Machine Learning in Magnetic Resonance-Guided Adaptive Radiotherapy

SAMUEL FRANSSON
Abstract

In radiotherapy, treatments are frequently distributed over multiple weeks, and the radiation dose delivered across several sessions. A significant hurdle in this approach is the anatomical changes that occur between the planning stage and subsequent treatment sessions, leading to uncertainties in the treatment. The MR-Linac system, which combines a linear accelerator with an MRI scanner, addresses this issue by allowing for daily adjustments to the treatment plan based on the patient's current anatomy. However, the process for making these adjustments, involving image fusion, re-contouring, and plan re-optimization, can be quite elaborate and time-consuming. This project aimed to identify opportunities within the daily treatment routine where machine learning and deep learning could streamline the process, thereby enhancing efficiency, with a focus on prostate cancer treatments due to their frequent occurrence at our facility. We leveraged deep learning to train patient-specific models for segmenting anatomical structures in daily MRI scans, matching the accuracy of existing deformable image registration techniques. Furthermore, we extended this concept to segmenting structures and predicting radiation dose distributions, offering a swift assessment of potential dose distribution before engaging in the more complex manual workflow. This could aid in selecting the most suitable adaptation method more quickly. Additionally, we developed motion models for intrafractional motion and for segmenting images at lower resolutions to facilitate a target tracking process. Throughout this project, we showed how machine learning and deep learning techniques could contribute to optimizing the daily MR-Linac workflow.

Samuel Fransson, Department of Surgical Sciences, Radiology, Akademiska sjukhuset, Uppsala University, SE-75185 Uppsala, Sweden.

© Samuel Fransson 2024

ISSN 1651-6206
ISBN 978-91-513-2114-1
URN urn:nbn:se:uu:diva-526296 (http://urn.kb.se/resolve?urn=urn:nbn:se:uu:diva-526296)
Till min familj
List of Papers

This thesis is based on the following papers, which are referred to in the text by their Roman numerals.

   My contributions: Conceptualization, Data Curation, Formal Analysis, Methodology, Software, Visualization, Writing – Original Draft Preparation

   My contributions: Conceptualization, Data Curation, Formal Analysis, Methodology, Software, Visualization, Writing – Original Draft Preparation

   My contributions: Conceptualization, Data Curation, Formal Analysis, Methodology, Resources, Software, Visualization, Writing – Original Draft Preparation

   My contributions: Conceptualization, Data Curation, Formal Analysis, Methodology, Software, Visualization, Writing – Original Draft Preparation

Reprints were made with permission from the respective publishers.
Related publications

V. Fransson, S. Comparing multi-image and image augmentation strategies for deep learning-based prostate segmentation. *Physics and Imaging in Radiation Oncology*, 29(2024)
Contents

1. Introduction........................................................................................................ 11
   1.1 Radiotherapy .................................................................................................. 11
   1.2 MR-guided adaptive radiotherapy with an MR-Linac ....................... 12
   1.3 Prostate radiotherapy ................................................................................. 14
   1.4 Machine learning in radiotherapy ............................................................... 14
   1.5 Purpose and aim ............................................................................................. 15

2. Deep learning ..................................................................................................... 16
   2.1 Feed-forward neural network ....................................................................... 16
   2.2 The activation function ................................................................................ 17
   2.3 The loss function .......................................................................................... 17
   2.4 Backpropagation and optimization ............................................................... 18
   2.5 Convolutional neural networks ...................................................................... 20
   2.6 Preventing overfitting ................................................................................... 20
      2.6.1 Network weight regularization .................................................................. 21
      2.6.2 Data augmentation .................................................................................. 21
   2.7 Transfer learning and fine-tuning ................................................................ 22
   2.8 Uncertainty quantification ........................................................................... 23
      2.8.1 Network ensembles for uncertainty quantification .................................. 24
   2.9 Multitask learning ......................................................................................... 25

3. Image segmentation ............................................................................................ 26
   3.1 Deep learning image segmentation in radiotherapy .............................. 26
      3.1.1 Evaluation metrics ................................................................................ 27
   3.2 Image segmentation for MR-Linac ............................................................... 28
      3.2.1 Patient-specific deep learning image segmentation ............................ 28
   3.3 Image segmentation uncertainty ................................................................. 29

4. Radiotherapy decision support ....................................................................... 31
   4.1 Deep learning dose prediction ..................................................................... 32
      4.1.1 Deep learning dose prediction for MR-Linac ........................................ 33

5. Real time motion management ........................................................................... 36
   5.1 Statistical motion models ............................................................................. 36
      5.1.1 Extracting the motion ........................................................................... 36
      5.1.2 Motion model through principal component analysis ............... 37
      5.1.3 Non-periodic motion ......................................................................... 38
5.1.4 Non-periodic motion model based on principal component analysis .............................................................................. 39
5.2 Deep learning-based motion models .................................................. 40
  5.2.1 Image super-resolution ............................................................... 40
  5.2.2 Simultaneous real-time image segmentation and super-resolution ................................................................................... 42
6. Summary and outlook ........................................................................... 44
7. Sammanfattning på svenska ...................................................................... 46
8. Acknowledgement .................................................................................... 48
9. References ................................................................................................. 49
<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RT</td>
<td>Radiotherapy</td>
</tr>
<tr>
<td>OAR</td>
<td>Organ At Risk</td>
</tr>
<tr>
<td>VMAT</td>
<td>Volumetric Modulated Arc Therapy</td>
</tr>
<tr>
<td>LINAC</td>
<td>Linear Accelerator</td>
</tr>
<tr>
<td>MLC</td>
<td>Multi-Leaf Collimator</td>
</tr>
<tr>
<td>CT</td>
<td>Computed Tomography</td>
</tr>
<tr>
<td>MRI</td>
<td>Magnetic Resonance Imaging</td>
</tr>
<tr>
<td>QA</td>
<td>Quality Assurance</td>
</tr>
<tr>
<td>CBCT</td>
<td>Cone Beam Computed Tomography</td>
</tr>
<tr>
<td>PTV</td>
<td>Planning Target Volume</td>
</tr>
<tr>
<td>ATP</td>
<td>Adapt To Position</td>
</tr>
<tr>
<td>ATS</td>
<td>Adapt To Shape</td>
</tr>
<tr>
<td>MSE</td>
<td>Mean Squared Error</td>
</tr>
<tr>
<td>MAE</td>
<td>Mean Absolute Error</td>
</tr>
<tr>
<td>CNN</td>
<td>Convolutional Neural Network</td>
</tr>
<tr>
<td>DSC</td>
<td>Dice Similarity Coefficient</td>
</tr>
<tr>
<td>DVC</td>
<td>Dose Volume Constraint</td>
</tr>
<tr>
<td>ITV</td>
<td>Internal Target Volume</td>
</tr>
<tr>
<td>PCA</td>
<td>Principal Component Analysis</td>
</tr>
<tr>
<td>DVF</td>
<td>Displacement Vector Field</td>
</tr>
<tr>
<td>SR</td>
<td>Super Resolution</td>
</tr>
<tr>
<td>PSNR</td>
<td>Peak Signal-to-Noise Ratio</td>
</tr>
<tr>
<td>GAN</td>
<td>Generative Adversarial Network</td>
</tr>
</tbody>
</table>
1. Introduction

1.1 Radiotherapy

Radiotherapy (RT) involves treating cancer using ionizing radiation, and in Sweden, almost half of all cancer patients undergo this treatment [1]. The objective is to administer a sufficiently high radiation dose to a specific target while minimizing exposure to healthy tissue and organs at risk (OARs). External Beam Radiotherapy is the most common form, where a linear accelerator (Linac) emits ionizing radiation from outside of the patient, either through static fields at predetermined angles or by e.g. VMAT (Volumetric Modulated Arc Therapy) where the Linac continuously rotates around the patient during irradiation. The radiation beam is shaped to the target shape using blocks and a Multi-Leaf Collimator (MLC) that are placed in the beam path as seen from each angle. Typically, the radiation dose is fractionated, delivered according to a predetermined schedule (usually one fraction per day) over several days to facilitate the recovery of healthy organs. The RT process begins with a simulation, during which Computed Tomography (CT) and/or Magnetic Resonance Imaging (MRI) images are acquired as planning references for the target and relevant OARs. The physician defines the treatment volume and OARs on these planning images.

A treatment plan is then created to balance predefined planning for the target and OAR goals. This planning, including quality assurance (QA), usually occurs over several days the weeks before the start of the actual treatment. At each treatment fraction, the patient is positioned as closely as possible to the CT and/or MRI setup, and setup accuracy can be confirmed through e.g. Cone Beam CT (CBCT), an imaging method performed in the treatment room, and surface scanning [2]. However, a significant uncertainty in the workflow is the inherent variability in the patient anatomy. The treatment plan is based on images acquired at a single point in time, while the treatment may be performed over several weeks, during which the anatomy can change, something called interfractional motion. Safety margins are applied by expanding the target structure, creating a PTV (Planning Target Volume), to mitigate this uncertainty [3], [4], leading to a larger treated volume. Additionally, the relative position between the target and OARs may change, impacting the dose to OARs compared to what was initially planned. While CBCT images can confirm the setup accuracy, they often lack soft tissue contrast and are more suited
for matching bony structures. Gold fiducial markers visible on CBCT can address this issue in some cases for the target, but their insertion involves an invasive procedure and carries the risk of marker migration. Intrafractional motion, or motion during each treatment fraction (e.g., breathing, muscle relaxation etc.), is another aspect of motion. Techniques like surface scanning during treatment can be used to pause or abort the treatment, triggered by movement above a certain threshold. However, such techniques rely on an external surrogate (patient outer contour) for internal motion, rather than providing a direct visualization of the anatomy. MR imaging, which provides high soft tissue contrast, is inherently capable of visualizing the internal anatomy and is hence a suitable imaging modality to handle both interfractional and intrafractional motion.

1.2 MR-guided adaptive radiotherapy with an MR-Linac

Adaptive radiotherapy involves adjusting treatment plans based on systematic feedback [5] to enhance radiation treatment by addressing inherent uncertainties like anatomical variations. The feedback can range in sophistication, from periodic modifications using CT-image acquisitions to more frequent updates with replanning on daily CBCT images. Further aspects include real-time gating (where the beam is turned on and off based on the target/surrogate location) and in the future MLC-tracking (where the beam shape is adjusted to follow the target during irradiation).

The MR-Linac, a hybrid machine combining an MR-scanner and a Linac, represents the cutting edge of adaptive RT, providing unparalleled soft tissue image quality compared to CBCT for daily replanning based on the evolving anatomy. Daily adaption to the anatomy could allow for e.g. reduced treatment margins and thereby reduced side effects of the treatment. In Elekta's implementation of the MR-Linac, named Unity (referred to as the MR-Linac hereafter), two distinct workflow options are available: Adapt To Position (ATP) and Adapt To Shape (ATS). ATP involves a rigid shift of the original anatomy, delivering a dose distribution similar to the dose distribution of the reference plan (obtained before the first fraction) at the new position. On the other hand, ATS comprises an initial contour propagation using deformable image registration (DIR) to establish contours on the daily image set, followed by manual correction and plan adaptation. While ATS represents a lengthier workflow, it can handle any type of interfractional motion. Figure 1 illustrates the ATS workflow for a prostate treatment at our institution using the MR-Linac.
Figure 1. Adapt To Shape workflow for a prostate treatment at our MR-Linac. Prior treatment, a reference plan is made, including imaging, delineation, and dose planning. At the treatment, contours are propagated onto the daily planning 3D MR-image using deformable image registration. The contours are manually adjusted, and a new plan is optimized for the new image and structures. During planning, an additional 3D image is acquired and compared to the planning image to determine any motion taking place during the planning process. While treating (beam on) rapid 2D images are acquired to see motion in real time, followed by another 3D image.

Each step in the online workflow can be considered time-critical. Not only is patient comfort an important consideration but also the correspondence between the anatomy at planning and treatment times. The longer the duration of the planning phase the higher the possibility the anatomy has changed prior to delivery, which in turn may cause yet another replanning. Manual adjustments of the contours preceded by the contour propagation step is one of the most time-consuming aspects. In Figure 2, an example of the average durations contributed by each component of a prostate treatment fraction at our institution is shown.
1.3 Prostate radiotherapy

Prostate cancer is the most common form of cancer in Sweden [6]. At the time of writing, this is the most common treatment at the MR-Linac at our institution. For this reason, this patient group was selected as the basis for this project.

1.4 Machine learning in radiotherapy

Machine learning refers to the capacity of machines to learn a specific task without explicit instructions [7]. Typically, machines are provided with an extensive set of training data tailored to the task at hand, enabling them to autonomously learn and execute the task. A specific subset of machine learning includes deep learning and artificial neural networks, drawing inspiration from the connectivity principles observed in the biological brain. The applicability of machine learning, especially deep learning, is particularly pronounced when dealing with large datasets.

In the domain of radiotherapy, where significant volumes of data, in the form of images collected before, during, and after a patient's treatment, are becoming standard, machine learning emerges as a valuable tool. This data can be
exploited to train machine-learning approaches contributing to the entire treatment chain. Examples of such applications include automating image segmentation to expedite the delineation process, predicting radiation doses for rapid plan assessments before full optimization, and generating synthetic CT-images from MR-images. In the context of MR-Linac treatments, an important goal is to enhance workflow efficiency without compromising treatment quality. Machine learning alternatives have the potential to streamline the entire treatment process, ensuring both speed and increased consistency.

1.5 Purpose and aim

The purpose of this project was to explore machine learning techniques within the daily adaptive radiotherapy process utilizing the MR-Linac. The primary objective was to develop and evaluate methods that can improve existing procedures by focusing on three key areas:

1. Deep learning for segmentation within the MR-Linac process.
2. Deep learning

Deep learning is a type of machine learning utilizing artificial neural networks. The term “deep” stems from the stacking of several layers of artificial neurons after each other. In the following sections, we go through some key aspects of training a deep learning network, and some specific techniques that we have used during this project.

2.1 Feed-forward neural network

Let’s start with one rudimentary example of a deep learning implementation. The feed-forward neural network operates in a unidirectional manner, where information flows from one layer to the next. Visualized as nodes in different layers with interconnected pathways, a simplified illustration is presented in Figure 3. This network comprises an input layer, two hidden layers, and an output layer. In this representation, each node is regarded an artificial neuron, receiving inputs from preceding layers and transmitting outputs to the subsequent layers.

Figure 3. Example of a feed-forward neural network (left) and the artificial neuron (right).
The mathematical expression for a single neuron is expressed as follows:

\[ y = \varphi \left( \sum_j w_j x_j + b \right) \]  

(1)

Here, \( x_j \) denotes the \( j \)-th input to the neuron, \( w_j \) is the corresponding weight, \( b \) represents a bias term, and \( \varphi \) signifies a non-linear activation function. The intricacy of the network arises from the interconnection of numerous such artificial neurons.

2.2 The activation function

The activation function introduced in equation 1 is crucial. Without a non-linear activation function, the network would lack the necessary complexity for effective learning. One commonly used non-linear activation function is the Rectified Linear Unit (ReLU) [8], defined as \( \varphi(x) = \max(0, x) \). This simple yet effective function helps introduce non-linearity to the network. Other variants include e.g. LeakyReLU, Exponential Linear Unit (ELU) [9], and Scaled Exponential Linear Unit (SELU) [10]. These variants are often employed in the input and hidden layers of the network, each with its advantages. Depending on the desired output, different activation functions may be applied in the output (last) layer. For tasks like logistic regression, where the network serves as a classifier, functions such as the logistic function (sigmoid) or the multinomial variant softmax are commonly used. These functions constrain the output to the range \([0,1]\), treating it as a probability assigned by the network to its prediction.

2.3 The loss function

Before the network can be used it needs to be trained, to achieve a good set of values for all the parameters of the network. Supervised learning achieves this from an (often very large) set of labeled data. As an example, training a network to classify images requires a set of images \((x)\) and the corresponding labels \((y_{true})\). Whilst training, an image is fed through the network and produces an output \(y_{pred}\). To determine how “good” this prediction is compared to the true value \((y_{true})\) we need a loss (or objective) function, denoted \(L\).
The choice of loss function is highly dependent on the task at hand. To mention a few, in the case of image classification the cross-entropy function is often applied:

\[ L_{ce} = - \sum_{c=1}^{N} y_{true,c} \log(y_{pred,c}) \]  

(2)

where \( c \) denotes the class and \( N \) is the total number of classes. Here, \( y_{true,c} \) is 1 for the true label and 0 otherwise. Other common loss functions are the Mean Squared Error (MSE):

\[ L_{MSE} = \frac{1}{N} \sum_{i=1}^{N} (y_{true,i} - y_{pred,i})^2 \]  

(3)

or the Mean Absolute Error (MAE):

\[ L_{MAE} = \frac{1}{N} \sum_{i=1}^{N} |y_{true,i} - y_{pred,i}| \]  

(4)

where \( i \) denotes the \( i:th \) observation and \( N \) the total number of observations.

2.4 Backpropagation and optimization

After conducting a forward pass and determining the loss function value for the current instance, the next step is updating the network weights. The loss function is typically designed to be minimized, signifying that lower values indicate a better alignment between predicted and true values. Good weights are those that achieve this minimization. To accomplish this, a process known as backpropagation is employed, in which the gradient of the loss function with respect to each parameter is calculated [11]–[13]. For each parameter, a step is taken in the direction that minimizes the loss function. One common optimization method is gradient descent, where each parameter is updated using the formula:
Here, $w_{\text{new}}$ and $w_{\text{old}}$ are the new and old values of the parameter, respectively, and $\eta$ is the learning rate, governing the step size. An example of the optimization process is given in Figure 4 using the MNIST database [14], [15]. This is a large database consisting of handwritten digits commonly used for training image processing systems.

![Figure 4](image)

Figure 4. Example of an optimization step for an image classification network. The image is fed to the network yielding a predicted output. The loss function determines the dissimilarity between the predicted output and true label, and an optimizer updates the network weights to minimize this. In this case, the objective is to classify handwritten digits in the range 0-9, hence the network output is a probability for each digit in this range.

While traditional gradient descent computes the gradient using the entire training dataset, making it computationally demanding, stochastic gradient descent [16] computes the gradient for a single instance, resulting in more frequent but noisy parameter updates. A practical compromise involves using mini-batches, where a subset $n$ of the training data (where $1 < n < N$, and $N$ is the training dataset size) is employed for parameter updates. This approach strikes a balance between the extremes of traditional and stochastic gradient descent. Several more advanced optimization algorithms are available, where Adam [17], RMSprop [18], and Adagrad [19] are some alternatives.
2.5 Convolutional neural networks

When addressing image data, fully connected networks encounter limitations. Images, characterized by a substantial number of pixels (or voxels in 3D), pose a challenge for fully connected networks. The process of feeding images into such networks involves "flattening" the image. For instance, a 28x28 pixel image is transformed into a 1x784 vector before being fed into the network. However, this method overlooks the spatial interdependence of pixels. Pixels in proximity to each other are not entirely independent, and a fully connected network may struggle to capture the spatial relationships. Moreover, fully connected networks are not inherently equipped to handle translation invariance, where the interpretation of features should remain consistent regardless of their position in the image.

A convolutional neural network (CNN) offers a more suitable approach [20], [21]. In a CNN, convolutional kernels (or filters), comprised of trainable parameters, are convolved with the image to generate feature maps, as exemplified in Figure 5.

![Figure 5. Example of convolution in two layers. The image is convolved with a set of filters (consisting of weights) creating feature maps. The resulting feature maps are in turn convolved with a set of filters to create the next level of features.](image)

Analogous to fully connected networks, convolutional layers can be stacked depth-wise, progressively creating feature maps with increasing abstraction levels. Fully convolutional networks, in contrast to fully connected networks, work independently of image size.

2.6 Preventing overfitting

Overfitting occurs when a neural network essentially memorizes the training data, learning features not applicable to new, unseen data. It is important to notice that the goal is to use the network on data it hasn't encountered during training. To address this, a portion of the training data is typically set aside as validation data during training. This dataset isn't used to update weights
directly; instead, the network's performance is continuously assessed on it. The accuracy on the validation data starts to increase as training progresses, but at some point, the accuracy on this dataset may start to drop. This suggests that the network is capturing specific features in the training set that don't generalize well. To mitigate overfitting, options include collecting a larger training dataset, reducing the number of network parameters, or employing regularization and/or augmentation techniques. The final performance of the model is then determined on a held-out test set, not part of the training procedure in any way.

2.6.1 Network weight regularization

One way to reduce the complexity of a neural network is to introduce a penalty term on the weights through regularization. This is achieved as an addition to the content-based loss function, denoted as

\[ L = L(y_{\text{true}}, y_{\text{pred}}) + \alpha \Omega(w) \]  

(6)

and combines the network's performance on the training dataset with an additional regularization term \( \Omega(w) \), weighted by \( \alpha \). Common regularization techniques are the L1 and L2 norms, which respectively involve adding the sum of absolute values and the sum of squares of the parameters to the total loss function. The L1 norm promotes sparsity by pushing parameters towards zero, while the L2 norm focuses more on higher-value parameters.

Another effective approach is the use of dropout [22], [23], a technique that randomly drops connections with a certain probability during training. This prevents the network from relying too heavily on specific connections. Dropout can be applied to all layers or a subset of them, with the dropout probability being a parameter set by the user or integrated into the optimization process [24]. There are different methods of dropout, with some tailored to CNNs [25], but the main purpose remains despite the specific implementation.

2.6.2 Data augmentation

Augmenting a dataset is an effective approach to reduce overfitting and enhance network performance, particularly in scenarios with limited data availability. In image processing, several simple augmentation strategies can be employed, for instance:
1. **Image flipping**: Can be done both vertically and horizontally
2. **Contrast**: Alter image contrast and brightness
3. **Resampling**: Change the image pixel size through interpolation
4. **Noise**: Add noise to the image
5. **Elastic deformation**: Deform the images
6. **Crop**: Remove a portion of the image

Examples of the augmentations are given in Figure 6

![Image examples](image.png)

Figure 6. Examples of image augmentation strategies.

However, it is important to select augmentation strategies based on the use-case and avoid applying augmentations that alter the class of the data. For instance, in digit classification, flipping a "6" could change its class.

### 2.7 Transfer learning and fine-tuning

In general, models are trained on a certain domain to perform a certain task [26]. For example, the domain might be CT-images and the task might be to segment (see later chapter on image segmentation) the prostate. The training data hence consists of a large number of CT-images, where the prostate has been manually delineated. After training and validation of the network, it can be used on new data to aid the clinicians by segmenting the prostate on unseen CT-images. The network is hence bound to the domain and the task. It cannot segment the prostate on MR images, and it cannot segment e.g. the bladder on CT images, since it was not trained to do so. MR is inherently a bit tricky in this context since it can provide a virtually infinite number of different contrasts that alter the domain. To work on a different domain/task, the network could be re-trained with a new training dataset. However, it might not always be necessary to re-train from randomly initialized weights (i.e. “from scratch”).

22
Transfer learning involves utilizing the knowledge gained from a pre-trained network and applying it to a new domain or task [26]. The advantage lies in potentially achieving better performance compared to training from scratch, demands less training data, and converges more quickly. For instance, a pre-trained network designed for segmenting the prostate in CT images can be transferred to perform a similar task, such as segmenting the prostate in MR images. This process involves "freezing" some of the weights in the pre-trained network and updating only a part of the network. Additionally, the entire pre-trained network can be updated, a process known as fine-tuning, using a low learning rate to preserve prior knowledge and prevent completely overwriting the knowledge from the pre-trained network.

2.8 Uncertainty quantification

How can we trust the output of a neural network? One way is to assign certainty (or uncertainty) to the output, enhancing the trustworthiness of the network's predictions. When classifying images using the sigmoid or softmax activation function, the output can be interpreted as a probability. For instance, an output of 0.9 for a certain class is considered a more certain prediction than 0.5. However, this value is not always a reliable measure of uncertainty, since the models are often overly confident, lack accurate calibration, and cannot accurately distinguish out-of-distribution data [27], [28]. In terms of medical image segmentation, the uncertainty of the provided segmentations may be included as a part of the treatment margins.

Uncertainty can generally be divided into two classes: Epistemic and aleatoric uncertainty [29]. Epistemic uncertainty is linked to the inherent uncertainty in the network due to factors like insufficient information in the data for training. Although the theoretical solution to address this is rather straightforward (add more data), it might not always be a viable solution. Aleatoric uncertainty is the irreducible part, such as inherent noise in measurement equipment, not reduced by increasing the dataset size.

While training a standard network, a set of weights is determined, and the network becomes deterministic (feeding the same input data through the network several times yields the same output). However, we can instead treat the weights as a distribution upon which we sample during inference, creating a so-called Bayesian neural network [30]–[32]. This results in different outputs when prompted with the same input several times. For instance, a mean and variance of the resulting distribution can then be obtained as a measure of the network uncertainty in its prediction. A more practical alternative is using dropout as a Bayesian approximation [33]. Activating dropout also during
inference alters the network's prediction and provides a distribution of the output after multiple passes of the data.

2.8.1 Network ensembles for uncertainty quantification

An alternative method to Bayesian networks involves utilizing ensembles, which is primarily employed to enhance the accuracy of the final prediction. In this technique, multiple networks are trained to perform the same task, and their outputs are combined into a final prediction. Ensemble creation can be approached in various ways, with the main objective being to generate a set of diverse models.

One approach is to split the training data into distinct cohorts and conduct a K-fold cross-validation procedure. For instance, setting $K=5$ means splitting the training dataset into five groups. Then, one group is picked as validation data and the remaining as training data from which a model is trained. The process is repeated five times, with each fold acting as validation data once. Setting $K=N$ (where $N$ is the number of instances) is referred to as leave-one-out cross-validation. Alternatively, bootstrap aggregation can be employed, in which new training sets are constructed through sampling the original training set with replacement (such that each datapoint can be chosen several times). Another strategy involves using different sets of model architectures and training schedules, or a combination of both approaches. For example, the same network architecture can be used to generate different models, with variations in training attributed mainly to a different set of training data. Similarly, one can maintain the same training data but alter the architecture and/or network size to produce different models or employ a distinct set of loss functions. A simpler approach involves leveraging the stochastic nature of the training procedure. Even with the same data and architecture, separate network optimizations will not achieve the same set of weights due to factors such as random weight initialization, shuffling of training data, the use of dropout, or other randomly applied augmentation techniques.

The drawback of the ensemble technique is the need to train several models, which is time-consuming. Techniques like longitudinal ensembles can help mitigate this issue [34]. This approach generates multiple models during a single training run, introducing diversity through "restarts" during training. The learning rate follows a cyclic cosine annealing schedule, taking large steps at various points in the training process to effectively "restart" the training. It should be noted that all of these approaches require several passes of the data through the network(s), which could increase the inference time.
2.9 Multitask learning

Different tasks do not necessarily require to be learned in isolation. Learning several things at once can sometimes boost the performance. This is called multitask learning [35]. However, it requires judgment that the tasks are sufficiently similar and do not contradict each other. Weight sharing can be performed in different flavors. Hard parameter sharing means that the tasks share the same weights such that the total loss function is combined from the different tasks. Soft parameter sharing is when the tasks have separate networks and individual weights, but the distance between the weights is added to the loss function as an additional term, enforcing similarity between the sets of weights [36].
3. Image segmentation

Image segmentation is the pixel (or voxel)-wise classification of an image. This task could be further subdivided into semantic segmentation (segmenting each class of objects into a single segment) and instance segmentation (segmenting each object individually). Image segmentation is prominent in many areas, for instance in computer vision systems and autonomous driving [37]. In medicine, here focusing on radiotherapy, image segmentation tools are used to delineate organs on e.g. CT images, a prerequisite for the treatment preparations. Automatic image segmentation can be done at various levels of sophistication. Low-complexity methods with low computational demand encompass e.g. thresholding, region-growing, and edge-detection techniques [38]. Increasing the complexity entails the usage of shape models and atlas-based segmentation. Increasing the complexity even further will take us to deep learning-based approaches, which may supersede traditional techniques in performance [39]–[44].

There are two main advantages of using (accurate) automatic image segmentation methods for the delineation of organs in radiotherapy. First, it creates a good starting point for the clinician performing the task. Instead of starting from scratch, they are presented with a suggestion of sufficient quality such that manual adjustments require less time than starting from scratch. Secondly, observer variability is a prevalent issue [45]–[52]. Different clinicians, presented with the same image and task, would delineate differently (inter-observer effect). By providing a good starting point, minimal adjustments may create less variation.

3.1 Deep learning image segmentation in radiotherapy

Segmentation with deep learning networks is generally performed with a CNN. Likely the most common network architecture for deep learning-based image segmentation is the U-net [53], initially designed for segmentation of 2D biomedical images. This architecture can be described as an encoder-decoder type of network. In essence, the first half of the network encodes the input image into a latent (hidden) space, and the second half decodes this latent space into the requested output. During encoding, the size of the feature maps
(the results of the convolution operations) is increasingly reduced while the opposite happens during decoding, such that we arrive at an output of the same size as the input. This can be visualized in the shape of a “U”, yielding the architecture name. Moreover, there are so-called “skip-connections” between corresponding levels in the encoder and decoder. An outline of the basic architecture is given in Figure 7.

![U-net architecture diagram](image)

Figure 7. An outline of the U-net architecture consisting of an encoding part (left) and a decoding part (right), with skip connections in between.

This architecture has then been extended to 3D [54], which in radiotherapy may be a more suitable variant due to the 3D nature of the images for treatment planning. There also exist intermediate methods leveraging different views or multi-slice approaches [55]–[58]. Attention maps have also been added between the encoder and decoder, providing focus areas that can improve the network performance [59]. Turning the transformer architecture [60] into a vision transformer for image segmentation is also a viable alternative [61], [62].

### 3.1.1 Evaluation metrics

Usually, straightforward quantitative metrics, such as Dice similarity coefficient (DSC) [63] and Hausdorff distances [64], are common choices to disseminate the performance of image segmentation methods. Although such methods are easy to calculate and understand, they do not necessarily correlate to what’s clinically relevant. A common motivation for the utilization of image segmentation methods in radiotherapy is the time-saving aspect, i.e. the time saved between correcting the results from the image segmentation method vs starting from scratch. Directly obtaining this value during the method development process is rather unfeasible. Straightforward geometric metrics have been shown to poorly correlate with the time saved [65], [66]
and have prompted the development of metrics such as the Added Path Length [67].

3.2 Image segmentation for MR-Linac

In conventional radiotherapy workflow, most often only a single CT image is used per patient for treatment planning. In adaptive radiotherapy with an MR-Linac this is not the case. Here, new images are acquired (and contoured if an ATS workflow) on a daily basis. This opens the opportunity to create not just a task or domain-specific model, but rather a patient-specific model. One can utilize a pre-trained network and fine-tune on the images and segmentations from previous treatment fractions and use on the next fraction [68]–[72]. Or, in case of lacking a suitable pre-trained network, one can simply start from scratch.

3.2.1 Patient-specific deep learning image segmentation

In Paper II, we used images and segmentations of the CTV, bladder, and rectum from the first 17 prostate patients treated at our MR-Linac to create patient-specific deep-learning segmentation models. We chose a 2D U-net as the architecture. For each patient, we simply picked the images and structures from the first fraction to train a network and applied it to the images of the subsequent fractions. We did not utilize the image from the reference plan, since this was acquired at a different scanner with different sequence settings, producing a slightly different image contrast. In this case, we did not utilize any pre-trained network but rather trained each network from randomly initialized weights. Also, we did not update the networks continuously with more data (i.e. the network for fraction three could have been trained with the images from the first two fractions etc.). Nonetheless, we still found this approach performed on par with the clinically used DIR algorithm in the treatment planning system. In terms of DSC, small differences were seen between the proposed networks and the DIR algorithm. For CTV/bladder/rectum the values were 0.92/0.93/0.84 for the network and 0.95/0.93/0.88 for the DIR. Example results are seen in Figure 8. However, due to the (highly) limited size of the training dataset, several measures to stabilize the training procedure and decrease the risk of overfitting were taken. Each image was cropped around the volume of interest containing the structures, to decrease the otherwise very large portion of background voxels. Common image augmentation techniques such as noise injection, contrast augmentation, and random elastic deformations were employed. In addition, dropout was used in the bottom layer, and training was performed with randomly extracted patches of the images. We also implemented a simple variant of a horizontal ensemble. During training each model, three instances of the model weights were saved when the
model had reached convergence, and at inference an ensemble, taken as the average of all model’s outputs, was employed. The conclusion from this work is that patient-specific deep learning-based segmentation models could be a viable alternative to DIR-based methods for prostate treatments at the MR-Linac.

![Figure 8](image_url)

Figure 8. Example results from two patients (rows), each with segmentations on four fractions (columns). Yellow color signifies the ground truth (manually delineated) structures, green results from the clinically used deformable image registration, and red results from the patient-specific deep learning network.

### 3.3 Image segmentation uncertainty

The uncertainty in the segmentation can be quantified, and one way is to simply add dropout layers to our network both during training and inference. In this way, it is possible to produce a distribution of outputs after performing several forward passes (see Chapter 2.8 on uncertainty quantification). We performed this on a set of images and segmentations from prostate patients treated at our MR-Linac. When applied to a held-out test dataset, one could discern the uncertainty mainly along the structure borders (see Figure 9). This can be interpreted as the network being most uncertain about its prediction at these positions. Some of this uncertainty could be mitigated by increasing the training set size since this was rather limited in our experiments (images from 20 patients were used for training). However, some more irreducible uncertainty could be attributed to the variations in the manual segmentations serving as the ground truth.
Figure 9. Example of segmentation accompanied with uncertainty. Red contours show the prediction of the prostate (left), bladder (middle), and rectum (right), whereas dotted yellow contours indicate a region of uncertainty (±2 standard deviations) surrounding the prediction.
In the daily MR-Linac treatment, the whole workforce is under time pressure since most of the work is taking place while the patient is on the treatment table. As seen in the previous section, fast and accurate image segmentation methods are a viable solution to address one of the most time-consuming aspects of the whole workflow. However, having additional information such as an early estimate of the radiation dose distribution could further aid the clinician in the decision-making process, such as which overall workflow (ATP or ATS) to aim for.

For prostate treatments at our institution, currently the ATS workflow is used almost exclusively, although being a more labor-intensive and time-consuming workflow compared to ATP. We simulated the ATP workflow by rigidly aligning the center of mass of the target from the reference plan and the daily scans for 38 patients (in total 190 fractions) and shifted the reference dose distribution accordingly. In such a way we could compare the ATS and a simulated ATP workflow. For the rectum, considerable variation between the ATS and ATP workflow was occasionally noticed (see Figure 10), although the best choice of adaptation method is not unambiguous. The ability to predict the cases exhibiting the largest dosimetric gain could potentially help to determine when to apply ATS.
Dose prediction involves predicting the dose distribution without undergoing optimization within the treatment planning system. While optimizing a dose distribution can be laborious and time-consuming, obtaining a prediction from a deep learning-based approach can be accomplished within seconds. A reliable predicted dose can help at various stages in the treatment process. It can enable dosimetric feedback during the delineation process prior to full optimization [73], [74]. Predicted doses can guide the dose-planning process, particularly if the dose prediction network is trained on distributions from highly experienced planners. Additionally, a predicted dose can serve as an input to the optimizer in the treatment planning system[75]–[78]. In the context of MR-Linac treatments, a dose prediction can additionally aid in deciding between the ATP or ATS workflow.

Typically, the training data for a deep learning-based dose prediction network consists of pairs of structure sets and optimized dose distributions. The network takes the structure set as input and produces a dose distribution. Alternatively, an image can be added along with the structures as input. In CT-based planning, pixel values correlate with electron density, required for the actual dose calculation. Several works have performed deep learning-based dose prediction for conventional prostate treatments [76], [79]–[82]. Some focus on the MR-Linac workflow with decision support in mind [83]. Few works
directly perform dose prediction on the image without the need for a structure set [84]. However, to evaluate the obtained dose distribution, dose-volume constraints (DVCs) are based on structures, making them necessary. However, structures don't necessarily need to be manually delineated, especially if serving as a decision support tool.

4.1.1 Deep learning dose prediction for MR-Linac

In Paper IV, a two-step approach for prostate MR-Linac treatments was devised, during which segmentations were predicted during the first step, and radiation dose was predicted based on input structures. For this purpose, data from 212 prostate treatments (35 patients) was available, including images, structures, and dose distributions. The images were split into a training and a testing cohort, containing the images from 25 and 10 patients, respectively. Utilizing a 3D U-net we first trained segmentation networks to segment the CTV, bladder, and rectum in a 5-fold cross-validation setting. At inference, an ensemble approach was taken as the mean of the outputs of each network. Then, utilizing the same 3D U-net architecture, we trained networks to predict the dose distribution with structures as input. Also here, a 5-fold cross-validation setting was utilized. For the dose prediction networks, we used the manual (ground truth) segmentations as input and not those predicted from the segmentation networks. Then, the full pipeline consists of predicting the structures from the first networks and using the results as input to the second networks producing the final dose distribution. We also compared this with dose distribution obtained with manual structures as input to be able to determine the influence of the predicted structures. While the variance in the dose distributions with predicted structures as input was slightly higher, the overall mean deviations for DVCs were less than 2% for the CTV (D98%/D95%/D2%), 1% for the bladder (Dmean/D2%) and 1 percentage point for the rectum (V33Gy/V38Gy/V41Gy). In Figure 11, DVCs based on predicted dose distributions are shown, both based on the manually delineated structures and the predicted structures. All results were evaluated on the manually delineated structures.
Figure 11. The difference in dose-volume constraint between the true and predicted dose for CTV, PTV, bladder, and rectum, respectively, evaluated when the input to the dose prediction network was the true structures (dashed lines) and predicted structures (solid lines).
Some representative examples of dose distributions are visualized in Figure 12.

![Figure 12. Dose and structures for two representative cases. The leftmost plot is the true structures and true dose, the middle the predicted structures and predicted dose, and the rightmost true structures with predicted dose.](image)

The conclusion from this work is that combining deep learning-based segmentation and dose prediction modules into a single pipeline could predict differences in DVC >2% for both target and OARs for prostate treatments at the MR-Linac.
5. Real time motion management

Changes in anatomy occur not only between consecutive days but also during each treatment session. Particularly in abdominal treatments the tumor motion may synchronize with breathing patterns. To address this, an additional margin, referred to as the Internal Target Volume (ITV), can be implemented that includes all known tumor positions. Advanced solutions to decrease the ITV may involve irradiation when the tumor is in a specific predefined position, such as during a breath-hold. In traditional external beam radiotherapy, monitoring internal anatomy typically involves tracking a surrogate, like surface scanning [2]. The MR-Linac however enables imaging during the irradiation. Even so, in MR-imaging, a necessary trade-off emerges. Acquiring a high spatial resolution 3D image over the treatment region can require several minutes of scan time. Despite the utilization of more rapid 3D acquisition during MR-Linac beam-on periods [85]–[87], the required update frequency for tracking abdominal motion is in the order of 3 Hz and overall latency maximum 500 ms [88]. This generally limits us to the use of 2D images. Although these images can be interleaved in various planes (e.g., transversal, sagittal, coronal), they may not provide complete coverage of the entire anatomy.

5.1 Statistical motion models

To bridge this gap, one can turn to motion models constructed in an offline (i.e. non real-time) scenario. This typically entails acquiring 3D images at different time points to capture the full range of expected motion. From this data, motion is extracted through e.g. manual or automatic feature tracking, or through image registration-based methods [89]. One approach is to utilize deformable image registration to extract the motion, followed by a principal component analysis of the resulting displacements [90]–[92], which is the approach described here.

5.1.1 Extracting the motion

Motion extraction can be performed with image registration techniques, which is the task of spatially aligning images. These methods come in different levels of complexity. Translational image registration rigidly moves one image to
match the other, which can be described with as few as three parameters in 3D. Other methods, in increasing order of complexity, include rigid registration (translation+rotation), affine registration, and deformable registration. Different algorithms can be employed, depending on the problem at hand, but the result can be described with a displacement vector field (DVF). This is a field describing, on a voxel level, how each voxel moves from one image to the other image. Often, regularization is imposed not to allow implausible displacement fields [93]. For medical imaging, regularization restricting the displacement to anatomically viable values is possible [94], [95]. Achieving an accurate displacement field is paramount if the use case is to construct a motion model built upon this.

5.1.2 Motion model through principal component analysis

Principal component analysis (PCA) is a way of changing the underlying basis of a dataset, in this case the obtained DVF:s from the deformable image registration. Often, original basis configuration is not optimal, hence leading to obfuscated important characteristics and redundant dimensions. PCA decomposes the original data into new basis functions spanned by the principal components (PCs) to minimize the covariance between the components, aiming to reveal the underlying dynamics of the data through finding the most prominent features. It assumes linearity where the PCs are orthogonal to each other, and the original data can then be constructed through a linear combination of these. The PCs are obtained by finding the eigenvectors of the covariance matrix $C$ expressed as

$$C = \frac{1}{N-1} PP^T$$  \hspace{1cm} (7)

where $N$ is the number of measurements (i.e. number of time points in the dataset), superscript $T$ denotes the transpose and for a DVF-based analysis $P$ is a matrix of concatenated and centered DVF:s according to

$$P = [\bar{X}_1, \bar{X}_2, ..., \bar{X}_N]$$  \hspace{1cm} (8)

and

$$\bar{X}_n = X_n - \mu$$  \hspace{1cm} (9)
where $X_n$ is the $n$:th DVF and $\mu$ the centered shape vector, i.e the mean value of all $X_n$:s.

However, this general expression is cumbersome in constructing a PCA DVF-based motion model for any anatomical site since the size of each $X_n$ generally has an overwhelming number of variables (generally 3x the number of voxels since each voxel is regarded a variable), resulting in a very large covariance matrix. This is an unfeasible scenario. However, by noting that if the number of measurements $N$ is significantly lower than the number of variables one can show [96] that one instead can find the PCs by finding the eigenvectors of the implicit covariance matrix $\tilde{C}$

$$\tilde{C} = \frac{1}{N-1} P^T P$$

with size of $N \times N$, where the eigenvectors are the required principal components and their corresponding eigenvalues are a measure of the respective importance. The original DVF:s can then hence be reconstructed through

$$X_n = \mu + \sum_{j=1}^{k} w_j PC_j$$

where $k$ is the number of chosen PCs (not necessarily all) and $w_j$ the scalar weight for the $j$:th PC where the optimal weight is the projection of the centered original data onto the PC. The scalar $w_j$ is the unknown and the one we need to determine when using the model. External surrogates such as tracking a point or a surface can be employed, but for the MR-Linac we can utilize real-time 2D images. We can perform an optimization procedure, in which the optimal set of $w_j$ is found where the warped model anatomy best matches the real-time image.

5.1.3 Non-periodic motion

The drawback of employing a motion model lies in the necessity to align motion patterns during both the offline scenario (data acquisition for the model) and the online scenario (when the model is utilized). While this requirement is typically met for abdominal motion patterns influenced by breathing due to their periodic nature, it may not hold true for other anatomies. For instance, pelvic motion, such as that of the prostate, is generally minimally affected by respiratory motion. Instead, other patterns, like shifts in position due to bladder and rectum movements, play a more significant role. Continuous bladder
filling may induce a gradual drift of the prostate, while rectum motion could equally impact its position. This may not conform to a periodic motion pattern during a treatment fraction.

5.1.4 Non-periodic motion model based on principal component analysis

In Paper I, we addressed the challenge of individual intrafractional non-periodic motion by using a dataset from nine healthy volunteers consecutively scanned with a 3D MR T2-weighted sequence collected over the prostate region. The sequence spanned 10-15 minutes, providing 10 3D time-frames. Each time-frame was manually annotated, segmenting the prostate, bladder, and rectum. We selected the first time-frame in each dataset as the fixed image and deformably registered all remaining frames to this fixed frame, using an optical flow algorithm [97]. A common scenario for model validation is a leave-one-out approach, where a model is built on all but one datapoint and evaluated on this datapoint. However, this approach might not be suitable for non-periodic motions, in which the intended use case is constructing a model during the first part of the treatment and using it in subsequent time points. Validating a model using a leave-one-out approach in cases of non-periodic motion, like continuous drift, would result in time-wise interpolation rather than extrapolation in time, where the latter aligns more with the intended use case. Therefore, we divided each dataset into training data, comprising the first six timeframes, and validation data as the last four time-frames. We constructed motion models through PCA decomposition of the DVFs in the training data for each dataset and validated them on the validation dataset. We observed that the model accuracy plateaued around two principal components, and the residual motion not captured by the model was less than 1 mm. In Figure 13, a plot of the displacement distribution is presented for each structure, comparing the usage of different numbers of principal components against not using a model at all. Here, the model was driven by the DVFs obtained from the image registration of the full 3D volumes on the validation set, and the results could be regarded as the best achievable.
The conclusion from this work is that patient-specific motion models of the prostate anatomy could be used to explain intrafractional motion down to a sub-millimeter level using two principal components.

5.2 Deep learning-based motion models

The motion-model approach addresses the trade-off between latency requirements and spatial resolution by incorporating a model along with a surrogate. However, a challenge with this method arises from the necessity of having the same motion pattern in both the model and during utilization. The real-time surrogate, despite being an anatomical image in the MR-Linac case, is confined to 2D planes, inherently lacking coverage of the entire anatomy. An alternative strategy involves obtaining 3D images of the entire relevant anatomy. In this case, addressing the latency issue is not about restricting the field of view but rather utilizing different means to shorten the acquisition (and reconstruction) time. One such technique is to reduce the spatial resolution, since the image acquisition time decreases with the resolution. However, directly utilizing images with extensively reduced resolution is hindered by the difficulty of distinguishing relevant anatomy. Nonetheless, there are methods attempting to restore resolution, known as super-resolution methods.

5.2.1 Image super-resolution

Image super-resolution (SR) involves artificially increasing the resolution of an image. Traditional methods, such as interpolation or statistical techniques [98], can achieve this, but deep learning-based methods provide an alternative. These methods can be categorized into four groups [99]: pre-upsampling, progressive upsampling, post-upsampling, and iterative up-and-down sampling, each with its advantages and drawbacks. Pre-upsampling involves initial
interpolation of a low-resolution (LR) image, followed by processing through a deep-learning network. Progressive upsampling continuously increases the resolution during the image's passage through the network. In post-upsampling, the LR image is processed through the network and then upsampled to the intended resolution. In iterative up-and-down sampling, the images are alternately up-and-down sampled to better learn the relationship between the high-resolution (HR) and LR images. Pre-upsampling allows dealing with various upsampling rates simultaneously but is slower in training and inference due to the larger initial image size. Regardless of the method used, residual learning is an effective approach [99]. Instead of requiring the entire image as the network output, the network outputs the difference between the input and the requested SR image. The rationale is that the input and requested images are similar, as only the resolution differs, making it redundant to learn to reconstruct the entire image.

Commonly used loss functions for training an SR network include MAE and MSE, aiming to enforce voxel-wise correspondence between the input and output images. While these functions yield a high Peak Signal-to-Noise Ratio (PSNR), they may lack perceptual quality [100]. More advanced loss functions, such as perceptual loss [101] using intermediate features from a pre-trained network, and the use of Generative Adversarial Networks (GANs) [102], have been proposed to address this. GANs involve two competing networks, a generator producing an SR image and a discriminator distinguishing between generated and true images. Although GANs can yield high perceptual quality [103], [104], they may face training instability and introduce non-real attributes.

Acquiring a good set of training data is crucial. In clinical settings, only HR images are generally obtained. Artificial downsampling methods, ranging from simple interpolation to optimization of a blur kernel [105], can be used to create these pairs. GANs offer a potential solution by not requiring image pairs, and zero-shot super-resolution [106], which uses internal recurrences of features in the upsampling, is another approach. However, paired images may still be needed for validation and testing.

As mentioned, achieving rapid 3D MR acquisition is feasible by lowering spatial resolution. To achieve imaging in scales considered real-time, this may necessitate a substantial reduction in spatial resolution. While applying a super-resolution network can restore spatial resolution, it alone is insufficient for decision-making in an MR-Linac workflow. The positions of relevant structures must be determined. This can be done as a separate step or learned by the network through a segmentation step, which is our approach in Paper III.
5.2.2 Simultaneous real-time image segmentation and super-resolution

In Paper III, we utilized clinical data from 36 prostate treatments on the MR-Linac, comprising T2-weighted MR images and segmentations of CTV (prostate), bladder, and rectum. We generated LR input data for training by downsampling MR images through k-space truncation at three different downsampling levels (1/2, 1/4, and 1/8 of the HR image along the phase-encoding directions of the acquisition). This allowed us to assess the impact of the resolution on the segmentation results. For each resolution level, we trained networks to obtain a HR segmentation. Additionally, we added the restoration of the corresponding HR image as an auxiliary task to determine the possible benefit of a multitask approach. We again utilized a 2D variant of the U-net architecture and a train/test split of all images from 25 and 11 patients in each set, respectively. A 5-fold cross-validation setting was employed, training on 20 images from the training dataset and validation on the remaining five. At inference, an ensemble approach was employed taking the mean value of all model’s outputs. We found that resolution had a rather small impact on segmentation results, where DSC values on the test dataset for the lowest (1/8) resolution were 0.85(4)/0.90(5)/0.85(4) for CTV/bladder/rectum while 0.88(3)/0.93(3)/0.86(4) for the HR image.

However, it's essential to note that this conclusion was based solely on artificially downsampled images. This approach may not fully replicate actual MR acquisitions at LR. To address this, we acquired images of ten healthy male volunteers at resolution levels similar to those in the artificially downsampled data. We manually segmented the HR images with prostate, bladder, and rectum. This way, a set of genuinely LR scans was acquired. Since a different MR acquisition setting was used (balanced steady-state free precession), producing a distinct contrast, we fine-tuned the network from the artificially downsampled images to this dataset. Conducting the same analysis as with the artificially downsampled data revealed a clearer performance degradation depending on the resolution level of 0.52(22)/0.70(14)/0.54(20) for CTV/bladder/rectum for the lowest resolution while 0.73(0.12)/0.76(13)/0.79(7) for the HR, as depicted in Figure 14. However, the decline in performance compared to the artificially downsampled dataset can be attributed not solely to the variance in downsampling methods. The alternative acquisition protocol led to reduced contrast, particularly in the delineation of the prostate and rectum against their surroundings, thereby complicating the segmentation process.

Acquisition times for the images of the healthy volunteers ranged from 15 seconds for the HR all the way down to 0.7 seconds for the lowest resolution image. However, inference times of the networks are also an important constituent of the latency if regarded a real-time scenario. In this work, we
cropped the images around the structures of interest, partly to decrease the number of background voxels to alleviate the training process, but also to decrease the inference times. This resulted in inference times of ~100 ms per network prediction on the whole volume, which of course also depends on the hardware applied (in this case an NVidia RTX3090 GPU). The conclusion from this work is that spatial resolution degrades deep learning-based image segmentation accuracy, although the effect could be deemed small when reduced to half the original resolution. The auxiliary task or simultaneously restoring an SR image did not improve the segmentation results.

Figure 14. DSC (upper row) and 95% Hausdorff (bottom row) values for three different structures (CTV, bladder, and rectum) and the different relative resolution levels along the x-axis. For each resolution level, the leftmost bar indicates training solely for segmentation, while the rightmost additionally includes the restoration of the image as an auxiliary task.
6. Summary and outlook

Radiotherapy has seen significant advancements with treatments increasingly tailored to individual patients. Advanced imaging modalities enable precise delineation of targets, while technologies like MLCs and techniques such as VMAT allow for more accurately shaped dose distributions to the target volume. Nevertheless, anatomical movement introduces considerable uncertainty due to both the fractionated nature of radiotherapy treatments and movement during treatment. In-room imaging methods like CBCT and surface scanning can reduce setup errors, but generally, the plan is not adapted to daily changes in anatomy, which might render it less than ideal.

A notable innovation in the field is the integration of an MRI scanner with a linear accelerator, creating the MR-Linac system. This combination offers superior soft-tissue visualization, enabling daily adjustment of treatment plans based on the anatomy of the day and potentially allowing for real-time monitoring and adjustment of the treatment beam to target movements. However, one limitation of this approach is the increased duration of treatment sessions, which can extend significantly longer than traditional treatments, sometimes exceeding an hour per session. Moreover, the nature of MRI technology limits the speed and resolution of three-dimensional imaging necessary for monitoring motion in real-time. This project aimed to explore opportunities within the daily treatment workflow where machine learning could expedite the current process. Given that prostate cancer patients currently constitute the majority of cases treated at our institution, the project predominantly focuses on this group. Nonetheless, the strategies developed could apply to other anatomical sites as well.

In Paper II, we approached the currently most time-consuming aspect of the workflow, namely the re-contouring step. By training patient-specific deep-learning networks to segment the relevant structures on a single image per patient, we arrived at networks performing on par with the clinically used deformable image registration algorithm, although faster.

In Paper IV, we combined a deep learning-based segmentation with a dose prediction network to be able to obtain a rapid overview of the likely dose distribution ahead of the much lengthier re-contouring and re-optimization
procedure required for treatment. This could allow for rapid decision-making on whether to use ATS and recontouring or if ATP is sufficient.

In Paper I and III we approached the real-time aspects of the treatment. In Paper I, we constructed and evaluated PCA-based individualized intrafractional motion models, and found that two principal components could explain motion down to a sub-millimeter level.

In Paper III we applied deep-learning networks to segment low spatial-resolution 3D images, down to resolution levels possible to acquire in real-time. Findings suggest that spatial resolution could be lowered down to at least 1/2 of the initial resolution and hence achieve a speed-up in acquisition time. During this project, we have gained valuable insight into the possibility of improving the workflow on the MR-Linac with machine learning techniques, with the next step to incorporate these into the clinical workflow. Hopefully, these techniques can alleviate the burden on the clinical staff and increase the treatment capacity, such that more patients can benefit from this treatment.

En av de största osäkerheterna är förändringar i anatomi mellan planeringstillfället och mellan behandlingstillfällena. För att hantera detta lägger man på marginaler på behandlingsområdet, och nyttjar tekniker som daglig bildtagning som matchas med planeringsbilderna. Nackdelen är dock att man alltid ger samma behandlingsplan oavsett om anatomin är förändrad.


Det vi har tittat på under detta projekt är möjligheter att med hjälp av maskininlärningsmetoder gå in i olika steg i denna behandlingsprocess, med syfte att göra den både snabbare och säkrare. Maskininlärning, i korthet, handlar om att med hjälp av (ofta stora mängder) data lära upp en maskin (dator) att

I Artikel II tittade vi på djupinlärningsmetoder för att automatiskt segmentera (rita ut) nödvändiga strukturer för prostatebehandlingar på MR-Linac. I Artikel IV tog vi detta ett steg längre och inte bara automatiskt segmenterade strukturerna, men även förutsåg hur behandlingsplanen (fördelningen av stråldos) skulle se ut.

I Artikel I och III tittade vi på en annan aspekt, nämligen hur man kan nytta maskinens möjlighet till bildtagning för att följa anatomin under själva bestrålningen. Detta som ett förste för möjligheten till tracking (att följa tumörens rörelse i realtid och eventuellt anpassa sin behandling till rörelsen). I Artikel I konstruerade vi en rörelsemodell av anatomin för prostatebehandlingar, och i Artikel III undersökte vi möjligheten att snabbt segmentera väldigt lågupplösta bilder (som kan samlas in snabbt).

Sammanfattningsvis har vi under detta projekt visat på potentialen i maskininlärningstekniker för att kunna förbättra arbetsflödet på MR-Linac.
8. Acknowledgement

I would first of all like to thank my main supervisor, Robin, and my co-supervisor, David, for their unwavering involvement and encouragement throughout every phase of this project. Your consistent support was vital for the project's progress. Additionally, I am thankful for the contributions of several co-supervisors, namely Håkan, Anders, and Zarah, whose valuable insights significantly enriched this project. A special mention goes to Tufve, whose guidance was pivotal in my decision to pursue studies in Uppsala.

To my partner, Mikaela. Thanks for the patience and the constant pushing me to finalize. At some point, my frequent answer “soon” would turn out to be accurate. And thanks to the rest of my family, with the same support and encouragement.

Furthermore, I am grateful to all my colleagues at the Medical Physics and Radiotherapy department. It was, and still is, a privilege to be working with such friendly and competent people.
9. References


A doctoral dissertation from the Faculty of Medicine, Uppsala University, is usually a summary of a number of papers. A few copies of the complete dissertation are kept at major Swedish research libraries, while the summary alone is distributed internationally through the series Digital Comprehensive Summaries of Uppsala Dissertations from the Faculty of Medicine. (Prior to January, 2005, the series was published under the title “Comprehensive Summaries of Uppsala Dissertations from the Faculty of Medicine”.)