博士論文　（要約）

Tumor position prediction using dynamically
trained recurrent neural networks for latency
compensation in lung cancer radiotherapy

(肺がん放射線治療における遅延補正のための
回帰型ニューラルネットワークの適応的学習
による腫瘍位置の予測)

ポル ミッシェル

Pohl Michel

Acknowledgments

I would like to express my sincere thanks to Professor Mitsuru Uesaka, who supervised my work for more than four years and gave me the opportunity to work on a very interesting and challenging problem related to artificial intelligence and radiotherapy. I express my sincere gratitude to Professor Hiroyuki Takahashi, who offered guidance and help during the last six months of my thesis. I thank Professor Kazuyuki Demachi and Dr. Ritu Bhusal Chhatkuli, who also accompanied my work and provided advice useful to my research.

I would like to thank Dr. Jeawoong Jang who help me much concerning adapting to my new life in Japan and gave me advice concerning research in general. I also thank Mr. Yoshinori Inoue who helped me settle in Japan and apply for scholarships. I thank Dr. Stephen Wells who proofread two research articles that I submitted to peer-reviewed journals. I thank Dr. Shi Chen, who gave advice related to computer hardware and helped me regarding data acquisition. I thank my former neighbor Mr. Surya Narayanan who also provided help related to MRI data reconstruction and with who I could enjoy cooking and discuss various topics related to computer science, amidst the pandemic.

I would like to express my sincere gratitude to the Epson International Scholarship Foundation, the Graduate School of Engineering of the University of Tokyo, as well as the Japan Student Service Organization, who provided financial support during my study in Japan.

More generally, I thank all the people who contributed to the success of my research at the University of Tokyo and showed me a beautiful image of Japan.

Finally, I thank my parents, who continuously and unconditionally supported me during my studies in Japan.

Contents

List of Figures

List of Tables

List of Algorithms

List of Abbreviations

1D one-dimensional

2D two-dimensional

3D three-dimensional

4D-CBCT four-dimensional cone beam computed tomography

4DCT four-dimensional computed tomography

ADAM adaptive momentum

AI artificial intelligence

AIP average intensity projection

ANFIS adaptive neuro-fuzzy inference system

ANN artificial neural network

AP antero-posterior

ARIMA Auto-Regressive Integrated Moving Average

ART adaptive radiotherapy

BPTT back-propagation through time

CNN convolutional neural network

CT computed tomography

DIR deformable image registration

DTA distance to agreement

DVF displacement vector field

EPID electronic portal imaging device

FCL fully connected layer

FCN fully convolutional network

GAN generative adversarial network

GPU graphics processing unit

GRU gated recurrent unit

GTV gross tumor volume

IGRT image-guided radiotherapy

IMRT intensity modulated radiation therapy

ITV internal target volume

kV kilovoltage

LINAC linear accelerator

LMS least mean squares

LR left-right

LSTM Long Short-Term Memory

MAE mean average error

MLC multileaf collimator

MLP multilayer perceptron

MR magnetic resonance

MRI magnetic resonance imaging

nRMSE normalized root-mean-square error

OAR organ at risk

OSTL online spatio-temporal learning

PCA principal component analysis

PET positron emission tomography

PTV planning target volume

QA quality assurance

RMS root-mean-square

RMSE root-mean-square error

RNN recurrent neural network

ROI region of interest

RPM Real-Time Position Management

RTRL real-time recurrent learning

SHL signal history length

SI superior-inferior

SnAp-1 sparse 1-step approximation

SNR signal to noise ratio

SVM support vector machine

UORO unbiased online recurrent optimization

List of Symbols

$I(\vec{x}, t)$ intensity of pixel/voxel $\vec{x}$ at time t

L signal history length

L_n RNN loss at time t_n

$W_{a,n}$ recurrent weight matrix of the RNN at time step n

$W_{b,n}$ input weight matrix of the RNN at time step n

$W_{c,n}$ output weight matrix of the RNN at time step n

Φ non-linear activation function of the RNN

η learning rate

σ_w standard deviation of the Gaussian kernel used in Nadaraya-Watson regression for image warping

σ_{LK} standard deviation of the Gaussian kernel used in the Lucas-Kanade optical flow

σ_{init} standard deviation of the Gaussian distribution of the RNN initial weights

σ_{init}^{DVF} standard deviation of the Gaussian low-pass filter initially applied to images before computing the displacement vector field

σ_{sub} standard deviation of the anti-aliasing Gaussian low-pass filter used at each layer to create the image pyramidal representation in the Lucas-Kanade optical flow algorithm

θ_n RNN parameter vector (synaptic weights) at time t_n

e_n instantaneous error vector

e_{DVF} root-mean-square registration error

h prediction horizon

h_w window size of the kernel used in Nadaraya-Watson regression for image warping

m size of the input layer of the RNN (without bias unit)

n_{iter} number of iterations when refining the deformation field at a given layer in the Lucas-Kanade optical flow algorithm

n_{layers} number of layers of the image pyramidal representation used in the Lucas-Kanade optical flow algorithm

p size of the output layer of the RNN

q number of hidden units of the RNN

u_n RNN input vector at time t_n

y_n RNN output vector at time t_{n-1}

y_n^* ground-truth signal (correct output) at time t_{n-1}

Chapter 1

Introduction

1.1 Overview of artificial intelligence applied to radiation therapy

1.1.1 General introduction

Lung cancer is the second most prevalent form of cancer worldwide with 2.2 million new cases in 2020, representing 11.4% of new total cancer cases Sung et al. [2020]. The most prevalent cancer type is female breast cancer, with 2.3 million new cases in 2020. Lung cancer is the leading cause of cancer death, with approximately 1.8 million deaths worldwide (18% of all cancer deaths), followed by colorectal and liver cancer, which represent respectively 9.4% and 8.3% of all cancer deaths. The survival rate of patients with lung cancer at 5 years after diagnosis is only 10% to 20% in most countries.

More than 50% of cancer patients are treated with radiotherapy. The radiotherapy process involves several steps: patient assessment, image acquisition, treatment planning, radiation delivery, and post-treatment follow-up. The data used in radiation oncology takes various forms, such as four-dimensional computed tomography (4DCT), four-dimensional cone beam computed tomography (4D-CBCT), magnetic resonance imaging (MRI), fluoroscopic imaging, positron emission tomography (PET) imaging, genomics data, and dosimetry data. In this context, artificial intelligence (AI) applications to radiotherapy are being developed to ensure better patient outcome (cf the literature review articles [Feng et al., 2018, Huynh et al., 2020, Jarrett et al., 2019, Meyer et al., 2018]). These need to tackle the challenges posed by complex treatment workflow and multimodal data described above.

The application of AI to radiotherapy also faces many difficulties that are shared with medical imaging in general. Publicly available datasets are limited, which prevents reproducibility of results and advances in the field. In addition, most datasets comprise few training examples, which hampers the performance of prediction al-

gorithms. Therefore, one of the current research directions focuses on leveraging the data available in the best possible way, for instance by using transfer learning. Real-time performance, as well as high reliability, are critical in image-guided robotic radiosurgery. Furthermore, AI faces lack of trust by medical practitioners. Indeed, in computer-aided diagnosis, a certain degree of explainability and interpretability of the results is highly desirable.

1.1.2   Artificial intelligence within the radiotherapy workflow

AI can help physicians appropriately decide the best treatment for each patient using a data-centric and quantitative approach. Indeed, software can provide information about the benefit to risk balance associated with different potential treatment plans (e.g., intensity modulated radiation therapy, 3D conformal radiotherapy, proton therapy, etc.), making selection of the best option for each individual patient easier. For example, recurrent marginal networks have successfully been applied to the forecast of the tumor volume over time depending on the treatment selected [Lim et al., 2018]. Computer-aided diagnosis and detection from medical images are also extremely valuable. For example, identification of nodal metastasis and tumor extranodal extension (ENE) from pre-treatment radiographic images, despite being a difficult task for clinicians, can be performed with convolutional neural networks (CNN) with high accuracy [Kann et al., 2018].

Image processing during radiotherapy planning can help reduce artifacts due to the presence of metal parts in the human body (dental fillings, spinal implants, hip prostheses, etc.) for better dose estimation using sinogram completion methods and model-based iterative algorithms [Giantsoudi et al., 2017]. Artificial neural network (ANN) architectures such as U-Nets [Han, 2017], fully convolutional networks (FCN) [Nie et al., 2016], and generative adversarial networks (GAN) [Wolterink et al., 2017] can help generate synthetic computed tomography (CT) images with electronic tissue density information necessary for treatment planning from magnetic resonance (MR) images, in which tumors are more easily distinguishable due to a higher visual contrast [Hsu et al., 2013]. AI is also used in reconstruction algorithms to obtain higher quality images with low-dose X-ray images, therefore reducing the risks of overexposure to radiation. In other words, AI developments in image reconstruction help improve the trade-off between image quality and dose exposure [Willemink and Noël, 2019].

Automatic segmentation of organs at risk (OAR) and target volume on the planning images can help physicians save time. It can also help mitigate the high degree of inter-observer variability as well as the absence of an absolute ground truth for contouring, which are some of the most important sources of uncertainty
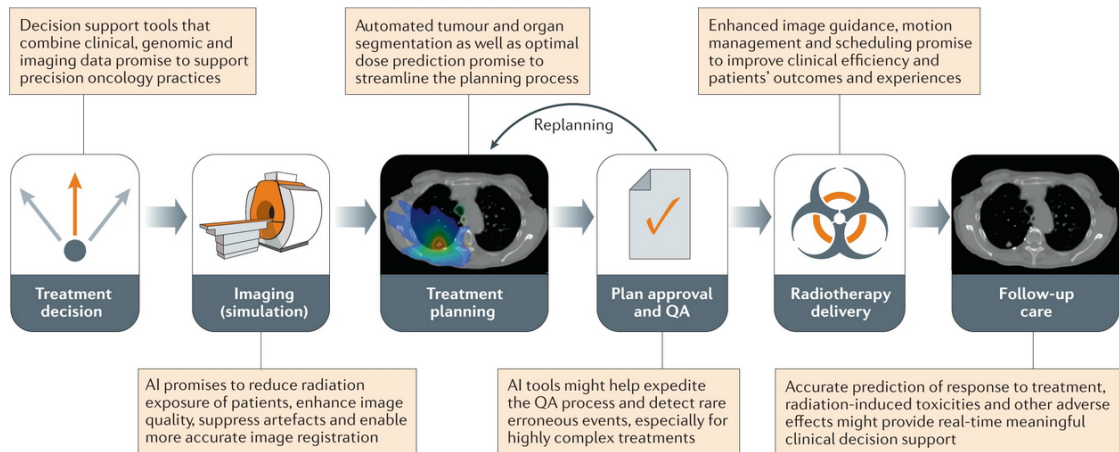
Figure 1.1: Applications of AI in the radiation therapy workflow[1]

in treatment planning [Roques, 2014]. A large-scale example of such work is the collaboration between DeepMind and the University College London Hospitals on the automatic segmentation of 21 OARs of the head and neck anatomy using a three-dimensional (3D) U-Net trained with 663 CT scans, which achieved human-level performance on 19 of the OARs studied [Nikolov et al., 2018].

After delineation, machine learning algorithms can be used to estimate dose distribution [Barragán-Montero et al., 2019, Shiraishi and Moore, 2016] and assess dosimetric trade-offs [Valdes et al., 2017]. Identification of the optimal parameters of the treatment system to achieve the best dose distribution is another area of research in AI for treatment planning. In adaptive radiotherapy (ART), the treatment plan is modified using a system involving feedback measurements obtained during the treatment. Deep reinforcement learning has recently been applied to ART by assessing the need to perform re-planning in the case of non-small-cell lung cancer [Tseng et al., 2017]. Machine learning models such as support vector machines (SVM) or random forests can also benefit quality assurance (QA) by helping detect anomalies with linear accelerator (LINAC) imaging systems [Valdes et al., 2015] and discrepancies between planned and recorded multi-leaf collimator movements [Carlson et al., 2016].

Positioning the beam precisely during the treatment is important for safety as high-dose irradiation should not be delivered to critical areas. For instance, in the case of head and neck cancer, guiding incorrectly the radiation beam can result in damage to the optic nerve (optic neuropathy). This can lead to visual loss in one or both eyes, visual field defects, or defective color vision [Mihalcea and Arnold, 2008]. Improvements in image registration algorithms contribute to precise patient positioning before treatment and help in motion management during the treatment [Brock et al., 2017]. Indeed, the recent addition of new imaging modalities such as MRI or optical imaging to treatment systems provides valuable information that

---

[1]Reprinted from [Huynh et al., 2020] with permission from Springer Nature, Copyright 2020.

can be used to make radiation delivery more accurate [Ma et al., 2018].

AI software could also help as a clinical support tool that allows better characterizing tumor response to the treatment. As an example, SVMs have successively been applied to detect early signs of lung cancer recurrence after stereotactic surgery [Mattonen et al., 2016].

1.2 Time series forecasting for motion management in lung radiation therapy

1.2.1 Clinical background

In lung cancer therapy, a certain amount of normal tissue surrounding the tumor receives irradiation. This is due to normal movements of the organs causing tumor displacements, such as breathing in the case of lung cancer. Lung tumors exhibit a rather cyclic motion, with some changes in frequency and amplitude over time. It has previously been reported that such motion can be up to 5cm [Chen et al., 2001]. Phase shift, as well as intrafractional baseline shift and drift, can be observed. The term "shift" refers to sudden changes in the mean tumor position whereas the term "drift" refers to continuous changes, during a single treatment. Baseline drifts of 1.65 ± 5.95 mm (mean position $\pm$ standard deviation), 1.50 ± 2.54 mm, and 0.45 ± 2.23 mm have respectively been reported concerning the spine axis, dorsoventral axis, and left-right direction in [Takao et al., 2016]. Patients tend to relax while lying on the treatment couch may and also slightly change their overall position from time to time, which contributes to the variability of the respiratory records. Sudden changes or irregular patterns may arise from yawning, hiccup, or cough. Noise is naturally present and can be partly caused by cardiac or gastrointestinal movements. The tumor shape is not rigid and deforms over time to a certain extent (Fig. 1.2). Moreover, the motion of lung tumors can vary across patients and fractions [Ehrhardt et al., 2013, Verma et al., 2010].

He et al. provided several examples illustrating the variability of the respiratory motion for a single individual. First, they compared two 4DCT sequences from the same subject, acquired at different times. Deformable image registration (DIR) was performed to estimate local chest tissue motion due to breathing. They found that the average magnitude of deformation both on the chest surface and lung field differed between the two imaging sessions (Fig. 1.3).

To further illustrate inter-fractional variability, they registered chest images of a single patient at different treatment days. The result of these calculations is shown in Fig. 1.4. The large discrepancies at the armpit are caused by the difficulty of performing prior positioning. There was a significant difference between the calculated displacements of the recorded chest surface during the different sessions,
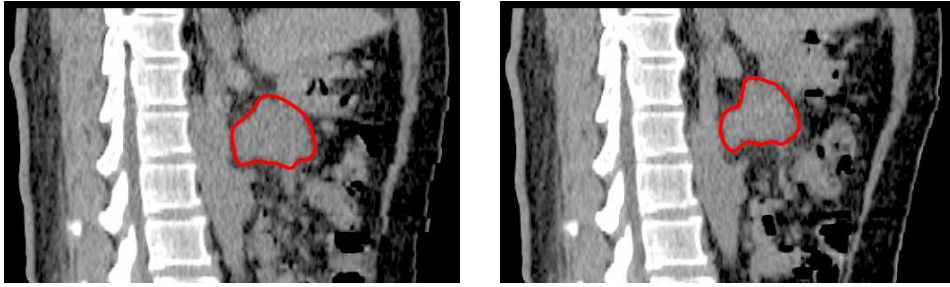
Figure 1.2: Pancreatic tumor at the end of inspiration phase (left) and end of expiration phase (right) in a series of CT scan images. The center of mass of the tumor moves up and down due to the breathing motion and the tumor shape also changes following that motion.[2]
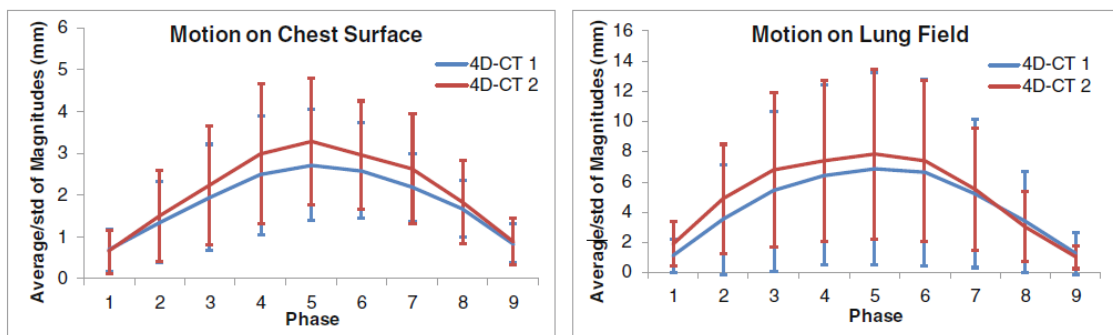
Figure 1.3: Magnitude of deformation from the baseline (phase 0) to the other respiratory phases in two 4DCT acquisitions of the same subject.[3]

of up to 4.5mm. Respiratory motion variability and safety requirements motivate the development of breathing motion management techniques in radiotherapy.

Different targets are defined by clinicians to take into account tumor position uncertainties during the treatment. In conventional motion encompassing methods, the internal target volume (ITV) takes into consideration internal physiological motion such as lung motion (Fig. 1.5). The final target, called planning target volume (PTV), also encompasses intra-fraction and inter-fraction setup errors influencing the dose delivery [Chhatkuli, 2016] (Fig. 1.6). PTV margins can be as large as 0.9cm, 1.0cm and 2.7cm in the left-right (LR), antero-posterior (AP) and superior-inferior (SI) directions, respectively [Li et al., 2016].

Irradiating a large area around the tumor results in high irradiation to surrounding healthy tissues, leading to potential undesirable side effects. Inflammation of the cells that line the alveoli, also called radiation pneumonitis, makes oxygen transfer less efficient and is characterized by symptoms such as cough, fever, and fullness of the chest. In pulmonary fibrosis, stiff scar tissue develops in the lungs, which reduces lung elasticity and therefore the amount of air one can breathe in. Some rare side

---

[2]Figure used in [Chhatkuli, 2016] (open access doctoral dissertation)
[3]Reprinted from [He et al., 2017] with permission from Springer Nature, Copyright 2017.
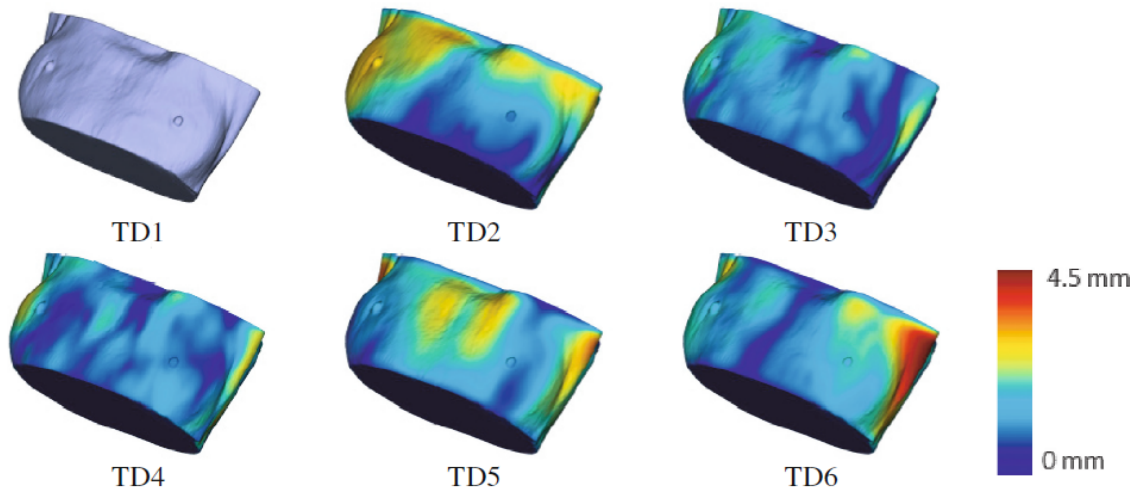
Figure 1.4: Magnitude of deformation of the chest surface of one patient under free breathing conditions, calculated from 4D-CBCT scan images acquired on different treatment days ("TD" here stands for "treatment day").[4]
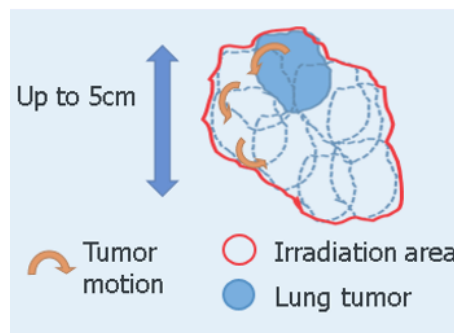
Figure 1.5: Large irradiated area in conventional lung radiotherapy [5]
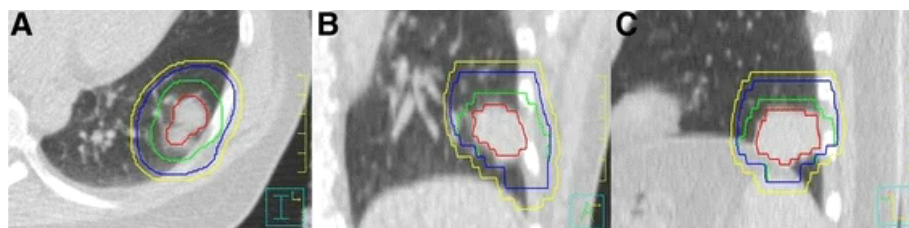
Figure 1.6: Visualization of the different target volumes in the case of a lung cancer patient, in the transverse (a), sagittal (b) and coronal (c) planes of a 4DCT image. The ITV is represented in blue and the PTV in yellow. The GTV is delineated in red. [6]

effects of lung radiation therapy include spinal cord damage, which causes walking difficulties and loss of sensation in the lower body, inflammation of the lining surrounding the heart (pericarditis), and weakening of the heart muscle (myopathy).

---

[4]Reprinted from [He et al., 2017] with permission from Springer Nature, Copyright 2017.
[5]Figure used in [Chhatkuli, 2016] (open access doctoral dissertation)
[6]Figure used in [Li et al., 2016] (open access article)

In rare cases, damage to the blood vessels supplying the heart may increase the risk of a heart attack [CancerConnect, 2020, N., 2015].

To decrease the amount of irradiation to healthy tissues, several options have been proposed. The first one is breath-holding and consists of asking the patient not to move for a certain duration. This causes much discomfort to the patient. Beam-gating is another method that consists of delivering radiation to the tumor only when it reaches a certain location by alternatively turning the beam on and off [Jiang, 2006], thereby making the treatment accurate but lengthy. Recent advances in robotics and sensor technology should however allow for continuous accurate tracking and irradiation of the tumor as it moves. Indeed, the latter could be directly tracked in real-time using kilovoltage (kV) or MR imaging [Bertholet et al., 2019]. In addition, indirect models can estimate the tumor location from chest optical imaging or the position of internal or external markers.

Direct real-time lung tumor tracking has been performed based on optical flow registration [Xu et al., 2008], Principal component analysis (PCA) followed by regression [Lin et al., 2009], dynamic decomposition of image intensities [Homma et al., 2013], histogram-based target modeling [Zhang et al., 2014], particle filtering [Bourque et al., 2016], and extended Kalman filters [Shieh et al., 2017]. Some recent tracking approaches focus on ANNs [Hirai et al., 2019, Terunuma et al., 2018, Yun et al., 2015].
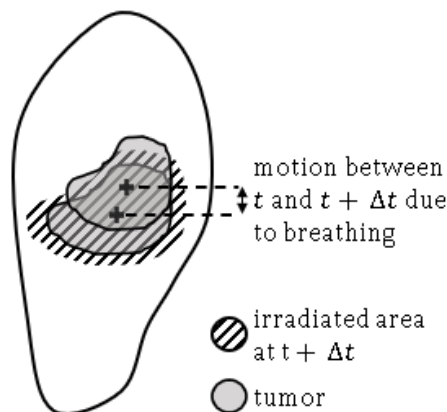
Figure 1.7: Excessive irradiation of healthy lung tissue due to an overall system delay Δt not compensated. The area irradiated, represented here using diagonal stripes, is larger than the tumor size, to take into consideration effects such as variation of the tumor shape during the treatment.

Current treatment systems generally suffer from an inherent time latency due to image acquisition and processing, communication delays, and preparation of the radiation delivery system. A latency of around 300ms has been reported for a robotic arm-mounted LINAC in [Schweikard et al., 2000]. Concerning gantry-mounted multileaf collimator (MLC) based LINACs, Shirato et al. reported a latency of 90ms [Shirato et al., 2000], whereas Poulsen et al. mentioned time lags from 350ms to

1,400ms for sampling intervals between 150ms and 1,000ms [Poulsen et al., 2010]. Verma et al. summarise the situation as follows: "For most radiation treatments, the latency will be more than 100ms, and can be up to two seconds" [Verma et al., 2010]. If this phenomenon is not compensated, it may lead to errors in the estimation of the tumor position, and thus to serious damage to healthy tissue and ineffective irradiation of the tumor (Fig. 1.7).

1.2.2 Recent advances in Artificial Neural Networks for time series forecasting

Time-series forecasting is a self-supervised task, which means that there is no need for human data labeling before training [LeCun and Misra, 2021]. Deterministic forecasting, which provides the most likely value of the estimated variable, differs from probabilistic forecasting, which provides a probability distribution for that variable, thereby estimating uncertainties numerically. Single-step prediction consists of providing the estimated value corresponding to a certain look-ahead time in the future, whereas in multi-step prediction, one estimates at once several values corresponding to multiple look-ahead times.

Classical predictive models such as Auto-Regressive Integrated Moving Average (ARIMA) [Box et al., 1994] or exponential smoothing [Hyndman et al., 2008] have become less used nowadays as ANNs perform well at extracting complex patterns and enable building global models using multiple related temporal datasets, as opposed to patient-specific models.

One of the current research directions concerning time-series forecasting is the elaboration of more sophisticated neural network architectures based on transformer networks (e.g., temporal fusion transformers [Lim et al., 2019], convolutional transformers [Li et al., 2019]), GANs (e.g., adversarial space transformers [Wu et al., 2020]), graph networks (e.g., spectral temporal graph network [Cao et al., 2021]), and convolutional and recurrent neural networks (e.g., temporal convolutional networks [Bai et al., 2018] [Lässig, 2020], DeepAR [Salinas et al., 2020], quasi-recurrent neural networks [Bradbury et al., 2016], dilated RNNs [Chang et al., 2017]). Some new approaches establish links between deep learning and classical methods such as linear Gaussian state spaces [de Bézenac et al., 2020]. Other recent works tackle specific problems arising in time series prediction such as the sharp prediction of sudden changes [Guen and Thome, 2019] [Guen and Thome, 2020] [Guen and Thome, 2021], prediction with missing data [Cui et al., 2020], prediction with logical assumptions on signal properties such as imposing a reasonable range of values [Ma et al., 2020], or interpretability of the predictions [Barić et al., 2021].

Time-series forecasting is partly motivated by practical problems arising in various areas such as finance (e.g., stock price, index, commodity price, volatility, foreign

exchange rate, and cryptocurrency price forecasting) [Sezer et al., 2020], the energy industry [Wei et al., 2019] [Wang et al., 2019], and the transportation industry (e.g., road traffic flow, road traffic speed, taxi demand, train passenger flow) [Jiang and Luo, 2021]. It has been applied in healthcare for epidemiology [Chimmula and Zhang, 2020] and diabetes management [Zhu et al., 2020]. More generally, time series analysis has other useful applications in healthcare such as cardiovascular disease detection from ECG signals [Faust et al., 2018].

1.2.3 Artificial Neural Networks for time series forecasting in external beam therapy

This section specifically reviews the application of ANNs to time series forecasting for motion management in radiotherapy. Other algorithms have also been used to predict respiratory motion (e.g., kernel density estimation [Ruan, 2010], relevance vector machines [Dürichen et al., 2013] [Fan et al., 2020], random convolution nodes [Wang et al., 2020]), and comparison of these with neural network approaches was performed in several works such as [Verma et al., 2010] [Lee and Motai, 2014, Chapter 2] [Ehrhardt et al., 2013, Chapter 12]. In some studies, it was observed that ANNs are stronger than classical methods at performing prediction with high response time values, superior to 500ms.

Comparison between the performance of different algorithms is difficult because the datasets generally differ from article to article. In particular, the sampling rate, the amplitude of the signals, and selection of the response time are different, although a sampling frequency of 30Hz is common. The prediction errors reported also differ between the studies. They generally comprise the mean average error (MAE), root-mean-square error (RMSE), normalized root-mean-square error (nRMSE), and maximum error.

The first studies about prediction in radiotherapy mainly involved ANNs with only one hidden layer, but deeper architectures and recurrent connections are common in recent works. The latter also tend to use more training and testing data (i.e., more time-series records with a higher duration) to improve performance. The study from Lin et al. in 2019 involves the largest largest amount of data in a study about time-series forecasting for motion management. That data comprises 1703 respiratory traces from 985 patients, and was acquired with the Real-Time Position Management (RPM) system in three clinical institutions [Lin et al., 2019].

The previous studies have discussed topics such as parameter selection (e.g., [Murphy and Pokhrel, 2009]), architecture selection (e.g., [Mafi and Moghadam, 2020]), training method selection (e.g., [Goodband et al., 2008]), the influence of the sampling frequency and horizon (e.g., [Sharp et al., 2004]), and the efficiency of dynamic learning (e.g., [Mafi and Moghadam, 2020]). Some works focused on the

Work	Network	Training method	Breathing data	Sampling rate	Amount of data	Signal amplitude	Response time	Prediction error
[Sharp et al., 2004]	1-layer MLP	Conjugate gradient (offline)	1 implanted marker	1 Hz to 30 Hz	14 records 48s to 352s per record	9.1mm to 31.6mm	1) 33ms 2) 200ms 3) 1.0s	RMSE ∈ 1) [1.8mm, 4.2mm] 2) [2.5mm, 4.9mm] 3) [4.3mm, 6.0mm]
[Isaksson et al., 2005]	1-layer MLP	Levenberg-Marquardt & adaptive training	Internal and external markers	10 Hz	3 patients 90s to 160s per patient	-	200ms to 800ms	nRMSE from 0.22 to 0.65 (external marker)
[Kakar et al., 2005]	5-layer ANFIS	Combination of grad. descent and LMS	1 external marker (RPM data)	25 Hz	11 patients	Greater than 6.95mm	-	RMSE 1.8mm
[Murphy and Dieterich, 2006]	1-layer MLP	Back propagation and LMS	1 external marker (Cyberknife Sync.)	30 Hz	9 patients 45min to 105min	1mm to 2mm	1) 100ms 2) 300ms	1) nRMSE ≤ 0.40 2) nRMSE ≤ 0.86
[Goodband et al., 2008]	1-layer MLP	1) Conjugate gradient (CG) 2) CG + Bayesian regularization	1 external marker a) average filter b) unfiltered	30 Hz	331 records of 4 min (24 patients)	8mm to 60mm	400ms	1a) RMSE 1.092mm 1b) RMSE 1.202mm 2a) RMSE 0.970mm 2b) RMSE 1.341mm
[Murphy and Pokhrel, 2009]	1-layer MLP	Back propagation and LMS	Cyberknife Synchrony data	30 Hz	27 records of 167s	-	1) 100ms 2) 500ms	1) nRMSE ≤ 0.178 2) nRMSE ≤ 0.889
[Lee et al., 2011]	1-layer RNN	Hybrid extended Kalman filter	Cyberknife data	26 Hz	15 records 50min to 120min	Normalized to [−1, 1]	1) 192ms 2) 500ms	1) nRMSE ≤ 0.146 2) nRMSE ≤ 0.192
[Krauss et al., 2011]	1-layer MLP	Gradient-based optimization with L_2 regularization	Tumor position from implanted markers	30 Hz 15 Hz 7.5 Hz	12 records of 83s	6mm to 14mm (SI direction)	600ms	RMSE 1.78mm averaged over the sampling rates
[Lee and Motai, 2014]	RNN -FCL	Customized pred. with multiple patient interactions	Cyberknife data	26 Hz	130 patients records from 25min to 132min	Normalized to [−1, 1]	1) 192ms 2) 500ms	1) nRMSE ≤ 0.177 2) nRMSE ≤ 0.314
[Choi et al., 2014]	3-layer MLP	Back propagation & adaptive training	RPM data (Varian)	10 Hz	87 records 3.6 to 8min	-	100ms 500ms	RMSE 0.041 RMSE 0.147

[Sun et al., 2017]	1-layer MLP	Levenberg-Marquardt & adaptive boosting	RPM data (Varian)	30 Hz	138 records of 330s	Normalized to $[-1, 1]$	500ms	RMSE 0.17 nRMSE 0.28
[Kai et al., 2018]	1-layer RNN	BPTT	1 implanted marker	30 Hz	7 records of 40s to 70s	-	1.0s	RMSE from 0.48mm to 1.37mm
[Wang et al., 2018]	7-layer bi-LSTM	BPTT	1D trace from external markers (Cyberknife)	26 Hz	306 records 25 to 132 min per record	-	1.0s	MAE 0.075mm RMSE 0.097mm nRMSE 0.081
[Teo et al., 2018]	1-layer MLP	Back propagation & adaptive training	Cyberknife Synchrony data	7.5 Hz	27 records of 1 min	2mm to 16mm (development set)	650ms	MAE 0.65mm RMSE 0.95mm Max error 3.94mm
[Yun et al., 2019]	2-layer LSTM & one FCL	BPTT with ADAM & adaptive training	3D center of mass of the tumor	25 Hz	158 records of 8 min	0.6mm to 51.2mm	280ms	RMSE 0.9mm
[Lin et al., 2019]	3-layer LSTM	BPTT with ADAM	RPM data (Varian)	30 Hz	1703 records of 2 to 5 min	Normalized to $[-1, 1]$	280ms 500ms	MAE 0.112 RMSE 0.139
[Mafi and Moghadam, 2020]	3-layer ANN (1 rec. layer)	RTRL	Cyberknife Synchrony	7.5 Hz	43 records of 2.2s to 6.4s	-	665ms	MAE 0.54mm RMSE 0.57mm
[Yu et al., 2020]	3-layer bi-GRU	BPTT with ADAM & online training	1D PCA signal from 3 ext. markers (accuTrack 250)	17 Hz	6 patients 15-20 min per patient	Normalized to $[0, 1]$	600ms	MAE < 0.16mm RMSE < 0.17mm Max error < 0.28mm
[Jöhl et al., 2020]	3-layer MLP	Levenberg-Marquardt	Cyberknife data SNR: 1) 30dB 2) 20dB	25 Hz	95 records 11min to 131min	Up to 12mm	160ms	nRMSE 0.38 nRMSE 0.66
[Lee et al., 2021]	1-layer LSTM & one FCL	BPTT	RPM data (Varian)	30 Hz	550 records 91s to 188s	11.9mm to 25.9mm	210ms	RMSE 0.28mm
[Wang et al., 2021]	2-layer LSTM & 2 FCLs	BPTT with ADAM mini-batch, dropout	External markers (accuTrack 250)	20 Hz	7 records 5min to 6min	-	450ms	RMSE < 0.5mm

Table 1.1: Previous ANN models proposed for prediction in radiotherapy. The term "RNN" designates a vanilla RNN, as opposed to LSTMs and GRUs. A field with " - " indicates that the information is not available in the corresponding research article. Abbreviations defined at the beginning of the thesis document.

combined use of surrogate signal prediction and correspondence models [Isaksson et al., 2005, Wang et al., 2021]. Dosimetric assessment has been conducted in [Lee et al., 2021] to understand more thoroughly how tumor position forecasting can positively impact dose delivery accuracy.

Adaptive or dynamic learning has been applied several times to radiotherapy, and some studies demonstrated the merits of that approach in comparison with static models [Krauss et al., 2011, Mafi and Moghadam, 2020, Teo et al., 2018]. Online learning is beneficial to prediction in radiotherapy because it helps the prediction system conform to changing respiratory patterns (Section 1.2.1). The introduction of online learning in radiotherapy may also have arisen from difficulties in accessing large amounts of data, which are due to regulations about patient personal information protection. In this context, adaptive learning can be regarded as a form of transfer learning that helps compensate for the inability of small training sets to capture all the variations in the test set.

Most of the literature about respiratory motion prediction has been focusing on single-step deterministic prediction of univariate signals. However, because the motion of the tumor's center of mass is three-dimensional, multivariate prediction is needed to reduce uncertainties in target localization. Probabilistic prediction could also help better conform the beam shape by taking into account target location uncertainties.

Most of the recent studies about time series forecasting for latency compensation in radiotherapy has been analyzing the capabilities recurrent neural networks (RNN) and their variants such as Long Short-Term Memory (LSTM) networks and gated recurrent units (GRU). These have also been used in related medical data processing problems such as cardiorespiratory motion prediction from X-ray angiography sequences [Azizmohammadi et al., 2019]. An RNN is a specific type of ANN characterized by a feedback loop that acts as a memory and allows retaining information over time. RNNs can efficiently learn features and long-term dependencies from sequential and time-series data. They have been applied to various problems in natural language processing (e.g., machine translation [Cho et al., 2014, Sutskever et al., 2014]), audio processing (e.g., speech recognition [Graves et al., 2013]), and computer vision (e.g., handwriting recognition [Graves and Schmidhuber, 2009, Graves et al., 2008] and scene labeling [Byeon et al., 2015]).

1.2.4 Impact of tumor position forecasting on dose delivery

Prediction of the lung tumor position is performed to reduce the dose delivered to healthy tissue during the radiotherapy treatment. It has experimentally been observed that forecasting respiratory signals with an RNN could improve radiation delivery accuracy [Lee et al., 2021]. In that study, the surface of a motion phantom was tracked by four external markers on the chest. An iron sphere was inserted
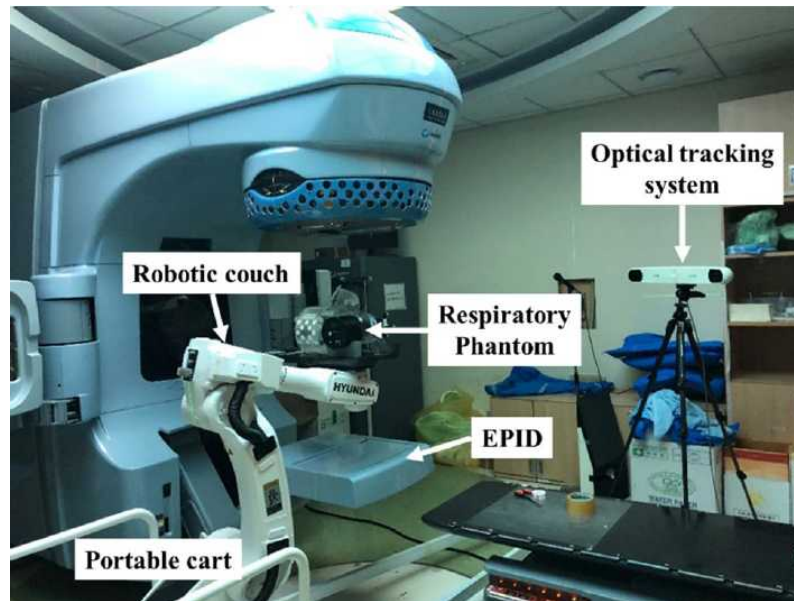
Figure 1.8: Experimental setup involving an articulated robotic couch system in [Lee et al., 2021]. "EPID" stands for "electronic portal imaging device". [7]

inside the phantom and a robotic treatment couch was used to compensate for the one-dimensional (1D) respiratory motion, so that the ball does not move in the beam axis (Fig. 1.8). Prediction was performed using a network composed of LSTM layer followed by a fully connected layer (FCL). Motion traces of 550 patients lasting from 91s to 488s sampled at 30Hz acquired with the RPM system (Varian) were used for training. Prediction was performed with a horizon of 210ms and resulted in an RMSE of 0.28mm over a test set comprising of the motion of 6 patients with an average amplitude of 17.8mm.

The two-dimensional (2D) distribution of the delivered dose was measured using a radiochromic film inserted into the respiratory phantom. Dosimetry analysis was conducted by calculating gamma index distributions to compare the planned and delivered dose [Low and Dempsey, 2003, Low et al., 1998, Radiofísica con Valdo (Youtube), 2019]. Gamma index measures are relative to arbitrary tolerance criteria: a spatial criterion called distance to agreement (DTA), expressed in millimeters, and the dose difference criterion, expressed in percentage. Given a certain location $\vec{x}$, a gamma index $\gamma(\vec{x})$ lower than 1 indicates high local correlation between the calculated and measured dose. The gamma passing rate is defined as the percentage of points $\vec{x}$ for which $\gamma(\vec{x}) < 1$. Concerning the the 2%/2mm criterion, the gamma passing rate associated with the RNN and averaged over the 6 patients, equal to 89.0%, was higher than that associated with double exponential smoothing and no prediction, respectively equal to 82.7% and 73.0%. Similarly, the gamma passing rate corresponding to the 3%/3mm criterion was higher for the RNN (97.4%) than double exponential smoothing (94.2%) and no prediction (87.8%).

---

[7]Reprinted from [Lee et al., 2021] with permission from Springer, Copyright 2021
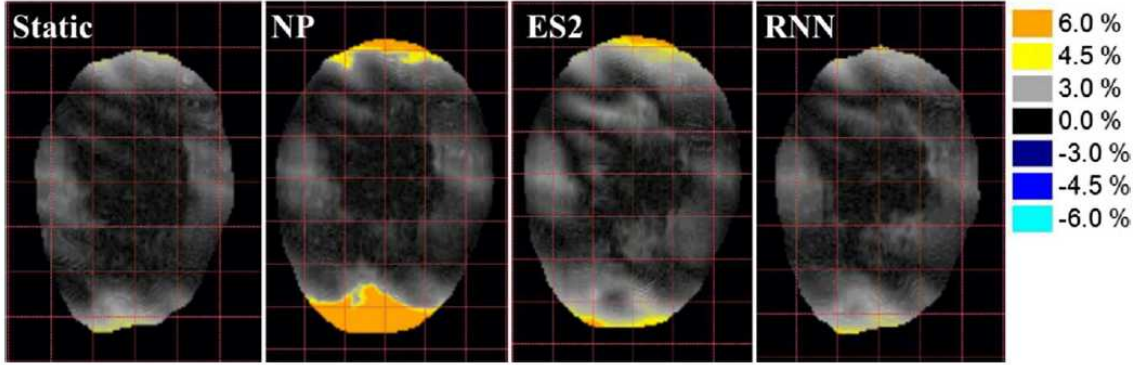
Figure 1.9: Dose difference distribution under the 3%/3 mm criteria for the dosimetric evaluation of Patient 3 in [Lee et al., 2021]. "NP" and "ES2" respectively refer to "no prediction" and the "double exponential smoothing" algorithm. [8]

The study of Lee et al. shows that latency compensation results in better conformity of the dose delivered with the planned dose distribution. The quantitative physical evaluation of the benefits of respiratory motion prediction regarding the delivered dose is not the goal of this thesis. The results above are rather mentioned to highlight the importance and positive impact of algorithmic developments in time-series data forecasting on external beam radiation therapy.

1.3 Online training of Recurrent Neural Networks

The design of learning algorithms for RNNs is one of the main areas of research in machine learning (see for instance [Ke et al., 2018, Liao et al., 2018, Salehinejad et al., 2017, Tallec, 2019]). A good introduction to RNNs and classical algorithms for training RNNs can be found in [Tallec, 2019, Chapter 2]. In this section, we introduce briefly the standard RNN model, also called vanilla RNN, and we review the latest dynamic methods for training RNNs.

The behavior of RNNs is characterized by the state equation, which describes the dynamics of the internal states, and the measurement equation, which describes how the RNN output is influenced by the hidden states (Eq. 1.1)[9]. In these equations, $u_n \in \mathbb{R}^{m+1}$, $x_n \in \mathbb{R}^q$, and $y_n \in \mathbb{R}^p$ respectively represent the input, state, and output vector at time t_n. The synaptic weight vector is denoted θ. Fig. 1.10 gives a graphical representation of these two equations. A one-hidden-layer RNN computes q internal states $x_{n+1}^1, ..., x_{n+1}^q$ (scalar values) from the input u_n and the internal states $x_n^1, ..., x_n^q$. The RNN output layer computes the output vector y_{n+1} from the internal states $x_{n+1}^1, ..., x_{n+1}^q$.

$$x_{n+1} = F_{st}(x_n, u_n, \theta) \qquad y_n = F_{out}(x_n, \theta) \tag{1.1}$$

---

[8] Reprinted from [Lee et al., 2021] with permission from Springer, Copyright 2021

[9] The general formulation of the measurement equation is $y_n = F_{out}(x_n, u_n, \theta)$ but we do not
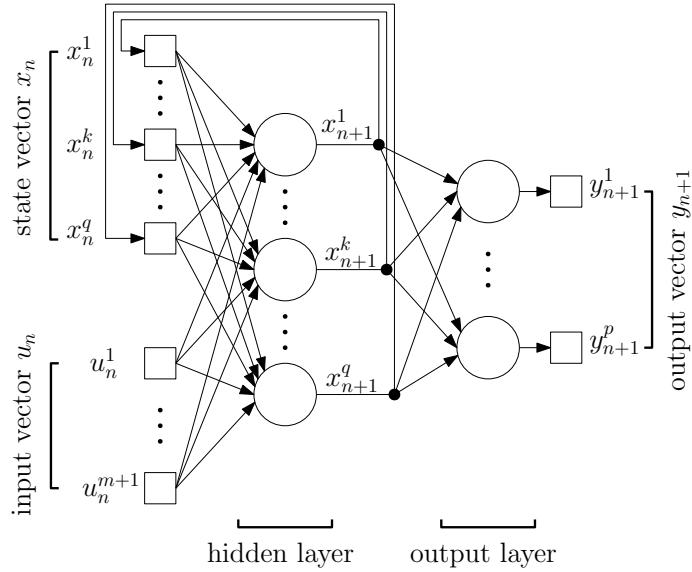
Figure 1.10: Structure of an RNN with one hidden layer. When predicting data, the input vector u_n corresponds to the the past data and the output vector y_{n+1} corresponds to the predicted data.

When incoming data arrives in a streaming fashion, with learning examples (u_n, y_{n+1}) coming one after another, it may be necessary to adapt the RNN synaptic weights θ with each newly available example. This process, called online learning, is opposed to offline learning. The design of online training algorithms for neural networks is an active area of research in machine learning (see for example [Jain et al., 2014, Pérez-Sánchez et al., 2018, Sahoo et al., 2017]). When performing online learning, the parameter vector is updated as the examples arrive, therefore we denote it by θ_n instead of θ. Equation 1.1 thus becomes Eq. 1.2.

$$x_{n+1} = F_{st}(x_n, u_n, \theta_n) \qquad y_n = F_{out}(x_n, \theta_n) \tag{1.2}$$

The classical methods for online training of RNNs are real-time recurrent learning (RTRL) and truncated back-propagation through time (BPTT). RTRL suffers from a high computational complexity $\mathcal{O}(q^4)$ and truncated BPTT from bias in the computation of the gradient loss. Reducing the time complexity of online learning algorithms for RNNs while calculating an unbiased estimate of the gradient of the loss L_n with respect to θ_n at each time step t_n has motivated the research community to devise more sophisticated training approaches (cf Table 1.2). Marschall et al. proposed a classification of these based on different characteristics. The algorithms referred to as "past-facing" adjust the synaptic weights θ_n based on a loss function defined over the training examples already observed. On the contrary, "future-facing" algorithms estimate and minimize a loss function defined over non-observed upcoming examples using only information until the current time step

---

use RNNs for which u_n directly influences the output y_n in our work.

Authors and year	Algorithm	Facing	Update of the synaptic weights	Bias	Memory complexity	Time complexity
[Williams and Zipser, 1989]	Real-time Recurrent Learning (RTRL)	Past	Deterministic	Closed-form	No	q^3	q^4
[Williams and Peng, 1990]	Truncated BPTT	Past	Deterministic	Closed-form	Yes	Tq	Tq^2
[Jaderberg et al., 2017]	Decoupled Neural Interfaces (DNI)	Future	Deterministic	Numerical	Yes	q^2	q^2
[Tallec and Ollivier, 2017a]	Unbiased Online Recurrent Optimization (UORO)	Past	Stochastic	Closed-form	No	q^2	q^2
[Mujika et al., 2018]	Kronecker-Factored RTRL (KF-RTRL)	Past	Stochastic	Closed-form	No	q^2	q^3
[Roth et al., 2018]	Kernel RNN learning (KeRNL)	Past	Deterministic	Numerical	Yes	q^2	q^2
[Benzing et al., 2019]	r-Optimal Kronecker-Sum Approximation (r-OK)	Past	Stochastic	Numerical	No	rq^2	rq^3
[Murray, 2019]	Random-Feedback Online Learning (RFLO)	Past	Deterministic	Closed-form	Yes	q^2	q^2
[Menick et al., 2020]	Sparse 1-Step Approximation (SnAp-1)	Past	Deterministic	Closed-form	Yes	q^2	q^2
[Marschall et al., 2020]	Reverse KF-RTRL	Past	Stochastic	Closed-form	No	q^2	q^3
[Marschall et al., 2020]	Efficient BPTT (E-BPTT)	-	Deterministic	Closed-form	No	Tq	q^2
[Marschall et al., 2020]	Future-facing BPTT (F-BPTT)	Future	Deterministic	Closed-form	No	Tq	Tq^2

Table 1.2: Classification of recent online learning algorithms for RNNs. In the last two columns, q designates the number of hidden units of the RNN. [10]

Adapted from [Marschall et al., 2020] (open-access article), Copyright JMLR 2020.

t_n. Past-facing algorithms compress the immediate influence matrix $\partial x_n/\partial \theta_n$, while future-facing algorithms predict the immediate credit assignment vector $\partial L_n/\partial x_n$. Another distinction concerns the update of the synaptic weights, which is either deterministic or stochastic, and uses either numerical approximations or a closed-form formula. For instance, reducing the RTRL complexity in unbiased online recurrent optimization (UORO) is made at the expense of introducing randomness in the update of the parameter vector. The last distinction is the tensor structure inherent to the influence matrix compressed by past-facing algorithms. Besides the online training algorithms listed in Table 1.2, we can also mention "NoBackTrack" [Ollivier et al., 2015], "Anticipated Reweighted Truncated Back Propagation" (ARTBP) [Tallec and Ollivier, 2017b], "Adaptive Truncated BPTT" [Aicher et al., 2020], and online spatio-temporal learning (OSTL) [Bohnstingl et al., 2020]. A theoretical proof of convergence for algorithms approximating RTRL, such as UORO and NoBackTrack has recently been established by Massé and Ollivier.

RNN characteristic	
Output layer size	p
Input layer size	m
Number of hidden layers	1
Size of the hidden layer	q
Activation function Φ	Hyperbolic tangent
Training algorithm	RTRL or UORO or SnAp-1
Optimization method	Stochastic gradient descent
Weights initialization	Gaussian
Gradient clipping	Yes
Input data normalization	Yes (online)

Table 1.3: Configuration of the RNNs forecasting the breathing signals in this thesis.

In the following, we use a vanilla or standard RNN architecture, described by Eq. 1.3, where the parameter vector θ_n consists of the elements of the matrices $W_{a,n}$, $W_{b,n}$, and $W_{c,n}$, of respective size $q \times q$, $q \times (m+1)$, and $p \times q$. Φ is the nonlinear activation function, and in this thesis we use the hyperbolic tangent function (Eq. 1.4).

$$F_{st}(x_n, u_n, \theta_n) = \Phi(W_{a,n}x_n + W_{b,n}u_n) \qquad F_{out}(x_n, \theta_n) = W_{c,n}x_n \qquad (1.3)$$

$$\Phi \begin{pmatrix} a_1 \\ ... \\ a_q \end{pmatrix} = \begin{pmatrix} \phi(a_1) \\ ... \\ \phi(a_q) \end{pmatrix} \quad \text{where} \quad \phi(a) = tanh(a) \qquad (1.4)$$

We normalize the multidimensional RNN input u_n by first calculating its mean μ_{train} and standard deviation σ_{train} of the training set and then by replacing each

new arriving vector input u_n with $(u_n - \mu_{train})/\sigma_{train}$ (the division is performed element-wise). This enables faster and more efficient learning with stochastic gradient descent. Moreover, RNNs updated by the gradient rule may be unstable. We thus prevent large weight updates by clipping the gradient norm to avoid numerical instability [Pascanu et al., 2013]. The characteristics of the RNNs studied are summarized in Table 1.3.

1.4 Study objectives and structure of the thesis

The primary goal of this thesis is to develop and evaluate algorithms involving online training of RNNs that will positively impact breathing motion management in radiotherapy. Forecasting the respiratory motion will help estimate the tumor position accurately and in turn, reduce irradiation to healthy tissues. This will ultimately lead to a decrease in the occurrence of side effects such as radiation pneumonitis or pulmonary fibrosis.

In Chapter 2, we track chest internal points by performing deformable registration of CT scan images with the pyramidal Lucas-Kanade optical flow algorithm. These points are close to the tumor and may represent implanted fiducial markers. RTRL is used to predict their 3D positions. A linear relationship between the latter and the chest internal displacement vector field (DVF) then enables estimating future images.

In Chapter 3, we consider the problem of forecasting the position of external markers placed on the chest and abdomen, recorded by infra-red cameras. We compare the performance of RTRL, UORO, least mean squares (LMS) and linear regression for different values of the horizon: the time interval in advance for which prediction is performed. We also examine robustness to irregular motion and trade-off between accuracy and jitter. The latter is a measure of the oscillatory behavior of the predicted signal and needs to be minimized to ensure proper control of the robotic system.

Chapter 2

Prediction of the position of chest internal points using Real-Time Recurrent Learning

2.1 Introduction

2.1.1 Systems for lung tumor tracking

In image-guided radiotherapy (IGRT), several methods have been designed to track the three-dimensional (3D) position of the lung tumor in real-time as accurately as possible. Indeed, it has been mentioned that "a systematic tracking error of 2mm can be significant" [Murphy, 2004] regarding dose delivery accuracy and safety. One of these modalities is called beam gating and consists of turning on and off a static beam according to the recorded tumor position. Conversely, in beam tracking, the radiation beam follows the tumor and conforms to its position as it moves.

Visualizing clearly a lung tumor in 3D during the radiotherapy treatment is difficult. Therefore, one often records surrogate signals and uses a correspondence model to infer the tumor location. Such signals may be the position of internal or external markers. Internal fiducial markers are small metallic objects, such as gold coils or spheres, that are implanted in the lung, close to the tumor, prior to the radiotherapy treatment (Fig. 2.1). Marker placement can be percutaneous (using a needle), endovascular, or endobronchial (e.g., Veran system). Their position can be measured by fluoroscopic imaging systems, such as CyberKnife's orthogonal X-ray imaging sources and flat-panel detectors (Fig. 2.2) [Khankan et al., 2017]. In contrast, external fiducial markers are objects attached to the patient's chest whose

---

The content of this chapter has been published in the journal "Computerized Medical Imaging and Graphics" [Pohl et al., 2021a]. More details are available in the journal article.
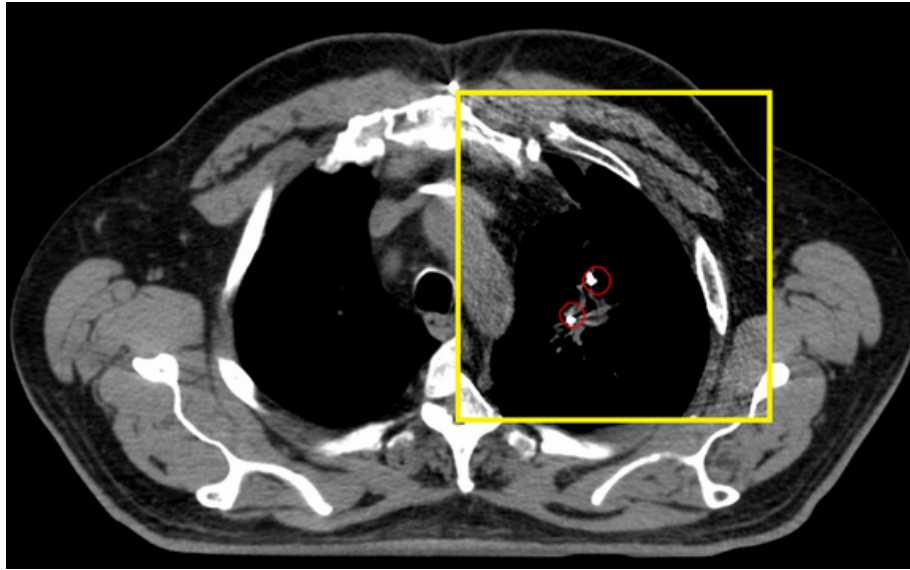
Figure 2.1: Implanted fiducial markers, circled in red, near a lung tumor on a 4DCT image. [1]
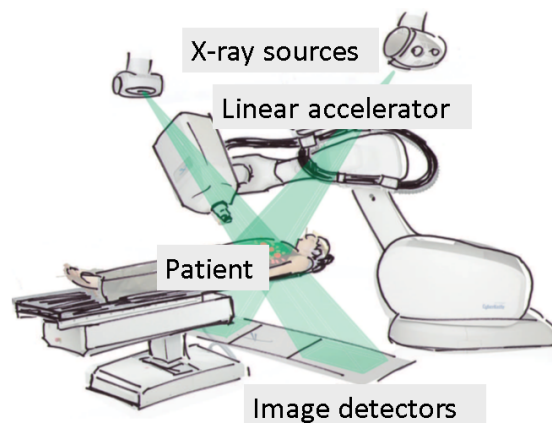
Figure 2.2: Cyberknife radiosurgery device. X-ray tubes and kV flat panel detectors are used to acquire images of the implanted fiducial markers, whose 3D position is automatically detected by an image analysis software. That position is used to guide the robotic arm as the patient breathes during the treatment. [2]

position can be recorded using an infrared tracking system.

2.1.2 Prediction for latency compensation

Due to mechanical limitations, signal exchange, and processing time, there is a delay between anatomical image acquisition and radiation beam activation. That latency time is specific to each treatment system and was reported to be between 100ms and 2s in general [Verma et al., 2010]. As that phenomenon impedes accurate radiation delivery and can lead to more damage to healthy tissue, there is a need for proper

---

[1]Figure used in [McDonald et al., 2019] (open access article)
[2]Figure used in [Roberge and Cabrera, 2011] (open access article)

compensation using forecasting algorithms (Fig. 1.7). In this work, we propose and investigate a standard online training algorithm for recurrent neural networks (RNN) called real-time recurrent learning (RTRL) and apply it to the prediction of lung tumor position.

2.1.3 Chest image registration

In the proposed study, we artificially track arbitrary internal points near the tumor by calculating the deformation or displacement vector field (DVF) in the whole chest in computed tomography (CT) scan images. This internal correspondence calculation process, known as deformable image registration (DIR), has been extensively studied for various applications in radiotherapy, such as tumor tracking, correction of the irradiation plan relative to the patient position on the couch, and ventilation imaging for lung function estimation. The different DIR algorithms can be classified into two categories. The first category is referred to as feature-based registration [Ehrhardt et al., 2013]. In feature-based registration, highly structured image regions such as vertebrae, ribs, the lung surface, the bronchial and vascular tree are first matched by algorithms such as the iterative closest point (ICP) [Besl and McKay, 1992], and a dense deformation field is subsequently calculated using interpolation methods such as B-splines [McClelland et al., 2006]. In contrast, intensity-based deformable registration methods consist in calculating directly the entire global deformation using only image intensity information without performing segmentation or feature extraction beforehand. The Lucas-Kanade optical flow [Lucas et al., 1981], the Horn-Schunck optical flow [Horn and Schunck, 1981], and the different variants of the "Demons algorithm" [Thirion, 1998, 1995] are examples in that category. Computing large displacements with these methods can be difficult. To cope with this problem, an approach referred to interchangeably as "coarse to fine strategy", "pyramidal implementation", or "multi-resolution scheme" can be used. It consists of iteratively calculating and refining the DVF of gradually more detailed versions of the images to be matched.

2.1.4 Contributions of the proposed study

The main contributions of this study are the following. First, we discuss in detail parameter optimization of the iterative and pyramidal version of the Lucas-Kanade optical flow algorithm in the context of DIR of chest CT scan images. That algorithm has often been used in chest imaging [Akino et al., 2014, Dhont et al., 2019, Xu et al., 2008], but there are no studies about proper selection of the parameters for accurate registration of chest CT scan images, to the extent of our knowledge. Secondly, this is the first application of RNNs trained with the RTRL algorithm to predict breathing signals and compensate for the inherent latency of treatment systems in
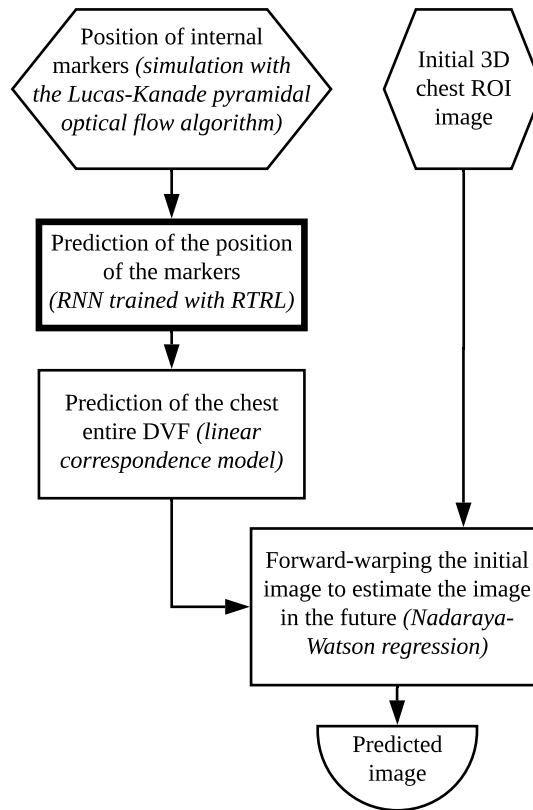
Figure 2.3: Overview of the proposed prediction algorithm

radiotherapy. The optimal choice of the RNN parameters is discussed thoroughly. In contrast to the related studies about marker position prediction with artificial neural networks (ANN) mentioned in the general introduction, our study describes the simultaneous prediction of the position of 3 markers, rather than the position of 1 marker only. Finally, we propose a simple method for reconstructing and predicting 3D lung tumor images given only the trajectory of internal markers and an initial 3D image of that tumor (Fig. 2.3).

2.2   Materials and methods

2.2.1   Chest image data

The data used in this study consists of chest 3D 16-bit image sequences of 4 patients with lung cancer. Each of the 4 sequences consists of ten 3D images of the chest at different phases of the breathing process. The first sequence is a four-dimensional cone beam computed tomography (4D-CBCT) (four-dimensional cone-beam computed tomography) sequence acquired by the Elekta Synergy XVI system in the University of Tokyo Hospital and the three remaining sequences are four-dimensional computed tomography (4DCT) (four-dimensional computed tomography) sequences acquired by a 16-slice helical CT simulator (Brilliance Big Bore, Philips Medical Sys-

$$t = t_1 \qquad t = t_3 \qquad t = t_5 \qquad t = t_7 \qquad t = t_9$$
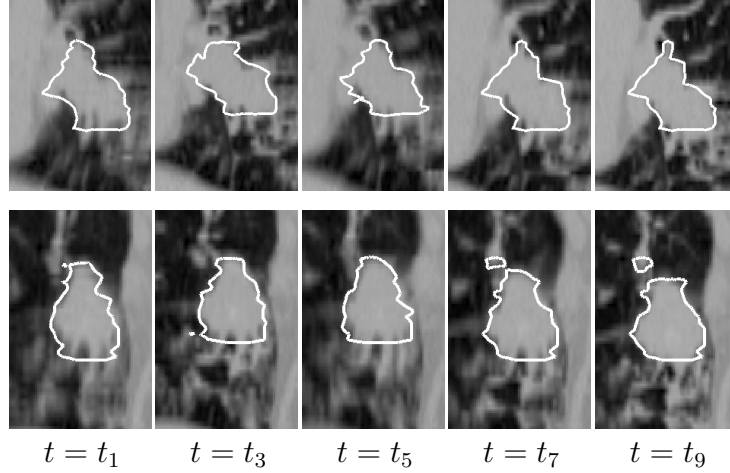
Figure 2.4: Sagittal (top line) and coronal (bottom line) cross-sections of the 3D ROI of patient 2 at different phases of the breathing cycle. The coordinates of the cross-sections are the same as in Fig. 2.5. The tumor was delineated by a physician in each image.

Patient	1	2	3	4
ROI size (in mm^3)	$65 \times 56 \times 82$	$76 \times 87 \times 116$	$41 \times 39 \times 56$	$80 \times 79 \times 67$
T (in s)	400	320	800	480
A (in mm)	2.0	1.5	4.0	2.5

Table 2.1: Description of the ROI size and motion parameters, defined in Eq. 2.1, for each patient.

tems) in Virginia Commonwealth University Massey Cancer Center.

Each sequence was resampled using trilinear interpolation such that 1 voxel corresponds to $1mm^3$. For each sequence, a 3D region of interest (ROI) encompassing the tumor was selected (Figs. 2.4, 2.5) and the size of each of them is indicated in Table 2.1. Then, each sequence was extended to $N = 2400$ images[3] by introducing a breathing drift in the z-direction (the spine axis). Indeed, it has been reported that the axis along which the respiratory drift is the greatest is the cranio-caudal axis [Takao et al., 2016]. More precisely, $I(\cdot, t_k)$, the image at time t_k, where $k \in \{1, ..., 2400\}$, results from the translation along the z-axis defined in Eq. 2.1. In the following, we denote by $I(\vec{x}, t)$ the image intensity at pixel $\vec{x}$ and time t.

$$I(\vec{x}, t_k) = I\left(\vec{x} + A sin\left(\frac{2\pi t_k}{T}\right)\vec{e_z}, t_{k \bmod 10}\right) \qquad (2.1)$$

In this equation, $\vec{x}$ refers to a selected voxel in the image $I(\cdot, t_k)$, $\vec{e_z}$ is a unit vector in the z-direction, and A and T are respectively the amplitude and the period

---

[3]Prior to the extension of the sequences, the 10 original images were permuted for each patient so that each series begins at a phase where the tumor is approximately located at its center position, with regards to the overall cyclic breathing motion. This is performed in order to increase the accuracy of the optical flow registration that follows.
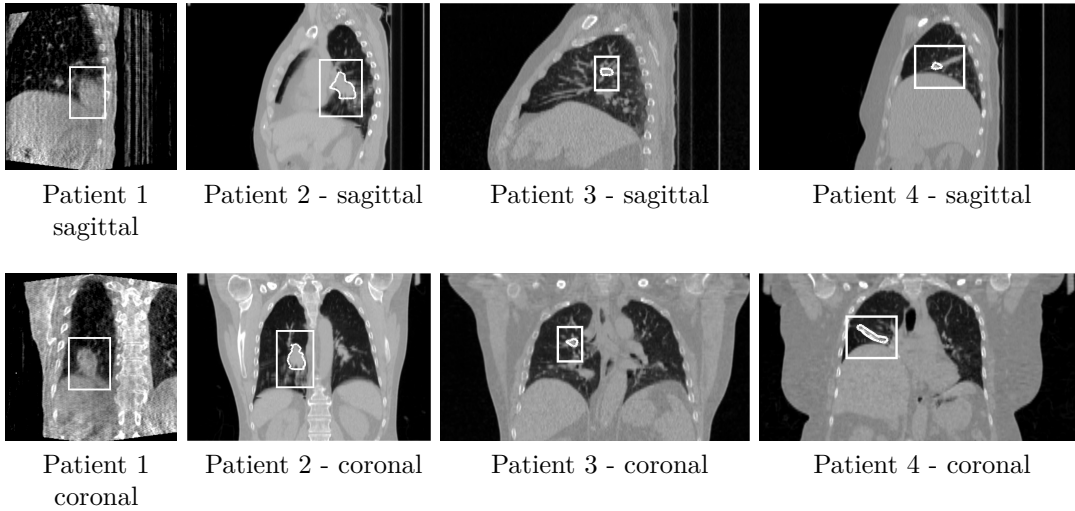
Figure 2.5: Sagittal (top line) and coronal (bottom line) cross-sections of the 3D ROI
of each patient at $t = t_1$. The tumor of each of patients 2, 3, and 4 was delineated
by a physician.

of the added sinusoidal drift (see Table 2.1). The voxel intensity values on the right
side of Eq. 2.1 are computed using trilinear interpolation. Finally, Poisson noise
with parameter $\lambda = 1,000$ is added to the extended sequences, given that this type
of noise is prevalent in CT scan imaging [Boas and Fleischmann, 2012, Diwakar and
Kumar, 2018]. Because the average breathing cycle of an adult lasts 4s [Barrett
et al., 2019], we can assume that the interval of time between each image is equal
to 400ms, or in other words, that the sampling rate is equal to 2.5Hz.

2.2.2 Chest image registration

First, the pyramidal and iterative Lucas-Kanade optical flow algorithm (Algorithm
1) is used to calculate $\vec{u}(\cdot, t)$, the DVF between the first image (at time t_1) and the
image at time t, which approximately satisfies Eq. 2.2.

$$I(\vec{x}, t_1) = I(\vec{x} + \vec{u}(\vec{x}, t), t) \tag{2.2}$$

In the pyramidal and iterative Lucas-Kanade optical flow algorithm, a multires-
olution representation of the two images to be registered, $I(\cdot, t_1)$ and $I(\cdot, t)$, is first
computed. For this purpose, an initial low-pass Gaussian filter of standard devi-
ation σ_{init}^{DVF} is first applied to both of them. Given the representations of $I(\cdot, t_1)$
and $I(\cdot, t)$ at the layer l, denoted by $I_l(\cdot, t_1)$ and $I_l(\cdot, t)$, these representations have
another low-pass Gaussian filter of standard deviation σ_{sub} applied to them. They
are then subsampled by a factor 2 to create their representations at the layer $l + 1$,
$I_{l+1}(\cdot, t_1)$ and $I_{l+1}(\cdot, t)$. Indeed, prior Gaussian filtering has been shown to increase
the accuracy of the resulting computed optical flow in general [Sharmin and Brad,

2012].

The displacement vector at a given voxel $\vec{x_0}$ and layer l between t_1 and t is the argument $\vec{v_0}$ that minimizes the energy $E(\vec{v})$ in Eq. 2.3.

$$E(\vec{v}) = \sum_{\vec{x}} K_{\sigma_{LK}}(\|\vec{x} - \vec{x_0}\|_2) \left[\vec{\nabla} I_l(\vec{x}, t_1) \cdot \vec{v} + \frac{\partial I_l}{\partial t}(\vec{x}, t_1) \right]^2 \qquad (2.3)$$

In that equation, $\vec{\nabla}$ refers to the spatial gradient operator, calculated here by applying the Scharr filter [Levkine, 2012, Wikipedia contributors, 2020]. Furthermore, $K_{\sigma_{LK}}$ refers to the probability density function of a centered normal distribution of standard deviation σ_{LK} (Eq. 2.4).

$$K_\sigma(x) = \frac{1}{\sqrt{2\pi\sigma^2}} exp\left(-\frac{x^2}{2\sigma^2}\right) \qquad (2.4)$$

The minimization of $E(\vec{v})$ is iterated to decrease the residual error, and the displacement field calculated at the layer l is propagated at the layer $l-1$ to give a first approximation of the displacement field at the layer $l-1$. The algorithm is detailed in [Bouguet et al., 2001, Fleet and Weiss, 2006].

2.2.3   Prediction of the position of internal points

After the computation of the optical flow, $r = 3$ internal points $\vec{x_1}, ..., \vec{x_r}$ are selected close to the tumor in the initial image at $t = t_1$. They are considered to be points of known position during the treatment. It is reported in [Harley et al., 2010] that internal markers are usually implanted near or inside the tumor and that their number is generally 3 or 4.

We predict the motion of these r points using a standard RNN (cf section 1.3). The input u_n of the RNN is a vector of size $3rL + 1$, where L represents the signal history length (SHL): the time interval in the past, the information of which is used for making one prediction. u_n consists of the concatenation of the displacement vectors $\vec{u}(\vec{x_p}, t_n), ..., \vec{u}(\vec{x_p}, t_{n+L-1})$ for each point $p \in [1, ..., r]$ (Eq. 2.5). An additional 1 was added to account for a bias unit. Each time-series $\left(u_d(\vec{x_p}, t_n)\right)_{n=1,...,N}$, for $d = x, y$, and z, and $p \in [1, ..., r]$, is normalized prior to being used as an input, in order to facilitate the learning process. The output y_{n+1} of the RNN is a vector of size $3r$ consisting of the position of these r points at the time t_{n+L} (Eq. 2.5). In particular, this means that the positions of all the markers are predicted simultaneously. Specifically, not only information concerning marker 1 but also the positions of markers 2,..., $r = 3$ are used to predict the position of that first marker, which may help in mitigating the influence of noise.

Algorithm 1 Pyramidal Iterative Lucas-Kanade Optical Flow

1: **Input** :
2: I initial image at time t_1, J image at an arbitrary time t
3: **Parameters** :
4: σ_{init}^{DVF}, σ_{sub}, σ_{LK} : standard deviation of various Gaussian filters
5: n_{layers} : number of layers, n_{iter} : number of iterations
6:
7: **Pyramidal representation of I and J**
8: In what follows $\mathcal{G}(\cdot,\sigma)$ designates the isotropic Gaussian filter operator with standard deviation σ, and $\mathcal{S}_2(\cdot)$ the subsampling operator by a factor 2, defined by $S_2(I)(\vec{x}) = I(2\vec{x})$
9: $I_1 := \mathcal{G}(I,\sigma_{init}^{DVF})$, $J_1 := \mathcal{G}(J,\sigma_{init}^{DVF})$ (initial filtering)
10: **for** $l = 1,...,n_{layers}-1$ **do**
11: $\quad I_{l+1} := \mathcal{S}_2(\mathcal{G}(I_l,\sigma_{sub}))$
12: $\quad J_{l+1} := \mathcal{S}_2(\mathcal{G}(J_l,\sigma_{sub}))$
13: $g_{n_{layers}} := 0$ (DVF guess initialization)
14:
15: **Computation of the DVF**
16: **for** $l = n_{layers},...,1$ **do**
17: $\quad$ **for** $x \in I_l$ **do**
18: $\quad\quad G(x) := \sum_v K_{\sigma_{LK}}(\|x-v\|_2) \begin{bmatrix} I_x(v)^2 & I_x(v)I_y(v) & I_x(v)I_z(v) \\ I_x(v)I_y(v) & I_y(v)^2 & I_y(v)I_z(v) \\ I_x(v)I_z(v) & I_y(v)I_z(v) & I_z(v)^2 \end{bmatrix}$
19: $\quad\quad$ where I_x (resp. I_y, I_z) is the partial derivative of I_l in the x-direction (resp. y and z directions) and $K_{\sigma_{LK}}$ is defined in Eq. 2.4
20: $\quad r_l^0 := 0$ (DVF refinement initialization)
21: $\quad$ **for** $i = 1,...,n_{iter}$ **do**
22: $\quad\quad$ **for** $x \in I_l$ **do**
23: $\quad\quad\quad \delta I^i(x) := I_l(x) - J_l(x + g_l(x) + r_l^{i-1}(x))$
24: $\quad\quad$ **for** $x \in I_l$ **do**
25: $\quad\quad\quad b(x) := \sum_v K_{\sigma_{LK}}(\|x-v\|_2) \begin{bmatrix} \delta I^i(v)I_x(v) \\ \delta I^i(v)I_y(v) \\ \delta I^i(v)I_z(v) \end{bmatrix}$
26: $\quad\quad\quad r_l^i(x) := r_l^{i-1}(x) + G(x)^{-1}b(x)$
27: $\quad$ **if** $l > 1$ **then**
28: $\quad\quad$ **for** $x \in I_{l-1}$ **do**
29: $\quad\quad\quad g_{l-1}(x) := 2(g_l(x/2) + r_l^{n_{iter}}(x/2))$
30:
31: **Output :** 3D displacement field $u(x) := g_1(x) + r_1^{n_{iter}}(x)$

$$u_n = \begin{pmatrix} 1 \\ u_x(\vec{x_1},t_n) \\ u_y(\vec{x_1},t_n) \\ u_z(\vec{x_1},t_n) \\ ... \\ u_z(\vec{x_r},t_n) \\ u_x(\vec{x_1},t_{n+1}) \\ ... \\ u_z(\vec{x_r},t_{n+L-1}) \end{pmatrix} \quad y_{n+1} = \begin{pmatrix} u_x(\vec{x_1},t_{n+L}) \\ u_y(\vec{x_1},t_{n+L}) \\ u_z(\vec{x_1},t_{n+L}) \\ ... \\ u_z(\vec{x_r},t_{n+L}) \end{pmatrix} \quad (2.5)$$

The RNN is trained using RTRL (Algorithm 2). Prior to the learning process,

each synaptic weight is initialized according to a normal distribution of standard deviation σ_{init}. RTRL is an online learning method, and so the weight matrices $W_{a,n}$, $W_{b,n}$, and $W_{c,n}$ are updated at every time step to take into account the recent changes in the breathing characteristics of the patient. Given the predicted positions of the markers y_n and the real position of the markers y_n^*, we can compute the instantaneous error vector e_n and instantaneous error function L_n as in Eq. 2.6.

$$e_n = y_n^* - y_n \qquad L_n = \frac{1}{2}\|e_n\|_2^2 \tag{2.6}$$

The weight matrix $W_{k,n+1}$ at time $n+1$, where $k = a, b$ or c, is computed from the corresponding weight matrix $W_{k,n}$ at time n by performing a single gradient descent update. However, RNNs updated by the gradient rule may be unstable, and as proposed in [Pascanu et al., 2013], we prevent large weight updates by clipping the gradient norm to address instability. Details concerning the calculation of the terms $\partial L_n / \partial W_{k,n}$ can be found in [Haykin et al., 2009], whose description was extended in this work to encompass RNNs with a multidimensional output vector (Appendix ??). The RNN main characteristics are summarized in Table 1.3. The RTRL computation complexity is $\mathcal{O}(q^2(q+m)(q+p))$.

2.2.4 Application to chest image prediction

In what follows, we propose a simple method to predict future 3D images of the ROI based on marker position prediction as described in 2.2.3. First, we assume that the motion of each voxel is linked to the motion of the markers via a linear relationship, which indirectly models the connectivity between the tissues (Eq. 2.7). The coefficients $\gamma_p(\vec{x})$ are calculated using linear regression.

$$\vec{u}(\vec{x}, t) = \sum_{p=1}^{r} \gamma_p(\vec{x})\vec{u}(\vec{x_p}, t) \tag{2.7}$$
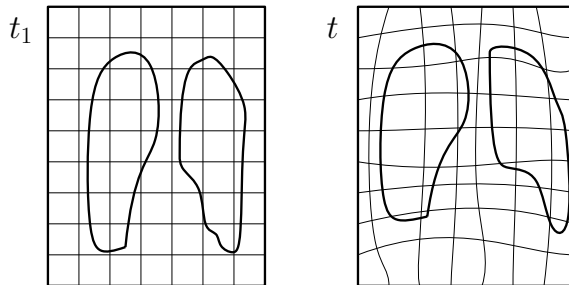
Figure 2.6: Warping the initial lung image at $t = t_1$ to estimate the lung image at t

Given the position of the markers at time $t_1, ..., t_n$, their position at time t_{n+1} can be predicted using the RNN, and the whole DVF at t_{n+1}, $\vec{u}(\cdot, t_{n+1})$, can then be recovered using Eq. 2.7. In order to estimate the image at time t_{n+1}, we can

---

Algorithm 2 Real-Time Recurrent Learning

1: **Parameters** :
2: L : signal history length, r : nb. of internal points, $m = 3rL$ input space dimension
3: q : state space dimension, $p = 3r$ output space dimension, η : learning rate
4: glstau : gradient threshold, σ_{init} : std. dev. of the initial weights
5:
6: **Initialization**
7: $W_{a,n=1}$: $q \times q$ matrix initialized according to a Gaussian distribution $\mathcal{N}(0, \sigma_{init}{}^2)$
8: $W_{b,n=1}$: $q \times (m+1)$ matrix initialized according to a Gaussian distribution $\mathcal{N}(0, \sigma_{init}{}^2)$
9: $W_{c,n=1}$: $p \times q$ matrix initialized according to a Gaussian distribution $\mathcal{N}(0, \sigma_{init}{}^2)$
10: State vector $x_{n=1} := 0_{q \times 1}$
11: **for** $j = 1, ..., q$ **do**
12: $\quad \Lambda_{j,n=1} := 0_{q \times (q+m+1)}$
13:
14: **Learning and prediction**
15: **for** $n = 1, 2, ...$ **do**
16: $\quad y_n := W_{c,n} x_n$ (prediction), $e_n := y_n^* - y_n$ (error vector update)
17: $\quad$ **for** $j = 1, ..., q$ **do** (gradient calculation)
18: $\quad\quad w_{j,n} := \begin{bmatrix} w_{a,j,n} \\ w_{b,j,n} \end{bmatrix}$ where $\begin{matrix} W_{a,n} = [w_{a,1,n}, ..., w_{a,q,n}]^T \\ W_{b,n} = [w_{b,1,n}, ..., w_{b,q,n}]^T \end{matrix}$
19: $\quad\quad \Delta w_{j,n} := \Lambda_{j,n}^T W_{c,n}^T e_n$
20: $\quad \Delta W_{c,n} := e_n \otimes x_n$
21: $\quad \kappa := \sqrt{\|\Delta w_{1,n}\|_2^2 + ... + \|\Delta w_{q,n}\|_2^2 + \|\Delta W_{c,n}\|_2^2}$
22: $\quad$ **if** $\kappa > \tau$ **then** (gradient clipping)
23: $\quad\quad$ **for** $j = 1, ..., q$ **do**
24: $\quad\quad\quad \Delta w_{j,n} := \frac{\tau}{\kappa} \Delta w_{j,n}$
25: $\quad\quad \Delta W_{c,n} := \frac{\tau}{\kappa} \Delta W_{c,n}$
26: $\quad W_{c,n+1} := W_{c,n} + \eta \Delta W_{c,n}$ (gradient update)
27: $\quad \xi_n := \begin{bmatrix} x_n \\ u_n \end{bmatrix}$, $\Phi_n := diag(\phi'(w_{1,n}^T \xi_n), ..., \phi'(w_{q,n}^T \xi_n))$
28: $\quad$ **for** $j = 1, ..., q$ **do**
29: $\quad\quad w_{j,n+1} := w_{j,n} + \eta \Delta w_{j,n}$ (gradient update)
30: $\quad\quad U_{j,n} := \begin{bmatrix} 0 \\ \xi_n^T \\ 0 \end{bmatrix} \leftarrow j^{th}$ row, $\Lambda_{j,n+1} := \Phi_n[W_{a,n}\Lambda_{j,n} + U_{j,n}]$
31: $\quad W_{a,n+1} := [w_{a,1,n+1}, ..., w_{a,q,n+1}]^T$, $W_{b,n+1} := [w_{b,1,n+1}, ..., w_{b,q,n+1}]^T$
32: $\quad x_{n+1} := \Phi(W_{a,n} x_n + W_{b,n} u_n)$ (hidden states update)

---

warp the initial image $I(\cdot, t_1)$ by the field $\vec{u}(\cdot, t_{n+1})$ (Fig. 2.6). This relies on the assumption that the image at t_{n+1} can be approximately reconstructed via warping the image at t_1.
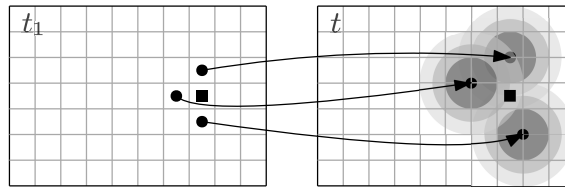
Figure 2.7: Forward-warping the initial image at $t = t_1$ using Nadaraya-Watson regression with a Gaussian kernel. The closer a point at $t = t_1$ arrives next to the square point at t, the more it contributes to the intensity of that square point at t.

In order to estimate the image at time t from the DVF at time t_1, we use the

Nadaraya-Watson non-parametric regression method, described in Fig. 2.7 and Eq. 2.8. The modified kernel $\widetilde{K}$ used in that equation is a variant of the Gaussian kernel K defined in Eq. 2.4. σ_w represents the standard deviation of the new kernel $\widetilde{K}$ and h_w represents the window size of the kernel calculation. Imposing an arbitrary window size h_w is necessary because the calculations would be slow otherwise[4]. However, this may lead to some voxels in the destination image $I(\cdot, t)$ not having any corresponding voxel in the source image $I(\cdot, t_1)$. Therefore, h_w needs to be chosen appropriately large. Furthermore, σ_w needs to be selected such that the images do not appear either too blurry or with too many artifacts, such as inappropriate impainting due to voxels in the destination image having only one antecedent voxel. Theoretical details about the Nadaraya-Watson statistical estimator can be found in [Tsybakov, 2008]. The computational complexity of image warping is $\mathcal{O}(V h_w^3)$ where V is the volume (in voxels) of the image considered.

$$I_{NW}(\vec{x}, t) = \frac{\sum_{\vec{p}} I(\vec{p}, t_1) \widetilde{K}_{\sigma_w, h_w} \left(\|\vec{x} - (\vec{p} + \vec{u}(\vec{p}, t))\|_2 \right)}{\sum_{\vec{p}} \widetilde{K}_{\sigma_w, h_w} \left(\|\vec{x} - (\vec{p} + \vec{u}(\vec{p}, t))\|_2 \right)} \tag{2.8}$$

$$\widetilde{K}_{\sigma_w, h_w}(x) = \begin{cases} K_{\sigma_w}(x) & \text{if } |x| < h_w \quad \text{(cf Eq. 2.4)} \\ 0 & \text{otherwise} \end{cases} \tag{2.9}$$

2.3 Results and Discussion

2.3.1 Chest image registration

In order to determine the parameters giving the most accurate DVF for each image sequence, we calculated the registration error defined in Eq. 2.10, for the following set of parameters, on the initial ROI sequences of $n = 10$ images :

- $\sigma_{init}^{DVF} \in \{0.2, 0.5, 1.0, 2.0\}$

- $\sigma_{sub} \in \{0.2, 0.5, 1.0, 2.0\}$

- $\sigma_{LK} \in \{1.0, 2.0, 3.0, 4.0\}$

- number of layers $n_{layers} \in \{1, 2, 3, 4\}$

- number of iterations $n_{iter} \in \{1, 2, 3\}$

$$e_{DVF} = \sqrt{\frac{1}{(n-1)|I|} \sum_{k=2}^{n} \sum_{\vec{x}} \left[I(\vec{x}, t_1) - I(\vec{x} + \vec{u}(\vec{x}, t_k), t_k) \right]^2} \tag{2.10}$$

---

[4]When calculating the optical flow, $\widetilde{K}$ was also used instead of K to process the data reasonably fast, but we did not introduce this notation for two reasons. First, it is generally assumed that there is a window when using a Gaussian kernel so that was implicit. Secondly, adjusting the size of the window is particularly important when reconstructing images, because of the problem of voxels without antecedent.
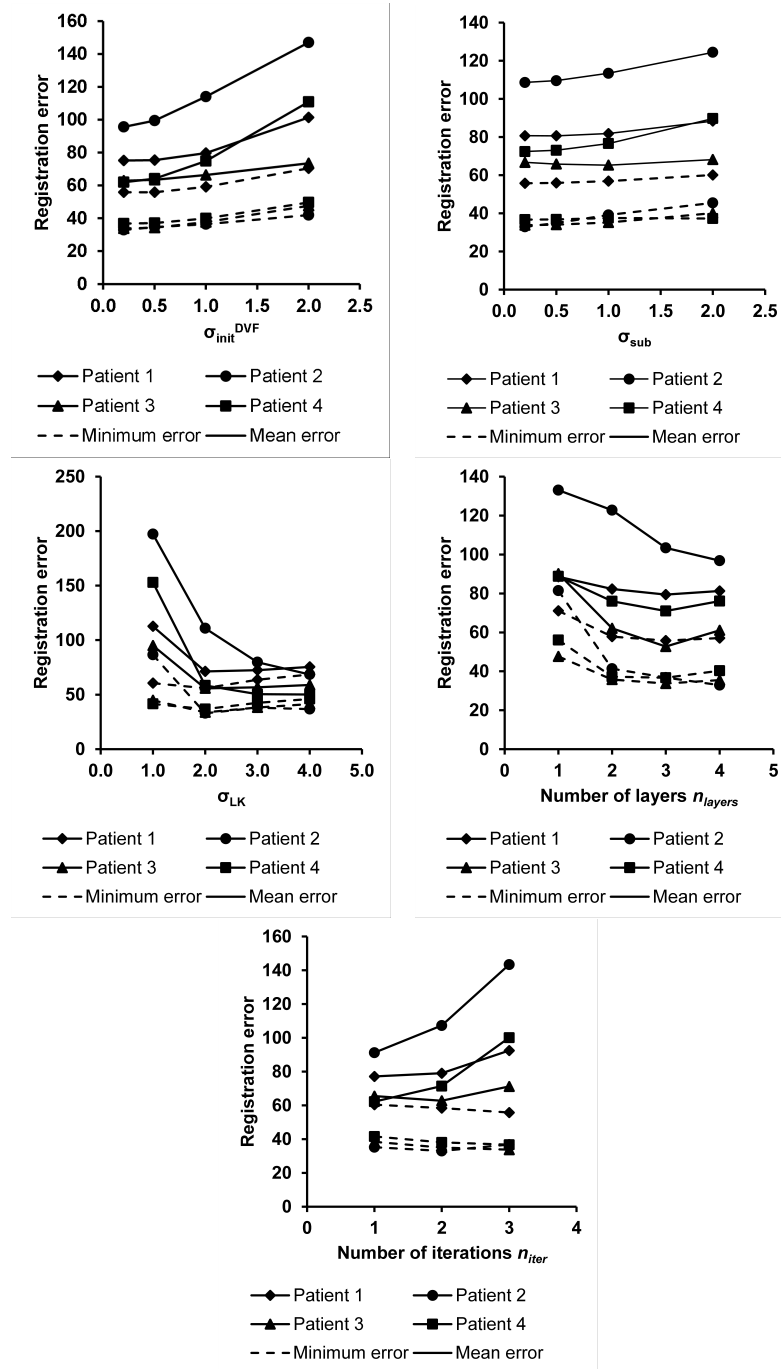
Figure 2.8: Registration error e_{DVF} as a function of the parameters of the Lucas-Kanade iterative and pyramidal optical flow algorithm (cf Eq. 2.10). The minimum error refers to the minimum of the registration error across every parameter, and the mean error refers to the registration error averaged over the four parameters not studied in each graph.

The results of this grid search optimization are displayed in Fig. 2.8 and Fig. 2.9. Fig. 2.8 shows for each parameter two different types of errors. The first one is the mean registration error: the registration error averaged over every other parameter. The second type is the minimum registration error, which represents the minimum error over the entire set of parameters. Both the minimum error
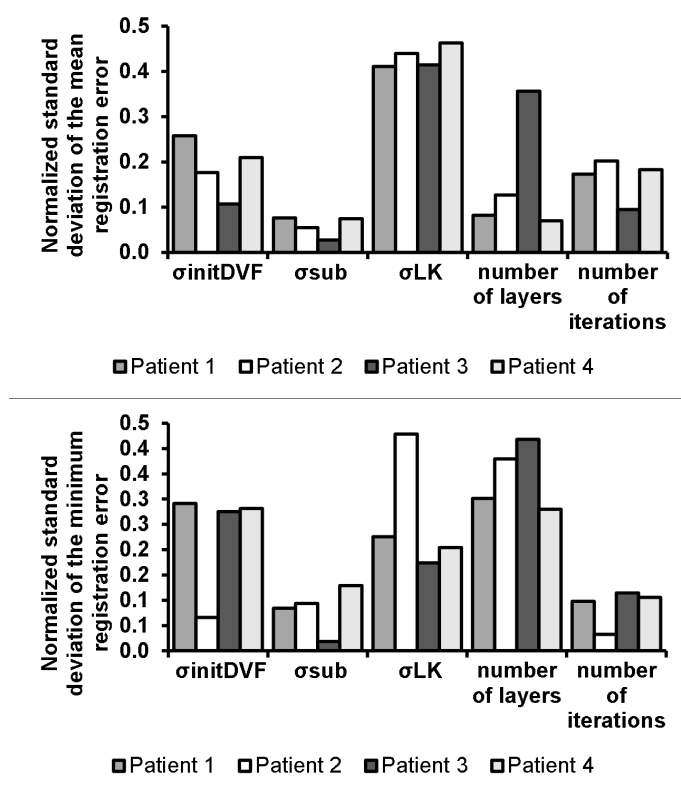
Figure 2.9: Relative influence of the parameters of the Lucas-Kanade optical flow algorithm on the registration error.

and mean error increase for every patient when σ_{init}^{DVF} increases, which means that initial filtering had a detrimental effect on the accuracy of the registration, because the initial images were not very noisy. Similarly, the registration minimum error increases with σ_{sub}, except for patient 3. Both errors as a function of σ_{LK} are either decreasing or strictly convex, except for the minimum error of patient 2. Setting $\sigma_{LK} = 1.0$ (lowest value tested) leads to large mean registration errors. Likewise, the errors associated with n_{layers} are either decreasing or strictly convex. Using only one layer (simple Lucas-Kanade algorithm) entails large errors, because the motion of the chest has a high amplitude relative to the imaging resolution. This supports the previous claims in the literature that a multiresolution scheme is generally needed for accurate registration of chest CT scan images [Xu et al., 2008, Zhang et al., 2008]. Increasing n_{iter} results in a decrease in the minimum error, except for patient 2, and an increase in the mean error, except for patient 3. For all the patients, $\sigma_{init}^{DVF} = 0.2$, $\sigma_{sub} = 0.2$, and $\sigma_{LK} = 2.0$ led to the highest displacement field accuracy. The registration was the most accurate using $n_{layers} = 3$ and $n_{iter} = 3$ for patients 1, 3, and 4, and using $n_{layers} = 4$ and $n_{iter} = 2$ for patient 2.

The normalized standard deviation of the mean error and minimum error relative to each parameter is reported in Fig. 2.9. "Normalization" means that for each patient, the sum of all the contributions was set to be equal to 1 by multiplying them by a proportionality coefficient. σ_{LK} is the parameter that contributes the most to
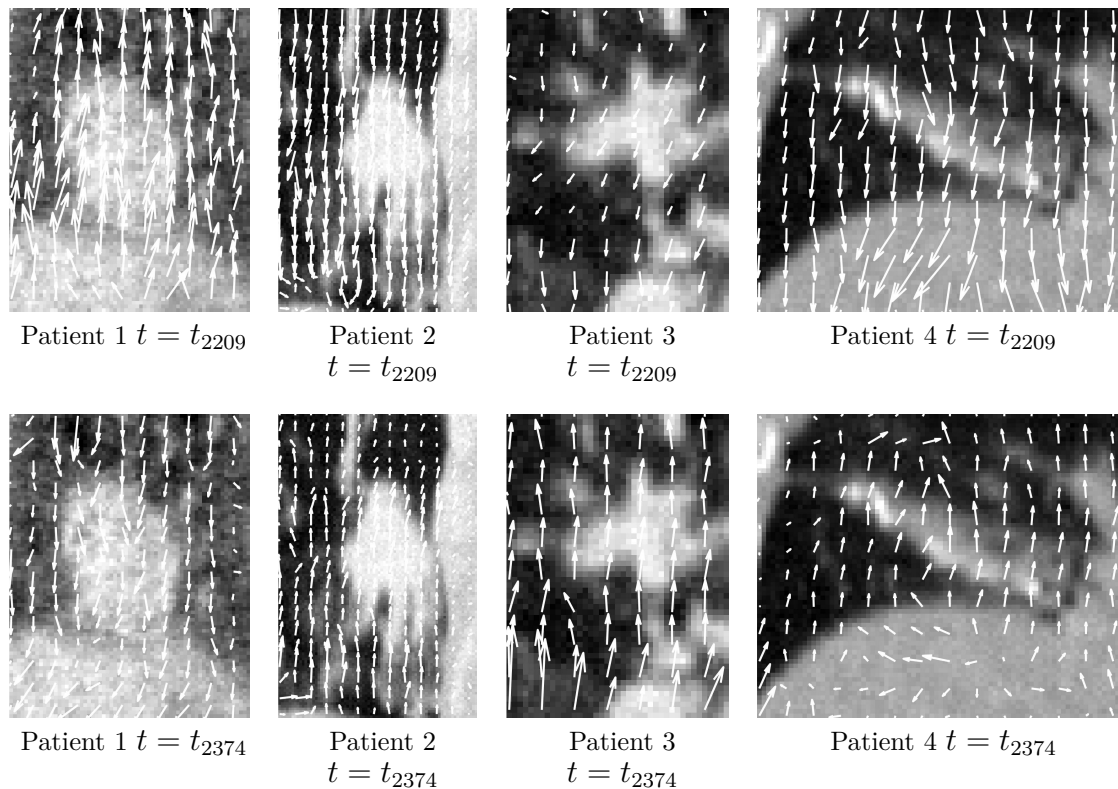
Figure 2.10: Displacement vector field in the ROI for each patient at $t = t_{2209}$ (end of expiration for patient 1 and end of inspiration for the other patients) and $t = t_{2374}$ (opposite case) projected in a coronal plane (same coordinates as in 2.5). The corresponding coronal cross-section at $t = t_1$ is displayed in the background. The origins of each of the displayed 2D displacement vectors are separated from each other by 6 voxels.

the variation in the mean error. σ_{LK} and n_{layers} are the two parameters that have the highest influence on the minimum registration error, and this emphasizes the importance of using more than one layer when performing lung image registration. The minimum registration error varied with σ_{LK} from 68.5 to 55.8 for patient 1, from 86.4 to 33.0 for patient 2, from 45.9 to 36.7 for patient 3, and from 44.7 to 33.8 for patient 4. In other words, optimizing σ_{LK} led to a 31.3% average decrease in the minimum registration error. Similarly, carefully selecting n_{layers} led to a 36.2% average decrease in the minimum registration error.

The deformation vectors in the lungs mainly point downwards during inspiration and upwards during expiration (Fig. 2.10). The trajectories of the selected points of each patient also reflect the up and down motion of the lung structures (Fig. 2.11). These points move predominantly along the z-direction (spine axis) but other directions can be non-negligible. Marker 3 of patient 3 is the only marker for which the motion in the z-direction is not the most significant. Indeed, its motion amplitude during one breathing cycle in the y-direction (dorsoventral direction) is approximately equal to 6.5mm, whereas it is approximately equal to 3.5mm along

| Patient 1 sagittal | Patient 2 sagittal | Patient 3 sagittal | Patient 4 sagittal |

| Patient 1 coronal | Patient 2 coronal | Patient 3 coronal | Patient 4 coronal |
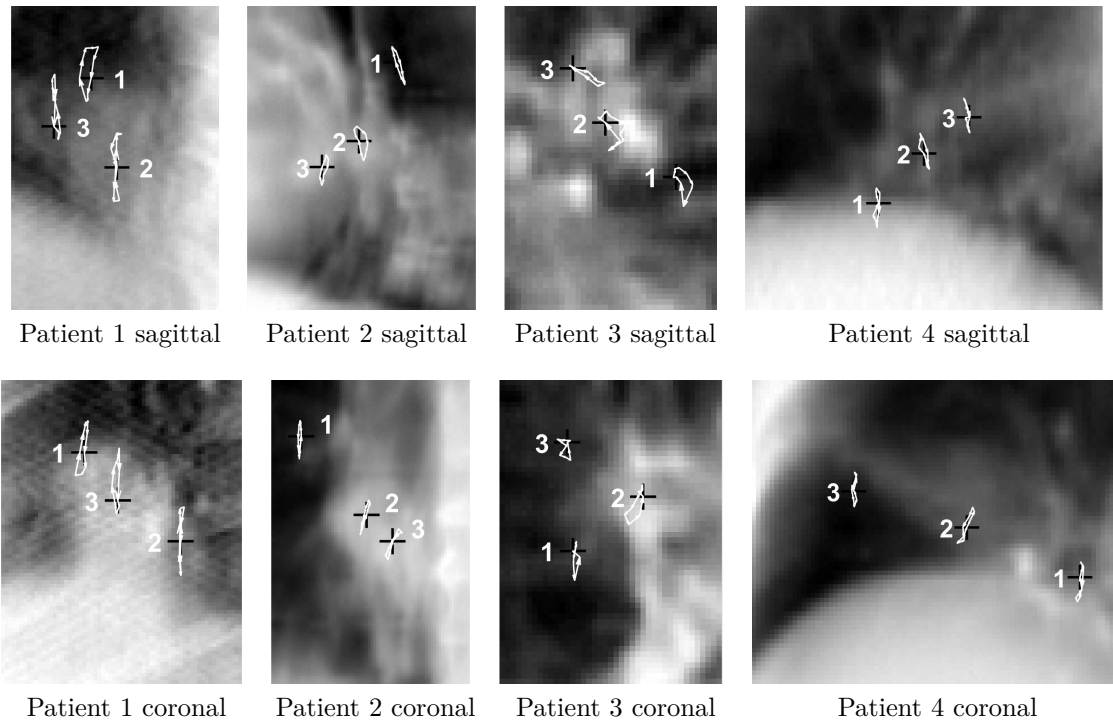
Figure 2.11: Trajectories of the internal points between $t = t_1$ and $t = t_{10}$ for each patient, calculated using the pyramidal Lucas-Kanade optical flow algorithm and displayed on top of the AIP of the ROI at $t = t_1$. The positions of these internal points at $t = t_1$ are denoted by black cross markers.

the z-direction (Fig. 2.14).

The optical flow algorithm optimized on the first breathing cycle (10 images) captured relatively well the z component of the motion, including the artificial drift, on the entire sequence of 2,400 images, despite the added noise (Fig. 2.12).

Patient number	1	2	3	4
Marker 1	19.4	19.7	15.8	13.2
Marker 2	22.7	17.5	17.7	13.7
Marker 3	21.9	16.7	12.0	13.6

Table 2.2: Amplitude of the motion of the selected internal points, in mm, between $t = t_1$ and $t = t_{2400}$

2.3.2 Prediction of the position of internal points

The parameters intervening in the RTRL learning algorithm have also been optimized by grid search, with the following range of parameters :

- gradient threshold $\tau \in \{0.5, 1.0, 2.0\}$

- learning rate $\eta \in \{0.01, 0.02, 0.05, 0.10\}$

46
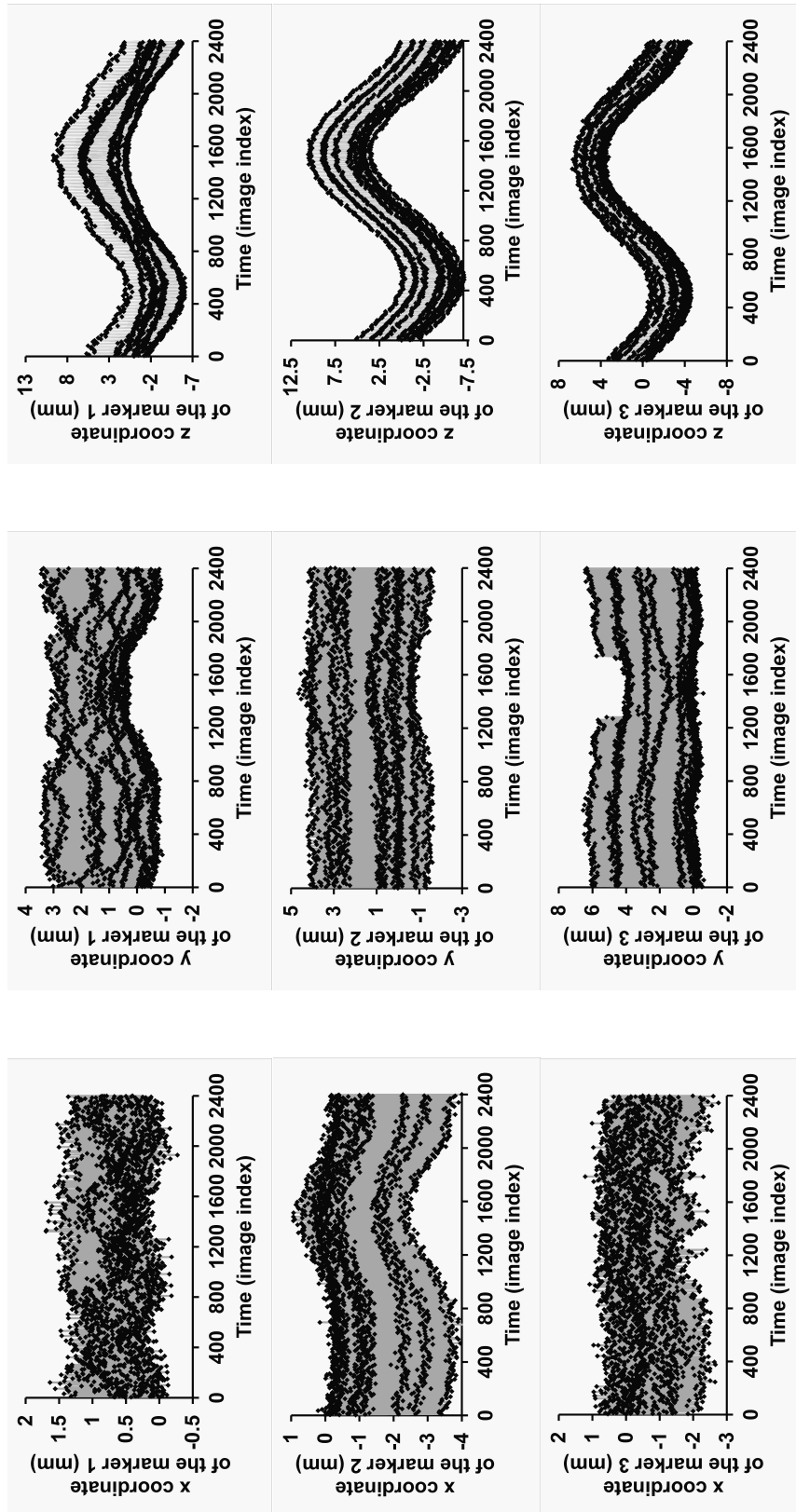
Figure 2.12: Motion of the markers of patient 3. The dot at time t corresponds to the signal sampled at time t. The axes are the same as in Fig. 2.11. The data is divided into 3 sets, namely the training set, between $t = t_1$ and $t = t_{2000}$, the cross-validation set, between $t = t_{2001}$ and $t = t_{2200}$, and the test set, between $t = t_{2201}$ and $t = t_{2400}$.
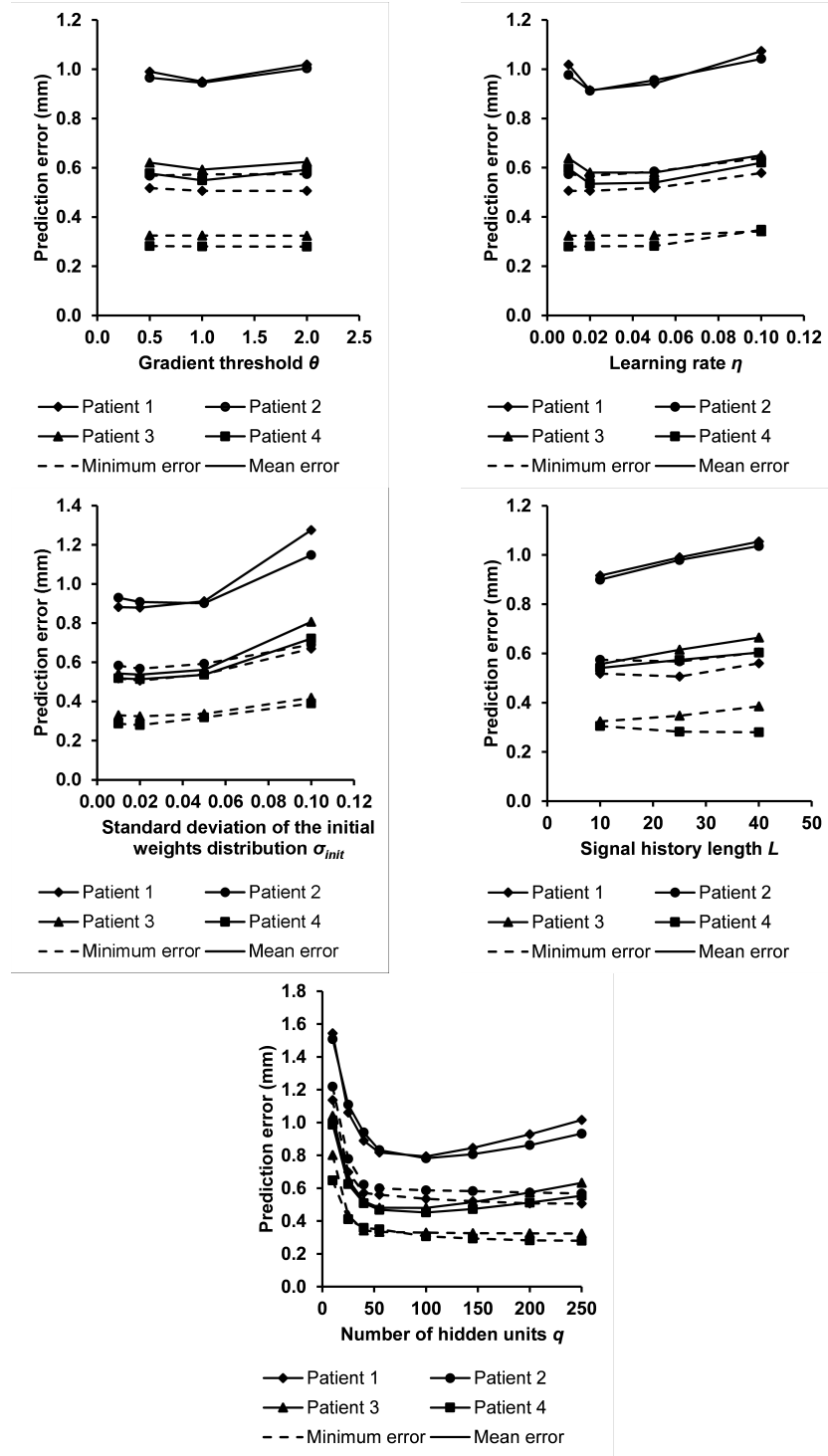
Figure 2.13: Prediction error e_{MAE} calculated on the cross-validation set between $t = t_{2001}$ and $t = t_{2200}$, as a function of the RNN parameters (Eq. 2.11).

- weights std. deviation $\sigma_{init} \in \{0.01, 0.02, 0.05, 0.10\}$

- signal history length $L \in \{10, 25, 40\}$

- nb. of hidden units $q \in \{10, 25, 40, 55, 100, 145, 200, 250\}$

Fig. 2.13 details how the prediction mean average error (MAE) on the cross-validation set between t_{2001} and t_{2200}, defined in Eq. 2.11, is affected by the choice
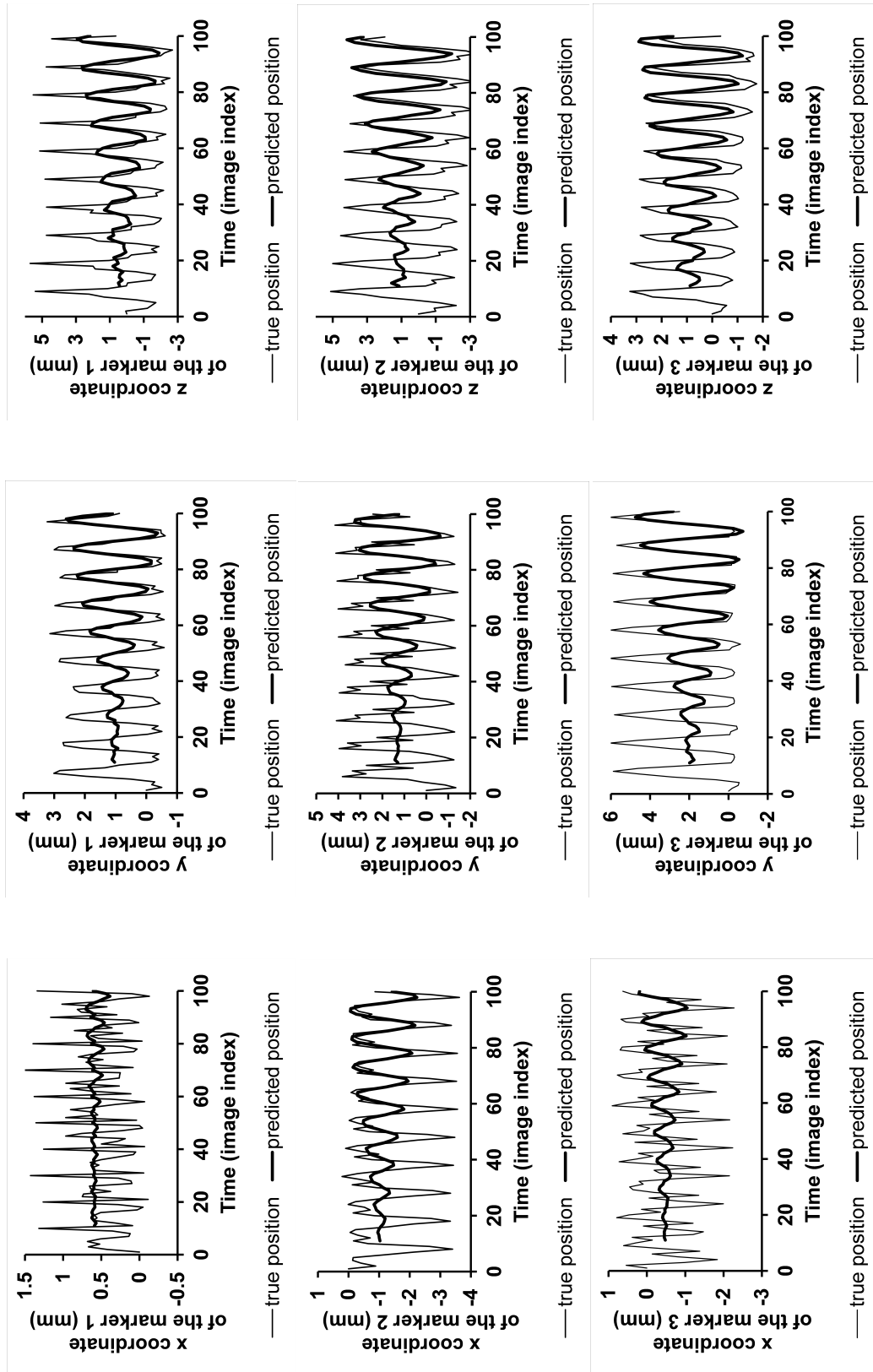
Figure 2.14: RNN training for predicting the position of marker 1 of patient 3, displayed between $t = t_1$ and $t = t_{100}$. The axes are the same as in Fig. 2.11.

of these parameters.

$$e_{MAE} = \frac{1}{200r} \sum_{k=2001}^{2200} \sum_{p=1}^{r} \left\| \overrightarrow{M_{true}^p(t_k)M_{pred}^p(t_k)} \right\|_2 \qquad (2.11)$$

In this equation, $M_{true}^p(t_k)$ is the 3D position of the pth marker at the instant t_k, calculated by the optical flow registration algorithm, $M_{pred}^p(t_k)$ is the predicted position of that marker at the same instant, and $\| \cdot \|_2$ refers to the euclidean norm. In order to take into consideration the random initialization of the initial synaptic weights, the MAE was averaged over 10 runs. Each graph in Fig. 2.13 describes the influence of one parameter and for each graph, two types of errors are displayed. The first one is the mean error: the MAE averaged over all the other parameters not studied in the graph. The second one is the minimum error: the minimum of the MAE across all the parameters. The mean prediction error as a function of η presents a bell shape. Both errors are maximum for $\eta = 0.10$ and we found the lowest minimum errors for $\eta = 0.01$ or $\eta = 0.02$, depending on the patient index. The mean error varies with σ_{init} from 1.27mm to 0.88mm for patient 1, from 1.14mm to 0.90mm for patient 2, from 0.81mm to 0.54mm for patient 3, and from 0.72mm to 0.51mm for patient 4. In other words, optimizing σ_{init} led to a 28.4% average decrease in the mean error. Both error curves are strictly convex because when the initial weights are too low, many time steps are required to grow them using the gradient descent updating rule, and when they are too high, they are difficult to control. Both errors were maximum for $\sigma_{init} = 0.1$ and attained their minimum for $\sigma_{init} = 0.02$, except the mean error of patient 2 which was minimized for $\sigma_{init} = 0.05$. The mean prediction error increases with the SHL, but the variation of the minimum error with the SHL was dependent on the patient index. Finally, the prediction error strongly decreases when q increases. The minimum error for $q = 10$, equal to 1.14mm, 1.22mm, 0.80mm, and 0.65mm respectively for patients 1,2,3 and 4, dropped down to 0.51mm, 0.57mm, 0.32mm, and 0.28mm for $q = 250$, which corresponds to a 56.3% error decrease on average. It is thus recommended to set a high value of q while keeping in mind that this may also result in a relatively high computing time. The mean error as a function of q is strictly convex and increases from $q = 100$ to $q = 250$.

The standard deviation of the mean prediction error and the minimum prediction error, relative to each parameter, is reported in Fig. 2.15. We observe that both σ_{init} and q are the parameters having the strongest impact on prediction accuracy. It would be interesting to evaluate the RNN trained with RTRL using less repetitive temporal data and reevaluate the importance of the SHL in that case.

The parameters that achieved the lowest (minimum) MAE error on the cross-validation set without leading to any numerical error have been used for evaluation on the test data between t_{2201} and t_{2400}. For every patient, we set $q = 250$ and
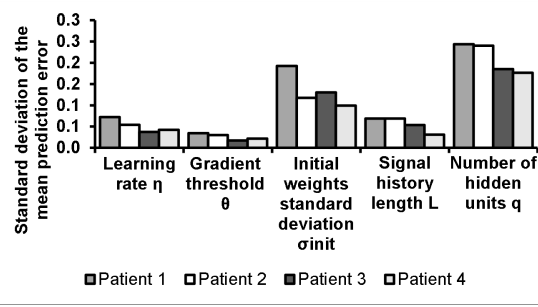
(a) Influence of each RNN parameter on the prediction mean error
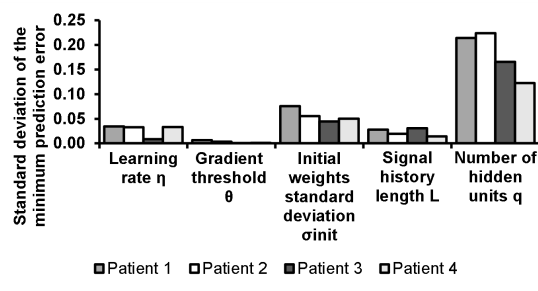
(b) Influence of each RNN parameter on the prediction minimum error

Figure 2.15: Relative influence of each of the RNN parameters on the prediction performance on the cross-validation set
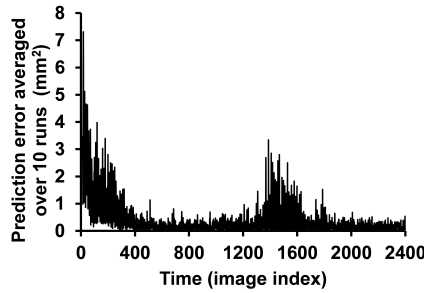
Figure 2.16: RNN loss function L_n on the normalized data for patient 3 (cf Eq. 2.6)

$\sigma_{init} = 0.02$. The value of η was set to 0.01 for patients 3 and 4, and 0.02 for patients 1 and 2. Table 2.3 shows the performance of the RNN on that test data, using the parameters selected as mentioned beforehand, in terms of the maximum prediction error, root-mean-square error (RMSE), and normalized root-mean-square error (nRMSE), defined respectively in Eq. 2.12, Eq. 2.13 and Eq. 2.14. In Eq. 2.14, μ_{true}^{p} designates the mean position of all observations of point p on the test set.

$$e_{max} = \max_{k=2201,...,2400} \max_{p=1,...,r} \left\| \overrightarrow{M_{true}^{p}(t_k)M_{pred}^{p}(t_k)} \right\|_2 \qquad (2.12)$$

$$e_{RMS} = \sqrt{\frac{1}{200r} \sum_{k=2201}^{2400} \sum_{p=1}^{r} \left\| \overrightarrow{M_{true}^{p}(t_k)M_{pred}^{p}(t_k)} \right\|_2^2} \qquad (2.13)$$

51
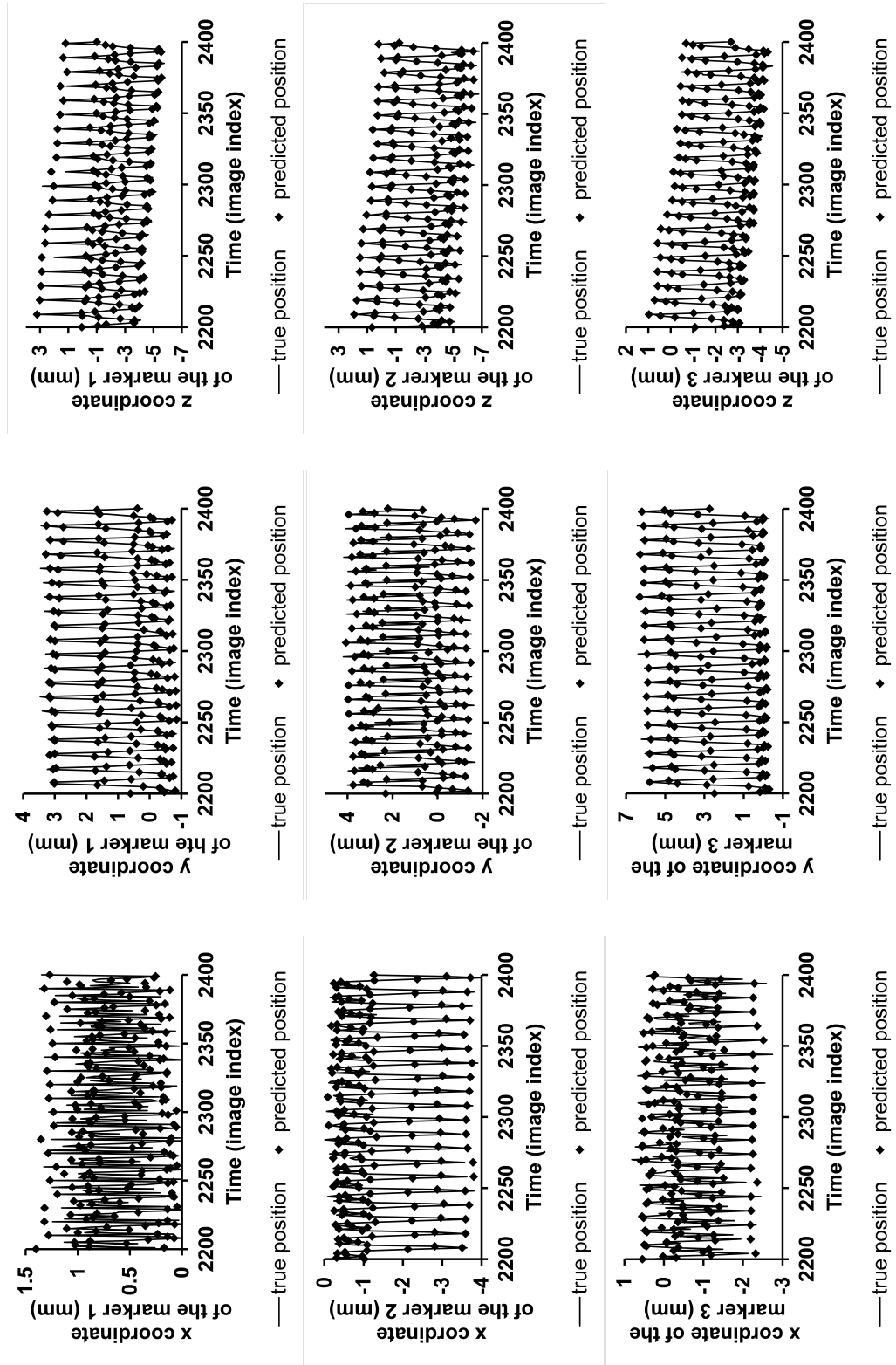
Figure 2.17: Prediction of the position of marker 1 of patient 3 on the test data. The axes are the same as in Fig. 2.11.

Error type	Prediction method	Patient 1	Patient 2	Patient 3	Patient 4	Error averaged over the 4 patients
Max error (in mm)	RTRL	1.82 ± 0.06	1.65 ± 0.04	1.16 ± 0.03	1.42 ± 0.06	1.51
	Lin. reg.	1.96	2.30	1.65	1.30	1.80
	LMS	2.07	1.69	1.40	1.21	1.59
	No prediction	9.11	5.98	4.66	4.60	6.09
RMSE (in mm)	RTRL	0.529 ± 0.005	0.585 ± 0.003	0.338 ± 0.002	0.324 ± 0.002	0.444
	Lin. reg.	0.512	0.610	0.333	0.341	0.449
	LMS	0.595	0.661	0.360	0.344	0.490
	No prediction	4.29	3.23	2.25	2.08	2.96
nRMSE (no unit)	RTRL	0.0829 ± 0.0007	0.118 ± 0.001	0.121 ± 0.001	0.109 ± 0.001	0.108
	Lin. reg.	0.080	0.124	0.121	0.115	0.110
	LMS	0.0932	0.133	0.129	0.116	0.118
	No prediction	0.671	0.651	0.807	0.701	0.708
Jitter (in mm)	RTRL	3.72	2.83	1.96	1.86	2.59
	Lin. reg.	3.69	2.86	1.98	1.82	2.59
	LMS	3.74	2.91	2.01	1.88	2.63
	No prediction	3.76	2.96	2.03	1.87	2.66

Table 2.3: RNN prediction performance computed on the test data, between $t = t_{2201}$ and $t = t_{2400}$, in comparison with other methods. Each cell indicates the maximum error, RMSE, nRMSE or jitter associated with the prediction of the position of the markers (Eq. 2.12, Eq. 2.13, Eq. 2.14 and Eq. 2.15). The error and 95% mean confidence interval mentioned for the RNN are calculated using 10 random initializations and assuming that the error distribution is Gaussian (Eq. 2.16 and Eq. 2.17). The confidence half range associated with the jitter measure has not been provided as the former is low compared with the latter (order of magnitude 10^{-3} mm).

$$e_{nRMS} = \frac{\sqrt{\sum_{k=2201}^{2400} \sum_{p=1}^{r} \left\| \overrightarrow{M_{true}^p(t_k) M_{pred}^p(t_k)} \right\|_2^2}}{\sqrt{\sum_{k=2201}^{2400} \sum_{p=1}^{r} \left\| \overrightarrow{M_{true}^p(t_k) \mu_{true}^p} \right\|_2^2}} \tag{2.14}$$

Furthermore, we evaluated the jitter of each prediction method on the test data. Jitter measures how oscillatory the predicted signal is (Eq. 2.15). Prediction with low jitter is desirable since it makes control of the treatment robot easier. The jitter measure J is minimized when the prediction is constant, thus there is a trade-off between accuracy and jitter.

$$J = \frac{1}{199r} \sum_{k=2201}^{2399} \sum_{p=1}^{r} \left\| \overrightarrow{M_{pred}^p(t_{k+1}) M_{pred}^p(t_k)} \right\|_2 \tag{2.15}$$

Because the RNN is evaluated using 10 runs with random weight initialization, not only the errors e_{max} and e_{RMS} are calculated, but also the corresponding 95% mean confidence intervals I_{max} and I_{RMS} (assuming that both e_{max} and e_{RMS} follow

a Gaussian distribution) defined in Eq. 2.16 and Eq. 2.17, where σ_{max} and σ_{RMS}
are the corresponding standard deviations of e_{max} and e_{RMS} over the 10 runs[5].

$$I_{max} = \left[e_{max} - \frac{1.96\sigma_{max}}{\sqrt{10}}, e_{max} + \frac{1.96\sigma_{max}}{\sqrt{10}} \right] \tag{2.16}$$

$$I_{RMS} = \left[e_{RMS} - \frac{1.96\sigma_{RMS}}{\sqrt{10}}, e_{RMS} + \frac{1.96\sigma_{RMS}}{\sqrt{10}} \right] \tag{2.17}$$

The performance of the RNN was compared with the point-wise and coordinate-wise linear predictor defined in Eq. 2.18. In that equation, p is the tracked point index, d represents the x,y, or z component of the 3D displacement $\vec{u}(\vec{x_p}, t)$, $(a_k^{d,p})$ are regression constants, and L_{lin} is the SHL, arbitrarily set to $L_{lin} = 10$. We also compared the RNN with the least mean squares (LMS) filter [Haykin, 2008] (Algorithm 3), for which we selected an SHL of $L_{LMS} = 10$ and a learning rate $\eta_{LMS} = 0.01$. The time series input data for the LMS algorithm was also normalized as described in Section 2.2.3.

$$u_d^{pred}(\vec{x_p}, t_{n+L_{lin}}) = a_0^{d,p} + \sum_{k=1}^{L_{lin}} a_k^{d,p} u_d(\vec{x_p}, t_{n+L_{lin}-k})$$
$$d = x, y, z \quad p = 1, 2, 3 \tag{2.18}$$

Algorithm 3 Least Mean Squares

1: **Parameters** :
2: L : signal history length
3: r : number of internal points considered
4: $m = 3rL$ dimension of the input space
5: $p = 3r$ dimension of the output space
6: η : learning rate
7:
8: **Initialization**
9: $W_{n=1} = 0_{p \times (m+1)}$
10:
11: **Learning and prediction**
12: **for** $n = 1, 2, ...$ **do**
13: $\quad y_n := W_n u_n$ (prediction)
14: $\quad W_{n+1} := W_n + \eta(y_n^* - y_n)u_n^T$ (weights update)

The RNN achieves a lower maximum and root-mean-square (RMS) prediction error as well as a lower jitter (averaged over the 4 patients) than linear prediction and LMS (Table 2.3). In particular, the maximum prediction error corresponding to the RNN, averaged over the 4 patients, equal to 1.51mm, is respectively 16.1% and 5.0% lower than the maximum error corresponding to linear prediction and LMS, equal to 1.80mm and 1.59mm. Furthermore, the averaged maximum prediction error and the averaged RMSE given by the RNN are respectively approximately 4 times and 7 times lower than the corresponding errors given by a system without prediction,

defined by $\overrightarrow{u_{pred}}(\cdot, t_{n+1}) = \vec{u}(\cdot, t_n)$. The prediction errors are higher for patients 1 and 2, which correlates with the overall highest motion amplitude of these patients' markers (Table 2.2). Concerning prediction with the RNN, the maximum tracking error for each patient is below the 2mm threshold recommended by Murphy and Lee et al.. By contrast, the maximum error with linear prediction corresponding to patient 2 and maximum error with LMS corresponding to patient 1 exceeded that threshold.

Prediction algorithm	Calculation time per time step (in ms)
RNN with RTRL	119.1
Linear regression	0.0052
LMS	0.318

Table 2.4: Time performance of the RNN in comparison with other prediction methods (Dell Intel Core i9-9900K 3.60GHz CPU NVidia GeForce RTX 2080 SUPER GPU 32Gb RAM with Matlab).

The average calculation time per time step (time for performing one prediction) of the RNN obtained with graphics processing unit (GPU) programming was equal to 119.1ms (Dell Intel Core i9-9900K 3.60GHz CPU NVidia GeForce RTX 2080 SUPER GPU 32Gb RAM with Matlab), which is lower than the marker position sampling time, approximately equal to 400ms (Table 2.4).

The prediction errors corresponding to the RNN in our study are lower than most of those reported in related studies about prediction in radiotherapy (Table 1.1). However, comparison with the prediction methods in the literature is difficult because the datasets, sampling rates, and look-ahead time vary between the studies. The regularity of the breathing motion as well as the low motion amplitudes in our dataset are factors that may have contributed to lower prediction errors in our research.

During the beginning of the learning process, the predicted values oscillate around the mean position signal and adjust progressively to reach the actual signal (Fig. 2.14). This is illustrated by the loss function decreasing for small values of the time index (Fig. 2.16). The error loss function of patient 3 rises again between t_{1200} and t_{1800} when variations in the marker motion pattern appear (cf Fig. 2.12). The predicted values on the test data follow closely the original motion signal. The breathing drift, corresponding to a decreasing trend in the z position of the markers on the test data for patient 3, is also well captured by the RNN (Fig. 2.17).

2.3.3   Chest image prediction

We chose the window size $h_w = 3$ and the standard deviation $\sigma_w = 0.5$ based on the visual quality of the resulting images, to warp $I(\cdot, t_1)$ using the Nadaraya-Watson
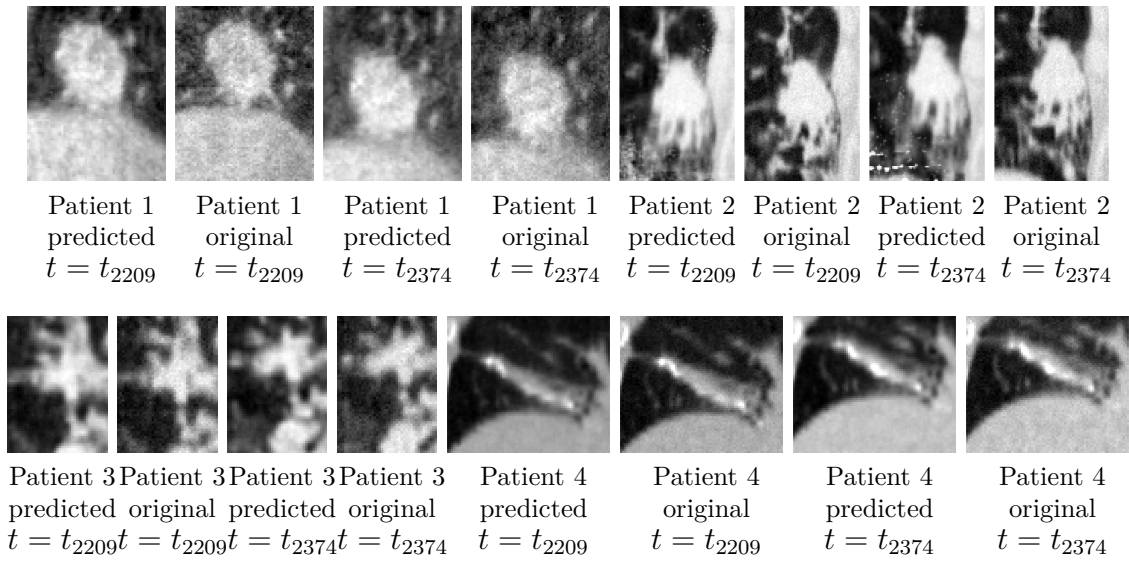
Figure 2.18: Original and predicted ROI coronal cross-sections (same coordinates as in Fig. 2.5), at $t = t_{2209}$ (end of expiration for patient 1 and end of inspiration for the other patients) and $t = t_{2374}$ (opposite case).
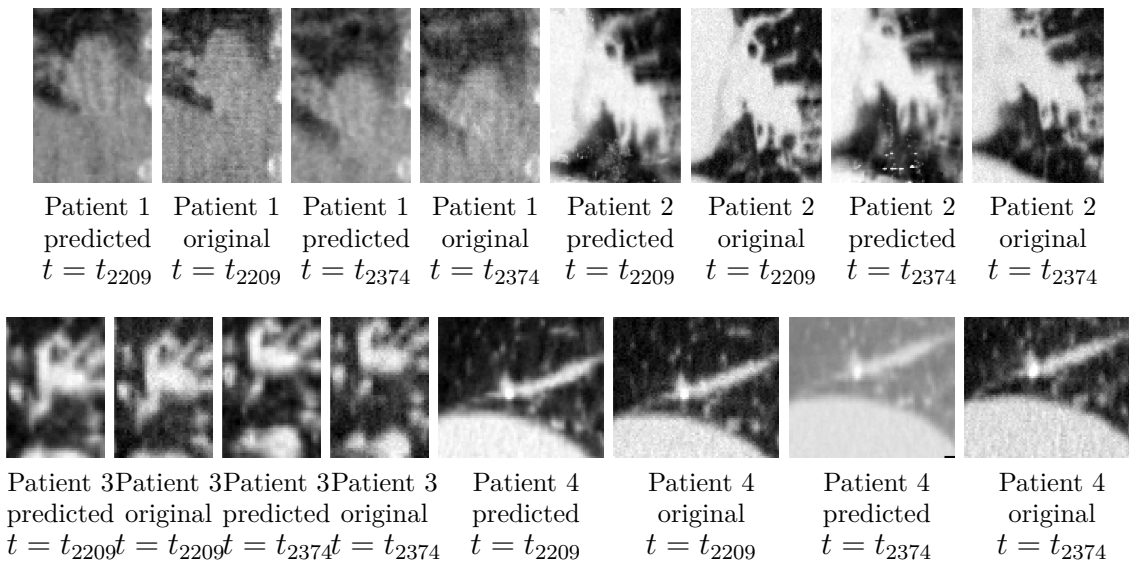
Figure 2.19: Original and predicted ROI sagittal cross-sections (same coordinates as in Fig. 2.5), at an end-of-exhale and an end-of-inhale positions. The predicted image at $t = t_{2374}$ for patient 4 seems to have high voxel intensity values but this is in fact due to post-processing with contrast enhancement, which takes into account the black voxels appearing on the lower right corner when displaying the image.

estimator (Appendix ??). The position of the tumor on the predicted images is almost the same as on the initial images (Fig. 2.18). The predicted images are less noisy due to the Gaussian filtering inherent to the warping process. However, some structures like blood vessels may have an unclear or imprecise position, or even be absent in the predicted images, such as the vessel on the bottom left of the tumor

DVF used for warping	Patient 1	Patient 2	Patient 3	Patient 4	Average
Initial DVF	0.960	0.982	0.976	0.992	0.978
DVF from markers	0.923	0.957	0.959	0.979	0.954
Predicted DVF from markers	0.923	0.958	0.959	0.978	0.955

Table 2.5: Precision of the DVF calculated at each step of the image prediction process. Each cell in the table corresponds to the cross-correlation between the initial ROI images, that is, the images from the sequence constructed in Section 2.2.1, and the warped initial image at $t = t_1$, averaged over the test data. The first line corresponds to the average for $k \in \{2201, ..., 2400\}$ of the cross-correlation between the initial image at time t_k and the initial image at time $t = t_1$ warped with the DVF directly calculated using the optical flow algorithm (cf section 2.2.2). The second line corresponds to the average for $k \in \{2201, ..., 2400\}$ of the mean cross-correlation between the initial image at time t_k, and the initial image at time t_1 warped with the DVF calculated from the markers' position using the linear correspondence model (Eq. 2.7), without prediction. Most importantly, the last line corresponds to the mean cross-correlation between the predicted and the initial images.

in the predicted coronal cross-section of patient 1 at $t = t_{2209}$ (Fig. 2.18). Artifacts consisting of trails of white dots and blurring appeared below the tumor of patient 2. These white trails may have appeared due to an inexact DVF and target voxels with only one antecedent voxel in the initial image at t_1. Moreover, points without antecedent voxels appeared for patient 4 (lower right corner of the sagittal cross-section at the end-of-exhale point), which resulted in voxels impainted in black by default.

The efficiency of the proposed image prediction algorithm is confirmed by the high cross-correlation between the predicted and original images averaged over the test data and the four patients, equal to 0.955 (Table 2.5). We also observe from Table 2.5 that the step most hampering the image prediction process is not the prediction of the markers' location, but the reconstruction of the entire DVF from the linear correspondence model (cf Eq. 2.7). We chose a simple correspondence model because it is not the main focus of the study. However, this model can be improved to take into account effects such as hysteresis and phase offset [Ehrhardt et al., 2013].

2.4 Conclusion

This is the first study of RNNs trained with RTRL for latency compensation in lung cancer radiotherapy, to the extent of our knowledge. RNNs are ANNs that are well suited for time-series prediction and the RTRL online learning method enables the predictor to continuously adapt to changes in the patient's respiratory patterns.

Gradient clipping was performed to minimize the likelihood of a numerical error while continually updating the synaptic weights. The image data used in this study consisted of four patients' temporal series of 10 3D chest CT scan images. Each of them was extended artificially into a series of 2,400 images by simulating the natural drift process while breathing. The sampling time is equal to approximately 400ms. Comparatively, it has been reported that the time delay of radiotherapy treatment systems ranges from 100ms up to 2s. The positions of internal points near the tumor of lung cancer patients, derived from the Lucas-Kanade pyramidal and iterative optical flow algorithm, were predicted with a 400ms response time. The amplitude of the motion of these points varied from 12.0mm to 22.7mm. The RMSE, maximum error, and jitter on the test set were all smaller than the corresponding performance measures given by linear prediction and LMS. In particular, the maximum prediction error given by the RNN trained with RTRL was equal to 1.51mm, which is respectively 16.1% and 5.0% lower than the maximum prediction error given by linear prediction and LMS (Table 2.3). In comparison, the maximum error and RMSE resulting from the prediction with the RNN were respectively 4 times and 7 times lower than the same errors resulting from a system without prediction. Furthermore, when performing prediction with the RNN, the maximum tracking error for each patient was below the 2mm threshold suggested by Murphy. The average calculation time per time step of the RNN was equal to 119.1ms (Dell Intel Core i9-9900K 3.60GHz CPU NVidia GeForce RTX 2080 SUPER GPU 32Gb RAM with Matlab), which is lower than the marker position sampling time, equal to 400ms. Finally, we combined prediction of the position of internal points using the RNN with a linear correspondence model and forward-warping using Nadaraya-Watson non-linear regression to perform 3D chest image prediction. The mean cross-correlation between the initial and predicted images is equal to 0.955 (Table 2.5), and the overall tumor position in the predicted images appears to be visually correct.

This research gives valuable insight concerning proper parameter adjustment for maximizing prediction performance with RNNs trained with RTRL in the context of radiotherapy. We performed grid search and found that $q = 250$ hidden units and an initial standard deviation of the synaptic weights equal to $\sigma_{init} = 0.02$ were optimal on the cross-validation set for all patients. These two parameters were the parameters having the largest impact on the prediction error on the cross-validation set. Optimizing q and σ_{init} respectively led to a decrease of 56.3% and 28.4% in the MAE. The minimum prediction error is a convex function of σ_{init} and decreases when q increases. However, the general variation of that prediction error as a function of the SHL was different from patient to patient, hence the optimal value of the SHL also varied among the patients.

This is also the first detailed study of the pyramidal iterative Lucas-Kanade optical flow algorithm applied to lung CT scan images providing details about the precise

influence of each parameter on the registration error. The pyramidal iterative Lucas-Kanade optical flow is a classical DIR algorithm, but proper parameter adjustment, which is key to ensure high accuracy of the deformation field, had not been discussed in detail in previous studies related to registration of CT scan images, to the extent of our knowledge. In this work, we provided experimental results about parameter selection for performance optimization. σ_{LK} and n_{layers} were the parameters having the most significant impact on registration performance. Carefully selecting σ_{LK} and n_{layers} respectively led to a decrease in the minimum registration error of 31.3% and 36.2%. With our dataset, we found optimal results with $\sigma_{LK} = 2.0$ and $n_{layers} = 3$ or $n_{layers} = 4$. It was confirmed that using only one layer was hampering the registration performance, which correlates with the observations in [Xu et al., 2008, Zhang et al., 2008]. This is due to the high amplitude of the lung motion in the CT scan images used, relative to the image resolution.

This study is a step forward in lung radiotherapy because better compensation of the treatment system latency will entail more accurate tumor targeting. In addition, it will enable reducing the radiation margin around the tumor for compensation of unexpected motion, leading thus to a decrease in the irradiation of surrounding healthy tissue, and in turn to less undesirable side effects such as radiation pneumonitis. Further research about prediction of more irregular breathing patterns will bring more insights into the capabilities of online learning methods such as RTRL to adapt to unexpected temporal events. This study of the RTRL algorithm could be further enriched by investigating the variation of the prediction performance as a function of the prediction horizon. Finally, we could extend this work by tracking more accessible surrogate signals such as points on the diaphragm recorded using kilovoltage (kV) imaging, or external markers placed on the skin [Ehrhardt et al., 2013].

Chapter 3

Prediction of the position of external markers using Unbiased Online Recurrent Optimization

3.1 Introduction

3.1.1 External markers in lung cancer radiotherapy

During lung cancer radiotherapy, respiratory motion makes tumor targeting difficult. Indeed, tumors can move up to 5cm within the same treatment fraction due to breathing [Chen et al., 2001]. Respiratory motion is largely cyclic but exhibits variations in frequency and amplitude, shifts and drifts, as well as differences across patients and fractions [Ehrhardt et al., 2013, Verma et al., 2010]. The term "shift" designates abrupt changes of the respiratory signal, whereas "drift" designates continuous variations. To overcome this problem, the position of external markers placed on the chest and abdomen can be recorded by infrared cameras (e.g. Cyberknife system [Khankan et al., 2017] in Fig. 3.1). By using an appropriate correspondence model, these positions may be correlated to the three-dimensional (3D) position and shape of the tumor for accurate irradiation [Ehrhardt et al., 2013, McClelland et al., 2013].

Radiotherapy treatment machines are subject to a time latency due to communication delays, robot control, and radiation system delivery preparation. It was reported that "for most radiation treatments, the latency will be more than 100ms, and can be up to two seconds" [Verma et al., 2010]. Delay compensation is necessary to minimize excessive damage to healthy tissues (Fig. 1.7), and to achieve this, various prediction methods have been proposed (see for instance [Jöhl et al., 2020,

---

This chapter has been published as a preprint on Arxiv [Pohl et al., 2021b].
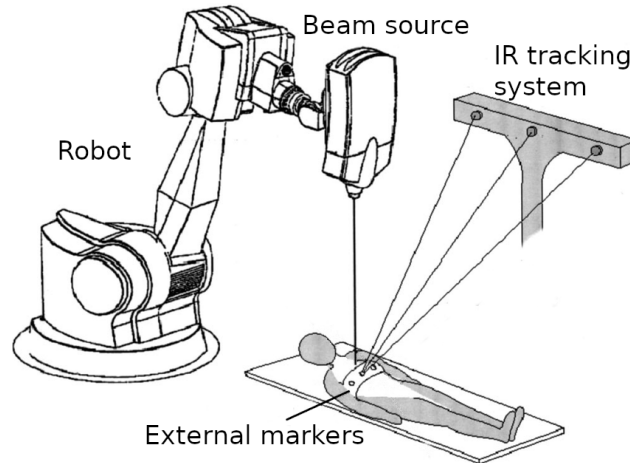
Figure 3.1: Radiotherapy treatment system (Cyberknife) using external markers to guide the irradiation beam[1].

Lee and Motai, 2014, Verma et al., 2010]). Our research investigates the feasibility to predict breathing motion with online training algorithms for recurrent neural networks (RNN). In contrast to offline methods, online methods update the synaptic weights with each new training example. That enables the neural network to adapt to the continuously changing breathing patterns of the patient, therefore providing robustness to complex motion.

3.1.2 Estimating internal respiratory motion using correspondence models

This chapter focuses on the prediction of surrogate respiratory signals (the position of external markers), but in this section, we provide general background information concerning respiratory correspondence models and the clinical workflow associated with the use of external markers. Respiratory correspondence models are statistical models that relate surrogate signals to the motion of internal anatomy. They are used when the target is not imaged directly during the treatment, as respiratory signals can provide information that can help estimate tumor position. Surrogate signals can be measurements of the volume or flow of air being inhaled or exhaled by the patient using a spirometer [Wilms et al., 2014], internal anatomy measurements (e.g., the motion of points on the diaphragm [Zhang et al., 2007], or full two-dimensional (2D) images [Stemkens et al., 2016]), as well as displacement of the surface of the chest and abdomen. The latter can be estimated using systems that measure the position infra-red markers such as the Real-Time Position Management (RPM) system (Varian), the Cyberknife system (Accuray), or the ExacTrac system (Brainlab). One can also use laser tracking systems [Seppenwoolde et al.,

---

[1]Adapted from [Schweikard et al., 2004] with permission from Wiley, Copyright 2004 American Association of Physicists in Medicine.

2007] or respiratory belts around the patient's chest (Anzai respiratory system from Siemens). Alternatively, the entire surface of the chest skin can be considered as a surrogate signal. In that scenario, stereo imaging techniques such as the AlignRT system from VisionRT [Seppenwoolde et al., 2007] can be employed to obtain surface information in real-time. The surface model is subsequently built by extracting the skin surface from computed tomography (CT) or magnetic resonance (MR) imaging. In Chapter 2 Sections 2.2.4 and 2.3.3, we used a linear correspondence model to link the motion of a few internal points to the dense displacement vector field (DVF) in a region of interest (ROI) containing the tumor (Eq. 2.7).

The process involved in the use of a motion correspondence model comprises two main steps. First, before the treatment, surrogate respiratory signals (in this chapter, the motion of external markers) are acquired simultaneously with imaging data. In the case of a surrogate signal based on chest surface motion, the imaging data itself is sufficient. For instance, the location where the external markers will be placed can be tracked directly using the volumetric images. The motion of the internal anatomy is measured from the imaging data using deformable image registration (DIR) (e.g., the Lucas-Kanade optical flow as presented in Chapter 2 Sections 2.1.3 and 2.2.2). Statistical modeling is performed to estimate the relationship between the internal motion and the surrogate signal (Fig. 3.3). Second, that relationship is used during the treatment phase to calculate the internal motion from the measured surrogate signal to guide gated or tracked treatments(Fig. 3.4).

Establishing statistical correlations between the surrogate signal and the target position is challenging for several reasons. First, the tumor trajectory is subject to hysteresis, i.e., it follows a different path between inhalation and exhalation. Also, there are phase offsets between the tumor and surrogate signals, which means there may be a time interval between the peaks of both signals. A simple linear model may thus be insufficient to accurately estimate the target position (cf the graphs corresponding to patients B and C in Fig. 3.5). The development of different mathematical tools for correspondence modeling is an active area of research. In this chapter, we will report geometrical errors related only to external marker position forecasting, but additional errors due to inaccuracies in the correspondence model also need to be considered in further comprehensive studies (Fig. 3.2).
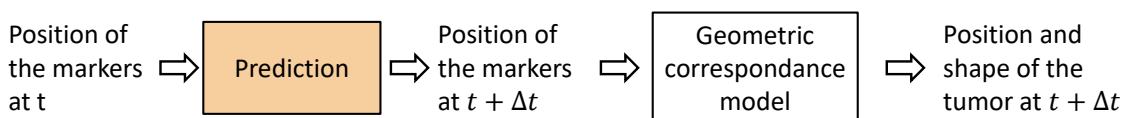
Figure 3.2: Deriving the future position of the tumor from external marker position measurements using a correspondence model. In this chapter, we focus exclusively on the prediction step to compensate for the latency of treatment systems.
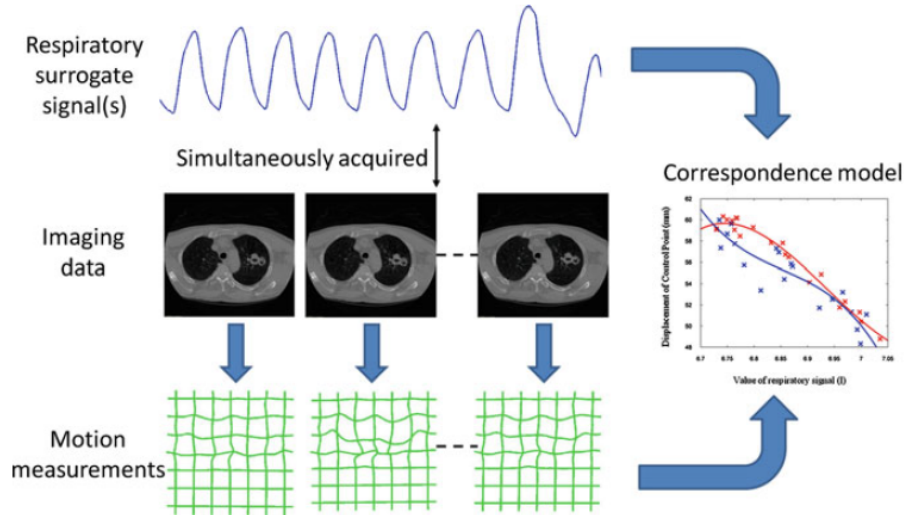
Figure 3.3: Correspondence model building process[2].
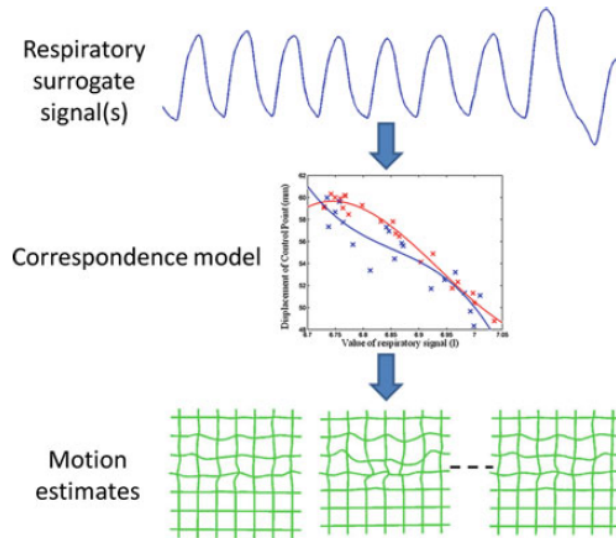
Figure 3.4: Correspondence model usage during the radiotherapy treatment.[3]

3.1.3   Contributions of this study

Many techniques for online training of RNNs have recently emerged [Marschall et al., 2020], such as unbiased online recurrent optimization (UORO) [Tallec and Ollivier, 2017a]. Most of these seek to approximate real-time recurrent learning (RTRL) [Haykin et al., 2009, Williams and Zipser, 1989], which suffers from a large computational complexity. They also aim to provide an unbiased estimation of the gradient estimates, that truncated back-propagation through time (BPTT) [Haykin et al., 2009, Jaeger, 2002] cannot compute, therefore guaranteeing theoretical convergence

---

[2]Reprinted from [Ehrhardt et al., 2013] with permission from Springer Nature, Copyright 2013.

[3]Reprinted from [Ehrhardt et al., 2013] with permission from Springer Nature, Copyright 2013.

[4]Reprinted from [Isaksson et al., 2005] with permission from Wiley, Copyright 2005 American Association of Physicists in Medicine.

(a) Patient A

(b) Patient B
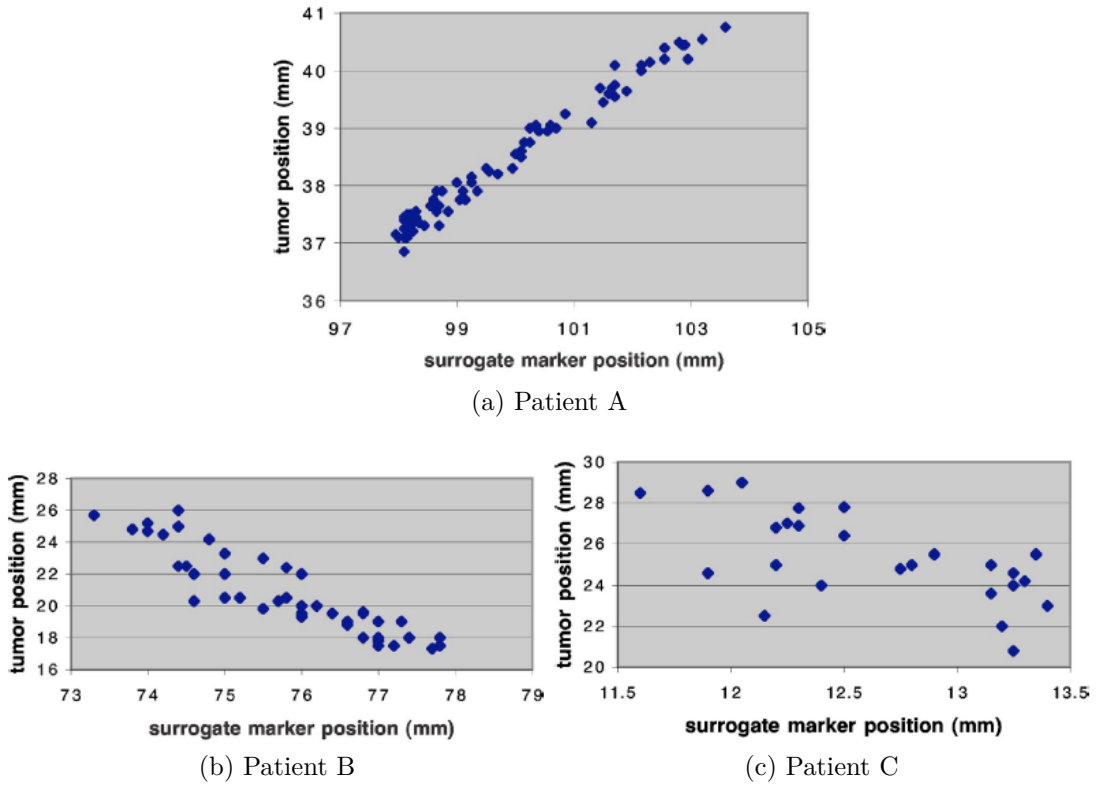
(c) Patient C

Figure 3.5: Correlation between tumor and external marker position reported for three patients in [Isaksson et al., 2005], plotted by sampling one data point every second. Patient A had a pancreatic tumor, and patients B and C had middle lobe lung tumors. Patients B and C exhibited irregular respiratory patterns associated with non-stationary correlations between the positions of the marker and the tumor. [4]

and appropriate balance between short-term and long-term temporal dependencies. The theoretical convergence of RTRL and UORO, which could not be proved by standard stochastic gradient descent theory, has recently been established [Massé and Ollivier, 2020].

This is the first study to evaluate the capabilities of RNNs trained online with UORO to predict the 3D position of external markers on the chest and abdomen for safety in radiotherapy. The proposed RNN model does not perform prediction for each marker separately but instead learns patterns about the correlation between their motion to potentially increase its forecasting accuracy. We compare UORO with different forecasting algorithms, namely RTRL, least mean squares (LMS), and linear regression, for different look-ahead values h, ranging from $h_{min} = 0.1s$ to $h_{max} = 2.0s$, by observing different prediction metrics as h varies. We divide the subjects' data into two groups: regular and irregular breathing, to quantify the robustness of each prediction algorithm. We analyze the influence of the hyper-parameters on the prediction accuracy of UORO as the horizon value changes and discuss the selection of the best hyper-parameters.

3.2 Materials and Methods

3.2.1 Marker position data

In this study, we use 9 records of the 3D position of 3 external markers on the chest and abdomen of individuals lying on a treatment couch (HexaPOD), acquired by an infrared camera (NDI Polaris). The duration of each sequence is between 73s and 320s and the sampling rate is 10Hz. The superior-inferior, left-right, and antero-posterior trajectories respectively range between 6mm and 40mm, between 2mm and 10mm, and between 18 mm and 45mm. In five of the sequences, the breathing motion is normal and in the four remaining sequences, the individuals were asked to perform actions such as talking or laughing. More details concerning the dataset can be found in [Krilavicius et al., 2016].

3.2.2 The RTRL and UORO algorithms for training RNNs

In this study, we train an RNN to predict the position of the 3 markers in the future (cf section 1.3). The instantaneous square loss L_n of the network can be calculated from the instantaneous error e_n between the correct output y_n^* and predicted output y_n (Eq. 3.1).

$$e_n = y_n^* - y_n \qquad L_n = \frac{1}{2}\|e_n\|_2^2 \tag{3.1}$$

By using the chain rule, one can derive Eqs. 3.2 and 3.3, which describe how changes of the parameter vector θ_n affect the instantaneous loss L_{n+1} and state vector x_{n+1}. Computation of the gradient of L_{n+1} with respect to the parameter vector using Eq. 3.2, followed by recursive computation of the influence matrix $\partial x_n/\partial\theta$ using Eq. 3.3 constitutes the RTRL algorithm. RTRL is computationally expensive, and UORO attempts to solve that problem by approximating the influence matrix with an unbiased rank-one estimator. In UORO, two random column vectors $\tilde{x}_n$ and $\tilde{\theta}_n$ are recursively updated at each time step so that $\mathbb{E}(\tilde{x}_n\tilde{\theta}_n^T) = \partial x_n/\partial\theta$ (Algorithm 4). It was reported that "UORO's noisy estimates of the true gradient are almost orthogonal with RTRL at each time point, but the errors average out over time and allow UORO to find the same solution" [Marschall et al., 2020] (Fig. 3.6).

$$\frac{\partial L_{n+1}}{\partial\theta} = \frac{\partial L_{n+1}}{\partial y}(y_{n+1})\left[\frac{\partial F_{out}}{\partial x}(x_n,\theta_n)\frac{\partial x_n}{\partial\theta} + \frac{\partial F_{out}}{\partial\theta}(x_n,\theta_n)\right] \tag{3.2}$$

$$\frac{\partial x_{n+1}}{\partial\theta} = \frac{\partial F_{st}}{\partial x}(x_n,u_n,\theta_n)\frac{\partial x_n}{\partial\theta} + \frac{\partial F_{st}}{\partial\theta}(x_n,u_n,\theta_n) \tag{3.3}$$

Algorithm 4 Unbiased Online Recurrent Optimization

1: **Parameters** :

2: $L \in \mathbb{N}^*$: signal history length, $n_M = 3$: number of external markers considered

3: $m = 3 n_M L$, $q \in \mathbb{N}^*$ and $p = 3 n_M$ dimension of the RNN input, state and output

4: $\eta \in \mathbb{R}_{>0}$ and $\tau \in \mathbb{R}_{>0}$: learning rate and gradient threshold

5: $\sigma_{init} \in \mathbb{R}_{>0}$: standard deviation of the Gaussian distribution of the initial weights

6: $\epsilon_{norm} = 1.10^{-7}, \epsilon_{prop} = 1.10^{-7}$

7:

8: **Initialization**

9: $W_{a,n=1}$, $W_{b,n=1}$, $W_{c,n=1}$ synaptic weight matrices of respective sizes $q \times q$, $q \times (m+1)$ and $p \times q$, initialized according to a Gaussian distribution with std. dev. σ_{init}.

10: *Notation :* $|W_a| = q^2$, $|W_b| = q(m+1)$, $|W_c| = pq$, and $|W| = q(p+q+m+1)$

11: $x_{n=1} := 0_{q \times 1}$: state vector, $\tilde{x}_{n=1} := 0_{q \times 1}, \tilde{\theta}_{n=1} := 0_{1 \times |W|}$: vectors such that $\partial x_n / \partial \theta \approx \tilde{x}_n \tilde{\theta}_n$

12: $\delta\theta := 0_{1 \times |W|}, \delta\theta_g = 0_{1 \times |W|}$ vectors defined by $\delta\theta = \dfrac{\partial L_{t+1}}{\partial y_{n+1}} \dfrac{\partial F_{out}}{\partial \theta}$ and $\delta\theta_g = \nu^T \dfrac{\partial F_{st}}{\partial \theta}$

13:

14: **Learning and prediction**

15: **for** $n = 1, 2, ...$ **do**

16: $\quad z_n := W_{a,n} x_n + W_{b,n} u_n$, $x_{n+1} := \Phi(z_n)$ (hidden state update)

17: $\quad y_{n+1} := W_{c,n} x_{n+1}$ (prediction), $e_{n+1} := y_{n+1}^* - y_{n+1}$ (error vector update)

18: $\quad [\delta\theta_{1+|W_a|+|W_b|}, ..., \delta\theta_{|W|}] := -[(e_{n+1} x_n^T)_{1,1}, ..., (e_{n+1} x_n^T)_{p,q}]$

19: $\quad \Delta\theta := -e_{n+1}^T W_{c,n} \tilde{x}_n \tilde{\theta}_n + \delta\theta$ (gradient estimate)

20: $\quad \nu$: column vector of size q with random values in $\{-1, 1\}$

21: $\quad \tilde{x}_{n+1} := \dfrac{\Phi[W_{a,n}(x_n + \epsilon_{prop}\tilde{x}_n) + W_{b,n} u_n] - x_{n+1}}{\epsilon_{prop}}$ (tgt forward propagation)

22: $\quad \delta\theta_g^{aux} := \nu * \Phi'(z_n)$ (element-wise or scalar product)

23: $\quad [(\delta\theta_g)_1, ..., (\delta\theta_g)_{|W_a|}] := [(\delta\theta_g^{aux} x_n^T)_{1,1}, ..., (\delta\theta_g^{aux} x_n^T)_{q,q}]$

24: $\quad [(\delta\theta_g)_{|W_a|+1}, ..., (\delta\theta_g)_{|W_a|+|W_b|}] := [(\delta\theta_g^{aux} u_n^T)_{1,1}, ..., (\delta\theta_g^{aux} u_n^T)_{q,m+1}]$

25:

26: $\quad \rho_0 := \sqrt{\dfrac{\|\tilde{\theta}\|_2}{\|\tilde{x}\|_2 + \epsilon_{norm}}} + \epsilon_{norm}, \quad \rho_1 := \sqrt{\dfrac{\|\tilde{\theta}_g\|_2}{\|\nu\|_2 + \epsilon_{norm}}} + \epsilon_{norm}$

27: $\quad \tilde{x}_{n+1} := \rho_0 \tilde{x}_{n+1} + \rho_1 \nu \quad \tilde{\theta}_{n+1} := \tilde{\theta}_n / \rho_0 + (\delta\theta_g)/\rho_1$

28: $\quad \theta_n := [(W_{a,n})_{1,1}, ..., (W_{a,n})_{q,q}, (W_{b,n})_{1,1}, ..., (W_{b,n})_{q,m+1}, (W_{c,n})_{1,1}, ..., (W_{c,n})_{p,q}]$

29: $\quad$ **if** $\|\Delta\theta\|_2 > \tau$ **then**

30: $\quad\quad \Delta\theta := \dfrac{\tau}{\|\Delta\theta\|_2} \Delta\theta$ (gradient clipping)

31: $\quad \theta_{n+1} := \theta_n - \eta \Delta\theta$ (weights update)

32: $\quad W_{a,n+1} := \begin{bmatrix} (\theta_{n+1})_1 & ... & (\theta_{n+1})_{q(q-1)+1} \\ ... & ... & ... \\ (\theta_{n+1})_q & ... & (\theta_{n+1})_{|W_a|} \end{bmatrix} W_{b,n+1} := \begin{bmatrix} (\theta_{n+1})_{|W_a|+1} & ... & (\theta_{n+1})_{|W_a|+qm+1} \\ ... & ... & ... \\ (\theta_{n+1})_{|W_a|+q} & ... & (\theta_{n+1})_{|W_a|+|W_b|} \end{bmatrix}$

33: $\quad W_{c,n+1} := \begin{bmatrix} (\theta_{n+1})_{|W_a|+|W_b|+1} & ... & (\theta_{n+1})_{|W_a|+|W_b|+p(q-1)+1} \\ ... & ... & ... \\ (\theta_{n+1})_{|W_a|+|W_b|+p} & ... & (\theta_{n+1})_{|W_a|+|W_b|+|W_c|} \end{bmatrix}$

34:

35: *Convention : for* $A \in \mathbb{R}^M \times \mathbb{R}^N$ *we note* $[A_{1,1}, ..., A_{M,N}] = [A_{1,1}, ..., A_{M,1}, A_{1,2}, ..., A_{M,N}]$

3.2.3 Online prediction of the position of the markers with a vanilla RNN

If we denote by $\vec{u}_j(t_k) = [u_j^x(t_k), u_j^y(t_k), u_j^z(t_k)]$ the normalized 3D displacement of marker j at time t_k, the input u_n of the RNN consists of the concatenation of the vectors $\vec{u}_j(t_n), ..., \vec{u}_j(t_{n+L-1})$ for each marker j, where L designates the signal
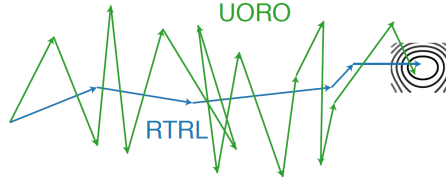
Figure 3.6: Weak alignment of the gradients of UORO with those of RTRL from the point of view of the loss optimization.[5]

history length (SHL), expressed here in number of time steps. The prediction of the displacement of the 3 markers is performed simultaneously to use information about the correlation between each of them. The output vector y_{n+1} consists of the position of these 3 points at time $t_{n+L+h-1}$, where h refers to the horizon value, expressed also in number of time steps (Eq. 3.4).

$$
u_n = \begin{pmatrix} 1 \\ u_1^x(t_n) \\ u_1^y(t_n) \\ u_1^z(t_n) \\ ... \\ u_3^z(t_n) \\ u_1^x(t_{n+1}) \\ ... \\ u_3^z(t_{n+L-1}) \end{pmatrix} \qquad y_{n+1} = \begin{pmatrix} u_1^x(t_{n+L+h-1}) \\ u_1^y(t_{n+L+h-1}) \\ u_1^z(t_{n+L+h-1}) \\ ... \\ u_3^z(t_{n+L+h-1}) \end{pmatrix} \qquad (3.4)
$$
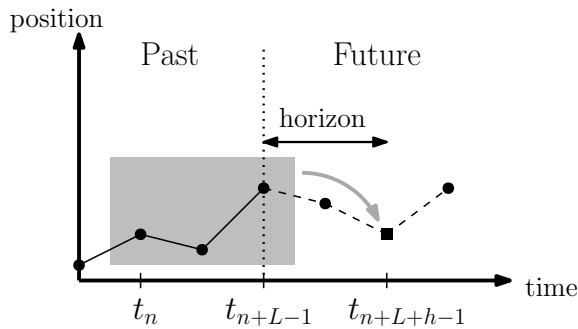
Figure 3.7: Forecasting a one-dimensional position signal. The signal history length L is the time interval in the past, the information of which is used for performing one prediction. The horizon h, also called response time or look-ahead time, is the time interval in advance for which the prediction is made (cf Eq. 3.4).

The RNN used in this chapter is a vanilla RNN (cf section 1.3). The implementation of the UORO algorithm in the case of the vanilla model described above is detailed in Algorithm 4 and the proof of the equations used are in Appendix **??**. The RNN characteristics are summarized in Table 1.3.

---

[5]Reprinted from [Marschall et al., 2020], Copyright JMLR 2020.

3.2.4   Experimental design

We compare RNNs trained with UORO with other prediction methods: RNNs trained with RTRL (Algorithm 2), LMS (Algorithm 3), and multivariate linear regression (Table 3.1). We clipped the gradient estimate of the instantaneous loss (Eq. 3.1) with respect to the parameter vector $\vec{\nabla}_\theta(L_n)$ for UORO, RTRL, and LMS when $\|\vec{\nabla}_\theta(L_n)\|_2 > \tau$ with the same threshold value $\tau = 2.0$ for these three algorithms.

Prediction method	Mathematical model	Training & cross-val.	Hyper-parameters' range for cross-val.
RNN UORO	$x_{n+1} = \Phi(W_{a,n}x_n + W_{b,n}u_n)$ $y_n = W_{c,n}x_n$	Training 30s cross-val. 30s	$\eta \in \{0.05, 0.1, 0.2\}$ $\sigma_{init} \in \{0.02, 0.05\}$ $L \in \{10, 30, 50, 70, 90\}$ $q \in \{10, 30, 50, 70, 90\}$
RNN RTRL	$x_{n+1} = \Phi(W_{a,n}x_n + W_{b,n}u_n)$ $y_n = W_{c,n}x_n$	Training 30s cross-val. 30s	$\eta \in \{0.02, 0.05, 0.1, 0.2\}$ $\sigma_{init} \in \{0.01, 0.02, 0.05\}$ $L \in \{10, 25, 40, 55\}$ $q \in \{10, 25, 40, 55\}$
LMS	$y_{n+1} = W_n u_n$	Training 30s cross-val. 30s	$\eta \in \{0.002, 0.005, 0.01,$ $\quad 0.02, 0.05, 0.1, 0.2\}$ $L \in \{10, 30, 50, 70, 90\}$
Linear regression	$y_{n+1} = W u_n$	Training 54s cross-val. 6s	$L \in \{10, 20, 30, 40, 50,$ $\quad 60, 70, 80, 90\}$

Table 3.1: Overview of the different forecasting methods compared in this study. The input vector u_n, corresponding to the positions in the past, and the output vector y_{n+1}, corresponding to the predicted positions, which appear in the second column, are defined in Eq. 3.4. The fourth column describes the hyper-parameter range for cross-validation with grid search. η refers to the learning rate, σ_{init} to the standard deviation of the initial Gaussian distribution of the synaptic weights, L to the SHL (expressed in number of time steps), and q to the number of hidden units. W_n and W are matrices used respectively in LMS and linear regression, and their size is $p \times (m + 1)$.

Learning is performed using only information from the sequence that is used for testing. Each time series is split into a training and development set of 1 min and the remaining test set. The training set comprises the data between 0s and 30s except in the case of linear regression as using more data is beneficial to offline methods. The hyper-parameters that minimize the root-mean-square error (RMSE) (Eq. 3.6) of the cross-validation set during the grid search process are selected for evaluation. The term $\delta_j(t_k)$ in Eq. 3.6 designates the instantaneous prediction error at time t_k due to marker j, defined in Eq. 3.5.

$$\delta_j(t_k) = \|\vec{u}_j^{pred}(t_k) - \vec{u}_j^{true}(t_k)\|_2 \tag{3.5}$$

$$RMSE = \sqrt{\frac{1}{3(k_{max} - k_{min} + 1)} \sum_{k=k_{min}}^{k_{max}} \sum_{j=1}^{3} \delta_j(t_k)^2} \qquad (3.6)$$

To analyze the forecasting performance of each algorithm, we compute the RMSE, but also the normalized root-mean-square error (nRMSE) (Eq. 3.7), mean average error (MAE) (Eq. 3.8), and maximum error (Eq. 3.9) of the test set. In Eq. 3.7, $\vec{\mu}_j^{true}$ designates the mean 3D position of marker j on the test set. Because the weights of the RNNs are initialized randomly, given each set of hyper-parameters, we average the RMSE of the cross-validation set over $n_{cv} = 50$ successive runs. Then, during performance evaluation, each metric is averaged over $n_{test} = 300$ runs.

$$nRMSE = \frac{\sqrt{\sum_{k=k_{min}}^{k_{max}} \sum_{j=1}^{3} \delta_j(t_k)^2}}{\sqrt{\sum_{k=k_{min}}^{k_{max}} \sum_{j=1}^{3} \|\vec{\mu}_j^{true} - \vec{u}_j^{true}(t_k)\|_2^2}} \qquad (3.7)$$

$$MAE = \frac{1}{3(k_{max} - k_{min} + 1)} \sum_{k=k_{min}}^{k_{max}} \sum_{j=1}^{3} \delta_j(t_k) \qquad (3.8)$$

$$e_{max} = \max_{k=k_{min},...,k_{max}} \max_{j=1,2,3} \delta_j(t_k) \qquad (3.9)$$

Furthermore, we examine the jitter of the test set, which evaluates the amplitude of the predicted signal oscillations (Eq. 3.10). High fluctuations of the prediction signal result in difficulties concerning robot control during the treatment. The jitter J is minimized when the prediction is constant, thus there is a trade-off between accuracy and jitter [Krilavicius et al., 2016].

$$J = \frac{1}{3(k_{max} - k_{min})} \sum_{k=k_{min}}^{k_{max}-1} \sum_{j=1}^{3} \|\vec{u}_j^{pred}(t_{k+1}) - \vec{u}_j^{pred}(t_k)\|_2 \qquad (3.10)$$

We assume that given an RNN training method, each associated error measure $e_{i,h}$ (the MAE, RMSE, nRMSE, maximum error, or jitter) corresponding to sequence i and horizon h follows a Gaussian distribution $\mathcal{N}(\mu_{i,h}, \sigma_{i,h}^2)$. Indeed, each realization of the random variable $e_{i,h}$ depends on the run index $r \in [\![1, ..., n_{test}]\!]$, and we denote that value $e_{i,h}^{(r)}$. This enables calculating the 95% confidence interval $[\overline{\mu}_{i,h} - \Delta\mu_{i,h}, \overline{\mu}_{i,h} + \Delta\mu_{i,h}]$ for $\mu_{i,h}$, where $\Delta\mu_{i,h}$ is defined in Eq. 3.11. [6]

$$\overline{\sigma}_{i,h}^2 = \frac{1}{n_{test} - 1} \sum_{r=1}^{n_{test}} \left(e_{i,h}^{(r)} - \overline{\mu}_{i,h} \right)^2 \qquad \Delta\mu_{i,h} = 1.96 \frac{\overline{\sigma}_{i,h}}{\sqrt{n_{test}}} \qquad (3.11)$$

The mean of the error $e_{i,h}$ over a subset $I \subseteq [\![1, ..., 9]\!]$ of the 9 sequences[7]and $h \in H = \{h_{min}, ..., h_{max}\}$, denoted by e_I, follows a Gaussian distribution with mean

---

[6]We write $\overline{\mu}_{i,h}$ instead of $\mu_{i,h}$ and $\overline{\sigma}_{i,h}$ instead of $\sigma_{i,h}$ to designate estimators of these parameters given the n_{test} runs.

μ_I. The half-range of the 95% confidence interval for μ_I, denoted by $\Delta\mu_I$, can be calculated according to Eq. 3.12.

$$\Delta\mu_I = \frac{1}{|I||H|}\sqrt{\sum_{i\in I}\sum_{h=h_{min}}^{h_{max}}(\Delta\mu_{i,h})^2} \tag{3.12}$$

3.3 Results

3.3.1 Prediction accuracy and oscillatory behavior of the predicted signal

UORO achieves the lowest RMSE, nRMSE and maximum error averaged over all the sequences and horizons (cf Table 3.2). It is relatively robust to irregular motion, as its nRMSE only increases by 10.6% between regular and irregular breathing. LMS is subject to high jitter values (cf also Fig. 3.8, Fig. 3.11, Fig. 3.12, and Fig. 3.13). The high maximum errors corresponding to RTRL, relative to UORO and LMS, can be observed in Fig. 3.14. The narrow 95% confidence intervals associated with the performance measures reported in Table 3.2 indicate that selecting $n_{test} = 300$ runs is sufficient for providing accurate results.

The graphs representing the performance of each algorithm as a function of the horizon value h appear to have irregular and changing local variations, especially in the case of RTRL and LMS, because the set of hyper-parameters automatically selected by cross-validation is different for each horizon value (Fig. 3.8). These instabilities may also be caused by the relatively low number of breathing records in our dataset. However, it can be observed that the prediction errors and jitter of the test set corresponding to each algorithm globally tend to increase with h.

Linear regression achieves the lowest RMSE and nRMSE for $h \leq 0.2s$ as well as the lowest MAE and maximum error for $h = 0.1s$. The RMSE corresponding to linear regression for $h = 0.2s$ is equal to 0.92mm. LMS gives the lowest RMSE for $0.3s \leq h \leq 0.5s$, the lowest MAE for $0.2s \leq h \leq 0.4s$, the lowest nRMSE for $h = 0.3s$ and $h = 0.4s$, and the lowest maximum error for $0.4s \leq h \leq 0.6s$. The RMSE corresponding to LMS for $h = 0.5s$ is equal to 1.23mm. UORO outperforms the other algorithms in terms of RMSE for $h \geq 0.6s$ and maximum error for $h \geq 0.7s$. The RMSE associated with UORO is rather constant and stays below 1.33mm across all the horizon values considered. RTRL and UORO both have a lower prediction MAE than LMS for $h \geq 0.5s$. Our analysis of the influence of the latency on the relative performance of linear filters, adaptive filters, and artificial neural networks (ANN) agrees with the review of Verma et al. [Verma et al., 2010].

---

[7]When I is the set $[\![1,...,9]\!]$, the confidence intervals calculated are those associated with the 9 records. Otherwise, we select I as the set of indexes associated with the regular or irregular breathing sequences.

Error measure	Prediction method	Average over the 9 sequences	Regular breathing	Irregular breathing[8]
MAE (in mm)	RNN UORO	0.845 ± 0.001	0.674 ± 0.001	0.916 ± 0.001
	RNN RTRL	0.834 ± 0.002	0.684 ± 0.002	0.973 ± 0.003
	LMS	0.957	0.907	1.18
	Lin. regression	4.45	3.23	6.57
	No prediction	3.27	2.89	3.43
RMSE (in mm)	RNN UORO	1.275 ± 0.001	1.030 ± 0.001	1.505 ± 0.002
	RNN RTRL	1.419 ± 0.005	1.119 ± 0.004	1.721 ± 0.005
	LMS	1.370	1.247	1.818
	Lin. regression	6.089	4.454	9.164
	No prediction	4.243	3.952	4.461
nRMSE (no unit)	RNN UORO	0.2824 ± 0.0002	0.2868 ± 0.0004	0.3211 ± 0.0004
	RNN RTRL	0.3027 ± 0.0007	0.2914 ± 0.0008	0.3688 ± 0.0010
	LMS	0.3116	0.2987	0.4198
	Lin. regression	1.411	1.181	2.132
	No prediction	0.9312	1.006	0.9833
Max error (in mm)	RNN UORO	8.81 ± 0.01	7.20 ± 0.02	12.34 ± 0.02
	RNN RTRL	11.68 ± 0.04	10.01 ± 0.04	14.56 ± 0.06
	LMS	9.31	8.59	12.9
	Lin. regression	30.6	23.2	49.0
	No prediction	14.8	13.9	18.2
Jitter (in mm)	RNN UORO	0.9672 ± 0.0004	0.7778 ± 0.0002	0.9973 ± 0.0007
	RNN RTRL	0.7532 ± 0.0015	0.6494 ± 0.0012	0.8735 ± 0.0014
	LMS	1.596	1.646	1.724
	Lin. regression	0.7767	0.6011	1.078
	No prediction	0.4395	0.3877	0.5045

Table 3.2: Comparison of the forecasting performance of each algorithm. Each error value corresponds to the average of a given performance measure of the test set over the sequences considered and the horizon values between 0.1s and 2.0s. The 95% mean confidence intervals associated with the RNNs are calculated assuming that the error distribution is Gaussian (Eq. 3.12).

The jitter associated with RTRL and UORO respectively increases from 0.71mm and 0.94mm for $h = 0.1s$ to 0.78mm and 0.96mm for $h = 2.0s$. However, the jitter associated with linear regression and LMS increases more significantly with h. The jitter corresponding to linear regression is the lowest among the four prediction methods for $h \leq 0.6s$.

The performance of each algorithm as a function of the horizon in the cases of

---

[8]Sequence 201205111057-LACLARUAR-3-O-72 (cf [Krilavicius et al., 2016]) has been removed from the sequences with abnormal respiratory motion when reporting performance measures in the last column, as it does not contain abrupt or sudden motion that typically makes forecasting difficult. In particular, this is why the nRMSE of UORO averaged over the 9 sequences is lower than nRMSE of UORO averaged over the regular or irregular breathing sequences.

[9]see the previous footnote.

Figure 3.8: Forecasting performance of each algorithm as a function of the prediction horizon. Each point corresponds to the average of one performance measure of the test set across the 9 sequences.

normal breathing and abnormal breathing is detailed in Figs. 3.9 and 3.10. The local unsteadiness of the variations of each performance measure with h is more pronounced in these two figures than in Fig. 3.8 because both situations involve averaging results over fewer respiratory traces. However, it still appears that the
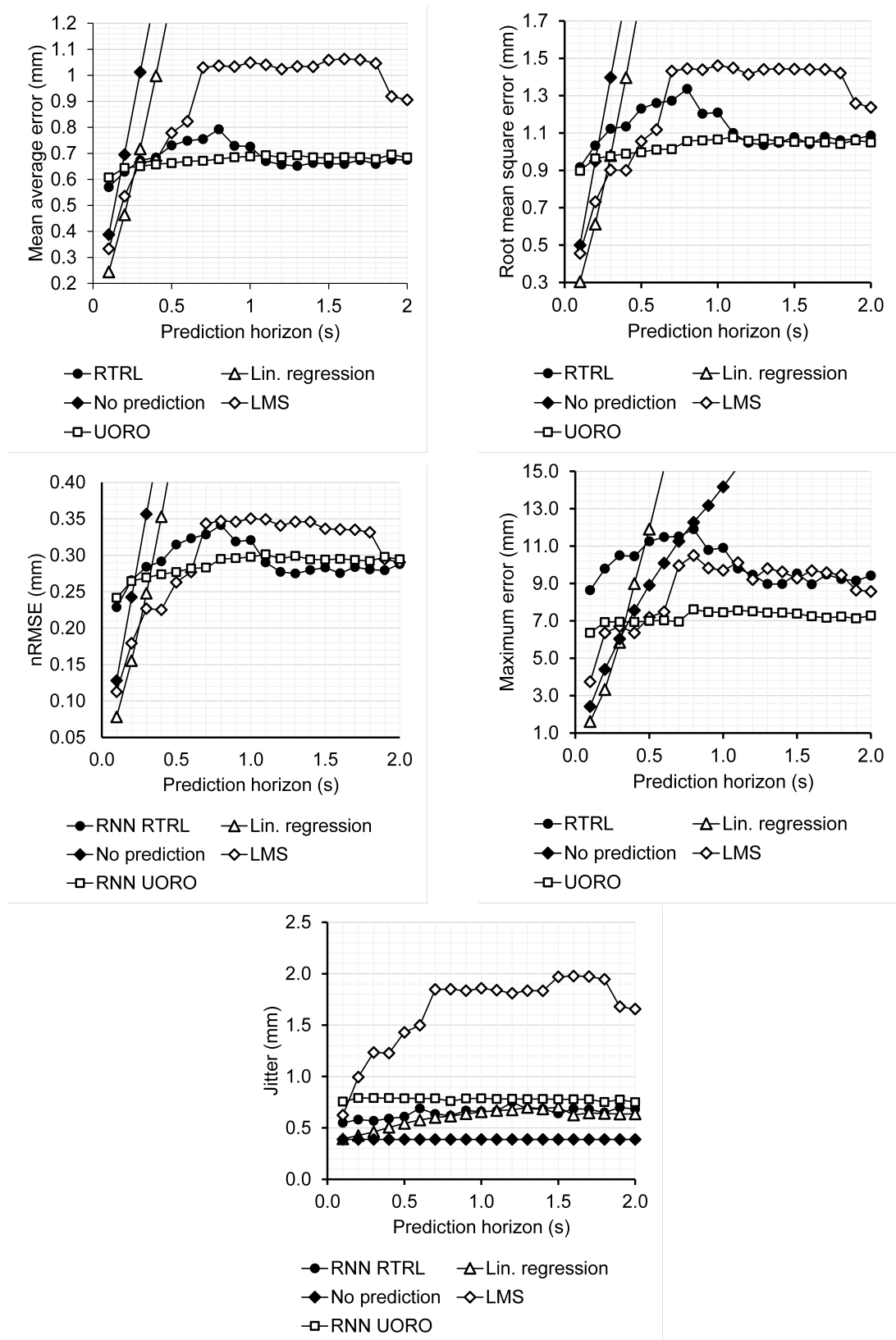
Figure 3.9: Forecasting performance of each algorithm as a function of the prediction horizon. Each point corresponds to the average of one performance measure of the test set across the sequences corresponding to regular breathing.
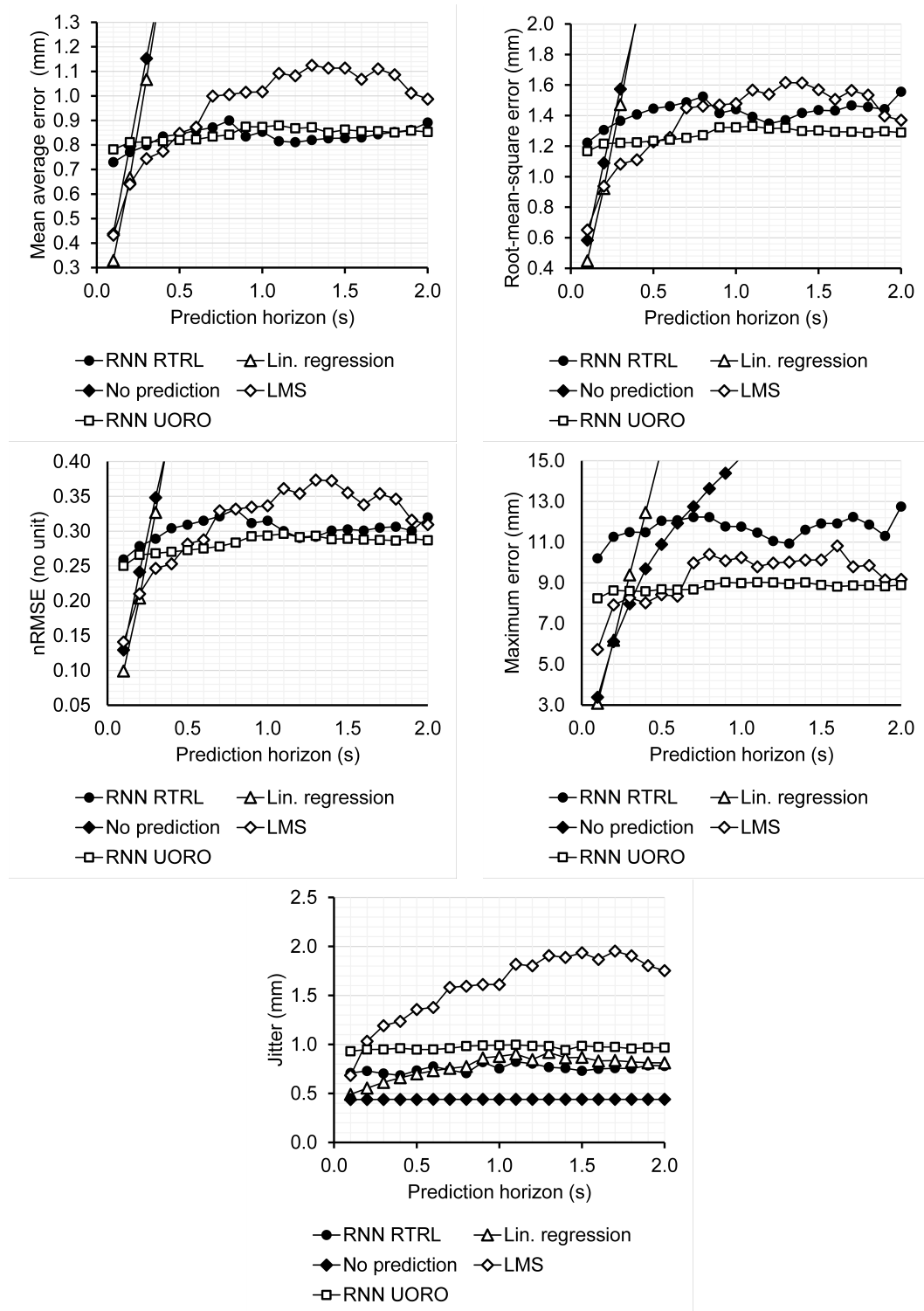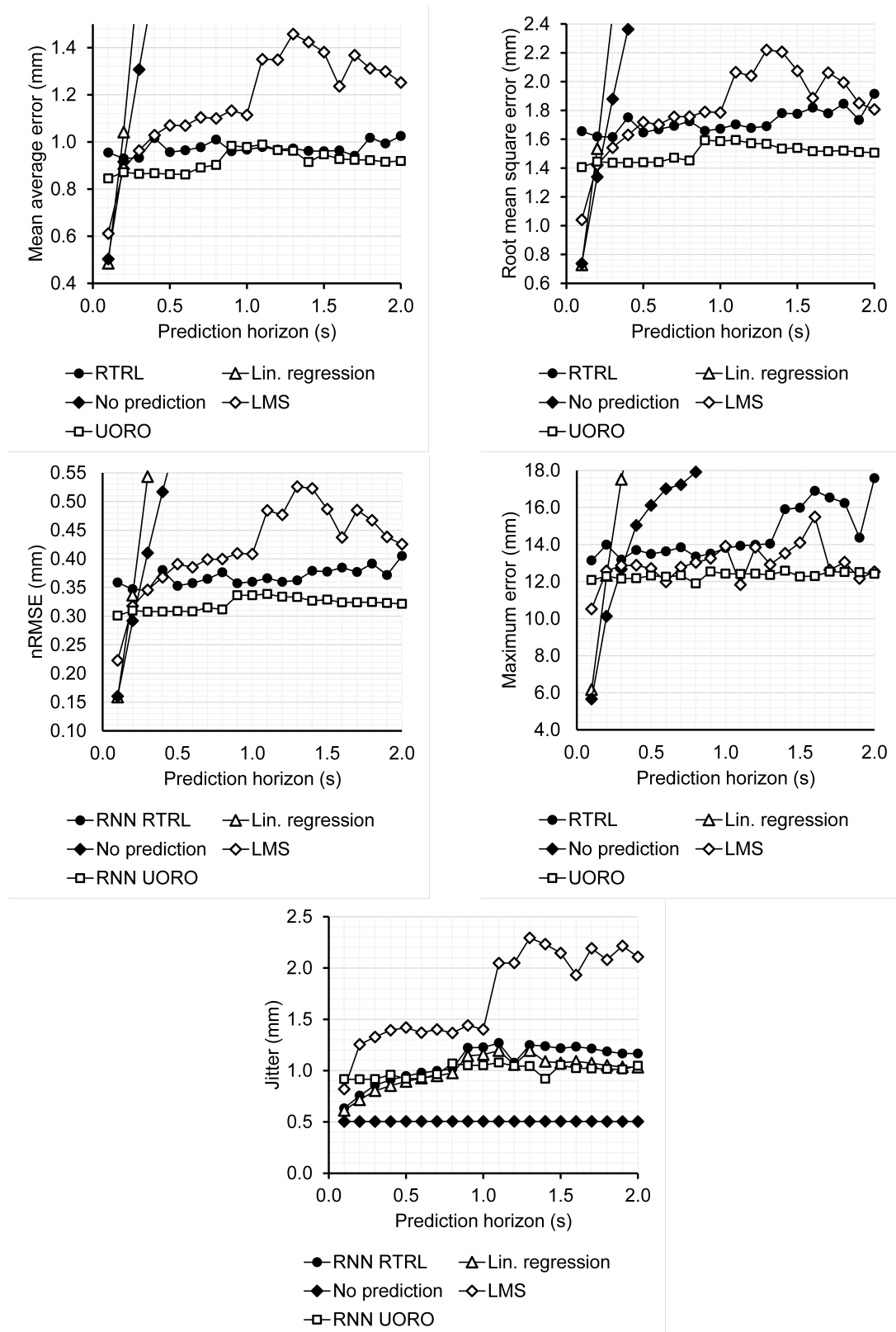
Figure 3.10: Forecasting performance of each algorithm as a function of the prediction horizon. Each point corresponds to the average of one performance measure of the test set across the records corresponding to irregular breathing. [9]
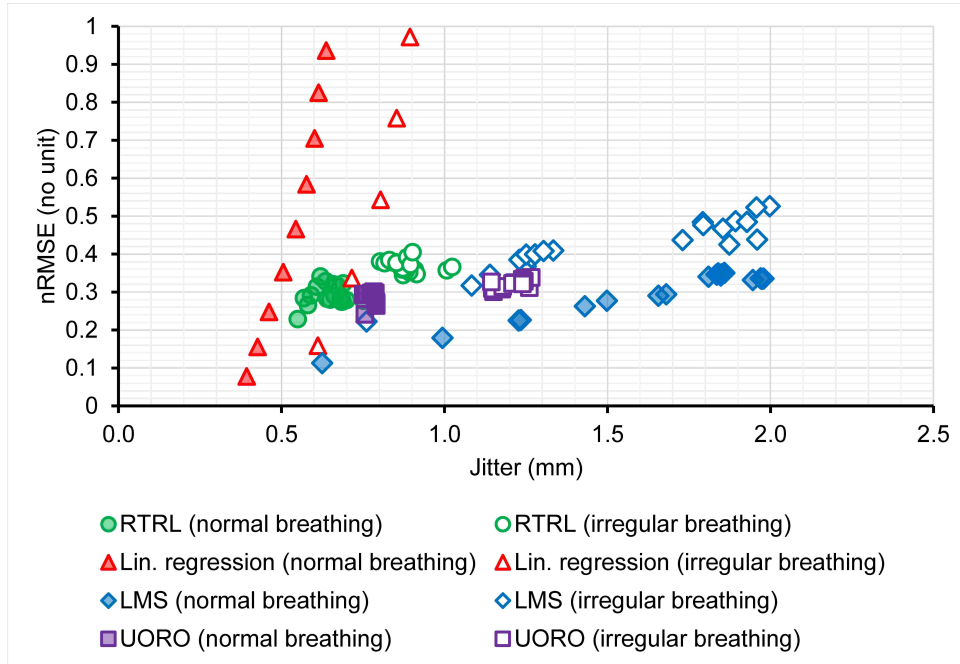
Figure 3.11: Prediction performance of each algorithm in terms of nRMSE and jitter. Each point corresponds to the mean of the nRMSE and jitter of a given algorithm of the test set over the regular or irregular breathing sequences for a single value of the horizon. Datapoints corresponding to linear regression with high horizon values have not been displayed for readability as they correspond to high nRMSEs.

prediction errors globally tend to increase with h in both cases. UORO performs better than the other algorithms for lower horizon values in the scenario of abnormal breathing. Indeed, it achieves the lowest RMSE and nRMSE for $h \geq 0.3s$, and the lowest MAE for $h \geq 0.2s$ (RTRL and UORO achieve comparable performance for higher horizons in terms of MAE).

3.3.2 Influence of the hyper-parameters on prediction accuracy

The prediction nRMSE of the cross-validation set tends to increase as the horizon value h increases (Fig. 3.15). On average over the 9 sequences and all the horizon values, $\eta = 0.1$ and $L = 7.0s$ give the best prediction results. However, for $h = 2.0s$, a higher learning rate $\eta = 0.2$ and a lower value of the SHL $L = 5.0s$ give better results (Figs. 3.15a, 3.15c). In other words, when performing prediction with a high look-ahead time, it is better to make the RNN more dependent on the recent inputs, and quickly correct the synaptic weights when large prediction errors occur. In our experimental setting, $\sigma_{init} = 0.02$ and $q = 90$ hidden units correspond to the lowest nRMSE of the cross-validation set (Figs. 3.15b, 3.15d). The nRMSE of the cross-validation set decreases as the number of hidden units increases, therefore we may achieve higher accuracy with more hidden units. However, that
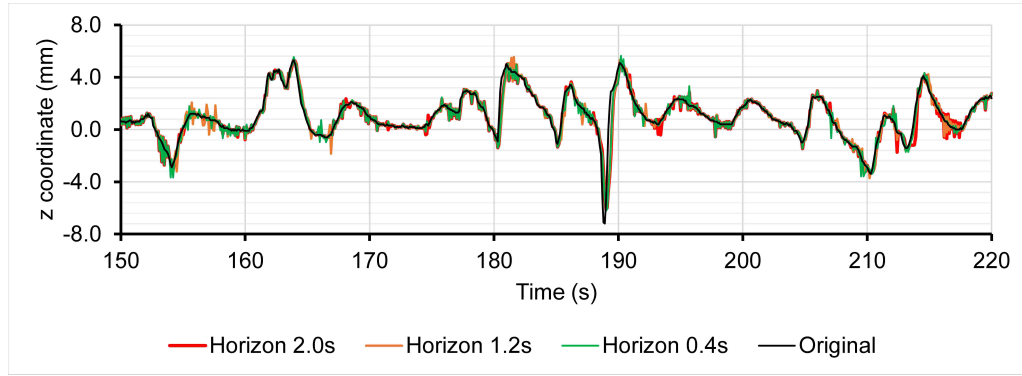
(a) Prediction with an RNN trained with UORO
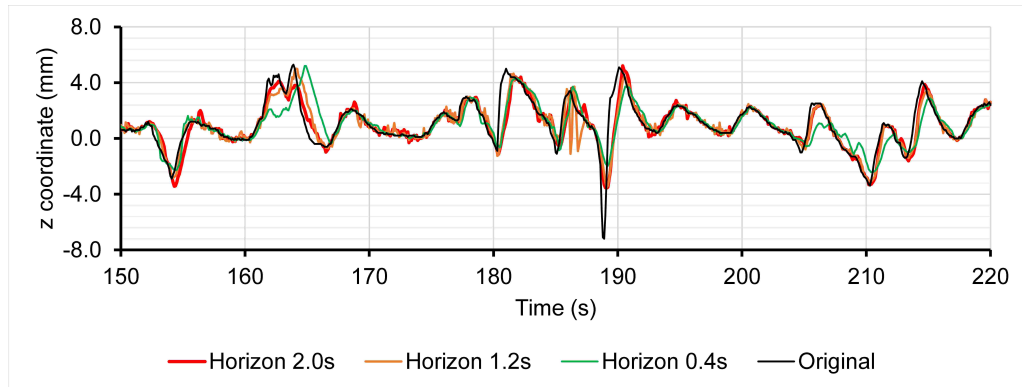
(b) Prediction with an RNN trained with RTRL
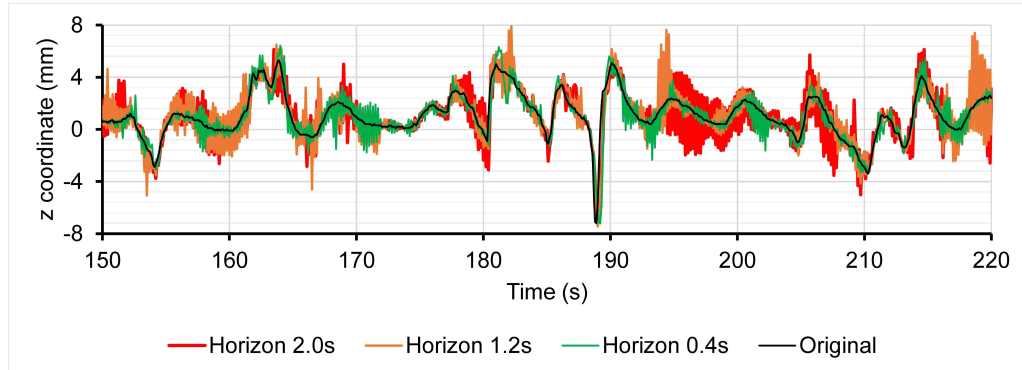
(c) Prediction with LMS

Figure 3.12: Comparison between RTRL, UORO, and LMS regarding the prediction of the position of the z coordinate (spine axis) of marker 3 in sequence 1 (person talking)

would consequently increase the computing time (Fig. 3.17). Similarly, it has been reported in Chapter 2 that increasing the number of hidden units of a vanilla RNN with a single hidden layer trained with RTRL to predict breathing signals led to a decrease of the prediction MAE (Fig. 2.13). Fig. 3.15 displays the nRMSE averaged over the 9 sequences, and the general aforementioned recommendations are not optimal for each sequence. Therefore, we recommend using cross-validation to determine the best hyper-parameter set for each breathing record. The learning rate and SHL appear to be the most important hyper-parameters to tune (Fig. 3.16).
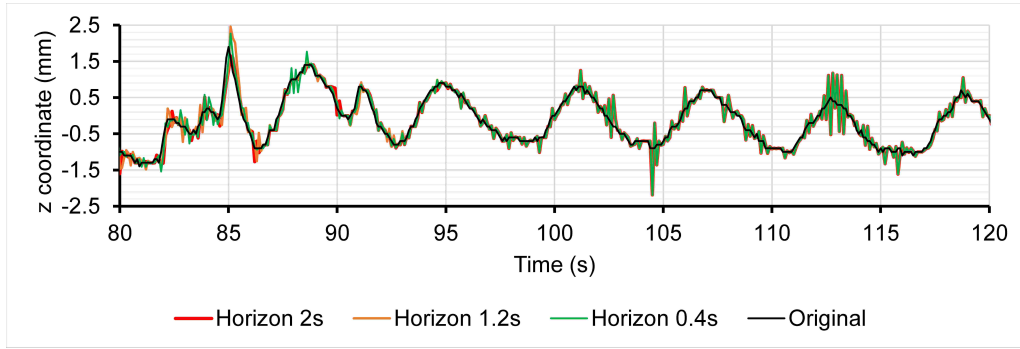
(a) Prediction with an RNN trained with UORO
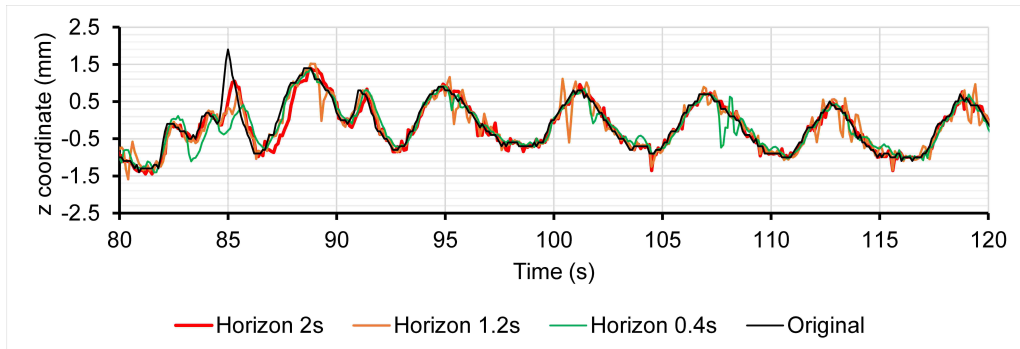
(b) Prediction with an RNN trained with RTRL
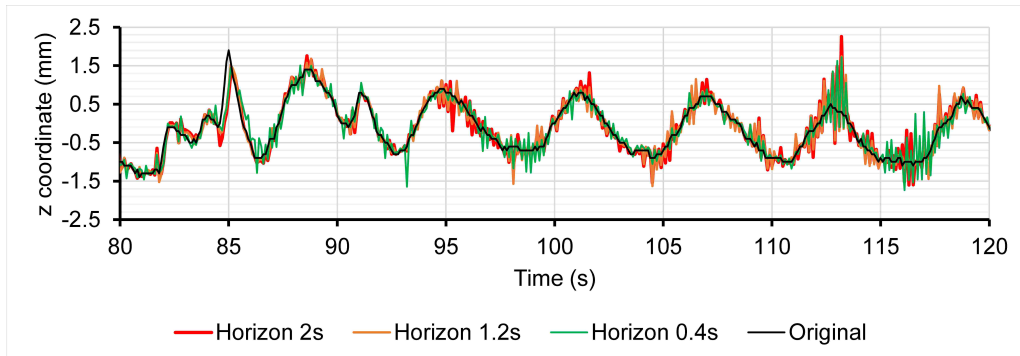
(c) Prediction with LMS

Figure 3.13: Comparison between RTRL, UORO, and LMS regarding the prediction of the position of the z coordinate (spine axis) of marker 3 in sequence 5 (normal breathing)

Appropriately selecting them resulted in a decrease of the mean cross-validation nRMSE of 18.2% (from 0.395 to 0.323) and 21.3% (from 0.417 to 0.329), respectively.

3.3.3   Time performance

UORO has a prediction time per time step equal to 2.8ms for 90 hidden neurons and an SHL of 9.0s, whereas RTRL requires 55ms to perform a single prediction using 55 hidden units with an SHL of 5.5s (Dell Intel Core i9-9900K 3.60GHz CPU 32Gb RAM with Matlab, Fig. 3.17). The complexity $\mathcal{O}(q^3(q + L))$ and resulting high computing time of RTRL is the reason why we performed cross-validation for

(a) Sequence 1 - UORO

(b) Sequence 5 - UORO

(c) Sequence 1 - RTRL

(d) Sequence 5 - RTRL

(e) Sequence 1 - LMS
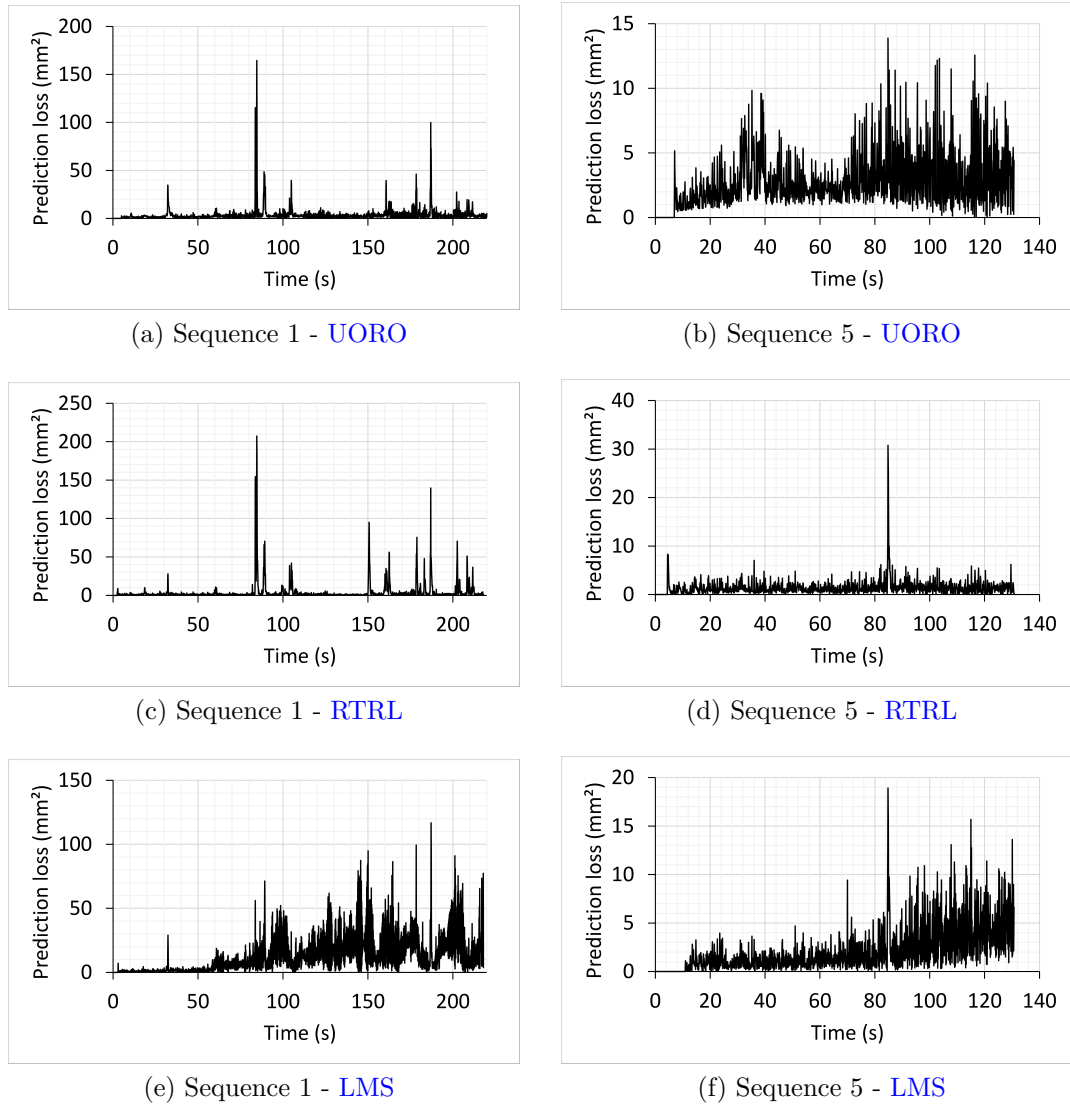
(f) Sequence 5 - LMS

Figure 3.14: Prediction instantaneous square loss (cf Eq. 3.1) for sequence 1 (person talking) and sequence 5 (normal breathing). The horizon value is $h = 2.0s$ and the loss is averaged over 300 runs.

RTRL with fewer hidden units and lower SHL values than UORO, which has a complexity $\mathcal{O}(q(q + L))$ (Table 3.1).

3.4 Discussion

3.4.1 Significance of our results relative to the dataset used

One drawback of our study is the number of sequences used and their duration, which are low in comparison with some other studies related to forecasting in radiotherapy (cf Section 1.2.3 and Table 1.1). Therefore, our numerical results might appear to lack a certain degree of confidence. However, the dataset used is representative of a large variety of breathing patterns including shifts, drifts, slow motion,

(a) Prediction nRMSE of the cross-validation set as a function of the learning rate

(b) Prediction nRMSE of the cross-validation set as a function of the standard deviation of the Gaussian distribution of the initial synaptic weights

(c) Prediction nRMSE of the cross-validation set as a function of the signal history length

(d) Prediction nRMSE of the cross-validation set as a function of the number of hidden units

Figure 3.15: Prediction nRMSE of the cross-validation set as a function of each RNN hyper-parameter, for different horizon values. Given one hyper-parameter, each color point of the associated graph corresponds to the minimum of the nRMSE over every possible combination of the other hyper-parameters in the cross-validation range (Table 3.1). Each nRMSE measure is averaged over the 9 sequences and 50 runs. The black dotted curves correspond to the nRMSE minimum averaged over the horizon values between 0.1s and 2.0s, and the associated error bars correspond to its standard deviation over these horizon values.

sudden irregularities, as well as resting and non-perturbed motion. In addition, our results are consistent with previous studies that claim that linear prediction, linear adaptive filters, and ANNs achieve high performance respectively for low, interme-
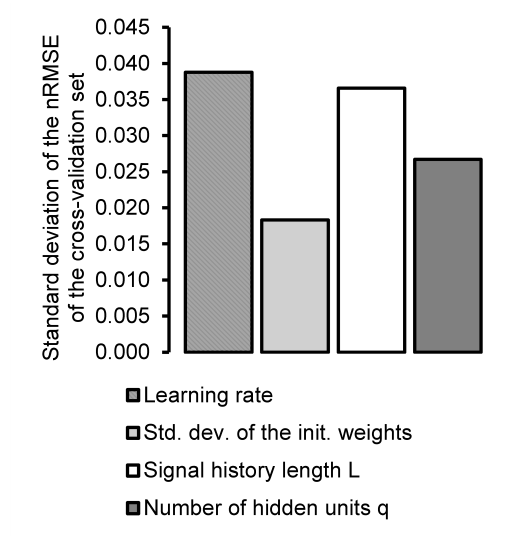
Figure 3.16: Standard deviation of the nRMSE of the cross-validation set (black dotted curves in Fig. 3.15) for each hyper-parameter. A hyper-parameter corresponding to a high standard deviation value has a high influence on the prediction error.
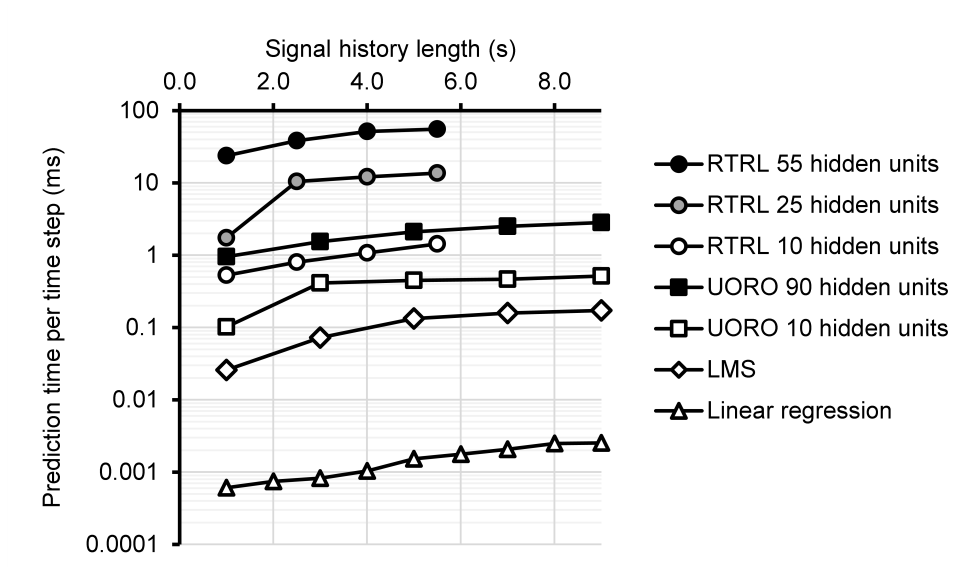
Figure 3.17: Time performance of each algorithm (Dell Intel Core i9-9900K 3.60GHz CPU 32Gb RAM with Matlab)

diate, and high horizon values (cf section 3.3.1). The algorithms studied in our work are online algorithms that do not need a high amount of prior data for making accurate predictions, as demonstrated by the high performance that we achieved with only one minute of training. Because of the reasons mentioned above, we think that the results presented in our study have a significantly high level of confidence and would generalize well to larger datasets.

The online availability of the dataset used is a particular strength of our study, as it enables reproducibility of our results. Most of the previous studies about the prediction of breathing signals for radiotherapy rely on datasets that are not publicly

available (cf section 1.2.3 and Table 1.1), which makes performance comparison difficult.

Laughing and talking are situations where prediction is difficult and they do not happen in a clinical setting. However, evaluating performance with such difficult scenarios gives information about other situations that will sometimes happen during treatment, such as yawning, hiccuping, and coughing. Detecting these anomalies and turning off the irradiation beam when they occur is currently the standard clinical approach. Distinguishing between normal and irregular breathing enabled us to objectively study and quantify the robustness of the algorithms compared (cf Table 3.2, and Figs. 3.9 and 3.10). Since irregular breathing sequences comprise almost half of our entire dataset, the numerical error measures averaged over the nine sequences should be higher than one can expect in more realistic scenarios.

3.4.2   Comparison with previous works

Table 3.3 compares the performance of UORO in our study with some of the results previously reported in the literature (cf Table 1.1 for a more comprehensive literature overview). Comparison with previous studies is complex because the datasets are different. In particular, the frequency, amplitude, and regularity of the signals vary from study to study. Furthermore, the response time, as well as the partition of the data into development and test set are usually arbitrarily selected, thus they also differ between the studies.

The prediction errors in our research might appear relatively large, but this is due to the low sampling frequency (10Hz), the high amplitude of the breathing signals, and the high proportion of irregular patterns in our dataset (cf section 3.4.1). Furthermore, the breathing records that we use have a relatively low duration and therefore the RNNs have fewer data available for training. When taking these circumstances into account, it appears that the errors reported in our study are consistent with the findings of the previous related works.

Our purpose is to examine the extent to which RNNs can efficiently learn to adaptively predict respiratory motion with little data. We do not aim to build a generalized model with a high amount of data. All the RNN-based models reported in Table 3.3 may benefit from adaptive retraining with UORO.

Teo et al. studied breathing records with a frequency of 7.5 Hz and reported lower errors using a multilayer perceptron (MLP) with one hidden layer trained first with backpropagation and retrained online [Teo et al., 2018]. Our higher errors are partly due to the amplitude of the breathing signals in our dataset, which are approximately 3 times higher. Mafi et al. also reported similar but lower prediction errors for a signal sampled at 7.5 Hz using RTRL [Mafi and Moghadam, 2020]. However, they do not provide information concerning signal amplitude, and our results demonstrate that UORO has more benefits than RTRL in practice. The

Work	Network	Training method	Breathing data	Sampling rate	Amount of data	Signal amplitude	Response time	Prediction error
[Sharp et al., 2004]	1-layer MLP	-	1 implanted marker	10 Hz	14 records 48s to 352s	9.1mm to 31.6mm	1) 200ms 2) 1s	RMSE 2.6mm RMSE 5.3mm
[Sun et al., 2017]	1-layer MLP	Levenberg-Marq. & adapt. boosting	RPM data (Varian)	30 Hz	data from 138 scans	Rescaling between -1 and 1	500ms	Max error 0.65 RMSE 0.17 nRMSE 0.28
[Kai et al., 2018]	1-layer RNN	BPTT	1 implanted marker	30 Hz	7 records of 40s to 70s	-	1.0s	RMSE from 0.48mm to 1.37mm
[Teo et al., 2018]	1-layer MLP	Backprop. & adapt. training	Cyberknife Synchrony	7.5 Hz	27 records of 1 min	2mm to 16mm (dvlpmt set)	650	MAE 0.65mm RMSE 0.95mm Max error 3.94mm
[Yun et al., 2019]	3-layer LSTM	- adapt. training	tumor 3D center of mass	25 Hz	158 records of 8 min	0.6mm to 51.2mm	280ms	RMSE 0.9mm
[Lin et al., 2019]	3-layer LSTM	-	RPM data (Varian)	30 Hz	1703 records of 2 to 5 min	Rescaling between -1 and 1	280ms 500ms	MAE 0.112 RMSE 0.139 Max error 1.811
[Mafi and Moghadam, 2020]	RNN -FCL	RTRL	Cyberknife Synchrony	7.5 Hz	43 records of 2.2s to 6.4s	-	665ms	MAE 0.54mm RMSE 0.57mm
[Lee et al., 2021]	LSTM -FCL	BPTT	RPM data (Varian)	30 Hz	550 records 91s to 188s	11.9mm to 25.9mm	210ms	RMSE 0.28mm
Our work	1-layer RNN	UORO	3 external markers (Polaris)	10 Hz	9 records 73s to 222s	6mm to 40mm (SI direction)	0.1s to 2.0s	MAE 0.85mm Max error 8.8mm RMSE 1.28mm nRMSE 0.28

Table 3.3: Comparison of our work with previous studies about time-series forecasting with ANNs for respiratory motion compensation in radiotherapy. In this table, the term "RNN" designates a vanilla RNN, as opposed to LSTMs. "LSTM-FCL" designates a combination of LSTM layers and fully connected layers. A field with " - " indicates that the information is not available. The results corresponding to our study are the performance measures averaged over the horizon values between 0.1s and 2.0s in Table 3.2. The interested reader can refer to Table 1.1 (Section 1.2.3) for further comparisons. [10]

[10]MLP, FCL, and RPM respectively stand for "multilayer perceptron", "fully connected layer" and "real-time position management". By abuse of language, the number of layers mentioned actually refers to the number of hidden layers. For instance, a "1-layer MLP" architecture refers to an MLP with 1 hidden layer.

RMSE error that we achieved is approximately 2 to 4 times lower than the RMSEs reported by Sharp et al., who used an MLP with one hidden layer and breathing records of the same frequency (10Hz) with similar amplitudes [Sharp et al., 2004]. The RMSE that we found is within the range reported by Kai et al., who predicted the position of an internal marker using an RNN with 1 hidden layer trained with BPTT with a much higher frequency (30 Hz) [Kai et al., 2018].

3.5 Conclusion

This is the first study of RNNs trained with UORO for forecasting the position of external markers on the chest and abdomen for safe radiotherapy, to the extent of our knowledge. This method can mitigate the latency of treatment systems due to robot control and radiation delivery preparation. This will in turn help decrease irradiation to healthy tissues and avoid lung radiation therapy side effects such as radiation pneumonitis or pulmonary fibrosis.

Online processing is suitable for breathing motion prediction during the radiotherapy treatment as it enables adaptation to each patient's individual respiratory patterns varying over time. We could efficiently train RNNs using only one minute of breathing data per sequence, as dynamic training can be implemented with limited data. Prediction was performed simultaneously for the three markers so that the RNN discovers and uses information from the correlation between their motion.

UORO achieved the lowest prediction RMSE for horizon values $h \geq 0.6s$, with an average value over 9 breathing sequences not exceeding 1.4mm. These sequences last from 73s to 222s, correspond to a sampling rate of 10Hz and marker position amplitudes varying from 6mm to 40mm in the superior-inferior direction. Moreover, UORO achieved the lowest maximum error for $h \geq 0.7s$ with an average value over the 9 sequences not exceeding 9.1mm. The average of the RMSE and maximum error over the sequences corresponding to regular breathing were respectively lower than 1.1mm and 7.7mm. The nRMSE of UORO only increased by 10.6% when performing the evaluation with the sequences corresponding to irregular breathing instead of regular breathing, which indicates good robustness to sudden changes in respiratory patterns. The calculation time per time step of UORO is equal to 2.8ms for 90 hidden units and an SHL of 9.0s (Dell Intel Core i9-9900K 3.60GHz CPU 32Gb RAM with Matlab). UORO has a much better time performance than RTRL, whose calculation time per time step is equal to 55.2s for 55 hidden units and an SHL of 5.5s.

Linear regression was the most efficient prediction algorithm for low look-ahead time values, with an RMSE lower than 0.9mm for $h \leq 0.2s$. LMS gave the best prediction results for intermediate look-ahead values, with an RMSE lower than 1.2mm for $h \leq 0.5s$. These observations regarding the influence of the horizon agree

with those in Verma et al. [2010]. The errors reported in our study may be higher than in clinical scenarios due to the high proportion of records corresponding to irregular breathing in our dataset.

Gradient clipping was used to ensure numerical stability and we selected a clipping threshold $\tau = 2.0$. The learning rate and SHL were the hyper-parameters with the strongest influence on the prediction performance. We found that a learning rate $\eta = 0.1$ and SHL of 7.0s gave the best results on average, except with high horizon values close to $h = 2.0s$, for which a higher learning rate $\eta = 0.2$ and lower SHL of 5.0s led to better performance. The prediction error decreased as the number of hidden units increased. That fact has previously been observed in the case of RTRL (Chapter 2 Fig. 2.13).

Long Short-Term Memory (LSTM) networks or gated recurrent units (GRU) could be used instead of a vanilla RNN structure, as that could lead to higher prediction accuracy. Furthermore, UORO could be used to dynamically retrain in real-time the last hidden layer of a deep RNN that predicts respiratory waveform signals, as a form of transfer learning. This could improve the robustness of that RNN to unseen examples corresponding to irregular breathing patterns.

Bibliography

Kohei Adachi. *Matrix-based introduction to multivariate data analysis.* Springer, 2016.

Christopher Aicher, Nicholas J Foti, and Emily B Fox. Adaptively truncating back-propagation through time to control gradient bias. In *Uncertainty in Artificial Intelligence*, pages 799–808. PMLR, 2020.

Yuichi Akino, Ryoong-Jin Oh, Norihisa Masai, Hiroya Shiomi, and Toshihiko Inoue. Evaluation of potential internal target volume of liver tumors using cine-MRI. *Medical physics*, 41(11):111704, 2014.

Senjian An, Wanquan Liu, and Svetha Venkatesh. Fast cross-validation algorithms for least squares support vector machine and kernel ridge regression. *Pattern Recognition*, 40(8):2154–2162, 2007.

Anna Andreychenko, B Denis de Senneville, RJM Navest, Rob HN Tijssen, Jan JW Lagendijk, and CAT van den Berg. Respiratory motion model based on the noise covariance matrix of a receive array. *Magnetic resonance in medicine*, 79(3): 1730–1735, 2018.

Fariba Azizmohammadi, Rémi Martin, Joaquim Miro, and Luc Duong. Model-free cardiorespiratory motion prediction from x-ray angiography sequence with lstm network. In *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 7014–7018. IEEE, 2019.

Mohammad Babaeizadeh, Chelsea Finn, Dumitru Erhan, Roy H Campbell, and Sergey Levine. Stochastic variational video prediction. *arXiv preprint arXiv:1710.11252*, 2017.

Shaojie Bai, J Zico Kolter, and Vladlen Koltun. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271*, 2018.

Hannah E. Bainbridge, Martin J. Menten, Martin F. Fast, Simeon Nill, Uwe Oelfke, and Fiona McDonald. Treating locally advanced lung cancer with a 1.5t mr-linac

-effects of the magnetic field and irradiation geometry on conventionally fractionated and isotoxic dose-escalated radiotherapy. *Radiotherapy and Oncology*, 125(2):280–285, 2017. ISSN 0167-8140. doi: https://doi.org/10.1016/j.radonc. 2017.09.009. URL https://www.sciencedirect.com/science/article/pii/S0167814017325719.

Simon Baker, Daniel Scharstein, JP Lewis, Stefan Roth, Michael J Black, and Richard Szeliski. A database and evaluation methodology for optical flow. *International journal of computer vision*, 92(1):1–31, 2011.

Domjan Barić, Petar Fumić, Davor Horvatić, and Tomislav Lipic. Benchmarking attention-based interpretability of deep learning in multivariate time series predictions. *Entropy*, 23(2):143, 2021.

Ana María Barragán-Montero, Dan Nguyen, Weiguo Lu, Mu-Han Lin, Roya Norouzi-Kandalan, Xavier Geets, Edmond Sterpin, and Steve Jiang. Three-dimensional dose prediction for lung imrt patients with deep neural networks: robust learning from heterogeneous beam configurations. *Medical physics*, 46(8): 3679–3691, 2019.

Kim E Barrett, Susan M Barman, Heddwen L Brooks, and Jason X-J Yuan. *Ganong's review of medical physiology*. McGraw-Hill Education, 2019.

Frederik Benzing, Marcelo Matheus Gauy, Asier Mujika, Anders Martinsson, and Angelika Steger. Optimal kronecker-sum approximation of real time recurrent learning. In *International Conference on Machine Learning*, pages 604–613. PMLR, 2019.

Jenny Bertholet, Antje Knopf, Björn Eiben, Jamie McClelland, Alexander Grimwood, Emma Harris, Martin Menten, Per Poulsen, Doan Trang Nguyen, Paul Keall, et al. Real-time intrafraction motion monitoring in external beam radiotherapy. *Physics in Medicine & Biology*, 64(15):15TR01, 2019.

Paul J Besl and Neil D McKay. Method for registration of 3-D shapes. In *Sensor fusion IV: control paradigms and data structures*, volume 1611, pages 586–606. International Society for Optics and Photonics, 1992.

F Edward Boas and Dominik Fleischmann. CT artifacts: causes and reduction techniques. *Imaging in medicine*, 4(2):229–240, 2012.

Thomas Bohnstingl, Stanisław Woźniak, Wolfgang Maass, Angeliki Pantazi, and Evangelos Eleftheriou. Online spatio-temporal learning in deep neural networks. *arXiv preprint arXiv:2007.12723*, 2020.

Jean-Yves Bouguet et al. Pyramidal implementation of the affine Lucas Kanade feature tracker, description of the algorithm. *Intel Corporation*, 5(1-10):4, 2001.

Alexandra E Bourque, Stéphane Bedwani, Édith Filion, and Jean-François Carrier. A particle filter based autocontouring algorithm for lung tumor tracking using dynamic magnetic resonance imaging. *Medical physics*, 43(9):5161–5169, 2016.

George E. P. Box, Gregory C. Reinsel, and Gwilym M. Jenkins. *Time series analysis : forecasting and control.* Prentice-Hall, Englewood Cliffs, NJ, 1994.

Dirk Boye, Golnoosh Samei, Johannes Schmidt, Gabor Székely, and Christine Tanner. Population based modeling of respiratory lung motion and prediction from partial information. In *Medical Imaging 2013: Image Processing*, volume 8669, page 86690U. International Society for Optics and Photonics, 2013.

James Bradbury, Stephen Merity, Caiming Xiong, and Richard Socher. Quasi-recurrent neural networks. *arXiv preprint arXiv:1611.01576*, 2016.

Kristy K Brock, Sasa Mutic, Todd R McNutt, Hua Li, and Marc L Kessler. Use of image registration and fusion algorithms and techniques in radiotherapy: Report of the aapm radiation therapy committee task group no. 132. *Medical physics*, 44 (7):e43–e76, 2017.

Thomas Brox, Andrés Bruhn, Nils Papenberg, and Joachim Weickert. High accuracy optical flow estimation based on a theory for warping. In *European conference on computer vision*, pages 25–36. Springer, 2004.

Wonmin Byeon, Thomas M Breuel, Federico Raue, and Marcus Liwicki. Scene labeling with lstm recurrent neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3547–3555, 2015.

CancerConnect. Side effect of canver treatment: Lung damage (acute pulmonary toxicity). https://news.cancerconnect.com/treatment-care/side-effect-of-cancer-treatment-lung-damage-acute-pulmonary-toxicity, 2020. [Online; accessed 08-April-2021].

Defu Cao, Yujing Wang, Juanyong Duan, Ce Zhang, Xia Zhu, Conguri Huang, Yunhai Tong, Bixiong Xu, Jing Bai, Jie Tong, et al. Spectral temporal graph neural network for multivariate time-series forecasting. *arXiv preprint arXiv:2103.07719*, 2021.

Joel N K Carlson, Jong Min Park, So-Yeon Park, Jong In Park, Yunseok Choi, and Sung-Joon Ye. A machine learning approach to the accurate prediction of multi-leaf collimator positional errors. *Physics in Medicine and Biology*, 61(6):

2514–2531, mar 2016. doi: 10.1088/0031-9155/61/6/2514. URL https://doi.org/10.1088/0031-9155/61/6/2514.

Shiyu Chang, Yang Zhang, Wei Han, Mo Yu, Xiaoxiao Guo, Wei Tan, Xiaodong Cui, Michael Witbrock, Mark Hasegawa-Johnson, and Thomas S Huang. Dilated recurrent neural networks. *arXiv preprint arXiv:1710.02224*, 2017.

Dong Chen, Hongzhi Xie, Shuyang Zhang, and Lixu Gu. Lung respiration motion modeling: a sparse motion field presentation method using biplane x-ray images. *Physics in Medicine and Biology*, 62(19):7855–7873, 2017. doi: 10.1088/1361-6560/aa8841. URL https://iopscience.iop.org/article/10.1088/1361-6560/aa8841.

Dong Chen, Hongzhi Xie, Lixu Gu, Wei Guo, Liang Tian, and Jing Liu. A statistical weighted sparse-based local lung motion modelling approach for model-driven lung biopsy. *International journal of computer assisted radiology and surgery*, 15(8): 1279–1290, 2020.

Haibin Chen, Zichun Zhong, Yiwei Yang, Jiawei Chen, Linghong Zhou, Xin Zhen, and Xuejun Gu. Internal motion estimation by internal-external motion modeling for lung cancer radiotherapy. *Scientific reports*, 8(1):1–14, 2018.

Qin-Sheng Chen, Martin S Weinhous, F Christopher Deibel, Jay P Ciezki, and Roger M Macklis. Fluoroscopic study of tumor motion due to breathing: facilitating precise radiation therapy for lung cancer patients. *Medical physics*, 28(9): 1850–1856, 2001.

Chhatkuli. *Development of a markerless tumor prediction system using principal component analysis and multi-channel singular spectral analysis with real-time respiratory phase recognition in radiation therapy*. PhD thesis, The University of Tokyo, 2016. URL https://repository.dl.itc.u-tokyo.ac.jp/record/48459/files/A32580_summary.pdf.

Ritu Bhusal Chhatkuli, Kazuyuki Demachi, Naoki Miyamoto, Mitsuru Uesaka, Akihiro Haga, et al. Dynamic image prediction using principal component and multi-channel singular spectral analysis: a feasibility study. *Open Journal of Medical Imaging*, 5(03):133, 2015.

Vinay Kumar Reddy Chimmula and Lei Zhang. Time series forecasting of covid-19 transmission in canada using lstm networks. *Chaos, Solitons & Fractals*, 135: 109864, 2020.

Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations

using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.

SeungWook Choi, Yongjun Chang, Namkug Kim, Sung Ho Park, Si Yeol Song, and Heung Sik Kang. Performance enhancement of respiratory tumor motion prediction using adaptive support vector regression: Comparison with adaptive neural network method. *International Journal of Imaging Systems and Technology*, 24(1):8–15, 2014. doi: https://doi.org/10.1002/ima.22073. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/ima.22073.

Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016.

S P M Crijns, B W Raaymakers, and J J W Lagendijk. Proof of concept of MRI-guided tracked radiation delivery: tracking one-dimensional motion. *Physics in Medicine and Biology*, 57(23):7863–7872, nov 2012. doi: 10.1088/0031-9155/57/23/7863. URL https://doi.org/10.1088/0031-9155/57/23/7863.

Zhiyong Cui, Ruimin Ke, Ziyuan Pu, and Yinhai Wang. Stacked bidirectional and unidirectional lstm recurrent neural network for forecasting network-wide traffic state with missing values. *Transportation Research Part C: Emerging Technologies*, 118:102674, 2020. ISSN 0968-090X. doi: https://doi.org/10.1016/j.trc.2020.102674. URL https://www.sciencedirect.com/science/article/pii/S0968090X20305891.

Emmanuel de Bézenac, Syama Sundar Rangapuram, Konstantinos Benidis, Michael Bohlke-Schneider, Richard Kurle, Lorenzo Stella, Hilaf Hasson, Patrick Gallinari, and Tim Januschowski. Normalizing kalman filters for multivariate time series analysis. *Advances in Neural Information Processing Systems*, 33, 2020.

Jennifer Dhont, Jef Vandemeulebroucke, Davide Cusumano, Luca Boldrini, Francesco Cellini, Vincenzo Valentini, and Dirk Verellen. Multi-object tracking in MRI-guided radiotherapy using the tracking-learning-detection framework. *Radiotherapy and Oncology*, 138:25–29, 2019.

Manoj Diwakar and Manoj Kumar. A review on CT image noise and its denoising. *Biomedical Signal Processing and Control*, 42:73–88, 2018.

Bin Dong, Yan Jiang Graves, Xun Jia, and Steve B Jiang. Optimal surface marker locations for tumor motion estimation in lung cancer radiotherapy. *Physics in Medicine & Biology*, 57(24):8201, 2012.

Robert Dürichen, Tobias Wissel, Floris Ernst, and Achim Schweikard. Respiratory motion compensation with relevance vector machines. In Kensaku Mori, Ichiro Sakuma, Yoshinobu Sato, Christian Barillot, and Nassir Navab, editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2013*, pages 108–115, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg. ISBN 978-3-642-40763-5.

Jan Ehrhardt, Cristian Lorenz, et al. *4D modeling and estimation of respiratory motion for radiation therapy*, volume 10. Springer, 2013.

biomedical image computing ETH Zürich. Datasets, 4d mri lung data. `https://bmic.ee.ethz.ch/research/datasets.html`, 2021. [Online; accessed 26-May-2021].

Biagio Gino Fallone. The rotating biplanar linac-magnetic resonance imaging system. *Seminars in Radiation Oncology*, 24(3):200–202, 2014. ISSN 1053-4296. doi: https://doi.org/10.1016/j.semradonc.2014.02.011. URL `https://www.sciencedirect.com/science/article/pii/S1053429614000289`. Magnetic Resonance Imaging in Radiation Oncology.

Qi Fan, Xiaoyang Yu, Yanqiao Zhao, and Shuang Yu. A respiratory motion prediction method based on improved relevance vector machine. *Mobile Networks and Applications*, 25(6):2270–2279, 2020.

Oliver Faust, Yuki Hagiwara, Tan Jen Hong, Oh Shu Lih, and U Rajendra Acharya. Deep learning for healthcare applications based on physiological signals: A review. *Computer Methods and Programs in Biomedicine*, 161:1–13, 2018. ISSN 0169-2607. doi: https://doi.org/10.1016/j.cmpb.2018.04.005. URL `https://www.sciencedirect.com/science/article/pii/S0169260718301226`.

Hadi Fayad, Jean-François Clément, Tinsu Pan, Christian Roux, C Cheze Le Rest, Olivier Pradier, and Dimitris Visvikis. Towards a generic respiratory motion model for 4d ct imaging of the thorax. In *2009 IEEE Nuclear Science Symposium Conference Record (NSS/MIC)*, pages 3975–3979. IEEE, 2009.

Hadi J Fayad, Christian Buerger, Charalampos Tsoumpas, Catherine Cheze-Le-Rest, and Dimitris Visvikis. A generic respiratory motion model based on 4d mri imaging and 2d image navigators. In *2012 IEEE Nuclear Science Symposium and Medical Imaging Conference Record (NSS/MIC)*, pages 4058–4061. IEEE, 2012.

Mary Feng, Gilmer Valdes, Nayha Dixit, and Timothy D Solberg. Machine learning in radiation oncology: opportunities, requirements, and needs. *Frontiers in oncology*, 8:110, 2018.

Chelsea Finn, Ian Goodfellow, and Sergey Levine. Unsupervised learning for physical interaction through video prediction. *arXiv preprint arXiv:1605.07157*, 2016.

Benjamin W. Fischer-Valuck, Lauren Henke, Olga Green, Rojano Kashani, Sahaja Acharya, Jeffrey D. Bradley, Clifford G. Robinson, Maria Thomas, Imran Zoberi, Wade Thorstad, Hiram Gay, Jiayi Huang, Michael Roach, Vivian Rodriguez, Lakshmi Santanam, Harold Li, Hua Li, Jessika Contreras, Thomas Mazur, Dennis Hallahan, Jeffrey R. Olsen, Parag Parikh, Sasa Mutic, and Jeff Michalski. Two-and-a-half-year clinical experience with the world's first magnetic resonance image guided radiation therapy system. *Advances in Radiation Oncology*, 2(3):485–493, 2017. ISSN 2452-1094. doi: https://doi.org/10.1016/j.adro.2017.05.006. URL https://www.sciencedirect.com/science/article/pii/S2452109417300969.

David Fleet and Yair Weiss. Optical flow estimation. In *Handbook of mathematical models in computer vision*, pages 237–257. Springer, 2006.

Noemi Garau, Riccardo Via, Giorgia Meschini, Danny Lee, Paul Keall, Marco Riboldi, Guido Baroni, and Chiara Paganelli. A roi-based global motion model established on 4dct and 2d cine-mri data for mri-guidance in radiation therapy. *Physics in Medicine & Biology*, 64(4):045002, 2019.

Drosoula Giantsoudi, Bruno De Man, Joost Verburg, Alexei Trofimov, Yannan Jin, Ge Wang, Lars Gjesteby, and Harald Paganetti. Metal artifacts in computed tomography for radiation therapy planning: dosimetric effects and impact of metal artifact reduction. *Physics in Medicine and Biology*, 62(8):R49–R80, mar 2017. doi: 10.1088/1361-6560/aa5293. URL https://doi.org/10.1088/1361-6560/aa5293.

Nina Golyandina and D Stepanov. Ssa-based approaches to analysis and forecast of multidimensional time series. In *proceedings of the 5th St. Petersburg workshop on simulation*, volume 293, page 298. St. Petersburg State University St. Petersburg, Russia, 2005.

John H Goodband, Olivier CL Haas, and JA Mills. A comparison of neural network approaches for on-line prediction in IGRT. *Medical physics*, 35(3):1113–1122, 2008.

Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *arXiv preprint arXiv:1406.2661*, 2014.

Alex Graves and Jürgen Schmidhuber. Offline handwriting recognition with multidimensional recurrent neural networks. In D. Koller, D. Schuurmans, Y. Bengio,

and L. Bottou, editors, *Advances in Neural Information Processing Systems*, volume 21. Curran Associates, Inc., 2009. URL https://proceedings.neurips.cc/paper/2008/file/66368270ffd51418ec58bd793f2d9b1b-Paper.pdf.

Alex Graves, Santiago Fernández, Marcus Liwicki, Horst Bunke, and Jürgen Schmidhuber. Unconstrained online handwriting recognition with recurrent neural networks. In *Advances in Neural Information Processing Systems 20, NIPS 2008*, 2008.

Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. Speech recognition with deep recurrent neural networks. In *2013 IEEE international conference on acoustics, speech and signal processing*, pages 6645–6649. Ieee, 2013.

Vincent Le Guen and Nicolas Thome. Shape and time distortion loss for training deep time series forecasting models. *arXiv preprint arXiv:1909.09020*, 2019.

Vincent Le Guen and Nicolas Thome. Probabilistic time series forecasting with structured shape and temporal diversity. *arXiv preprint arXiv:2010.07349*, 2020.

Vincent Le Guen and Nicolas Thome. Deep time series forecasting with shape and temporal criteria. *arXiv preprint arXiv:2104.04610*, 2021.

Xiao Han. Mr-based synthetic ct generation using a deep convolutional neural network method. *Medical Physics*, 44(4):1408–1419, 2017. doi: https://doi.org/10.1002/mp.12155. URL https://aapm.onlinelibrary.wiley.com/doi/abs/10.1002/mp.12155.

Daniel P Harley, William S Krimsky, Saiyad Sarkar, David Highfield, Cengis Aygun, and Burak Gurses. Fiducial marker placement using endobronchial ultrasound and navigational bronchoscopy for stereotactic radiosurgery: an alternative strategy. *The Annals of thoracic surgery*, 89(2):368–374, 2010.

Wendy Harris, Lei Ren, Jing Cai, You Zhang, Zheng Chang, and Fang-Fang Yin. A technique for generating volumetric cine-magnetic resonance imaging. *International Journal of Radiation Oncology* Biology* Physics*, 95(2):844–853, 2016.

Simon S Haykin. *Adaptive filter theory*. Pearson, 2008.

Simon S Haykin et al. *Neural networks and learning machines*. Pearson, 2009.

Tiancheng He, Zhong Xue, Weixin Xie, and Stephen TC Wong. Online 4-d ct estimation for patient-specific respiratory motion based on real-time breathing signals. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 392–399. Springer, 2010.

Tiancheng He, Zhong Xue, Nam Yu, Paige L Nitsch, Bin S Teh, and Stephen T Wong. Estimating dynamic lung images from high-dimension chest surface motion using 4d statistical model. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 138–145. Springer, 2014.

Tiancheng He, Ramiro Pino, Bin Teh, Stephen Wong, and Zhong Xue. Dynamic respiratory motion estimation using patch-based kernel-pca priors for lung cancer radiotherapy. In *Molecular Imaging, Reconstruction and Analysis of Moving Body Organs, and Stroke Imaging and Treatment*, pages 55–65. Springer, 2017.

Ryusuke Hirai, Yukinobu Sakata, Akiyuki Tanizawa, and Shinichiro Mori. Real-time tumor tracking using fluoroscopic imaging with deep neural network analysis. *Physica Medica*, 59:22–29, 2019.

Noriyasu Homma, Yoshihiro Takai, Haruna Endo, Kei Ichiji, Yuichiro Narita, Xiaoyong Zhang, Masao Sakai, Makoto Osanai, Makoto Abe, Norihiro Sugita, et al. Markerless lung tumor motion tracking by dynamic decomposition of x-ray image intensity. *Journal of medical engineering*, 2013, 2013.

Berthold KP Horn and Brian G Schunck. Determining optical flow. In *Techniques and Applications of Image Understanding*, volume 281, pages 319–331. International Society for Optics and Photonics, 1981.

Shu-Hui Hsu, Yue Cao, Ke Huang, Mary Feng, and James M Balter. Investigation of a method for generating synthetic CT models from MRI scans of the head and neck for radiation therapy. *Physics in Medicine and Biology*, 58(23):8419–8435, nov 2013. doi: 10.1088/0031-9155/58/23/8419. URL https://doi.org/10.1088/0031-9155/58/23/8419.

Elizabeth Huynh, Ahmed Hosny, Christian Guthier, Danielle S Bitterman, Steven F Petit, Daphne A Haas-Kogan, Benjamin Kann, Hugo JWL Aerts, and Raymond H Mak. Artificial intelligence in radiation oncology. *Nature Reviews Clinical Oncology*, 17(12):771–781, 2020.

Rob Hyndman, Anne B Koehler, J Keith Ord, and Ralph D Snyder. *Forecasting with exponential smoothing: the state space approach*. Springer Science & Business Media, 2008.

Marcus Isaksson, Joakim Jalden, and Martin J Murphy. On using an adaptive neural network to predict lung tumor motion during respiration for radiotherapy applications. *Medical physics*, 32(12):3801–3809, 2005.

Max Jaderberg, Karen Simonyan, Andrew Zisserman, and Koray Kavukcuoglu. Spatial transformer networks. *arXiv preprint arXiv:1506.02025*, 2015.

Max Jaderberg, Wojciech Marian Czarnecki, Simon Osindero, Oriol Vinyals, Alex Graves, David Silver, and Koray Kavukcuoglu. Decoupled neural interfaces using synthetic gradients. In *International Conference on Machine Learning*, pages 1627–1635. PMLR, 2017.

Herbert Jaeger. *Tutorial on training recurrent neural networks, covering BPPT, RTRL, EKF and the "echo state network" approach*, volume 5. GMD-Forschungszentrum Informationstechnik Bonn, 2002.

Lakhmi C Jain, Manjeevan Seera, Chee Peng Lim, and Pagavathigounder Balasubramaniam. A review of online learning in supervised neural networks. *Neural computing and applications*, 25(3):491–509, 2014.

Daniel Jarrett, Eleanor Stride, Katherine Vallis, and Mark J Gooding. Applications and limitations of machine learning in radiation oncology. *The British journal of radiology*, 92(1100):20190001, 2019.

Steve B. Jiang. Technical aspects of image-guided respiration-gated radiation therapy. *Medical Dosimetry*, 31(2):141–151, 2006. ISSN 0958-3947. doi: https://doi.org/10.1016/j.meddos.2005.12.005. URL https://www.sciencedirect.com/science/article/pii/S0958394705001998. Image-Guided Radiation Therapy, Part 2: Cone-Beam Imaging and Respiratory Motion.

Weiwei Jiang and Jiayun Luo. Graph neural network for traffic forecasting: A survey. *arXiv preprint arXiv:2101.11174*, 2021.

Xiaojie Jin, Huaxin Xiao, Xiaohui Shen, Jimei Yang, Zhe Lin, Yunpeng Chen, Zequn Jie, Jiashi Feng, and Shuicheng Yan. Predicting scene parsing and motion dynamics in the future. *arXiv preprint arXiv:1711.03270*, 2017.

Alexander Jöhl, Stefanie Ehrbar, Matthias Guckenberger, Stephan Klöck, Mirko Meboldt, Melanie Zeilinger, Stephanie Tanadini-Lang, and Marianne Schmid Daners. Performance comparison of prediction filters for respiratory motion tracking in radiotherapy. *Medical physics*, 47(2):643–650, 2020.

Jiang Kai, Fumitake Fujii, and Takehiro Shiinoki. Prediction of lung tumor motion based on recurrent neural network. In *2018 IEEE International Conference on Mechatronics and Automation (ICMA)*, pages 1093–1099. IEEE, 2018.

Manish Kakar, Håkan Nyström, Lasse Rye Aarup, Trine Jakobi Nøttrup, and Dag Rune Olsen. Respiratory motion prediction by using the adaptive neuro fuzzy inference system (anfis). *Physics in Medicine & Biology*, 50(19):4721, 2005.

Benjamin H Kann, Sanjay Aneja, Gokoulakrichenane V Loganadane, Jacqueline R Kelly, Stephen M Smith, Roy H Decker, B Yu James, Henry S Park, Wendell G

Yarbrough, Ajay Malhotra, et al. Pretreatment identification of head and neck cancer nodal metastasis and extranodal extension using deep learning neural networks. *Scientific reports*, 8(1):1–11, 2018.

Nan Rosemary Ke, Anirudh Goyal, Olexa Bilaniuk, Jonathan Binas, Michael C Mozer, Chris Pal, and Yoshua Bengio. Sparse attentive backtracking: Temporal creditassignment through reminding. *arXiv preprint arXiv:1809.03702*, 2018.

Azzam Khankan, Saif Althaqfi, et al. Demystifying Cyberknife stereotactic body radiation therapy for interventional radiologists. *The Arab Journal of Interventional Radiology*, 1(2):55, 2017.

Andrew P King, Christian Buerger, Charalampos Tsoumpas, Paul K Marsden, and Tobias Schaeffter. Thoracic respiratory motion estimation from mri using a statistical model and a 2-d image navigator. *Medical image analysis*, 16(1):252–264, 2012.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Benjamin Klein, Lior Wolf, and Yehuda Afek. A dynamic convolutional layer for short range weather prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4840–4848, 2015.

Andreas Krauss, Simeon Nill, and U Oelfke. The comparative performance of four respiratory motion predictors for real-time tumour tracking. *Physics in Medicine & Biology*, 56(16):5303, 2011.

Tomas Krilavicius, Indre Zliobaite, Henrikas Simonavicius, and Laimonas Jaruevicius. Predicting respiratory motion for real-time tumour tracking in radiotherapy. In *2016 IEEE 29th International Symposium on Computer-Based Medical Systems (CBMS)*, pages 7–12. IEEE, 2016.

Yann LeCun and Ishan Misra. Self-supervised learning: The dark matter of intelligence. https://ai.facebook.com/blog/self-supervised-learning-the-dark-matter-of-intelligence, 2021. [Online; accessed 9-May-2021].

Minsik Lee, Min-Seok Cho, Hoyeon Lee, Chiyoung Jeong, Jungwon Kwak, Jinhong Jung, Su Ssan Kim, Sang Min Yoon, Si Yeol Song, Sang-wook Lee, et al. Geometric and dosimetric verification of a recurrent neural network algorithm to compensate for respiratory motion using an articulated robotic couch. *Journal of the Korean Physical Society*, 78(1):64–72, 2021.

Suk Jin Lee and Yuichi Motai. *Prediction and classification of respiratory motion.* Springer, 2014.

Suk Jin Lee, Yuichi Motai, and Martin Murphy. Respiratory motion estimation with hybrid implementation of extended Kalman filter. *IEEE Transactions on Industrial Electronics*, 59(11):4421–4432, 2011.

G Levkine. Prewitt, Sobel and Scharr gradient 5x5 convolution matrices. *Image Process. Articles, Second Draft*, 2012.

Ruijiang Li, Xun Jia, John H Lewis, Xuejun Gu, Michael Folkerts, Chunhua Men, and Steve B Jiang. Real-time volumetric image reconstruction and 3d tumor localization based on a single x-ray projection image for lung cancer radiotherapy. *Medical physics*, 37(6Part1):2822–2826, 2010.

Ruijiang Li, John H. Lewis, Xun Jia, Xuejun Gu, Michael Folkerts, Chunhua Men, William Y. Song, and Steve B. Jiang. 3d tumor localization through real-time volumetric x-ray imaging for lung cancer radiotherapy. *Medical Physics*, 38(5): 2783–2794, 2011a. doi: https://doi.org/10.1118/1.3582693. URL https://aapm.onlinelibrary.wiley.com/doi/abs/10.1118/1.3582693.

Ruijiang Li, John H Lewis, Xun Jia, Tianyu Zhao, Weifeng Liu, Sara Wuenschel, James Lamb, Deshan Yang, Daniel A Low, and Steve B Jiang. On a pca-based lung motion model. *Physics in Medicine & Biology*, 56(18):6009, 2011b.

Shiyang Li, Xiaoyong Jin, Yao Xuan, Xiyou Zhou, Wenhu Chen, Yu-Xiang Wang, and Xifeng Yan. Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting. *arXiv preprint arXiv:1907.00235*, 2019.

Yi Li, Jing-lu Ma, Xin Chen, Feng-wen Tang, and Xiao-zhi Zhang. 4dct and cbct based ptv margin in stereotactic body radiotherapy (sbrt) of non-small cell lung tumor adhered to chest wall or diaphragm. *Radiation Oncology*, 11(1):1–9, 2016.

Renjie Liao, Yuwen Xiong, Ethan Fetaya, Lisa Zhang, KiJung Yoon, Xaq Pitkow, Raquel Urtasun, and Richard Zemel. Reviving and improving recurrent back-propagation. In *International Conference on Machine Learning*, pages 3082–3091. PMLR, 2018.

Bryan Lim, Ahmed Alaa, and Mihaela van der Schaar. Forecasting treatment responses over time using recurrent marginal structural networks. *NeurIPS*, 18: 7483–7493, 2018.

Bryan Lim, Sercan O Arik, Nicolas Loeff, and Tomas Pfister. Temporal fusion transformers for interpretable multi-horizon time series forecasting. *arXiv preprint arXiv:1912.09363*, 2019.

Hui Lin, Chengyu Shi, Brian Wang, Maria F Chan, Xiaoli Tang, and Wei Ji. Towards real-time respiratory motion prediction based on long short-term memory neural networks. *Physics in Medicine & Biology*, 64(8):085010, 2019.

Tong Lin, Laura I Cervino, Xiaoli Tang, Nuno Vasconcelos, and Steve B Jiang. Fluoroscopic tumor tracking for image-guided lung cancer radiotherapy. *Physics in Medicine & Biology*, 54(4):981, 2009.

W Liu, A Sawant, and D Ruan. Th-cd-207a-07: Prediction of high dimensional state subject to respiratory motion: A manifold learning approach. *Medical Physics*, 43(6Part45):3881–3881, 2016a.

Wenyang Liu, Amit Sawant, and Dan Ruan. Prediction of high-dimensional states subject to respiratory motion: a manifold learning approach. *Physics in Medicine & Biology*, 61(13):4989, 2016b.

Ziwei Liu, Raymond A Yeh, Xiaoou Tang, Yiming Liu, and Aseem Agarwala. Video frame synthesis using deep voxel flow. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4463–4471, 2017.

William Lotter, Gabriel Kreiman, and David Cox. Deep predictive coding networks for video prediction and unsupervised learning. *arXiv preprint arXiv:1605.08104*, 2016.

William Lotter, Gabriel Kreiman, and David Cox. Deep predictive coding networks for video prediction and unsupervised learning, 2017.

D. Low, S. Mutic, T. Chmielewski, G. Fought, G. Gerganov, M. Hernandez, I. Kawrakow, E. Kishawi, G. Kuduvalli, A. Sharma, S. Shvartsman, J. Tabachnik, and J.F. Dempsey. The physics of a novel compact linear accelerator-based magnetic resonance imaging-guided radiation therapy system. *International Journal of Radiation Oncology*Biology*Physics*, 96(2, Supplement):E634, 2016. ISSN 0360-3016. doi: https://doi.org/10.1016/j.ijrobp.2016.06.2217. URL https://www.sciencedirect.com/science/article/pii/S0360301616325433. Proceedings of the American Society for Radiation Oncology.

Daniel A Low and James F Dempsey. Evaluation of the gamma dose distribution comparison method. *Medical physics*, 30(9):2455–2464, 2003.

Daniel A Low, William B Harms, Sasa Mutic, and James A Purdy. A technique for the quantitative evaluation of dose distributions. *Medical physics*, 25(5):656–661, 1998.

Francesco Lässig. Temporal convolutional networks and forecasting. https://medium.com/unit8-machine-learning-publication/

`temporal-convolutional-networks-and-forecasting-5ce1b6e97ce4`, 2020. [Online; accessed 9-May-2021].

Pauline Luc, Camille Couprie, Yann Lecun, and Jakob Verbeek. Predicting future instance segmentation by forecasting convolutional features. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 584–599, 2018.

Bruce D Lucas, Takeo Kanade, et al. An iterative image registration technique with an application to stereo vision. 1981.

Meiyi Ma, Ji Gao, Lu Feng, and John Stankovic. Stlnet: Signal temporal logic enforced multivariate recurrent neural networks. *Advances in Neural Information Processing Systems*, 33, 2020.

Zhao Ma, Wei Zhang, Yi Su, Peiji Liu, Yinghua Pan, Gang Zhang, and Yipeng Song. Optical surface management system for patient positioning in interfractional breast cancer radiotherapy. *BioMed research international*, 2018, 2018.

Majid Mafi and Saeed Montazeri Moghadam. Real-time prediction of tumor motion using a dynamic neural network. *Medical & biological engineering & computing*, 58(3):529–539, 2020.

Catalina Margulis. High-tech tool will target tumours with extraordinary precision. `https://health.sunnybrook.ca/magazine/spring-2016/mr-linac-cancer-imaging-radiation/`, 2016. [Online; accessed 23-June-2021].

Owen Marschall, Kyunghyun Cho, and Cristina Savin. A unified framework of online learning algorithms for training recurrent neural networks. *Journal of Machine Learning Research*, 21(135):1–34, 2020.

Pierre-Yves Massé and Yann Ollivier. Convergence of online adaptive and recurrent optimization algorithms. *arXiv preprint arXiv:2005.05645*, 2020.

Michael Mathieu, Camille Couprie, and Yann LeCun. Deep multi-scale video prediction beyond mean square error. *arXiv preprint arXiv:1511.05440*, 2015.

Sarah A. Mattonen, David A. Palma, Carol Johnson, Alexander V. Louie, Mark Landis, George Rodrigues, Ian Chan, Roya Etemad-Rezai, Timothy P.C. Yeung, Suresh Senan, and Aaron D. Ward. Detection of local cancer recurrence after stereotactic ablative radiation therapy for lung cancer: Physician performance versus radiomic assessment. *International Journal of Radiation Oncology*Biology*Physics*, 94(5):1121–1128, 2016. ISSN 0360-3016. doi: https://doi.org/10.1016/j.ijrobp.2015.12.369. URL `https://www.sciencedirect.com/science/article/pii/S0360301615272239`.

Jamie R McClelland, Jane M Blackall, Ségolene Tarte, Adam C Chandler, Simon Hughes, Shahreen Ahmad, David B Landau, and David J Hawkes. A continuous 4D motion model from multiple respiratory cycles for use in lung radiotherapy. *Medical Physics*, 33(9):3348–3358, 2006.

Jamie R McClelland, David J Hawkes, Tobias Schaeffter, and Andrew P King. Respiratory motion models: a review. *Medical image analysis*, 17(1):19–42, 2013.

Andrew M. McDonald, Tyler Colvin, D. Hunter Boggs, Sharon A. Spencer, Richard A. Popple, Ravinder Clayton, Douglas Minnich, and Michael C. Dobelbower. Longitudinal assessment of anchored transponder migration following lung stereotactic body radiation therapy. *Journal of Applied Clinical Medical Physics*, 20(1):17–22, 2019. doi: https://doi.org/10.1002/acm2.12454. URL https://aapm.onlinelibrary.wiley.com/doi/abs/10.1002/acm2.12454.

Jacob Menick, Erich Elsen, Utku Evci, Simon Osindero, Karen Simonyan, and Alex Graves. A practical sparse approximation for real time recurrent learning. *arXiv preprint arXiv:2006.07232*, 2020.

Martin J. Menten, Martin F. Fast, Simeon Nill, Cornelis P. Kamerling, Fiona McDonald, and Uwe Oelfke. Lung stereotactic body radiotherapy with an mr-linac -quantifying the impact of the magnetic field and real-time tumor tracking. *Radiotherapy and Oncology*, 119(3):461–466, 2016. ISSN 0167-8140. doi: https://doi.org/10.1016/j.radonc.2016.04.019. URL https://www.sciencedirect.com/science/article/pii/S0167814016310593.

Philippe Meyer, Vincent Noblet, Christophe Mazzara, and Alex Lallement. Survey on deep learning for radiotherapy. *Computers in biology and medicine*, 98:126–146, 2018.

Payam Samadi Miandoab, Ahmad Esmaili Torshabi, and Saber Nankali. Investigation of the optimum location of external markers for patient setup accuracy enhancement at external beam radiotherapy. *Journal of applied clinical medical physics*, 17(6):32–43, 2016.

Oana Mihalcea and AC Arnold. Side effect of head and neck radiotherapy: optic neuropathy. *Oftalmologia (Bucharest, Romania: 1990)*, 52(1):36–40, 2008.

Asier Mujika, Florian Meier, and Angelika Steger. Approximating real-time recurrent learning with random kronecker factors. *Advances in Neural Information Processing Systems*, 31:6594–6603, 2018.

Martin J Murphy. Tracking moving organs in real time. In *Seminars in radiation oncology*, volume 14, pages 91–100. Elsevier, 2004.

Martin J Murphy and Sonja Dieterich. Comparative performance of linear and nonlinear neural networks to predict irregular breathing. *Physics in Medicine & Biology*, 51(22):5903, 2006.

Martin J Murphy and Damodar Pokhrel. Optimization of an adaptive neural network to predict breathing. *Medical physics*, 36(1):40–47, 2009.

James M Murray. Local online learning in recurrent networks with random feedback. *ELife*, 8:e43299, 2019.

Panakis N. *Radiotherapy to the lung, Information for patients.* Oxford University Hospitals NHS Trust, 2015. URL https://www.ouh.nhs.uk/patient-guide/leaflets/files/4522Plung.pdf.

Shahabedin Nabavi, Monireh Abdoos, Mohsen Ebrahimi Moghaddam, and Mohammad Mohammadi. Respiratory motion prediction using deep convolutional long short-term memory network. *Journal of Medical Signals and Sensors*, 10(2):69, 2020.

Dong Nie, Xiaohuan Cao, Yaozong Gao, Li Wang, and Dinggang Shen. Estimating ct image from mri data using 3d fully convolutional networks. In Gustavo Carneiro, Diana Mateus, Loïc Peter, Andrew Bradley, João Manuel R. S. Tavares, Vasileios Belagiannis, João Paulo Papa, Jacinto C. Nascimento, Marco Loog, Zhi Lu, Jaime S. Cardoso, and Julien Cornebise, editors, *Deep Learning and Data Labeling for Medical Applications*, pages 170–178, Cham, 2016. Springer International Publishing. ISBN 978-3-319-46976-8.

Stanislav Nikolov, Sam Blackwell, Ruheena Mendes, Jeffrey De Fauw, Clemens Meyer, Cían Hughes, Harry Askham, Bernardino Romera-Paredes, Alan Karthikesalingam, Carlton Chu, et al. Deep learning to achieve clinically applicable segmentation of head and neck anatomy for radiotherapy. *arXiv preprint arXiv:1809.04430*, 2018.

Yann Ollivier, Corentin Tallec, and Guillaume Charpiat. Training recurrent networks online without backtracking. *arXiv preprint arXiv:1507.07680*, 2015.

Frank Ong, Xucheng Zhu, Joseph Y Cheng, Kevin M Johnson, Peder EZ Larson, Shreyas S Vasanawala, and Michael Lustig. Extreme mri: Large-scale volumetric dynamic imaging from continuous non-gated acquisitions. *Magnetic resonance in medicine*, 84(4):1763–1780, 2020.

Sergiu Oprea, Pablo Martinez-Gonzalez, Alberto Garcia-Garcia, John Alejandro Castro-Vargas, Sergio Orts-Escolano, Jose Garcia-Rodriguez, and Antonis Argyros. A review on deep learning techniques for video prediction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

C Paganelli, S Portoso, N Garau, G Meschini, R Via, G Buizza, P Keall, M Riboldi, and G Baroni. Time-resolved volumetric mri in mri-guided radiotherapy: an in silico comparative analysis. *Physics in Medicine & Biology*, 64(18):185013, 2019.

Christopher C Paige and Michael A Saunders. Lsqr: An algorithm for sparse linear equations and sparse least squares. *ACM Transactions on Mathematical Software (TOMS)*, 8(1):43–71, 1982.

Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. On the difficulty of training recurrent neural networks. In *International conference on machine learning*, pages 1310–1318, 2013.

Beatriz Pérez-Sánchez, Oscar Fontenla-Romero, and Bertha Guijarro-Berdiñas. A review of adaptive online learning for artificial neural networks. *Artificial Intelligence Review*, 49(2):281–299, 2018.

Federica Pirro, Valentina Silvestri, Giancarlo Savino, Riccardo Marano, Biagio Merlino, Agostino Meduri, Lorenzo Bonomo, et al. Chest x-ray cardiac anatomy and pathology: correlation with angiocardiography, ct, and mr imaging. European Congress of Radiology-ECR 2013, 2013.

Michel Pohl, Mitsuru Uesaka, Kazuyuki Demachi, and Ritu Bhusal Chhatkuli. Prediction of the motion of chest internal points using a recurrent neural network trained with real-time recurrent learning for latency compensation in lung cancer radiotherapy. *Computerized Medical Imaging and Graphics*, page 101941, 2021a. ISSN 0895-6111. doi: https://doi.org/10.1016/j.compmedimag.2021.101941. URL https://www.sciencedirect.com/science/article/pii/S0895611121000902.

Michel Pohl, Mitsuru Uesaka, Hiroyuki Takahashi, Kazuyuki Demachi, and Ritu Bhusal Chhatkuli. Prediction of the position of external markers using a recurrent neural network trained with unbiased online recurrent optimization for safe lung cancer radiotherapy. *arXiv preprint arXiv:2106.01100*, 2021b.

Per Rugaard Poulsen, Byungchul Cho, Amit Sawant, Dan Ruan, and Paul J Keall. Detailed analysis of latencies in image-based dynamic MLC tracking. *Medical physics*, 37(9):4998–5005, 2010.

B W Raaymakers, J J W Lagendijk, J Overweg, J G M Kok, A J E Raaijmakers, E M Kerkhof, R W van der Put, I Meijsing, S P M Crijns, F Benedosso, M van Vulpen, C H W de Graaff, J Allen, and K J Brown. Integrating a 1.5 t MRI scanner with a 6 MV accelerator: proof of concept. *Physics in Medicine and Biology*, 54(12):N229–N237, may 2009. doi: 10.1088/0031-9155/54/12/n01. URL https://doi.org/10.1088/0031-9155/54/12/n01.

Radiofísica con Valdo (Youtube). S1 v9.- dose-volume histograms. https://www.youtube.com/watch?v=mLtEIGHn1sc, Dec 2019.

David Roberge and Tatiana Cabrera. Percutaneous liver fiducial implants: Techniques, materials and complications. *Liver Biopsy in Modern Medicine*, 2011. URL http://www.intechopen.com/books/liver-biopsy-in-modern-medicine/percutaneous-liverfiducial-implants-techniques-materials-and-complications.

Liset Vázquez Romaguera, Rosalie Plantefève, Francisco Perdigón Romero, François Hébert, Jean-François Carrier, and Samuel Kadoury. Prediction of in-plane organ deformation during free-breathing radiotherapy via discriminative spatial transformer networks. *Medical image analysis*, 64:101754, 2020.

T.W. Roques. Patient selection and radiotherapy volume definition −can we improve the weakest links in the treatment chain? *Clinical Oncology*, 26(6):353–355, 2014. ISSN 0936-6555. doi: https://doi.org/10.1016/j.clon.2014.02.013. URL https://www.sciencedirect.com/science/article/pii/S093665551400082X.

Christopher Roth, Ingmar Kanitscheider, and Ila Fiete. Kernel rnn learning (kernl). In *International Conference on Learning Representations*, 2018.

Dan Ruan. Kernel density estimation-based real-time prediction for respiratory motion. *Physics in Medicine and Biology*, 55(5):1311–1326, feb 2010. doi: 10.1088/0031-9155/55/5/004. URL https://doi.org/10.1088/0031-9155/55/5/004.

Doyen Sahoo, Quang Pham, Jing Lu, and Steven CH Hoi. Online deep learning: Learning deep neural networks on the fly. *arXiv preprint arXiv:1711.03705*, 2017.

Hojjat Salehinejad, Sharan Sankar, Joseph Barfett, Errol Colak, and Shahrokh Valaee. Recent advances in recurrent neural networks. *arXiv preprint arXiv:1801.01078*, 2017.

David Salinas, Valentin Flunkert, Jan Gasthaus, and Tim Januschowski. Deepar: Probabilistic forecasting with autoregressive recurrent networks. *International Journal of Forecasting*, 36(3):1181–1191, 2020.

Achim Schweikard, Greg Glosser, Mohan Bodduluri, Martin J Murphy, and John R Adler. Robotic motion compensation for respiratory movement during radiosurgery. *Computer Aided Surgery: Official Journal of the International Society for Computer Aided Surgery (ISCAS)*, 5(4):263–277, 2000.

Achim Schweikard, Hiroya Shiomi, and John Adler. Respiration tracking in radiosurgery. *Medical physics*, 31(10):2738–2741, 2004.

Yvette Seppenwoolde, Ross I Berbeco, Seiko Nishioka, Hiroki Shirato, and Ben Heijmen. Accuracy of tumor motion compensation algorithm from a robotic respiratory tracking system: a simulation study. *Medical physics*, 34(7):2774–2784, 2007.

Omer Berat Sezer, Mehmet Ugur Gudelek, and Ahmet Murat Ozbayoglu. Financial time series forecasting with deep learning: A systematic literature review: 2005–2019. *Applied Soft Computing*, 90:106181, 2020.

Bobak Shahriari, Kevin Swersky, Ziyu Wang, Ryan P Adams, and Nando De Freitas. Taking the human out of the loop: A review of bayesian optimization. *Proceedings of the IEEE*, 104(1):148–175, 2015.

Nusrat Sharmin and Remus Brad. Optimal filter estimation for Lucas-Kanade optical flow. *Sensors*, 12(9):12694–12709, 2012.

Gregory C Sharp, Steve B Jiang, Shinichi Shimizu, and Hiroki Shirato. Prediction of respiratory tumour motion for real-time image-guided radiotherapy. *Physics in Medicine & Biology*, 49(3):425, 2004.

Xingjian Shi, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-Kin Wong, and Wang-chun Woo. Convolutional lstm network: A machine learning approach for precipitation nowcasting. *arXiv preprint arXiv:1506.04214*, 2015.

Chun-Chien Shieh, Vincent Caillet, Michelle Dunbar, Paul J Keall, Jeremy T Booth, Nicholas Hardcastle, Carol Haddad, Thomas Eade, and Ilana Feain. A bayesian approach for three-dimensional markerless tumor tracking using kv imaging during lung radiotherapy. *Physics in Medicine & Biology*, 62(8):3065, 2017.

Satomi Shiraishi and Kevin L. Moore. Knowledge-based prediction of three-dimensional dose distributions for external beam radiotherapy. *Medical Physics*, 43(1):378–387, 2016. doi: https://doi.org/10.1118/1.4938583. URL https://aapm.onlinelibrary.wiley.com/doi/abs/10.1118/1.4938583.

Hiroki Shirato, Shinichi Shimizu, Tatsuya Kunieda, Kei Kitamura, Marcel Van Herk, Kenji Kagei, Takeshi Nishioka, Seiko Hashimoto, Katsuhisa Fujita, Hidefumi Aoyama, et al. Physical aspects of a real-time tumor-tracking system for gated radiotherapy. *International Journal of Radiation Oncology* Biology* Physics*, 48(4):1187–1195, 2000.

Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012.

Bjorn Stemkens, Rob HN Tijssen, Baudouin Denis De Senneville, Jan JW Lagendijk, and Cornelis AT Van Den Berg. Image-driven, model-based 3d abdominal motion estimation for mr-guided radiotherapy. *Physics in Medicine & Biology*, 61(14): 5335, 2016.

WZ Sun, MY Jiang, L Ren, J Dang, T You, and FF Yin. Respiratory signal prediction based on adaptive boosting and multi-layer perceptron neural network. *Physics in Medicine & Biology*, 62(17):6822, 2017.

Hyuna Sung, Jacques Ferlay, Rebecca L. Siegel, Mathieu Laversanne, Isabelle Soerjomataram, Ahmedin Jemal, and Freddie Bray. Global cancer statistics 2020: Globocan estimates of incidence and mortality worldwide for 36 cancers in 185 countries. *CA: A Cancer Journal for Clinicians*, n/a(n/a), 2020. doi: https://doi. org/10.3322/caac.21660. URL https://acsjournals.onlinelibrary.wiley. com/doi/abs/10.3322/caac.21660.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. *arXiv preprint arXiv:1409.3215*, 2014.

Seishin Takao, Naoki Miyamoto, Taeko Matsuura, Rikiya Onimaru, Norio Katoh, Tetsuya Inoue, Kenneth Lee Sutherland, Ryusuke Suzuki, Hiroki Shirato, and Shinichi Shimizu. Intrafractional baseline shift or drift of lung tumor motion during gated radiation therapy with a real-time tumor-tracking system. *International Journal of Radiation Oncology* Biology* Physics*, 94(1):172–180, 2016.

Corentin Tallec. *Recurrent Neural Networks and Reinforcement Learning: Dynamic Approaches*. PhD thesis, Université Paris-Saclay, 2019.

Corentin Tallec and Yann Ollivier. Unbiased online recurrent optimization. *arXiv preprint arXiv:1702.05043*, 2017a.

Corentin Tallec and Yann Ollivier. Unbiasing truncated backpropagation through time. *arXiv preprint arXiv:1705.08209*, 2017b.

Troy P Teo, Syed Bilal Ahmed, Philip Kawalec, Nadia Alayoubi, Neil Bruce, Ethan Lyn, and Stephen Pistorius. Feasibility of predicting tumor motion using online data acquired during treatment and a generalized neural network optimized with offline patient tumor trajectories. *Medical physics*, 45(2):830–845, 2018.

Toshiyuki Terunuma, Aoi Tokui, and Takeji Sakae. Novel real-time tumor-contouring method using deep learning to prevent mistracking in x-ray fluoroscopy. *Radiological physics and technology*, 11(1):43–53, 2018.

J-P Thirion. Image matching as a diffusion process: an analogy with Maxwell's demons. *Medical Image Analysis*, 2(3):243–260, 1998.

Jean-Philippe Thirion. *Fast Non-Rigid Matching of 3D Medical Images*. PhD thesis, INRIA, 1995.

Virginia Torczon. On the convergence of pattern search algorithms. *SIAM Journal on optimization*, 7(1):1–25, 1997.

Elena H Tran, Björn Eiben, Andreas Wetscherek, Uwe Oelfke, Gustav Meedt, David J Hawkes, and Jamie R McClelland. Evaluation of mri-derived surrogate signals to model respiratory motion. *Biomedical Physics & Engineering Express*, 6(4):045015, 2020.

Huan-Hsin Tseng, Yi Luo, Sunan Cui, Jen-Tzung Chien, Randall K. Ten Haken, and Issam El Naqa. Deep reinforcement learning for automated radiation adaptation in lung cancer. *Medical Physics*, 44(12):6690–6705, 2017. doi: https://doi.org/10.1002/mp.12625. URL https://aapm.onlinelibrary.wiley.com/doi/abs/10.1002/mp.12625.

Alexandre B Tsybakov. *Introduction to nonparametric estimation*. Springer Science & Business Media, 2008.

Gilmer Valdes, Olivier Morin, Yanisley Valenciaga, Niel Kirby, Jean Pouliot, and Cynthia Chuang. Use of truebeam developer mode for imaging qa. *Journal of Applied Clinical Medical Physics*, 16(4):322–333, 2015. doi: https://doi.org/10.1120/jacmp.v16i4.5363. URL https://aapm.onlinelibrary.wiley.com/doi/abs/10.1120/jacmp.v16i4.5363.

Gilmer Valdes, Charles B. Simone, Josephine Chen, Alexander Lin, Sue S. Yom, Adam J. Pattison, Colin M. Carpenter, and Timothy D. Solberg. Clinical decision support of radiotherapy treatment planning: A data-driven machine learning strategy for patient-specific dosimetric decision making. *Radiotherapy and Oncology*, 125(3):392–397, 2017. ISSN 0167-8140. doi: https://doi.org/10.1016/j.radonc.2017.10.014. URL https://www.sciencedirect.com/science/article/pii/S0167814017326543.

Aaron Van Oord, Nal Kalchbrenner, and Koray Kavukcuoglu. Pixel recurrent neural networks. In *International Conference on Machine Learning*, pages 1747–1756. PMLR, 2016.

John R. van Sörnsen de Koste, Miguel A. Palacios, Anna M.E. Bruynzeel, Ben J. Slotman, Suresh Senan, and Frank J. Lagerwaard. Mr-guided gated stereotactic radiation therapy delivery for lung, adrenal, and pancreatic tumors: A geometric analysis. *International Journal of Radiation Oncology*Biology*Physics*, 102(4):858–866, 2018. ISSN 0360-3016. doi: https://doi.org/10.1016/j.ijrobp.

2018.05.048. URL https://www.sciencedirect.com/science/article/pii/S036030161830899X. Imaging in Radiation Oncology.

Poonam Verma, Huanmei Wu, Mark Langer, Indra Das, and George Sandison. Survey: real-time tumor motion prediction for image-guided radiation treatment. *Computing in Science & Engineering*, 13(5):24–35, 2010.

Ruben Villegas, Jimei Yang, Seunghoon Hong, Xunyu Lin, and Honglak Lee. Decomposing motion and content for natural video sequence prediction. *arXiv preprint arXiv:1706.08033*, 2017.

Martin von Siebenthal, Gabor Szekely, Urs Gamper, Peter Boesiger, Antony Lomax, and Ph Cattin. 4d mr imaging of respiratory organ motion and its variability. *Physics in Medicine & Biology*, 52(6):1547, 2007.

Guangyu Wang, Zhibin Li, Guangjun Li, Guyu Dai, Qing Xiao, Long Bai, Yisong He, Yaxin Liu, and Sen Bai. Real-time liver tracking algorithm based on lstm and svr networks for use in surface-guided radiation therapy. *Radiation Oncology*, 16 (1):1–12, 2021.

Huaizhi Wang, Zhenxing Lei, Xian Zhang, Bin Zhou, and Jianchun Peng. A review of deep learning for renewable energy forecasting. *Energy Conversion and Management*, 198:111799, 2019. ISSN 0196-8904. doi: https://doi.org/10.1016/j.enconman.2019.111799. URL https://www.sciencedirect.com/science/article/pii/S0196890419307812.

Ran Wang, Xiaokun Liang, Xuanyu Zhu, and Yaoqin Xie. A feasibility of respiration prediction based on deep bi-lstm for real-time tumor tracking. *IEEE Access*, 6: 51262–51268, 2018.

Yubo Wang, Zhibin Yu, Tatinati Sivanagaraja, and Kalyana C. Veluvolu. Fast and accurate online sequential learning of respiratory motion with random convolution nodes for radiotherapy applications. *Applied Soft Computing*, 95:106528, 2020. ISSN 1568-4946. doi: https://doi.org/10.1016/j.asoc.2020.106528. URL https://www.sciencedirect.com/science/article/pii/S1568494620304671.

Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

Nan Wei, Changjun Li, Xiaolong Peng, Fanhua Zeng, and Xinqian Lu. Conventional models and artificial intelligence-based models for energy consumption forecasting: A review. *Journal of Petroleum Science and Engineering*, 181:106187, 2019.

Wikipedia contributors. Principal component analysis — Wikipedia, the free encyclopedia, 2018. URL https://en.wikipedia.org/w/index.php?title=Principal_component_analysis&oldid=1029739653. [Online; accessed 9-July-2018].

Wikipedia contributors. Sobel operator — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Sobel_operator&oldid=950766970, 2020. [Online; accessed 7-May-2020].

Wikipedia contributors. Nonlinear dimensionality reduction — Wikipedia, the free encyclopedia, 2021. URL https://en.wikipedia.org/w/index.php?title=Nonlinear_dimensionality_reduction&oldid=1028252724. [Online; accessed 14-June-2021].

Martin J Willemink and Peter B Noël. The evolution of image reconstruction for ct —from filtered back projection to artificial intelligence. *European radiology*, 29(5): 2185–2195, 2019.

Ronald J Williams and Jing Peng. An efficient gradient-based algorithm for on-line training of recurrent network trajectories. *Neural computation*, 2(4):490–501, 1990.

Ronald J Williams and David Zipser. A learning algorithm for continually running fully recurrent neural networks. *Neural computation*, 1(2):270–280, 1989.

M Wilms, R Werner, J Ehrhardt, A Schmidt-Richberg, HP Schlemmer, and H Handels. Multivariate regression approaches for surrogate-based diffeomorphic estimation of respiratory motion in radiation therapy. *Physics in Medicine & Biology*, 59(5):1147, 2014.

Jacob S Witt, Stephen A Rosenberg, and Michael F Bassetti. Mri-guided adaptive radiotherapy for liver tumours: visualising the future. *The Lancet Oncology*, 21 (2):e74–e82, 2020.

Jelmer M. Wolterink, Anna M. Dinkla, Mark H. F. Savenije, Peter R. Seevinck, Cornelis A. T. van den Berg, and Ivana Išgum. Deep mr to ct synthesis using unpaired data. In Sotirios A. Tsaftaris, Ali Gooya, Alejandro F. Frangi, and Jerry L. Prince, editors, *Simulation and Synthesis in Medical Imaging*, pages 14–23, Cham, 2017. Springer International Publishing. ISBN 978-3-319-68127-6.

Sifan Wu, Xi Xiao, Qianggang Ding, Peilin Zhao, Ying Wei, and Junzhou Huang. Adversarial sparse transformer for time series forecasting. *Advances in Neural Information Processing Systems*, 33, 2020.

Qianyi Xu, Russell J Hamilton, Robert A Schowengerdt, Brian Alexander, and Steve B Jiang. Lung tumor tracking in fluoroscopic video based on optical flow. *Medical physics*, 35(12):5351–5359, 2008.

Zhong Xue, Ramiro Pino, and Bin Teh. Estimating lung respiratory motion using combined global and local statistical models. In *International Workshop on Patch-based Techniques in Medical Imaging*, pages 133–140. Springer, 2016.

Shumei Yu, Jiateng Wang, Jinguo Liu, Rongchuan Sun, Shaolong Kuang, and Lining Sun. Rapid prediction of respiratory motion based on bidirectional gated recurrent unit network. *IEEE Access*, 8:49424–49435, 2020.

Jing Yuan, Oi Lei Wong, Yihang Zhou, Kin Yin Chueng, and Siu Ki Yu. A fast volumetric 4d-mri with sub-second frame rate for abdominal motion monitoring and characterization in mri-guided radiotherapy. *Quantitative Imaging in Medicine and Surgery*, 9(7), 2019. ISSN 2223-4306. URL https://qims.amegroups.com/article/view/26960.

J Yun, S Rathee, and BG Fallone. A deep-learning based 3D tumor motion prediction algorithm for non-invasive intra-fractional tumor-tracked radiotherapy (nifteRT) on Linac-MR. *International Journal of Radiation Oncology, Biology, Physics*, 105(1):S28, 2019.

Jihyun Yun, Eugene Yip, Zsolt Gabos, Keith Wachowicz, Satyapal Rathee, and BG Fallone. Neural-network based autocontouring algorithm for intrafractional lung-tumor tracking using linac-mr. *Medical physics*, 42(5):2296–2310, 2015.

Geoffrey Zhang, Tzung-Chi Huang, Thomas Guerrero, Kang-Ping Lin, Craig Stevens, George Starkschall, and Ken Forster. Use of three-dimensional (3D) optical flow method in mapping 3D anatomic structure and tumor contours across four-dimensional computed tomography data. *Journal of applied clinical medical physics*, 9(1):59–69, 2008.

Qinghui Zhang, Alex Pevsner, Agung Hertanto, Yu-Chi Hu, Kenneth E Rosenzweig, C Clifton Ling, and Gig S Mageras. A patient-specific respiratory model of anatomical motion for radiation treatment planning. *Medical physics*, 34(12): 4772–4781, 2007.

Xiaoyong Zhang, Noriyasu Homma, Kei Ichiji, Makoto Abe, Norihiro Sugita, Yoshihiro Takai, Yuichiro Narita, and Makoto Yoshizawa. A kernel-based method for markerless tumor tracking in kv fluoroscopic images. *Physics in Medicine & Biology*, 59(17):4897, 2014.

Yufan Zhou, Haiwei Dong, and Abdulmotaleb El Saddik. Deep learning in next-frame prediction: a benchmark review. *IEEE Access*, 8:69273–69283, 2020.

Taiyu Zhu, Kezhi Li, Jianwei Chen, Pau Herrero, and Pantelis Georgiou. Dilated recurrent neural networks for glucose forecasting in type 1 diabetes. *Journal of Healthcare Informatics Research*, 4(3):308–324, 2020.