of the surgical procedure of ACDF can be seen in Figure 2-8.

Another factor involved in the consideration for surgical treatment is the potential risks to the patients undergoing ACD surgery. As with any surgical procedure, there are risks and possible complications for

the patient undergoing the ACD procedure. Although the prevalence of severe risks and complications is low, they should still be considered. The most important risks are listed below:

- Severe blood loss by haemorrhage or haematoma
- Wound infection
- Mechanical deficiency of the allograft and/or plate, e.g. plate or graft fracture, graft migration, retracting screw, etc.
- Development of painful pseudoarthrosis
- Damage to surrounding tissue, depending on the segment involved in the surgery

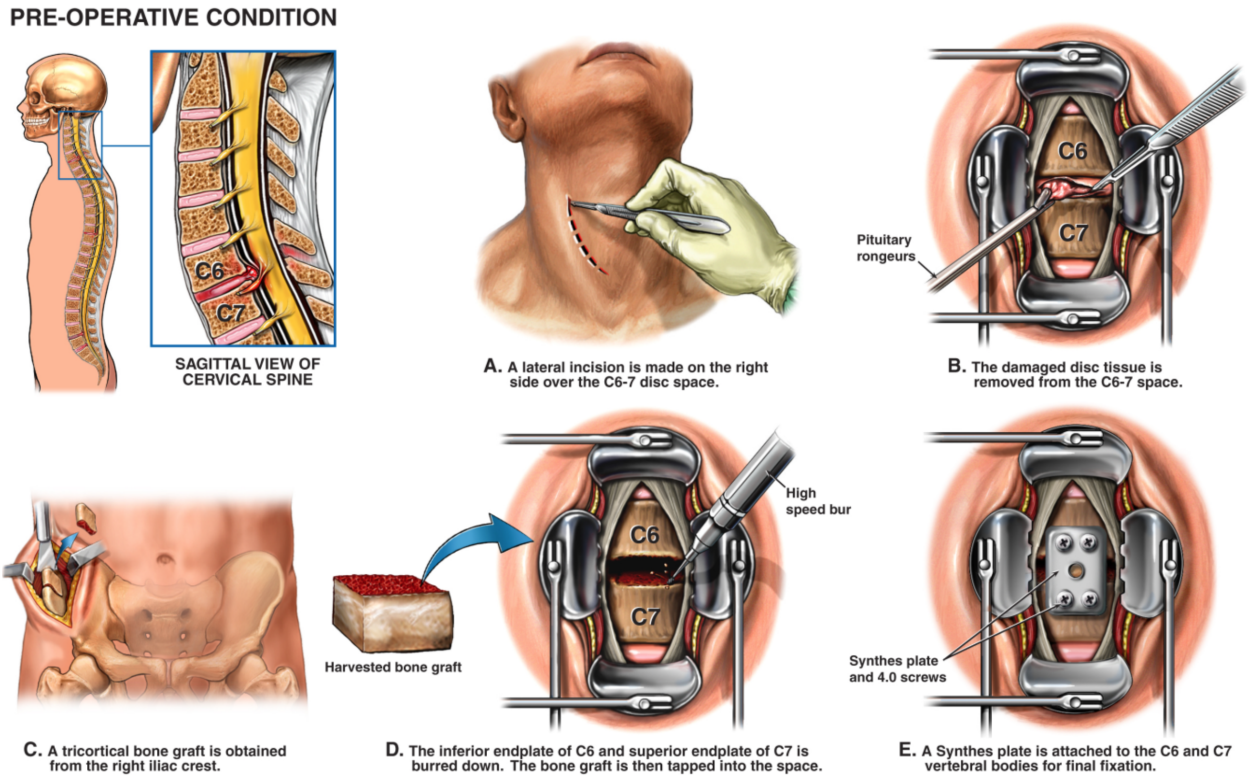


Figure 2-8: Illustration of the anterior cervical discectomy and fusion procedure [21].

Chapter 3

Technical Background Information

3-1 Machine Learning

Machine learning (ML) has gained increasing popularity over the past recent years for applications in the medical world, due to the nature of ML programming. ML is very suitable for applications that require an output based on predetermined input characteristics in the data, which is the case, for example, with image segmentation [12]. This aspect is supported by the fact that in 2015, a deep neural network defeated an expert human operator in the ImageNet Large Scale Visual Recognition Challenge, a well known image classification contest [12].

In ML, which is a specific category of artificial intelligence (AI), an algorithm will be trained by a large amount of data to do a certain task. The ML algorithm is able to "learn," which means in this context that the algorithm can improve itself by previous experience or provided/obtained data, without the input of mathematical formulas and therefore not being explicitly programmed [12, 22]. The goal of a trained ML model is to perform the trained task on data it has never seen before. The positioning of ML and deep learning (DL) with respect to each other and other important AI-related models, can be seen in Figure 3-1. In this section, artificial neural networks (ANNs) and convolutional neural networks (CNNs) are further elaborated upon, as those algorithms are used in this research.

The central nervous system (CNS) could be described as a large network of subunits, communicating with each other by electrical signals. Based on this principle, the artificial neural networks (ANNs) were developed [12]. The ANN mimics the neurons in the human brain structure, in which functions of increased complexity can be approximated by increasing the amount of implemented layers, eventually forming a deep neural network (DNN) [12]. Most recent implementations of ML are based on DNNs. Examples of applications are voice recognition, image processing, face recognition and natural language processing models [12]. Moreover, the availability of big data has been a key driver for the quick development of deep learning models [12].

Image processing and computer vision research have achieved important development steps in recent years using ANNs, both for regression and classification tasks. With X-ray and MRI analysis, ANNs are mostly used for supervised learning tasks, but they also have great potential for unsupervised and reinforcement learning [12].

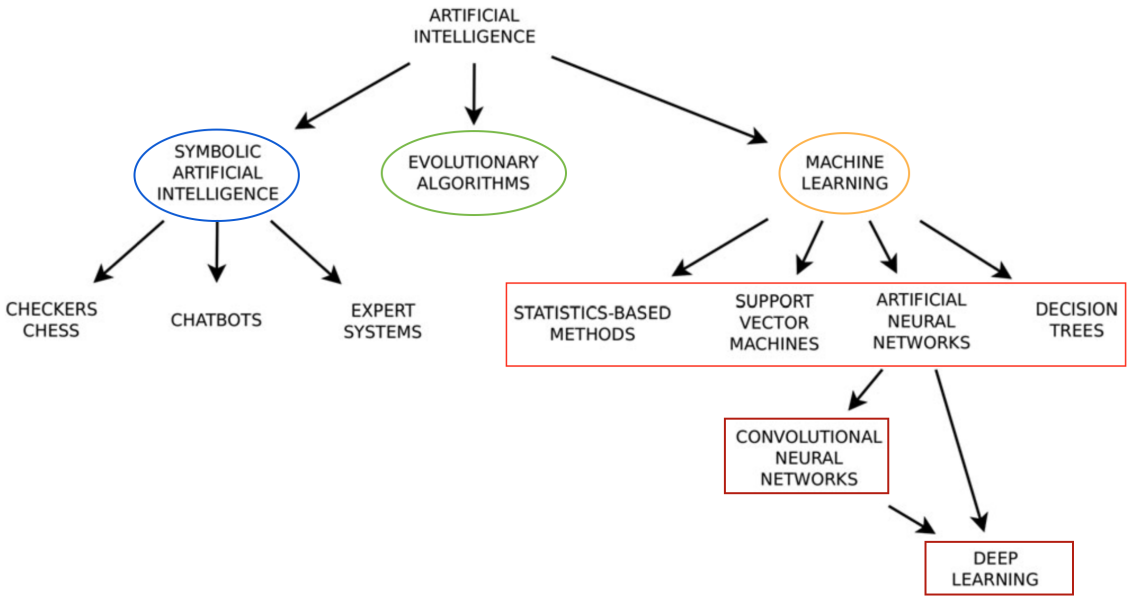


Figure 3-1: Visual summary of the main categories of artificial intelligence (AI), including machine learning (ML) and convolutional neural networks (CNNs) [12].

3-2 Artificial Neural Networks (ANNs)

Artificial neural networks (ANNs) have evolved impressively in recent years, resulting in growing opportunities in spine research. The general idea is that many computational units, so called artificial neurons, become "intelligent" by interacting with each other [22]. A visualization of the schematic structure of an ANN is shown in Figure 3-2. The layers between the input and the output layer are described as the 'hidden layers' of the neural network. It can be seen that the amount of hidden layers in an ANN (Figure 3-2, A) is smaller compared to a deep neural network (Figure 3-2, B). The addition of more layers increases the complexity of the model. An ANN consisting of three or more layers is called a deep neural network (DNN). Due to the increased complexity, the model can deal with more complex situations or patterns in the data, as there are more degrees of freedom to adapt to those complex situations. Therefore, a DNN is used for a complex form of machine learning, called Deep Learning (DL) [12].

The functioning of a DL network can be described as follows. First, a basic processing step is performed to make the training phase more stable. In this step, the input value of the first neurons will become between -1 and 1, based on the general input of the model. Subsequently, the value of a neuron in a next layers will depend on connected neuron values from a previous layer. Those connections can be seen in Figure 3-2, indicated by arrows. The fundamental formula of this process is defined by Srivastava et al. (2014) [23]:

$$y_j^{(l+1)} = f(z_j^{(l+1)}) = f\left(\sum_{i=1}^{n} w_{ij}^{(l+1)} y_i^{(l)} + b_j^{(l+1)}\right) \tag{3-1}$$

The subscript in the formula indicates a certain neuron. The superscript describes the layer in which the neuron is located. The activation function is described by $f(z_j^{(l+1)})$, in which z_j is the output of the certain neuron before the activation function is applied. The n in the summation specifies the amount of neurons in one layer. The y is the final output value of the neuron, with w_{ij} denoting the weight value from neuron i to j, and b_j a bias term [23]. In some cases, the bias is set to zero, as it is not relevant for the particular situation [24].

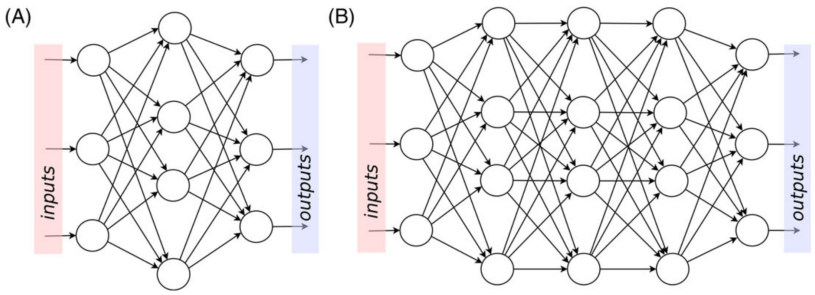


Figure 3-2: Schematic visualization of an artificial neural network (ANN) (A) and a deep neural network (B), which consists of more layers [12].

3-3 Convolutional Neural Networks (CNNs)

The structure of a convolutional neural network (CNN) is based on the neural design of the visual cortex of cats and monkeys [12]. Research of the visual cortex of those animals showed that only specific clusters of neurons are stimulated by small subareas of the visual field [12]. Within those clusters, the neurons are subdivided according to their specific task, like edge orientation, direction or shape [12]. The visual perception is a result of combining the information, obtained by the neuron clusters. Convolutional neural networks tend to mimic this neural structure. Image processing, an important part of computer vision, uses CNNs as a basis because of this strong visual orientation [12]. It is important to note that the algorithm analyses the input image based on pixel values. The CNN cannot "see" the image in a human way, but it analyses the numerical values and features captured in the image. This extraction of information is called feature learning. A schematic representation of a CNN can be seen in Figure 3-3, where an X-ray image of a vertebra is used as the input.

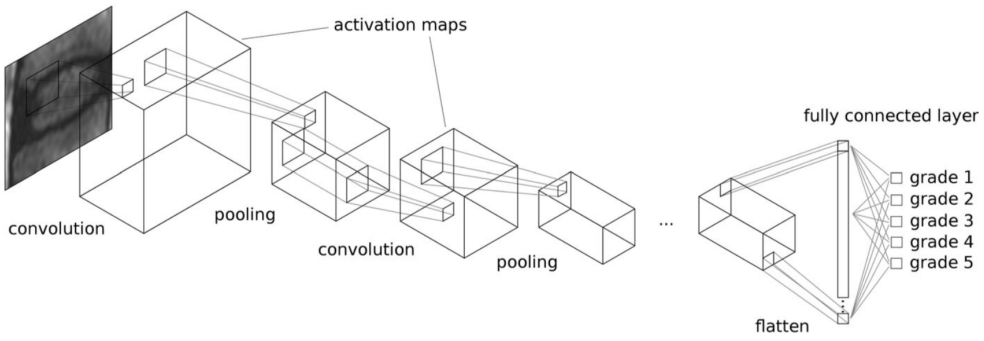


Figure 3-3: Schematic working principle of a convolutional neural network (CNN) [12].

A CNN consists of several layers [12]:

- Convolutional layer
- Pooling layer
- Flatten layer
- Dense (fully connected) layer

The operation and necessity of these layers will be further explained below.

Convolutional layer

The convolutional layer translates, or "convolves", the input image by a kernel to obtain an activation map (Figure 3-4). The kernel functions as a filter, which screens the input image (Figure 3-4, purple area: image matrix) and reproduces one output pixel per matrix field (Figure 3-4, green area). In this study, the dimensions of the kernels are 3x3 pixels, which results in a matrix of 9 pixels in total. The weights of the kernel matrix (Figure 3-4, yellow area: kernel matrix) are optimized during the training phase, which is further explained below. The result of one kernel scan is always one value, an example of which is shown in Figure 3-5. This process repeats from left to right and top to bottom, with a step of one or more pixel units, depending on the settings of the CNN model, finally resulting in one feature map. This process is performed per kernel. Each convolutional layer contains a different number of kernels, creating one or more feature maps depending on the algorithm and output settings of the model. After the final convolutional layer, the results from the final feature maps are combined to arrive at a final classification, which will be explained in the flatten and dense layer sections below.

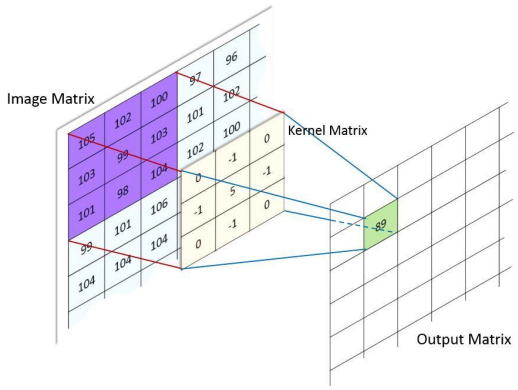


Figure 3-4: Working principle of a convolutional layer [25].

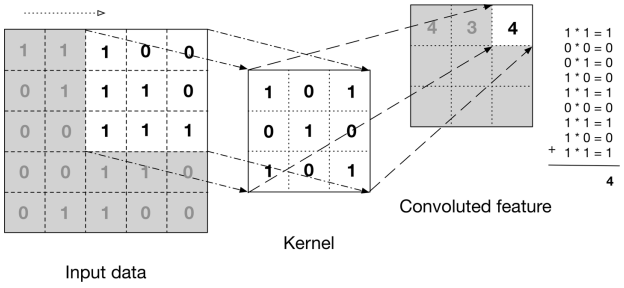


Figure 3-5: Calculation of a single value in the feature map by a kernel [26].

Pooling layer

There are several methods to perform pooling on data in a CNN structure. The most commonly used pooling methods are average pooling, minimum pooling and maximum pooling. As the names indicate, average pooling yields the average value from the 2x2 pixel grid, minimum and maximum pooling extract from the pixel grid the lowest and highest value, respectively. A comparison of these pooling methods can be seen in Figure 3-6. The comparison shows that maximum pooling results in the most distinctive features in the resulting image after pooling. The examples in Figure 3-6 show a white line on a black background, which is similar to the lighter bony areas on a black background in the X-ray image. If the situation were reversed, e.g. a black line on a white background, the minimal

pooling would result in the best distinctive features. Because of the distinctive performance, maximum pooling is used with a 2x2 filter and a stride 2 (Figure 3-7), to emphasize the anatomical features. During this maximum pooling process, the highest value of the 2x2 pixel area of the filter is extracted and stored in the reduced output image, which can be seen in Figure 3-7. It is important to note that, unlike the scanning process of the kernel, there is no overlap in the analyzed areas, because the stride is equal to the filter size, whereby all pixels are scanned once. These maximum pooling layers enable the reduction of image dimensions, without losing the important features. Because the maximum pooling layer reduces the amount of parameters, both the computational costs of the model and the risk of overfitting are reduced, which is beneficial for the performance and the overall training time.

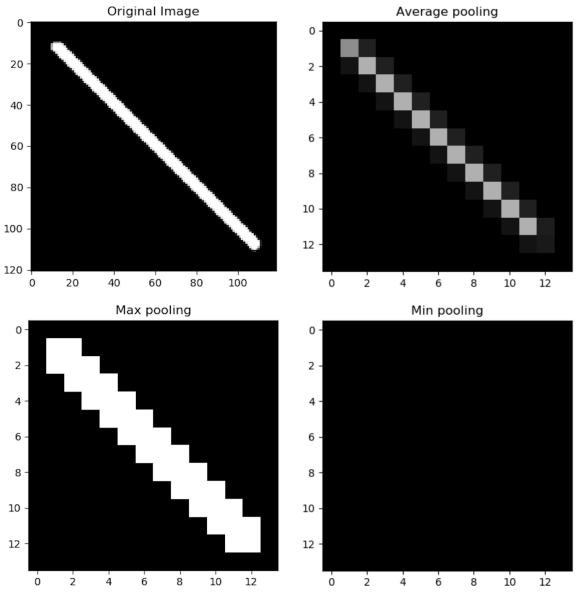


Figure 3-6: Comparison of average (right top), maximum (left bottom) and minimum (right bottom) pooling. The original image can be seen in the left upper corner [27].

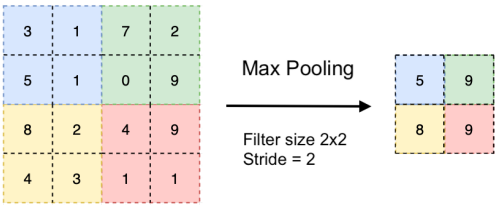


Figure 3-7: Visualization of the maximum pooling layer [28].

Flatten layer

This layer is inserted between the convolutional output and the dense (fully connected) layer. The flatten layer does not change or adjust the pixel values, but translates the two dimensional feature maps into a one dimensional vector, which is called flattening. The flattening is performed by listing the pixels from left to right, top to bottom, into the array (Figure 3-8). This one dimensional vector is the input for the final binary classifier. The flatten layer is very important, as the final output is a single, one dimensional value. Without the one dimensional vector, no final value could be calculated in the fully connected layer.

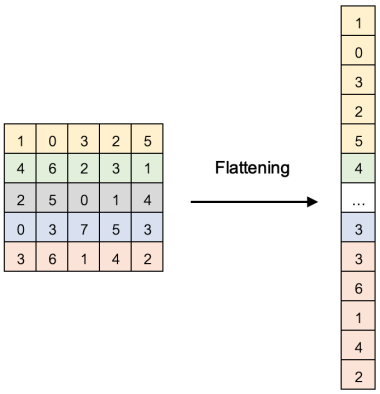


Figure 3-8: Visualization of the working principle of a flatten layer.

Dense (fully connected) layer

As explained in the previous section, every feature map becomes a vector by the flatten layer. As can be seen in Figure 3-9, each value in the yellow box has an integral connection to all values in the previous layer (Figure 3-9, orange box). When a sigmoid activation function is applied after the previous layer, the values of the neurons in the fully connected layer are between 0 and 1 [12].The fully connected layer at the end of the CNN model, combined with a sigmoid acitvation function in this research, performs the final classification task by computing the output score.

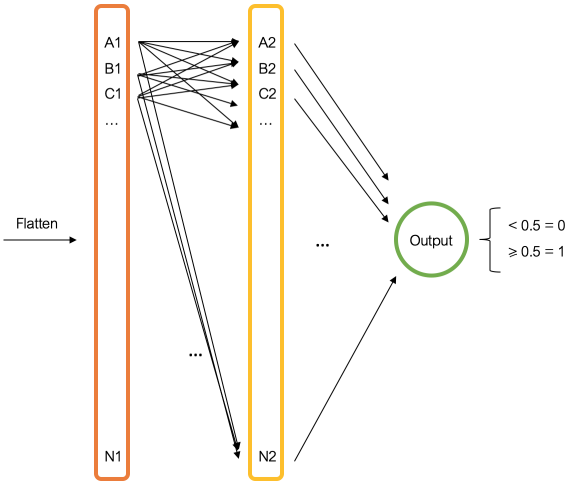


Figure 3-9: Visualization of the working principle of a dense layer, which calculates an output based on the input vector. The output in this example is from a binary classifier, which has a threshold at 0.5 between the classes.

Activation functions

In addition to the layers, activation functions are important to integrate non-linearities into the model. Those non-linear factors are essential to train complex features for the image classification. The ReLU (Rectified Linear Unit) activation function is the recommended activation function for convolutional layers, as it is most comparable to the neural network mechanism of the human brain [29]. For binary

classifiers, the sigmoid activation function is recommended in the dense, fully connected layer. The activation functions are visualized in Figure 3-10.

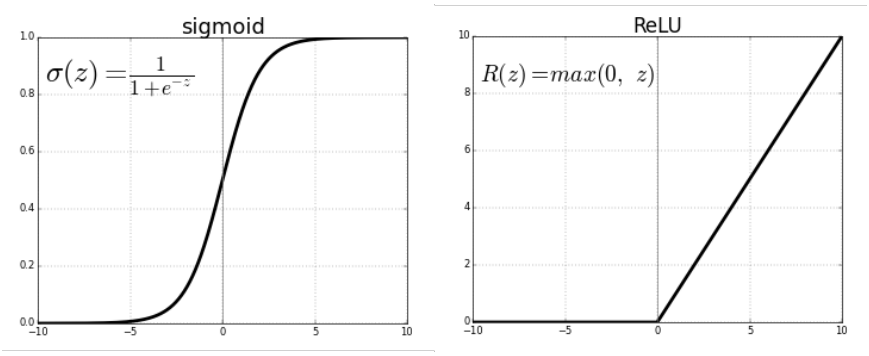


Figure 3-10: Visualization of sigmoid activation function (left) and ReLU activation function (right) [30].

Batch normalization

Moreover, batch normalization can be used to standardize the data. It is often applied after the first convolutional layer, and not as a preprocessing step of the input data. The purpose of batch normalization is increasing the learning ability of the CNN. During batch normalization, the output values of a convolutional layer are rescaled to a mean of 0 and a variance of 1.

3-3-1 CNN Training Process

The training process is an essential part to obtain the desired performance from the model. During the training phase, the weights and biases of the kernels are optimized to give the best possible prediction. Therefore, it should be monitored regularly and training parameters adjusted as necessary to optimize the training process. The extension of the training can be adapted by changing the number of learning steps per training round, the number of training rounds (epochs), the addition of dropout function, or a combination of those factors. The aim of the training process is to minimize the loss function of the model. The loss of the CNN in this research is calculated using the binary cross-entropy loss function [31]:

$$L = -\frac{1}{N}\sum_{i=1}^{N}(y_i * ln(p_c) + (1 - y_i) * ln(1 - p_c)) \tag{3-2}$$

In this equation, p_c denotes the output of the CNN for a class (c). The y_i is the ground truth and N is the total amount of samples during the training phase. This cross-entropy loss function uses a logarithmic scale, which makes this function very suitable as a binary classifier. If the true class y_i is zero (equation 3-3) or one (equation 3-4), the loss function will be as follows:

$$L(y_i = 0) = -\frac{1}{N}\sum_{i=1}^{N}(0 * ln(p_c) + (1 - 0) * ln(1 - p_c)) = -\frac{1}{N}\sum_{i=1}^{N} ln(1 - p_c)) \tag{3-3}$$

$$L(y_i = 1) = -\frac{1}{N}\sum_{i=1}^{N}(1 * ln(p_c) + (1 - 1) * ln(1 - p_c)) = -\frac{1}{N}\sum_{i=1}^{N} ln(p_c)) \tag{3-4}$$

This explains why, in the case that the predicted class p_c is the same as the true class y_i, the loss L becomes zero.

Although zero loss is not a realistic value, it is the optimal target value for the loss function of the CNN [31]. The cross-entropy loss function is very suitable for the binary CNN classifier, as it determines the performance of a model with an output value between zero and one. The loss function is large (close to one) when the predicted class is very different from the actual label, and small (close to zero) when the prediction is close to the desired output value. When all of the predictions are correct, the loss will be zero.

The CNN training process contains the following two steps, which alternate [31]:

1. **Forward phase:** In this phase, the information is processed from input to output through the CNN. Each layer will save used data, e.g. intermediate values, weights, biases etc.
2. **Backward phase:** In the backward phase, the weights and biases of the model are updated using backpropagation. If the prediction of the image was not correct, and therefore a high loss function L is determined, different values of the weights and biases will be used to obtain a lower loss in the next forward phase.

At the end of the forward phase, the loss L is determined as the difference between the output of the model and the desired output, defined by the ground truth labels of the training data. The loss of the model can be influenced by the trainable parameters, which are the weights and biases of the CNN model [33]. The aim of the training process is to minimize the loss function through backpropagation. During backpropagation, the gradient of the loss with respect to its trainable parameters is calculated [33]. Therefore, this process is also referred to as gradient descent, as this loss gradient is reduced relative to the trainable parameters. The loss gradient is used to find the minimum of the loss function,

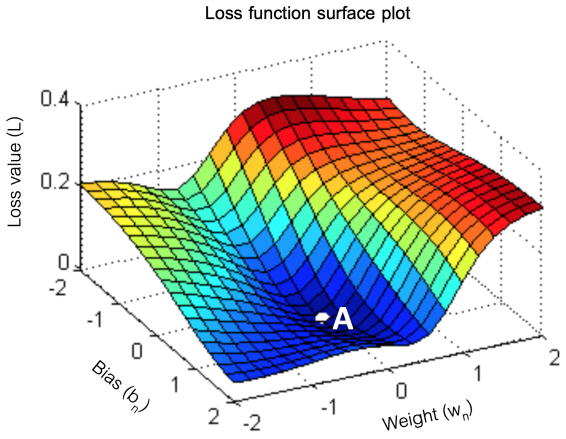


Figure 3-11: Example visualization of the loss function with its local minimum (point A), plotted against the weight and bias at a certain neuron [32].

of which a simplified visualization can be seen in Figure 3-11. The higher the gradient of the loss, the steeper the slope of the surface plot. The gradient descent algorithm iterates until, hopefully, a minimum of the loss function is found. The visualization in Figure 3-11 shows a simplified situation of the gradient descent in a CNN, as the loss function is visualised at a certain neuron (n), with one weight value (w_n) and one bias value (b_n) taken into account. The local minimum can be seen in point A in Figure 3-11, which will hopefully be found during the backpropagation process. This minimum can be determined when the tangent of the loss function reaches zero, with respect to the weights and biases, in an ideal situation, so $\frac{\partial L}{\partial \mathbf{w}} = 0$ and $\frac{\partial L}{\partial \mathbf{b}} = 0$. However, as it is unlikely that all of the loss gradients ($\frac{\partial L}{\partial \mathbf{w}}$ and $\frac{\partial L}{\partial \mathbf{b}}$) will reach zero, the training process will be stopped before $\frac{\partial L}{\partial \mathbf{w}} = 0$ and $\frac{\partial L}{\partial \mathbf{b}} = 0$ are reached. The training process should ideally be stopped just before the accuracy of the model starts to decrease or the loss starts to increase, indicating overfitting, which will be explained in the next section. At that point, the model has the smallest loss in the training cycle with respect to the training dataset [34]. The training can be stopped by a predetermined number of training cycles, or to implement a cut-off value, for example an accuracy or loss value, ending the training process.

During the backward phase, the weights and biases of the network are updated by an interative process using gradient descent update formulas. More specifically, let the vector $\mathbf{w}$ contain all the weights of the network and let vector $\mathbf{b}$ contain all of the biases. Since the loss function L is a function of these parameters, we have $L = L(\mathbf{w}, \mathbf{b})$. For given initial weights $\mathbf{w_0}$ and biases $\mathbf{b_0}$, new weights and biases are computed using the following update formulas (Equation 3-5 and 3-6).

$$\mathbf{w}_{n+1} = \mathbf{w}_n - \eta_w \frac{\partial L}{\partial \mathbf{w}}\bigg|_{\mathbf{w}=\mathbf{w_n}} \tag{3-5}$$

$$\mathbf{b}_{n+1} = \mathbf{b}_n - \eta_b \frac{\partial L}{\partial \mathbf{b}}\bigg|_{\mathbf{b}=\mathbf{b_n}} \tag{3-6}$$

In these formulas, $\eta_w > 0$ and $\eta_b > 0$ are the so-called learning rates for the weights and biases, respectively. Furthermore, $\frac{\partial L}{\partial \mathbf{w}}$ is the gradient of the loss function with respect to $\mathbf{w}$, that needs to be evaluated at $\mathbf{w} = \mathbf{w_n}$ in Equation 3-5. The gradient of the loss function with respect to $\mathbf{b}$ is $\frac{\partial L}{\partial \mathbf{b}}$, which needs to be evaluated at $\mathbf{b} = \mathbf{b_n}$ in Equation 3-6.

To explain how those equations are applied and computed, consider a neuron k in layer $(r-1)$ of the CNN and a neuron j located in layer r, as illustrated in Figure 3-12. The activation functions of

layers $(r-1)$ and r are denoted by a_{r-1} and a_r, respectively. From Figure 3-12 we observe that the output of neuron k in layer $(r-1)$ is weighted by w_{jk} and the result serves as an input for neuron j, located in layer r. In addition, a bias b_j is added to the input as well, making the total input for neuron j (z_r): $z_r = w_{jk} * a_{r-1} + b_j$. Since neuron j is located in layer r, its final output is given by $a_r(z_r) = a_r(w_{jk} * a_{r-1} + b_j)$. When the output of the network is defines as $y = a_r(z_r)$, then the sensitivity of L with respect to the weights w_jk can be computed as

$$\frac{\partial L}{\partial w_{jk}} = \frac{\partial L}{\partial a_r}\frac{\partial a_r}{\partial z_r}\frac{\partial z_r}{\partial w_{jk}} \tag{3-7}$$

Similar, we have the sensitivity of L with respect to the bias b_j

$$\frac{\partial L}{\partial b_j} = \frac{\partial L}{\partial a_r}\frac{\partial a_r}{\partial z_r}\frac{\partial z_r}{\partial b_j} \tag{3-8}$$

Note that $\frac{\partial z_r}{\partial b_j} = 1$ and $\frac{\partial z_r}{\partial w_{jk}} = a_{r-1}$. The origin of the partial derivatives in Equation 3-7 and 3-8 is visualized in Figure 3-13 A and B, respectively.

In this example, the loss with respect to only one weight w_{jk} and bias b_j is described. During the training process, a similar sensitivity analysis is performed on the loss with respect to all of the weights and biases, using the backpropagation algorithm. Backpropagation is performed all the way from the determined loss L to the input layer of the model.

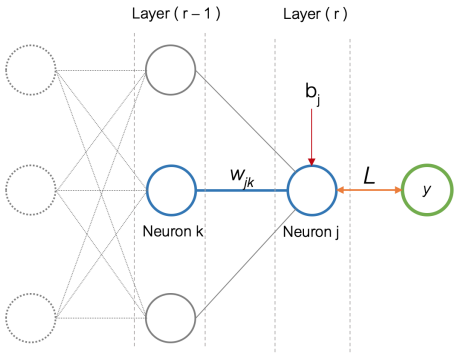


Figure 3-12: Simple neural network visualization. Blue: the connection between neuron j in layer r and neuron k in layer $(r-1)$, with an associated weight w_{jk}. A bias b_j is added to neuron j in this example. Green: desired output y.

$$\frac{\partial L}{\partial w_{jk}} = \frac{\partial L}{\partial a_r}\frac{\partial a_r}{\partial z_r}\frac{\partial z_r}{\partial w_{jk}} \qquad \frac{\partial L}{\partial b_j} = \frac{\partial L}{\partial a_r}\frac{\partial a_r}{\partial z_r}\frac{\partial z_r}{\partial b_j}$$

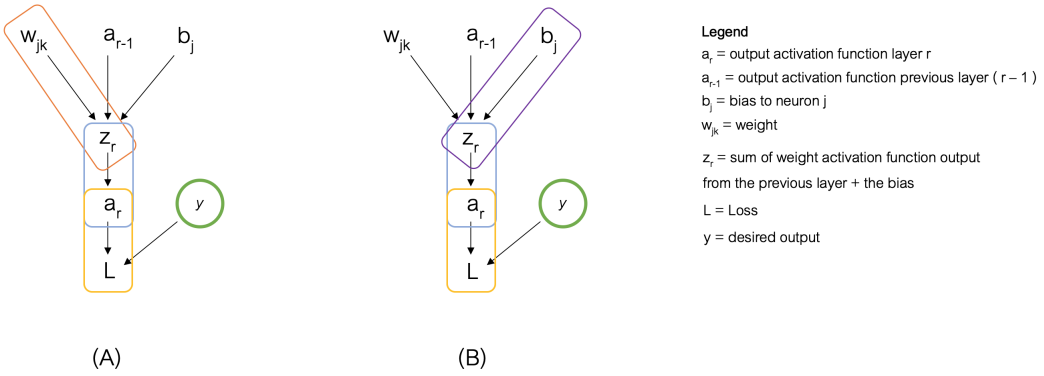


Figure 3-13: Visualization of the path from loss L to a certain weight w_{jk} (A) and to a certain bias b_j (B). The different variables are described in the legend (right-hand side). The coloured blocks indicate the certain parts of the chain-rule.

During the training process, a plausible fit is desired, which can be seen on the left hand side in Figure 3-14. A plausible fit is created when a model extracts enough features for the classification task, without focusing too much on irrelevant specific situation. Underfitting may occur if the amount of training data is insufficient, the training stage was terminated too soon, or the model is not complex enough to define the characteristics of the input data [12]. In Figure 3-14 (center) can be seen that the green line does not make a sufficient boundary between the red and blue dots, because the characteristics of the blue and red dotted areas are not extracted well enough, which indicates underfitting. When a CNN has trained for too long, or on too much specific data, overfitting can occur (Figure 3-14, right). Overfitting means that the CNN model has adapted to the noise of the specific training dataset. In this case, the results of the CNN fit the input data exactly, but the model is not able to make accurate predictions on never-before-seen data [12]. In Figure 3-14, the complex green boundary line between the blue and red dotted areas can be seen. It can be assumed that irrelevant features are implemented to determine the green boundary, which captured almost every dot of the same colour in the same area.

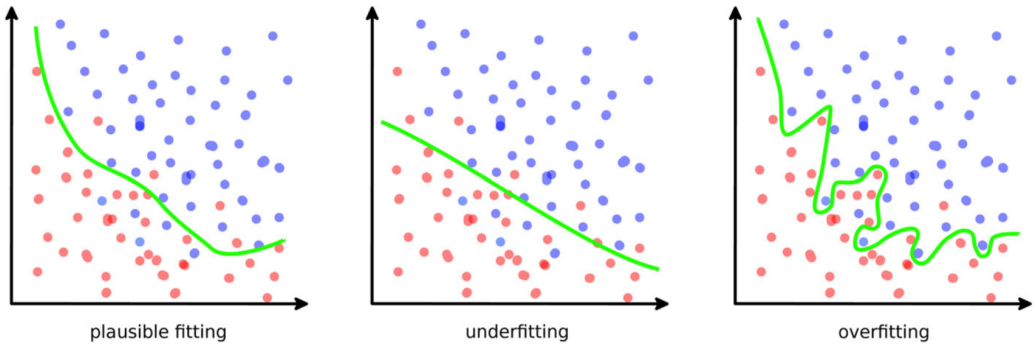


Figure 3-14: Visualization of a proper fitting (left), underfitting (center), and overfitting (right) in a binary classification (red or blue) decision task [12].

One way of preventing overfitting is to use the so called 'dropout' method, by implementing dropout-layers [35, 23]. Dropout layers can be used during the training process, by randomly switching off certain neurons for a short amount of time. This is done throughout the whole network, including all the layers of the CNN [35]. A schematic visualization of dropout can be seen in Figure 3-15. When the CNN is tested and evaluated with the validation data, all neurons are of course used. However, dropout slows down the learning process, as fewer neurons are used per iteration [35].

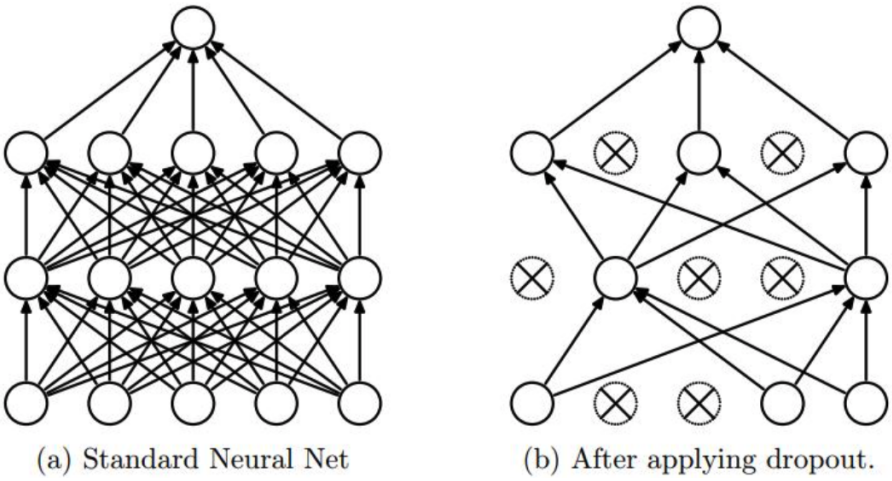


Figure 3-15: The standard ANN (a) an the ANN after dropout is applied (b) [23]

Chapter 4

Method

This research was preceded by a literature study about cervical degeneration and the implementation of ML in spine research. Of the available machine learning techniques, artificial neural networks show the best classification accuracy with image classification. Convolutional Neural Networks (CNNs) in particular are applicable for the computer vision classification task of this study. That is why, in consultation with the neurosurgery department of the Leiden University Medical Center (LUMC), it was decided to focus on the development of a Convolutional Neural Network (CNN) to perform the binary classification task. The aim of the CNN classifier is to predict the surgery success, based on the baseline X-ray images, which are acquired during the intake process of the patient at the Neurosurgery department.

4-1 Data Acquisition

The original dataset was obtained from the neurosurgery department of the LUMC. The dataset was part of the NECK-trial dataset. This dataset contained anonymized, sagittal X-ray images of the cervical spine, together with clinical data. The following information was provided per patient:

- Baseline X-ray images in both flexion (head tilted forwards) and extension (head tilted backwards)
- Follow-up X-ray images one year (12 months) after surgery, in both flexion and extension
- Follow-up X-ray images two years (24 months) after surgery, in both flexion and extension
- Success score of the surgical treatment, based on Neck Disability Index (NDI) and patient reported outcome measures (PROMs) [36]
- Clinical data, like gender, age, pain perception, freedom of movement (perception, objective/-subjective)

The original NECK-trial dataset consisted of 106 patients with clinical information. Not all of the patients could be included in this research, because of missing information or insufficient image quality. The flowchart for exclusion can be seen in Figure 4-1. From the original NECK-trial dataset, the baseline X-ray images in flexion were selected, because the aim is to make a predictive algorithm. The flexion was chosen because it is the most neutral position and the spacing between the cervical

vertebrae is optimal. Patients without a baseline X-ray image or success score were excluded from the study, a total of 31, leaving 75 patients. From those 75 patients, 5 were excluded, because the success score was missing, which resulted in 70 final patients included in the research.

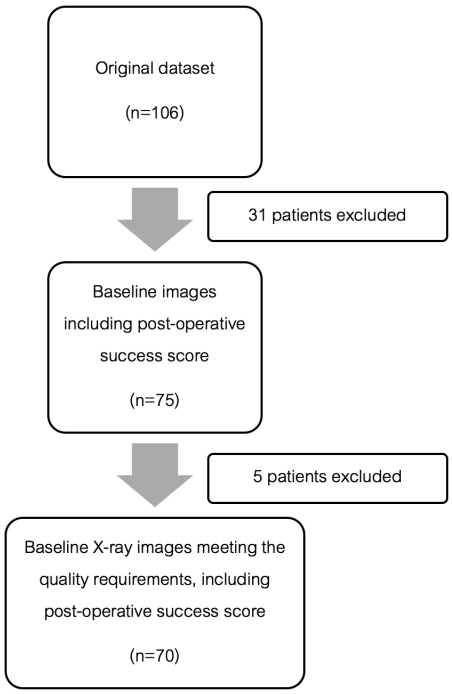


Figure 4-1: Patient exclusion flowchart

The success score was determined and provided by the neurosurgery department of the LUMC as well. Whether the operation was a success or not depended on multiple factors, based on the research of Mjaset et al. (2020) [36]:

- Neck disability index (NDI) (scale 0 - 100)
- Numeric rating scale for arm pain (scale 0 - 10)
- Numeric rating scale for neck pain (scale 0 - 10)
- Health-related quality-of-life by EuroQol (scale 0 - 1.0)
- General health status by EuroQol (scale 0 - 100)

However, the success score is a subjective measure, as it is not possible to measure all certain parameters objectively, for example pain perception. The score provided by the LUMC, with the success cut-off value based on the NDI after 12 months of surgical treatment, is considered ground-truth in this research. The threshold between the two surgery success categories, as used in the dataset, is a score of 24.2 on the NDI scale, with all scores below corresponding to surgery success and the scores above the threshold to the no success category [36].

4-2 Preprocessing

The training of a CNN requires a significant amount of data, based on the specific task. Depending on the model structure and the risk of overfitting, more training data can improve the accuracy of

the CNN model. However, there is no strict minimum of input data. According to literature, this minimum depends on the classification task, possibilities for data augmentation and the number of features that a model must differentiate [37]. In this research, the minimum was estimated to be 400 training images and 100 validation images, based on the complexity of the classification task and the complexity of the built CNN model. Nevertheless, a larger sample size is preferred.

As the available, usable data from the NECK-trial were just 70 baseline flexion X-rays, data augmentation was necessary in order to train the model properly. The following steps were performed in order to increase the number of input images:

- Rotation
  Clockwise: 0, 5, 10 and 15 degrees
  Counter-clockwise: 5, 10 and 15 degrees
- Horizontal flipping

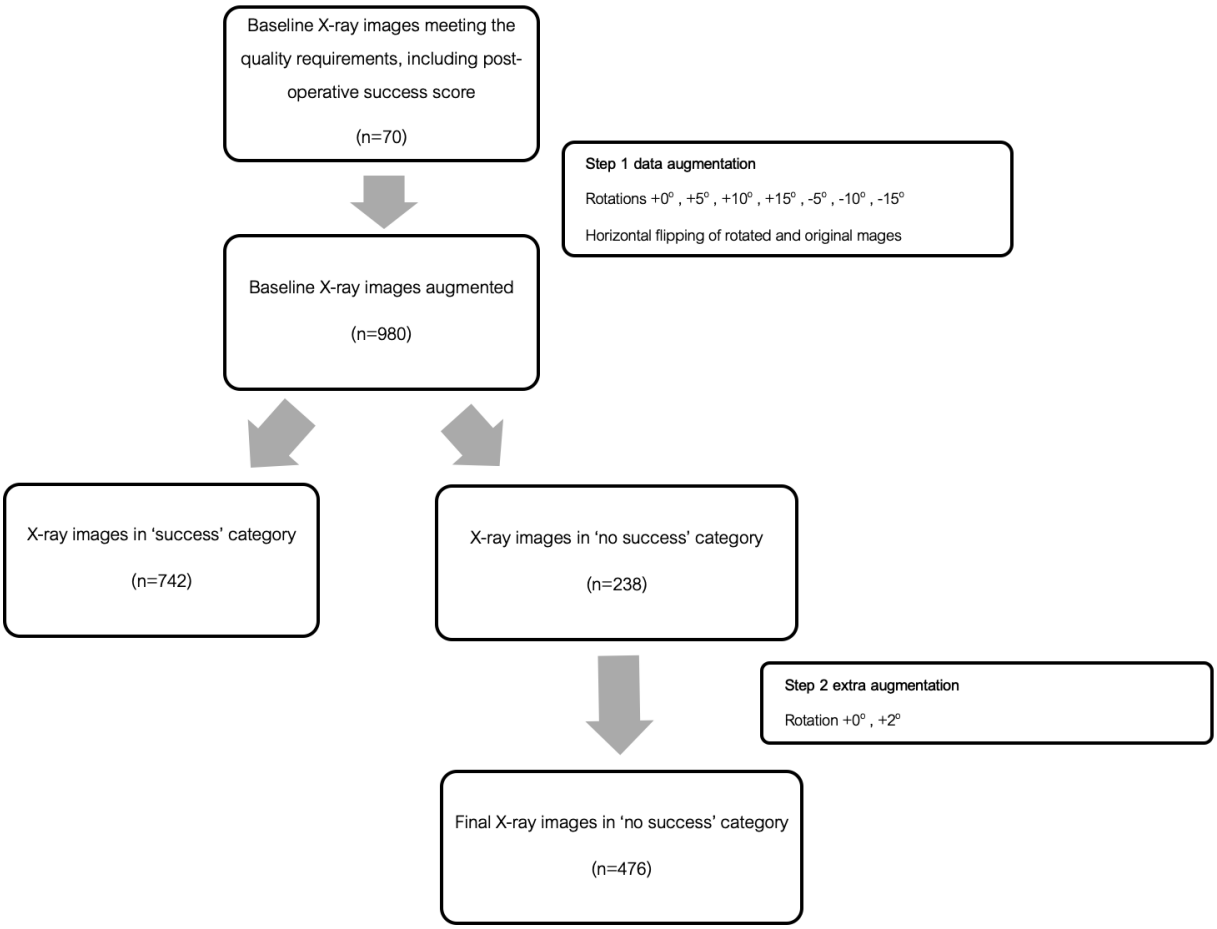


Figure 4-2: Data augmentation flowchart

During this augmentation process, which is summarized in Figure 4-2, the images were automatically resized and cropped after the rotation step. This was a necessary modification, because of the wide variability of the size and the centering of the spine within the original X-ray images. Those steps reduced the amount of unnecessary information and variations between images, as the clippings are

not relevant for the cervical spine analysis. Moreover, the CNN should not focus on the irrelevant information such as the amount of black pixels on the X-ray image. An example result of the data augmentation and preprocessing can be seen in Figure 4-3. All X-rays in the new dataset contain the most important information, the cervical vertebrae, but differ slightly from each other in orientation and centering.

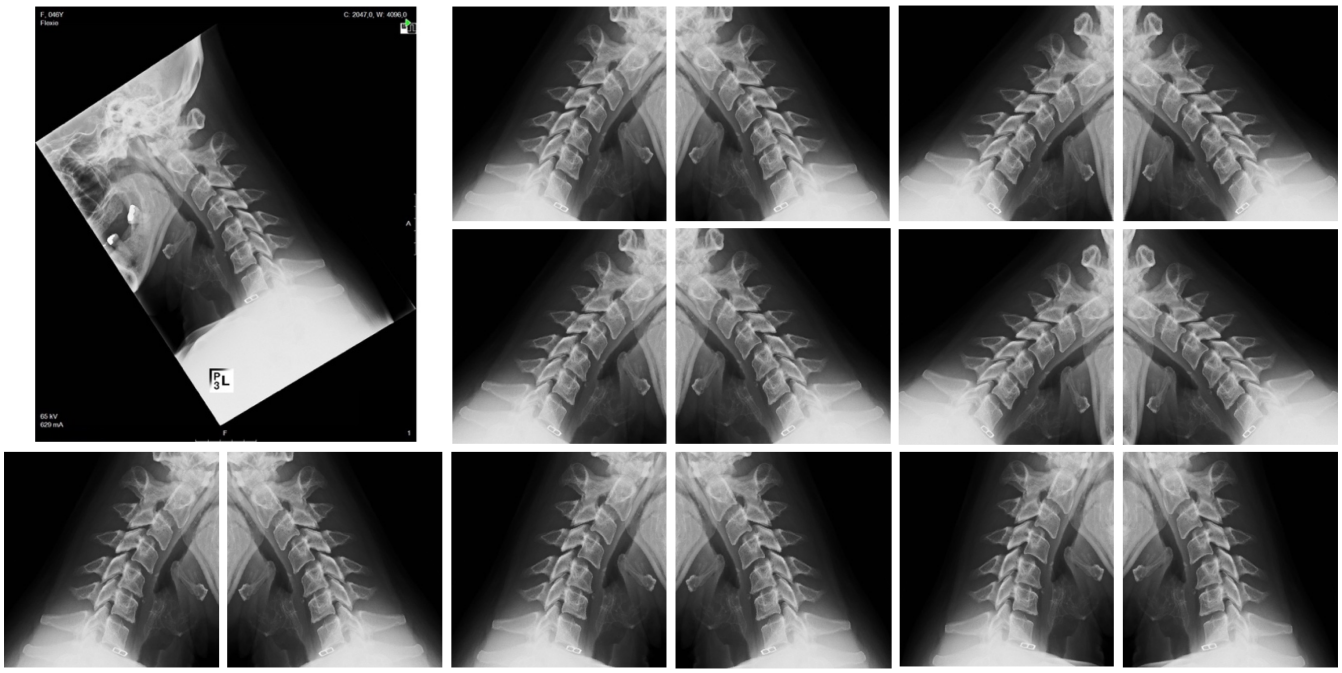


Figure 4-3: The original X-ray image can be seen in the left upper corner. The other square X-ray images are the result from the data augmentation and preprocessing steps, based on the original image.

The flipping was performed after the resizing and cropping steps, as flipping can greatly affect the positioning of the cervical vertebrae on the image, with the risk of cutting off important parts of the X-ray. By the first augmentation step, the amount of input images has been increased from 70 to 980, as the number of original images is multiplied by seven times the rotation steps, and accordingly doubled by the flipping step, which results in a factor 14 per original image. Although the procedure for taking an X-ray is standardized in every hospital, discrepancies persist between different institutions and patients. Therefore, the subtle variations according to the original image could improve the results and the flexibility of the CNN on data from the field, in our case from the hospitals.

Because the 'success' (n=53 patients) and 'no success' (n=17 patients) categories were not evenly distributed, data augmentation was necessary in order to prevent an unbalanced dataset for the CNN training as well. Therefore, the images of the 'no success' category were doubled by an extra two degrees rotation step, resulting in a total number of 1218 input X-ray images. From the flowchart (Figure 4-2) it can be seen that the final number of X-ray images were 742 and 476 in the 'success' and 'no success' category, respectively. Thereby, the final distribution became 60% of the images in the success category and 40% of the images in the no success category, which is acceptable for the CNN training. The final dataset was distributed in 80% training data and 20% validation data, which is widely recommended for small datasets in literature [38]. Different divisions of the data over the training and validation categories were used, based on random assignment of patient numbers per category. All the configurations had a 80:20 distribution and a random order in the training data flow to prevent the order of the images from influencing the training process.

4-3 CNN Model Building

First, existing ANN architectures used for image classification were analysed regarding the possibilities on the preprocessed NECK-trail dataset. Well-known architectures are AlexNet, ResNet, VGGNet and Inception (GoogLeNet) [39]. Because of the limited amount of input data, major overfitting occurred, as the existing algorithms are designed for very extensive image datasets and multiclass classification tasks.

Second, it was decided to use the general principle of the existing ANN image classifiers, but to adapt it to the NECK-trial dataset, as our dataset was significantly different from the open source Imagenet dataset on which the existing computer vision architectures are based. As explained in Chapter 3, a convolutional neural network (CNN) would be most suitable for the binary X-ray image classification task. Therefore, it was decided to build a novel CNN model, optimized for the binary classification of the X-ray images from the NECK-trial dataset. The aim was to implement the largest amount of trainable neurons, without overtraining the model. In order to prevent overfitting of the model, a decaying learning rate is integrated by a factor 0.75 per epoch.

As every train-test cycle gave slightly different outputs, as a result of the varied image data flow, ten train-test cycles were performed. After every training cycle, the output of the model was cleared and the kernel was restarted, in order to prevent influence from earlier training cycles. In addition, different configurations have been made between the train and test categories. During the whole CNN training, the image flow was shuffled in the training data, to prevent that the ordering of the images has any influence on the final result.

Model Optimization

The number of layers, the type of layers and additions, such as dropout, are adjusted and applied according to the desired goal of the model, minimizing the risk of overfitting and optimizing the distinctive power of the model. A basic setup was created, which was used as a general proof of concept. For the programming part of this research Python 3.7.9 was used in combination with Jupyter Notebook.

The principle proof of concept was performed by a 5-layer CNN. A Rectified Linear Unit (ReLU) was used as the activation function in the convolutional layers and the first dense layer. The sigmoid activation function was used in the final dense output layer. In addition, binary cross-entropy was used to calculate the loss function. Those settings are widely recommended for image classification tasks [40]. To make sure this would be the best basic setting for the model, the loss functions 'sparse categorical cross-entropy', 'squared loss' and 'hinge-squared loss' were tested as well. The hyperbolic tangent activation function was tested as well on the convolutional layers and the output layer. At the end of this phase, it was decided to keep the initial settings for the CNN, regarding the activation and loss functions per layer.

As expected for classification of a specific task with limited input data, overfitting occurred in the first model. Several possible causes were identified:

- **Batch normalization** was not applied in the first version of the CNN model. The addition of batch normalization improved the performance of the CNN model significantly.
- **Complexity of the model:** The most common cause of overfitting is a too high complexity of the model. This unwanted side-effect of increased complexity can be remedied by adding dropout layers and increasing the amount of input data.
- **Training configuration:** By reducing the number of epochs and batch size in the training phase of the CNN model, overfitting is avoided, as those factors affect the termination and extension of the training.

Different CNN model configurations were evaluated in order to optimize the CNN model and training phase, without overfitting on the available data. This was determined by structured trials in which one factor was varied to determine the effect of each factor on the model and thereby the results of the classification. Hereby, the CNN algorithm could be optimized for the available preprocessed data. The following variables were tested:

- **Number of layers:** 2, 3, 4, 5, and 6 convolutional layers were implemented and tested.
- **Training parameters:** In all of the above mentioned layer configurations, the steps and epochs were varied in order to prevent overfitting. The steps describe the number of batch iterations per training epoch. Therefore, the batch size and the number of steps are inversely proportional. Steps were varied in the range of 10 to 50, with a 10 step interval. The number of epochs was varied from 10 to 60, also with a 10 epoch interval. The results were compared based on the Area Under Curve (AUC) on the Receiver Operating Characteristic (ROC) plot, which determines the distinctive power of the model for the classification task. An AUC of 1.0 would be a perfect classifier and the dotted 0.5 line would be comparable with flipping a coin. The parameters were all compared to each other in a crosstab.
- **Dropout:** Both the dropout value and the kind of dropout method were varied: Dropout applied to all the neurons, dropout applied per convolutional layer, and 'spatial dropout' was applied, which excludes entire feature maps from the pooling part of the CNN. No dropout (value 0) was compared to 0.2, 0.5, and 0.7. In the best overall performing model configuration, all dropout values were analysed per interval (0.1 to 0.9). In addition, general dropout per layer was compared to spatial dropout in this setting as well.

Following the results of the diverse trials, the optimized configuration was determined for the NECK-trial dataset. More diverse training configurations were applied and evaluated in the optimal model setting from the trial results. Hereby, the boundary settings of the CNN model could be analyzed as well.

Besides the training and testing on preprocessed data, the model was trained with the original X-ray baseline images. The original data were augmented with the same augmentation steps applied on the preprocessed images. As expected, the preprocessing was necessary for the training stage, and the model was not able to extract the specific features from the original X-ray images. However, the validation of the model could be executed on the original X-ray images, after the training on the preprocessed data, which could be beneficial for implementation in the medical field. In addition, heatmaps were created for those original X-ray images as well, to analyze if the same features are highlighted.

4-4 Output validation

The outputs of the final CNN model were evaluated based on accuracy and loss per epoch, ROC plot, and AUC. The ROC curve integrates all classification thresholds between the categories by plotting the true positive rate against the false positive rate, allowing visualization of the distinctive power of the CNN model. When a lower classification threshold is applied, there should be more positive classifications, leading to an increase in both false and true positive predicted patient outcomes. A higher threshold would have the opposite effect. This threshold change ultimately creates the ROC curve. A confusion matrix was extracted as well, which gives a visual performance summary of the model. One diagonal represents the correct predicted classes, the other diagonal of the confusion matrix represents the number of incorrect predictions, since the number of true and predicted classes are plotted against each other. Together with the confusion matrix, the following associated parameters are determined:

- True positive rate
- True negative rate
- False positive rate
- False negative rate
- Positive predictive value
- Negative predictive value
- Sensitivity
- Specificity
- F1 score
- Matthews correlation coefficient (MCC)
- Overall accuracy

Moreover, the feature maps and kernel configurations were made visible and were evaluated per training session. This provided more insight into the working mechanism of the CNN and the path towards the eventual outcome class. The weights of the trained kernels were visualized as well. The visualization is performed using colours. The initial weights of the model, before the CNN training process, were equal and determined by the default settings of the convolutional layer. The kernel initializer of the default Conv2D layer is based on the Glorot uniform method, with the bias initialized as all zeros [41, 42]. The Glorot initializer uses random samples from a uniform distribution between $-limit$ and $limit$, where $limit$ is determined by [42]:

$$limit = \sqrt{\frac{6}{(n_{in} + n_{out})}} \tag{4-1}$$

In equation 4-1, the value of n_{in} is the number of input neurons in the initial weight tensor and n_{out} denotes the number of output neurons of the initial weight tensor.

4-4-1 Heatmaps (Grad-Cam)

In order to increase clinical relevance of the output, so called heatmaps were created by Gradient-weighted Class Activation Mapping (Grad-CAM), developed by Selvaraju et al. (2019) [43]. Although the feature maps could give an indication of the focus locations of the CNN model and the important features, it remains unclear which pixels were most important during the classification. The heatmaps provide a visual explanation of a complex CNN model. This leads to a more transparent model, decreases the 'black-box' feeling for the implementation in medical settings. Moreover, unexpected wrong classifications can be explained by the heatmaps and dataset bias could be identified by the visualizations [43]. As a result, the developers, and in this research the medical specialists as well, know where the model " looked " and therefore where possible important radiological characteristics are located [43]. This allows the medical specialists to analyze how and why the model made the decision, rather than relying on numerical outputs only.

In default programming, the accuracy of the Grad-CAM image is based on the dimensions of the last convolutional layer. In this research the dimension of the 64 feature maps in the fourth convolutional layer is 21x21 pixels. However, we applied it to other convolutional layers, in order to change the number of pixels in the 'class activation heatmap', and thereby change the accuracy of the eventual heatmap output. Grad-CAM was tested on all of the layers and the results, which can be seen in Figure 4-4, were analysed. Thereafter, it was decided to use the Grad-CAM heatmap based on the second convolutional layer of the CNN model, with a dimension of 97x97 pixels, as this configuration showed the best precision.

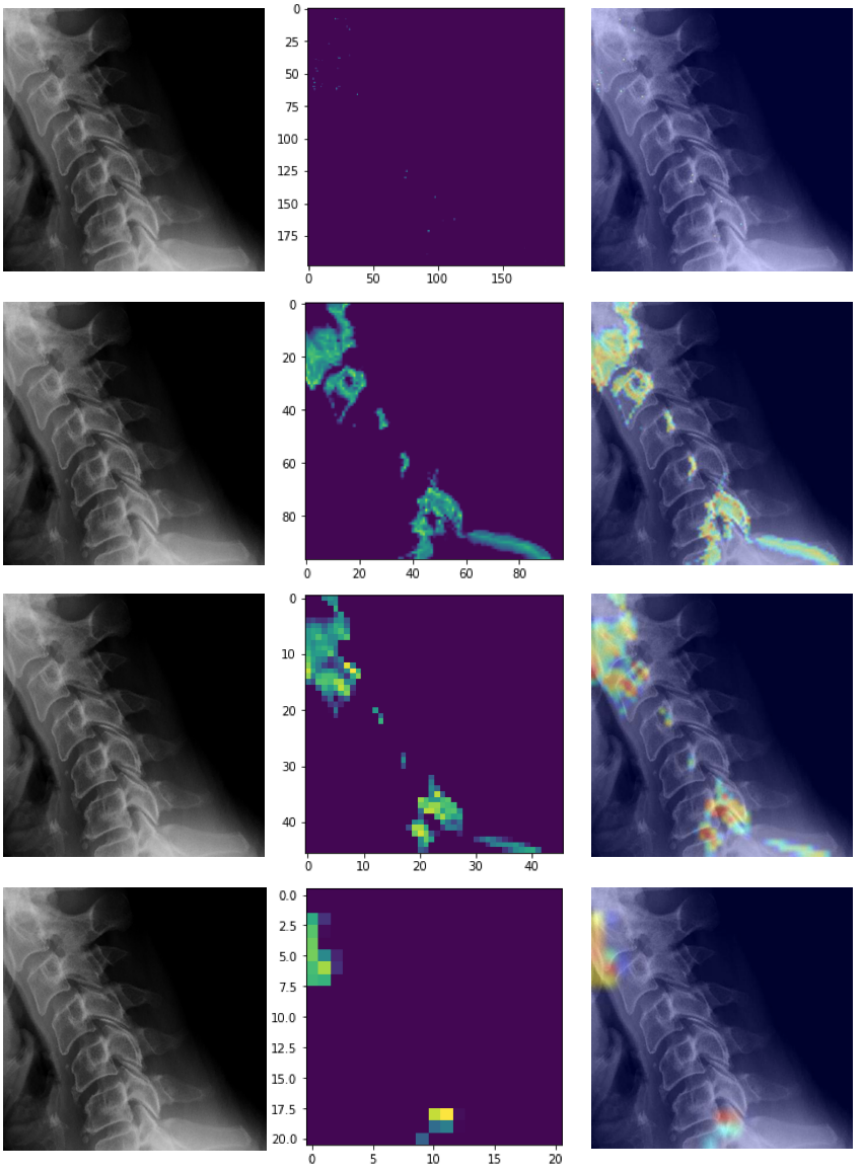


Figure 4-4: Comparison of different heatmap configurations by Grad-CAM. The first row shows the configuration based on the first convolutional layer, the second and third rows are based on the second and third convolutional layer respectively, and the fourth row shows the configuration based on the fourth (last) convolutional layer. In the left column, the input X-ray image can be seen, the generated heatmap and the superimposed heatmap on the X-ray image can be seen in the middle and right column, respectively.

Chapter 5

Results

5-1 Final CNN Model

5-1-1 Optimization Process

Before the optimal configuration was achieved, different parameters were varied and tested. Overfitting was the most important problem, which resulted in insufficient classification results. Examples of these intermediate results, with their parameter set, can be seen in Appendix A.

5-1-2 Final Model Configurations

The final configuration of the convolutional neural network (CNN) consists of four convolutional layers with ReLU activation function, and maximal pooling function to reduce the size of the feature maps. This maximal pooling function for two-dimensional feature maps has a typical window of 2x2 pixels. The kernel of the convolutional layer is a consistent 3x3 pixel size. In addition, batch normalization was applied in the first layer of the model in order to create a more stable training environment. The number of kernels per layer were 16, 32, 64, 64, from layer one to four, respectively. The summary of the model can be seen in Appendix B. A visualization of the final CNN can be seen in Figure 5-1.

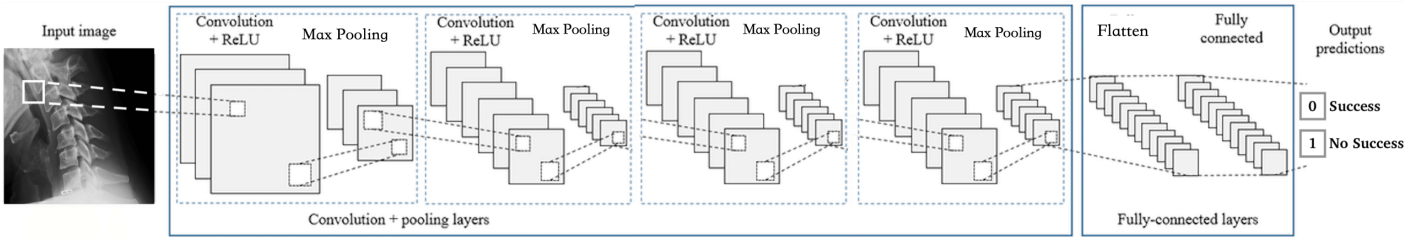


Figure 5-1: Illustration of the final CNN. On the left hand side, the sagittal X-ray image can be seen. The four convolutional layers with ReLU can be seen in the blue box, followed by the flatten and the dense (fully connected) layer, which are one dimensional vectors. The final prediction can be seen on the right hand side as binary output.

The optimal training parameters were 60 epochs, each containing 30 training steps. A decaying learning rate was integrated by a factor 0.75 per epoch. In addition, a dropout rate of 0.5 was implemented in every convolutional layer to prevent overfitting. For the specified configuration, a total training-test cycle lasts 4 minutes and 27 seconds, using one GPU.

5-2 Output CNN Model

The best result and the average result of the ten training cycles can be seen in Table 5-1. The results and outputs of all the train-test cycles can be seen in Appendix C and D, respectively. An example plot can be seen in Figure 5-2. In this plot, the confusion matrix in the left upper corner shows the amount of true positive predicted and true negative predicted outcomes on one diagonal, and the amount of false predicted outcomes on the other diagonal. The percentages of the true or false positive (TPR and FPR) and the true or false negative rate (TNR and FNR) can be seen in Table 5-1. As the aim of the research is to prevent unbeneficial surgeries, the amount of false positive predictions should be minimized. The FPR is 19% on average and 15% during the best performing run, which is a promising result. The sensitivity of the model denotes the ability to correctly predict the successful surgical procedures, which is the same as the true positive rate. The specificity describes the ability to correctly predict the unsuccessful surgical procedures, which corresponds to the true negative rate. Therefore, the mean specificity of 81%, compared to the mean sensitivity of 74 %, shows that the ability of the model to correctly predict an unsuccessful surgery is higher compared to the prediction of successful surgeries. The F1 score is the average of the positive predictive value and the true positive rate and has a value between 0 and 1, whereby 1 is the best, highest possible value and 0 the lowest possible value.

The Matthews Correlation Coefficient (MCC) is a valuable measure to describe the performance of a binary classifier, as it takes unbalanced data into account. This is relevant for this research, as the data is augmented, but there remains a size gap between the 'success' and 'no success' categories used in the CNN. The MCC has a value between -1 and +1, and is a measure to describe the quality of prediction by a binary classifier. The value +1 shows a perfect prediction by the model, an MCC of zero indicates a random prediction, and -1 shows an overall disagreement between the prediction of the model and the true classes, based on the ground truth labels. The MCC is based on all of the four values of the confusion matrix (TPR, FPR, TNR and FNR), proportional to the batch size of the two initial categories of the input dataset [44]. This prevents unrealistic high values, which could occur in for example the F1-score, based on an unbalanced datasets [44].

Furthermore, the receiver operating characteristic (ROC) curve shows an area under curve (AUC) of 0.91, whereby 1.0 would be a perfect classifier and the dotted 0.5 line would be comparable with flipping a coin.

In contrast to the ROC and AUC values, the model accuracy is determined based on a fixed threshold: 0.5. Values under 0.5 are denoted as 0 (success category) and values above 0.5 as 1 (no success category). The accuracy and the loss can be seen on the right hand side in Figure 5-2, plotted per epoch. As the learning rate decreases, the model accuracy stabilizes. During the model analysis, the main focus is on the test data results (orange), as these determine the classification of the never-before-seen X-rays images, which would be the accuracy in the clinical setting as well. The decreasing loss shows no overfitting in the final CNN model, because overfitting would increase the loss of the test data while the loss of the training data decreases.

Table 5-1: Best (left), average (center) result of the CNN, together with the results when using the original X-ray images during the validation stage (right), per output value

Output value	Best Run	Average	Original X-ray Validation
True positive rate	0,81	0,74	0,70
True negative rate	0,85	0,81	0,72
False positive rate	0,15	0,19	0,28
False negative rate	0,19	0,26	0,30
Positive predictive value	0,84	0,79	0,72
Negative predictive value	0,81	0,76	0,71
Sensitivity	0,81	0,74	0,70
Specificity	0,85	0,81	0,72
F1 score	0,82	0,76	0,71
Matthews correlation coefficient MCC	0,55	0,44	0,31
Overall accuracy based on CM	0,83	0,77	0,71
ROC AUC	0,91	0,86	0,67

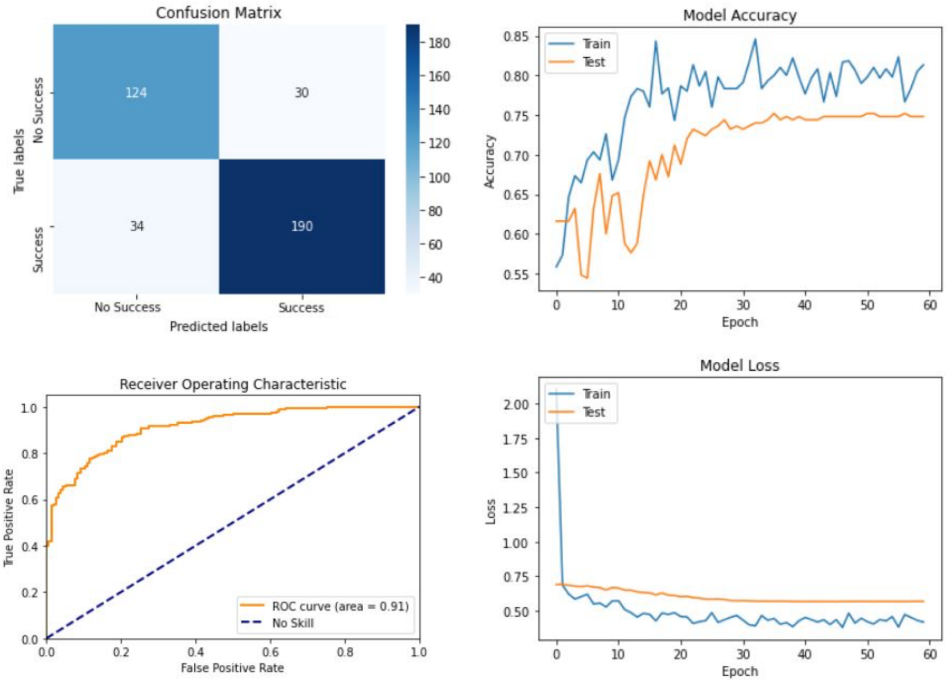


Figure 5-2: Outcome metrics of training cycle number 1: *Top left:* Confusion Matrix, *Bottom left:* ROC curve with AUC value, *Top right:* Accuracy of the CNN training and validation, per epoch, *Bottom right:* Loss of the CNN training and validation, per epoch

5-2-1 Feature Maps

To gain more insight into the working mechanism of the CNN model and to gain some understanding what the model is "looking at", feature maps are created. For every input X-ray image, a feature map is created per training and test cycle. An example of a feature map can be seen in Figure 5-3. It can be seen that each kernel creates a different feature image and focuses on different levels within one training cycle, e.g. texture, contour features and contrast boundaries. Some example feature map images, which are extracted from the total feature map in Figure 5-3 to give a clear overview of different possible extracted features, can be seen in Figure 5-4.

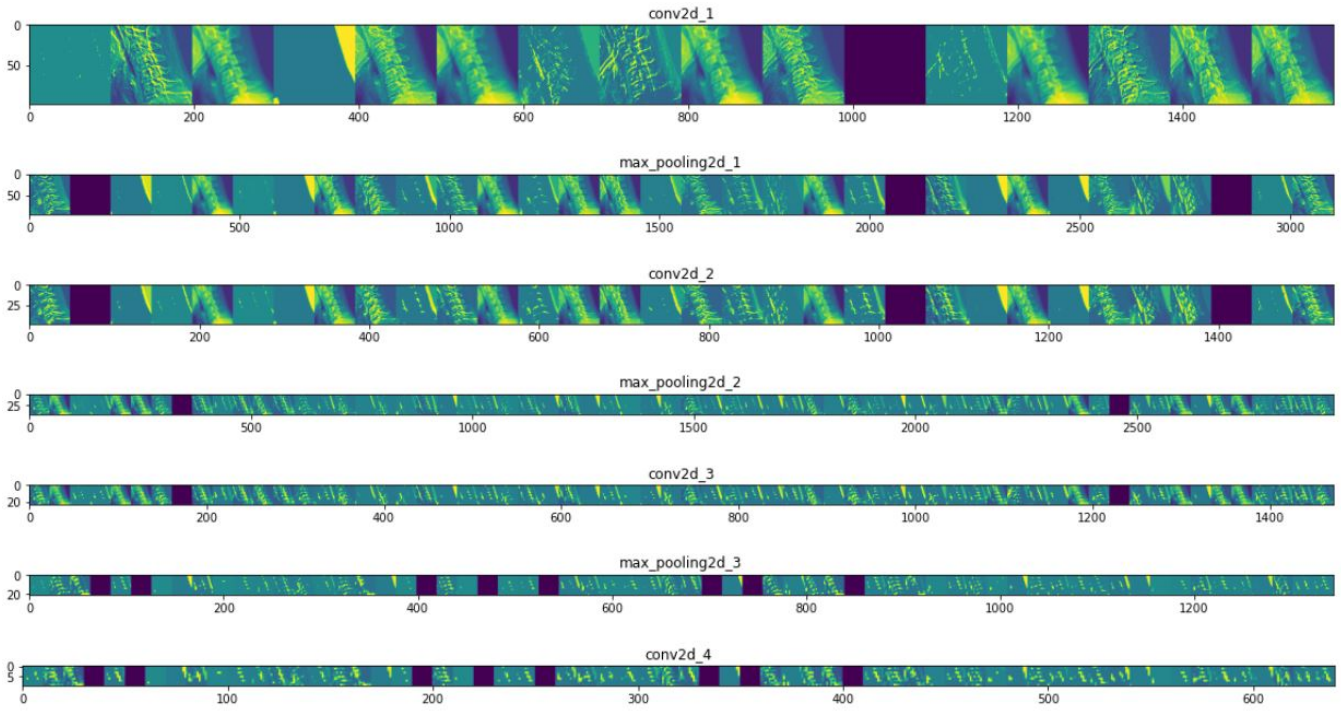


Figure 5-3: Example feature map CNN model. From top to bottom, the feature images, produced by the different kernel weight calculations, are shown per layer of the CNN model. One feature map is based on one single input image, and shows the highlighted features, visualized by a separate feature image, per kernel.

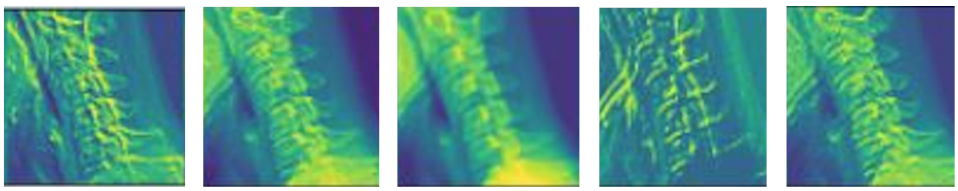


Figure 5-4: Featured images with the focus on different specific characteristics, from left to right: texture, bone density, zygapophysial joints, cortical bone edges, bone density with contrast influence

5-2-2 Grad-CAM Heatmaps

In the heatmap plots (Figure 5-5), the pixels are highlighted which were most important in the classification task, after which this pixel map is projected on the original X-ray image. The colourscale of the overprojected pixels can be seen on the right side of the plot in Figure 5-5. It can be seen that the same features and pixels are highlighted in the slightly rotated (lower) and not rotated image (upper) (Figure 5-5). The heatmaps of different classes, patients and cervical orientations can be seen in Appendix E. Those heatmaps could support the medical specialists in determining what to focus on. However, those heatmaps still only give an indication of important anatomical locations and are not suitable for diagnostics on their own. Nevertheless, it can be seen that the CNN model is focusing on relevant parts of the X-ray image, instead of the black areas, for example, which is promising for the reliability of the CNN model. Remarkable is the focus on the zygapophysial joints, the posterior part of the vertebrae, and the C1-C2 segment. Moreover, the edges of the shoulder bones seems important as well, which could be questionable, as this is not the cervical target area. It can be seen

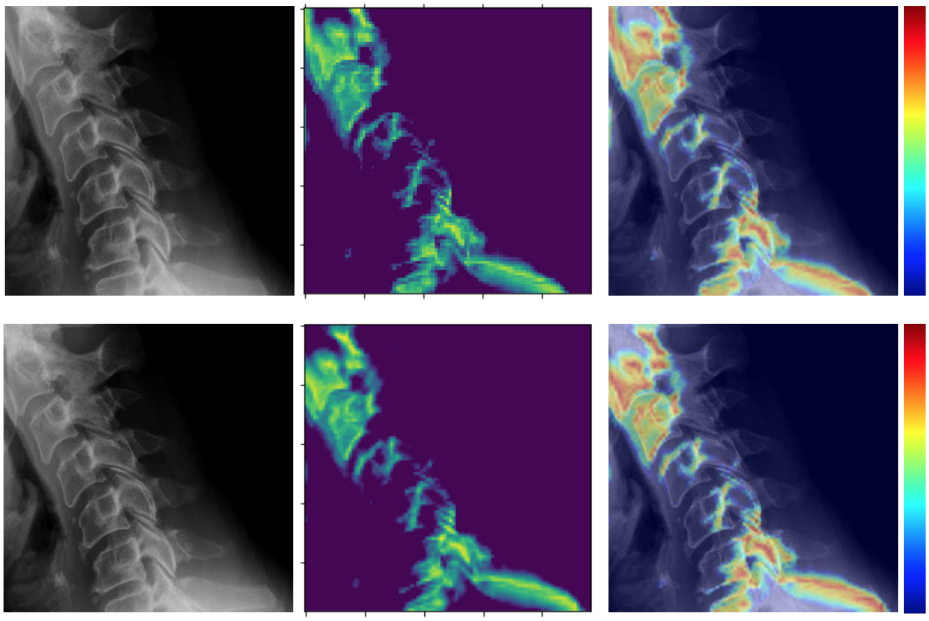


Figure 5-5: Heatmap of two slightly different X-ray images. The upper row contains the original configuration and the lower row shows the image with a five degree rotation counterclockwise. Left column: original X-ray image, middle column: heatmap image, right column: heatmap superimposed on original X-ray image

that the results of the CNN model based on validation on the original X-ray baseline images are not as good as the results of the validation on the preprocessed images. This can be visualized using the Grad-CAM heatmaps as well. Figure 5-6 shows similar features highlighted on the heatmap on both the preprocessed and the original X-ray images during the validation process. Some disturbances for classification could be seen on the shoulder area and the cortical bone of the jaw and the skull. However, the major cervical characteristics are the same. In Figure 5-7, again both the preprocessed and the original X-ray images during the validation process are shown, but from a different X-ray image. Unlike the similar features shown in Figure 5-6, the CNN model is significantly less accurate regarding the features on the original X-ray image on the lower row of Figure 5-7. The cervical vertebrae are properly extracted, but it does not extract the certain features which can be seen in the upper row of Figure 5-7. Nevertheless, it is promising that the CNN model is able to detect the cervical vertebrae on the original images with a lot more noise and disturbances, compared to the preprocessed X-ray images.

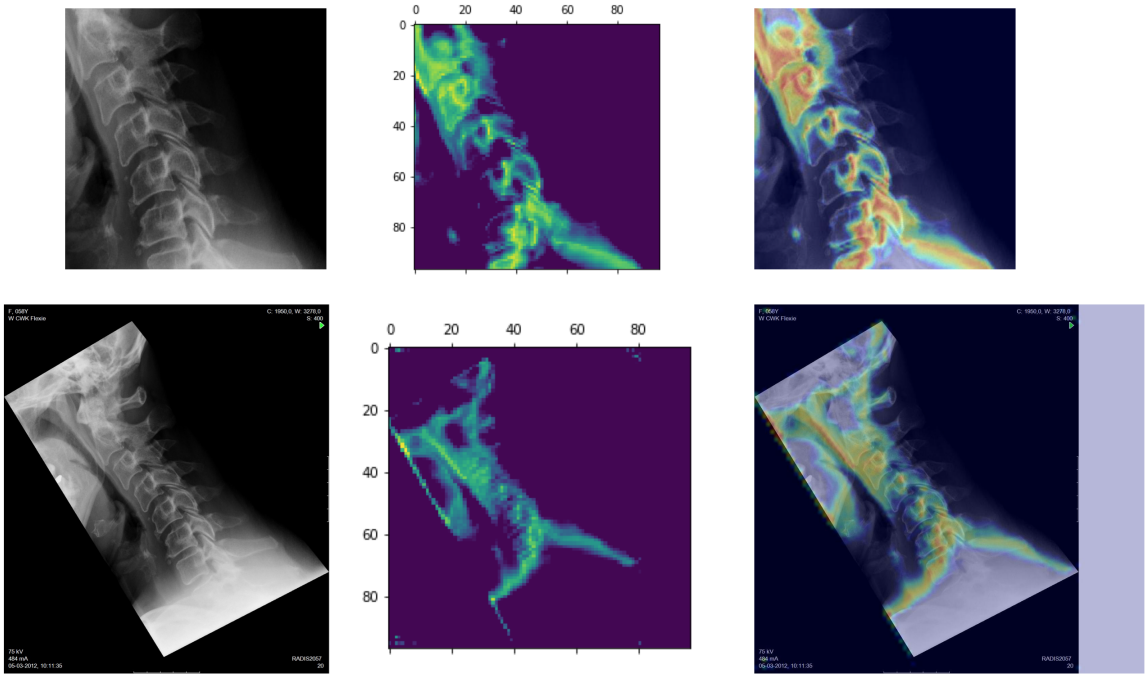


Figure 5-6: The upper row contains the heatmap of the preprocessed image. The lower row shows the heatmap of the validation process on the original baseline X-ray image, without preprocessing. This example shows a good extraction of specific cervical features. Left column: original X-ray image, middle column: heatmap image, right column: heatmap superimposed on original X-ray image

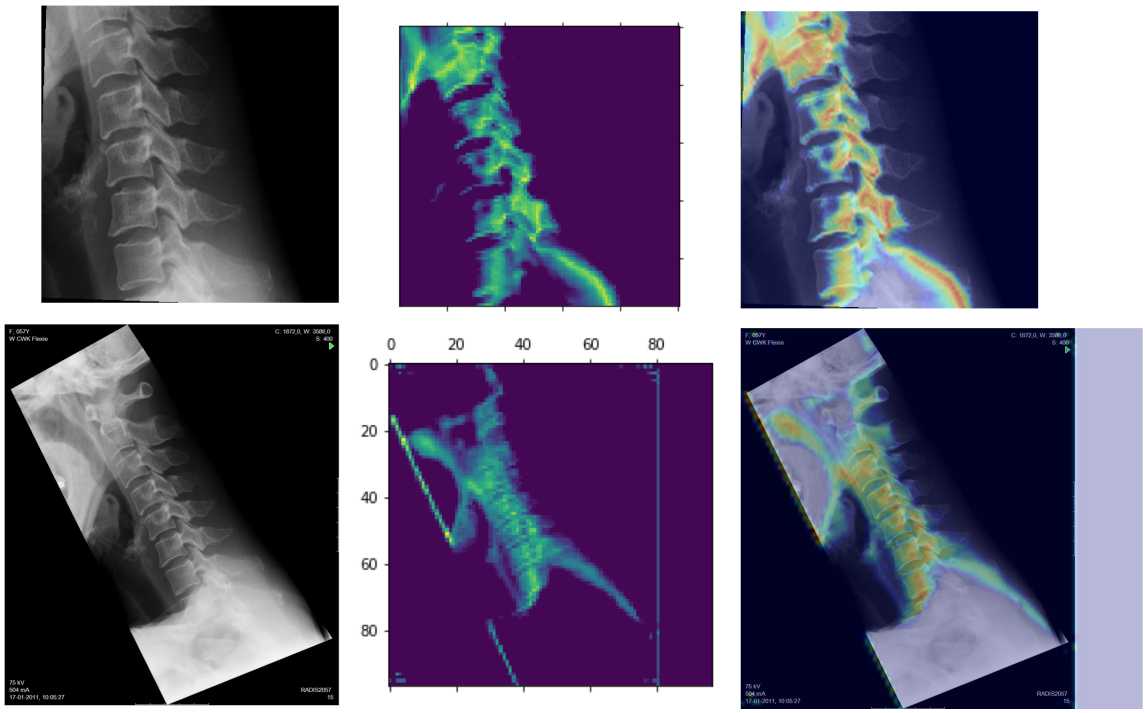


Figure 5-7: The upper row contains the heatmap of the preprocessed image. The lower row shows the heatmap of the validation process on the original baseline X-ray image, without preprocessing. This example shows that the cervical vertebrae are highlighted, but the feature extraction is less specific, compared to the example in Figure 5-6. Left column: original X-ray image, middle column: heatmap image, right column: heatmap superimposed on original X-ray image

Chapter 6

Discussion

In this research, a new CNN model is developed to predict whether a patient would benefit from invasive surgical spine surgery. While the results seem promising, there are some aspects to consider.

Model Analysis

First, the original dataset of useful X-rays consisted of only 70 different patient cases. Although data augmentation is applied, there are only 70 distinctive cases on which the model has been trained. Because each patient situation is unique, it can be assumed that there are many more clinical situations that are not included in the model, so it cannot be said with certainty that the same results could be achieved in a clinical setting. Moreover, the validation results based on the original X-rays show inferior results, as the training process is performed on the preprocessed and augmented data. This demonstrates the essential preprocessing step before the X-ray could be analyzed by the CNN model, which could increase the risk of errors when this step has to be integrated into the medical field. Moreover, the fluctuation in output per train-test cycle shows the influence of different train-test distributions, which could be explained by the limited amount of substantially different X-ray images and thereby patient cases.

Second, the original X-ray images are preprocessed in order to create a more consistent input and prevent noise in the training procedure of the CNN. The original input images had a wide variety of configurations, with some outliers having the cervical spine on the side of the image. Although the heatmaps showed no significant influence due to the location on the image (Appendix D), there may still be an influence on the classification result. In addition, the dataset was expanded to create a necessary larger dataset for training and testing the CNN model. The augmentation was performed by steps of 5 and 2 degrees of (counter-) clockwise rotation, to create a significant change, but to maintain the original orientation of the spine. For example, turning the spine upside down would create an unrealistic situation. Since the CNN model only analyzes the pixel values in comparison to the surrounding pixels, a small rotation can provide a completely new 'view' for the computer. Nevertheless, some changes could have been too small, leaving the training dataset of an insufficient size to distinguish different features.

Third, the test train distribution is performed manually, based on patient number, to avoid influence of the same image in a rotated view. Randomization is performed by an automatic random shuffle of patient numbers in Microsoft Excel. While it may seem random, it should not be completely random as the Excel program could have a preprogrammed method for creating the new order of data.

Finally, the training of the CNN model did not reach the optimum. Training accuracy did not come to the expected 100%, due to the declining learning rate. Therefore, the trade-off has been made between overfitting or a decrease in training accuracy. Since the X-ray image of a 'real' patient would be analyzed after the training cycle, in the validation phase, it is decided to focus on the overall model performance and the accuracy of the test cycle, as this shows performance on never-before-seen data. Nevertheless, the lower training accuracy shows room for improvement in the training phase. Despite the room for improvement, the CNN shows better results compared to the predictive power of the medical specialists, which could be estimated by an AUC of 0.5.

Implementation in the clinical field

This research is performed with the purpose of computer aided decision-making. There is still a medical specialist needed to make the final decision, that the CNN model could influence. In healthcare, this is not only based on the feelings of the patient, but also on ethical aspects and regulations. After all, the medical specialist remains responsible for the final decisions and treatment planning for the individual patient. Although machine learning, and especially convolutional neural networks, are without a doubt a very important research and development area for medical purposes, there are still major hurdles to overcome before it could be implemented in patient care.

First of all, the medical specialists for whom this CNN model has been developed, should be willing to use the CNN model. There must be sufficient training in the medical centers on how the model works, what the purpose of the model is and what the limitations of the model are in their specific clinical practice and decision-making process.

Second, privacy and data security are major problems in this data driven approach in the medical world. One aspect in this discussion is the support of medical specialists and the opinion of patients [45]. Before a data driven approach is accepted and implemented, the medical specialists should have trust in the processing of the data and the safety of the patient [45]. This can only be achieved by proven safe infrastructure for storage and data processing. After all, it is not only the secure processing of the data, but also the collection, transmission and storage of the enormous amount of data associated with a patient. Subsequently, the involvement of the patient requires informed consent [12]. Anonymization of medical data is used to improve data security. However, even anonymized data is subjected to strict regulations by the General Data Protection Regulation in the European Union [12].

Third, ethical aspects counteract the implementation of ML techniques. The accountability of the diagnosis and medical outcome or the risk of false positive or negative predictions are for example very relevant [12]. Is it more important to avoid unnecessary operations or to perform high-risk operations, with the chance that the patient's well-being will improve? These questions require a change of perspective on the new technologies flowing into clinical practice. Especially for new implementations with direct clinical consequences, which is the case with CAD or ML guided predictions [12]. Moreover, the acceptance of the new techniques is correlated with the understanding of those new technologies. This aspect is difficult to overcome, as neither the medical specialist nor the patient has the knowledge to understand what the model or algorithm actually does, and they are therefore unable to properly assess the risk of inaccuracy in the results of the ML model. Although the user interface could be programmed to explain this as clearly as possible, for example by the Grad-CAM heatmaps, the understanding remains difficult. This limitation in understanding of the method induces the issue of accountability of made decisions. Even if a CNN model is only used in order to support the medical specialist, and the clinician makes the final decision, there remain question marks in case of mistakes. Therefore, the market approval of novel ML technologies is unfavourable, whereby more thorough testing is requested compared to other innovations, which increases the production costs and implementation time-scope [12].

Chapter 7

Conclusion

This research started with the research question: "How could a convolutional neural network algorithm be implemented in order to predict the success of an invasive cervical spine surgery, based on baseline X-ray images?", with the technical aim to build a binary classifier in order to predict 'surgery success' or 'no surgery success', based on sagittal baseline X-ray images of the cervical spine.

It can be concluded that this classification task could be performed by a convolutional neural network (CNN), adapted to the specific classification task. The best performing model configuration consisted of four convolutional layers with 16, 32, 64, 64 kernels from layer one to four, respectively. The highest achieved distinctive power (AUC) of the model was 91% and the lowest lowest achieved false positive rate of the model was 15%. This shows a major advantage compared to the current clinical situation at the department of neurosurgery in the Leiden University Medical Center, where about 25% of invasive spinal surgical operations, involving cervical degeneration, of no benefit to the patient.

Despite the promising results, the application of convolutional neural networks would take time. Not only because of the data-related infrastructure in healthcare, but also because of the ethical aspects, data processing and data security issues. Adjustments are needed in mindset and in the general data-driven framework of the medical field in order for these innovations to work properly.

Nevertheless, the use of convolutional neural network algorithms to support the medical specialist in decision-making and personal treatment planning seems very promising for the healthcare of the future.

Chapter 8

Recommendations

It would be interesting to train the model on a larger patient dataset. This would allow for a slightly higher complexity of the model and thereby the training accuracy could be improved. Moreover, it would be more applicable, as the training data would cover more relevant anatomical situations of the cervical spine.

The certainty of the outcome has not been determined in the developed CNN model. The model only provides the classification. It would be clinically relevant to determine the probability of the model or certainty that an outcome is true, as this is obviously an important factor in the decision-making process.

In future research, the CNN model could be compared to other neural networks. Due to the novelty of this research niche, there are currently no predictive models based on sagittal X-rays. It would be interesting to ascertain the preferability of a CNN in comparison to other artificial neural networks based on research conducted with the same goal.

In addition, the CNN model could also be trained for other clinical predictions or diagnostics that can be determined on an X-ray, for example, the existence of pneumonia or residual damage from COVID-19. Necessary adjustment would be needed, but it could be used as a foundation for other clinical decision makers.

According to the clinical output, I would recommend investigating the degeneration process of the facet joints in the cervical vertebrae. Because previous research mainly focused on the intervertebral height and the shape of the vertebral bodies, the heat maps of this research show a significant influence of the facet joints on the classification process. This may indicate new diagnostic features.

The last suggestion I want to make is the relevance of a long-term prospective clinical trial. Using the CNN model in the decision-making process of invasive spinal surgery, the effect could be analyzed over a long term. The absolute clinical relevance could be determined based on the reduced number of surgeries performed that do not appear to be of benefit to the patient in the long term.

Bibliography

[1] Cohen, S. P., "Epidemiology, diagnosis, and treatment of neck pain," *Mayo Clinic proceedings*, vol. 90, p. 284–299, 2015.

[2] UC Davis Spine Center, "Degenerative spine conditions," *https://health.ucdavis.edu/spine/specialties/degenerative.html*, accessed: 11-09-2020.

[3] Peng, B., & DePalma, M. J., "Cervical disc degeneration and neck pain," *Journal of pain research*, vol. 11, p. 2853–2857, 2018.

[4] Xing, R., Liu, W., Li, X., Jiang, L., Yishakea, M., Dong, J. , "Characteristics of cervical sagittal parameters in healthy cervical spine adults and patients with cervical disc degeneration," *BMC Musculoskeletal Disorders*, vol. 19, p. 37, 2018.

[5] Iyer, S., Kim, H. J., "Cervical Radiculopathy," *Current Reviews in Musculoskeletal Medicine*, vol. 9, p. 272–280, 2016.

[6] Davies, B.M., Mowforth, O.D., Smith, E.K., Kotter, M.R., "Degenerative cervical myelopathy," *BMJ*, vol. 186, p. 360, 2018.

[7] Nam, K., Seo, I., Kim, D., Lee, J., Choi, B., Han, I., "Machine Learning Model to Predict Osteoporotic Spine with Hounsfield Units on Lumbar Computed Tomography," *Journal of Korean Neurosurgical Society*, vol. 62, pp. 442–449, 2019.

[8] Burneikiene, S., Nelson, E.L., Mason, A., et al., "The duration of symptoms and clinical outcomes in patients undergoing anterior cervical discectomy and fusion for degenerative disc disease and radiculopathy," *Spine Journal*, vol. 15, p. 427–32, 2015.

[9] Arts et al., "The NEtherlands Cervical Kinematics (NECK) Trial: Cost-effectiveness of anterior cervical discectomy with of without interbody fusion and arthroplasty in the treatment of cervical disc herniation; a double-blind randomised multicenter study," *BMC Musculoskeletal Disorders*, vol. 11, 2010.

[10] Urrutia, J., Zamora, T., Yurac, R., Campos, M., Palma, J., Mobarec, S., Prada, C., "An Independent Inter- and Intraobserver Agreement Evaluation of the AOSpine Subaxial Cervical Spine Injury Classification System," *Spine*, vol. 42, pp. 293–303, 2017.

[11] Jakubicek, R., Chmelik, J., Jan, J., Ourednicek, P., Lambert, L., Gavelli, G., "Learning-based vertebra localization and labeling in 3D CT data of possibly incomplete and pathological spines," *Computer methods and programs in biomedicine*, p. 183, 2020.

[12] Galbusera F, Casaroli G, Bassani T., "Artificial intelligence and machine learning in spine research," *JOR Spine*, vol. 2, p. e1044, 2019.

[13] Cramer, G.D., "Clinical Anatomy of the Spine, the cervical region," *Spinal Cord, and Ans (Third Edition)*, 2014.

[14] Fadial, T., "Lateral Cervical Spine Illustration," *https://ddxof.com/cervical-spine-injuries/c-spine-lateral/*, accessed: 30-09-2020.

[15] Nouri, A., Martin, A. R., Mikulis, D., Fehlings, M. G., "Magnetic resonance imaging assessment of degenerative cervical myelopathy: a review of structural changes and measurement techniques," *Neurosurgical focus*, vol. 40, 2016.

[16] Khan et al., "Inflammatory biomarkers of low back pain and disc degeneration: a review," *Annals of the New York Academy of Sciences*, vol. 1410, pp. 68–84, 12 2017.

[17] Spine-Health, "How Cervical Disc Degeneration Occurs," *https://www.spine-health.com/conditions/neck-pain/how-cervical-disc-degeneration-occurs*, accessed: 23-09-2020.

[18] Adams, M. A., Roughley, P. J., "What is intervertebral disc degeneration, and what causes it?," *Spine*, vol. 31, p. 2151–2161, 2006.

[19] Raj P. P., "Intervertebral disc: anatomy-physiology-pathophysiology-treatment," *Pain practice : the official journal of World Institute of Pain*, vol. 8, pp. 18–44, 2008.

[20] Jaumard, N. V., Welch, W. C., & Winkelstein, B. A., "Spinal facet joint biomechanics and mechanotransduction in normal, injury and degenerative conditions," *Journal of biomechanical engineering*, vol. 133, 2011.

[21] NBG Drafting and Design, "Visualization of Anterior Cervical Discectomy and Fusion with Synthes Plate," *https://nbgdrafting.com/project/c6-7-anterior-cervical-discectomy-and-fusion-with-synthes-plate/*, accessed: 06-01-2021.

[22] Goodfellow, I., Bengio, Y., and Courville, A., "Deep learning," *MIT Press: http://www.deeplearningbook.org*, 2016.

[23] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R., "Dropout: A simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, p. 1929–1958, 2014.

[24] LeCun, Y., Bengio, Y., and Hinton, G., "Deep learning," *Nature*, vol. 512, p. 436–444, 2015.

[25] Hirschkind, N. et al., "Convolutional Neural Network," *Brilliant Math & Science Wiki, https://brilliant.org/wiki/convolutional-neural-network*, Accessed: 7-10-2020.

[26] Azaria, A., "Deep learning and natural language processing," *Lecture Notes Ariel University: https://www.science.co.il/moshe/documents/deep-learning/CNN/*, accessed: 15-02-2021.

[27] Basavarajaiah, M., "Maxpooling vs Minpooling vs Average pooling," *https://medium.com/@bdhuma/which-pooling-method-is-better-maxpooling-vs-minpooling-vs-average-pooling*, accessed: 15-02-2021.

[28] Rana, K., "Pooling Layer, Short and Simple," *https://medium.com/ai-in-plain-english/pooling-layer-beginner-to-intermediate*, accessed: 15-02-2021.

[29] Lei, X., Pan, H., Huang, X., "A Dilated CNN Model for Image Classification," *IEEE Access*, vol. 7, pp. 124087–124095, 2019.

[30] Bhattarai, S., "What are Activation Functions in Neural Networks (NN)?," *https://saugatbhattarai.com.np/what-is-activation-functions-in-neural-network-nn/*, accessed: 09-02-2021.

[31] Godoy, D., "Understanding binary cross-entropy: a visual explanation," *https://towardsdatascience.com/understanding-binary-cross-entropy-log-loss-a-visual-explanation*, accessed: 18-02-2021.

[32] Natarajan, A., "Gradient Descent for Machine Learning," *https://medium.com/redblacktree/gradient-descent-for-machine-learning*, accessed: 08-03-2021.

[33] Solai, P., "Convolutions and Backpropagations," *https://medium.com/pavisj/convolutions-and-backpropagations*, accessed: 23-02-2021.

[34] Bishop, C.M., "Pattern Recognition and Machine Learning (Information Science and Statistics)," *ISBN-13: 978-0387310732*, p. 259, 2006.

[35] Dahl, G., Sainath, T., and Hinton, G., "Improving Deep Neural Networks for LVCSR using rectified linear units and dropout," *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013.

[36] Mjaset, C., et al., "Criteria for success after surgery for cervical radiculopathy: estimates for a substantial amount of improvement in core outcome measures," *The Spine Journal*, vol. 20, p. 14131421, 2020.

[37] Mitsa, T., "How Do You Know You Have Enough Training Data," *https://towardsdatascience.com/how-do-you-know-you-have-enough-training-data*, accessed: 9-12-2020.

[38] Brownlee, J., "Train-Test Split for Evaluating Machine Learning Algorithms," *https://machinelearningmastery.com/train-test-split-for-evaluating-machine-learning-algorithms/*, accessed: 5-1-2021.

[39] Anwar, A., "Difference between AlexNet, VGGNet, ResNet, and Inception," *https://towardsdatascience.com/the-w3h-of-alexnet-vggnet-resnet-and-inception-7baaaecccc96*, accessed: 9-12-2020.

[40] Dewa, C.K., "Suitable CNN Weight Initialization and Activation Function for Javanese Vowels Classification," *Procedia Computer Science*, vol. 144, pp. 124–132, 2018.

[41] Keras API reference, "Conv2D layer," *https://keras.io/api/layers/convolution_layers/convolution2d*, accessed: 09-02-2021.

[42] Glorot, X., Bengio, Y., "Understanding the difficulty of training deep feedforward neural networks," *DIRO, Universite de Montreal, Quebec, Canada*, pp. 250–256.

[43] Selvaraju, R.R., et al., "Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization," *International Journal of Computer Vision*, vol. abs/1610.02391, 2019.

[44] Chicco, D., Jurman, G., "The advantages of the Matthews Correlation Coefficient (MCC) over F1 score and accuracy in binary classification evaluation," *BMC Genomics*, vol. 21, 2020.

[45] Heilbron, B. and Koopman, E., "Elektronisch Patienten Dossier," *https://www.platform-investico.nl/artikel/nieuw-patientdossier-heeft-weinig-kans-van-slagen*, accessed: 11-09-2020.

Appendix A

Intermediate Test Result Examples

Before the optimal configuration was achieved, different parameters were varied and tested. Overfitting was the most important problem, which resulted in insufficient classification results. Examples of these intermediate results, with their parameter set, can be seen in this Appendix.

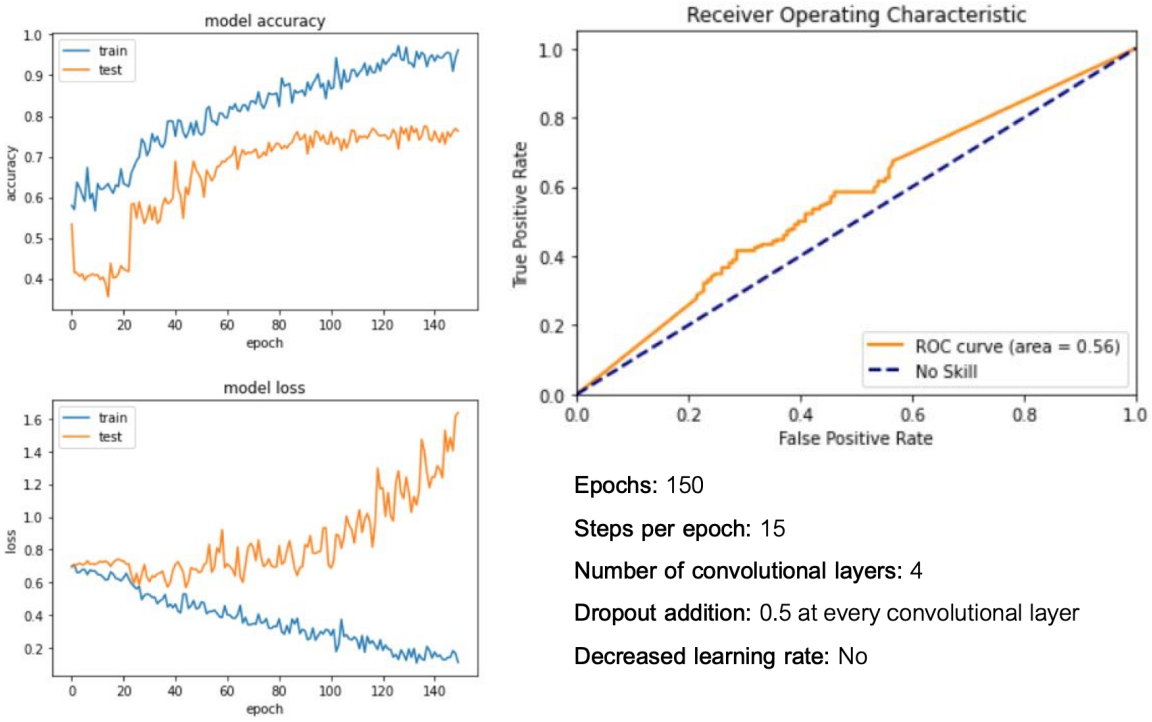
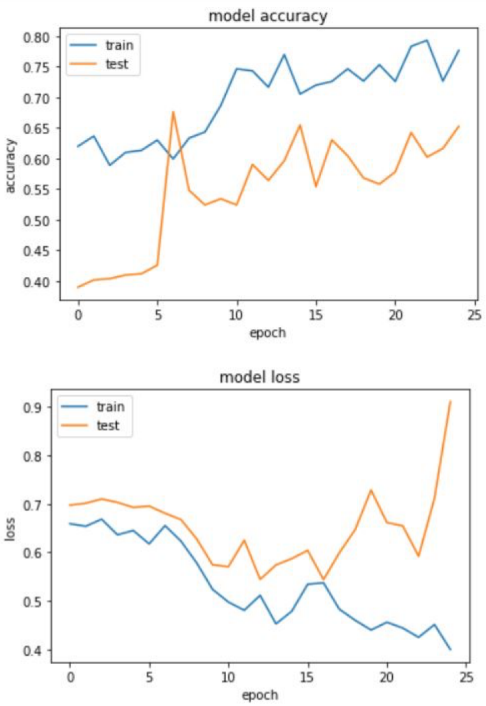
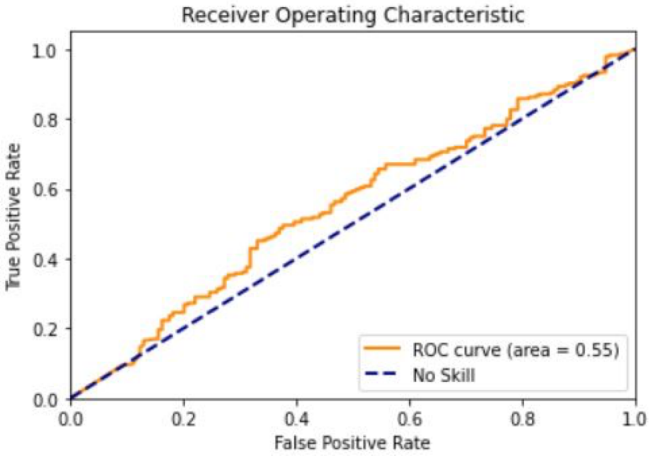


Figure A-1: In this example, major overfitting can be seen. The loss function of the test dataset strongly increases after 80 epochs, whilst the test accuracy does not improve after 80 epochs. This indicates that the amount of epoch is too much for the used dataset.





Epochs: 25

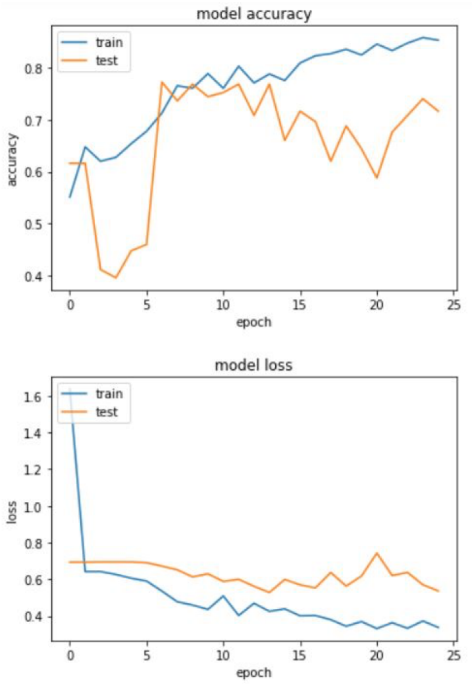
Steps per epoch: 15

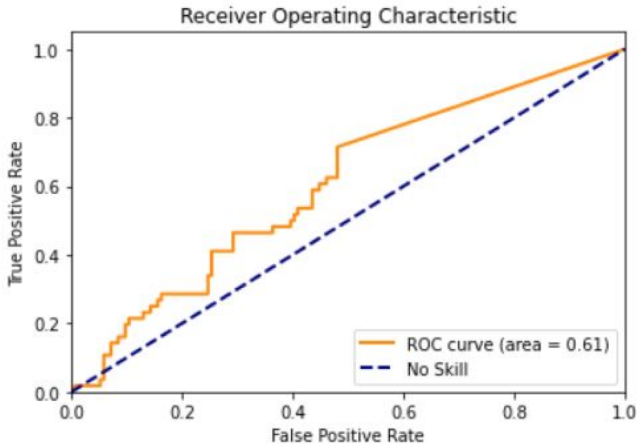
Number of convolutional layers: 5

Dropout addition: 0.5 at every convolutional layer

Decreased learning rate: No

Figure A-2: This 5-layer CNN model shows overfitting, by the increasing loss function. In addition, the train and test accuracy is not very good, resulting in an AUC 0f 0.55, which is close to a gamble.





Epochs: 25

Steps per epoch: 40

Number of convolutional layers: 4

Dropout addition: 0.5 at every convolutional layer

Decreased learning rate: No

Figure A-3: In this configuration, there is no to limited overfitting. However, the accuracy of the test data seems instable. However, the AUC is better compared to the overfitting models.

Appendix B

Summary CNN Model

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 198, 198, 16)	448
batch_normalization (BatchNo	(None, 198, 198, 16)	64
max_pooling2d (MaxPooling2D)	(None, 99, 99, 16)	0
dropout (Dropout)	(None, 99, 99, 16)	0
conv2d_1 (Conv2D)	(None, 97, 97, 32)	4640
max_pooling2d_1 (MaxPooling2	(None, 48, 48, 32)	0
dropout_1 (Dropout)	(None, 48, 48, 32)	0
conv2d_2 (Conv2D)	(None, 46, 46, 64)	18496
max_pooling2d_2 (MaxPooling2	(None, 23, 23, 64)	0
dropout_2 (Dropout)	(None, 23, 23, 64)	0
conv2d_3 (Conv2D)	(None, 21, 21, 64)	36928
max_pooling2d_3 (MaxPooling2	(None, 10, 10, 64)	0
dropout_3 (Dropout)	(None, 10, 10, 64)	0
flatten (Flatten)	(None, 6400)	0
dense (Dense)	(None, 512)	3277312
dense_1 (Dense)	(None, 1)	513

Total params: 3,338,401
Trainable params: 3,338,369
Non-trainable params: 32

Figure B-1: Summary of the layers in the final CNN model

Appendix C

Results Train-Test Cycles Overview

The results and outputs of the 10 train-test cycles, together with the output parameters on average, can be seen in Table C-1. Figure C-1 is a boxplot of these output parameters over the 10 train-test cycles, showing the range and the mean value per parameter.

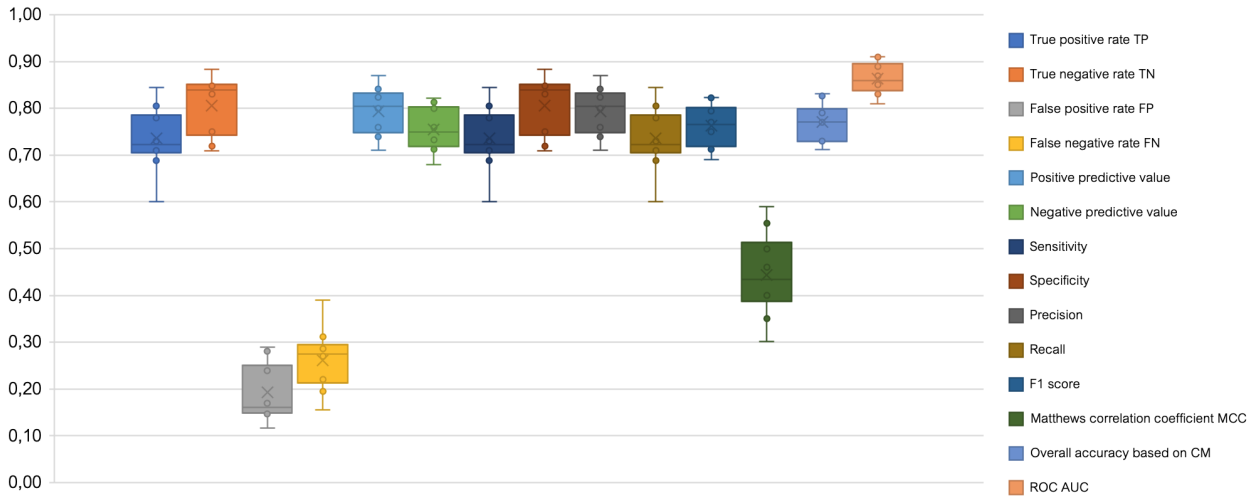


Figure C-1: Boxplot of model outcomes, the legend list order from top to bottom corresponds to left to right in the graph.

Table C-1: Outputs per training cycle

Output value	Training Cycle 1	2	3	4	5	6	7	8	9	10	Average
True positive rate TP	0,81	0,84	0,71	0,69	0,78	0,60	0,71	0,71	0,73	0,78	0,74
True negative rate TN	0,85	0,72	0,71	0,85	0,88	0,85	0,75	0,83	0,85	0,76	0,81
False positive rate FP	0,15	0,28	0,29	0,15	0,12	0,15	0,24	0,17	0,15	0,24	0,19
False negative rate FN	0,19	0,16	0,29	0,31	0,22	0,39	0,29	0,28	0,27	0,22	0,26
Positive predictive value	0,84	0,75	0,71	0,82	0,87	0,80	0,74	0,81	0,83	0,76	0,79
Negative predictive value	0,81	0,82	0,71	0,73	0,80	0,68	0,72	0,74	0,76	0,77	0,76
Sensitivity	0,81	0,84	0,71	0,69	0,78	0,60	0,71	0,71	0,73	0,78	0,74
Specificity	0,85	0,72	0,71	0,85	0,88	0,85	0,75	0,83	0,85	0,76	0,81
Precision	0,84	0,75	0,71	0,82	0,87	0,80	0,74	0,81	0,83	0,76	0,79
Recall	0,81	0,84	0,71	0,69	0,78	0,60	0,71	0,71	0,73	0,78	0,74
F1 score	0,82	0,79	0,71	0,75	0,82	0,69	0,72	0,76	0,78	0,77	0,76
Matthews correlation coefficient MCC	0,55	0,41	0,30	0,47	0,59	0,40	0,35	0,46	0,5	0,41	0,44
Overall accuracy based on CM	0,83	0,78	0,71	0,77	0,83	0,73	0,73	0,77	0,79	0,77	0,77
ROC AUC	0,91	0,87	0,81	0,89	0,91	0,83	0,85	0,85	0,87	0,84	0,86

Appendix D

Results Train-Test Cycles per Run

The results and outputs of the 10 train-test cycles are visualized per run in this Appendix. The confusion matrix and ROC curve are shown per run. Moreover, the accuracy and the loss during the training process are plotted against the number of epochs. Each run uses a different random split of train test data, based on random assignment of patient numbers in the 80:20 (train:test) ratio.

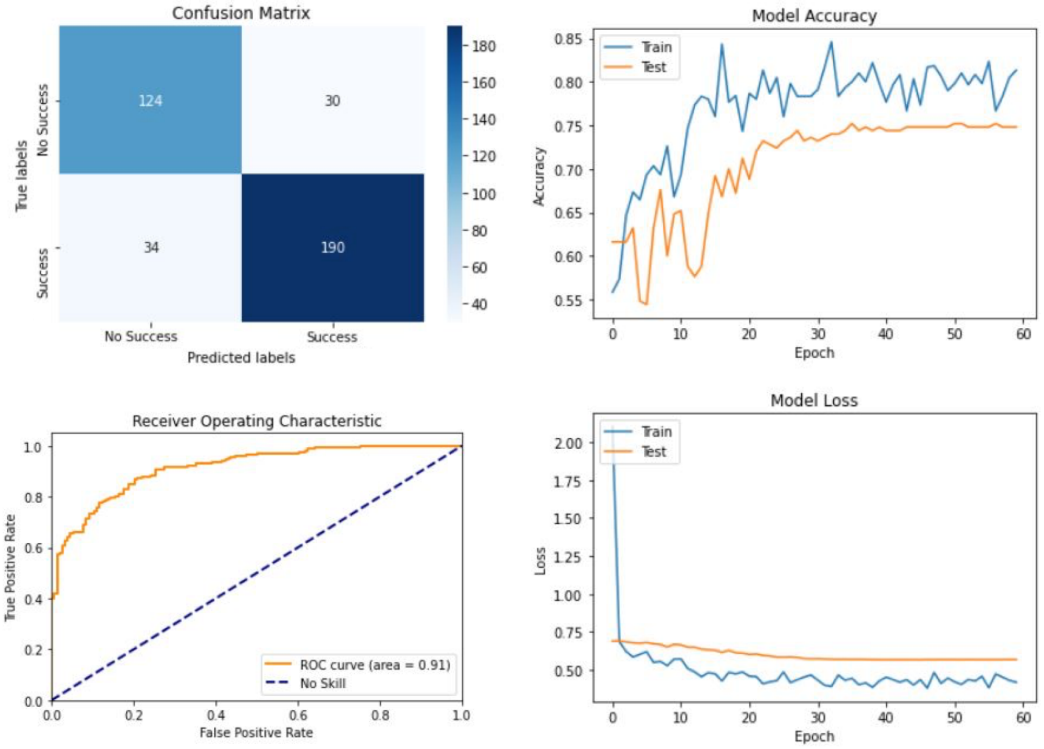


Figure D-1: Outcome metrics of training cycle number 1: *Top left:* Confusion Matrix, *Bottom left:* ROC curve with AUC value, *Top right:* Accuracy of the CNN training and validation, per epoch, *Bottom right:* Loss of the CNN training and validation, per epoch

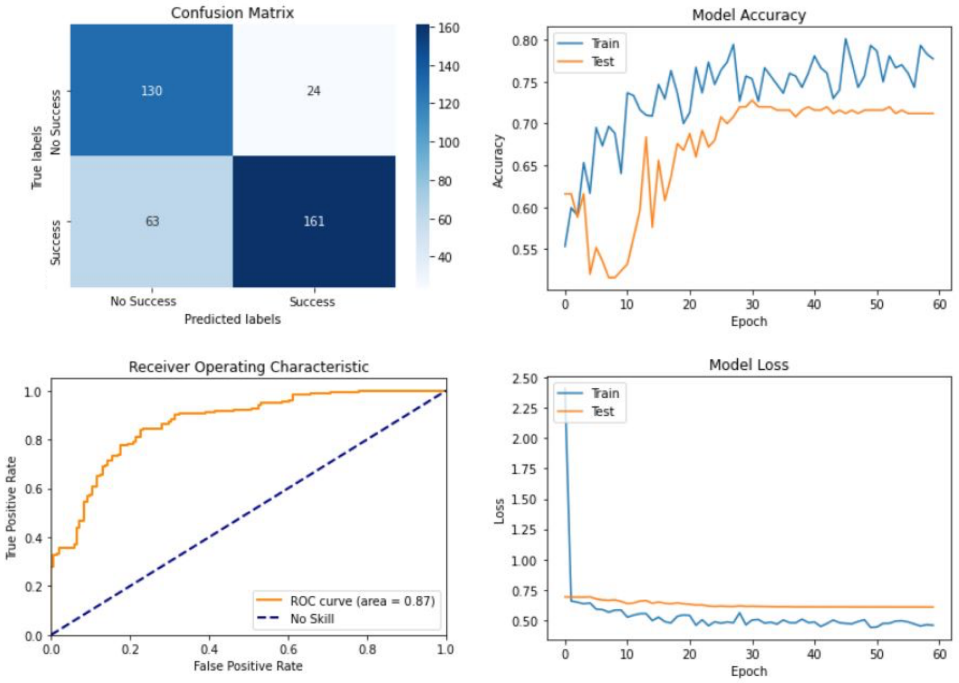


Figure D-2: Outcome metrics of training cycle number 2: *Top left:* Confusion Matrix, *Bottom left:* ROC curve with AUC value, *Top right:* Accuracy of the CNN training and validation, per epoch, *Bottom right:* Loss of the CNN training and validation, per epoch

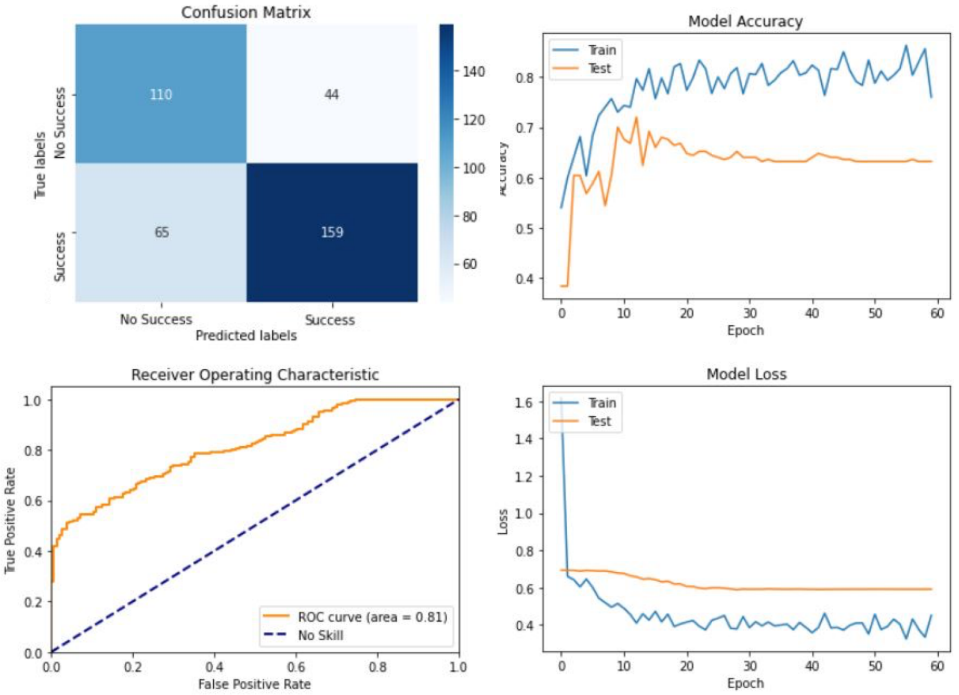


Figure D-3: Outcome metrics of training cycle number 3: *Top left:* Confusion Matrix, *Bottom left:* ROC curve with AUC value, *Top right:* Accuracy of the CNN training and validation, per epoch, *Bottom right:* Loss of the CNN training and validation, per epoch

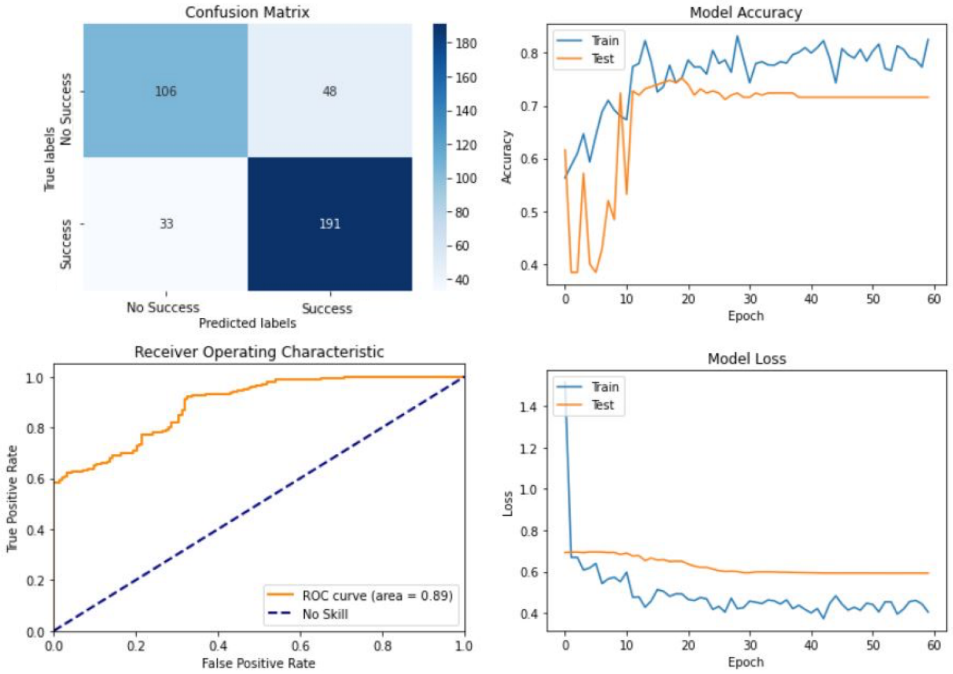


Figure D-4: Outcome metrics of training cycle number 4: *Top left:* Confusion Matrix, *Bottom left:* ROC curve with AUC value, *Top right:* Accuracy of the CNN training and validation, per epoch, *Bottom right:* Loss of the CNN training and validation, per epoch

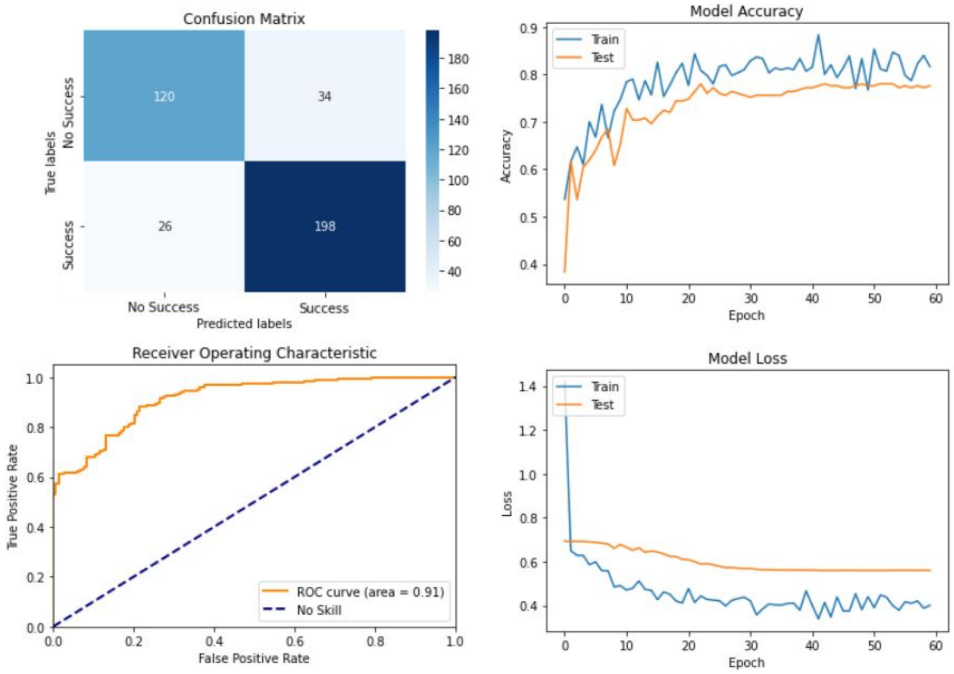


Figure D-5: Outcome metrics of training cycle number 5: *Top left:* Confusion Matrix, *Bottom left:* ROC curve with AUC value, *Top right:* Accuracy of the CNN training and validation, per epoch, *Bottom right:* Loss of the CNN training and validation, per epoch

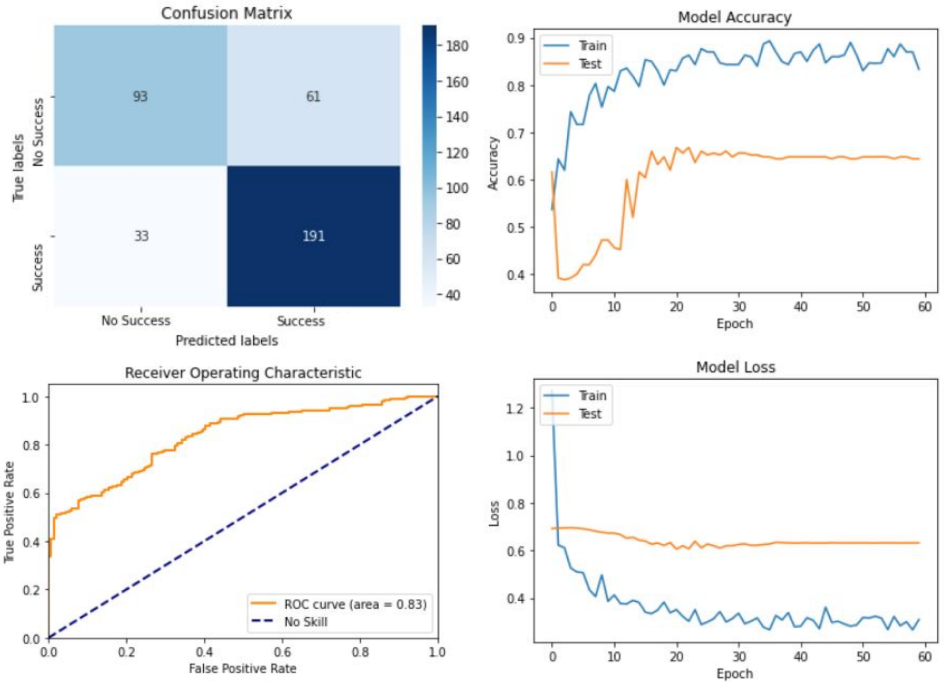


Figure D-6: Outcome metrics of training cycle number 6: *Top left:* Confusion Matrix, *Bottom left:* ROC curve with AUC value, *Top right:* Accuracy of the CNN training and validation, per epoch, *Bottom right:* Loss of the CNN training and validation, per epoch

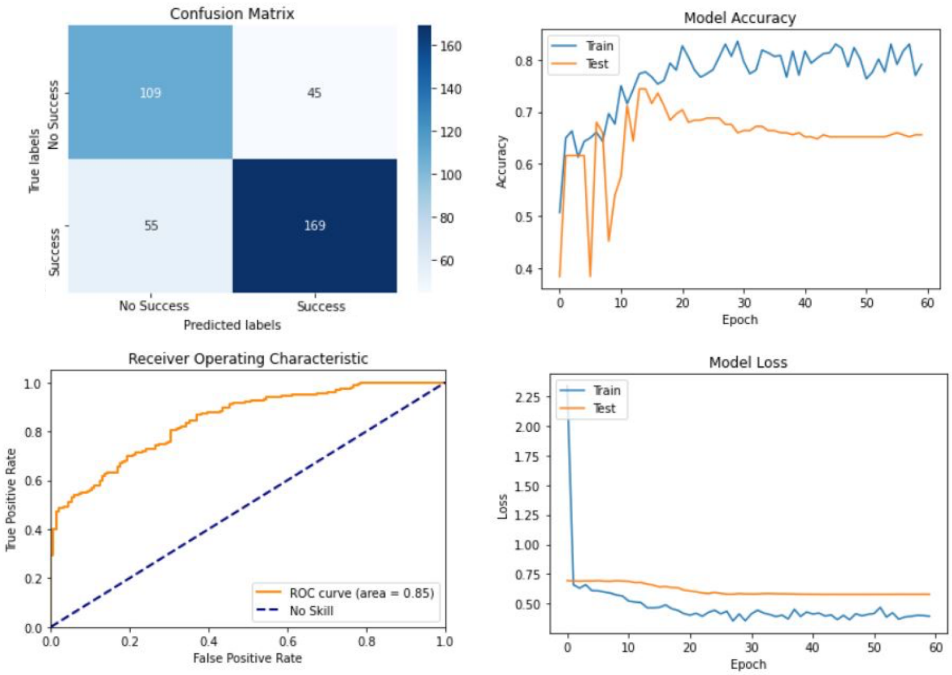


Figure D-7: Outcome metrics of training cycle number 7: *Top left:* Confusion Matrix, *Bottom left:* ROC curve with AUC value, *Top right:* Accuracy of the CNN training and validation, per epoch, *Bottom right:* Loss of the CNN training and validation, per epoch

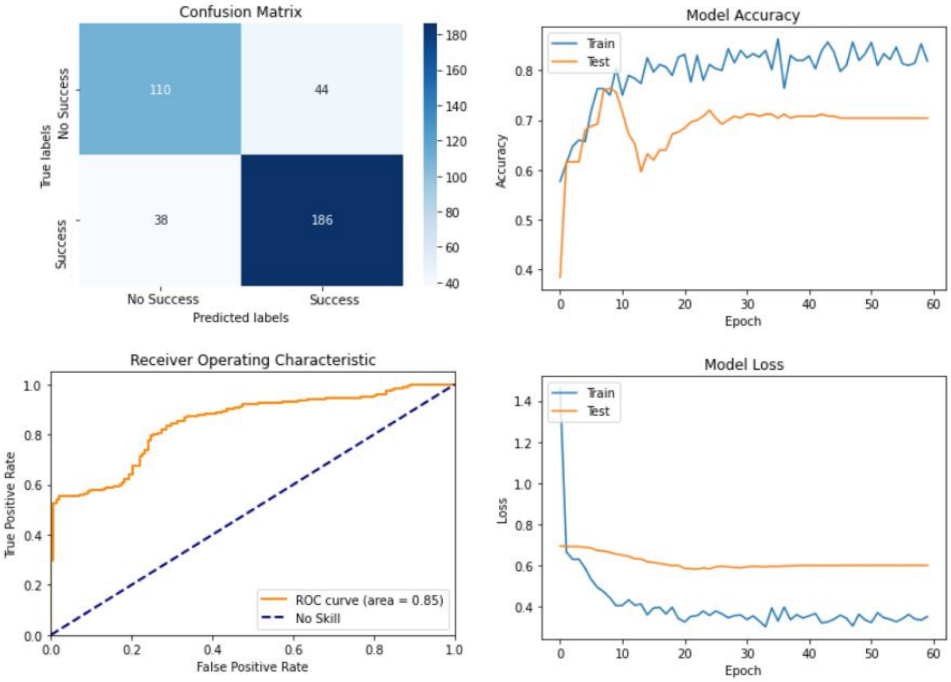


Figure D-8: Outcome metrics of training cycle number 8: *Top left:* Confusion Matrix, *Bottom left:* ROC curve with AUC value, *Top right:* Accuracy of the CNN training and validation, per epoch, *Bottom right:* Loss of the CNN training and validation, per epoch

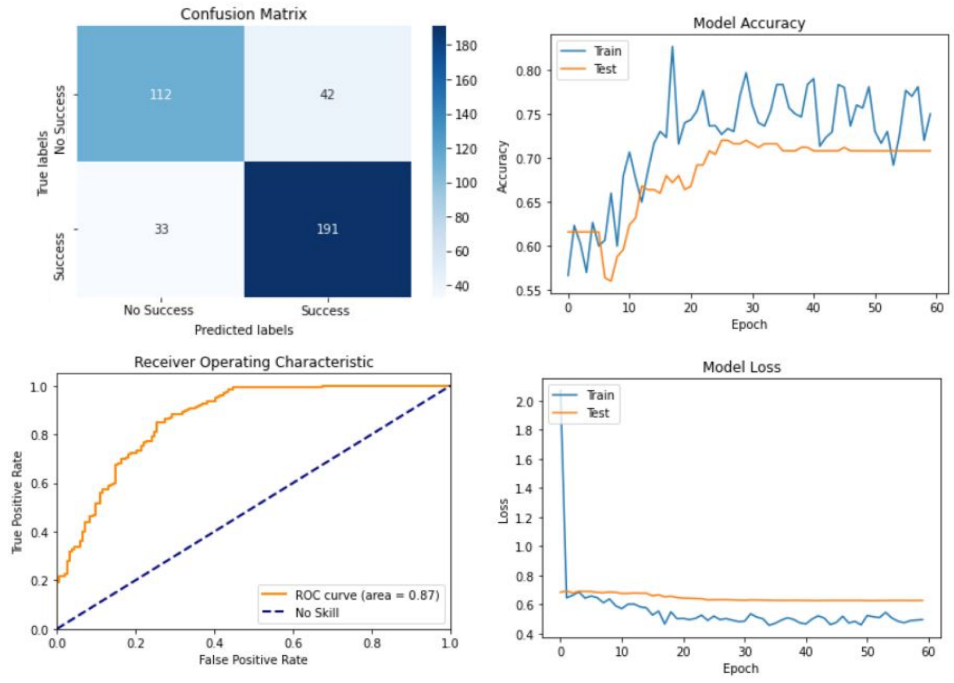


Figure D-9: Outcome metrics of training cycle number 9: *Top left:* Confusion Matrix, *Bottom left:* ROC curve with AUC value, *Top right:* Accuracy of the CNN training and validation, per epoch, *Bottom right:* Loss of the CNN training and validation, per epoch

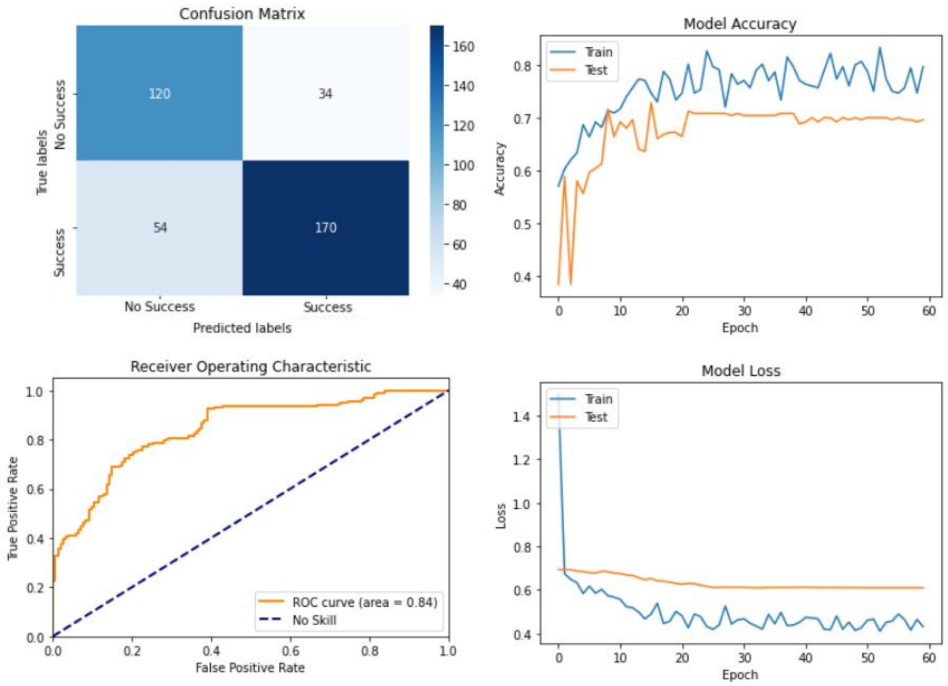


Figure D-10: Outcome metrics of training cycle number 10: *Top left:* Confusion Matrix, *Bottom left:* ROC curve with AUC value, *Top right:* Accuracy of the CNN training and validation, per epoch, *Bottom right:* Loss of the CNN training and validation, per epoch

Appendix E

Heatmaps (Grad-CAM)

The heatmaps of different classes, patients and cervical orientations can be seen in this Appendix. The superimposed heatmaps on the X-ray images can be seen as well, to show the possible relation with certain anatomical structures.

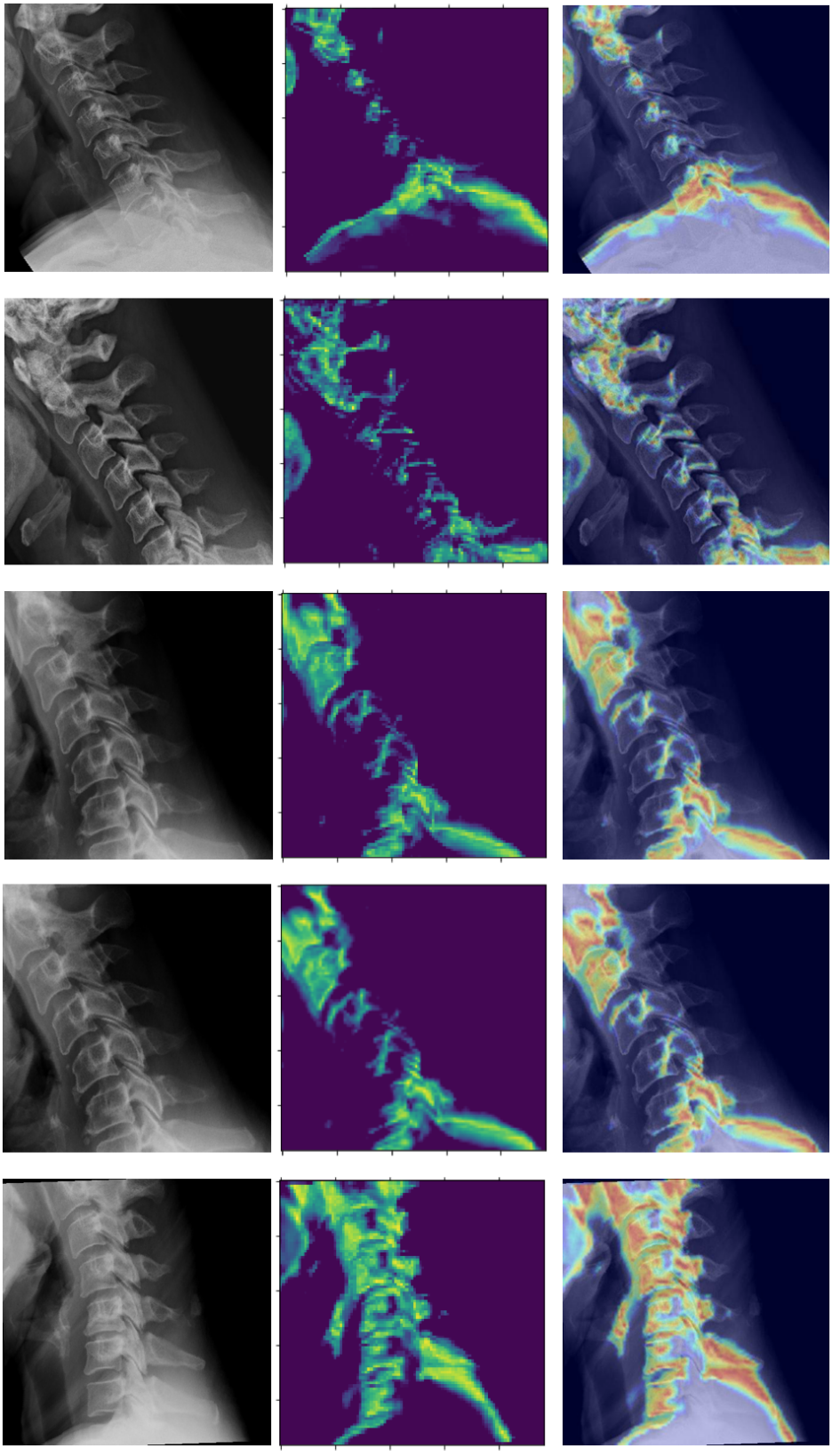


Figure E-1: Selection of heatmap images. Left column: original X-ray image, middle column: heatmap image, right column: heatmap superimposed on original X-ray image

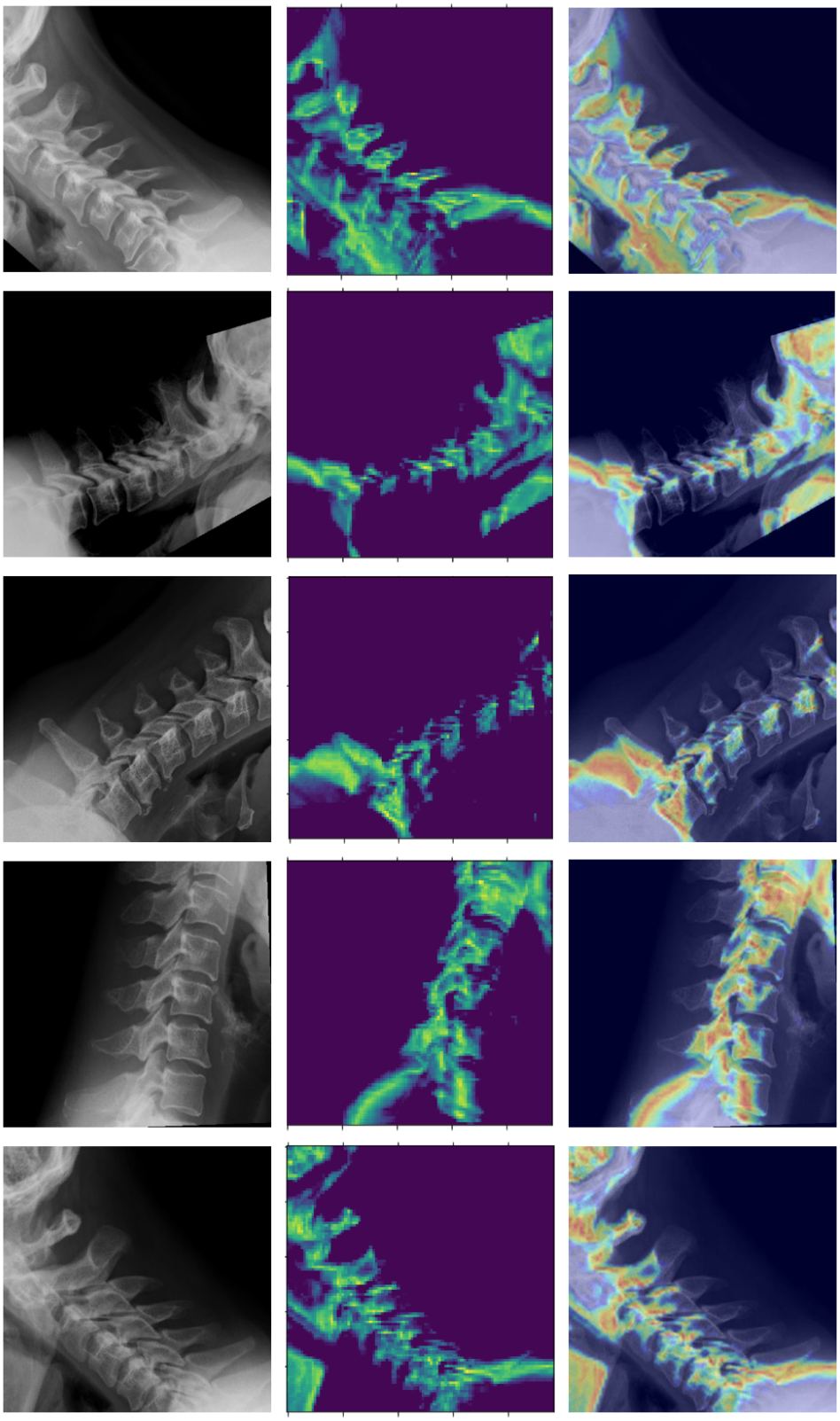


Figure E-2: Selection of heatmap images. Left column: original X-ray image, middle column: heatmap image, right column: heatmap superimposed on original X-ray image