Unterschrift BetreuerIn

BACHELORARBEIT

Software implementation of the quality assurance tool for magnetic resonance imaging distortion assessment

Betreuer:

Ao.Univ.Prof. Dipl.-Ing. Dr.techn. Martin Gröschl
Institut für Angewandte Physik
Technische Universität Wien

in Kooperation mit:

Peter Kuess, PhD
& Piotr Andrzejewski, MSc Eng
Department of Radiooncology
Medical University Vienna, AKH Vienna

von:

David Blacher
1327545

Technische Physik
E033261

October 20, 2017

Unterschrift StudentIn

Contents

1 Introduction **1**

1.1 Photon - matter interactions . . . 1

1.2 Basics of Radiobiology . . . 3

1.2.1 The human cell . . . 3

1.2.2 Effects of radiation . . . 4

1.3 Imaging modalities . . . 6

1.3.1 X-ray projection imaging . . . 7

1.3.2 Computer Tomography - CT . . . 8

1.3.3 Magnetic Resonance Imaging - MRI . . . 12

1.4 Radiation therapy . . . 19

1.4.1 Role of CT . . . 23

1.4.2 Role of MRI . . . 23

1.5 Aim of this work . . . 23

2 Material and methods **25**

2.1 Scanners and Imaging protocols . . . 25

2.1.1 MR . . . 25

2.1.2 CT . . . 27

2.1.3 MRI scanner - field distribution . . . 27

2.2 Custom build phantom . . . 32

2.2.1 Frame and rods . . . 32

2.2.2 Rod fillings . . . 36

2.2.3 Handling of bubbles . . . 37

2.3 Used software applications . . . 37

2.4 Pre-processing MRI and CT scans . . . 37

2.5 Developed software tool . . . 41

2.5.1 Calculation: dice coefficient (DC) . . . 42

2.5.2 Calculation: warp & centre of mass (COM) . . . 44

2.5.3 Detection of irregularities . . . 49

3 Results **53**

3.1 Obtained MRI and CT scans . . . 53

3.2 Tested solutions . . . 56

3.2.1 Visibility on CT/MRI scans . . . 56

3.2.2 Mechanical properties of solutions . . . 57
3.3 Distortion assessment . . . 60
3.3.1 First Set, Rod #5 . . . 61
3.3.2 rod #16 . . . 66

4 Discussion **71**
4.1 Phantom design . . . 71
4.2 Tested solutions . . . 72
4.2.1 Thoughts about choosing possible candidates . . . 72
4.2.2 Choosing a promising candidate . . . 74
4.3 Distortion . . . 75
4.3.1 Calculation Methods . . . 75
4.3.2 Irregularities . . . 77
4.3.3 Measured distortion - Rod #5 . . . 77
4.3.4 Measured distortion - Rod #16 . . . 77
4.3.5 Effect of resampling . . . 78

5 Conclusion and Outlook **85**
5.1 "filling" for phantom . . . 85
5.2 Recommended resample rates . . . 85
5.3 Future improvement of software tool . . . 86
5.4 Scale of distortion . . . 87

Bibliography **89**

Appendix **97**
Python - Class and functions . . . 98
Python - rod #5 . . . 113
Python - rod #16 . . . 116
Output - rod #5 . . . 119
Output - rod #16 . . . 126

List of Figures

1.1 Mass attenuation coefficient of iron. 3
1.2 Simplified attenuation matrix (4,4,1). Image source: [2] 9
1.3 The spins, initially oriented randomly in space, become aligned either parallel or antiparallel to an externally applied magnetic field B_0. Image source: [2] 13
1.4 MR scanner coil. Image source: Siemens Healthineers 15
1.5 MR image before and after applying build-in distortion correction. Image source: Courtesy Piotr Andrzejewski (unpublished data) 16
1.6 Energy release of ionising radiation 21
1.7 Example of Radiotherapy treatment plan (image source: [23]) 22
1.8 Rendered MR image before and after applying build-in distortion correction. Image source: Courtesy Piotr Andrzejewski (unpublished data) . . 24

2.1 Magnetom C! Image source: [38] 28
2.2 Magnetom C! field strength side view along z-axis, distant view. Image source: [38] 29
2.3 Magnetom C! field gradient side view along z-axis. Image source: [38] . . 30
2.4 Magnetom C! field gradient front view along x-axis. Image source: [38] . 30
2.5 Magnetom C! field gradient top view along y-axis, distant view. Image source: [38] 31
2.6 Comparison: MRI only shows liquid filling, CT also the plastic rod and pane 32
2.7 Rendered CT image of phantom. Image source: Courtesy Piotr Andrzejewski (unpublished data) 33
2.8 Axial view of one of three plastic panes that make up the frame of the phantom. 34
2.9 Schematic of an empty plastic rod 34
2.10 Coronal CT and MRI images showing rods and 2 attached vitamin-e pills 35

2.11 Rendered CT/MRI image of phantom after steps 1-3 of pre-processing. . 39
2.12 Rendered CT/MRI image of phantom after steps 4 and 5 of pre-processing. 39
2.13 Rendered image of phantom after steps 4 and 5 of pre-processing. 40
2.14 `DC` calculation for a MR scan (rod #5 @ slice 200) yielding a value of 0.9454 45
2.15 Simple method to find `threshold` for `COM` calculation based on real rod size (inner diameter: $2\,mm$, outer diameter: $4\,mm$) 46
2.16 Iteration method to find `threshold`, `DC` and `COM` 47
2.17 COM iteration method, 5 repetitions. 50
2.18 Check for irregularities 51

3.1 Coronal images of 16 rods fileld with tested liquids. 54
3.2 Axial images of 16 rods fileld with tested liquids. 55
3.3 Rod #5 showed some bubbles after 2 months. 59
3.4 Rod #16 contained no bubbles after more than 6 months. 59
3.5 The rest of the tested rods (all except #5 and #16) after more than 6 months. 60
3.6 Rod #5: brightness along z-axis, CT x100. 61
3.7 Rod #5: brightness along z-axis, MRI x100. 63
3.8 Rod #5: warp XY (iteration method), CT-MRI x100. 63
3.9 Rod #5: warp Magnitude (iteration method), CT-MRI x100. 64
3.10 Rod #5: DC (iteration method) for CT & MRI & MRI (using CT COM), x100. 64
3.11 Rod #5: images showing the location of MRI/CT centroids at various slices along z-axis; x100 65
3.12 Rod #16: colour coded map; colours reflect $warpM^*$; based on x100 . . . 66
3.13 Rod #16: brightness along z-axis, CT x100. 66
3.14 Rod #16: brightness along z-axis, MRI x100. 68
3.15 Rod #16: warp XY (iteration method), CT-MRI x100. 68
3.16 Rod #16: warp Magnitude (iteration method), CT-MRI x100. 69
3.17 Rod #16: DC (iteration method) for CT & MRI & MRI (using CT COM), x100. 69

4.1 Slices at edge of plastic pane in CT image showing rod #5 (true colours): a black crescent remains at the bottom where there is a hole in the pane that is not filled with a rod at the moment 76

4.2 Rod #5: warp Magnitude [mm] (iteration method), CT-MRI x1 and x100; plastic pane from $-10mm$ to $10mm$, bubble at about $110mm$ to $135mm$ 80
4.3 Rod #5; x1: DC (iteration method) for CT & MRI & MRI (using CT COM); plastic pane from $-10mm$ to $10mm$, bubble at about $110mm$ to $135mm$ 80
4.4 Rod #5; x4: DC (iteration method) for CT & MRI & MRI (using CT COM); plastic pane from $-10mm$ to $10mm$, bubble at about $110mm$ to $135mm$ 81
4.5 Rod #16: warp Magnitude [mm] (iteration method), CT-MRI x1 and x100; plastic pane from $-8mm$ to $12mm$ 82
4.6 Rod #16; x1: DC (iteration method) for CT & MRI & MRI (using CT COM); plastic pane from $-8mm$ to $12mm$ 83
4.7 Rod #16; x4: DC (iteration method) for CT & MRI & MRI (using CT COM); plastic pane from $-8mm$ to $12mm$ 83

List of Tables

1.1 Average HU values for various types of human tissue 10

2.1 MR scanner and used protocol 26
2.2 CT scanner and used protocol 27
2.3 The composition of all tested solutions. (components in g/L; exception: Primovist in volume-%) 36
2.4 pixel Spacing (rounded values) [mm] 38

3.1 Liquid visibility on MRI scan. 56
3.2 Observations made in the first week regarding the mechanical properties of the tested solutions 58
3.3 Measured air accumulated in observed rods. 59
3.4 Rod #5: script generated data. 62
3.5 Rod #16: script generated data. 67

4.1 Expectancy and standard deviation of difference between calculated distortion for x1 and x100 resample rate; rod #5 79
4.2 Expectancy and standard deviation of difference between calculated distortion for x1 and x100 resample rate; rod #16 82

Kurzzusammenfassung

Ziel dieser Arbeit war die Entwicklung einer Software zur Berechnung der geometischen Verzerrung eines $0,35\,T$ MRT Scanners. Für den Erfolg von Tele- und Brachytherapie ist die Genauigkeit der genutzten bildgebenden Verfahren bei der Erstellung von Bestrahlungsplänen mitbestimmend. CT Aufnahmen sind geometrisch unverzerrt, und können daher zur Korrektur von MRT Aufnahmen verwendet werden. So kann die erhöhte Genauigkeit bei der Abgrenzung von Tumoren mithilfe des hohen Weichteilkontrastes bei MRT Bildern in die Berechnung einfließen. Moderne MRT scanner sind mit Algorithmen ausgestattet, die einen Großteil der Verzerrung korrigieren. Durch den Verzicht auf CT würde nicht nur Zeit und Geld eingespart, sondern auch die dem Patienten zugeführte Strahlenbelastung reduziert werden.

Um die Verzerrung des MRT scanners zu bestimmen, wurden MRT und CT Bilder eines eigens konzipierten und gefertigten Phantoms erstellt und verglichen. Dafür wurden sie zunächst registriert und mit einem Interpolationsverfahren neu berechnet und in höheren Auflösungen gespeichert. Anschließend bestimmte das entwickelte Programm die relative Verschiebung und Verformung unter Verwendung der frei verfügbaren Python Bibliothek SimpleITK. Das Phantom bilden ein Rahmen aus Plexiglass und mehr als 300 Röhren. Da Plastik für MRT unsichtbar ist, wurden die Röhren mit einer geeigneten Flüssigkeit gefüllt. Diese sollte ein ausreichend starkes Signal liefern und auf Dauer keine Gasblasen bilden.

Für die Verwendung als ungünstig erwiesen sich fast alle auf Wasser basierende Flüssigkeiten, da sie einerseits bei zu dünnen Rohrwänden verdampfen, andererseits darin gelöste Gase frei werden können. Bei dickeren Wänden jedoch könnte eine Lösung von $CuSO_4 \cdot 5H_2O$ und $NaCl$ in Wasser mit etwas Seife und Vitamin C die Anforderungen erfüllen. Die gelösten Salze führen zu ausreichender Signalstärke während Vitamin C der Bildung von Sauerstoff-Bläschen entgegen wirkt. Sollten dennoch Blasen entstehen, führt die Seife zu ausreichend Mobilität, sodass sie durch Kippen des Phantoms an ein Ende und somit aus dem Bildbereich bewegt werden können. Die Verzerrung

des $0,35\,T$ MRT scanners konnte nicht bewertet werden, da das Entwickelte Programm vorerst nicht den gesamten Bildbereich evaluieren kann. Für die Berechnung der geometrischen Abweichung wird die Interpolation auf das 4-fache der ursprünglichen Auflösung empfohlen, da dies bei vergleichsweise geringem Rechenaufwand bereits zu signifikanter Verbesserung der Genauigkeit führt.

Abstract

For radiotherapy treatment planning, knowledge concerning the reliability of the utilised imaging modality is crucial. Even though MR scanners promise superior soft tissue contrast, the geometric precision is not as accurate as CT imaging. To assess the spatial distortion of a $0.35\,T$ MR scanner, a software tool was developed. It compares the MR and reference CT images of a custom designed phantom to calculate the occurring deformation and overall shift. This was accomplished with use of the freely available python library SimpleITK. Images obtained from CT and MR scans were registered and resampled to a higher resolution prior to being processed by the software tool. While increased pixel numbers result in longer computing time, interpolated data lead to more detailed information. A 4 times higher resolution is recommended as it strikes a balance between limiting the CPU workload and enhancing the outputs accuracy. As the implemented tool is not yet able to calculate the distortion throughout the whole field of view, a conclusion whether treatment planning based solely on this scanner would be feasible, is not possible at this point.

Additionally to the development of the script, candidates for a suitable liquid of the phantom were produced and tested. Without a filling, the phantom is made up only by acrylic glass parts; namely a frame and more than 300 hollow rods. As the plastic material itself is not visible on MR images, the rods need to be filled with a liquid resulting in acceptable signal intensity. Synthetic oil was found to yield exceptional signal strength while promising long term reliability. In comparison, water based liquids are less suited. Not only do they leak from the rods due to evaporation, but they also contain dissolved gases which lead to the formation of gas bubbles. Possible solutions

dealing with these problems are not ruled out entirely, especially if thicker rod walls were able to eliminate evaporation completely. Adding ascorbic acid to a solution of $CuSO_4 \cdot 5H_2O$ and $NaCl$ might limit the forming of bubbles while soap separates them from the walls so they can easily be moved from the field of view.

The approach taken with this software tool alongside the preparation of a suitable phantom pave the way for a comprehensive analysis of the MR scanner's geometric distortion. Even though more development is necessary, the groundwork for accomplishing this task has been laid.

Preface

Medicine has always been a scientific area where the crossing of different fields of research accounted for breakthroughs leading to a better understanding and, consequently, improved therapeutic methods. The success of modern approaches and the development of future techniques owes to the increased interdisciplinary research and work done by both medical personnel such as doctors and nurses, but also scientists like biologists, chemists, physicists and engineers. As the knowledge about our own human body grows, the problems that we face are becoming more and more complex and call for interdisciplinary expertise. Even though some therapeutic questions appear easy to answer, because they are easily understood on a general level, the actual treatment of a real patient is something entirely different. The creation of a treatment plan becomes exceedingly complex as we try to improve its precision and go towards targeted therapies which are tailor fit to the needs of an individual patient. Improving their quality of life and chance of survival has always come with increased costs and effort as trade-off. To make the best treatments available for everyone and, in the long run, also reduce the cost of used resources: this work shall be a small contribution to this development.

1 Introduction

1.1 Photon - matter interactions

As light (visible and invisible wavelengths) passes through matter, its intensity decreases. This phenomenon is due to photons interacting with electrons, nuclei and their electric fields. All processes either change the direction they travel in, alter their energy, or result in the disappearance of single photons. The probability of these interactions differ for each material (dependent on its density; proton number Z) and photon energy ($h\nu$).

If a photon's energy exceeds the binding energy of an orbital electron, the photoelectric interaction can occur. Also known as 'photo effect', it describes a photon being completely absorbed by a tightly bound orbital electron which then is ejected from its atom. The now free electron is called 'photo-electron'. Its kinetic energy is the difference of the photon's energy and the electron's binding energy:

$$E_{kin} = h\nu - E_{binding} \tag{1.1}$$

Instead of being absorbed, photons might also just 'bounce off' electrons or entire atoms, transferring momentum and, in some cases, part of their energy to the particle they collide with. Rayleigh (coherent) scattering happens when a photon interacts with a tightly bound orbital electron (transferring momentum to the entire atom). This event can be seen as elastic, because only a negligible part of the photon's energy is transferred.

The Compton effect (incoherent scattering) involves a essentially free electron, such as an orbital electron with a relatively small binding energy compared to the photon's energy. Due to the weak binding, momentum is transferred only to the electron. This 'recoil electron' (or 'Compton electron') leaves its atom with a significant kinetic energy,

which originated from the scattered photon. Since the photon loses part of its energy, the event is considered inelastic.

When a photon with an energy above $1.022\,MeV$ passes through the electric field of a nucleus, it might disappear to create an electron-positron pair. This effect is called pair production. The threshold of $1.022\,MeV$ equals exactly the rest mass $E_m = 2m_ec^2$ for the two equally heavy particles (electron and positron). The new particles travel in opposite directions with the same kinetic energy:

$$E_{kin} = \frac{h\nu - 1.022\,MeV}{2} \tag{1.2}$$

A photon with energy of the order of $2\,MeV$ or higher can also interact directly with the nucleus. Such a Photonuclear reaction is similar to the photo effect, in the sense that the photon is completely absorbed. Its energy is transferred to the nucleus resulting in the emission of either a proton or neutron.

Attenuation

The aforementioned interactions result in a gradual decrease of radiation intensity as it travels through matter. The combined effect is described by Beer's law:

$$I(x) = I_0 e^{-\mu(h\nu,Z)x} \tag{1.3}$$

where x is the thickness of a homogeneous material and μ its linear attenuation coefficient. The different probabilities for the interactions to occur is implicitly considered by the attenuation coefficient $\mu(h\nu, Z)$ (see Figure 1.1).

For a photon being transmitted through matter with varying properties, the attenuation coefficient changes, too. After travelling a distance d, the intensity can be expressed as:

$$\tag{1.4}$$

Where $\mu(x)$ describes the attenuation at every distance x. (For whole chapter 1.1: [1], [2])

Figure 1.1: Mass attenuation coefficient of iron. [1]

1.2 Basics of Radiobiology

1.2.1 The human cell

All higher organisms consist of cells working together to form what is called tissue. A collection of tissues which perform one or more functions is considered an organ.

Even though different types of cells exhibit distinctive traits which set them apart, all of them originated from the same totipotent zygote containing a original set of DNA. A zygote is a stem cell, it has the ability to replicate indefinitely, passing its DNA on to the resulting daughter cells. At the same time, it can change into any type of body cell. This feature is why it is called 'totipotent'. As soon as the zygote has divided into a sufficient number of identical cells, all of them differentiate into the various human tissues. In favour of becoming more specialised, cells lose their totipotency. During the early stages of an embryo they are still capable of developing into a number of different cell types, but already restricted to their own tissue type; either nerve, skin, or blood & muscle tissue. As those cells further specialise, they limit their potential even more. In a fully grown human body there are still stem cells present, such as bone marrow stem

[1] image source: by Materialscientist - `https://commons.wikimedia.org/wiki/File:Ironattenuation.PNG`, CC BY-SA 3.0 (`https://creativecommons.org/licenses/by-sa/3.0`) or GFDL (`http://www.gnu.org/copyleft/fdl.html`), via Wikimedia Commons

cells. Other than the zygote, bone marrow can only give rise to blood cells, but not to e.g. nerve or skin cells. A blood cell itself cannot replicate, it is considered a 'mature cell'.

The whole process follows guidelines dictated by the DNA. Every cell inherited its own personal copy of the original set. Inevitably, mistakes happen during its replication resulting in changes to the DNA called 'mutation'. Most of these alterations are repaired or do not lead to changes in the cells behaviour.

The time needed for a reparation process to be completed is not the same as the period between cell reproduction. Changes to the DNA (e.g. mutations) occurring less frequently than one reparation cycle are less probable to result in permanent alterations, than those taking place more frequently. This is the reason why cells in tissues and organs that divide more frequently (i.e. gonads) are more prone to, for example, radiation induced mutations than those reproducing more slowly (i.e. bones).

As the human body ages, the repair mechanism loses some of its efficiency and mutations accumulate. At one point, a cell may be reprogrammed to act in a unpredictable way, giving up its duties and duplicating without restraint, possibly forming tumours. Also, external factors are known to influence cell behaviour and induce such 'malign' cells (carcinogenesis). Cancer cells usually replicate more frequent than healthy cells, eventually leading to characteristic symptoms.

Different approaches have been developed to treat cancer, not all of which are suited to tackle every type of tumour. If the tumour's location is unknown or metastases have formed already in many places, chemotherapy might be considered. An easily accessible tumour could be removed in a surgery. Non invasive therapies also include radiotherapy, destroying cancer cells using ionising radiation. (see chapter 1.2.2.)

Generally, early treatments have high chances of success, but tumours are often not noticed until they reached a certain stage. Reliable ways of diagnosing tumours are made possible by imaging techniques visualising the interior of the human body. [3], [4]

1.2.2 Effects of radiation

Cell damage could either be caused by radiation interacting directly with the critical target in a cell or with other molecules and atoms within the cell. For X-rays, two thirds of the biological damage is attributed to indirect action. As described in 1.1, radiation transfers some of its energy to the medium it passes through. Most interactions, such as the photo effect, incoherent (Compton) scattering and pair production, result in free

electrons. This is why it is called "ionising radiation". If the electron has a sufficiently high kinetic energy, it may free additional orbital electrons from atoms in its vicinity. The remaining ions are positively charged, with a single unpaired valence electron. This type of chemical and free electrons are both referred to as 'radicals' and considered extremely reactive. As the human body consists mainly of water, the radicals created are often H_2O^+ (water ion) and $OH\bullet$ (hydroxyl radical). They are likely to take part in chains of chemical events leading to the breakage of chemical bonds which can disrupt the structure of macromolecules. Such processes can induce changes in DNA sequences and eventually produce biological damage.

A irradiated cell can be affected in various ways ranging from no effect to immediate cell death. The cell might also survive containing a minor mutation. A more fundamental mutation might lead to carcinogenesis. Irradiated cells might also send signals to their neighbours, inducing genetic damage known as 'bystander effects'. On the other hand, surviving cells can also react to irradiation and becoming more resistant.

One classification separates cell damage into lethal, sublethal or potentially lethal. Sublethal damage can be repaired, provided it occurs only once before the repair cycle is complete. Potentially lethal damage can be manipulated by repair, provided the cell is allowed to remain in a non-dividing state.

The relative biological effectiveness (RBE), which describes the damage done by a specific type of radiation (compared to a reference test radiation) to a certain type of tissue, is dependent on various factors, including the rate at which dose is delivered. In the case of radiation causing sublethal damage, for instance, the dose rate significantly affects the RBE. If the average time between two sublethal damages in a single cell is longer than the time necessary for a full repair cycle, the cell will have a fair chance of survival. Does it sustain damage more frequently, the cell will die with a much higher probability. Increasing the radiation rate above this threshold results in a jump of the RBE.

Raising the RBE does not automatically correspond in better treatment. Only if a differential effect reduces the RBE for healthy tissue compared to the tumour, there is a therapeutic advantage. Fortunately, tissues react differently to the same type of radiation. This behaviour can be used to increase the RBE for tumours and reduce it for healthy tissue.

On a larger scale, the sum of damages done to individual cells gives rise to characteristic symptoms. Generally, these harmful effects of radiation are defined as either stochastic or deterministic. The probability of stochastic effects is directly proportional to the dose, but their severity in affected individuals is not. These effects arise in single cells (e.g. carcinogenesis), and it is assumed that the probability for such an occurrence is always greater than zero, even for small doses. If many cells show mutations, the probability of cancer development is higher, but the symptoms of growing tumours will not be worse. For deterministic effects, on the other hand, the severity scales with the dose. They are connected to tissue reaction caused by damage to a population of cells. The higher the dose, the more cells die, the graver are the biological consequences.

Changes to the DNA might not become apparent ever, others take years until they result in biological effects. The same goes for tissue effects, which could either be acute (soon after exposure) or delayed (chronic). A possible long term consequence of ionising radiation is leukaemia. Damage to germ cells (sperm/egg) might even result in genetic damages expressed in subsequent generations.

While imaging modalities utilising X-rays are designed to apply a dose as little as possible to keep effects of irradiation low, radiotherapy makes use of the lethal effects targeting cancer cells. [1], [2]

1.3 Imaging modalities

The imaging of human body's interior has diametrically changed medicine. It has its use in almost all medical disciplines. Currently, there are many ways to acquire section images (or also volumes) of our organism without causing serious and sustained side effects. They differ not only in size of the depicted volume and the image quality, but also in what additional information they provide besides purely morphologic data. These other features can be functional, for example describing effectiveness of a metabolic process; or even molecular, revealing pathways of a certain molecule's distribution. In the next chapters, X-ray planar imaging and the imaging modalities used for this thesis (computer tomography and magnetic resonance imaging) will be explained in more detail.

1.3.1 X-ray projection imaging

A widely used imaging technique based on photon interactions is X-ray projection. Its setup is made up by a radiation source, the object of interest and a detector. Since the technique is about projection, a patient needs to be placed between an X-ray tube and the detector (usually a film-cassette or digital sensor). In the first stage of the imaging process, X-ray photons emitted by the tube enter the body. Next, while travelling through human tissue, they interact with its atoms in various ways as described above (see 1.1). These processes govern how much radiation is absorbed or scattered. Finally, photons which make it through the patient are recorded as they reach the detector on the opposite side. This results in a negative greyscale image, where brightness values correspond to the intensity reduction. Low intensity (= high absorption) leads to bright spots on the image and vice-versa. The whole process could also be described as 'the projection of attenuation shadows onto the detector', since the radiation absorption directly depends on the attenuation coefficient. The attenuation, on the other hand, depend on the tissue's properties (e.g. atomic number Z, density, etc). Consequently, the attenuation shadows depict a projection of the patient.

Soft tissue contrast

Soft tissue such as brain matter and muscles absorb only little radiation, casting a lighter shadow (dark areas on image) than bone which absorbs more photons (bright areas). Practically, in the human body, anything other than bone differs only slightly in attenuation, owing to the relatively small difference in atomic numbers and density. For this reason, X-ray projection imaging is considered reliable when it comes to diagnose bone fractures, while at the same time, it is not suited to clearly delineate soft tissue structures. The use of contrast agents, which effectively increase the density (atomic number) of certain structures or fluids, can help tackle this shortcoming. Such substances fill e.g. the bloodstream with heavier atoms, which can be clearly seen against the dark background of surrounding soft-tissue. In CT angiography, for instance, iodine is administered intravenously enhancing vessel to vessel-wall contrast. In studies of the abdomen a diluted iodine solution or barium compounds swallowed by the patient leads to improved visibility of the gastrointestinal tract. For some examinations, the patient inhales a contrast agent.

For patients allergic to those chemicals, a number of alternative agents have been developed. Unfortunately, most introduce minor, sometimes serious side-effects. There

is ongoing research to find materials yielding enhanced contrast while at the same time minimising adverse reactions, a promising candidate being gold nano particles. [1], [2]

Other imaging modalities are potentially better suited for soft tissue imaging, like medical ultrasound and Magnetic Resonance Imaging (MRI), to name a few. They are preferred for non invasive soft tissue examinations. The choice of suitable imaging modality depends a lot on the particular diagnostic needs and capabilities. It is for the clinician to decide how detailed the information needs to be and how fast it has to be provided. Often less accurate and/or cheaper methods are used first and, if necessary, followed by more sophisticated ones.

1.3.2 Computer Tomography - CT

Computer Tomography (CT) is a three-dimensional (3-D) imaging modality based on the measurement of X-ray planar projections. The technique has evolved from 2-D X-ray scanning. By mounting source and detector on a rotary ring with a patient at the centre, projections from any angle can be obtained. However, in contrast to 2-D projection methods, the detector resembles an arc made up by several hundreds of neighbouring detector elements. A single 'image' taken by the detector is therefore only in 1-D. Yet, by repeating this process from a sufficient number of different angles and along the entire patient (z-axis) a 3-D model can be computed (based on 'Radon transformation'). In contrast to 2-D methods, where the patients interior is projected/compressed onto a flat image, CT preserves the exact location information. This feature led to a radical improvement in diagnostics.

Since its clinical introduction in 1971 by Godfrey Hounsfield, CT has become a widely used 3-D imaging modality for a range of applications including radiation oncology. Especially in radiation therapy, knowledge of the exact geometry is crucial, which is why CT plays such a pivotal role in treatment planning (see 1.4). [1], [2]

3-D image reconstruction

As a photon passes through the patient, it encounters different materials associated with characteristic linear attenuation coefficients. It is practical to think of the scanned body as a collection of $N = N_X \cdot N_Y \cdot N_Z$ finite size cubes (Δx cube length) called 'voxels' (analogous to pixels in a 2-D digital photograph). The entire model can then

be regarded as a 3-D matrix, with the attenuation coefficients μ_i of the voxels as its entries. Figure 1.2 represents a $(4, 4, 1)$ matrix. It depicts the path an X-ray may follow passing through voxels with different values μ_i. This discretisation allows us to change equation 1.4 to:

$$I(x) = I_0 e^{-\sum_{i=1}^{N_X} \mu_i \Delta x} \tag{1.5}$$

The initial and final intensities can be read of the settings of the X-ray tube and the detected signal. Based on these values, image reconstruction algorithms derive the three-dimensional linear attenuation coefficient matrix. For convenience, the computed numbers are converted to Hounsfield Units which are displayed in the final image. [1], [2]

Figure 1.2: Simplified attenuation matrix (4,4,1). Image source: [2]

Hounsfield Units

In a final CT scan, voxel values are recorded in Hounsfield Units (HU), which relate to the attenuation of water at room temperature:

$$HU_{material} = \frac{\mu_{material} - \mu_{water}}{\mu_{water}} \cdot 1000 \tag{1.6}$$

Table 1.1 lists types of human tissue and their values on the HU scale. Generally, HU values range from -1024 to +3071 (12 bit), but the upper limit can be extended to 15,359 (14 bit) if materials with even higher attenuation need to be visualised (e.g.

Table 1.1: Average HU values for various types of human tissue

Substance	HU
Air	–1000
Lung	–750 (–950 to –600)
Fat	–90 (–100 to –80)
Water	0
Muscle	+25 (+10 to +40)
Brain, white matter	+25 (+20 to +30)
Kidneys	+30 (+20 to +40)
Brain, grey matter	+35 (+30 to +40)
Blood	+55 (+50 to +60)
Liver	+60 (+50 to +70)
Compact bone	+1000 (+300 to +2500)

implants).

Typically, CT scans are displayed on computer monitors, which imposes the need to map the HU values to a 8-bit greyscale (256 steps of luminosity). Since the number of possible values (dynamic range) on the HU scale is 16 times the shades of grey on a screen (12-8 = 4 bit difference; equivalent to a factor of 2^4) the screen cannot convey all details at the same time. A linear mapping would result in 16 neighbouring HU values being compressed to the same brightness on the sreen. This way, the brightest (bone) and darkest parts (soft tissue) of the image would be clearly distinguishable. At the same time small differences (<16 HU) would appear to have exactly the same intensity. However, most of the time, the doctor's focus might lie either on soft tissue or bone material. Bearing in mind that soft tissue values range only from 10 HU to 70 HU at most (see table 1.1), such a compression would make distinguishing tissues using CT very unreliable. Instead of showing detail from the lowest to the highest value, a range of values - a so called window - can be chosen. Let's assume, for example, a range from -100 to 155 HU to be of interest. This selected range can be mapped directly and uncompressed to a 8-bit greyscale. Any values above 155 HU will be assigned the brightest value (white = 255), below -100 the darkest (black = 0). While showing very good soft tissue contrast, all bones would be depicted with exactly the same brightness (255), even though they might have a varying HU values. For bone structures, a range from 300 to 2500 HU might show sufficient contrast. Standard computer programs used

to display CT images allow the user to change the window interactively to any value range. [1], [2]

Image acquisition

The time necessary to collect 1-D attenuation projections from sufficient angles is called 'acquisition time'. In 2-D X-ray scanning only one picture is taken, while a 3-D CT model is made up of a photo sequence. If the patient moves during the imaging process, the final model would show motion artefacts which might lead to wrong conclusions. Consequently, CT scanners are designed to minimise acquisition time while ensuring sufficient image quality. Very fast CT protocols result in smaller resolution, because less images are taken. It has to be said, though, that CT acquisition time is usually already significantly shorter than MRI. [1], [2]

Image quality

Additionally to the relatively short acquisition time, CT scans show little distortion compared MRI (see 1.3.3), which is why they are often used as 'gold standards' (reference scans used for MRI distortion assessment).

While bone structures are clearly visible in CT scans, 'soft tissue contrast' is relatively low compared to MRI. In other words, parts of the body which are considered as 'soft tissue' (intestines, brain, blood vessels, etc.) differ little in brightness and are therefore hard to distinguish. See 1.3.1 for more information.

Another aspect of image quality is the 'low contrast resolution' of the scan. It directly relates to how much structures and their surroundings have to differ in signal intensity to be clearly distinguishable by doctors. The quality of the 'low contrast resolution' is mainly limited by noise. Noise is a random pattern underlying the actual signal and is always present to some extent. Its prominence in the final image is described by the Signal to Noise Ratio (SNR). If the SNR is too low, fine structures blend with the noise and cannot be distinguished. Strategies to achieve a high SNR include raising the initial photon flux (intensity) or employing contrast agents. The intensity is governed by the tube current, which is limited by the heat capacity of the tube and health considerations regarding the patient dose.

Alternatively, the spatial resolution can be decreased, effectively combining neighbouring image slices. This way the SNR for the combined slices is increased, but fine structures along the z-axis might be lost due to the reduced resolution. [1], [2]

Health considerations

CT scans describe the attenuation throughout a patient, which is directly related to how much energy is transferred from photons to matter. Only because X-rays are absorbed by the human body, this imaging modality gives insight in the density distribution of a body's interior. However, this transferred energy is capable of causing biological damage. (see 1.2.2)

While the radiation dose administered during a single CT scan is relatively small (typically not more than 15 mSv) and almost negligible compared to the dose administered during a potential radiation therapy, patients receiving this dose regularly end up with a potentially harmful accumulated amount of radiation. Cancer patients, for instance, need to be imaged frequently during treatment planning. However, patients might die before those consequences come into effect. Therefore, it is typically children (who received a great number of CTs) to suffer from induced cancer occurring up to 40 years later. So while the benefit from using CT for diagnostics far outweighs the damage, there have been major efforts to reduce dose while maintaining reasonable image quality. [5]–[10]

1.3.3 Magnetic Resonance Imaging - MRI

Magnetic Resonance Imaging (MRI) is a 3-D imaging modality based on Nuclear Magnetic Resonance (NMR), a phenomenon discovered by physicist Isidor I. Rabi in 1938. Atomic particles such as protons have an inherent quantum mechanic feature called 'spin', which is associated with a magnetic moment μ. Without an external field, a proton's spin is oriented in a random direction in space and so is its magnetic moment. The sum of magnetic moments belonging to a number of protons results in a net magnetisation. Due to their random orientation, the net magnetisation will be zero for a sufficiently high number of particles. This is because, on average, for every proton's spin there is always another particle's spin oriented exactly the opposite way cancelling its magnetic moment.

In the case of an applied external magnetic field (this static field is often called B0), the spins will either align parallel (pointing in the same direction) or anti-parallel (opposite direction) to this field, where their energy reaches a local minimum. Parallel protons have an even lower energy than those pointing the other way. In a collection of many spins, the number of parallel spins will therefore slightly dominate, resulting in a net

magnetisation greater than zero (see figure 1.3). In other words, only the amount of protons that is not compensated by those looking in the opposite direction contributes to a detectable magnetic field.

Figure 1.3: The spins, initially oriented randomly in space, become aligned either parallel or antiparallel to an externally applied magnetic field B_0. Image source: [2]

By applying a short radio frequency pulse (often referred to as B1), the total external magnetic field changes and the magnetic moments start precessing around that new external field. The pulse duration is usually chosen as to flip the spins by a 90° angle. They are now oriented in the transverse plane to the original external field, and so is the resulting net magnetisation. Similar to how a spinning top rotating at an angle to the direction of gravity precesses, the magnetic moments will now precess about the direction of the external field with a frequency linearly proportional to the external field strength. This precession movement can be detected as induced current in a receiver coil, because the net magnetisation still follows the spins' orientation. Again, the particles would like to minimise their energy by aligning their spins to the external field, but in order to do so they need to give away the surplus energy, transferring it to the surrounding lattice. Those spin-lattice interactions happen with different efficiency depending on the tissue. The time it takes the spins to align is expressed in a material specific time constant T_1. Shortly after applying the radio frequency pulse, regions of the body where magnetic moments align quickly (short T_1) have a stronger net magnetisation (in the direction of the external field) than those where energy is being transferred slowly (long T_1).

At the same time, the spins interact with each other, affecting the local magnetic field and spins in their vicinity. The magnetic moments, which started out precessing in phase directly after the radio frequency pulse flipped them, will precess at slightly different

frequencies, due to the small fluctuations of the local magnetic field. The differences cause the collection of magnetic moments to 'de-phase' and the net magnetisation in the transverse plane to vanish. This process caused by spin-spin interactions is described by the material specific time constant T_2. Eventually, all spins will be again aligned either parallel or anti-parallel to the external field, just as they were before the RF-pulse.

Applying the RF pulse with a homogeneous field strength along the whole body would excite all spins simultaneously. In order to localise differences in tissue magnetisation, the RF pulse is instead combined with a linear magnetic gradient field 'selecting' a slice to be imaged at a time. The rest of the body is unaffected, the signal measured directly after such a pulse was applied originates only from the chosen slice. Eventually, by collecting data on the net magnetisation throughout the body in different locations ('scanning' the patient slice by slice), a 3-D image can be computed.

Depending on the information required from the examination, Doctors can choose to create images that reflect spin-lattice interactions (T_1 weighted) or spin-spin interactions (T_2 weighted).

Governed by the chosen settings, particular tissues will displayed varying contrasts. For example, areas with increased water level will be dark on T1 weighted (T_1w) MRI whereas the same areas will be bright on T_2w MRI. Additionally, certain types of materials (tissues) can be intentionally not imaged (suppressed) to reveal others that are in their close proximity. This is often done with fatty tissue that can cover relevant parts of the field of view. There are various methods how this can be achieved. MR imaging gives practically endless possibilities in terms of selective imaging and is mostly limited to imaging time and technical characteristics of the scanner. Medical physicists and scanner vendors are incessantly working on new MR imaging methods and applications.

The set of parameters governing how the tissue is excited and data acquired is called 'image sequence'. Delineating tumours or lesions is often accomplished by looking at both T_1w and T_2w weighted images and drawing the right conclusions.

Soft tissue contains a lot of water, which is made up by oxygen and hydrogen. Hydrogen nuclei are single protons and their nuclear magnetic resonance is what MRI is tuned to visualise. This is why soft tissue appears as bright areas in MRI, whereas bone material has only little contrast. [11]

Most the scanners are build to house a receiver coil in the gantry and they are able to measure the signal using only this one coil. However, in practise, to obtain a stronger signal, a smaller coil is typically placed closer to the source of the signal, the patient.

As the region of interest (ROI) is usually limited to a specific organ, receiver coils are available in different sizes and shapes, often designed to fit the patient with a comfortable but narrow space in between.

To get even closer, so called 'surface coils' can be placed on the patient. 'Spine coils' are sometimes hidden in the table on which the patient lies during the examination. Typically, for creating images of a patient's head, coils with a fixed arc-like geometry are used. This type of coil (see figure 1.4) was used for the data acquisition of this thesis.

Figure 1.4: MR scanner coil: (1) upper part, Body/Spine Array Coil; (2,3) central positioning markers; (4) lower part, Body/Spine Array Coil; (5) connection ports; (6) colour label. Image source: Siemens Healthineers

Studies have shown that delineating certain types of tumours, for example prostate cancer, is more accurate using MR images than using CT. [12]–[14]

In diagnostics, MR images prove to be very useful. Also for radiation therapy treatment planning, the superior soft tissue contrast is exploited during the definition of organs at risk and targets. Unfortunately, due to the physical principles of MRI, it indirectly provides the information about the proton density, whereas CT can provide information about electron density. It is the knowledge of the electron density which is necessary in the treatment planning process. During this, the dose distributions are calculated based on the applied beam geometry and the distribution of matter on its way.

Image Quality

Contrary to CT, MRI is prone to distortion due to field inhomogeneities. Organs might appear shifted, elongated or shrunk. The effect is most prominent along the outer edges of the scanner's field-of-view (FOV). In the isocentre (middle) of the scanner, the distortion is smaller, because here the field is least aberrant. For most applications, small position shifts and deformations are of minor importance. MRI scanners usually come equipped with an internal distortion correction algorithm. Figure 1.5 shows the unmodified and the corrected version of an image coming from such a scanner. Those methods are developed by the company designing the scanners. Knowing the technical details enables them to write tailor-fit scripts which drastically reduce the distortion.

However, for radiotherapy treatment planning this might not be enough and it is necessary to additionally monitor the distortion and, if necessary, take additional corrective measures.

While its soft tissue contrast is superior to CT, a relatively long acquisition time is necessary to achieve a sufficiently high SNR. This leads to the risk of motion artefacts (patients moving during the scanning procedure). To tackle this issue, resolution can be reduced, effectively combining signal from several voxels to create a single voxel, reducing the overall noise. The trade-off is that fine structures might get lost.

Figure 1.5: On the left is the original image, on the right the automatically corrected version of the same MR image (inverted colours). Image source: Courtesy Piotr Andrzejewski (unpublished data)

Health considerations

Strong, static magnetic fields (typically up to $3T$) are present around MRI scanners at all times, and precautions have to be taken to ensure safety for patients and medical personnel. Ferromagnetic materials (such as steel and iron) can become dangerous projectiles in vicinity of the MRI scanner. They must be excluded from the room housing the magnet without exception.

The RF pulses repeatedly put spins in excited states, transferring energy to the human body. MRI scanners are designed to limit the rise of a patient's body temperature to 0.5^oC during standard imaging. Only combined with either medical or appropriate psychological monitoring the limit can be raised to 1^oC. An ethics committee approval is necessary for even higher values. In general, patients should be exposed to RF fields only as strong as their thermoregulatory system is capable to cope with.

Finally, magnetic field gradients are applied together with the RF pulse. They are switched at high frequencies leading to induced currents in conducting body tissue. In principle, those currents stimulate nerves which might result in muscle twitching or pain. However, gradient levels are set to avoid stimulation. During have reported some subtle biological effects, but there was no evidence pointing towards harm caused by short term exposures. At the same time, patients suffering from epilepsy might show increased sensibility to induced electric fields in the cortex and should be imaged with caution. [2] Some patients might be allergic to contrast agents used for specific MRI examinations. Here, safety procedures are similar to those performed during administration of regular drugs.

Open bore MRI scanners

The radiation oncology department of the Vienna General Hospital (AKH) is equipped with an $0.35T$ open-bore, c-arm MRI scanner (see Section 2.1.3 for more details). This open design has proven to drastically improve the well-being of patients experiencing anxiety in closed-bore scanners. For this reason, the number of incomplete MR examinations due to a claustrophobic events is relatively low. [15], [16] Besides, patients who would not fit in closed designed scanners can be imaged. Furthermore, brachytherapy patients can be placed in the scanner with applicators attached. Brachytherapy is a form of radiation therapy in which radiation sources are typically inserted into the hu-

man body to perform the irradiation from close distance; imaging is performed for dose distribution planning as well as verification of applicators position after their insertion (for more information see section 1.4).

This scanner's magnetic field is weaker than the fields of closed-bore scanners that are widely used (typically 1-3 Tesla). High field strengths result in greater resolution, better SNR ratio and faster imaging time. Generally, diagnostics benefit from greater image quality. However, at some point, diagnostic accuracy stops increasing with field strength. At the same time significant improvements can be achieved at low fields. A "combination of field independent polarisation [...] with frequency optimized MRI detection coils [...] results in low-field MRI sensitivity approaching and even rivalling that of high-field MRI." [17]

Low field MRI scanners are also typically characterised by less distorted images. [18]

Apart from the often satisfactory image quality, there are considerable cost advantages to the use of lower field MRI. The initial purchase price and the ongoing maintenance expenses are considerably lower than those of high field scanners, which often use superconducting magnets cooled with liquid helium. [19] Permanent magnets might be weaker, but do not require constant cooling. Also, low fields allow facilities to build smaller rooms and magnetic objects are less dangerous.

Diffusion Weighed Imaging - an example for non morphologic Imaging

Despite the fact that this type of MR imaging was not used for this thesis, it will be mentioned here as an example of the vast range of measurements possible with MRI techniques.

Diffusion weighted (DW) imaging quantifies molecular diffusion in the body. This imaging technique uses MRI technology differently: Additionally to the gradients needed to select a slice (strength $3-5\,mT/m$, duration $2-4\,ms$), the sequence for DW imaging applies two long, strong, consecutive and opposite gradients (strength $30-50\,mT/m$, duration $20\,ms$) during which molecules may move due to diffusion. After those two gradients (which, usually, will be applied in all three Cartesian directions), the remaining signal (net magnetisation) is measured by the receiver coil.

Molecules which are restricted in their movement will experience two equally strong but opposite magnetic fields. The first will cause them to precess with a certain speed (linear with field strength), effectively changing their phase. The second will cause

them to precess exactly the other way around (same strength, but other direction), returning them to their initial state. Now they will again be in phase and result in a net magnetisation which is not zero, but visible as bright areas on the scan.

Those molecules which are free to move, however, will not experience a constant field strength, because the field has a gradient. As they move through the body they will precess at varying speeds during the first and then at different speeds during the second gradient. As a result, they will be out of phase when the net magnetisation is measured by the receiver coil, and will not cause bright areas in the scan.

DW imaging can be used to diagnose acute strokes (brain infarct), because areas with restricted diffusion (blocked blood flow) show a strong signal compared to healthy tissue with normal diffusion. Another interpretation of low diffusion (high measured signal) can be the increased cellular density (so dense that free diffusion of water molecules is suppressed) which is characteristic for cancerous tissue.

Since the time necessary to allow the molecules to move during the two gradients is relatively long, the image will naturally be T_2 weighted. This is taken into account by creating a second image which is also T_2 weighted, but does not apply two consecutive opposite gradients. oThe difference between the DW and the not DW weighted sequences reflects the actual contribution of diffusion (apparent diffusion coefficient - ADC). [20]

1.4 Radiation therapy

Radiation therapy utilizes ionising radiation to damage or kill cancer cells in order to stop them from multiplying. This prevents the growth of tumours, makes them shrink in size and hopefully cures the patient.

During radiotherapy treatment planning (RTP), 3-D models of the patient are used to define targets (regions where the dose should be delivered to) and organs at risk (where the dose should be delivered to). This ensures that vulnerable organs are spared from radiation while making sure the tumours receive sufficient dose. Moreover, methods to quantify the amount of radiation that different body parts absorbed are needed, because the actual treatment might differ from the plan.

While travelling through matter most types of radiation release energy mainly due

to coulomb interactions with the outer shell electrons of atoms. Knowing the electron density of the targeted tissue area is therefore essential. In order to reach a specific penetration depth, the particles' initial energy has to be chosen accordingly. The necessity to treat the tumour with a required amount of radiation leads to a radiation therapy treatment plan.

There are two well established methods for applying the radiation. External beam radiotherapy (EBRT) is performed from afar: Gantries are able to position the radiation source around the patient in a way covering virtually any possible angle. During a treatment session, fractions of the total dose are administered from many different angles (or continuously with alternating gantry angle). The sum of those individual treatments results in the required dose distribution. Figure 1.7 shows an example of two differently calculated treatment plans.

In conventional EBRT, photons (X-rays) in the range of 4MeV to 20MeV are used to deposit the necessary dose at the location of the tumour. Unfortunately, radiation interacts with all cells it passes until it is fully absorbed. It releases its energy along its entire path while travelling through the patient. This behaviour may result in dose being delivered to cells all the way from the point of entry to the point where the (weakened) ray leaves the patient. Other types of ionising radiation are also used, but less common. Electrons and low energy X-rays are favoured for superficial tumours; rare methods using neutrons and even muons also exist. Charged particle therapy (using e.g. protons or carbon ions) is on the rise, but far from reaching the availability of X-rays. This type of radiation minimises the damage done to healthy tissue due to its distinctive behaviour in energy loss called "Bragg Peak". They release most of their energy only shortly before being stopped completely. [21] This effect can be used to spare tissue lying behind the tumour from radiation entirely and also reduce the amount of energy transferred to organs located before. [22] A comparison between the behaviour of X-rays and protons is shown in figure 1.6.

Brachytherapy, on the other hand, is when the radiation source is placed close to or inside of the patient. The source is either moved close to the target area using applicators (temporary treatment) or implanted permanently. The latter method is done by inserting so called "seeds" (sealed metal containers with radioactive material) directly into the target area where they release high amounts of radiation. Over time they become less active and eventually the treatment stops automatically. These rice grain sized

implants can remain in the body without causing any harm. Treatment of prostate and cervix cancer is often done with this technique. In comparison to EBRT, Brachytherapy allows higher doses while at the same time minimising the radiation reaching organs at risk; precise dose distributions can be achieved. At the same time, not all cancer types can be treated this way. For some, a non-invasive method (e.g. EBRT) is a better alternative.

Figure 1.6: Energy release of ionising radiation [2]

[2]image source: by Cepheiden - https://commons.wikimedia.org/wiki/File:Dose_Depth_Curves.svg, [GFDL (http://www.gnu.org/copyleft/fdl.html)], via Wikimedia Commons

Figure 1.7: Example of Radiotherapy treatment plan: coloured areas represent dose values deposited during treatment. The plans were calculated using different treatment techniques: with protons (IMPR) and photons (VMAT). Image source: [23]

1.4.1 Role of CT

Until recently, RTP relied almost entirely on CT. There are two main reasons for this:

Firstly, calculating the electron density using data obtained with CT is an straightforward task. Secondly, CT images generate 3D images with little distortion. Exact geometries are needed for correct RTP. It is the most reliable approach to create precise radiotherapy treatment plans. [24], [25]

1.4.2 Role of MRI

MR images also record luminosity values, but they do not correspond to radiodensity. Due to the better visibility of tumours on MR images, RTP often uses combined data from both imaging modalities. Additional information derived from DW MRI, for instance, can also support response prediction and assessment. However, there are some difficulties arising from combining CT and MRI for EBRT: in order to profit from separately acquired data, the resulting images must be aligned (registered) either manually or automatically. This is a hard task since non-rigid objects (organs) change their shape and location between measurements which may lead to inaccuracies. Algorithms supporting non-rigid registrations are already under development, but there is still room for improvement. For now, only local rigid registration is capable of reliable target and organ at risk definition.

Alternatively, MRI-only radiation therapy protocols are being developed: one way of doing this is to use MRI data to create a Pseudo-CT, which contains information about electron density. Comparisons to using CTs and MRI-based pseudo CTs have shown acceptable deviations for X-ray therapy. In charged particle therapy the resulting dose gain in healthy tissue and dose loss in cancer regions owed to inaccurately assigned electron density values is bigger. However, further improvement of accuracy promises to reduce time and money needed for RTP when CT is no longer needed. Furthermore, patients would be spared the additional dose of CT examinations. [26]–[30]

1.5 Aim of this work

The idea of only using MRI for treatment planning is approaching the clinics, but there are still some issues that need to be addressed.

Due to the possible image distortion, great care needs to be taken and the MR images

must be verified before they are used for RT target definition and dose calculation. The available MRI scanner at the AKH is equipped with an on board correction algorithm which is supposed to reduce distortion. See figure 1.8 for an example of how this correction affects an image. Distorted images might lead to wrong calculations of how much energy is needed for the radiation to accumulate exactly at the target region. If, for example, bone structure is depicted as thicker than it really is, RTP would suggest a treatment which would deposit more energy behind the tumour than intended. The opposite holds for cases where tissue appears to be thinner, which would result in areas lying before the tumour being irradiated.

The goal of this work is to: find an optimal (providing satisfactory image quality and convenience of use) liquid filling for the rod cavities in an already existing custom designed distortion phantom provided by the Medical University of Vienna to be able to acquire the reference CT image and test MR acquisitions.

Develop and implement a method to assess and illustrate the distortion of the MR image based on tracking of the distortion of one of the phantom rods, with an option to further extend the tool for mutli-rod tracking.

Therefore, this work focuses mainly on the assessment of the acquired imaging data (using the implemented software tools), choosing which liquids to fill the phantom with, but not its entire design. However, possible fillings have to be produced and tested. Similar approaches are being used for distortion correction by other facilities. [31]–[36]

(a) MR scan with no distortion correction

(b) MR scan after internal distortion correction

Figure 1.8: Rendered MR scan showing the difference before (a) and after (b) applying the build-in distortion correction Image source: Courtesy Piotr Andrzejewski (unpublished data)

2 Material and methods

2.1 Scanners and Imaging protocols

For this thesis a CT scanner and an open-bore, c-arm MRI scanner in the radiation oncology department of the Vienna General Hospital (AKH) were used.

2.1.1 MR

Following the general practice of phantom measurements, and the AAPM recommendations on phantom filling materials [AAPM 1990 and 2010]/ [37] oils or water with addition of paramagnetic substances (to reduce the spin-lattice relaxations times) were chosen (generally short T1 time of 200-400 ms). This, combined with a T1-weighted test sequence, improves the efficiency of the measurements assuring relatively high SNR in short time. To further improve image quality, the averaging parameter of the test sequence was set to 4. This setting practically means the measurement is repeated and averaged n times by the scanners software. Remaining sequence parameters were optimized to yield a acceptable resolution for the chosen ROI setting (which is relatively large as it covers the whole phantom). The test sequence was chosen in a way, such as to deviate as little as possible from a typically used clinical imaging setting. However, to fully understand and investigate the possible image distortions, each clinically used sequence should be commissioned with a distortion phantom. Detailed sequence parameters are listed in table 2.1

system	MR
manufacturer	Siemens
product name	Magnetom C!
coil	Body/Spine Array Coil XL
[internal W x H]	[50 x 30.5 cm (19.7 x 12 in)]
orientation	axial
sequence	T1 weighted 3D-FLASH Vibe
	(Volumetric interpolated breath-hold)
slice thickness	2.3 mm
pixel spacing	2.3 mm
matrix	128 x 120
repetition time	7,07ms
echo time	2,7
bandwidth	240
flip angle	6 deg
averages n	4
acquisition time	17.5 min

Table 2.1: MR scanner and used protocol

2.1.2 CT

To generate a reference CT image set a typically used clinical pelvis imaging protocol for radiation therapy planning (with increased resolution 0.6 mm) was used (see table 2.2).

system	CT
manufacturer	Siemens
product name	SOMATOM Definition AS
software version	syngo CT 2013B
protocol	Prostatae_VMATAdult
slice thickness	0.6 mm
row spacing	0.801mm; 512
column spacing	0.801mm; 512
kVP	120kV
x-ray tube current	19 mA
convolution kernel	I30f, 3

Table 2.2: CT scanner and used protocol

2.1.3 MRI scanner - field distribution

Diagrams of the magnetic field strength are depicted in figure 2.2); its gradient in figures 2.3, 2.5 and 2.4). Figure 2.1 shows an image and different views of the scanner and the planes along which the field strength and gradient are displayed. For imaging, the patient is positioned as to place the ROI at the isocentre, the area where the B_0 field is most homogeneous (where there is a low gradient).

(a) Image of scanner
x,y, & z -axis

(b) side view along z-axis,
black line represents plane for front view (see fig. 2.4)

(c) front view along x-axis,
black line represents plane for top view (see fig. 2.5)

(d) front view along x-axis,
black line represents plane for side view (see fig. 2.3 and 2.2)

Figure 2.1: Magnetom C! Image source: [38] (with kind support of Siemens Healthineers)

Figure 2.2: Magnetom C! field strength
side view along z-axis, distant view (see fig. 2.1d). Image source: [38] (with kind support of Siemens Healthineers)

Figure 2.3:]
Magnetom C! field gradient
side view along z-axis (see fig. 2.1d). Image source: [38] (with kind support of Siemens Healthineers)

Figure 2.4: Magnetom C! field gradient
front view along x-axis (see fig. 2.1b). Image source: [38] (with kind support of Siemens Healthineers)

Figure 2.5: Magnetom C! field gradient
top view along y-axis, distant view (see fig. 2.1c). Image source: [38] (with kind support of Siemens Healthineers)

2.2 Custom build phantom

To compare images from different scanners and asses occurring distortion, a rigid object with known dimensions is necessary. Such a 'phantom' is often made from plastics filled with liquids which are easy to handle and typically well seen on MR and CT images. The AKH's design is made up from an array of replaceable, fillable plastic rods.

(a) MR (b) CT (c) CT

Figure 2.6: Comparison (inverted colours): MRI only shows liquid filling, CT also the plastic rod and pane (horizontal black bar crossing middle and right rod); (**a:**) *MRI* - filled rod, plastic not visible (field of view too small to show entire rod); (**b:**) *CT* - empty rod, plastic visible; (**c:**) *CT* - filled rod, plastic and filling visible.

2.2.1 Frame and rods

The phantom was build to fit the largest available rigid coil for the MRI scanner. Three parallel acrylic glass panes in the shape of the coil serve as a frame for the plastic rods. In the middle an empty area was reserved for an optional additional smaller phantom (not used for this work). Figure 2.7 shows a rendered CT picture of the phantom. See also figure 2.8 showing a CT image of one pane (with no rods inserted).
More than 300 plastic rods (length: $50\,cm$, outer diameter: $8\,mm$, inner diameter: $4\,mm$, volume: approx. $6\,ml$) could be placed in the phantom. See figure 2.9 for a schematic sketch of one rod. The bottom part of each rod was sealed with a glued plastic plug,

the top could be closed with a plastic screw. Frame and rods were already build and assembled before the author started working on this project.

Additionally, to the rods which would be used to assess the distortion, a number of vitamin pills[1] were attached to the frame as reference markers (see figure 2.10). These pills are visible in both CT and MRI images and were used to align them (see section 2.4). This way there is another way of checking the alignment during the prototyping process in addition to the rods.

Figure 2.7: Rendered CT image of phantom. Image source: Courtesy Piotr Andrzejewski (unpublished data)

[1]soft gel capsules containing vitamin E and A [39]

Figure 2.8: An axial view of one of three plastic panes that make up the frame of the phantom. This scan shows how the pane looks like with no rods inserted. A little 'x' in one of the holes slightly to the right of the lower part of the pane marks where rod #5 was inserted later for imaging (see section 3.3.1); a little 'z' in the centre of the upper area indicates where rod #16 was inserted for another imaging sequence (see section 3.3.2).

Figure 2.9: A schematic of an empty plastic rod as they are used for the phantom (inverted colours). On the left the rod ends with a glued plastic stopper, on the right hand side a plastic screw seals it. The figure does not show true proportions.

(a) CT scan

(b) MRI scan

Figure 2.10: CT (a) MR (b) coronal images (inverted colours) showing rods and 2 attached vitamin-e pills visible on the right hand side of the rods. In the CT scan the plastic pane and adhesive tape used to hold the pills in place are also visible.

2.2.2 Rod fillings

For this study 17 different liquids were produced to be tested as possible fillings. They are listed in Table 2.3.

No.	$NaCl$	$CuSO_4 \cdot 5H_2O$	Soap	Ascorbic Acid	Agar	Primovist [volume-%]
water based:						
#1						
#2	3.6	1.96				
#3	3.6	3.92				
#4	3.6	19.6				
#5	3.6	1.96	1			
#6	3.6	1.96	5			
#7	3.6	1.96	20			
#8	3.6	1.96		0.36		
#9	3.6	1.96		3.6		
#10	3.6	1.96		36		
#11	3.6					0.1%
#12	3.6					1%
#13	3.6					10%
#14	3.6	1.96			0.5	
#15	3.6	1.96			20	
non water based:						
#16		Motor Oil:	*Castrol Power1*			
#17		Silicon Oil:	*Charge: 15HLVY023*			

Table 2.3: The composition of all tested solutions.
(components in g/L; exception: Primovist in volume-%)

Being closed at one end and having a capillary shape (small diameter) makes it impossible to fill the rods by simply pouring the liquid through the opening. Instead of adding the fluid at the top, it has to be injected starting at the bottom. This way the contained air is pushed out by the injected liquid through the opening at the top. A long, thin plastic tube was inserted and used for injection, leaving enough room for the gas to escape. Between injections of different liquids, the tube was flushed with #**1** (distilled water) or #**2** (main component of most solutions).

In order to minimise the amount of gas dissolved, the liquids were brought to boil shortly before injecting. Gas solubility generally decreases with rising temperature [40], [41]. After injecting the solution in the rods, they were left to cool down. Before closing, the rods were topped up completely (no trapped air bubbles). The oil based liquids,

#**16** and #**17**, were not brought to boil.

2.2.3 Handling of bubbles

On the second day of working with the filled rods the one containing liquid #**6** broke (leakage). It happened when delicately knocking it against on the table while standing upright. This was intended to mobilise bubbles that sticked to the wall and make them travel vertically to on end of the rod. (see table 3.2) The plastic stopper on the lower end came loose. The rod containing filling #**6** was not replaced. Consequently, all CT/MRI images used to asses signal intensities of the tested solutions show only 16 rods.

2.3 Used software applications

Prior to analysing the data, the scans had to be prepared. For this task, two applications were used:

- *MIRADA RTx*: a commercial software solution for both diagnostic imaging and radiation oncology [42] was used to align CT and MR images.
- *3D Slicer*: (Versions: Slicer-4.5.0-1-linux-amd64, Slicer-4.6.2-win-amd64) a "free and open source software package for visualisation and medical image computing". [43], [44] was used to crop and resample images, quickly read values and visualise the results.

2.4 Pre-processing MRI and CT scans

Figures 2.11, 2.12 and 2.13 show the most important stages in the image processing work flow performed to prepare the data for the developed script.

Step 1 After loading the CT and MRI scan into *MIRADA*, they were aligned using the vitamin-e pills, yielding maximum overlap in the centre of the image.

Step 2 Next, as the MRI image had a lower resolution than the CT scan, the MRI scan was resampled. Its voxel's size were changed to match the CT voxels and both scans exported.

Step 3 Both layers (MRI and CT) were loaded into *3D Slicer*.

Step 4 Its module 'annotations' was used to set a new region of interest (ROI) to include only a single rod.

Step 5 With the module called 'crop volume' (setting: voxel based cropping) the scans were reduced to show only the selected ROI.

Step 6 Using the module 'resample scalar volume' a number of interpolated (setting: 'bspline') higher resolution pairs (CT/MRI) were created.

Step 7 All new pairs and the cropped original CT/MRI pair were exported with 'create a dicom series' and saved in a separate folder each.

After this procedure a number of pairs based on the original CT and MRI were available, all of which had the same number of slices along the z-axis parallel to the phantom's rods. They only differ in the number of pixels making up each slice, their resolution varying from the original up to a hundred times finer. Each pair (CT/MRI) has the same pixel spacing (and resolution) in x and y direction. See table 2.4 for more details. Figure 2.13 depicts 3 CT/MRI scans of a single rod (axial) with different resolutions. "x1" stands for the original CT scan resolution (MRI resampled to match). "x4" is a resolution caused by 1 pixel being split in 4 smaller pixels, "x9" in 9, and so on and so forth. For better visibility, images shown as figures in this work are printed with inverted colours. Dark pixels have a high density/intensity value, white pixels are equivalent to air (low density/intensity).

resample factor	z (not affected)	y (same as x)	x
x1	0.60	0.98	0.98
x4	0.60	0.49	0.49
x9	0.60	0.33	0.33
x25	0.60	0.2	0.2
x100	0.60	0.1	0.1

Table 2.4: pixel Spacing (rounded values) [mm]

(a) CT (b) MR

Figure 2.11: Rendered CT/MRI image of phantom after steps 1-3 of pre-processing.

(a) CT (b) MR

Figure 2.12: Rendered CT/MRI image of phantom after steps 4 and 5 of pre-processing.

(a) CT x1 (b) CT x9 (c) CT x100

(d) MRI x1 (e) MRI x9 (f) MRI x100

Figure 2.13: CT/MRI axial images of the same rod (filling #5, inverted colours) after steps 6 and 7. The images on the very left show the original (CT) resolution, the resolution of those in the middle and the right were increased by resampling (9 times and 100 times finer);
pixel lengths: x1 $\rightarrow 0.98\,mm$, x9 $\rightarrow 0.33\,mm$, x100 $\rightarrow 0.1\,mm$

2.5 Developed software tool

In order to asses the distortion of the MRI scanner, a software tool was programmed. It is written in Python 2.7 and uses the *SimpleITK* package to read and process *DICOM* ("*Digital Imaging and Communications in Medicine*") files. [45], [46] *SimpleITK* is a object-oriented "C++ library with wrappers for Python, Java, CSharp, R, Tcl and Ruby". [47], [48] Its versatility is one of the reasons why this approach was favoured. It is a simplified layer built on top of the National Library of Medicine Insight Segmentation and Registration Toolkit (ITK). SimpleITK is also used by Applications like *3D Slicer*. Documentation and code examples of SimpleITK can be found at [49], [50] An alternative way to handle DICOM data in Python would be Pydicom. [51], [52]

This is an extensive list of python packages used to process data after using 3D Slicer:

- SimpleITK
- numpy
- scipy
- matplotlib.pyplot [53]
- skimage.draw
- datetime
- os

Capabilities - Overview

The developed software tool is not able to automatically detect individual rods in a CT or MRI scan depicting the whole phantom. Instead the acquired 3D images have to be cropped so they include only a single rod (see section 2.4).

The python script can be used to:

- separate bright areas which are not connected ('masking' might be used as future method to automatically detect individual rods)
- find and mark slices which show irregularities
- calculate the centroid coordinates along the rod
- measure the local distortion described by the
  - location shift (refered to as "warp")
  - dice coefficient (the "DC" refers to an object's roundness)
- visualise individual rod slices
- plot the average/peak brightness, warp, or `DC` along the rod

- write warp and DC for each slice in a combined ".txt" file
- export a rod shaped scan where the pixel values reflect the distortion occurring in each slice instead of their brightness as a ".mha" file (useful for visualisation, see figure 3.12).

Measuring distortion

Since the rods have a cylindrical shape, distortion can only be assessed in radial direction. The z-axis is parallel to the rods, x and y are radial. Ideally, each slice (z = const.) should depict the bright circular profile of the liquid (and of the plastic rod in CT) surrounded by black pixels (air).
Two phenomena were chosen to reflect the amount of distortion occurring in each slice of the MRI scans. The distance which the rod appears to be shifted in the MRI slice compared to the CT slice is referred to as "warp". The rod's deformation (deviation from circular profile) is described using the dice-coefficient "DC" (also known as Sorensen-Index).

2.5.1 Calculation: dice coefficient (DC)

The DC was chosen as indicator for the deviation from a circular profile. The implementation as python function is based on the open source python package "Medpy". [54] A part of its module called "metric" was adapted. [55]

The calculation of the DC is performed for each slice individually. Additionally, to asses the overall distortion occurring along the rod, the average of all those values is also saved. As this aspect of distortion does not need a reference scan, the DC is measured for CT and MRI images separately. The dice coefficient or Sorensen index [56] is defined as:

$$DC = \frac{2|A \cap B|}{|A| + |B|} \tag{2.1}$$

Figure 2.14 describes the process of calculating the DC. It compares a binary image (input A) to a circle (reference B). In a binary image there are only 2 possible pixel values: "0" and "1". However, in a CT image, values most often lie in a range between -1024 and 1000 or higher. In order to reduce the original image to a binary image A, the script needs to split the pixels in 2 groups. A copy of the original picture is created

where all pixels with a value above a certain `threshold` are set to the value of "1". These are regarded as part of the rod. Those which are darker are set to "0". (See section 2.5.2 for details on how the `threshold` is calculated). Ideally, this procedure separates the surrounding dark area from the bright rod by using a suitable `threshold`. This way the new binary image A still shows were the rod is, but it lost all information on the actual brightness.

Reference B is a circle with its midpoint typically placed at the centre of mass (COM) of the rod. The `COM` is usually calculated using those pixels which exceed the `threshold`, but weighted with their actual brightness (not using the binary image)

This way the script is supposed to place the reference circle B in the centre of the input image A. The position of the circle's centre and its radius highly influence the outcome. If the COM coordinates used to position the reference circle B lie outside of the image (e.g. '-1,-1'), the DC is set to '-1', indicating that no meaningful result could be obtained.

The `DC` ranges from 0 to 1. A value of 1 indicates a perfect circular shape. A low `DC`, on the other hand, means the shape differs greatly from a circle and could be caused by many things such as: little overlap (e.g. a ring or crescent shape); a very dark image hindering delineation of the rod from background; a small circle with a radius close to a only a few pixels.

The obvious choice for the radius of the reference circle B is to use the size of the physical rod/liquid as it is should be visible on the scan. For CT images this would be $4\,mm$ (outer diameter of the plastic rod), for MRI images it would be $2\,mm$ (inner diameter of the plasic rod, equivalent to the diameter of enclosed liquid). The script calculates the `DC` using various radii close to those values and returns the result yielding in the highest average `DC` for the whole rod.

Using the CT COM

Alternatively, the DC fo the MR scan can be calculated with the reference circle B placed in the COM of the corresponding CT image. This value could be regarded as a combined distortion guide number as it is influenced by the COM shift and the deformation simultaneously. One should bear in mind, though, that the meaning of it is neither equivalent to a real `DC` nor to the warp and should be interpreted with caution.

If the CT COM is far from the MR image, A and B have little overlap resulting in a

small DC. As the implemented DC calculation tries a variety of radii, the circle B could, theoretically, always be chosen big enough to have some overlap with the binary image A. However, as B grows, A will stay the same. Therefore, only a fraction of B will contribute to the overlap and the rest will counter-act the benefits (see equation 2.1). Consequently, the DC would become so small that the script will choose a smaller radius for B.

2.5.2 Calculation: warp & centre of mass (COM)

To calculate the location shift between rods shown in CT and MRI, the coordinates of the centre of mass (COM) were subtracted. The x- and y-shift (warpXY) measured in each slice was saved in an array. Furthermore, the absolute value of the coordinate shift (warpMagnitude) was calculated.

The calculation of the COM is done with help of the "scipy" python package. Its module "ndimage" contains the function "$center_of_mass()$", which returns the COM's coordinates of a given input array. Only pixels representing the rod or the liquid should be used for the calculation. Otherwise the almost black voxels surrounding the rod would influence the result. To be regarded as part of the rod, the pixels' value has to reach a certain threshold. In order to find the relevant pixels two methods were developed:

1. a simple method calculating the number of pixels based on rod size

2. an iteration method finding a COM resulting in a good DC

Both methods rely on a single reference slice to calculate the threshold. This reference should be representative for the whole scan, because the threshold deduced from it will be used to find pixels belonging to the rod in all other slice, too.

1. Simple Method

The inner ($2mm$) and outer ($4mm$) radius of the plastic rods are known. So is the pixel spacing, the size of a voxel in real space (mm). Calculating the number of pixels which make up the more or less circular profile of the rod in a slice is calculated as follows:

$$pixelNumber = (radius^2 \cdot \pi) \,/\, (spacing^2) \tag{2.2}$$

For CT images the $radius = 4mm$, in MRI scans the $radius = 2mm$. $spacing$ is the

Figure 2.14: DC calculation for a MR scan (rod #5 @ slice 200) yielding a value of 0.9454

pixel spacing in x and y direction (developed script is only capable to process images with isotropic pixels pacing, as it usually is the case). Next, the pixels are sorted by brightness. The top $pixelNumber$ pixels are then used to calculate the COM. The value of the darkest pixel that is still counted as part of the rod is saved as threshold for

future calculations (e.g. finding the DC associated with the COM, see section 2.5.1). The method is summarised in figure 2.15. Now, the DC can be obtained as described earlier using the now known threshold and COM coordinates.

Figure 2.15: Simple method to find threshold for COM calculation based on real rod size (inner diameter: $2\,mm$, outer diameter: $4\,mm$)

2. Iteration Method

This algorithm is an iteration method. Figure 2.16 shows in what order the scripts executes individual steps during the iteration.

To begin with, it looks at the whole range of possible pixelNumbers, from 0% to 100% (**1**). As a reasonable first guess it assumes that 50% of all pixels belong to the rod (**2**). Now, in the first iteration (**3**), to find out whether more or less pixels would result in a better DC, it considers two new guesses: One halfway from the lower limit (0%) to its

Figure 2.16: Iteration method to find `threshold`, `DC` and `COM`

current guess (50%) which is:

$$\frac{lower + current}{2} = \frac{0 + 50}{2} = 25\% \tag{2.3}$$

and one halfway from the upper limit (100%) to its current guess (50%) which is:

$$\frac{upper + current}{2} = \frac{100 + 50}{2} = 75\% \tag{2.4}$$

Those numbers correspond to a `threshold` each, separating the chosen percent of brighter pixels from all darker pixels in the slice. Using the `thresholds`, the script calculates the `COM` and `DC` for both possible guesses. Comparing the average over the whole rod for both `DC`s will decide which guess is closer to representing the rod better (**4**). If the lower number of pixels yields a better average `DC` (**5a**), the upper half of the range will be neglected in the next iteration or in other words, the new upper limit takes the value of the former guess (50%). If the higher percentage yields a better average `DC` (**5b**), the lower half of the range will be neglected in the next iteration, the new lower limit takes the value of the former guess (50%). At the end of the iteration (**6**), the percentage resulting in the higher average `DC` is saved as the new current guess. After this first iteration (**7**) the script can either repeat steps 3-6 or end it by returning the best guess and its (average) `DC` (**8**).

To get a better understanding, let's suppose the iteration is repeated. At the start of the second iteration, the range is smaller (half of the entire range) and the current guess is set exactly in its middle. If, for example, the average `DC` for 25% was higher than for 75%, the next guess will be 25%, because the new range goes from 0% to 50%. In that case, `DC`s for the lower half of that range (12, 5%) and the upper half (18, 5%) will be calculated and compared to decide which half to eliminate in the third iteration. If, on the other hand, the average `DC` for 75% was higher, the next guess will then be 75%, because the new range goes from 50% to 75%. In that case, `DC`s for 62, 5% and 82, 5% will be compared.

The iteration continues until further steps yield no better average `DC` or a set number of steps (default setting limit is set to five iterations) has been performed. After the iteration process, the algorithm will return the `COM` which resulted in the best DC. The percentage of pixels that led to this `DC` is equivalent to a `threshold` which is saved for future calculations. Figure 2.17 shows the `DC` found in the course of trying different

percentages during the iteration method.

2.5.3 Detection of irregularities

The method to find irregularities is described in figure 2.18. After loading the image data, the script calculates the mean brightness of a reference slice (which was chosen by the user) (**1.**). This reference slice will be used to decide if and which other slices might show irregularities, for example air bubbles, markers, or plastic panes. It is the user's responsibility to make sure the reference is free from any such objects. Ideally, it is located near the isocentre and has a brightness that is representative for the whole image.

The script will compare each slice of the volume to the reference slice individually (**2.**). To decide whether a particular slice is "irregular", its average brightness will be compared to the reference slice's (**4.**). If the difference exceeds a certain value (**5b.**), the current slice will be marked as irregular and consequently will not be used to calculate its `DC` or centre of mass (COM). A value of $40\,HU$ was found to be yield good results for CT scans.

Irregular slices: centre of mass & dice coefficient

As irregular slices cannot be used to asses distortion caused by the scanner, numbers describing their distortion are not of interest. Instead of calculating `DC`, `COM` coordinates and warp regularly (based on the actual pixels in the slice), the values of these quantities are all set to "-1". This is done, because the relevant slice was marked as not suited for distortion assessment.

The `COM` cannot lie outside the image, yet coordinates "$(-1, -1)$" would indicate this (pixel coordinates are defined to be always positive in x and y direction). Similarly, the value referred to as `warpMagnitude` (absolute distance between MRI and CT COM) and the `DC` are defined to be positive numbers. All three are therefore easily understood to be invalid, indicating that the particular slice was marked as "irregular". The values representing x- and y-shift, on the other hand, are allowed to be negative or positive. To be consistent, they are still set to "-1". It is essential to bear this in mind when interpreting the script's output.

(a) full iteration process

(b) close up of same iteration as in (a)

Figure 2.17: COM iteration method, 5 repetitions; red triangles correspond to upper guesses, blue dots to lower guesses.

Figure 2.18: Check for irregularities

3 Results

3.1 Obtained MRI and CT scans

Figure 3.1 shows a coronal view of the 16 rods filled with the tested liquids and a reference rod on either side. In Figure 3.1b a trapped air bubble is clearly visible at the lower half of rod #5. Figure 3.2 shows an axial view of the tested rods and some surrounding rods. In figure 3.2b a water filled plastic bottle placed in the middle of the phantom is also visible. This was necessary, because the MRI scanner needs sufficient signal for shimming prior to the start of imaging. Without the bottle, the limited number of rods used for this scan would not have created enough signal. During a future distortion assessment where all available rods (over 300) are used, they will result in the required signal strength on their own.

(a) CT: The periodic black lines in upper and lower part of image (indicated with arrows) depict the plastic panes and their holes from above; the faint line across upper end of rods shows adhesive tape used to hold the rods in place; in rod #16 an air bubble is clearly visible close to the upper end.

(b) MR: The rods appear to be thinner than in the CT scan, because only the liquid filling is visible. The tested liquids result in different signal intensity (brightness); all plastic parts (rods and panes) are not visible; there is a trapped air bubble visible in rod #5.

Figure 3.1: Coronal CT/MRI (inverted colours; same scale; cropped images) images of 16 rods (tested liquids, numbering starting from the right, #6 excluded) + 2 reference rods (filled with water) on the sides.

(a) CT: The black bars visible just above the 16 tested rods, at the very top, and to the sides show plastic parts of the phantom holding it together. The faint grey line running below the tested rods represents the table on which the phantom was positioned during imaging.

(b) MRI: The black circle in the centre depicts a water bottle which was placed in the middle of the phantom (necessary for MRI scanner to start imaging).

Figure 3.2: Axial CT/MRI (inverted colours, same scale) images of the 16 rods filled with tested liquids (numbering starting from the right, #6 excluded); surrounded by reference rods (filled with water) and one empty rod (marked with arrow) which is not visible on the MRI scan.

3.2 Tested solutions

3.2.1 Visibility on CT/MRI scans

It should be noticed that in the CT image there is little difference in brightness between the tested liquids. The plastic rods themselves result in brighter pixels than any of the tested solutions.

On MRI scans most liquids had a mean and a max brightness value above 1000 (see table 3.1).
Only #**1**, #**9**, #**10**, & #**16** resulted in significantly less signal (below < 1000).

No.	Min	Max	Mean	Median	σ
1	182	371	288	269	69.8
2	1044	1921	1443.8	1405	312.9
3	941	2075	1451.2	1394.5	413.2
4	1176	1709	1440	1437.5	232.5
5	1125	2111	1583.8	1549.5	355
7	971	2241	1466.8	1316	471.2
8	1459	1947	1704	1705	180.5
9	385	584	486.8	489	93
10	247	502	343.6	266	111
11	830	1268	1036.2	1023.5	163.2
12	1158	2211	1648.8	1613	394.2
13	836	1657	1146.8	1047	321.2
14	800	2062	1383	1335	473.1
15	1156	1829	1476.2	1460	272.7
16	1102	1967	1509	1483.5	325.8
17	356	938	629.6	602	223.6

Table 3.1: Liquid visibility on MRI scan.

3.2.2 Mechanical properties of solutions

Rods were filled with the liquids and observed for several months. Number #**14** could be injected without problems, the solution remained fluid even after reaching room temperature. Number #**15** on the other hand changed to a gel like consistence and clogged the injection tube at the quickly after the rod was filled. The tube could not be used again.

Each rod was free of bubbles directly after sealing. All rods containing water based solutions contained some air after 2 months and the amount of liquid continued to decrease further (see tables 3.2 and 3.3). After 6 months the volume of air further increased (proportionally to the behaviour observed until then). Figure 3.5 shows the rods after a time of over 6 months. In some of the tested rods, the occurring air bubbles would stick to the wall. Only after gently hitting the rod they would start moving. Knowing the inner diameter d of the rods and measuring the length l of trapped air bubbles, their volume can be estimated:

$$V = \frac{d^2}{4} \cdot \pi \cdot l \tag{3.1}$$

While adding generic washing soap (#**5**, #**6** and #**7**) did not hinder air bubbles from forming, it significantly improved their mobility. Not only did they move quickly when the rod was tilted, large quantities of air also did not block the entire diameter of the rod. Instead they formed large but cohesive bubbles that could be moved to one end of the rod easily and at no point sticked to the plastic wall.

The ascorbic acid present in #**8** (concentration of 0.36 g/L corresponds to approx. 0.00204 mol/L), #**9** (3.6g/L), and #**10** (36g/L) seemed to have held back the formation of air bubbles for up to one week. After two months of observation, however, the rods also contained some air and all three liquids turned brown, the colour being more saturated for higher concentrations of ascorbic acid. The liquid with the highest concentration changed colour after only one day while the other two remained colourless for more than two weeks.

The rods filled with Primovist (#**11** to #**13**) contained some air bubbles already after two days. Moreover, the bubbles clinged to the walls and could not be moved by neither tilting nor knocking the rod. Only by shaking the rod could the bubbles be detached them from the walls.

The rod containing the low concentration of agar (#**14**) contained no air for more

than a week. When a bubble formed in the centre of the rod, it was impossible to coerce it to either end of the rod due to the liquid's viscous consistency. In rod #**15**, on the other hand, bubbles did not form at the middle of the rod. Instead the gel seemed to have dried starting at the end with the plastic stopper, leaving room for air where the material shrunk due to evaporation.

	after 1 day		*after 2 days*		*after 1 week*	
No.	bubbles	hit req.	bubbles	hit req.	bubbles	hit req.
#1	yes	no	no		no	
#2	yes	yes	no		no	
#3	yes	yes	no		no	
#4	yes	yes	no		no	
#5	yes	no	yes	no	no	
#6	yes	no	*rod was leaking*			
#7	yes	no	yes	no	yes	no
#8	no		no		no	
#9	no		no		no	
#10	no[1]		yes	yes	yes	yes
#11	no		yes,	*sticked to wall*	yes	yes
#12	yes	yes	yes,	*sticked to wall*	yes	yes
#13	yes	yes	yes,	*sticked to wall*	yes	yes
#14	no		yes	no	yes	yes
#15	no		no		no	
#16	no		no		no	
#16	no		no		no	

Table 3.2: Observations made in the first week regarding the mechanical properties of the tested solutions. Some rods contained bubbles which would move simply from buoyancy (when the rod was tilted or put upright), others required a knock to detach from the walls ('hit req.').

[1] liquid exhibited brown colouring.

No.	*after 2 months* length of trapped bubble l $[mm]$	approx. volume V $[mm^3]$
#1	2	25.13
#2	1.8	22.62
#3	1+1 (air blockage, at lower end)	25.13
#4	4	50.27
#5	1.5 (many small bubbles)	18.85
#6	*rod was leaking*	
#7	2 (many small bubbles)	25.13
#8	2.3	28.90
#9	3	37.70
#10	2.4	30.16
#11	2	25.13
#12	2	25.13
#13	2.3	28.90
#14	1.5+0.5 (big immobile bubble, at center)	25.13
#15	3.4 (agar gel dried)	42.73
#16	0	0.00
#17	0.5	6.28

Table 3.3: Measured air accumulated in observed rods.

Figure 3.3: Rod #5 showed some bubbles after 2 months.

Figure 3.4: Rod #16 contained no bubbles after more than 6 months.

Figure 3.5: The rest of the tested rods (all except #5 and #16) after more than 6 months.

3.3 Distortion assessment

Two sets of images were taken using CT and MRI. For the first, all possible liquids were scanned to find their signal intensity in MRI. A few months later, in the second imaging, one of the most promising candidates was imaged again on its own. Varying resample rates of both sets were used to test the developed software tool. If not mentioned otherwise, results and discussion refer to the scans resampled to a 100-times finer resolution (x100).

The output generated by the script is attached in the appendix. Tables 3.4 and 3.5 summarise the most interesting regions along the rods. They contain the calculated centroid shift in x and y direction and its magnitude ($warp_x$, $warp_y$, $warpM$); the DC for CT using the CT-centroid (DC_{CT}); MRI using the MRI-centroid (DC_{MR}); and MRI using the CT-centroid ($DC_{MR(CT-COM)}$). These six numbers were generated using the simple method (see figure 2.15) and the iteration method (figure 2.16) of finding COM and DC. The latter are marked in the table with a * (e.g. $warpM^*$).

3.3.1 First Set, Rod #5

From the first MRI and CT scans, only the rod containing liquid #5 was analysed. The hole in which it was placed is marked with a little 'x' in figure 2.8. It was decided to use this rod, because it had a reasonably good signal strength and it contained a small air bubble. This situation provided data containing a irregularity (bubble) and was therefore well suited for testing some of the software tool's capabilities and limitations. Figure 3.6 and 3.7 show the brightness of the rod on CT and MR scans. Figures 3.11, 3.8, 3.9 and 3.10 visualise the calculated output. All plots were created using the data obtained with the iteration method.

Figure 3.6: Rod #5: brightness of all pixels in each slice along z-axis, CT x100; the plastic pane fills the area from $-10mm$ to $10mm$.

slice	dist	$warp_x$	$warp_y$	$warpM$	DC_{CT}	DC_{MR}	$DC_{MR(CT-COM)}$	$warp^*_x$	$warp^*_y$	$warpM^*$	DC^*_{CT}	DC^*_{MR}	$DC^*_{MR(CT-COM)}$
0	-183	-0.1046	-1.115	1.1199	0.983	0.8444	0.6102	-0.1047	-1.0571	1.0623	0.9945	0.9504	0.7546
1	-182	-0.1205	-1.1132	1.1197	0.9807	0.8547	0.6091	-0.1199	-1.0527	1.0595	0.9929	0.9496	0.7573
2	-181	-0.1278	-1.1107	1.1181	0.9819	0.8666	0.6135	-0.1334	-1.0472	1.0556	0.9925	0.9501	0.7583
:													
:													
171	-12	0.0025	-0.6891	0.6891	0.9786	0.9421	0.7848	-0.0261	-0.6921	0.6926	0.9941	0.9528	0.8437
172	-11	0.0089	-0.6825	0.6825	0.9723	0.9425	0.7862	-0.026	-0.6869	0.6874	0.9898	0.9527	0.8445
173	-10	-1	-1	-1	-1	0.9421	-1	-1	-1	-1	-1	0.9529	-1
174	-9	-1	-1	-1	-1	0.9433	-1	-1	-1	-1	-1	0.9535	-1
:	:	:	:	:	:			:	:	:			
:	:	:	:	:	:			:	:	:			
192	9	-1	-1	-1	-1	0.9468	-1	-1	-1	-1	-1	0.9517	-1
193	10	-1	-1	-1	-1	0.9472	-1	-1	-1	-1	-1	0.9525	-1
194	11	0.0379	-0.709	0.71	0.9707	0.9481	0.778	0.0072	-0.7123	0.7123	0.9878	0.9531	0.8349
195	12	0.0364	-0.7048	0.7058	0.9731	0.9465	0.7789	0.0053	-0.7131	0.7132	0.9905	0.9535	0.8345
:													
:													
301	118	0.2997	-0.8702	0.9204	0.9838	0.4306	0.409	0.1153	-0.7938	0.8021	0.9909	0.8851	0.7757
302	119	0.3943	-1.1317	1.1984	0.981	0.145	0.1459	0.1144	-0.8896	0.8969	0.9903	0.8631	0.7477
303	120	-1	-1	-1	0.9817	-1	0	0.1093	-1.0022	1.0081	0.9915	0.832	0.7129
304	121	-1	-1	-1	0.9809	-1	0	0.0979	-1.1751	1.1792	0.9928	0.7853	0.6619
305	122	-1	-1	-1	0.9826	-1	0	0.0881	-1.4076	1.4104	0.9918	0.7164	0.5901
306	123	-1	-1	-1	0.982	-1	0	0.0787	-1.5728	1.5748	0.9918	0.6762	0.5439
307	124	-1	-1	-1	0.9812	-1	0	0.0866	-1.5327	1.5351	0.9917	0.7164	0.5705
308	125	-1	-1	-1	0.9805	-1	0	0.0927	-1.5189	1.5218	0.9925	0.7516	0.5913
309	126	-1	-1	-1	0.9811	-1	0	0.0915	-1.4745	1.4773	0.9922	0.7849	0.6167
310	127	0.3786	-2.0028	2.0383	0.981	0.0755	0.0219	0.0792	-1.4414	1.4436	0.9915	0.8148	0.6361
311	128	0.3434	-1.9836	2.0131	0.9808	0.1647	0.0613	0.0819	-1.3987	1.401	0.9916	0.843	0.6576
:													
:													
393	210	0.1083	-0.2563	0.2783	0.9822	0.9527	0.9048	0.0989	-0.2569	0.2753	0.9943	0.9675	0.9294
394	211	0.1147	-0.263	0.2869	0.9808	0.9522	0.9024	0.1039	-0.2608	0.2808	0.9917	0.9681	0.9287
395	212	0.1214	-0.2454	0.2738	0.9815	0.9552	0.9089	0.1114	-0.2424	0.2668	0.9921	0.9691	0.9311

Table 3.4: rod #5: script generated data; dist and *warp* in [*mm*].

Figure 3.7: Rod #5: brightness of all pixels in each slice along z-axis, MRI x100; drop of max brightness at around $110mm$ to $135mm$ is due to air bubble.

Figure 3.8: Rod #5: warp XY $[mm]$ (iteration method), CT-MRI x100; plastic pane present from $-10mm$ to $10mm$, bubble at about $110mm$ to $135mm$.

Figure 3.9: Rod #5: warp Magnitude [mm] (iteration method), CT-MRI x100; plastic pane present from $-10mm$ to $10mm$, bubble at about $110mm$ to $135mm$.

Figure 3.10: Rod #5: DC (iteration method) for CT & MRI & MRI (using CT COM), x100; plastic pane at $-10mm$ to $10mm$, bubble at about $110mm$ to $135mm$.

(a) CT @ 0 (b) CT @ 171 (c) CT @ 304

(d) MRI @ 0 (e) MRI @ 171 (f) MRI @ 304

Figure 3.11: Rod #5: images showing the location of MRI/CT centroids at various slices along z-axis; x100 (true colours). The dark blue dot represents the MRI centroid, the bright red dot represents the CT centroid.
Slice 171 is located approximately at the isocentre, 0 on the very end of the image, 304 on the other but also close to an air bubble.

3.3.2 rod #16

After deciding which of the possible liquids might be a suitable choice for the future use of the phantom, another set of MRI amd CT scans was taken. This second data set shows the rod containing liquid #16. The hole in which it was placed is marked with a little 'z' in figure 2.8. Figure 3.13 and 3.14 show the brightness of the rod on CT and MR scans. Figures 3.15, 3.16 and 3.17 visualise the spatial distortion assessed using the iteration method. All plots were created using the data obtained with the iteration method.

Figure 3.12: Rod #16: colour coded map; colours reflect $warpM^*$; based on x100

Figure 3.13: Rod #16: brightness of all pixels in each slice along z-axis, CT x100; the plastic pane fills the area from $-10mm$ to $10mm$.

slice	dist	$warp_x$	$warp_y$	$warpM$	DC_{CT}	DC_{MR}	$DC_{MR(CT-COM)}$	$warp_x^*$	$warp_y^*$	$warpM^*$	DC_{CT}^*	DC_{MR}^*	$DC_{MR(CT-COM)}^*$
0	-361	2.9415	0.4286	2.9725	0.9676	0.033	0.0062	3.3112	0.8531	3.4193	0.944	0.6126	0.3313
1	-360	2.94	0.4294	2.9712	0.9706	0.0331	0.0062	3.3129	0.8427	3.4184	0.9456	0.6126	0.3308
2	-359	2.9441	0.4224	2.9742	0.9725	0.0391	0.0077	3.2904	0.8202	3.3911	0.9475	0.6121	0.3346
:													
:													
351	-10	-0.2049	-0.0356	0.208	0.9896	0.9165	0.9091	-0.2696	-0.1097	0.2911	0.9901	0.9294	0.9248
352	-9	-0.2217	-0.0174	0.2223	0.9937	0.916	0.9112	-0.282	-0.0942	0.2973	0.9834	0.9283	0.9251
353	-8	-1	-1	-1	-1	0.916	-1	-1	-1	-1	-1	0.9274	-1
354	-7	-1	-1	-1	-1	0.9144	-1	-1	-1	-1	-1	0.9258	-1
355	-6	-1	-1	-1	-1	0.912	-1	-1	-1	-1	-1	0.9229	-1
356	-5	-1	-1	-1	-1	0.9113	-1	-1	-1	-1	-1	0.9209	-1
357	-4	-1	-1	-1	-1	0.909	-1	-1	-1	-1	-1	0.9186	-1
358	-3	-1	-1	-1	-1	0.9079	-1	-1	-1	-1	-1	0.9156	-1
359	-2	-1	-1	-1	-1	0.9082	-1	-1	-1	-1	-1	0.9153	-1
360	-1	-1	-1	-1	-1	0.9088	-1	-1	-1	-1	-1	0.9131	-1
361	0	-1	-1	-1	-1	0.9112	-1	-1	-1	-1	-1	0.9102	-1
362	1	-1	-1	-1	-1	0.9102	-1	-1	-1	-1	-1	0.9042	-1
363	2	-1	-1	-1	-1	0.9096	-1	-1	-1	-1	-1	0.9013	-1
364	3	-1	-1	-1	-1	0.9088	-1	-1	-1	-1	-1	0.896	-1
365	4	-1	-1	-1	-1	0.9068	-1	-1	-1	-1	-1	0.8931	-1
366	5	-1	-1	-1	-1	0.9047	-1	-1	-1	-1	-1	0.8908	-1
367	6	-1	-1	-1	-1	0.907	-1	-1	-1	-1	-1	0.8967	-1
368	7	-1	-1	-1	-1	0.9079	-1	-1	-1	-1	-1	0.9006	-1
369	8	-1	-1	-1	-1	0.9077	-1	-1	-1	-1	-1	0.903	-1
370	9	-1	-1	-1	-1	0.9082	-1	-1	-1	-1	-1	0.9057	-1
371	10	-1	-1	-1	-1	0.915	-1	-1	-1	-1	-1	0.9105	-1
372	11	-1	-1	-1	-1	0.92	-1	-1	-1	-1	-1	0.915	-1
373	12	-1	-1	-1	-1	0.9268	-1	-1	-1	-1	-1	0.9187	-1
374	13	-0.2506	0.1242	0.2797	0.9911	0.9281	0.9168	-0.2511	0.0942	0.2682	0.9867	0.9212	0.9397
375	14	-0.2181	0.1166	0.2473	0.9858	0.9298	0.9225	-0.223	0.0966	0.243	0.9937	0.9231	0.9436
:													
:													
609	248	3.0734	0.7814	3.1712	0.9838	0.917	0.3481	3.0773	0.9077	3.2084	0.9967	0.9221	0.5231
610	249	3.1422	0.7546	3.2315	0.984	0.9269	0.328	3.1288	0.8802	3.2503	0.9967	0.9312	0.5192
611	250	3.2214	0.7399	3.3053	0.9834	0.9359	0.3064	3.1835	0.8563	3.2967	0.9971	0.9282	0.5145

Table 3.5: rod #5: script generated data; dist and *warp* in [*mm*].

Figure 3.14: Rod #16: brightness of all pixels in each slice along z-axis, MRI x100

Figure 3.15: Rod #16: warp XY [mm] (iteration method), CT-MRI x100; plastic pane fills region from $-8mm$ to $12mm$

Figure 3.16: Rod #16: warp Magnitude (iteration method), CT-MRI x100; plastic pane fills region from $-8mm$ to $12mm$

Figure 3.17: Rod #16: DC (iteration method) for CT & MRI & MRI (using CT COM), x100; plastic pane fills region from $-8mm$ to $12mm$

4 Discussion

4.1 Phantom design

To be able to assess the spacial distortion of an MRI scan, a rigid object with known dimensions needs to be scanned so the cross referencing to the ground truth can be performed. Such phantoms are commercially available, but often expensive and designed for a specific calibration protocol. Some institutions build their own to fulfil exactly the requirements of a given application. The scanner used at the AKH is a relatively rare model, which is why no off-the-shelf phantom that would fit its coil is available.

A previously used phantom did not fill the whole FOV, while at the same time weighing about 45kg. This was due to the fact that it resembled a cuboid water tank with a thin filled or solid plastic rod construction inside. As the peripheral zones of the FOV are those where distortion is most pronounced, a bigger phantom was needed to assess those regions. The new design deals reaches the outer regions and weighs much less at the same time.

Due to the underlying physics, plastics are not visible in MR scans, but CT scans visualize them. See figure 2.6 for a comparison (MRI/CT visibility). Therefore, it was decided to use plastic rods with a suitable fluid filling. Such a liquid should be easily produced, non-toxic and yielding sufficient signal in MRI scans.

Commercially available phantoms often resemble water tanks containing rigid plastic grids used as reference. While this design results in stronger signal, its weight might be considered as inconvenient. There are few brands offering solutions utilising liquid fiducial markers in the shape of pellets. They are arranged in a regular pattern surrounded by air or plastic. The AKH's design, however, relies on replaceable rods, which makes it a novelty.

Observed issues

Interestingly, all water based solutions seemed to have evaporated partly. As the rod filled with liquid #**15** has dried starting at the end with the plastic stopper, it seems likely that, at least in this particular case, the plastic stopper did not effectively close the rod. It might also be that the rod itself does not prevent volatile liquids from escaping slowly. This was tested in a small experiment where a filled rod with only both its ends submerged in water, but the middle part in contact with air was observed over a period of some months (upright rod placed in a glass of water so that the lower end was surrounded by water, a plastic cup of water with a hole in its bottom through which the rod was stuck, glued to seal the hole). Just like in all other rods which were not partly submerged in water, air bubbles formed along its wall. This indicates that not (only) the plastic stopper, but the wall (which was the only part of the rod that was in contact with air) is too thin to keep the filling from evaporating. An airtight container might have led to other conclusions regarding the formation of air bubbles. It is hard to tell if they were caused by evaporation only or if dissolved gases played a role, too. Whatever the reasons are, the use of water based liquids seems to be suboptimal. Despite this, the observed behaviour will still be discussed as different phantom design might benefit from drawn conclusions.

4.2 Tested solutions

For measuring the position of the rods in the CT scans, the plastic rods without filling would be enough already. That is why the visibility of the liquids on CT is not important at all. Hollow plastic rods would not be visible on the MRI scans, though. From now on 'signal (strength)' or 'visibility' will refer to MRI scans only (see table 3.1).

4.2.1 Thoughts about choosing possible candidates

The tested solutions were chosen for a number of reasons:

- Generally, imaging techniques aims for a high signal-to-noise ratio. Therefore, liquids resulting in brighter pixels are favoured.
- The amount of gas in the rods should be minimised.
- If air bubbles form, tilting the entire phantom slightly should be enough to move them to one side. The FOV of the MRI scanner is too small to show the entire

phantom anyway.

- Most tested fluids are based on water, as they are easily removed from the rods leaving little residue. Therefore, filling the rods again with a different liquid would be possible if ever needed.
- Preferably, the components which are chosen to be used for the entire phantom should be non-toxic.

Rod #**1** was filled with plain distilled water and intended to be used only as a reference. It was clear from the beginning that it would not result in high signal and was never considered a possible filling.

Aiming for high SNR

To achieve a better SNR, liquids #**2** to #**10** and #**14** to #**15** are based on a solution of sodium chloride ($NaCl$ concentration of $0.36\,g/L$) and copper(II) sulfate pentahydrate ($CuSO_4 \cdot 5H_2O$ concentration of $1.96\,g/L$ as suggested by AAPM MR Subcommittee [37]; #**3** and #**4** contain double and ten-fold the concentration) in distilled water. Most of these liquids resulted in an about 5 times brighter signal than plain distilled water. Regarding the toxicity of $CuSO_4 \cdot 5H_2O$, the minimum dose to have caused acute toxic effects in humans is reported to be $11mg/Kg$.

Primovist (#**11** to #**13**) is a common contrast agent used for MRI scans [57]–[59] intended to yield an even stronger signal than $CuSO_4 \cdot 5H_2O$ based liquids. The major drawback is its tendency to separate from the water and stick to the container's wall. This results in low signal at the centre and high signal along the wall, which would not necessarily pose a problem, but as the liquid forms bubbles, it is not guaranteed that the walls would be covered homogeneously. The uneven distribution might result in wrong calculations, especially if the software tool is not programmed to cope with this behaviour. At the same time the removal of such a solution would be hardly possible.

Handling dissolved gas

Unfortunately, dissolved gases may eventually leave the liquid and form air bubbles trapped in the rod. To improve the mobility of trapped air bubbles, generic washing-up

soap was added (#**5**, #**6**, and #**7**; suggestion by Data Spectrum Corporation [60]). The higher concentrations of soap were tested as reference. If the liquid should happen to leak from the rod, the relatively low concentration of soap would not add to its toxicity. For those reasons, and because the liquid is cheap and easy to produce, it appears to be a promising candidate.

Liquids #**8** ($0.36g/L$), #**9** ($3.6g/L$) and #**10** ($36g/L$) contain ascorbic acid. Adding this was supposed to reduce forming of air bubbles by binding dissolved oxygen and eventually degrade to dehydro-ascorbic acid and water. The amount suggested by [61], [62] is 0.00204 mol/L which corresponds to approx. ($0.36g/L$).

In an attempt to limit the forming of gas, *agar* was used in solutions #**14** and #**15**. Agar and agarose are commonly used as basic reference material for MRI phantoms [63], [64]

Non water-based liquids

As an alternative to water based solutions, two oils were proposed. Since oil is neither soluble in air, nor able to evaporate, a rod completely filled with oil should stay free from air bubbles. Yet, oil is not as easily removed from a rod as a water based liquid. At the same time, it might not be necessary to ever replace the oil. Once filled, the rods could be used until the surrounding plastic breaks or starts leaking. Using vegetable oil would be a non-toxic solution, but has been ruled out as a filling from the beginning, because it would eventually rot. Synthetic oil on the other hand does not rot, however, it might be toxic if consumed.

4.2.2 Choosing a promising candidate

As all rods containing water continued to lose liquid due to evaporation, only the early forming of air bubbles might indicate whether solutions effectively hinder dissolved gases to result in trapped air bubbles. Apart from the solutions containing ascorbic acid (#**8** ($0.36g/L$), #**9** ($3.6g/L$)) all water based liquids produced some air bubbles after at least two days (see table 3.2.2). Considering the low visibility of #**9**, the only suitable water based liquid capable of staying free from air bubbles might be #**8**. At the same time,

long-term observations performed in an airtight container might have shown that even #**8** only delays the process. In the case of the used rods, such a conclusion cannot be drawn with certainty.

The rod containing the highest concentration of ascorbic acid showed a yellow colouring, caused by dehydroascorbic acid which is the result of a oxygenation process. However, as the high concentration in #**9** and #**10** led to a radical reduction in signal with the tested MRI sequence, this solutions is not considered a suitable filling anyway.

Primovist (#**11** to #**13**) lead to a good signal, but its limited mobility of air bubbles; its tendency to accumulate along the wall rule it out as a candidate.

If the forming of a small amount of gas is not considered a problem, adding soap appears to be a reasonable solution. The smallest tested amount of soap (#**5**, 1 g/L) was already enough to result in sufficient mobility of air bubbles, and an even lower concentration might also be acceptable. Interestingly, the rod filled with this solution contained the least amount of gas after 2 months, but this might be because the particular rod closed better than the others. The visibility recorded was among the higher candidates, too. For those reasons, and because the liquid is cheap and easy to produce, it is a promising candidate.

Solutions containing *agar* (#**14** and #**15**) are even harder to remove from rods if not impossible. As they might lead to air bubbles, too, which cannot be moved to either side of the rod, agar is not suited for this phantom.

Finally, the synthetic motor oil (#**16**) resulted in the highest signal intensity of all candidates. Besides the question of its toxicity, it seems to be a good alternative to water based liquids. The silicon oil (#**17**) on, the other side, had a low signal compared to most candidates and is therefore not suited.

The question of which liquid to use as a filling is adressed in section 5.1.

4.3 Distortion

4.3.1 Calculation Methods

`DC` and warp calculated with the simple and iteration method do not differ much. Moreover, both methods choose very similar thresholds for their calculations (see header of appended '.txt' output file) As the simple method uses additional information on the

rod's true dimension, it is supposed to yield accurate, reliable results. The iteration process, oblivious to the imaging modality, supports this claim as it produces very similar numbers.

The biggest difference in the results obtained with both methods are the values in the region of the air bubble in rod #5: Slices 303-310 in table 3.4 mark the area where the maximum brightness dropped significantly due to the absence of liquid (see figure 3.7). The simple method for finding the COM did not manage to calculate coordinates in this region, due to the lack of pixels above the threshold based on the reference slice (which was 827; see appendix). Consequently, warp and DC for MRI were set to '-1'. However, the iteration method tried various thresholds and settled for the one resulting in the highest average DC for the whole scan. As slices with values of '-1' influence the average dramatically, the chosen threshold is low enough (339; see appendix) to include some of the brightest pixels in slices 303-310.

(a) slice 175 (b) slice 176 (c) slice 177

Figure 4.1: Slices at edge of plastic pane in CT image showing rod #5 (true colours): a black crescent remains at the bottom where there is a hole in the pane that is not filled with a rod at the moment

4.3.2 Irregularities

All obtained CT scans include a region where the plastic panes are visible. Figure 4.1 shows how the plastic pane makes it impossible to define the rod on the CT image. These areas cannot be used for distortion assessment as the COM cannot be calculated reliably. Figures 3.6 and 3.13 exhibit spikes in the average brightness where the image shows a plastic pane. The script recognised this difference and marked all affected slices as 'irregular'. Consequently, in table 3.4 and 3.5, the values of the COM shift ($warp_x$, $warp_y$, $warpM$); the DC for CT using the CT-centroid (DC_{CT}); and the DC for MRI using the CT-centroid ($DC_{MR(CT-COM)}$) are all set to '-1'.

In the future, the threshold value used to determine a significant change in brightness could be automatically chosen based on the image modality and the image size.

4.3.3 Measured distortion - Rod #5

Figure 3.11 visualises the centroid shift at interesting slices of rod #5. It is clearly visible that the shift is bigger at the end of the rod (slice 0) compared to a slice close to its middle (slice 171). The values of $warpM^*$ in table 3.4 verify this (1.0623mm $>$ 0.6926mm). Furthermore, the distortion measured at the location of the air bubble (slice 304) shows a maximum for the COM shift ($warpM^* = 1.5748$mm) and a minimum for the DC ($DC^*_{MR} = 0.6762$). These values originate not from true spatial distortion, but are caused by the presence of gas in the rod. Since the bubble floated on top of the liquid, the COM shift is most pronounced in the y direction (up-down) as shown by figure 3.8. The overall impact on the area of 110mm to 140mm is clearly visible in figures 3.9 and 3.10. In conclusion, it is imperative for the distortion assessment to exclude bubbles from the FOV.

4.3.4 Measured distortion - Rod #16

Similar to the findings for rod #5, the distortion is bigger in the peripheral regions of the FOV (see figure 3.17). Interestingly, the y shift changes its orientation around the centre (see figure 3.15). This might be a local feature of the distortion and cannot be put in relation, because no distortion map of the entire FOV is available yet.

As rod #16 does not contain any air bubbles, the brightness plotted in figure 3.14 shows no sudden drops. However, the left hand side has a drastically lower signal strength than the right. A closer look revealed that rod #16 contained oil of two dif-

ferent colours. It seems that some residue of the other oil, which was injected into rod #17, was still present in the syringe when rod #16 was filled. To avoid this, the injecting syringe and the capillary were flushed with the next liquid before injection. This procedure was not enough to clean it thoroughly. The liquids separated into parts with different density while the rod was stored upright and were still distributed unevenly during imaging. To avoid this problem in the future, a new and unused syringe should be used for filling.

It is likely, that this influenced the calculated warp and DC as the lighter oil (which results in a different signal strength) floated on top of the other, resulting in an unwanted shift of the calculated COM. This might explain why the y shift changes its orientation close to the isocentre as this is the point where the signal intensity starts to change, too. To be sure of the actual impact of the oil mix on the distortion assessment, a new rod should be filled and the imaging repeated. For now, the already gained results will be used as no new measurement could be obtained in the limited time that the scanner is available for such experiments.

In the region beyond -330, there is a drop of the DC for the CT. Investigations revealed that this is caused by adhesive tape attached to the rod to hold it in place during transport from the CT scanner to the MRI scanner. The values calculated for this area are most likely influenced by this.

4.3.5 Effect of resampling

Figures 4.2 and 4.5 depict the measured $warpM^*$ of rod #5 and #16 for x1 and x100 resample rates. The location shift calculated with x1 and x100 resample rate are relatively close (see tables 4.1 and 4.2).

The calculated DCs for the scans in original CT resolution (x1) are shown in figures 4.3 and 4.6 and for a x4 finer resolution in figures 4.4 and 4.7. Compared to the DCs of the x100 resampled scans (figures 3.10 and 3.17), the low resolution x1 yields significantly less smooth curves. A resample rate of x4, on the other hand, is already much closer to the quality of the x100 scans.

While the original resolution might be sufficient for the calculation of the COM shift, the DC calculation benefits a lot from a interpolated image. It is easier to spot sig-

nificant changes as the plot shows less jumps. At the same time, the computing time for calculations scales with the resolution of the images and should be kept minimal, because a future assessment would include more than 300 individual rods.

Now that the appearance of the curves has been addressed, it will be discussed how the calculated distortion varies with increased resample rate.

Rod #5

Table 4.1 lists the expectancy and standard deviation (σ_Δ) of the difference between the calculated distortion for x1 and x100 resample rate of rod #5:

The expectancy value E was calculated as the sum of the difference in each slice over the number of slices; it is the average difference. A random deviation should result in some slices with positive and some with negative local differences. Therefore, the average should be close to zero. Otherwise, an increased resample rate would either result in an overall drop or rise of measured distortion. Indeed, E is relatively small compared to the typical COM shift (even smaller than σ_Δ), which indicates that the increased resample rate does only result in a smoother curve, but not in very different values overall.

The COM shift mostly lies in a range of 0.5 to 0.7mm; σ_Δ is approximately one order of magnitude smaller. To verify whether this hints at increased accuracy or not, an external reference would be needed. In any case, a difference of less than $0.1mm$ would not influence treatment planning and can be regarded negligible.

The value of the MRI DC lies between 0.9 and 1 (except for the air bubble region); σ_Δ is roughly a 45th of the average DC value $(0.9/0,02)$. This is small enough to be neglected, too.

	E	σ_Δ
$\Delta warpM^*$ [mm]	0.038	0.058
ΔDC^*_{MRI}	0.000518	0.019070
$\Delta DC^*_{MRI(CT-COM)}$	-0.008905	0.022050

Table 4.1: Expectancy and standard deviation of difference between calculated distortion for x1 and x100 resample rate; rod #5

Figure 4.2: Rod #5: warp Magnitude [mm] (iteration method), CT-MRI x1 and x100; plastic pane from $-10mm$ to $10mm$, bubble at about $110mm$ to $135mm$

Figure 4.3: Rod #5; x1: DC (iteration method) for CT & MRI & MRI (using CT COM); plastic pane from $-10mm$ to $10mm$, bubble at about $110mm$ to $135mm$

Figure 4.4: Rod #5; x4: DC (iteration method) for CT & MRI & MRI (using CT COM); plastic pane from $-10mm$ to $10mm$, bubble at about $110mm$ to $135mm$

Rod #16

Table 4.2 lists the expectancy and standard deviation σ_Δ of the difference between calculated distortion for x1 and x100 resample rate of rod #16:

Again, the expectancy E is relatively small; hinting at a low overall shift of the calculated values.

Just like with rod #5, σ_Δ for the COM shift is approximately one order of magnitude smaller than the occurring warp and can therefore also be regarded a negligible.

Similarly, σ_Δ for the MRI DC is approximately a 45th of the average DC value $(0.9/0,02)$ and can be neglected.

	E	σ_Δ
$\Delta warpM^*$ [mm]	0.005	0.025
ΔDC^*_{MRI}	0.004835	0.019823
$\Delta DC^*_{MRI(CT-COM)}$	0.012554	0.019429

Table 4.2: Expectancy and standard deviation of difference between calculated distortion for x1 and x100 resample rate; rod #16

Figure 4.5: Rod #16: warp Magnitude $[mm]$ (iteration method), CT-MRI x1 and x100; plastic pane from $-8mm$ to $12mm$

Figure 4.6: Rod #16; x1: DC (iteration method) for CT & MRI & MRI (using CT COM); plastic pane from $-8mm$ to $12mm$

Figure 4.7: Rod #16; x4: DC (iteration method) for CT & MRI & MRI (using CT COM); plastic pane from $-8mm$ to $12mm$

5 Conclusion and Outlook

5.1 "filling" for phantom

As topping up over 300 rods regularly is too time consuming, and all water-based liquids continued to evaporate, they are not long-term solution for this phantom. They are, however, useful for prototyping. For short time experiments with this phantom (not airtight rods) #**5** is recommended, because the soap allows the air to be moved out of the FOV. If another set of rods with airtight walls were obtained, adding ascorbic acid to the solution (a combination of #**5** and #**8**) might be an even better filling:

- distilled water
- $0.36\,g/L$ of $NaCl$
- $1.96\,g/L$ of $CuSO_4 \cdot 5H_2O$
- $1\,g/L$ of soap
- $0.36g/L$ of ascorbic acid

In that case, the forming of air bubbles might be either avoided entirely (due to the ascorbic acid) or delayed and then easily taken care of by tilting the whole phantom slightly to move the bubbles out of the FOV. The long-term behaviour of the mix might lead to adverse properties and should be tested, though.

In the current set-up, it seems best to fill the rods of the phantom with the type of oil that does not rot and yields high signal. The tested synthetic motor oil (#**16**) fits those requirements.

5.2 Recommended resample rates

The accuracy of the distortion assessment, especially of the DC calculation, can be enhanced by using interpolated scans. Resample rates of x4, x9, x25 and x100 were used to create finer images which were all analysed using the same algorithms. The original resolution (x1) might be sufficient for the COM shift calculation, but the DC curve gains

significant smoothness at already x4 finer resolution. This small interpolation could be already enough to increase accuracy while keeping necessary computing power low. A future distortion assessment of the whole FOV with 300 individual rods would profit from a effective procedure at low resolution. Using the simple method to find the COM would use less time, too.

5.3 Future improvement of software tool

The developed tool took a simplified approach to the problem by looking only at a single rod. Distortion assessment can be performed using the generated tool, however, it needs to be implemented on more general scale to be able to measure the distortion for all the rods automatically. This could be done by a auto-trace function which detects individual rods and applies the already implemented algorithms to each of them separately.

The two implemented methods of finding COM and DC need to be assessed themselves. In order to tell which of the two gets closer to the absolute truth, additional checks should be performed. A possible accuracy test could look like this: two CT scans of the phantom which differ only by a known displacement of one rod are registered and resampled. The software tool is used to calculate the COM shift between the images which can be compared to the real displacement.

At the moment the iteration method does not take into account the steepness of the DC curve. In some cases this might result in neglecting the left hand side close to very low percentages. This happens when the maximum lies roughly in the middle of the current range, but a little bit to the left. Because the slope is steeper on the left (close to 0%), a value representing the left hand side (also close to 0%) yields a much lower result than the value on the right hand side (flat slope). This should be taken into account during further improvement of the method.

In the current version of the software tool, only changes in the mean brightness are used to characterise irregularities. It might be usefull to consider drops in the peak brightness as irregularities, too. This way, the distortion would not be calculated in the region of an air bubble. For the time being, running the script over those regions might aid troubleshooting and give better understanding how to improve the code.

5.4 Scale of distortion

For the investigated position of the rod (rod #16), the obtained values for the occurring spatial distortion report a COM shift below $1mm$ in a region of at least $40cm$ (from -283mm to 185mm; simple method). This does not rule out the possibility of using the MR scanner for radiotherapy planning. Whether the distortion is small enough to guarantee accurate treatment planning can only be discussed after a map for the entire FOV has been created.

Bibliography

[1] International Atomic Energy Agency, *Radiation Oncology Physics: A Handbook for Teachers and Students*, E. B. Podgorsak, Ed. International Atomic Energy Agency, 2005, ISBN: 9201073046.

[2] A. Maidment and M. Yaffe, *Diagnostic Radiology Physics: A Handbook For Teachers And Students*. International Atomic Energy Agency, 2014, pp. 209–235, ISBN: 978-92-0-131010-1.

[3] K. Baumann, "Stem cells: A key to totipotency", *Nature Reviews Molecular Cell Biology*, vol. 18, no. 3, pp. 137–137, Feb. 2017, ISSN: 1471-0072. DOI: 10.1038/nrm.2017.9.

[4] M. Joiner and A. van der Kogel, *Basic Clinical Radiobiology*. Taylor & Francis Ltd., Apr. 11, 2009, 288 pp., ISBN: 9780340929667.

[5] A. Sodickson, P. F. Baeyens, K. P. Andriole, L. M. Prevedello, R. D. Nawfel, R. Hanson, and R. Khorasani, "Recurrent CT, cumulative radiation exposure, and associated radiation-induced cancer risks from CT of adults", *Radiology*, vol. 251, no. 1, pp. 175–184, Apr. 2009, ISSN: 0033-8419. DOI: 10.1148/radiol.2511081296.

[6] C. H. McCollough, A. N. Primak, N. Braun, J. Kofler, L. Yu, and J. Christner, "Strategies for reducing radiation dose in CT", *Radiologic Clinics of North America*, vol. 47, no. 1, pp. 27–40, Jan. 2009. DOI: 10.1016/j.rcl.2008.10.006.

[7] M. J. Murphy, J. Balter, S. Balter, J. A. BenComo, I. J. Das, S. B. Jiang, C.-M. Ma, G. H. Olivera, R. F. Rodebaugh, K. J. Ruchala, H. Shirato, and F.-F. Yin, "The management of imaging dose during image-guided radiotherapy: Report of the AAPM task group 75", *Medical Physics*, vol. 34, no. 10, pp. 4041–4063, Sep. 2007. DOI: 10.1118/1.2775667.

[8] A. Smith, W. Dillon, R. Gould, and M. Wintermark, "Radiation dose-reduction strategies for neuroradiology CT protocols", *American Journal of Neuroradiology*, vol. 28, no. 9, pp. 1628–1632, Oct. 2007. DOI: 10.3174/ajnr.a0814.

[9] A. R. Goldman and P. D. Maldjian, "Reducing radiation dose in body CT: A practical approach to optimizing CT protocols", *American Journal of Roentgenology*, vol. 200, no. 4, pp. 748–754, Apr. 2013. DOI: 10.2214/ajr.12.10330.

[10] D. J. Brenner, C. D. Elliston, E. J. Hall, and W. E. Berdon, "Estimated risks of radiation-induced fatal cancer from pediatric CT", *American Journal of Roentgenology*, vol. 176, no. 2, pp. 289–296, 2001, ISSN: 0361803X. DOI: 10.2214/ajr.176.2.1760289. arXiv: 01/1762âĂŞ289 [0361-803X].

[11] S. Currie, N. Hoggard, I. J. Craven, M. Hadjivassiliou, and I. D. Wilkinson, "Understanding MRI: Basic MR physics for physicians", *Postgraduate Medical Journal*, vol. 89, no. 1050, pp. 209–223, 2013, ISSN: 0032-5473. DOI: 10.1136/postgradmedj-2012-131342. eprint: http://pmj.bmj.com/content/89/1050/209.full.pdf.

[12] C. Rasch, I. Barillot, P. Remeijer, A. Touw, M. van Herk, and J. V. Lebesque, "Definition of the prostate in CT and MRI: A multi-observer study", *International Journal of Radiation Oncology Biology Physics*, vol. 43, no. 1, pp. 57–66, Jan. 1999. DOI: 10.1016/s0360-3016(98)00351-4.

[13] M. Debois, R. Oyen, F. Maes, G. Verswijvel, G. Gatti, H. Bosmans, M. Feron, E. Bellon, G. Kutcher, H. V. Poppel, and L. Vanuytsel, "The contribution of magnetic resonance imaging to the three-dimensional treatment planning of localized prostate cancer", *International Journal of Radiation Oncology Biology Physics*, vol. 45, no. 4, pp. 857–865, Nov. 1999. DOI: 10.1016/s0360-3016(99)00288-6.

[14] M. Roach, P. Faillace-Akazawa, C. Malfatti, J. Holland, and H. Hricak, "Prostate volumes defined by magnetic resonance imaging and computerized tomographic scans for three-dimensional conformal radiotherapy", *International Journal of Radiation Oncology Biology Physics*, vol. 35, no. 5, pp. 1011–1018, Jul. 1996. DOI: 10.1016/0360-3016(96)00232-5.

[15] "Reduction of claustrophobia with short-bore versus open magnetic resonance imaging: A randomized controlled trial", *PLoS ONE*, vol. 6, no. 8, J. Laks, Ed., Aug. 2011. DOI: 10.1371/journal.pone.0023494.

[16] C. Bangard, J. Paszek, F. Berg, G. Eyl, J. Kessler, K. Lackner, and A. Gossmann, "MR imaging of claustrophobic patients in an open 1.0 T scanner: Motion artifacts and patient acceptability compared with closed bore magnets", *European Journal*

of Radiology, vol. 64, no. 1, pp. 152–157, 2007, ISSN: 0720048X. DOI: 10.1016/j.ejrad.2007.02.012.

[17] A. M. Coffey, M. L. Truong, and E. Y. Chekmenev, "Low-field MRI can be more sensitive than high-field MRI", *Journal of Magnetic Resonance*, vol. 237, pp. 169–174, 2013, ISSN: 10960856. DOI: 10.1016/j.jmr.2013.10.013. arXiv: NIHMS150003.

[18] A. Fransson, P. Andreo, and R. Pötter, "Aspects of MR image distortions in radiotherapy treatment planning", *Strahlentherapie und Onkologie*, vol. 177, no. 2, pp. 59–73, Jan. 2001. DOI: 10.1007/pl00002385.

[19] B. K. Rutt and D. H. Lee, "The impact of field strength on image quality in MRI", *Journal of magnetic resonance imaging : JMRI*, vol. 6, no. 1, pp. 57–62, 1996, ISSN: 1053-1807. DOI: 10.1002/jmri.1880060111.

[20] D. W. McRobbie, E. A. Moore, M. J. Graves, and M. R. Prince, *MRI from Picture to Proton*. Cambridge University Press, 2006, pp. 329–335, ISBN: 9780521865272.

[21] K. Nakamura and (. D. Group), "Passage of particles through matter", *Journal of Physics*, vol. G, no. 37, p. 075 021, 2010, ISSN: 1434-6044. [Online]. Available: http://iopscience.iop.org/0954-3899/37/7A/075021/media/rpp2010_0001-0007.pdf (visited on 09/29/2017).

[22] H. Paganetti, T. Bortfeld, and H. Kooy, "Proton beam radiotherapy - The state of the art", *Medical Physics*, vol. 32, no. 6Part13, pp. 2048–2049, 2005, ISSN: 2473-4209. DOI: 10.1118/1.1999671. [Online]. Available: http://www.aapm.org/meetings/05AM/pdf/18-4016-65735-22.pdf (visited on 09/29/2017).

[23] P. Andrzejewski, P. Kuess, B. Knäusl, K. Pinker, P. Georg, J. Knoth, D. Berger, C. Kirisits, G. Goldner, T. Helbich, R. Pötter, and D. Georg, "Feasibility of dominant intraprostatic lesion boosting using advanced photon-, proton- or brachytherapy", *Radiotherapy and Oncology*, vol. 117, no. 3, pp. 509–514, Dec. 2015. DOI: 10.1016/j.radonc.2015.07.028.

[24] C. Constantinou, J. C. Harrington, and L. A. DeWerd, "An electron density calibration phantom for CT-based treatment planning computers.", *Medical Physics*, vol. 19, no. 2, pp. 325–327, 1992, ISSN: 00942405. DOI: 10.1118/1.596862.

[25] U. Schneider, E. Pedroni, and A. Lomax, "The calibration of CT Hounsfield units for radiotherapy treatment planning.", *Physics in medicine and biology*, vol. 41, no. 1, pp. 111–24, 1996, ISSN: 0031-9155. DOI: 10.1088/0031-9155/41/1/009. arXiv: arXiv:1011.1669v3.

[26] C. M. Rank, N. Hünemohr, A. M. Nagel, M. C. Röthke, O. Jäkel, and S. Greilich, "MRI-based simulation of treatment plans for ion radiotherapy in the brain region", *Radiotherapy and Oncology*, vol. 109, no. 3, pp. 414–418, 2013, ISSN: 01678140. DOI: 10.1016/j.radonc.2013.10.034.

[27] T. Stanescu, J. Hans-Sonke, P. Stavrev, and B. G. Fallone, "3T MR-based treatment planning for radiotherapy of brain lesions", *Radiology and Oncology*, vol. 40, no. 2, pp. 125–132, 2006, ISSN: 13182099. [Online]. Available: http://www.scopus.com/inward/record.url?eid=2-s2.0-33746661222&partnerID=tZOtx3y1 (visited on 09/29/2017).

[28] J. H. Jonsson, M. G. Karlsson, M. Karlsson, and T. Nyholm, "Treatment planning using mri data: An analysis of the dose calculation accuracy for different treatment regions", *Radiation Oncology*, vol. 5, pp. 62–, 2010. DOI: 10.1186/1748-717X-5-62.

[29] P. Greer, D. Ph, J. Dowling, D. Ph, P. Pichler, J. Sun, M. Sc, H. Richardson, B. M. R. Sc, D. Rivest-henault, D. Ph, S. Ghose, D. Ph, J. Martin, C. Wratten, J. Arm, L. Best, J. Denham, and P. Lau, "Development of MR-only Planning for Prostate Radiation Therapy Using Synthetic CT", pp. 61–62, 2015.

[30] L. Chen, R. A. Price, L. Wang, J. Li, L. Qin, S. McNeeley, C. M. C. Ma, G. M. Freedman, and A. Pollack, "MRI-based treatment planning for radiotherapy: Dosimetric verification for prostate IMRT", *International Journal of Radiation Oncology Biology Physics*, vol. 60, no. 2, pp. 636–647, 2004, ISSN: 03603016. DOI: 10.1016/j.ijrobp.2004.05.068.

[31] R. G. Price, M. Kadbi, J. Kim, J. Balter, I. J. Chetty, and C. K. Glide-Hurst, "Technical note: Characterization and correction of gradient nonlinearity induced distortion on a 1.0 t open bore MR-SIM", *Medical Physics*, vol. 42, no. 10, pp. 5955–5960, Sep. 2015, ISSN: 0094-2405. DOI: 10.1118/1.4930245.

[32] B. Petersch, J. Bogner, A. Fransson, T. Lorang, and R. Pötter, "Effects of geometric distortion in 0.2 T MRI on radiotherapy treatment planning of prostate cancer",

Radiotherapy and Oncology, vol. 71, no. 1, pp. 55–64, 2004, ISSN: 01678140. DOI: 10.1016/j.radonc.2003.12.012.

[33] T. Torfeh, R. Hammoud, M. McGarry, N. Al-Hammadi, and G. Perkins, "Development and validation of a novel large field of view phantom and a software module for the quality assurance of geometric distortion in magnetic resonance imaging.", *Magnetic Resonance Imaging*, vol. 33, no. 7, pp. 939–949, 2015, ISSN: 0730725X. DOI: 10.1016/j.mri.2015.04.003.

[34] D. Wang, D. M. Doddrell, and G. Cowin, "A novel phantom and method for comprehensive 3-dimensional measurement and correction of geometric distortion in magnetic resonance imaging", *Magnetic Resonance Imaging*, vol. 22, no. 4, pp. 529–542, 2004, ISSN: 0730725X. DOI: 10.1016/j.mri.2004.01.008.

[35] D. Wang, W. Strugnell, G. Cowin, D. M. Doddrell, and R. Slaughter, "Geometric distortion in clinical MRI systems: Part II: Correction using a 3D phantom", *Magnetic Resonance Imaging*, vol. 22, no. 9, pp. 1223–1232, 2004, ISSN: 0730725X. DOI: 10.1016/j.mri.2004.08.014.

[36] T. Mizowaki, Y. Nagata, K. Okajima, M. Kokubo, Y. Negoro, N. Araki, and M. Hiraoka, "Reproducibility of geometric distortion in magnetic resonance imaging based on phantom studies", *Radiotherapy and Oncology*, vol. 57, no. 2, pp. 237–242, 2000, ISSN: 01678140. DOI: 10.1016/S0167-8140(00)00234-6.

[37] E. F. Jackson, "MR Acceptance Testing and Quality Control : Previous MR Committee TG Reports MRI Acceptance Testing - E . Jackson Levels of Involvement MR Acceptance Test Reality Check MR Siting Issues", vol. 17, no. 1, pp. 287–295, 2009.

[38] S. AG, *System owner manual - mr compatibility data sheet - magnetom c!*, 2013, pp. 89–116.

[39] J. Blue. (Oct. 3, 2017). Ingredients in soft gel capsules, [Online]. Available: http://www.livestrong.com/article/138531-ingredients-soft-gel-capsules/ (visited on 10/05/2017).

[40] W. Henry, "Experiments on the Quantity of Gases Absorbed by Water, at Different Temperatures, and under Different Pressures", *Philosophical Transactions of the Royal Society of London*, vol. 93, no. 1, pp. 29–274, Jan. 1803, ISSN: 02610523. DOI: 10.1098/rstl.1803.0004.

[41] R. Sander, "Compilation of Henry's law constants (version 4.0) for water as solvent", *Atmospheric Chemistry and Physics*, vol. 15, no. 8, pp. 4399–4981, 2015, ISSN: 16807324. DOI: 10.5194/acp-15-4399-2015.

[42] (). Mirada medical, [Online]. Available: http://mirada-medical.com/ (visited on 10/05/2017).

[43] (). 3D Slicer, [Online]. Available: https://www.slicer.org/ (visited on 09/29/2017).

[44] R. Kikinis. (2012). Overview: functions of 3D Slicer, [Online]. Available: http://perk.cs.queensu.ca/sites/perk.cs.queensu.ca/files/2014-04-30-Kikinis-Slicer-reduced.pdf (visited on 09/29/2017).

[45] (). Python 2.7.13, [Online]. Available: https://www.python.org/ (visited on 01/10/2017).

[46] (). DICOM: About DICOM, [Online]. Available: http://dicom.nema.org/Dicom/about-DICOM.html (visited on 01/08/2017).

[47] (). SimpleITK, [Online]. Available: http://www.simpleitk.org/ (visited on 12/19/2016).

[48] (). SimpleITK - Wiki, [Online]. Available: https://itk.org/Wiki/SimpleITK (visited on 01/10/2017).

[49] (). SimpleITK: IPython Notebooks, [Online]. Available: http://insightsoftwareconsortium.github.io/SimpleITK-Notebooks/ (visited on 12/19/2016).

[50] A. Kyriakou. (). Image Segmentation with Python and SimpleITK, [Online]. Available: https://pyscience.wordpress.com/2014/10/19/image-segmentation-with-python-and-simpleitk/ (visited on 09/29/2017).

[51] (). Pydicom, [Online]. Available: http://pydicom.readthedocs.io/en/stable/getting_started.html (visited on 01/10/2017).

[52] ——, (). DICOM in Python: Importing medical image data into NumPy with PyDICOM and VTK, [Online]. Available: https://pyscience.wordpress.com/2014/09/08/dicom-in-python-importing-medical-image-data-into-numpy-with-pydicom-and-vtk/ (visited on 01/10/2017).

[53] J. D. Hunter, "Matplotlib: A 2d graphics environment", *Computing in Science & Engineering*, vol. 9, no. 3, pp. 90–95, 2007. DOI: 10.1109/mcse.2007.55. [Online]. Available: https://matplotlib.org/ (visited on 09/29/2017).

[54] (). MedPy 0.2.2 : Python Package Index, [Online]. Available: https://pypi.python.org/pypi/MedPy (visited on 12/19/2016).

[55] (). medpy.metric.binary — MedPy 0.2.2 documentation, [Online]. Available: http://pythonhosted.org/MedPy/_modules/medpy/metric/binary.html#dc (visited on 12/19/2016).

[56] O. Maier. (2015). medpy.metric.binary.dc — MedPy 0.2.2 documentation, [Online]. Available: http://pythonhosted.org/MedPy/generated/medpy.metric.binary.dc.html#medpy.metric.binary.dc (visited on 12/19/2016).

[57] B. E. Van Beers, C. M. Pastor, and H. K. Hussain, "Primovist, eovist: What to expect?", *Journal of Hepatology*, vol. 57, no. 2, pp. 421–429, Aug. 2012, ISSN: 01688278. DOI: 10.1016/j.jhep.2012.01.031.

[58] M. Rohrer, H. Bauer, J. Mintorovitch, M. Requardt, and H.-J. Weinmann, "Comparison of Magnetic Properties of MRI Contrast Media Solutions at Different Magnetic Field Strengths",

[59] Bayer-Group. (Jul. 7, 2007). Primovist: Physico-Chemical Properties, [Online]. Available: http://www.medsafe.govt.nz/profs/Datasheet/p/primovistsol.pdf (visited on 09/29/2017).

[60] Data Spectrum Corporation. (). Frequently Asked Questions - How do I eliminate air bubbles in the Mini, Micro and Ultra Micro line of phantoms?, [Online]. Available: http://www.spect.com/faq.html (visited on 09/29/2017).

[61] S. M. Abtahi, M. Shahriari, M. H. Zahmatkesh, H. Khalafi, S. Akhlaghpoor, and S. Bagheri, "A new approach to contrast enhancement in MAGICA gel dosimeter image with MRI technique", *Iranian Journal of Radiation Research*, vol. 6, no. 3, pp. 151–156, 2008, ISSN: 17284562.

[62] R. S. Bodannes and P. C. Chan, "Ascorbic acid as a scavenger of singlet oxygen", *FEBS Letters*, vol. 105, no. 2, pp. 195–196, Sep. 1979, ISSN: 00145793. DOI: 10.1016/0014-5793(79)80609-2.

[63] M. Bucciolini, L. Ciraolo, and B. Lehmann, "Simulation of biologic tissues by using agar gels at magnetic resonance imaging", *Acta Radiologica*, vol. 30, no. 6, pp. 667–669, Jan. 1989. DOI: 10.3109/02841858909174735.

[64] R. Mathur-De Vre, R. Grimee, F. Parmentier, and J. Binet, "The use of agar gel as a basic reference material for calibrating relaxation times and imaging parameters.", *Magnetic resonance in medicine : official journal of the Society of Magnetic Resonance in Medicine / Society of Magnetic Resonance in Medicine*, vol. 2, no. 2, pp. 176–179, 1985, ISSN: 0740-3194. DOI: 10.1002/mrm.1910020208.

Appendix: Script & Output

The current version of the python script 'FunITK.py' with all functions described in this thesis used to calculate the presented numbers is available at:
`https://github.com/cyberspeck/mri-for-rt/blob/master/scripts`

The data of the cropped images showing rod #5 and #16, the created '.txt' and '.mha' files are stored at:
`https://github.com/cyberspeck/mri-for-rt/tree/master/data`

The following pages contain today's version of the python script ('FunITK.py'), the commands used to process scans of rod #5 ('rod_5.py') & #16 ('rod_16.py') and the generated data for rod #5 ('rod #5 output') & #16 ('rod #16 output') using a resample rate of x100.

```python
# -*- coding: utf-8 -*-
"""
Created on Sat Jul 16 22:45:11 2016
@author: david

Volume class
custom FUNcitons using SimpleITK
https://itk.org/Wiki/SimpleITK/GettingStarted#Generic_Distribution

works only with cropped CT and MRI images (showing only one rod),
both Volumes should have the same PixelSpacing,
 and x and y PixelSpacing shoult be equal
sitk_write() creates .mha file with pixel values corresponding
 to distortion in pixel distance * PixelSpacing (mm)

important to remember:
    sitk.Image saves Volume like this (x,y,z)
    array returned by sitk.GetArrayFromImage(Image)
    is transposed: (z,y,x)

based on:
https://pyscience.wordpress.com/2014/10/19/image-segmentation-with-python-and-SimpleITK/

"""

import numpy as np
from scipy import ndimage
import SimpleITK as sitk
import matplotlib.pyplot as plt
import os
from skimage.draw import circle


class Volume:
    '''
    Create a Volume (SimpleITK.Image with convenient properties and functions)
    recommended use:
    create new Volume (optional use denoise=True)
    Volume.getThresholds()

    Parameters
    ----------
    path : string_like
        directory containing DICOM data
    method : string_like, recommended
        either "CT or "MR", used for automatic calculations
    denoise : bool, optional
        If true, the imported data will be denoised using
        SimpleITK.CurvatureFlow(image1=self.img,
                                timeStep=0.125,
                                numberOfIterations=5)
    ref : int, optional
        slice used to make calculations (idealy isocenter) e.g. thresholds
        all plots show this slice
        by default it is set to be in the middle of the image (z-axis)
    resample : int, optional
        resample rate, becomes part of title
    seeds : array_like (int,int,int), optional
        coordinates (pixel) of points inside rod, used for segmentation
        by default list of brightest pixel in each slice
    radius: double, optional
        overrides radius value (default CT:4mm, MR:2mm)
    spacing: double, optional
        by default SitpleITK.img.GetSpacing is used to find relation of pixels
        to real length (in mm)
    skip: int, optional
        neglecting first 'skip' number of slices
    leave: int, optional
        neglects last 'leave' number of slices
    rotate: bool, optional
        if True: mirrors x- & z-axis, effectively rotating the image by 180 deg
        (looked at from above), this is applied after skip&size
```

```
    '''
    def __init__(self, path=None, method=None, denoise=False, ref=None,
                 resample=False, seeds='auto', radius=0, spacing=0, skip=0,
                 leave=False, rotate=False):
        if(path is None):
            print("Error: no path given!")
        else:
            self.path = path
            self.method = method
            self.denoise = denoise
            self.resample = resample
            self.centroid = False
            self.mask = False
            self.masked = False
            self.title = method
            self.radius = radius
            self.bestRadius = 0
            self.lower = False
            self.upper = False

            file_no = len([name for name in os.listdir(path)
                           if os.path.isfile(os.path.join(path, name))])
            size = file_no - skip - leave
            if size <= 0:
                print("There nothing left to load after skipping {} file(s) "
                      "and ignoring the last {} files.".format(skip,leave))
                print("The directory only contains {} files!".format(file_no))
            else:

                print("Import {} DICOM Files from: {}\n".format(size, path))
                shortened_img = sitk_read(path, denoise)[
                                    :, :, skip:(file_no + skip - leave)]
                if rotate is True:
                    self.img = shortened_img[::-1,:,::-1]
                else:
                    self.img = shortened_img

                if (self.img and self.denoise):
                    a = self.title
                    self.title = a + " denoised"

                if resample:
                    a = self.title
                    self.title = a + ", x" + str(resample)

                self.xSize, self.ySize, self.zSize = self.img.GetSize()
                if spacing == 0:
                    self.xSpace, self.ySpace, self.zSpace = self.img.GetSpacing()

                if type(ref) == int:
                    self.ref = ref
                else:
                    self.ref = int(self.zSize / 2)

                # niceSlice used to remember which slices show irregularities such
                # as parts of plastic pane (CT)
                # and should therefore not be used to calculate COM, dice, etc.
                self.niceSlice = np.ones((self.zSize, 1), dtype=bool)
                self.maxBrightness = np.zeros((self.zSize, 1))
                self.meanBrightness = np.zeros((self.zSize, 1))
                arr = sitk.GetArrayFromImage(self.img)
                average = np.average(arr[ref])
    #               print("\nAverage @ ref: ", average)
                for index in range(self.zSize):
                    # save value of brightest pixel in each slice
                    self.maxBrightness[index] = arr[index].max()
                    self.meanBrightness[index] = np.average(arr[index])
                    # if average value of slice differs too much -> badSlice
                    # difference between ref-Slice and current chosen arbitratry
                    # seems to be big enough not to detect air bubble in MRI
                    # entire air block (no liquid) should be recognised, though.
                    # small enough to notice plastic pane
```

```
                    if np.absolute(self.meanBrightness[index] - average) > 40:
                        print("Irregularities detected in slice {}".format(index))
                        self.niceSlice[index] = False
                        # maybe also set slice prior and after current slice as
                        # self.niceSlice[index+1] = self.niceSlice[index+1] = False
                        # because small changes happening around irregularities
                        # might not have been big enough for detection, but already
                        # leading to false calculations?

                if type(seeds) == list:
                    self.seeds = seeds
                elif seeds == 'auto':
                    self.seeds = []
                    for index in range(self.zSize):
                        yMax = int(arr[index].argmax() / self.xSize)
                        xMax = arr[index].argmax() - yMax*self.xSize
                        if self.niceSlice[index] == True:
                            self.seeds.append((xMax, yMax, index))
    #                   print("{}: found max at ({},{})"
    #                         .format(index, xMax, yMax))

    def show(self, pixel=False, interpolation=None, ref=None, save=False):
        '''
        plots ref slice of Volume

        Parameters
        ----------
        pixel: bool, optional
            if True, changes axis from mm to pixels
        interpolation: "string", optional, default: 'nearest'
            using build-in interpolation of matplotlib.pyplot.imshow
            Acceptable values are 'none', 'nearest', 'bilinear', 'bicubic',
            'spline16', 'spline36', 'hanning', 'hamming', 'hermite', 'kaiser',
            'quadric', 'catrom', 'gaussian', 'bessel', 'mitchell', 'sinc',
            'lanczos'
        ref: int, optional
            slice to be plotted instead of self.ref (default: 0)
        '''

        if ref is None:
            ref = self.ref

        if interpolation is None:
            a = 'nearest'

        extent = None
        if pixel is False:
            extent = (-self.xSpace/2, self.xSize*self.xSpace - self.xSpace/2,
                      self.ySize*self.ySpace - self.ySpace/2, -self.ySpace/2)
# The location, in data-coordinates, of the lower-left and upper-right corners
# (left, right, bottom, top)

        sitk_show(img=self.img, ref=ref, extent=extent, title=self.title,
                  interpolation=a, save=save)

    def showSeed(self, pixel=False, interpolation='nearest', ref=None, save=False):
        '''
        plots slice containing seed

        Parameters
        ----------
        pixel: bool, optional
            if True, changes axis from mm to pixels
        interpolation: "string", optional, default: 'nearest'
            using build-in interpolation of matplotlib.pyplot.imshow
            Acceptable values are 'none', 'nearest', 'bilinear', 'bicubic',
            'spline16', 'spline36', 'hanning', 'hamming', 'hermite', 'kaiser',
            'quadric', 'catrom', 'gaussian', 'bessel', 'mitchell', 'sinc',
            'lanczos'
        ref: int, optional
            slice of seed to be plotted instead of self.ref (default: zSize/2)
        '''
```

```
        if ref is None:
            ref = self.ref

        if type(self.seeds[ref]) != tuple:
            print("No seed found @ slice {}".format(ref))
            return None

        x, y = -1,-1
        extent = None
        if pixel is False:
            extent = (-self.xSpace/2, self.xSize*self.xSpace - self.xSpace/2,
                      self.ySize*self.ySpace - self.ySpace/2, -self.ySpace/2)
            x = (self.seeds[ref][0] * self.xSpace)
            y = (self.seeds[ref][1] * self.xSpace)
        else:
            x, y, z = self.seeds[ref]

        arr = sitk.GetArrayFromImage(self.img)
        fig = plt.figure()
        plt.set_cmap("gray")
        plt.title(self.title + ", seed @ {}".format(self.seeds[ref]))

        plt.imshow(arr[ref, :, :], extent=extent, interpolation=interpolation)
        plt.scatter(x, y)
        plt.show()
        if save != False:
            fig.savefig(str(save) + ".png")

    def getThresholds(self, pixelNumber=0, scale=1):
        '''
        Calculates threshold based on number of pixels representing rod.
        If no pixelNumber is given, self.radius is used to get estimated
        pixelNumber. If self.raduis == 0: use method to get raduis
         All calculations based on ref-slice.

        approx. number of pixels being part of rod:
        pn = realRadius^2 * pi / pixelSpacing^2

        Parameters
        ----------
        pixelNumber: int, optional
            if 0, uses self.radis to calculate pixelnumber
            if self.radius also 0, uses self.method instead (CT: 4mm, MR: 2mm)
        scale: double, optional
            factor altering pixelNumber

        Returns
        -------
        lower and upper threshold value: (double, double)
        '''

        if pixelNumber == 0:
            if self.radius != 0:
                realRadius = self.radius
            else:
                if self.method == "CT":
                    realRadius = 4
                if self.method == "MR":
                    realRadius = 2
                if self.method != "MR" and self.method != "CT":
                    print("method is unknown, please set pixelNumber!")
                    return None
            pixelNumber = (np.power(realRadius, 2) * np.pi
                           / np.power(self.xSpace, 2) * scale)

        pn = int(pixelNumber)
        arr = sitk.GetArrayFromImage(self.img)
        self.upper = np.double(arr.max())

    #    hist, bins = np.histogram(arr[self.ref, :, :].ravel(), bins=100)
    # alternatively, increase number of bins for images with many pixels
        hist, bins = np.histogram(arr[self.ref, :, :].ravel(), bins=int(pn*2))
```

```
        self.lower = np.double(bins[np.argmax((np.cumsum(hist[::-1]) < pn)[::-1])])
        print("number of pixels (pn): {}\n "
              "lower: {}\n "
              "upper: {}".format(pn, self.lower, self.upper))

        return (self.lower, self.upper)

    def getCentroid(self, threshold='default', pixelNumber=0, scale=1,
                    percentLimit=False, iterations=5, top = 1,
                    plot=False, save=False):
        '''
        Calculates centroid, either by setting threshold or percentLimit

        Parameters
        ----------
        threshold: float or 'auto', default='auto'
            if 'auto': uses getThreshold(pixelnumber, scale) and then
                sitk_centroid(threshold=self.lower)
                sets self.lower and self.upper
        percentLimit: float from 0 to 1 (or "auto" =experimental)
            if percentLimit is True: used instead of threshold method
            if 'auto': makes 5 iterations by default, uses getThreshold()
                and getDice(), but does NOT set self.mask
                sets self.lower and self.upper
        plot, save: bool, optional
            plot and save iteration (percentLimit='auto')

        Returns
        -------
        self.centroid: numpy.ndarray
        '''

        if ((threshold is False and percentLimit is False)
            or (percentLimit == "auto" and threshold is not False
                and threshold != 'default')):
            print("Please use percentLimit or threshold! "
                  "(default setting: threshold = 'auto')")
            return None

        if ((percentLimit == "auto" and threshold is False) or
            (percentLimit == "auto" and threshold == 'default')):
            # EXPERIMENTAL!!!
            # looks at whole range of possible percentLimits
            # reduces range by finding out which half yields higher result
            # starts at A=25% and B=75% of all pixels
            # if DC(A) > DC(B): next values come from lower half (0-50%)
            # else: upper half (50-100%)
            # calculates 5 centroids with different percentLimits
            # gets dice coefficient for each centroid percentLimit combination
            # returns best result

            print("\n\n")
            arr = sitk.GetArrayFromImage(self.img)
            direction = np.zeros(iterations)
            left = np.zeros(iterations)
            right = np.zeros(iterations)
            left[0] = 0
            right[0] = top
            guess = np.zeros(iterations)
            guess[0] = (left[0]+right[0])/2
            thresholdsA = np.zeros((iterations,2))
            thresholdsB = np.zeros((iterations,2))
            centroidScoreA = np.zeros(iterations)
            centroidScoreB = np.zeros(iterations)
            centroidsA = np.zeros((iterations, self.zSize, 2))
            centroidsB = np.zeros((iterations, self.zSize, 2))
            diceA = np.zeros((iterations, self.zSize, 1))
            diceB = np.zeros((iterations, self.zSize, 1))
            for index in range(iterations):
                print("    ITERATION #{}, current guess: ~{:.4f}\n"
                      "A @ ~{:.4f}%"
                      .format(index, guess[index] *100,
```

```
                                (guess[index] + left[index]) / 2*100))
                    thresholdsA[index] = self.getThresholds(pixelNumber=self.xSize
                        * self.ySize * (guess[index] + left[index]) / 2)
                    # create mask including all pixels relevant for guess
                    maskA = sitk.ConnectedThreshold(image1=self.img,
                                                    seedList=self.seeds,
                                                    lower=self.lower,
                                                    upper=self.upper,
                                                    replaceValue=1)
                    # shift values so that they're all positive and apply mask
                    maskedA2 = sitk_applyMask(self.img - arr.min(), maskA)
                    # now shift values back, this results in all masked pixels
                    # to be assigned the minimum value
                    maskedA = maskedA2 + arr.min()
                    # use all pixels above minimum value for centroid:
                    centroidsA[index] = (self.xSpace * sitk_centroid(maskedA,
                                            ref=self.ref, threshold=arr.min() + 1))
                    diceA[index] = self.getDice(centroidsA[index], maskA)
                    # all irregular Slices result in DC of -1:
                    diceA[index][np.where(self.niceSlice == False)] = -1
                    # for the final DC score it will look only at the niceSlices:
                    centroidScoreA[index] = np.average(diceA[index,
                                                        self.niceSlice == True])
                    # alternatively we could look at those with DC values > 0:
#                    centroidScoreA[index] = np.average(diceA[index,diceA[index]>-1])
                    # or just take all into account, regardless of their value
#                    centroidScoreA[index] = np.average(diceA[index])


                    print("\nB @ ~{:.4f}%"
                          .format((guess[index] + right[index]) / 2*100))
                    thresholdsB[index] = self.getThresholds(pixelNumber=self.xSize
                        * self.ySize * (guess[index] + right[index]) / 2)
                    maskB = sitk.ConnectedThreshold(image1=self.img,
                                                    seedList=self.seeds,
                                                    lower=self.lower,
                                                    upper=self.upper,
                                                    replaceValue=1)
                    maskedB2 = sitk_applyMask(self.img - arr.min(), maskB)
                    maskedB = maskedB2 + arr.min()
                    centroidsB[index] = (self.xSpace * sitk_centroid(maskedB,
                                            ref=self.ref, threshold=arr.min() + 1))
                    diceB[index] = self.getDice(centroidsB[index], maskB)
                    # all irregular Slices result in DC of -1:
                    diceB[index][np.where(self.niceSlice == False)] = -1
                    # for the final DC score it will look only at the niceSlices:
                    centroidScoreB[index] = np.average(diceB[index,
                                                        self.niceSlice == True])
#                    centroidScoreB[index] = np.average(diceB[index,diceB[index]>-1])
                    # or just take all into account, regardless of their value
#                    centroidScoreB[index] = np.average(diceB[index])

                    if (centroidScoreA[index] < centroidScoreB[index]
                        and index < iterations-1):

                        left[index+1] = guess[index]
                        right[index+1] = right[index]
                        guess[index+1] = (left[index+1] + right[index+1]) / 2
                        direction[index] = 1
                    elif (centroidScoreA[index] > centroidScoreB[index]
                          and index < iterations-1):

                        right[index+1] = guess[index]
                        left[index+1] = left[index]
                        guess[index+1] = (left[index+1] + right[index+1]) / 2
                        direction[index] = -1
                    elif (centroidScoreA[index] == centroidScoreB[index]
                          and index < iterations-1):

                        right[index+1] = (guess[index] + right[index]) / 2
                        left[index+1] = (guess[index] + left[index]) / 2
                        guess[index+1] = guess[index]
```

```
                else:
                    break

                print("--------------------------")
                print("next guess (#{}): ~{:.4f}% \n\n\n\n"
                      .format(index+1, guess[index+1]*100))


            if centroidScoreA.max() > centroidScoreB.max():
                self.centroid = centroidsA[centroidScoreA.argmax()]
                self.lower, self.upper = thresholdsA[centroidScoreA.argmax()]
                self.dice = diceA[centroidScoreA.argmax()]
                self.diceAverage = centroidScoreA.max()
                print("\nmax dice-coefficient obtained "
                      "during iteration #{}: ~{:.4f}"
                      .format(centroidScoreA.argmax(), centroidScoreA.max()))
            elif (centroidScoreA.max() <= centroidScoreB.max()
                  and centroidScoreB.max() != 0):

                self.centroid = centroidsB[centroidScoreB.argmax()]
                self.lower, self.upper = thresholdsB[centroidScoreB.argmax()]
                self.dice = diceB[centroidScoreB.argmax()]
                self.diceAverage = centroidScoreB.max()
                print("\nmax dice-coefficient obtained "
                      "during iteration #{}: ~{:.4f}"
                      .format(centroidScoreB.argmax(), centroidScoreB.max()))
            else:
                return None

            print("\n\n-o-o-o-o-- Summary: --o-o-o-o-\n")
            for index in range(np.size(guess)):
                print("\n  Iteration #{}: range({},{})"
                      .format(index, left[index]*100, right[index]*100))
                if centroidScoreA[index] > centroidScoreB[index]:
                    print("A @ {}%, Score: {}  <---"
                          .format((guess[index] + left[index]) / 2*100,
                                  centroidScoreA[index]))
                    print("B @ {}%, Score: {}"
                          .format((guess[index] + right[index]) / 2*100,
                                  centroidScoreB[index]))
                if centroidScoreA[index] < centroidScoreB[index]:
                    print("A @ {}%, Score: {}"
                          .format((guess[index] + left[index]) / 2*100,
                                  centroidScoreA[index]))
                    print("B @ {}%, Score: {}  <---"
                          .format((guess[index] + right[index]) / 2*100,
                                  centroidScoreB[index]))
                if centroidScoreA[index] == centroidScoreB[index]:
                    print("A @ {}% same as for B @ {}%, Score: = {}"
                          .format((guess[index] + left[index]) / 2*100,
                                  (guess[index] + right[index]) / 2*100,
                                  centroidScoreA[index]))


            if plot == True:
                fig = plt.figure()
                for index in range(iterations):
                    if guess[index] > 0 and centroidScoreA[index] > 0:
                        plt.plot((guess[index] + left[index]) / 2*100,
                                 centroidScoreA[index], 'bo')
                    if guess[index] > 0 and centroidScoreB[index] > 0:
                        plt.plot((guess[index] + right[index]) / 2*100,
                                 centroidScoreB[index], 'rv')
                plt.show()
                if save != False:
                    fig.savefig(str(save) + ".png")


        if percentLimit != "auto" and percentLimit is not False:
            self.centroid = (self.xSpace * sitk_centroid(self.img, ref=self.ref,
                                                         percentLimit=percentLimit))

```

```
        if ((threshold == 'auto' or threshold == 'default')
            and percentLimit is False):

            self.getThresholds(pixelNumber=pixelNumber, scale=scale)
            self.centroid = (self.xSpace * sitk_centroid(self.img, ref=self.ref,
                                                         threshold=self.lower))
        if ((threshold != "auto" and threshold != 'default')
            and threshold is not False and percentLimit is False):

            self.centroid = (self.xSpace * sitk_centroid(self.img, ref=self.ref,
                                                         threshold=threshold))
        for index in range(self.zSize):
            if not self.niceSlice[index]:
                self.centroid[index] = -1, -1
            if self.centroid[index,0] < 0 or self.centroid[index,1] < 0 :
                self.centroid[index] = -1, -1
        print("\n\n")
        return self.centroid


    def showCentroid(self, img=None, com2=0, title=None, pixel=False,
                     interpolation='nearest', ref=None, save=False):
        '''
        shows slice with centroid coordinates

        Parameters
        ----------
        img: SimpleITK.img, optional
            slice of this volume will be shown
            default: self.img
        com2: numpy.ndarray
            supposed to be of same length as img
            will also be shown in plot alongside self.centroid
            helps creating nice plot for comparing COM-shift
        pixel: bool, optional
            if True, changes axis from mm to pixels
        interpolation: "string", optional, default: 'nearest'
            using build-in interpolation of matplotlib.pyplot.imshow
            Acceptable values are 'none', 'nearest', 'bilinear', 'bicubic',
            'spline16', 'spline36', 'hanning', 'hamming', 'hermite', 'kaiser',
            'quadric', 'catrom', 'gaussian', 'bessel', 'mitchell', 'sinc',
            'lanczos'
        ref: int, optional
            slice to be plotted instead of self.ref (default: 0)
        save: string, optional
            save plot as save + ".png"
        '''

        if self.centroid is False:
            print("Volume has no centroid yet. use Volume.getCentroid() first!")
            return None

        if title is None:
            title = self.title
        if ref is None:
            ref = self.ref
        if img is None:
            img = self.img

        if pixel is False:
            extent = (-self.xSpace/2, self.xSize*self.xSpace - self.xSpace/2,
                      self.ySize*self.ySpace - self.ySpace/2, -self.ySpace/2)
            sitk_centroid_show(img=img, com=self.centroid, com2=com2,
                               extent=extent, save=save, title=title,
                               interpolation=interpolation, ref=ref)
        else:
            sitk_centroid_show(img=img, com=self.centroid/self.xSpace,
                               com2=com2/self.xSpace, save=save, title=title,
                               interpolation=interpolation, ref=ref)

    def getMask(self, lower=False, upper=False):

```

```
        if lower is False and self.lower is not False:
            lower = self.lower
        if upper is False and self.upper is not False:
            upper = self.upper

        if lower is False:
            print("Lower threshold missing!")
            return None
        if upper is False:
            print("Upper threshold missing!")
            return None

        self.mask = sitk_getMask(self.img, self.seeds, upper, lower)
        return self.mask

    def applyMask(self, mask=0, replaceArray=False, scale=1000):
        if mask == 0:
            if self.mask:
                mask = self.mask
            else:
                print("Volume has no mask yet. use Volume.getMask() first!")
                return None

        self.masked = sitk_applyMask(self.img, mask, scale=scale,
                                     replaceArray=replaceArray)

        return self.masked

    def showMask(self, interpolation=None, ref=None, save=False, pixel=False):
        if self.mask is False:
            print("Volume has no mask yet. use Volume.getMask() first!")
            return None

        if ref is None:
            ref = self.ref

        if interpolation is None:
            interpolation = 'nearest'

        title = self.title + ", mask"

        extent = None
        if pixel is False:
            extent = (-self.xSpace/2, self.xSize*self.xSpace - self.xSpace/2,
                      self.ySize*self.ySpace - self.ySpace/2, -self.ySpace/2)

        sitk_show(img=self.mask, ref=ref, title=title, extent=extent,
                  interpolation=interpolation, save=save)

    def showMasked(self, interpolation=None, ref=None, save=False, pixel=False):
        if self.masked is False:
            print("Volume has not been masked yet. use Volume.applyMask() first!")
            return None
        if ref is None:
            ref = self.ref

        if interpolation is None:
            interpolation = 'nearest'

        title = self.title + ", masked"

        extent = None
        if pixel is False:
            extent = (-self.xSpace/2, self.xSize*self.xSpace - self.xSpace/2,
                      self.ySize*self.ySpace - self.ySpace/2, -self.ySpace/2)

        sitk_show(img=self.masked, ref=ref, title=title, extent=extent,
                  interpolation=interpolation, save=save)

    def getDice(self, centroid=None, mask=None, iterations=15,
                CT_guess=(3.5,5.5), MR_guess=(1.5,4.5),
                show=False, showAll=False, plot=False, save=False, pixel=False):
```

```
        '''
        Calculates dice coefficient ('DC') and average DC of the volume
        if iterations > 0: varies radius and finds DC with best average DC
        else: if self.raduis == 0: use method to get raduis for DC calculation
        average DC is mean value of all slices, except those with DC of -1

        slice DC is set to -1 if centroid lies outside image or reference
        circle exceeds image

        Parameters
        ----------
        centroid: numpy.ndarray, optional
            centroid to place circles in instead of self.centroid
        mask: SimpleITK image, optional
            binary image to calculate DC of instead of self.mask
        iterations: int, optional
        show: int, optional
            shows circle used to compare mask to in slice nr. "show"
        showAll: bool, optional
            shows all circles tried during iteration
        plot, save: bool, optional
            plot and save iteration

        Returns
        -------
        self.dice: numpy.ndarray
        '''
        if centroid is None:
            centroid = self.centroid
        # to get from mm to pixel coordinates:
        com = centroid / self.xSpace
        if mask is None:
            if self.mask is False:
                self.getMask()
            mask = self.mask

        extent = None
        if pixel is False:
            extent = (-self.xSpace/2, self.xSize*self.xSpace - self.xSpace/2,
                      self.ySize*self.ySpace - self.ySpace/2, -self.ySpace/2)

        if self.radius != 0:
            print("{}_x{}.radius is {} and will therefore be used to calculate DC."
                  .format(self.method, self.resample, self.radius))
            self.dice = sitk_dice_circle(img=mask, centroid=com, extent=extent,
                                         radius=self.radius/self.xSpace, show=show)

        if self.radius == 0 and iterations == 0:
            if self.method == "CT":
                self.dice = sitk_dice_circle(img=mask, centroid=com, extent=extent,
                                             radius=4/self.xSpace, show=show)
            if self.method == "MR":
                self.dice = sitk_dice_circle(img=mask, centroid=com, extent=extent,
                                             radius=2/self.xSpace, show=show)
            if self.method != "CT" and self.method != "MR":
                print("Unknown method!")
                return None

        if self.radius == 0 and iterations > 0:
            low, up = 0, 0
            if self.method == "CT":
                low, up = CT_guess
                radii = np.linspace(low, up, num=iterations) / self.xSpace
            if self.method == "MR":
                low, up = MR_guess
                radii = np.linspace(low, up, num=iterations) / self.xSpace
            if self.method != "CT" and self.method != "MR":
                # radii = np.linspace(1.5, 4.5, num = 11)
                print("Unknown method!")
                return None

            DCs = np.zeros(len(radii))
```

```
            for index, r in enumerate(radii, start=0):
                dice = sitk_dice_circle(img=mask, centroid=com, radius=r,
                                        show=showAll, extent=extent)
#                DCs[index] = np.average(dice)
#                DCs[index] = np.average(dice[dice>-1])
                DCs[index] = np.average(dice[self.niceSlice==True])

            if plot == True:
                fig = plt.figure()
#                plt.ylim(ymin=0.6, ymax=1)
#                plt.xlim(xmin=(low-.1), xmax=(up+.1))
                plt.plot(radii*self.xSpace, DCs, '+-')
                if save is not False:
                    fig.savefig(str(save) + ".png")

            self.dice = sitk_dice_circle(img=mask, centroid=com, show=show,
                                         extent=extent, radius=radii[DCs.argmax()])
            self.bestRadius = radii[DCs.argmax()]*self.xSpace
            print("max dice-coefficient obtained for {} when "
                  "compared to circle with radius = {}"
                  .format(self.method, self.bestRadius))

# Instead of using only DC > 0 for the average:
#        self.diceAverage = np.average(self.dice[self.dice>-1])
# we include the "-1" values in our overall DC average:
        self.diceAverage = np.average(self.dice)
        print("dice-coefficient average for the whole volume is: {:.4f}"
              .format(self.diceAverage))
        return self.dice



def sitk_read(directory, denoise=False):
    '''
    returns DICOM files as "SimpleITK.Image" data type (3D)
    if denoise is True: uses SimpleITK to denoise data
    '''
    reader = sitk.ImageSeriesReader()
    filenames = reader.GetGDCMSeriesFileNames(directory)
    reader.SetFileNames(filenames)
    if denoise:
        print("\n...denoising...")
        imgOriginal = reader.Execute()
        return sitk.CurvatureFlow(image1=imgOriginal,
                                  timeStep=0.125,
                                  numberOfIterations=5)
    else:
        return reader.Execute()


def sitk_write(image, output_dir='', filename='3DImage.mha'):
    '''
    saves image as .mha file
    '''
    output_file_name_3D = os.path.join(output_dir, filename)
    sitk.WriteImage(image, output_file_name_3D)


def sitk_show(img, ref=0, extent=None, title=None, interpolation='nearest', save=False):
    """
    shows plot of img at z=ref
    """
    arr = sitk.GetArrayFromImage(img)
    fig = plt.figure()
    plt.set_cmap("gray")
    if title:
        plt.title(title)

    plt.imshow(arr[ref], extent=extent, interpolation=interpolation)
    plt.show()
    if save != False:
        fig.savefig(str(save) + ".png")
```

```python

def sitk_centroid(img, ref=False, percentLimit=False, threshold=False):
    '''
    returns array with y&x coordinate of centroid for every slice of img
    centroid[slice, y&x-coordinate]
    if no pixel has value > threshold:
        centroid x&y-coordinate of that slice = -1,-1
    '''
    if ((threshold is False and percentLimit is False)
        or (threshold is True and percentLimit is True)):

        print("Please set either percentLimit or threshold!")
        return None

    arr = sitk.GetArrayFromImage(img)
    z, y, x = np.shape(arr)
    # create array with centroid coordinates of rod in each slice
    com = np.zeros((z, 2))

    if ref is False:
        ref = int(z/2)

    if threshold is False:
#        hist, bins = np.histogram(arr[ref, :, :].ravel(), density=True, bins=100)
     #  alternatively, increase number of bins for images with many pixels
        hist, bins = np.histogram(arr[ref, :, :].ravel(), density=True, bins=int(y*x))
        threshold = bins[np.concatenate((np.array([0]), np.cumsum(hist)))
                         * (bins[1] - bins[0]) > percentLimit][0]

    for index in range(z):
        if arr[index].max() > threshold:
            # structuring_element=[[1,1,1],[1,1,1],[1,1,1]]
            segmentation, segments = ndimage.label(arr[index] > threshold)
            # print("segments: {}".format(segments))
            # add ', structuring_element' to label() for recognising
            # diagonal pixels as part of object
            com[index, ::-1] = ndimage.center_of_mass(arr[index, :, :]-threshold,
                                                      segmentation)
            # add ', range(1,segments)' to center_of_mass for list of centroids
            # in each slice (multiple rods!)
        else:
            com[index] = (-1,-1)

    return com

def sitk_centroid_show(img, com, com2=0, extent=None, title=None,
                       save=False, interpolation='nearest', ref=0):

        arr = sitk.GetArrayFromImage(img)
        fig = plt.figure()
        plt.set_cmap("gray")
        if title:
            plt.title(title + ", centroid")
        x = y = 0
        plt.imshow(arr[ref], extent=extent, interpolation=interpolation)
        if type(com2) == np.ndarray:
            x = [com[ref,0],com2[ref,0]]
            y = [com[ref,1],com2[ref,1]]
        else:
            x, y = com[ref]
        plt.scatter(x, y, c=['b','r'])
        plt.show()
        if save != False:
            fig.savefig(str(save) + ".png")

def sitk_coordShift(first, second):
    '''
    returns array with difference of y&x coordinates for every
    centroid[slice, y&x-coordinate]
    '''
    if (np.shape(first) == np.shape(second)
        and np.shape((np.shape(first))) == (2,)):
```

```python

        z, xy = np.shape(first)
        diff = np.zeros((z, 2))
        for slice in range(z):
            if (    first[slice,0]==-1 or first[slice,1]==-1
                    or second[slice,0]==-1 or second[slice,0]==-1):
                diff[slice, 0] = diff[slice, 1] = -1
            else:
                diff[slice, 0] = first[slice, 0] - second[slice, 0]
                diff[slice, 1] = first[slice, 1] - second[slice, 1]
        return diff
    else:
        print("Wrong shape! sitk_coordShift returned 'False'")
        return False


def sitk_coordDist(shift):
    '''
    calculates norm for each entry of array
    returns array with list of calculated values
    '''
    if np.shape(shift)[1] != 2:
        print("shift has wrong shape!")
        return False

    dist = np.zeros((len(shift), 1))
    for slice in range(len(shift)):
        if shift[slice,0] == -1 or shift[slice,1] == -1:
            dist[slice,:] =-1
        else:
            dist[slice, :] = np.linalg.norm(shift[slice, :])
    return dist


def sitk_getMask(img, seedList, upper, lower):
    '''
    creates new SimpleITK.img using a SimpleITK segmentation function
    which is made up by all pixels with values between upper and lower and
    connected to a seed from seedList.
    Returns binary image (SimpleITK.img)
    '''

    if seedList is False:
        print("no seeds given!")
        return None

    return sitk.ConnectedThreshold(image1=img, seedList=seedList,
                                   lower=lower, upper=upper,
                                   replaceValue=1)


def sitk_applyMask(img, mask, replaceArray=False, scale=1000, errorValue=-1):
    '''
    masks img (SimpleITK.Image) using mask (SimpleITK.Image)
    if a replaceArray is given, the values*scale (default scale=1000) of the
    array will be used as pixel intensity for an entire slice each
    '''
    if img.GetSize() != mask.GetSize():
        print(mask.GetSize())
        print(img.GetSize())

        print("mask and image are not the same size!")
        return False

    arr = sitk.GetArrayFromImage(img)
    maskA = sitk.GetArrayFromImage(mask)
    xSize, ySize, zSize = img.GetSize()

    imgMaskedA = (arr - arr.min() + 1)*maskA

    if (np.shape(replaceArray) == (img.GetDepth(), 1)
        or np.shape(replaceArray) == (img.GetDepth(),)):
```

```python

        for slice in range(zSize):
            imgMaskedA[slice][imgMaskedA[slice] != 0] = replaceArray[slice]*scale
            imgMaskedA[slice][imgMaskedA[slice] < 0] = errorValue


    return sitk.GetImageFromArray(imgMaskedA)


def sitk_dice_circle(img, centroid, radius=2.1, show=False, extent=None,
                     interpolation='nearest', save=False):
    """
    Dice coefficient, inspired by
     Medpy (http://pythonhosted.org/MedPy/_modules/medpy/metric/binary.html)

    Computes the Dice coefficient (akas Sorensen index) between a binary
    object in an image and a circle.

    The metric is defined as:

        DC=\frac{2|A\cap B|}{|A|+|B|}

    where A is the first and B the second set of samples (here: binary objects)

    Parameters
    ----------
    input_umg : SimpleITK.Image
        Input data containing objects. Can be any type but will be converted
        into binary: background where 0, object everywhere else.
    centroid : array_like
        array with coordinates for circle centre
    radius : float
        radius for creating reference circles

    Returns
    -------
    DC : array_like
        The Dice coefficient between the object(s) in ```input``` and the
        created circles. It ranges from 0 (no overlap) to 1 (perfect overlap).
        if centroid coordinates + radius would create circle exceeding image
        size: DC of this slice = -1
        Other errors occuring during the calculation should also result in -1
    """

    xSize, ySize, zSize = img.GetSize()
    xSpace, ySpace, zSpace = img.GetSpacing()
    profile = np.zeros((zSize, ySize, xSize), dtype=np.uint8)
    DC = np.zeros((zSize, 1))
    for slice in range(zSize):
        if (centroid[slice,0]+radius < xSize and centroid[slice, 1]+radius < ySize
            and centroid[slice,0]-radius > 0 and centroid[slice, 1]-radius > 0):

            rr, cc = circle(centroid[slice, 0], centroid[slice, 1], radius, (xSize,ySize))
            profile[slice, cc, rr] = 1
        else:
            # print("something's fishy!")
            DC[slice]= -1

    input = sitk.GetArrayFromImage(img)

    input = np.atleast_1d(input.astype(np.bool))
    reference = np.atleast_1d(profile.astype(np.bool))

    intersection = np.zeros((zSize, 1))
    size_input = np.zeros((zSize, 1))
    size_reference = np.zeros((zSize, 1))
    for slice in range(zSize):
        intersection[slice] = np.count_nonzero(input[slice, :, :] & reference[slice, :, :])
        size_input[slice] = np.count_nonzero(input[slice, :, :])
        size_reference[slice] = np.count_nonzero(reference[slice, :, :])

        try:
```

```
            if (DC[slice] == 0) and (float(size_input[slice] + size_reference[slice]) != 0):
                DC[slice] = (2. * intersection[slice] / float(size_input[slice]
                             + size_reference[slice]))

        except ZeroDivisionError:
            DC[slice] = -1

    if show != False:
        profile_img = sitk.GetImageFromArray(profile)
        sitk_centroid_show(profile_img, centroid*xSpace, extent=extent,
                           title="profile, radius: {:03.2f}".format(radius*xSpace),
                           ref=show, save=save)

    return DC


# to view in 3D Slicer, type this in IPython console or in jupyter notebook:
# %env SITK_SHOW_COMMAND /home/david/Downloads/Slicer-4.5.0-1-linux-amd64/Slicer
# sitk.Show(imgFillingCT)
```

```python
# -*- coding: utf-8 -*-
"""
Created on Sun Jul 17 15:17:57 2016
@author: davidblacher
"""

import FunITK as fun
from FunITK import Volume
import datetime
import numpy as np
import matplotlib.pyplot as plt
import os

idxSlice = 200
ph2_CT = Volume(path="../data/phantom2/CT_x1", method="CT",
                 resample=1, ref=idxSlice)
ph2_CT_x4 = Volume(path="../data/phantom2/CT_x4", method="CT",
                    resample=4, ref=idxSlice)
ph2_CT_x9 = Volume(path="../data/phantom2/CT_x9", method="CT",
                    resample=9, ref=idxSlice)
ph2_CT_x25 = Volume(path="../data/phantom2/CT_x25", method="CT",
                     resample=25, ref=idxSlice)
ph2_CT_x100 = Volume(path="../data/phantom2/CT_x100", method="CT",
                      resample=100, ref=idxSlice)

ph2_MR = Volume(path="../data/phantom2/MR_x1", method="MR",
                 resample=1, ref=idxSlice)
ph2_MR_x4 = Volume(path="../data/phantom2/MR_x4", method="MR",
                    resample=4, ref=idxSlice)
ph2_MR_x9 = Volume(path="../data/phantom2/MR_x9", method="MR",
                    resample=9, ref=idxSlice)
ph2_MR_x25 = Volume(path="../data/phantom2/MR_x25", method="MR",
                     resample=25, ref=idxSlice)
ph2_MR_x100 = Volume(path="../data/phantom2/MR_x100", method="MR",
                      resample=100, ref=idxSlice)

vol_list = [[ph2_CT, ph2_CT_x4, ph2_CT_x9, ph2_CT_x25, ph2_CT_x100],
            [ph2_MR, ph2_MR_x4, ph2_MR_x9, ph2_MR_x25, ph2_MR_x100]]
modality, sets = np.shape(vol_list)

length = ph2_CT_x100.zSize
spacing = ph2_CT_x100.zSpace
sliceNumbers = np.arange(length, dtype=int)

# for data centered around iso-centre, this is real x-axis:
iso = 183
dist = ( (sliceNumbers - iso ) ).round(2)

warp_simple = np.zeros((sets, length, 2))
warp_iter = np.zeros((sets, length, 2))
warpMagnitude_simple = np.zeros((sets, length, 1))
warpMagnitude_iter = np.zeros((sets, length, 1))
lows_CT = np.zeros((sets, 2))
radii_CT = np.zeros((sets, 2))
lows_MR = np.zeros((sets, 4))
radii_MR = np.zeros((sets, 4))

# 2 DC for CT, 3 DC for MR (2 using MR.centroid, 1 using CT.centroid!)
DC_CT = np.zeros((sets, length, 2))
DC_CT_average = np.zeros((sets, 2))
DC_MR = np.zeros((sets, length, 4))
DC_MR_average = np.zeros((sets, 4))

for i in range(sets):
    vol_list[0][i].getCentroid()
    CT_DC_simple = vol_list[0][i].getDice()
    CT_DC_simple_average = vol_list[0][i].diceAverage
    CT_lower_simple = vol_list[0][i].lower
    CT_radius_simple = vol_list[0][i].bestRadius

    vol_list[1][i].getCentroid()
    MR_DC_simple = vol_list[1][i].getDice()
```

```
    MR_DC_simple_average = vol_list[1][i].diceAverage
    MR_lower_simple = vol_list[1][i].lower
    MR_radius_simple = vol_list[1][i].bestRadius
    vol_list[1][i].getMask()
    MR_DC_simple_CT_COM = vol_list[1][i].getDice(centroid=vol_list[0][i].centroid)
    MR_DC_simple_CT_COM_average = vol_list[1][i].diceAverage
    MR_lower_simple_CT_COM = vol_list[1][i].lower
    MR_radius_simple_CT_COM = vol_list[1][i].bestRadius
    # this calculates the coordinate difference of MR.centroid relative to CT.centroid
    warp_simple[i] = fun.sitk_coordShift(vol_list[0][i].centroid, vol_list[1][i].centroid)
    # this calculates the norm (=absolute distance) between the centroids in each slice
    warpMagnitude_simple[i] = fun.sitk_coordDist(warp_simple[i])


    vol_list[0][i].getCentroid(percentLimit='auto', iterations=5, top=0.20)
    CT_DC_iter = vol_list[0][i].dice
    CT_DC_iter_average = vol_list[0][i].diceAverage
    CT_lower_iter = vol_list[0][i].lower
    CT_radius_iter = vol_list[0][i].bestRadius

    vol_list[1][i].getCentroid(percentLimit='auto', iterations=6, top=0.20)
    MR_DC_iter = vol_list[1][i].dice
    MR_DC_iter_average = vol_list[1][i].diceAverage
    MR_lower_iter = vol_list[1][i].lower
    MR_radius_iter = vol_list[1][i].bestRadius
    vol_list[1][i].getMask()
    MR_DC_iter_CT_COM = vol_list[1][i].getDice(centroid=vol_list[0][i].centroid)
    MR_DC_iter_CT_COM_average = vol_list[1][i].diceAverage
    MR_lower_iter_CT_COM = vol_list[1][i].lower
    MR_radius_iter_CT_COM = vol_list[1][i].bestRadius

    # this calculates the coordinate difference of MR.centroid relative to CT.centroid
    warp_iter[i] = fun.sitk_coordShift(vol_list[0][i].centroid, vol_list[1][i].centroid)
    # this calculates the norm (=absolute distance) between the centroids in each slice
    warpMagnitude_iter[i] = fun.sitk_coordDist(warp_iter[i])


    DC_CT[i] = np.column_stack((CT_DC_simple,CT_DC_iter))
    DC_CT_average[i] = CT_DC_simple_average,CT_DC_iter_average

    DC_MR[i] = np.column_stack((MR_DC_simple, MR_DC_iter,
                                MR_DC_simple_CT_COM, MR_DC_iter_CT_COM))
    DC_MR_average[i] = (MR_DC_simple_average, MR_DC_iter_average,
                        MR_DC_simple_CT_COM_average, MR_DC_iter_CT_COM_average)
    lows_CT[i] = CT_lower_simple, CT_lower_iter
    lows_MR[i] = MR_lower_simple, MR_lower_iter, MR_lower_simple_CT_COM, MR_lower_iter_CT_COM
    radii_CT[i] = CT_radius_simple, CT_radius_iter
    radii_MR[i] = MR_radius_simple, MR_radius_iter, MR_radius_simple_CT_COM, MR_radius_iter_CT_COM


now = datetime.datetime.now()

COLUMNS  = ('sliceNo dist warp_x  warp_y  warpMagnitude  DC_CT  DC_MR '
            ' DC_MR_CT-COM  warp_x*  warp_y*  warpMagnitude*   DC_CT*'
            ' DC_MR*  DC_MR_CT-COM*')
for i in range(sets):
    DATA = np.column_stack((sliceNumbers.astype(str),
                            dist.astype(str),

                            warp_simple[i].round(4).astype(str),
                            warpMagnitude_simple[i].round(4).astype(str),
                            DC_CT[i,:,0].round(4).astype(str),
                            DC_MR[i,:,0].round(4).astype(str),
                            DC_MR[i,:,2].round(4).astype(str),

                            warp_iter[i].round(4).astype(str),
                            warpMagnitude_iter[i].round(4).astype(str),
                            DC_CT[i,:,1].round(4).astype(str),
                            DC_MR[i,:,1].round(4).astype(str),
                            DC_MR[i,:,3].round(4).astype(str)))
 #   text = np.row_stack((NAMES, DATA))
```

```
    head0 = ("{}_x{}\n path: {}\n thresholds:\n lower (simple): {},\n"
    " lower (iter): {}\n upper: {}\n DC-average (simple): {} (bestRadius: {})\n"
    " DC-average (iter): {} (bestRadius: {})\n".format(vol_list[0][i].method,
    vol_list[0][i].resample, vol_list[0][i].path, lows_CT[i][0], lows_CT[i][1],
    vol_list[0][i].upper, DC_CT_average[i][0], radii_CT[i][0],
    DC_CT_average[i][1], radii_CT[i][1]))

    head1 = ("{}_x{}\n path: {}\n thresholds:\n lower (simple): {},\n"
    " lower (iter): {}\n lower (simple_CT-COM): {}\n lower (iter_CT-COM): {}\n"
    " upper: {}\n DC-average (simple): {} (bestRadius: {})\n DC-average (iter): {}"
    " (bestRadius: {})\n DC-average (CT-COM, simple): {} (bestRadius: {})\n"
    " DC-average (CT-COM, iter): {} (bestRadius: {})".format(vol_list[1][i].method,
     vol_list[1][i].resample, vol_list[1][i].path, lows_MR[i][0], lows_MR[i][1],
     lows_MR[i][2], lows_MR[i][3], vol_list[1][i].upper, DC_MR_average[i][0],
     radii_MR[i][0], DC_MR_average[i][1], radii_MR[i][1], DC_MR_average[i][2],
     radii_MR[i][2], DC_MR_average[i][3], radii_MR[i][3]))

    head = str(now) + '\n'+ head0 + head1 + '\n' + COLUMNS
    np.savetxt('../data/output_txt/phantom2_out_txt/CT-MR_x{}_{}_{}.txt'
               .format(vol_list[0][i].resample, now.date(), now.time()), DATA,
               delimiter="   &  ", header=head, comments="# ", fmt='%3s')


for i in range(sets):

# creates CT.masked using CT.mask,
# but assigns each slice the centroid distance*1000*spacing as pixel value
    vol_list[0][0].applyMask(replaceArray=warp_simple[i][:,0])
# exports 3D image as .mha file
    fun.sitk_write(vol_list[0][0].masked, "../data/output_img/ph2_out_img/mha_files",
                   "ph2_out_x{}_warpX_simple.mha".format(vol_list[0][i].resample))
    vol_list[0][0].applyMask(replaceArray=warp_simple[i][:,1])
    fun.sitk_write(vol_list[0][0].masked, "../data/output_img/ph2_out_img/mha_files",
                   "ph2_out_x{}_warpY_simple.mha".format(vol_list[0][i].resample))

    vol_list[0][0].applyMask(replaceArray=warp_iter[i][:,0])
    fun.sitk_write(vol_list[0][0].masked, "../data/output_img/ph2_out_img/mha_files",
                   "ph2_out_x{}_warpX_iter.mha".format(vol_list[0][i].resample))
    vol_list[0][0].applyMask(replaceArray=warp_iter[i][:,1])
    fun.sitk_write(vol_list[0][0].masked, "../data/output_img/ph2_out_img/mha_files",
                   "ph2_out_x{}_warpY_iter.mha".format(vol_list[0][i].resample))

    vol_list[0][0].applyMask(replaceArray=warpMagnitude_simple[i])
    fun.sitk_write(vol_list[0][0].masked, "../data/output_img/ph2_out_img/mha_files",
                   "ph2_out_x{}_warpMagnitude_simple.mha".format(vol_list[0][i].resample))

    vol_list[0][0].applyMask(replaceArray=warpMagnitude_iter[i])
    fun.sitk_write(vol_list[0][0].masked, "../data/output_img/ph2_out_img/mha_files",
                   "ph2_out_x{}_warpMagnitude_iter.mha".format(vol_list[0][i].resample))

    vol_list[0][0].applyMask(replaceArray=DC_MR[i,:,0])
    fun.sitk_write(vol_list[0][0].masked, "../data/output_img/ph2_out_img/mha_files",
                   "ph2_out_x{}_DC_MR_simple.mha".format(vol_list[0][i].resample))

    vol_list[0][0].applyMask(replaceArray=DC_MR[i,:,2])
    fun.sitk_write(vol_list[0][0].masked, "../data/output_img/ph2_out_img/mha_files",
                   "ph2_out_x{}_DC_MR_CT-COM_simple.mha".format(vol_list[0][i].resample))


    vol_list[0][0].applyMask(replaceArray=DC_MR[i,:,1])
    fun.sitk_write(vol_list[0][0].masked, "../data/output_img/ph2_out_img/mha_files",
                   "ph2_out_x{}_DC_MR_iter.mha".format(vol_list[0][i].resample))

    vol_list[0][0].applyMask(replaceArray=DC_MR[i,:,3])
    fun.sitk_write(vol_list[0][0].masked, "../data/output_img/ph2_out_img/mha_files",
                   "ph2_out_x{}_DC_MR_CT-COM_iter.mha".format(vol_list[0][i].resample))


# instead of opening the created file manually, you can use this lines in
# the IPython console to start 3D Slicer and open it automatically:
# %env SITK_SHOW_COMMAND /home/david/Downloads/Slicer-4.5.0-1-linux-amd64/Slicer
# sitk.Show(CT.masked)
```

```python
# -*- coding: utf-8 -*-
"""
Created on Tue Jul 25 16:22:50 2017
@author: davidblacher
"""

import FunITK as fun
from FunITK import Volume
import datetime
import numpy as np
import matplotlib.pyplot as plt
import os

idxSlice = 130
ph3_CT = Volume(path="../data/phantom3_MR_v2/ph3_CT_x1", method="CT",
                 resample=1, ref=idxSlice)
ph3_CT_x4 = Volume(path="../data/phantom3_MR_v2/ph3_CT_x4", method="CT",
                    resample=4, ref=idxSlice)
ph3_CT_x9 = Volume(path="../data/phantom3_MR_v2/ph3_CT_x9", method="CT",
                    resample=9, ref=idxSlice)
ph3_CT_x25 = Volume(path="../data/phantom3_MR_v2/ph3_CT_x25", method="CT",
                     resample=25, ref=idxSlice)
ph3_CT_x100 = Volume(path="../data/phantom3_MR_v2/ph3_CT_x100", method="CT",
                      resample=100, ref=idxSlice)

ph3_MR_v2 = Volume(path="../data/phantom3_MR_v2/ph3_MR_v2_x1", method="MR",
                    resample=1, ref=idxSlice)
ph3_MR_v2_x4 = Volume(path="../data/phantom3_MR_v2/ph3_MR_v2_x4", method="MR",
                      resample=4, ref=idxSlice)
ph3_MR_v2_x9 = Volume(path="../data/phantom3_MR_v2/ph3_MR_v2_x9", method="MR",
                      resample=9, ref=idxSlice)
ph3_MR_v2_x25 = Volume(path="../data/phantom3_MR_v2/ph3_MR_v2_x25", method="MR",
                       resample=25, ref=idxSlice)
ph3_MR_v2_x100 = Volume(path="../data/phantom3_MR_v2/ph3_MR_v2_x100", method="MR",
                        resample=100, ref=idxSlice)

vol_list = [[ph3_CT, ph3_CT_x4, ph3_CT_x9, ph3_CT_x25, ph3_CT_x100,],
            [ph3_MR_v2, ph3_MR_v2_x4, ph3_MR_v2_x9, ph3_MR_v2_x25, ph3_MR_v2_x100]]
modality, sets = np.shape(vol_list)

length = ph3_CT_x100.zSize
lspacing = ph3_CT_x100.zSpace
sliceNumbers = np.arange(length, dtype=int)

# for data centered around iso-centre, this is real x-axis:
iso = 361
dist = ( (sliceNumbers - iso ) ).round(2)

warp_simple = np.zeros((sets, length, 2))
warp_iter = np.zeros((sets, length, 2))
warpMagnitude_simple = np.zeros((sets, length, 1))
warpMagnitude_iter = np.zeros((sets, length, 1))
lows_CT = np.zeros((sets, 2))
radii_CT = np.zeros((sets, 2))
lows_MR = np.zeros((sets, 4))
radii_MR = np.zeros((sets, 4))

# 2 DC for CT, 3 DC for MR (2 using MR.centroid, 1 using CT.centroid!)
DC_CT = np.zeros((sets, length, 2))
DC_CT_average = np.zeros((sets, 2))
DC_MR = np.zeros((sets, length, 4))
DC_MR_average = np.zeros((sets, 4))

for i in range(sets):
    vol_list[0][i].getCentroid()
    CT_DC_simple = vol_list[0][i].getDice()
    CT_DC_simple_average = vol_list[0][i].diceAverage
    CT_lower_simple = vol_list[0][i].lower
    CT_radius_simple = vol_list[0][i].bestRadius

    vol_list[1][i].getCentroid()
    MR_DC_simple = vol_list[1][i].getDice()
```

```
    MR_DC_simple_average = vol_list[1][i].diceAverage
    MR_lower_simple = vol_list[1][i].lower
    MR_radius_simple = vol_list[1][i].bestRadius
    vol_list[1][i].getMask()
    MR_DC_simple_CT_COM = vol_list[1][i].getDice(centroid=vol_list[0][i].centroid)
    MR_DC_simple_CT_COM_average = vol_list[1][i].diceAverage
    MR_lower_simple_CT_COM = vol_list[1][i].lower
    MR_radius_simple_CT_COM = vol_list[1][i].bestRadius
    # this calculates the coordinate difference of MR.centroid relative to CT.centroid
    warp_simple[i] = fun.sitk_coordShift(vol_list[0][i].centroid, vol_list[1][i].centroid)
    # this calculates the norm (=absolute distance) between the centroids in each slice
    warpMagnitude_simple[i] = fun.sitk_coordDist(warp_simple[i])


    vol_list[0][i].getCentroid(percentLimit='auto', iterations=5, top=0.20)
    CT_DC_iter = vol_list[0][i].dice
    CT_DC_iter_average = vol_list[0][i].diceAverage
    CT_lower_iter = vol_list[0][i].lower
    CT_radius_iter = vol_list[0][i].bestRadius

    vol_list[1][i].getCentroid(percentLimit='auto', iterations=6, top=0.20)
    MR_DC_iter = vol_list[1][i].dice
    MR_DC_iter_average = vol_list[1][i].diceAverage
    MR_lower_iter = vol_list[1][i].lower
    MR_radius_iter = vol_list[1][i].bestRadius
    vol_list[1][i].getMask()
    MR_DC_iter_CT_COM = vol_list[1][i].getDice(centroid=vol_list[0][i].centroid)
    MR_DC_iter_CT_COM_average = vol_list[1][i].diceAverage
    MR_lower_iter_CT_COM = vol_list[1][i].lower
    MR_radius_iter_CT_COM = vol_list[1][i].bestRadius

    # this calculates the coordinate difference of MR.centroid relative to CT.centroid
    warp_iter[i] = fun.sitk_coordShift(vol_list[0][i].centroid, vol_list[1][i].centroid)
    # this calculates the norm (=absolute distance) between the centroids in each slice
    warpMagnitude_iter[i] = fun.sitk_coordDist(warp_iter[i])


    DC_CT[i] = np.column_stack((CT_DC_simple,CT_DC_iter))
    DC_CT_average[i] = CT_DC_simple_average,CT_DC_iter_average

    DC_MR[i] = np.column_stack((MR_DC_simple, MR_DC_iter,
                                MR_DC_simple_CT_COM, MR_DC_iter_CT_COM))
    DC_MR_average[i] = (MR_DC_simple_average, MR_DC_iter_average,
                        MR_DC_simple_CT_COM_average, MR_DC_iter_CT_COM_average)
    lows_CT[i] = CT_lower_simple, CT_lower_iter
    lows_MR[i] = MR_lower_simple, MR_lower_iter, MR_lower_simple_CT_COM, MR_lower_iter_CT_COM
    radii_CT[i] = CT_radius_simple, CT_radius_iter
    radii_MR[i] = MR_radius_simple, MR_radius_iter, MR_radius_simple_CT_COM, MR_radius_iter_CT_COM


now = datetime.datetime.now()

COLUMNS  = ('sliceNo dist warp_x  warp_y  warpMagnitude  DC_CT  DC_MR '
            ' DC_MR_CT-COM  warp_x*  warp_y*  warpMagnitude*   DC_CT*'
            ' DC_MR*  DC_MR_CT-COM*')
for i in range(sets):
    DATA = np.column_stack((sliceNumbers.astype(str),
                            dist.astype(str),

                            warp_simple[i].round(4).astype(str),
                            warpMagnitude_simple[i].round(4).astype(str),
                            DC_CT[i,:,0].round(4).astype(str),
                            DC_MR[i,:,0].round(4).astype(str),
                            DC_MR[i,:,2].round(4).astype(str),

                            warp_iter[i].round(4).astype(str),
                            warpMagnitude_iter[i].round(4).astype(str),
                            DC_CT[i,:,1].round(4).astype(str),
                            DC_MR[i,:,1].round(4).astype(str),
                            DC_MR[i,:,3].round(4).astype(str)))
 #   text = np.row_stack((NAMES, DATA))
```

```
    head0 = ("{}_x{}\n path: {}\n thresholds:\n lower (simple): {},\n"
    " lower (iter): {}\n upper: {}\n DC-average (simple): {} (bestRadius: {})\n"
    " DC-average (iter): {} (bestRadius: {})\n".format(vol_list[0][i].method,
    vol_list[0][i].resample, vol_list[0][i].path, lows_CT[i][0], lows_CT[i][1],
    vol_list[0][i].upper, DC_CT_average[i][0], radii_CT[i][0],
    DC_CT_average[i][1], radii_CT[i][1]))

    head1 = ("{}_x{}\n path: {}\n thresholds:\n lower (simple): {},\n"
    " lower (iter): {}\n lower (simple_CT-COM): {}\n lower (iter_CT-COM): {}\n"
    " upper: {}\n DC-average (simple): {} (bestRadius: {})\n DC-average (iter): {}"
    " (bestRadius: {})\n DC-average (CT-COM, simple): {} (bestRadius: {})\n"
    " DC-average (CT-COM, iter): {} (bestRadius: {})".format(vol_list[1][i].method,
     vol_list[1][i].resample, vol_list[1][i].path, lows_MR[i][0], lows_MR[i][1],
     lows_MR[i][2], lows_MR[i][3], vol_list[1][i].upper, DC_MR_average[i][0],
     radii_MR[i][0], DC_MR_average[i][1], radii_MR[i][1], DC_MR_average[i][2],
     radii_MR[i][2], DC_MR_average[i][3], radii_MR[i][3]))

    head = str(now) + '\n'+ head0 + head1 + '\n' + COLUMNS
    np.savetxt('../data/output_txt/phantom3_out_txt/CT-MR_v2_x{}_{}_{}.txt'
               .format(vol_list[0][i].resample, now.date(), now.time()), DATA,
               delimiter="   &  ", header=head, comments="# ", fmt='%3s')


for i in range(sets):

# creates CT.masked using CT.mask,
# but assigns each slice the centroid distance*1000*spacing as pixel value
    vol_list[0][0].applyMask(replaceArray=warp_simple[i][:,0])
# exports 3D image as .mha file
    fun.sitk_write(vol_list[0][0].masked, "../data/output_img/ph3_MR_v2_out_img/mha_files",
                   "ph3_v2_out_x{}_warpX_simple.mha".format(vol_list[0][i].resample))
    vol_list[0][0].applyMask(replaceArray=warp_simple[i][:,1])
    fun.sitk_write(vol_list[0][0].masked, "../data/output_img/ph3_MR_v2_out_img/mha_files",
                   "ph3_v2_out_x{}_warpY_simple.mha".format(vol_list[0][i].resample))

    vol_list[0][0].applyMask(replaceArray=warp_iter[i][:,0])
    fun.sitk_write(vol_list[0][0].masked, "../data/output_img/ph3_MR_v2_out_img/mha_files",
                   "ph3_v2_out_x{}_warpX_iter.mha".format(vol_list[0][i].resample))
    vol_list[0][0].applyMask(replaceArray=warp_iter[i][:,1])
    fun.sitk_write(vol_list[0][0].masked, "../data/output_img/ph3_MR_v2_out_img/mha_files",
                   "ph3_v2_out_x{}_warpY_iter.mha".format(vol_list[0][i].resample))

    vol_list[0][0].applyMask(replaceArray=warpMagnitude_simple[i])
    fun.sitk_write(vol_list[0][0].masked, "../data/output_img/ph3_MR_v2_out_img/mha_files",
                   "ph3_v2_out_x{}_warpMagnitude_simple.mha".format(vol_list[0][i].resample))

    vol_list[0][0].applyMask(replaceArray=warpMagnitude_iter[i])
    fun.sitk_write(vol_list[0][0].masked, "../data/output_img/ph3_MR_v2_out_img/mha_files",
                   "ph3_v2_out_x{}_warpMagnitude_iter.mha".format(vol_list[0][i].resample))

    vol_list[0][0].applyMask(replaceArray=DC_MR[i,:,0])
    fun.sitk_write(vol_list[0][0].masked, "../data/output_img/ph3_MR_v2_out_img/mha_files",
                   "ph3_v2_out_x{}_DC_MR_simple.mha".format(vol_list[0][i].resample))

    vol_list[0][0].applyMask(replaceArray=DC_MR[i,:,2])
    fun.sitk_write(vol_list[0][0].masked, "../data/output_img/ph3_MR_v2_out_img/mha_files",
                   "ph3_v2_out_x{}_DC_MR_CT-COM_simple.mha".format(vol_list[0][i].resample))


    vol_list[0][0].applyMask(replaceArray=DC_MR[i,:,1])
    fun.sitk_write(vol_list[0][0].masked, "../data/output_img/ph3_MR_v2_out_img/mha_files",
                   "ph3_v2_out_x{}_DC_MR_iter.mha".format(vol_list[0][i].resample))

    vol_list[0][0].applyMask(replaceArray=DC_MR[i,:,3])
    fun.sitk_write(vol_list[0][0].masked, "../data/output_img/ph3_MR_v2_out_img/mha_files",
                   "ph3_v2_out_x{}_DC_MR_CT-COM_iter.mha".format(vol_list[0][i].resample))


# instead of opening the created file manually, you can use this lines in
# the IPython console to start 3D Slicer and open it automatically:
# %env SITK_SHOW_COMMAND /home/david/Downloads/Slicer-4.5.0-1-linux-amd64/Slicer
# sitk.Show(CT.masked)
```

2017-08-30 14:35:38.568013
CT_x100
path: ../data/phantom2/CT_x100
thresholds:
lower (simple): -460.940227704,
lower (iter): -188.922061524
upper: 329.0
DC-average (simple): 0.87667059562 (bestRadius: 3.92857142857)
DC-average (iter): 0.992641237859 (bestRadius: 3.64285714286)
MR_x100
path: ../data/phantom2/MR_x100
thresholds:
lower (simple): 827.518223235,
lower (iter): 339.31049961
lower (simple_CT-COM): 827.518223235
lower (iter_CT-COM): 339.31049961
upper: 2095.0
DC-average (simple): 0.893466506573 (bestRadius: 1.92857142857)
DC-average (iter): 0.941615552809 (bestRadius: 3.0)
DC-average (CT-COM, simple): 0.685706722238 (bestRadius: 1.92857142857)
DC-average (CT-COM, iter): 0.755705950588 (bestRadius: 2.78571428571)

sliceNo	dist	warpX	warpY	warpM	DC_{CT}	DC_{MR}	$DC_{MR(CT)}$	warpX*	warpY*	warpM*	DC^*_{CT}	DC^*_{MR}	$DC^*_{MR(CT)}$
0	-183	-0.105	-1.115	1.120	0.983	0.844	0.610	-0.105	-1.057	1.062	0.995	0.950	0.755
1	-182	-0.121	-1.113	1.120	0.981	0.855	0.609	-0.120	-1.053	1.060	0.993	0.950	0.757
2	-181	-0.128	-1.111	1.118	0.982	0.867	0.614	-0.133	-1.047	1.056	0.993	0.950	0.758
3	-180	-0.146	-1.099	1.108	0.983	0.873	0.617	-0.152	-1.033	1.044	0.994	0.951	0.762
4	-179	-0.148	-1.065	1.075	0.982	0.886	0.634	-0.157	-1.001	1.013	0.995	0.949	0.770
5	-178	-0.165	-1.067	1.079	0.982	0.896	0.634	-0.165	-1.000	1.013	0.993	0.947	0.772
6	-177	-0.156	-1.042	1.054	0.981	0.905	0.644	-0.166	-0.984	0.998	0.994	0.947	0.776
7	-176	-0.181	-1.031	1.047	0.981	0.904	0.649	-0.184	-0.979	0.996	0.993	0.946	0.776
8	-175	-0.168	-1.029	1.043	0.982	0.907	0.650	-0.177	-0.980	0.996	0.993	0.945	0.776
9	-174	-0.184	-1.046	1.062	0.982	0.909	0.645	-0.191	-0.991	1.009	0.994	0.944	0.774
10	-173	-0.191	-1.021	1.039	0.983	0.914	0.650	-0.200	-0.978	0.998	0.993	0.944	0.776
11	-172	-0.183	-1.033	1.049	0.981	0.918	0.651	-0.198	-0.992	1.012	0.994	0.943	0.773
12	-171	-0.189	-1.027	1.044	0.982	0.921	0.651	-0.202	-0.993	1.013	0.996	0.942	0.771
13	-170	-0.204	-1.026	1.046	0.984	0.923	0.655	-0.215	-0.993	1.016	0.995	0.941	0.772
14	-169	-0.211	-1.023	1.045	0.982	0.926	0.658	-0.225	-1.001	1.026	0.991	0.942	0.770
15	-168	-0.210	-1.020	1.041	0.984	0.928	0.658	-0.219	-0.996	1.019	0.993	0.942	0.772
16	-167	-0.205	-1.007	1.028	0.983	0.929	0.663	-0.214	-0.991	1.013	0.995	0.942	0.772
17	-166	-0.210	-0.983	1.005	0.983	0.930	0.667	-0.217	-0.973	0.997	0.995	0.942	0.777
18	-165	-0.212	-0.985	1.007	0.983	0.933	0.668	-0.221	-0.979	1.003	0.994	0.942	0.775
19	-164	-0.217	-0.984	1.008	0.983	0.935	0.669	-0.226	-0.980	1.006	0.993	0.943	0.775
20	-163	-0.212	-0.980	1.002	0.982	0.934	0.676	-0.214	-0.972	0.995	0.993	0.944	0.777
21	-162	-0.188	-0.989	1.007	0.981	0.937	0.674	-0.191	-0.981	0.999	0.993	0.944	0.778
22	-161	-0.185	-0.983	1.000	0.982	0.942	0.673	-0.182	-0.978	0.994	0.993	0.947	0.780
23	-160	-0.190	-0.971	0.989	0.982	0.948	0.681	-0.182	-0.964	0.981	0.995	0.948	0.783
24	-159	-0.179	-0.955	0.971	0.982	0.948	0.690	-0.172	-0.947	0.962	0.992	0.948	0.789
25	-158	-0.173	-0.949	0.965	0.981	0.949	0.691	-0.157	-0.942	0.955	0.993	0.950	0.792
26	-157	-0.153	-0.934	0.946	0.983	0.950	0.699	-0.135	-0.934	0.944	0.991	0.951	0.795
27	-156	-0.148	-0.907	0.919	0.983	0.952	0.705	-0.129	-0.904	0.914	0.991	0.951	0.801
28	-155	-0.138	-0.913	0.923	0.982	0.949	0.706	-0.121	-0.906	0.915	0.992	0.952	0.801
29	-154	-0.122	-0.920	0.928	0.981	0.955	0.702	-0.105	-0.917	0.923	0.994	0.951	0.798
30	-153	-0.123	-0.906	0.914	0.981	0.956	0.704	-0.108	-0.907	0.914	0.994	0.951	0.801
31	-152	-0.120	-0.907	0.915	0.981	0.957	0.704	-0.098	-0.901	0.907	0.994	0.952	0.803
32	-151	-0.112	-0.899	0.906	0.981	0.959	0.709	-0.093	-0.886	0.891	0.992	0.951	0.807
33	-150	-0.097	-0.885	0.890	0.980	0.957	0.714	-0.080	-0.878	0.882	0.993	0.950	0.808
34	-149	-0.110	-0.880	0.887	0.981	0.959	0.717	-0.087	-0.867	0.871	0.995	0.950	0.812
35	-148	-0.112	-0.869	0.876	0.982	0.959	0.719	-0.091	-0.858	0.863	0.995	0.949	0.813
36	-147	-0.097	-0.855	0.860	0.983	0.956	0.727	-0.085	-0.847	0.851	0.994	0.949	0.817

37	-146	-0.097	-0.838	0.843	0.983	0.958	0.732	-0.087	-0.832	0.837	0.995	0.948	0.820
38	-145	-0.090	-0.811	0.816	0.983	0.958	0.738	-0.081	-0.809	0.813	0.995	0.948	0.825
39	-144	-0.101	-0.804	0.810	0.981	0.958	0.742	-0.086	-0.799	0.804	0.992	0.947	0.827
40	-143	-0.110	-0.804	0.811	0.981	0.960	0.740	-0.098	-0.798	0.804	0.994	0.947	0.828
41	-142	-0.104	-0.813	0.820	0.982	0.961	0.736	-0.094	-0.805	0.810	0.994	0.947	0.827
42	-141	-0.092	-0.798	0.804	0.982	0.963	0.739	-0.086	-0.795	0.799	0.991	0.947	0.829
43	-140	-0.104	-0.786	0.792	0.982	0.960	0.744	-0.091	-0.785	0.791	0.993	0.947	0.830
44	-139	-0.097	-0.779	0.785	0.983	0.961	0.748	-0.081	-0.785	0.789	0.992	0.947	0.829
45	-138	-0.088	-0.746	0.751	0.982	0.962	0.760	-0.077	-0.754	0.758	0.993	0.947	0.834
46	-137	-0.093	-0.761	0.767	0.982	0.962	0.754	-0.081	-0.765	0.769	0.993	0.947	0.833
47	-136	-0.081	-0.759	0.764	0.981	0.963	0.754	-0.070	-0.761	0.764	0.994	0.948	0.834
48	-135	-0.080	-0.758	0.763	0.983	0.962	0.753	-0.067	-0.764	0.767	0.995	0.949	0.833
49	-134	-0.076	-0.744	0.748	0.982	0.966	0.758	-0.066	-0.756	0.759	0.995	0.950	0.834
50	-133	-0.081	-0.740	0.745	0.983	0.967	0.759	-0.076	-0.759	0.763	0.995	0.951	0.832
51	-132	-0.076	-0.731	0.735	0.982	0.969	0.763	-0.070	-0.751	0.754	0.994	0.951	0.834
52	-131	-0.061	-0.723	0.725	0.982	0.969	0.766	-0.054	-0.747	0.749	0.994	0.952	0.834
53	-130	-0.076	-0.723	0.727	0.983	0.967	0.767	-0.067	-0.743	0.746	0.995	0.952	0.836
54	-129	-0.075	-0.723	0.727	0.985	0.966	0.767	-0.064	-0.745	0.748	0.994	0.953	0.835
55	-128	-0.072	-0.704	0.707	0.981	0.962	0.774	-0.072	-0.730	0.734	0.993	0.955	0.837
56	-127	-0.089	-0.721	0.727	0.983	0.959	0.771	-0.090	-0.744	0.750	0.993	0.956	0.833
57	-126	-0.080	-0.695	0.699	0.984	0.959	0.778	-0.075	-0.724	0.728	0.994	0.957	0.838
58	-125	-0.077	-0.687	0.691	0.985	0.960	0.779	-0.076	-0.719	0.723	0.995	0.958	0.839
59	-124	-0.077	-0.694	0.698	0.984	0.958	0.780	-0.077	-0.722	0.726	0.994	0.959	0.838
60	-123	-0.070	-0.693	0.696	0.984	0.958	0.780	-0.069	-0.720	0.723	0.992	0.958	0.840
61	-122	-0.078	-0.691	0.696	0.983	0.957	0.781	-0.084	-0.713	0.718	0.992	0.959	0.841
62	-121	-0.088	-0.687	0.692	0.983	0.960	0.784	-0.091	-0.703	0.709	0.990	0.958	0.843
63	-120	-0.085	-0.667	0.672	0.982	0.963	0.788	-0.080	-0.684	0.689	0.993	0.957	0.847
64	-119	-0.075	-0.651	0.655	0.983	0.963	0.793	-0.078	-0.668	0.672	0.993	0.956	0.851
65	-118	-0.074	-0.639	0.643	0.983	0.967	0.796	-0.071	-0.660	0.664	0.994	0.955	0.852
66	-117	-0.072	-0.645	0.649	0.982	0.967	0.794	-0.071	-0.657	0.661	0.992	0.955	0.852
67	-116	-0.057	-0.657	0.659	0.982	0.967	0.789	-0.058	-0.671	0.673	0.993	0.956	0.848
68	-115	-0.064	-0.650	0.653	0.981	0.965	0.790	-0.063	-0.666	0.669	0.994	0.957	0.850
69	-114	-0.069	-0.640	0.643	0.982	0.964	0.793	-0.073	-0.659	0.663	0.993	0.957	0.851
70	-113	-0.063	-0.630	0.633	0.981	0.966	0.795	-0.067	-0.650	0.654	0.992	0.957	0.853
71	-112	-0.070	-0.618	0.622	0.983	0.968	0.801	-0.071	-0.634	0.638	0.993	0.957	0.858
72	-111	-0.068	-0.627	0.631	0.981	0.967	0.797	-0.068	-0.636	0.640	0.993	0.957	0.856
73	-110	-0.063	-0.622	0.626	0.983	0.966	0.800	-0.074	-0.630	0.634	0.992	0.955	0.859
74	-109	-0.068	-0.611	0.615	0.981	0.966	0.803	-0.079	-0.619	0.624	0.992	0.954	0.861
75	-108	-0.074	-0.608	0.613	0.982	0.965	0.807	-0.085	-0.614	0.620	0.993	0.952	0.863
76	-107	-0.078	-0.594	0.599	0.982	0.965	0.812	-0.082	-0.602	0.608	0.993	0.951	0.864
77	-106	-0.074	-0.584	0.589	0.981	0.964	0.817	-0.086	-0.590	0.597	0.993	0.951	0.866
78	-105	-0.076	-0.602	0.607	0.982	0.965	0.808	-0.086	-0.603	0.609	0.995	0.949	0.865
79	-104	-0.083	-0.610	0.615	0.983	0.964	0.806	-0.090	-0.605	0.611	0.993	0.950	0.865
80	-103	-0.071	-0.604	0.609	0.980	0.965	0.806	-0.085	-0.596	0.602	0.993	0.950	0.867
81	-102	-0.084	-0.594	0.600	0.982	0.965	0.811	-0.093	-0.587	0.595	0.995	0.949	0.867
82	-101	-0.079	-0.597	0.602	0.982	0.965	0.808	-0.090	-0.589	0.595	0.993	0.950	0.868
83	-100	-0.073	-0.597	0.601	0.982	0.964	0.809	-0.082	-0.587	0.592	0.995	0.949	0.867
84	-99	-0.062	-0.588	0.591	0.984	0.963	0.811	-0.073	-0.575	0.580	0.993	0.948	0.869
85	-98	-0.065	-0.585	0.589	0.986	0.964	0.813	-0.072	-0.573	0.578	0.994	0.948	0.869
86	-97	-0.052	-0.610	0.612	0.981	0.964	0.808	-0.064	-0.593	0.597	0.995	0.947	0.867
87	-96	-0.053	-0.600	0.602	0.983	0.962	0.811	-0.061	-0.581	0.584	0.993	0.947	0.869
88	-95	-0.055	-0.567	0.570	0.982	0.961	0.820	-0.060	-0.563	0.566	0.994	0.945	0.873
89	-94	-0.059	-0.578	0.581	0.983	0.961	0.817	-0.069	-0.567	0.571	0.996	0.944	0.871
90	-93	-0.055	-0.554	0.556	0.982	0.961	0.823	-0.065	-0.552	0.556	0.994	0.943	0.874
91	-92	-0.052	-0.567	0.569	0.982	0.959	0.821	-0.060	-0.560	0.563	0.994	0.943	0.872
92	-91	-0.062	-0.555	0.559	0.983	0.959	0.824	-0.067	-0.552	0.556	0.993	0.942	0.874
93	-90	-0.067	-0.559	0.563	0.984	0.959	0.824	-0.076	-0.554	0.559	0.994	0.942	0.873
94	-89	-0.062	-0.558	0.561	0.984	0.958	0.824	-0.069	-0.552	0.556	0.994	0.942	0.873
95	-88	-0.067	-0.553	0.557	0.983	0.959	0.824	-0.078	-0.551	0.556	0.989	0.942	0.872
96	-87	-0.071	-0.555	0.560	0.983	0.956	0.825	-0.082	-0.551	0.557	0.992	0.942	0.871

97	-86	-0.054	-0.572	0.574	0.982	0.956	0.821	-0.068	-0.565	0.569	0.993	0.941	0.870
98	-85	-0.065	-0.578	0.581	0.981	0.956	0.819	-0.072	-0.567	0.572	0.993	0.942	0.869
99	-84	-0.065	-0.562	0.566	0.980	0.953	0.826	-0.083	-0.555	0.561	0.991	0.942	0.869
100	-83	-0.071	-0.576	0.580	0.981	0.951	0.821	-0.087	-0.559	0.566	0.991	0.942	0.869
101	-82	-0.064	-0.582	0.586	0.982	0.952	0.817	-0.077	-0.569	0.574	0.993	0.942	0.869
102	-81	-0.062	-0.572	0.575	0.982	0.951	0.822	-0.078	-0.564	0.569	0.993	0.943	0.870
103	-80	-0.057	-0.562	0.565	0.982	0.950	0.824	-0.076	-0.558	0.563	0.993	0.940	0.872
104	-79	-0.065	-0.579	0.583	0.981	0.951	0.819	-0.080	-0.567	0.573	0.992	0.941	0.870
105	-78	-0.060	-0.574	0.577	0.982	0.949	0.819	-0.077	-0.569	0.575	0.993	0.940	0.870
106	-77	-0.052	-0.578	0.580	0.981	0.949	0.818	-0.068	-0.573	0.577	0.994	0.940	0.870
107	-76	-0.057	-0.573	0.576	0.980	0.948	0.820	-0.072	-0.571	0.576	0.993	0.940	0.870
108	-75	-0.061	-0.574	0.578	0.983	0.947	0.820	-0.074	-0.575	0.579	0.995	0.941	0.869
109	-74	-0.068	-0.569	0.573	0.984	0.947	0.824	-0.076	-0.571	0.576	0.995	0.941	0.870
110	-73	-0.063	-0.573	0.577	0.984	0.947	0.822	-0.080	-0.576	0.581	0.994	0.941	0.869
111	-72	-0.053	-0.582	0.584	0.981	0.946	0.818	-0.071	-0.581	0.585	0.992	0.941	0.869
112	-71	-0.058	-0.577	0.580	0.982	0.945	0.820	-0.076	-0.578	0.583	0.993	0.943	0.869
113	-70	-0.072	-0.564	0.569	0.984	0.945	0.825	-0.083	-0.570	0.576	0.993	0.942	0.870
114	-69	-0.067	-0.573	0.577	0.982	0.946	0.821	-0.085	-0.574	0.580	0.993	0.942	0.870
115	-68	-0.062	-0.560	0.563	0.982	0.946	0.826	-0.083	-0.561	0.567	0.994	0.942	0.872
116	-67	-0.061	-0.582	0.585	0.981	0.947	0.818	-0.079	-0.579	0.585	0.995	0.941	0.869
117	-66	-0.060	-0.581	0.584	0.982	0.947	0.819	-0.079	-0.577	0.582	0.992	0.941	0.868
118	-65	-0.058	-0.574	0.577	0.981	0.947	0.821	-0.077	-0.574	0.579	0.992	0.941	0.869
119	-64	-0.052	-0.587	0.590	0.982	0.946	0.817	-0.066	-0.578	0.582	0.993	0.942	0.869
120	-63	-0.066	-0.591	0.594	0.982	0.945	0.817	-0.077	-0.585	0.590	0.994	0.942	0.867
121	-62	-0.048	-0.586	0.588	0.984	0.944	0.816	-0.063	-0.581	0.584	0.994	0.942	0.868
122	-61	-0.061	-0.593	0.596	0.983	0.943	0.815	-0.076	-0.589	0.594	0.992	0.942	0.866
123	-60	-0.055	-0.603	0.605	0.983	0.942	0.812	-0.070	-0.598	0.602	0.995	0.941	0.865
124	-59	-0.047	-0.612	0.613	0.982	0.941	0.809	-0.059	-0.611	0.614	0.995	0.942	0.864
125	-58	-0.051	-0.626	0.628	0.981	0.940	0.804	-0.061	-0.626	0.629	0.994	0.942	0.861
126	-57	-0.037	-0.633	0.634	0.979	0.939	0.799	-0.053	-0.630	0.632	0.993	0.942	0.860
127	-56	-0.030	-0.621	0.621	0.981	0.939	0.804	-0.048	-0.614	0.616	0.993	0.944	0.864
128	-55	-0.023	-0.616	0.616	0.982	0.939	0.805	-0.044	-0.620	0.621	0.994	0.943	0.863
129	-54	-0.032	-0.603	0.604	0.982	0.940	0.811	-0.054	-0.615	0.617	0.992	0.944	0.864
130	-53	-0.042	-0.622	0.623	0.981	0.939	0.804	-0.061	-0.625	0.628	0.994	0.945	0.861
131	-52	-0.035	-0.610	0.611	0.981	0.939	0.808	-0.055	-0.612	0.615	0.994	0.946	0.864
132	-51	-0.041	-0.617	0.618	0.983	0.939	0.805	-0.060	-0.625	0.627	0.992	0.946	0.861
133	-50	-0.034	-0.612	0.613	0.981	0.939	0.806	-0.057	-0.620	0.623	0.991	0.946	0.862
134	-49	-0.034	-0.605	0.606	0.982	0.940	0.808	-0.057	-0.617	0.620	0.993	0.946	0.862
135	-48	-0.033	-0.612	0.613	0.982	0.939	0.804	-0.052	-0.614	0.616	0.994	0.945	0.863
136	-47	-0.039	-0.630	0.632	0.981	0.939	0.801	-0.056	-0.629	0.632	0.995	0.945	0.859
137	-46	-0.039	-0.616	0.618	0.980	0.939	0.805	-0.055	-0.612	0.614	0.993	0.944	0.863
138	-45	-0.040	-0.626	0.627	0.981	0.938	0.801	-0.063	-0.622	0.626	0.992	0.945	0.861
139	-44	-0.037	-0.607	0.608	0.980	0.938	0.808	-0.055	-0.607	0.610	0.992	0.944	0.864
140	-43	-0.031	-0.608	0.609	0.983	0.938	0.806	-0.053	-0.612	0.614	0.992	0.944	0.863
141	-42	-0.033	-0.596	0.597	0.981	0.936	0.813	-0.053	-0.599	0.601	0.990	0.944	0.866
142	-41	-0.028	-0.614	0.614	0.982	0.935	0.804	-0.047	-0.609	0.611	0.994	0.944	0.865
143	-40	-0.015	-0.628	0.628	0.980	0.934	0.803	-0.034	-0.614	0.615	0.993	0.943	0.864
144	-39	-0.016	-0.651	0.651	0.982	0.935	0.793	-0.034	-0.639	0.639	0.994	0.943	0.859
145	-38	-0.020	-0.643	0.643	0.981	0.933	0.795	-0.042	-0.628	0.629	0.995	0.943	0.861
146	-37	-0.013	-0.639	0.639	0.984	0.934	0.797	-0.034	-0.634	0.635	0.993	0.943	0.860
147	-36	-0.001	-0.649	0.649	0.983	0.935	0.794	-0.027	-0.641	0.641	0.995	0.944	0.860
148	-35	-0.016	-0.658	0.658	0.982	0.934	0.791	-0.035	-0.644	0.645	0.994	0.945	0.859
149	-34	-0.022	-0.654	0.654	0.980	0.935	0.792	-0.041	-0.644	0.645	0.994	0.944	0.858
150	-33	-0.015	-0.665	0.665	0.982	0.935	0.790	-0.040	-0.656	0.657	0.992	0.946	0.856
151	-32	-0.017	-0.661	0.662	0.982	0.937	0.790	-0.043	-0.655	0.656	0.993	0.945	0.855
152	-31	-0.010	-0.655	0.656	0.982	0.936	0.791	-0.033	-0.648	0.649	0.994	0.947	0.855
153	-30	-0.015	-0.660	0.660	0.983	0.936	0.790	-0.036	-0.653	0.654	0.994	0.946	0.855
154	-29	-0.018	-0.683	0.684	0.984	0.937	0.786	-0.036	-0.672	0.673	0.995	0.947	0.852
155	-28	-0.019	-0.671	0.672	0.983	0.937	0.788	-0.043	-0.663	0.665	0.994	0.948	0.853
156	-27	-0.016	-0.677	0.678	0.982	0.937	0.787	-0.040	-0.668	0.669	0.995	0.948	0.853

157	-26	-0.024	-0.673	0.673	0.983	0.939	0.788	-0.043	-0.662	0.663	0.993	0.949	0.854
158	-25	-0.014	-0.663	0.663	0.981	0.939	0.791	-0.043	-0.658	0.659	0.992	0.950	0.853
159	-24	-0.009	-0.675	0.675	0.983	0.938	0.788	-0.036	-0.664	0.665	0.994	0.950	0.852
160	-23	-0.009	-0.682	0.682	0.983	0.940	0.787	-0.037	-0.669	0.670	0.994	0.950	0.852
161	-22	-0.005	-0.672	0.672	0.984	0.940	0.788	-0.032	-0.663	0.664	0.994	0.950	0.852
162	-21	0.003	-0.679	0.679	0.981	0.939	0.787	-0.024	-0.669	0.670	0.994	0.950	0.851
163	-20	-0.002	-0.690	0.690	0.984	0.940	0.784	-0.024	-0.681	0.681	0.996	0.950	0.848
164	-19	0.004	-0.692	0.692	0.983	0.940	0.785	-0.022	-0.685	0.685	0.996	0.950	0.847
165	-18	0.005	-0.688	0.688	0.984	0.942	0.787	-0.022	-0.685	0.686	0.995	0.951	0.847
166	-17	0.007	-0.690	0.690	0.984	0.943	0.786	-0.019	-0.687	0.688	0.995	0.952	0.846
167	-16	0.001	-0.700	0.700	0.983	0.943	0.783	-0.023	-0.697	0.698	0.996	0.952	0.843
168	-15	0.006	-0.697	0.697	0.983	0.944	0.783	-0.024	-0.695	0.695	0.996	0.953	0.844
169	-14	0.003	-0.702	0.702	0.983	0.943	0.781	-0.026	-0.697	0.698	0.996	0.953	0.843
170	-13	0.002	-0.698	0.698	0.982	0.943	0.784	-0.027	-0.697	0.698	0.996	0.953	0.843
171	-12	0.003	-0.689	0.689	0.979	0.942	0.785	-0.026	-0.692	0.693	0.994	0.953	0.844
172	-11	0.009	-0.683	0.683	0.972	0.943	0.786	-0.026	-0.687	0.687	0.990	0.953	0.845
173	-10	-1.000	-1.000	-1.000	-1.000	0.942	-1.000	-1.000	-1.000	-1.000	-1.000	0.953	-1.000
174	-9	-1.000	-1.000	-1.000	-1.000	0.943	-1.000	-1.000	-1.000	-1.000	-1.000	0.954	-1.000
175	-8	-1.000	-1.000	-1.000	-1.000	0.942	-1.000	-1.000	-1.000	-1.000	-1.000	0.954	-1.000
176	-7	-1.000	-1.000	-1.000	-1.000	0.941	-1.000	-1.000	-1.000	-1.000	-1.000	0.954	-1.000
177	-6	-1.000	-1.000	-1.000	-1.000	0.942	-1.000	-1.000	-1.000	-1.000	-1.000	0.954	-1.000
178	-5	-1.000	-1.000	-1.000	-1.000	0.943	-1.000	-1.000	-1.000	-1.000	-1.000	0.953	-1.000
179	-4	-1.000	-1.000	-1.000	-1.000	0.944	-1.000	-1.000	-1.000	-1.000	-1.000	0.954	-1.000
180	-3	-1.000	-1.000	-1.000	-1.000	0.943	-1.000	-1.000	-1.000	-1.000	-1.000	0.955	-1.000
181	-2	-1.000	-1.000	-1.000	-1.000	0.943	-1.000	-1.000	-1.000	-1.000	-1.000	0.955	-1.000
182	-1	-1.000	-1.000	-1.000	-1.000	0.943	-1.000	-1.000	-1.000	-1.000	-1.000	0.955	-1.000
183	0	-1.000	-1.000	-1.000	-1.000	0.943	-1.000	-1.000	-1.000	-1.000	-1.000	0.955	-1.000
184	1	-1.000	-1.000	-1.000	-1.000	0.944	-1.000	-1.000	-1.000	-1.000	-1.000	0.955	-1.000
185	2	-1.000	-1.000	-1.000	-1.000	0.944	-1.000	-1.000	-1.000	-1.000	-1.000	0.956	-1.000
186	3	-1.000	-1.000	-1.000	-1.000	0.944	-1.000	-1.000	-1.000	-1.000	-1.000	0.955	-1.000
187	4	-1.000	-1.000	-1.000	-1.000	0.944	-1.000	-1.000	-1.000	-1.000	-1.000	0.954	-1.000
188	5	-1.000	-1.000	-1.000	-1.000	0.945	-1.000	-1.000	-1.000	-1.000	-1.000	0.954	-1.000
189	6	-1.000	-1.000	-1.000	-1.000	0.947	-1.000	-1.000	-1.000	-1.000	-1.000	0.954	-1.000
190	7	-1.000	-1.000	-1.000	-1.000	0.947	-1.000	-1.000	-1.000	-1.000	-1.000	0.953	-1.000
191	8	-1.000	-1.000	-1.000	-1.000	0.947	-1.000	-1.000	-1.000	-1.000	-1.000	0.952	-1.000
192	9	-1.000	-1.000	-1.000	-1.000	0.947	-1.000	-1.000	-1.000	-1.000	-1.000	0.952	-1.000
193	10	-1.000	-1.000	-1.000	-1.000	0.947	-1.000	-1.000	-1.000	-1.000	-1.000	0.953	-1.000
194	11	0.038	-0.709	0.710	0.971	0.948	0.778	0.007	-0.712	0.712	0.988	0.953	0.835
195	12	0.036	-0.705	0.706	0.973	0.947	0.779	0.005	-0.713	0.713	0.991	0.954	0.835
196	13	0.034	-0.703	0.704	0.978	0.947	0.778	-0.002	-0.715	0.715	0.995	0.954	0.834
197	14	0.036	-0.696	0.697	0.981	0.946	0.781	0.001	-0.708	0.708	0.995	0.953	0.835
198	15	0.037	-0.694	0.695	0.982	0.945	0.781	0.004	-0.707	0.707	0.995	0.954	0.836
199	16	0.041	-0.692	0.693	0.983	0.944	0.783	0.009	-0.706	0.706	0.995	0.955	0.835
200	17	0.047	-0.697	0.699	0.983	0.945	0.781	0.016	-0.713	0.713	0.994	0.955	0.834
201	18	0.053	-0.701	0.703	0.983	0.947	0.778	0.023	-0.717	0.717	0.994	0.955	0.833
202	19	0.056	-0.706	0.708	0.983	0.948	0.776	0.029	-0.724	0.725	0.994	0.956	0.833
203	20	0.062	-0.697	0.699	0.984	0.948	0.778	0.037	-0.717	0.718	0.994	0.957	0.833
204	21	0.067	-0.696	0.699	0.985	0.949	0.779	0.043	-0.716	0.717	0.993	0.955	0.833
205	22	0.069	-0.697	0.701	0.985	0.951	0.776	0.047	-0.721	0.722	0.992	0.955	0.832
206	23	0.069	-0.694	0.698	0.986	0.952	0.777	0.047	-0.717	0.719	0.992	0.957	0.833
207	24	0.064	-0.696	0.699	0.985	0.951	0.776	0.050	-0.719	0.720	0.992	0.956	0.833
208	25	0.068	-0.695	0.698	0.986	0.950	0.776	0.050	-0.718	0.720	0.992	0.954	0.833
209	26	0.067	-0.688	0.691	0.984	0.949	0.778	0.053	-0.710	0.712	0.993	0.954	0.834
210	27	0.070	-0.683	0.687	0.984	0.950	0.780	0.057	-0.705	0.707	0.993	0.952	0.835
211	28	0.066	-0.685	0.688	0.984	0.952	0.780	0.056	-0.701	0.704	0.994	0.951	0.835
212	29	0.069	-0.696	0.699	0.983	0.953	0.775	0.056	-0.709	0.711	0.994	0.950	0.835
213	30	0.069	-0.690	0.694	0.983	0.952	0.778	0.057	-0.705	0.707	0.994	0.949	0.836
214	31	0.061	-0.685	0.688	0.983	0.953	0.783	0.052	-0.697	0.699	0.995	0.949	0.838
215	32	0.068	-0.678	0.681	0.982	0.955	0.784	0.054	-0.692	0.695	0.995	0.949	0.839
216	33	0.064	-0.666	0.669	0.982	0.953	0.786	0.048	-0.689	0.691	0.994	0.950	0.840

217	34	0.069	-0.669	0.672	0.983	0.952	0.787	0.054	-0.686	0.688	0.993	0.950	0.841
218	35	0.073	-0.664	0.668	0.983	0.952	0.789	0.057	-0.688	0.690	0.995	0.951	0.841
219	36	0.069	-0.652	0.656	0.982	0.952	0.791	0.059	-0.678	0.681	0.993	0.952	0.842
220	37	0.079	-0.669	0.674	0.981	0.953	0.785	0.064	-0.692	0.695	0.993	0.950	0.839
221	38	0.080	-0.680	0.684	0.982	0.952	0.782	0.072	-0.707	0.711	0.993	0.949	0.835
222	39	0.090	-0.673	0.679	0.983	0.953	0.783	0.078	-0.696	0.701	0.992	0.948	0.837
223	40	0.099	-0.677	0.684	0.983	0.951	0.780	0.087	-0.704	0.710	0.994	0.945	0.834
224	41	0.094	-0.690	0.697	0.984	0.952	0.777	0.088	-0.708	0.714	0.993	0.943	0.832
225	42	0.088	-0.697	0.703	0.984	0.951	0.775	0.083	-0.712	0.717	0.993	0.943	0.831
226	43	0.088	-0.681	0.687	0.984	0.951	0.780	0.086	-0.697	0.702	0.993	0.941	0.833
227	44	0.089	-0.678	0.684	0.984	0.951	0.782	0.085	-0.699	0.704	0.992	0.942	0.833
228	45	0.075	-0.664	0.669	0.983	0.950	0.787	0.073	-0.686	0.690	0.992	0.941	0.837
229	46	0.073	-0.650	0.654	0.985	0.947	0.792	0.072	-0.675	0.679	0.991	0.942	0.841
230	47	0.073	-0.639	0.643	0.984	0.948	0.796	0.075	-0.661	0.666	0.993	0.942	0.844
231	48	0.078	-0.654	0.659	0.983	0.948	0.790	0.079	-0.677	0.681	0.993	0.940	0.841
232	49	0.086	-0.625	0.631	0.982	0.944	0.801	0.083	-0.648	0.653	0.993	0.940	0.847
233	50	0.083	-0.618	0.623	0.982	0.941	0.802	0.086	-0.645	0.651	0.992	0.940	0.849
234	51	0.086	-0.612	0.618	0.981	0.942	0.803	0.083	-0.642	0.648	0.992	0.942	0.849
235	52	0.081	-0.605	0.610	0.983	0.945	0.805	0.086	-0.629	0.635	0.989	0.944	0.851
236	53	0.087	-0.568	0.575	0.982	0.944	0.814	0.089	-0.599	0.606	0.993	0.944	0.858
237	54	0.088	-0.547	0.554	0.982	0.946	0.820	0.088	-0.592	0.598	0.991	0.946	0.860
238	55	0.088	-0.538	0.545	0.981	0.946	0.825	0.086	-0.588	0.594	0.991	0.947	0.860
239	56	0.103	-0.546	0.556	0.981	0.944	0.825	0.098	-0.586	0.594	0.995	0.946	0.861
240	57	0.107	-0.526	0.537	0.982	0.946	0.828	0.105	-0.567	0.576	0.992	0.944	0.862
241	58	0.109	-0.498	0.510	0.981	0.944	0.834	0.112	-0.547	0.558	0.992	0.942	0.866
242	59	0.117	-0.469	0.483	0.980	0.950	0.844	0.118	-0.520	0.533	0.992	0.941	0.871
243	60	0.109	-0.447	0.460	0.983	0.950	0.852	0.114	-0.495	0.508	0.992	0.940	0.876
244	61	0.099	-0.424	0.435	0.983	0.948	0.859	0.109	-0.475	0.487	0.993	0.935	0.879
245	62	0.096	-0.393	0.405	0.984	0.949	0.867	0.111	-0.445	0.459	0.992	0.935	0.883
246	63	0.104	-0.353	0.368	0.983	0.953	0.879	0.120	-0.410	0.427	0.992	0.935	0.889
247	64	0.105	-0.339	0.355	0.982	0.952	0.885	0.111	-0.402	0.417	0.992	0.934	0.892
248	65	0.108	-0.331	0.348	0.984	0.950	0.888	0.116	-0.387	0.404	0.992	0.935	0.896
249	66	0.108	-0.335	0.352	0.983	0.947	0.885	0.114	-0.387	0.404	0.991	0.935	0.896
250	67	0.107	-0.324	0.341	0.984	0.942	0.890	0.111	-0.379	0.395	0.990	0.934	0.898
251	68	0.107	-0.332	0.348	0.983	0.939	0.887	0.115	-0.386	0.403	0.992	0.933	0.896
252	69	0.105	-0.325	0.342	0.981	0.936	0.890	0.107	-0.383	0.398	0.992	0.934	0.897
253	70	0.089	-0.336	0.348	0.981	0.938	0.887	0.092	-0.393	0.403	0.989	0.934	0.894
254	71	0.100	-0.341	0.355	0.981	0.939	0.884	0.103	-0.405	0.418	0.991	0.936	0.891
255	72	0.094	-0.370	0.382	0.982	0.943	0.873	0.102	-0.429	0.441	0.989	0.935	0.885
256	73	0.095	-0.382	0.393	0.981	0.943	0.872	0.101	-0.444	0.456	0.990	0.936	0.881
257	74	0.107	-0.394	0.408	0.983	0.941	0.866	0.113	-0.455	0.469	0.991	0.938	0.878
258	75	0.115	-0.431	0.446	0.983	0.943	0.855	0.121	-0.486	0.501	0.993	0.936	0.870
259	76	0.116	-0.461	0.475	0.982	0.946	0.847	0.126	-0.512	0.528	0.992	0.939	0.863
260	77	0.121	-0.463	0.478	0.982	0.946	0.845	0.131	-0.515	0.532	0.992	0.937	0.864
261	78	0.126	-0.456	0.473	0.981	0.943	0.848	0.135	-0.508	0.526	0.991	0.937	0.868
262	79	0.125	-0.482	0.498	0.982	0.941	0.839	0.135	-0.525	0.542	0.992	0.936	0.864
263	80	0.125	-0.492	0.508	0.982	0.942	0.837	0.137	-0.529	0.547	0.993	0.936	0.865
264	81	0.136	-0.494	0.512	0.982	0.942	0.838	0.148	-0.522	0.543	0.992	0.935	0.867
265	82	0.128	-0.485	0.502	0.981	0.938	0.842	0.141	-0.520	0.539	0.991	0.935	0.868
266	83	0.121	-0.515	0.529	0.983	0.937	0.833	0.135	-0.536	0.553	0.991	0.935	0.867
267	84	0.124	-0.528	0.543	0.982	0.937	0.829	0.131	-0.547	0.562	0.991	0.937	0.864
268	85	0.118	-0.531	0.544	0.983	0.939	0.830	0.118	-0.547	0.560	0.993	0.937	0.865
269	86	0.127	-0.528	0.543	0.982	0.942	0.830	0.133	-0.543	0.559	0.993	0.938	0.865
270	87	0.115	-0.554	0.565	0.981	0.942	0.822	0.123	-0.565	0.579	0.990	0.939	0.860
271	88	0.111	-0.570	0.581	0.983	0.943	0.815	0.117	-0.575	0.587	0.994	0.941	0.859
272	89	0.115	-0.547	0.559	0.981	0.943	0.824	0.121	-0.555	0.568	0.991	0.942	0.862
273	90	0.121	-0.562	0.575	0.982	0.946	0.819	0.124	-0.565	0.578	0.992	0.940	0.860
274	91	0.106	-0.578	0.588	0.984	0.948	0.809	0.112	-0.575	0.586	0.991	0.936	0.855
275	92	0.107	-0.586	0.596	0.983	0.946	0.806	0.111	-0.577	0.587	0.993	0.934	0.854
276	93	0.117	-0.615	0.626	0.983	0.946	0.794	0.121	-0.596	0.608	0.992	0.932	0.849

277	94	0.112	-0.622	0.632	0.983	0.946	0.792	0.114	-0.596	0.607	0.992	0.929	0.849
278	95	0.128	-0.634	0.647	0.983	0.946	0.785	0.128	-0.606	0.620	0.992	0.926	0.845
279	96	0.132	-0.650	0.663	0.983	0.942	0.778	0.133	-0.617	0.631	0.990	0.925	0.842
280	97	0.125	-0.617	0.630	0.983	0.945	0.792	0.122	-0.587	0.599	0.993	0.928	0.851
281	98	0.122	-0.602	0.615	0.983	0.939	0.801	0.123	-0.579	0.592	0.993	0.929	0.854
282	99	0.116	-0.577	0.589	0.982	0.937	0.810	0.113	-0.554	0.565	0.994	0.928	0.862
283	100	0.117	-0.541	0.553	0.983	0.936	0.826	0.116	-0.523	0.536	0.992	0.930	0.873
284	101	0.111	-0.508	0.520	0.982	0.934	0.840	0.115	-0.495	0.509	0.991	0.932	0.879
285	102	0.092	-0.497	0.506	0.982	0.925	0.843	0.096	-0.488	0.498	0.992	0.932	0.884
286	103	0.085	-0.488	0.495	0.982	0.922	0.846	0.088	-0.473	0.481	0.993	0.933	0.888
287	104	0.095	-0.495	0.504	0.982	0.924	0.845	0.099	-0.478	0.489	0.994	0.934	0.886
288	105	0.100	-0.517	0.526	0.982	0.927	0.839	0.099	-0.496	0.505	0.993	0.935	0.881
289	106	0.099	-0.536	0.545	0.982	0.930	0.830	0.099	-0.514	0.523	0.993	0.936	0.877
290	107	0.098	-0.536	0.545	0.981	0.931	0.830	0.098	-0.519	0.528	0.992	0.937	0.875
291	108	0.093	-0.557	0.564	0.982	0.931	0.826	0.093	-0.538	0.546	0.993	0.937	0.872
292	109	0.097	-0.574	0.582	0.980	0.934	0.820	0.096	-0.551	0.560	0.991	0.940	0.869
293	110	0.109	-0.584	0.594	0.982	0.938	0.814	0.104	-0.561	0.570	0.993	0.939	0.865
294	111	0.106	-0.595	0.605	0.982	0.940	0.799	0.100	-0.572	0.581	0.990	0.936	0.860
295	112	0.115	-0.614	0.625	0.981	0.936	0.780	0.106	-0.594	0.604	0.992	0.933	0.851
296	113	0.133	-0.603	0.617	0.981	0.914	0.774	0.112	-0.583	0.593	0.993	0.928	0.849
297	114	0.148	-0.624	0.641	0.982	0.864	0.741	0.120	-0.599	0.611	0.993	0.923	0.841
298	115	0.163	-0.658	0.678	0.982	0.803	0.704	0.120	-0.636	0.648	0.991	0.914	0.826
299	116	0.196	-0.691	0.718	0.983	0.715	0.647	0.118	-0.666	0.676	0.993	0.907	0.816
300	117	0.239	-0.744	0.781	0.985	0.603	0.563	0.128	-0.707	0.718	0.992	0.899	0.800
301	118	0.300	-0.870	0.920	0.984	0.431	0.409	0.115	-0.794	0.802	0.991	0.885	0.776
302	119	0.394	-1.132	1.198	0.981	0.145	0.146	0.114	-0.890	0.897	0.990	0.863	0.748
303	120	-1.000	-1.000	-1.000	0.982	-1.000	0.000	0.109	-1.002	1.008	0.992	0.832	0.713
304	121	-1.000	-1.000	-1.000	0.981	-1.000	0.000	0.098	-1.175	1.179	0.993	0.785	0.662
305	122	-1.000	-1.000	-1.000	0.983	-1.000	0.000	0.088	-1.408	1.410	0.992	0.716	0.590
306	123	-1.000	-1.000	-1.000	0.982	-1.000	0.000	0.079	-1.573	1.575	0.992	0.676	0.544
307	124	-1.000	-1.000	-1.000	0.981	-1.000	0.000	0.087	-1.533	1.535	0.992	0.716	0.571
308	125	-1.000	-1.000	-1.000	0.981	-1.000	0.000	0.093	-1.519	1.522	0.993	0.752	0.591
309	126	-1.000	-1.000	-1.000	0.981	-1.000	0.000	0.092	-1.475	1.477	0.992	0.785	0.617
310	127	0.379	-2.003	2.038	0.981	0.076	0.022	0.079	-1.441	1.444	0.992	0.815	0.636
311	128	0.343	-1.984	2.013	0.981	0.165	0.061	0.082	-1.399	1.401	0.992	0.843	0.658
312	129	0.306	-1.960	1.984	0.979	0.246	0.099	0.089	-1.355	1.358	0.992	0.863	0.681
313	130	0.283	-1.903	1.924	0.979	0.319	0.153	0.095	-1.182	1.186	0.991	0.885	0.731
314	131	0.263	-1.713	1.733	0.979	0.436	0.270	0.097	-1.001	1.006	0.991	0.900	0.775
315	132	0.237	-1.412	1.432	0.978	0.577	0.419	0.091	-0.879	0.884	0.989	0.902	0.799
316	133	0.215	-1.116	1.136	0.979	0.701	0.543	0.101	-0.780	0.787	0.992	0.906	0.824
317	134	0.183	-0.888	0.907	0.978	0.792	0.642	0.099	-0.685	0.692	0.991	0.909	0.842
318	135	0.160	-0.735	0.752	0.980	0.858	0.710	0.099	-0.620	0.628	0.992	0.913	0.857
319	136	0.150	-0.606	0.624	0.980	0.898	0.763	0.106	-0.557	0.567	0.989	0.914	0.870
320	137	0.153	-0.546	0.567	0.979	0.915	0.795	0.113	-0.523	0.535	0.991	0.917	0.876
321	138	0.141	-0.526	0.545	0.978	0.923	0.809	0.112	-0.522	0.533	0.991	0.922	0.876
322	139	0.119	-0.475	0.490	0.979	0.923	0.835	0.100	-0.477	0.487	0.993	0.926	0.886
323	140	0.105	-0.452	0.464	0.979	0.927	0.847	0.091	-0.467	0.475	0.993	0.927	0.887
324	141	0.101	-0.391	0.404	0.980	0.928	0.872	0.087	-0.417	0.426	0.992	0.931	0.896
325	142	0.108	-0.362	0.378	0.980	0.927	0.880	0.100	-0.394	0.407	0.991	0.935	0.899
326	143	0.101	-0.361	0.375	0.981	0.928	0.883	0.094	-0.390	0.402	0.992	0.936	0.901
327	144	0.098	-0.366	0.379	0.980	0.928	0.884	0.085	-0.400	0.409	0.990	0.936	0.900
328	145	0.099	-0.362	0.375	0.979	0.928	0.885	0.088	-0.390	0.399	0.992	0.936	0.903
329	146	0.095	-0.333	0.346	0.979	0.927	0.893	0.083	-0.361	0.371	0.990	0.936	0.909
330	147	0.090	-0.332	0.344	0.980	0.925	0.894	0.084	-0.360	0.369	0.992	0.935	0.910
331	148	0.089	-0.330	0.342	0.980	0.923	0.895	0.076	-0.355	0.363	0.989	0.935	0.912
332	149	0.084	-0.326	0.337	0.979	0.924	0.896	0.075	-0.347	0.355	0.989	0.935	0.913
333	150	0.110	-0.304	0.323	0.980	0.924	0.898	0.090	-0.336	0.348	0.990	0.935	0.913
334	151	0.104	-0.307	0.324	0.978	0.927	0.895	0.091	-0.343	0.355	0.990	0.934	0.911
335	152	0.102	-0.319	0.335	0.979	0.930	0.895	0.093	-0.351	0.363	0.991	0.936	0.910
336	153	0.115	-0.337	0.356	0.979	0.931	0.885	0.102	-0.369	0.383	0.994	0.938	0.905

337	154	0.119	-0.320	0.341	0.977	0.933	0.892	0.107	-0.356	0.371	0.991	0.939	0.907
338	155	0.114	-0.314	0.334	0.979	0.934	0.894	0.096	-0.363	0.376	0.989	0.940	0.905
339	156	0.114	-0.320	0.340	0.978	0.933	0.894	0.096	-0.371	0.383	0.990	0.941	0.902
340	157	0.111	-0.316	0.335	0.978	0.932	0.894	0.096	-0.361	0.374	0.991	0.941	0.904
341	158	0.106	-0.299	0.317	0.979	0.933	0.898	0.093	-0.346	0.359	0.990	0.941	0.909
342	159	0.116	-0.277	0.300	0.978	0.932	0.904	0.102	-0.332	0.347	0.989	0.942	0.912
343	160	0.109	-0.267	0.288	0.980	0.934	0.908	0.102	-0.321	0.337	0.991	0.943	0.914
344	161	0.102	-0.248	0.268	0.980	0.933	0.911	0.095	-0.302	0.316	0.991	0.944	0.919
345	162	0.109	-0.243	0.267	0.980	0.931	0.912	0.098	-0.295	0.311	0.992	0.944	0.921
346	163	0.108	-0.246	0.269	0.980	0.930	0.911	0.099	-0.297	0.313	0.991	0.945	0.921
347	164	0.108	-0.256	0.278	0.978	0.932	0.911	0.094	-0.305	0.319	0.991	0.945	0.918
348	165	0.087	-0.261	0.275	0.981	0.933	0.910	0.078	-0.308	0.317	0.991	0.944	0.918
349	166	0.089	-0.240	0.256	0.981	0.933	0.911	0.079	-0.293	0.303	0.992	0.944	0.921
350	167	0.092	-0.241	0.258	0.980	0.932	0.910	0.083	-0.287	0.299	0.991	0.943	0.922
351	168	0.096	-0.263	0.280	0.980	0.934	0.908	0.087	-0.306	0.318	0.992	0.942	0.917
352	169	0.090	-0.259	0.274	0.981	0.933	0.911	0.077	-0.296	0.306	0.993	0.943	0.920
353	170	0.107	-0.249	0.271	0.980	0.933	0.908	0.095	-0.287	0.303	0.993	0.944	0.921
354	171	0.110	-0.262	0.284	0.979	0.933	0.905	0.100	-0.304	0.320	0.990	0.946	0.917
355	172	0.103	-0.266	0.285	0.980	0.935	0.907	0.098	-0.308	0.324	0.993	0.950	0.916
356	173	0.115	-0.247	0.272	0.979	0.936	0.909	0.114	-0.290	0.312	0.992	0.952	0.918
357	174	0.117	-0.221	0.250	0.979	0.937	0.916	0.118	-0.272	0.297	0.993	0.955	0.923
358	175	0.103	-0.230	0.252	0.980	0.938	0.915	0.106	-0.280	0.300	0.990	0.957	0.921
359	176	0.098	-0.241	0.260	0.982	0.940	0.913	0.101	-0.284	0.302	0.991	0.960	0.921
360	177	0.092	-0.229	0.247	0.984	0.940	0.917	0.100	-0.272	0.290	0.991	0.961	0.925
361	178	0.090	-0.228	0.245	0.983	0.941	0.917	0.097	-0.272	0.289	0.992	0.962	0.925
362	179	0.095	-0.209	0.229	0.982	0.942	0.921	0.097	-0.255	0.273	0.991	0.963	0.929
363	180	0.101	-0.211	0.233	0.981	0.943	0.922	0.104	-0.248	0.269	0.991	0.962	0.929
364	181	0.101	-0.211	0.233	0.982	0.944	0.922	0.108	-0.244	0.267	0.992	0.963	0.930
365	182	0.099	-0.223	0.244	0.984	0.944	0.918	0.103	-0.249	0.269	0.995	0.964	0.931
366	183	0.095	-0.230	0.249	0.982	0.945	0.916	0.100	-0.254	0.272	0.993	0.966	0.930
367	184	0.093	-0.247	0.264	0.982	0.947	0.914	0.096	-0.264	0.282	0.992	0.967	0.928
368	185	0.084	-0.236	0.250	0.981	0.948	0.917	0.086	-0.260	0.274	0.992	0.967	0.929
369	186	0.074	-0.224	0.236	0.982	0.949	0.922	0.081	-0.246	0.259	0.993	0.968	0.934
370	187	0.084	-0.246	0.260	0.983	0.949	0.916	0.088	-0.259	0.273	0.993	0.970	0.929
371	188	0.088	-0.238	0.254	0.981	0.951	0.917	0.096	-0.258	0.276	0.989	0.971	0.928
372	189	0.091	-0.256	0.272	0.980	0.952	0.913	0.100	-0.272	0.289	0.990	0.972	0.925
373	190	0.091	-0.250	0.266	0.984	0.953	0.914	0.096	-0.270	0.287	0.993	0.973	0.925
374	191	0.089	-0.255	0.270	0.983	0.951	0.913	0.101	-0.268	0.286	0.993	0.974	0.926
375	192	0.098	-0.250	0.268	0.983	0.951	0.914	0.104	-0.258	0.278	0.992	0.975	0.929
376	193	0.091	-0.226	0.244	0.983	0.951	0.920	0.101	-0.242	0.262	0.994	0.974	0.933
377	194	0.084	-0.231	0.246	0.983	0.952	0.919	0.089	-0.245	0.261	0.992	0.975	0.933
378	195	0.087	-0.208	0.225	0.984	0.955	0.925	0.094	-0.224	0.243	0.994	0.975	0.938
379	196	0.095	-0.196	0.217	0.984	0.956	0.929	0.101	-0.213	0.236	0.993	0.975	0.940
380	197	0.087	-0.208	0.225	0.982	0.958	0.926	0.092	-0.221	0.239	0.993	0.975	0.938
381	198	0.090	-0.219	0.237	0.983	0.960	0.924	0.094	-0.228	0.247	0.991	0.975	0.936
382	199	0.098	-0.207	0.229	0.983	0.959	0.928	0.102	-0.214	0.237	0.994	0.976	0.940
383	200	0.092	-0.225	0.243	0.982	0.962	0.923	0.095	-0.234	0.252	0.992	0.974	0.935
384	201	0.095	-0.242	0.260	0.982	0.963	0.917	0.095	-0.244	0.262	0.993	0.974	0.932
385	202	0.099	-0.268	0.286	0.982	0.962	0.908	0.093	-0.262	0.278	0.994	0.974	0.929
386	203	0.097	-0.251	0.269	0.979	0.961	0.913	0.096	-0.248	0.266	0.991	0.975	0.932
387	204	0.090	-0.270	0.285	0.980	0.962	0.906	0.091	-0.264	0.279	0.993	0.974	0.929
388	205	0.105	-0.239	0.261	0.981	0.960	0.915	0.098	-0.243	0.263	0.992	0.973	0.932
389	206	0.110	-0.271	0.292	0.982	0.957	0.902	0.100	-0.262	0.280	0.994	0.972	0.929
390	207	0.099	-0.258	0.276	0.981	0.957	0.909	0.092	-0.263	0.279	0.992	0.970	0.928
391	208	0.115	-0.255	0.280	0.982	0.955	0.909	0.110	-0.253	0.276	0.991	0.969	0.929
392	209	0.108	-0.260	0.282	0.981	0.954	0.905	0.101	-0.260	0.279	0.991	0.969	0.928
393	210	0.108	-0.256	0.278	0.982	0.953	0.905	0.099	-0.257	0.275	0.994	0.968	0.929
394	211	0.115	-0.263	0.287	0.981	0.952	0.902	0.104	-0.261	0.281	0.992	0.968	0.929
395	212	0.121	-0.245	0.274	0.982	0.955	0.909	0.111	-0.242	0.267	0.992	0.969	0.931

2017-08-30 14:30:24.872680
CT_x100
path: ../data/phantom3_MR_v2/ph3_CT_x100
thresholds:
lower (simple): -451.940037951,
lower (iter): -624.945436002
upper: 287.0
DC-average (simple): 0.915364967433 (bestRadius: 4.07142857143)
DC-average (iter): 0.994021385791 (bestRadius: 4.35714285714)
MR_x100
path: ../data/phantom3_MR_v2/ph3_MR_v2_x100
thresholds:
lower (simple): 263.1579347,
lower (iter): 101.050751445
lower (simple_CT-COM): 263.1579347
lower (iter_CT-COM): 101.050751445
upper: 1129.0
DC-average (simple): 0.78450336966 (bestRadius: 2.78571428571)
DC-average (iter): 0.879376043843 (bestRadius: 4.07142857143)
DC-average (CT-COM, simple): 0.662185740948 (bestRadius: 2.78571428571)
DC-average (CT-COM, iter): 0.755707740781 (bestRadius: 4.07142857143)

sliceNo	dist	warpX	warpY	warpM	DC_{CT}	DC_{MR}	$DC_{MR(CT)}$	warpX*	warpY*	warpM*	DC^*_{CT}	DC^*_{MR}	$DC^*_{MR(CT)}$
0	-361	2.942	0.429	2.973	0.968	0.033	0.006	3.311	0.853	3.419	0.944	0.613	0.331
1	-360	2.940	0.429	2.971	0.971	0.033	0.006	3.313	0.843	3.418	0.946	0.613	0.331
2	-359	2.944	0.422	2.974	0.973	0.039	0.008	3.290	0.820	3.391	0.948	0.612	0.335
3	-358	2.943	0.421	2.973	0.972	0.056	0.014	3.230	0.755	3.317	0.948	0.606	0.343
4	-357	2.945	0.406	2.973	0.974	0.074	0.018	3.178	0.679	3.250	0.950	0.602	0.350
5	-356	2.917	0.407	2.946	0.973	0.089	0.026	3.088	0.623	3.150	0.951	0.599	0.364
6	-355	2.905	0.433	2.937	0.972	0.110	0.036	3.047	0.640	3.114	0.950	0.610	0.371
7	-354	2.888	0.457	2.924	0.971	0.136	0.050	3.019	0.697	3.098	0.951	0.633	0.382
8	-353	2.874	0.492	2.916	0.969	0.156	0.058	2.981	0.753	3.075	0.949	0.653	0.392
9	-352	2.871	0.526	2.919	0.970	0.181	0.067	2.956	0.805	3.064	0.950	0.673	0.400
10	-351	2.865	0.530	2.914	0.971	0.175	0.064	2.935	0.832	3.050	0.950	0.679	0.404
11	-350	2.861	0.545	2.912	0.969	0.154	0.056	2.906	0.873	3.034	0.950	0.676	0.405
12	-349	2.850	0.559	2.904	0.969	0.134	0.048	2.889	0.913	3.029	0.950	0.674	0.404
13	-348	2.852	0.570	2.908	0.970	0.109	0.038	2.863	0.934	3.012	0.952	0.669	0.407
14	-347	2.827	0.540	2.878	0.973	0.105	0.039	2.796	0.873	2.929	0.953	0.665	0.416
15	-346	2.821	0.505	2.866	0.974	0.103	0.041	2.732	0.792	2.844	0.955	0.658	0.425
16	-345	2.805	0.460	2.842	0.975	0.103	0.044	2.650	0.696	2.740	0.956	0.652	0.439
17	-344	2.779	0.415	2.810	0.983	0.098	0.045	2.563	0.597	2.632	0.963	0.646	0.452
18	-343	2.802	0.412	2.832	0.989	0.096	0.042	2.569	0.532	2.623	0.973	0.625	0.444
19	-342	2.809	0.398	2.837	0.990	0.088	0.039	2.560	0.479	2.604	0.978	0.603	0.436
20	-341	2.791	0.388	2.818	0.987	0.083	0.038	2.526	0.419	2.561	0.981	0.582	0.432
21	-340	2.785	0.381	2.811	0.989	0.085	0.039	2.494	0.396	2.526	0.981	0.574	0.431
22	-339	2.757	0.428	2.790	0.987	0.117	0.057	2.395	0.488	2.445	0.980	0.604	0.454
23	-338	2.712	0.441	2.747	0.989	0.149	0.079	2.295	0.573	2.365	0.980	0.634	0.476
24	-337	2.646	0.479	2.689	0.987	0.191	0.107	2.211	0.646	2.303	0.980	0.658	0.495
25	-336	2.627	0.502	2.674	0.991	0.219	0.127	2.173	0.723	2.290	0.981	0.673	0.503
26	-335	2.629	0.528	2.682	0.990	0.205	0.117	2.155	0.779	2.291	0.983	0.672	0.503
27	-334	2.642	0.537	2.696	0.990	0.194	0.108	2.133	0.839	2.292	0.986	0.668	0.503
28	-333	2.641	0.556	2.699	0.990	0.182	0.099	2.123	0.882	2.299	0.989	0.665	0.502
29	-332	2.646	0.525	2.697	0.988	0.160	0.091	2.093	0.875	2.268	0.990	0.663	0.508
30	-331	2.624	0.507	2.672	0.985	0.138	0.081	2.068	0.835	2.230	0.994	0.658	0.513
31	-330	2.638	0.469	2.680	0.984	0.116	0.069	2.078	0.793	2.224	0.994	0.655	0.515
32	-329	2.656	0.459	2.696	0.982	0.098	0.056	2.072	0.758	2.206	0.995	0.652	0.519
33	-328	2.650	0.446	2.687	0.981	0.103	0.061	2.064	0.721	2.186	0.995	0.660	0.526
34	-327	2.616	0.436	2.652	0.982	0.125	0.076	2.030	0.658	2.134	0.996	0.671	0.539
35	-326	2.563	0.433	2.600	0.982	0.158	0.105	2.016	0.602	2.104	0.994	0.682	0.548
36	-325	2.483	0.416	2.517	0.981	0.194	0.136	1.989	0.554	2.065	0.994	0.695	0.560

37	-324	2.372	0.393	2.404	0.983	0.203	0.151	1.934	0.476	1.991	0.995	0.682	0.562
38	-323	2.284	0.355	2.312	0.981	0.209	0.162	1.871	0.408	1.915	0.994	0.671	0.564
39	-322	2.188	0.338	2.214	0.983	0.211	0.170	1.813	0.335	1.844	0.996	0.660	0.564
40	-321	2.061	0.317	2.086	0.985	0.218	0.181	1.750	0.284	1.773	0.996	0.653	0.566
41	-320	1.982	0.308	2.006	0.983	0.247	0.205	1.726	0.340	1.760	0.996	0.663	0.573
42	-319	1.892	0.324	1.919	0.983	0.277	0.231	1.689	0.406	1.737	0.996	0.674	0.580
43	-318	1.837	0.353	1.870	0.981	0.305	0.254	1.668	0.481	1.736	0.996	0.684	0.587
44	-317	1.776	0.375	1.815	0.982	0.332	0.276	1.640	0.531	1.723	0.995	0.692	0.593
45	-316	1.716	0.388	1.760	0.982	0.372	0.307	1.636	0.550	1.726	0.995	0.701	0.596
46	-315	1.685	0.388	1.729	0.984	0.397	0.328	1.638	0.549	1.727	0.995	0.712	0.601
47	-314	1.668	0.398	1.715	0.984	0.433	0.352	1.643	0.557	1.735	0.995	0.719	0.603
48	-313	1.627	0.412	1.678	0.984	0.454	0.368	1.631	0.570	1.728	0.995	0.729	0.610
49	-312	1.617	0.422	1.671	0.983	0.459	0.373	1.616	0.561	1.710	0.995	0.736	0.617
50	-311	1.586	0.417	1.640	0.983	0.468	0.381	1.597	0.558	1.692	0.995	0.741	0.623
51	-310	1.553	0.417	1.608	0.984	0.470	0.388	1.578	0.557	1.674	0.995	0.750	0.631
52	-309	1.527	0.421	1.584	0.983	0.471	0.389	1.566	0.562	1.664	0.996	0.755	0.635
53	-308	1.474	0.425	1.534	0.984	0.470	0.397	1.538	0.584	1.645	0.996	0.760	0.641
54	-307	1.427	0.446	1.495	0.983	0.470	0.401	1.520	0.611	1.638	0.996	0.767	0.644
55	-306	1.372	0.454	1.445	0.983	0.464	0.402	1.494	0.634	1.622	0.994	0.771	0.647
56	-305	1.350	0.442	1.421	0.985	0.466	0.407	1.477	0.603	1.595	0.996	0.780	0.656
57	-304	1.295	0.420	1.361	0.985	0.473	0.417	1.424	0.559	1.530	0.997	0.790	0.671
58	-303	1.250	0.414	1.316	0.985	0.477	0.427	1.389	0.521	1.483	0.997	0.800	0.683
59	-302	1.197	0.410	1.265	0.984	0.482	0.437	1.333	0.481	1.417	0.996	0.810	0.699
60	-301	1.184	0.411	1.253	0.984	0.482	0.439	1.312	0.479	1.397	0.996	0.812	0.702
61	-300	1.169	0.397	1.235	0.984	0.484	0.441	1.295	0.478	1.381	0.995	0.815	0.705
62	-299	1.139	0.395	1.206	0.984	0.484	0.445	1.274	0.468	1.357	0.995	0.815	0.710
63	-298	1.125	0.408	1.197	0.983	0.486	0.449	1.258	0.471	1.344	0.995	0.816	0.713
64	-297	1.132	0.383	1.195	0.984	0.483	0.447	1.267	0.419	1.334	0.996	0.817	0.715
65	-296	1.107	0.361	1.165	0.984	0.480	0.448	1.253	0.370	1.307	0.997	0.819	0.719
66	-295	1.096	0.345	1.149	0.983	0.479	0.447	1.248	0.323	1.289	0.996	0.822	0.722
67	-294	1.082	0.327	1.131	0.983	0.478	0.450	1.229	0.277	1.260	0.995	0.822	0.726
68	-293	1.052	0.310	1.097	0.984	0.485	0.459	1.187	0.243	1.212	0.996	0.823	0.730
69	-292	1.017	0.293	1.058	0.984	0.493	0.468	1.138	0.209	1.157	0.997	0.823	0.735
70	-291	0.984	0.280	1.023	0.983	0.499	0.476	1.093	0.169	1.106	0.995	0.827	0.741
71	-290	0.971	0.279	1.010	0.984	0.508	0.484	1.068	0.162	1.081	0.995	0.829	0.744
72	-289	0.987	0.285	1.027	0.983	0.514	0.487	1.094	0.169	1.107	0.996	0.827	0.739
73	-288	0.998	0.278	1.036	0.983	0.521	0.490	1.116	0.173	1.129	0.996	0.825	0.736
74	-287	1.028	0.293	1.069	0.983	0.527	0.492	1.146	0.189	1.161	0.995	0.822	0.730
75	-286	1.025	0.301	1.068	0.984	0.529	0.494	1.147	0.179	1.161	0.996	0.821	0.728
76	-285	0.993	0.276	1.030	0.985	0.529	0.495	1.114	0.159	1.126	0.995	0.823	0.731
77	-284	0.964	0.270	1.001	0.985	0.530	0.502	1.079	0.148	1.089	0.994	0.825	0.733
78	-283	0.916	0.274	0.956	0.983	0.530	0.508	1.040	0.143	1.050	0.996	0.822	0.734
79	-282	0.893	0.283	0.937	0.984	0.536	0.516	1.007	0.156	1.019	0.995	0.825	0.739
80	-281	0.900	0.287	0.945	0.983	0.545	0.520	1.017	0.165	1.030	0.995	0.827	0.738
81	-280	0.903	0.303	0.953	0.983	0.556	0.526	1.011	0.184	1.027	0.995	0.831	0.742
82	-279	0.916	0.302	0.964	0.982	0.563	0.530	1.022	0.190	1.039	0.996	0.834	0.743
83	-278	0.883	0.335	0.944	0.984	0.565	0.534	0.973	0.245	1.004	0.996	0.831	0.743
84	-277	0.862	0.350	0.930	0.984	0.568	0.539	0.935	0.296	0.981	0.995	0.826	0.740
85	-276	0.827	0.377	0.909	0.983	0.570	0.543	0.894	0.349	0.960	0.996	0.822	0.738
86	-275	0.802	0.399	0.896	0.984	0.574	0.549	0.848	0.405	0.940	0.996	0.818	0.738
87	-274	0.792	0.422	0.897	0.984	0.570	0.550	0.847	0.434	0.952	0.995	0.819	0.737
88	-273	0.766	0.413	0.870	0.984	0.568	0.551	0.833	0.446	0.945	0.996	0.820	0.739
89	-272	0.751	0.435	0.868	0.983	0.567	0.553	0.828	0.481	0.957	0.996	0.822	0.739
90	-271	0.724	0.443	0.849	0.984	0.565	0.555	0.812	0.500	0.954	0.996	0.822	0.740
91	-270	0.743	0.438	0.863	0.983	0.569	0.556	0.824	0.502	0.964	0.995	0.824	0.742
92	-269	0.740	0.461	0.872	0.982	0.572	0.559	0.833	0.512	0.978	0.995	0.827	0.743
93	-268	0.757	0.456	0.884	0.983	0.574	0.557	0.841	0.514	0.985	0.996	0.829	0.745
94	-267	0.750	0.442	0.870	0.983	0.573	0.558	0.836	0.506	0.977	0.996	0.830	0.746
95	-266	0.737	0.427	0.852	0.984	0.569	0.558	0.809	0.482	0.941	0.996	0.833	0.750
96	-265	0.713	0.418	0.826	0.984	0.564	0.555	0.788	0.465	0.915	0.995	0.833	0.752

97	-264	0.684	0.397	0.791	0.983	0.560	0.558	0.750	0.450	0.875	0.995	0.836	0.757
98	-263	0.661	0.406	0.776	0.984	0.558	0.556	0.722	0.457	0.855	0.995	0.843	0.763
99	-262	0.644	0.418	0.768	0.984	0.566	0.565	0.696	0.470	0.839	0.995	0.848	0.770
100	-261	0.641	0.417	0.765	0.983	0.570	0.570	0.683	0.483	0.836	0.995	0.854	0.775
101	-260	0.640	0.432	0.772	0.983	0.577	0.577	0.666	0.490	0.827	0.996	0.861	0.781
102	-259	0.631	0.446	0.772	0.983	0.579	0.580	0.665	0.498	0.831	0.995	0.864	0.784
103	-258	0.629	0.436	0.765	0.983	0.584	0.585	0.673	0.485	0.830	0.995	0.867	0.786
104	-257	0.644	0.437	0.778	0.982	0.586	0.586	0.683	0.484	0.837	0.995	0.870	0.788
105	-256	0.640	0.436	0.775	0.983	0.592	0.591	0.695	0.476	0.842	0.996	0.871	0.790
106	-255	0.636	0.421	0.763	0.983	0.598	0.597	0.684	0.470	0.830	0.995	0.875	0.795
107	-254	0.633	0.435	0.768	0.984	0.604	0.604	0.672	0.481	0.827	0.995	0.878	0.797
108	-253	0.627	0.430	0.760	0.982	0.611	0.611	0.659	0.479	0.814	0.994	0.879	0.800
109	-252	0.613	0.440	0.755	0.984	0.617	0.616	0.636	0.485	0.800	0.995	0.882	0.803
110	-251	0.608	0.440	0.750	0.983	0.622	0.622	0.624	0.483	0.789	0.995	0.885	0.807
111	-250	0.584	0.453	0.739	0.985	0.627	0.626	0.605	0.490	0.779	0.995	0.887	0.809
112	-249	0.571	0.440	0.720	0.984	0.630	0.629	0.590	0.478	0.759	0.995	0.889	0.813
113	-248	0.562	0.427	0.705	0.983	0.636	0.635	0.576	0.464	0.739	0.995	0.893	0.816
114	-247	0.537	0.406	0.673	0.984	0.638	0.637	0.555	0.430	0.702	0.996	0.897	0.822
115	-246	0.535	0.369	0.650	0.983	0.639	0.640	0.544	0.381	0.664	0.994	0.900	0.827
116	-245	0.515	0.350	0.623	0.984	0.643	0.643	0.523	0.345	0.626	0.995	0.906	0.833
117	-244	0.497	0.347	0.607	0.984	0.649	0.650	0.503	0.329	0.601	0.996	0.907	0.836
118	-243	0.488	0.350	0.601	0.983	0.655	0.655	0.486	0.334	0.590	0.995	0.908	0.838
119	-242	0.481	0.357	0.599	0.984	0.662	0.662	0.479	0.342	0.588	0.995	0.909	0.838
120	-241	0.481	0.354	0.598	0.983	0.668	0.668	0.475	0.338	0.583	0.995	0.909	0.839
121	-240	0.477	0.348	0.591	0.983	0.674	0.674	0.463	0.336	0.572	0.995	0.910	0.841
122	-239	0.474	0.359	0.594	0.984	0.678	0.678	0.463	0.353	0.582	0.995	0.912	0.844
123	-238	0.475	0.359	0.595	0.983	0.683	0.684	0.463	0.358	0.585	0.995	0.912	0.845
124	-237	0.468	0.352	0.586	0.983	0.689	0.689	0.458	0.356	0.580	0.995	0.912	0.848
125	-236	0.480	0.354	0.596	0.983	0.691	0.692	0.462	0.359	0.585	0.995	0.915	0.850
126	-235	0.460	0.354	0.580	0.983	0.692	0.693	0.451	0.365	0.580	0.996	0.919	0.852
127	-234	0.454	0.361	0.580	0.984	0.690	0.690	0.444	0.373	0.580	0.996	0.922	0.855
128	-233	0.454	0.373	0.588	0.984	0.691	0.692	0.435	0.379	0.576	0.996	0.927	0.859
129	-232	0.441	0.381	0.583	0.982	0.686	0.687	0.438	0.390	0.586	0.996	0.927	0.855
130	-231	0.438	0.390	0.586	0.981	0.681	0.683	0.431	0.404	0.590	0.997	0.924	0.853
131	-230	0.454	0.401	0.606	0.984	0.678	0.679	0.449	0.419	0.614	0.996	0.920	0.849
132	-229	0.456	0.407	0.611	0.983	0.673	0.674	0.452	0.417	0.615	0.996	0.917	0.847
133	-228	0.436	0.383	0.581	0.984	0.675	0.675	0.426	0.376	0.568	0.996	0.918	0.851
134	-227	0.422	0.335	0.539	0.984	0.679	0.678	0.393	0.331	0.513	0.995	0.917	0.854
135	-226	0.409	0.311	0.514	0.984	0.681	0.681	0.366	0.301	0.474	0.996	0.918	0.856
136	-225	0.382	0.306	0.489	0.984	0.687	0.687	0.335	0.278	0.435	0.997	0.916	0.856
137	-224	0.376	0.307	0.485	0.985	0.693	0.693	0.332	0.270	0.428	0.995	0.918	0.858
138	-223	0.380	0.289	0.477	0.984	0.703	0.704	0.329	0.253	0.415	0.996	0.922	0.861
139	-222	0.367	0.288	0.467	0.985	0.711	0.711	0.329	0.254	0.416	0.995	0.924	0.862
140	-221	0.375	0.294	0.476	0.984	0.718	0.717	0.336	0.256	0.422	0.995	0.925	0.863
141	-220	0.377	0.284	0.472	0.983	0.720	0.720	0.355	0.258	0.439	0.995	0.928	0.865
142	-219	0.379	0.292	0.479	0.983	0.727	0.727	0.371	0.267	0.457	0.995	0.930	0.867
143	-218	0.379	0.288	0.477	0.984	0.730	0.730	0.384	0.269	0.469	0.996	0.932	0.868
144	-217	0.367	0.288	0.467	0.984	0.740	0.740	0.385	0.269	0.470	0.996	0.936	0.872
145	-216	0.369	0.276	0.461	0.984	0.755	0.755	0.385	0.259	0.464	0.995	0.939	0.875
146	-215	0.376	0.260	0.457	0.983	0.771	0.771	0.387	0.246	0.459	0.996	0.945	0.880
147	-214	0.363	0.243	0.437	0.983	0.783	0.784	0.379	0.231	0.444	0.996	0.949	0.884
148	-213	0.353	0.247	0.430	0.984	0.783	0.783	0.350	0.233	0.421	0.995	0.951	0.887
149	-212	0.354	0.248	0.433	0.984	0.780	0.780	0.328	0.235	0.404	0.995	0.952	0.889
150	-211	0.361	0.239	0.433	0.983	0.774	0.774	0.315	0.220	0.384	0.996	0.952	0.892
151	-210	0.371	0.231	0.437	0.983	0.771	0.771	0.307	0.207	0.370	0.996	0.951	0.892
152	-209	0.364	0.210	0.420	0.982	0.769	0.770	0.309	0.180	0.357	0.996	0.951	0.891
153	-208	0.364	0.200	0.416	0.983	0.769	0.769	0.318	0.158	0.355	0.996	0.950	0.891
154	-207	0.356	0.194	0.405	0.984	0.767	0.767	0.316	0.141	0.346	0.995	0.949	0.890
155	-206	0.350	0.185	0.396	0.985	0.766	0.765	0.322	0.126	0.345	0.995	0.948	0.889
156	-205	0.332	0.168	0.372	0.985	0.767	0.766	0.292	0.119	0.315	0.995	0.945	0.888

157	-204	0.314	0.154	0.349	0.982	0.770	0.768	0.263	0.101	0.281	0.995	0.944	0.889
158	-203	0.310	0.138	0.339	0.985	0.772	0.771	0.247	0.085	0.261	0.995	0.942	0.887
159	-202	0.300	0.132	0.328	0.984	0.774	0.775	0.234	0.082	0.248	0.995	0.941	0.887
160	-201	0.317	0.125	0.341	0.984	0.779	0.779	0.255	0.079	0.267	0.995	0.942	0.888
161	-200	0.323	0.118	0.343	0.983	0.785	0.784	0.265	0.078	0.276	0.996	0.943	0.888
162	-199	0.325	0.121	0.347	0.985	0.790	0.789	0.270	0.082	0.282	0.995	0.944	0.889
163	-198	0.311	0.109	0.329	0.983	0.796	0.795	0.266	0.080	0.278	0.995	0.945	0.890
164	-197	0.322	0.120	0.344	0.983	0.799	0.798	0.277	0.090	0.292	0.996	0.946	0.890
165	-196	0.305	0.106	0.323	0.983	0.804	0.803	0.273	0.084	0.286	0.997	0.947	0.891
166	-195	0.303	0.125	0.328	0.983	0.808	0.807	0.282	0.099	0.299	0.995	0.946	0.890
167	-194	0.287	0.121	0.312	0.983	0.814	0.812	0.272	0.097	0.289	0.996	0.947	0.892
168	-193	0.284	0.119	0.308	0.985	0.818	0.817	0.258	0.094	0.274	0.995	0.949	0.896
169	-192	0.274	0.102	0.293	0.984	0.826	0.826	0.240	0.082	0.253	0.995	0.950	0.899
170	-191	0.259	0.094	0.275	0.983	0.834	0.833	0.225	0.067	0.234	0.995	0.953	0.903
171	-190	0.265	0.081	0.277	0.984	0.834	0.833	0.229	0.056	0.235	0.997	0.952	0.902
172	-189	0.265	0.061	0.272	0.984	0.830	0.832	0.232	0.041	0.236	0.995	0.951	0.900
173	-188	0.281	0.062	0.288	0.985	0.827	0.829	0.238	0.031	0.240	0.996	0.952	0.899
174	-187	0.299	0.053	0.303	0.985	0.825	0.825	0.249	0.024	0.250	0.996	0.950	0.898
175	-186	0.286	0.062	0.292	0.984	0.824	0.825	0.220	0.020	0.221	0.997	0.948	0.897
176	-185	0.268	0.058	0.274	0.983	0.821	0.822	0.202	0.020	0.203	0.996	0.947	0.895
177	-184	0.262	0.047	0.266	0.983	0.820	0.820	0.183	0.007	0.183	0.996	0.944	0.894
178	-183	0.252	0.048	0.257	0.984	0.819	0.817	0.168	0.009	0.168	0.996	0.943	0.893
179	-182	0.251	0.048	0.256	0.985	0.830	0.830	0.194	0.006	0.194	0.996	0.946	0.896
180	-181	0.258	0.042	0.262	0.984	0.842	0.841	0.217	-0.002	0.217	0.995	0.949	0.900
181	-180	0.246	0.050	0.251	0.985	0.855	0.856	0.233	0.002	0.233	0.996	0.951	0.902
182	-179	0.229	0.043	0.233	0.984	0.864	0.865	0.239	-0.012	0.239	0.996	0.951	0.906
183	-178	0.222	0.043	0.226	0.984	0.870	0.871	0.215	-0.019	0.216	0.996	0.953	0.908
184	-177	0.203	0.041	0.207	0.984	0.877	0.878	0.193	-0.027	0.195	0.996	0.956	0.911
185	-176	0.193	0.040	0.197	0.985	0.881	0.885	0.176	-0.032	0.179	0.996	0.957	0.914
186	-175	0.171	0.041	0.176	0.983	0.885	0.888	0.156	-0.034	0.160	0.997	0.957	0.914
187	-174	0.149	0.030	0.152	0.983	0.883	0.885	0.123	-0.045	0.130	0.997	0.953	0.912
188	-173	0.147	0.022	0.148	0.983	0.878	0.880	0.100	-0.049	0.111	0.997	0.950	0.909
189	-172	0.133	-0.002	0.133	0.984	0.875	0.876	0.074	-0.062	0.097	0.995	0.946	0.906
190	-171	0.133	-0.006	0.133	0.985	0.875	0.876	0.069	-0.073	0.100	0.995	0.944	0.903
191	-170	0.139	-0.023	0.141	0.985	0.875	0.874	0.088	-0.095	0.129	0.995	0.942	0.899
192	-169	0.140	-0.026	0.142	0.985	0.876	0.875	0.100	-0.105	0.145	0.994	0.941	0.896
193	-168	0.142	-0.041	0.148	0.985	0.875	0.873	0.131	-0.120	0.178	0.995	0.933	0.890
194	-167	0.132	-0.053	0.143	0.985	0.879	0.876	0.132	-0.127	0.183	0.995	0.932	0.890
195	-166	0.134	-0.031	0.138	0.984	0.884	0.882	0.135	-0.119	0.180	0.996	0.934	0.890
196	-165	0.122	-0.038	0.128	0.984	0.885	0.883	0.138	-0.126	0.187	0.995	0.935	0.890
197	-164	0.110	-0.041	0.117	0.984	0.887	0.887	0.133	-0.135	0.190	0.995	0.936	0.891
198	-163	0.110	-0.060	0.125	0.983	0.889	0.890	0.116	-0.142	0.184	0.995	0.938	0.894
199	-162	0.103	-0.060	0.119	0.984	0.890	0.891	0.091	-0.152	0.177	0.995	0.939	0.895
200	-161	0.087	-0.058	0.105	0.984	0.892	0.894	0.064	-0.157	0.170	0.995	0.939	0.897
201	-160	0.076	-0.063	0.099	0.984	0.897	0.897	0.042	-0.156	0.162	0.995	0.939	0.898
202	-159	0.066	-0.083	0.106	0.984	0.897	0.896	0.044	-0.181	0.186	0.995	0.940	0.899
203	-158	0.070	-0.084	0.110	0.985	0.899	0.898	0.055	-0.193	0.201	0.995	0.941	0.901
204	-157	0.086	-0.101	0.133	0.983	0.901	0.899	0.074	-0.214	0.226	0.996	0.942	0.901
205	-156	0.085	-0.104	0.134	0.984	0.903	0.901	0.085	-0.225	0.240	0.996	0.942	0.900
206	-155	0.068	-0.109	0.128	0.984	0.904	0.901	0.072	-0.225	0.236	0.996	0.939	0.897
207	-154	0.072	-0.108	0.130	0.985	0.905	0.901	0.074	-0.218	0.230	0.995	0.937	0.896
208	-153	0.076	-0.074	0.106	0.984	0.905	0.904	0.075	-0.202	0.216	0.995	0.935	0.894
209	-152	0.072	-0.088	0.113	0.983	0.905	0.904	0.073	-0.201	0.214	0.996	0.935	0.895
210	-151	0.068	-0.104	0.124	0.985	0.907	0.905	0.065	-0.210	0.219	0.995	0.937	0.897
211	-150	0.079	-0.097	0.125	0.985	0.906	0.907	0.070	-0.201	0.212	0.996	0.938	0.898
212	-149	0.093	-0.080	0.123	0.984	0.908	0.908	0.077	-0.190	0.205	0.997	0.940	0.899
213	-148	0.110	-0.089	0.141	0.985	0.908	0.909	0.090	-0.193	0.213	0.996	0.940	0.900
214	-147	0.102	-0.096	0.140	0.984	0.910	0.910	0.094	-0.202	0.222	0.996	0.939	0.898
215	-146	0.112	-0.112	0.158	0.984	0.911	0.909	0.117	-0.213	0.243	0.996	0.938	0.897
216	-145	0.109	-0.111	0.156	0.983	0.912	0.911	0.127	-0.222	0.255	0.997	0.936	0.895

217	-144	0.085	-0.099	0.130	0.983	0.913	0.914	0.106	-0.217	0.242	0.996	0.936	0.896
218	-143	0.079	-0.101	0.128	0.983	0.918	0.914	0.086	-0.219	0.235	0.996	0.938	0.897
219	-142	0.063	-0.107	0.124	0.984	0.921	0.916	0.072	-0.219	0.231	0.996	0.939	0.898
220	-141	0.048	-0.104	0.114	0.984	0.924	0.918	0.053	-0.218	0.224	0.996	0.940	0.900
221	-140	0.043	-0.103	0.112	0.983	0.924	0.919	0.049	-0.224	0.230	0.996	0.941	0.899
222	-139	0.030	-0.098	0.102	0.983	0.921	0.918	0.045	-0.226	0.231	0.995	0.939	0.898
223	-138	0.021	-0.088	0.090	0.983	0.920	0.916	0.041	-0.231	0.235	0.995	0.938	0.897
224	-137	0.011	-0.093	0.094	0.983	0.918	0.914	0.032	-0.238	0.240	0.997	0.936	0.896
225	-136	0.004	-0.106	0.106	0.984	0.921	0.915	0.025	-0.250	0.251	0.996	0.936	0.895
226	-135	-0.006	-0.110	0.110	0.983	0.923	0.914	0.024	-0.267	0.268	0.996	0.935	0.893
227	-134	-0.003	-0.142	0.142	0.983	0.927	0.913	0.024	-0.291	0.292	0.996	0.935	0.891
228	-133	0.002	-0.159	0.159	0.984	0.931	0.910	0.036	-0.313	0.315	0.996	0.934	0.889
229	-132	0.011	-0.190	0.190	0.984	0.933	0.907	0.029	-0.354	0.355	0.995	0.936	0.887
230	-131	0.007	-0.230	0.230	0.984	0.932	0.898	0.008	-0.394	0.394	0.994	0.938	0.885
231	-130	0.002	-0.251	0.251	0.985	0.932	0.891	-0.007	-0.418	0.418	0.995	0.940	0.885
232	-129	0.009	-0.262	0.263	0.983	0.934	0.891	-0.009	-0.428	0.428	0.995	0.941	0.884
233	-128	-0.019	-0.281	0.281	0.983	0.936	0.884	-0.020	-0.435	0.436	0.996	0.940	0.884
234	-127	-0.014	-0.279	0.279	0.984	0.938	0.888	-0.021	-0.432	0.433	0.996	0.938	0.885
235	-126	-0.027	-0.289	0.290	0.984	0.939	0.885	-0.023	-0.435	0.435	0.997	0.938	0.886
236	-125	-0.017	-0.287	0.287	0.985	0.939	0.888	-0.013	-0.429	0.429	0.996	0.938	0.885
237	-124	-0.032	-0.280	0.282	0.984	0.940	0.887	-0.010	-0.430	0.430	0.996	0.939	0.885
238	-123	-0.030	-0.266	0.267	0.984	0.939	0.891	0.001	-0.429	0.429	0.996	0.937	0.883
239	-122	-0.048	-0.249	0.253	0.983	0.938	0.893	-0.007	-0.422	0.422	0.996	0.936	0.882
240	-121	-0.038	-0.238	0.241	0.984	0.937	0.895	0.000	-0.419	0.419	0.997	0.934	0.882
241	-120	-0.019	-0.244	0.244	0.985	0.937	0.896	0.014	-0.426	0.426	0.996	0.932	0.880
242	-119	-0.012	-0.259	0.259	0.984	0.936	0.893	0.021	-0.432	0.432	0.996	0.930	0.878
243	-118	-0.019	-0.253	0.254	0.984	0.937	0.893	0.020	-0.431	0.431	0.996	0.929	0.877
244	-117	-0.039	-0.259	0.262	0.985	0.937	0.891	-0.001	-0.427	0.427	0.995	0.930	0.879
245	-116	-0.066	-0.268	0.276	0.984	0.936	0.886	-0.025	-0.431	0.431	0.997	0.932	0.880
246	-115	-0.088	-0.262	0.277	0.984	0.939	0.886	-0.041	-0.427	0.429	0.996	0.934	0.880
247	-114	-0.104	-0.270	0.290	0.983	0.939	0.882	-0.067	-0.429	0.434	0.996	0.932	0.881
248	-113	-0.110	-0.289	0.309	0.984	0.939	0.878	-0.064	-0.452	0.456	0.997	0.931	0.878
249	-112	-0.115	-0.299	0.321	0.983	0.938	0.876	-0.074	-0.466	0.472	0.997	0.930	0.874
250	-111	-0.124	-0.301	0.325	0.984	0.939	0.875	-0.089	-0.471	0.479	0.995	0.930	0.874
251	-110	-0.123	-0.292	0.317	0.984	0.937	0.875	-0.089	-0.475	0.483	0.997	0.929	0.873
252	-109	-0.118	-0.294	0.317	0.984	0.936	0.875	-0.096	-0.477	0.487	0.996	0.932	0.875
253	-108	-0.103	-0.286	0.304	0.985	0.936	0.878	-0.095	-0.473	0.483	0.997	0.935	0.878
254	-107	-0.087	-0.271	0.284	0.985	0.933	0.881	-0.091	-0.464	0.473	0.996	0.937	0.882
255	-106	-0.081	-0.277	0.289	0.984	0.933	0.881	-0.091	-0.461	0.470	0.996	0.937	0.882
256	-105	-0.079	-0.265	0.277	0.984	0.936	0.885	-0.080	-0.436	0.444	0.996	0.938	0.885
257	-104	-0.071	-0.247	0.257	0.985	0.936	0.888	-0.072	-0.417	0.423	0.996	0.938	0.885
258	-103	-0.048	-0.250	0.255	0.984	0.936	0.891	-0.056	-0.403	0.407	0.996	0.937	0.887
259	-102	-0.065	-0.227	0.236	0.983	0.936	0.895	-0.061	-0.383	0.388	0.996	0.938	0.888
260	-101	-0.068	-0.223	0.233	0.984	0.938	0.895	-0.070	-0.373	0.379	0.997	0.940	0.891
261	-100	-0.072	-0.229	0.241	0.985	0.940	0.895	-0.082	-0.367	0.376	0.996	0.940	0.892
262	-99	-0.070	-0.215	0.227	0.985	0.939	0.898	-0.089	-0.348	0.359	0.996	0.941	0.893
263	-98	-0.071	-0.207	0.219	0.984	0.939	0.899	-0.082	-0.347	0.356	0.996	0.940	0.893
264	-97	-0.073	-0.209	0.222	0.984	0.939	0.899	-0.066	-0.356	0.362	0.995	0.939	0.893
265	-96	-0.076	-0.216	0.229	0.983	0.941	0.898	-0.059	-0.360	0.365	0.995	0.937	0.891
266	-95	-0.058	-0.223	0.230	0.984	0.940	0.899	-0.032	-0.371	0.372	0.996	0.935	0.889
267	-94	-0.054	-0.239	0.245	0.985	0.937	0.895	-0.023	-0.387	0.388	0.996	0.935	0.887
268	-93	-0.061	-0.258	0.265	0.984	0.938	0.891	-0.022	-0.400	0.401	0.996	0.933	0.885
269	-92	-0.058	-0.266	0.272	0.985	0.937	0.889	-0.016	-0.411	0.411	0.995	0.930	0.883
270	-91	-0.055	-0.266	0.271	0.983	0.938	0.889	-0.012	-0.410	0.410	0.996	0.929	0.882
271	-90	-0.058	-0.264	0.270	0.984	0.938	0.890	-0.018	-0.410	0.411	0.996	0.930	0.884
272	-89	-0.068	-0.267	0.276	0.983	0.939	0.890	-0.026	-0.416	0.416	0.996	0.931	0.884
273	-88	-0.066	-0.262	0.270	0.983	0.939	0.891	-0.036	-0.413	0.414	0.996	0.933	0.884
274	-87	-0.070	-0.257	0.266	0.985	0.940	0.891	-0.039	-0.420	0.422	0.995	0.935	0.885
275	-86	-0.084	-0.271	0.284	0.982	0.936	0.887	-0.053	-0.417	0.420	0.994	0.936	0.887
276	-85	-0.091	-0.259	0.275	0.984	0.936	0.887	-0.079	-0.406	0.413	0.995	0.931	0.885

277	-84	-0.107	-0.260	0.281	0.984	0.935	0.886	-0.112	-0.405	0.420	0.995	0.927	0.882
278	-83	-0.122	-0.248	0.277	0.983	0.935	0.885	-0.134	-0.398	0.419	0.995	0.925	0.879
279	-82	-0.108	-0.257	0.279	0.984	0.931	0.883	-0.130	-0.401	0.422	0.995	0.922	0.876
280	-81	-0.119	-0.256	0.282	0.984	0.928	0.882	-0.136	-0.400	0.423	0.996	0.919	0.875
281	-80	-0.119	-0.256	0.282	0.983	0.925	0.880	-0.140	-0.397	0.421	0.996	0.916	0.873
282	-79	-0.123	-0.261	0.288	0.983	0.923	0.877	-0.136	-0.403	0.425	0.996	0.914	0.871
283	-78	-0.115	-0.243	0.269	0.983	0.921	0.878	-0.122	-0.406	0.424	0.996	0.914	0.871
284	-77	-0.108	-0.247	0.269	0.984	0.919	0.875	-0.111	-0.413	0.428	0.996	0.913	0.870
285	-76	-0.114	-0.230	0.257	0.984	0.917	0.874	-0.104	-0.412	0.425	0.996	0.912	0.870
286	-75	-0.117	-0.243	0.270	0.984	0.916	0.871	-0.113	-0.418	0.433	0.996	0.913	0.870
287	-74	-0.124	-0.231	0.263	0.984	0.918	0.875	-0.127	-0.405	0.425	0.996	0.913	0.871
288	-73	-0.137	-0.235	0.272	0.985	0.915	0.874	-0.147	-0.400	0.426	0.996	0.916	0.873
289	-72	-0.163	-0.234	0.286	0.986	0.913	0.872	-0.176	-0.396	0.434	0.996	0.915	0.873
290	-71	-0.164	-0.221	0.275	0.985	0.915	0.875	-0.178	-0.373	0.413	0.996	0.914	0.874
291	-70	-0.143	-0.204	0.249	0.986	0.916	0.880	-0.166	-0.350	0.387	0.996	0.915	0.877
292	-69	-0.119	-0.187	0.222	0.985	0.921	0.887	-0.160	-0.331	0.367	0.996	0.915	0.878
293	-68	-0.122	-0.161	0.202	0.984	0.925	0.894	-0.166	-0.303	0.345	0.996	0.914	0.879
294	-67	-0.116	-0.153	0.192	0.985	0.925	0.894	-0.151	-0.296	0.332	0.995	0.915	0.880
295	-66	-0.096	-0.174	0.199	0.985	0.924	0.892	-0.129	-0.299	0.326	0.995	0.915	0.882
296	-65	-0.087	-0.161	0.182	0.985	0.927	0.897	-0.114	-0.293	0.314	0.996	0.917	0.884
297	-64	-0.082	-0.157	0.177	0.985	0.927	0.898	-0.103	-0.288	0.305	0.996	0.918	0.884
298	-63	-0.070	-0.171	0.185	0.984	0.925	0.895	-0.089	-0.295	0.308	0.996	0.920	0.885
299	-62	-0.072	-0.180	0.194	0.985	0.925	0.893	-0.088	-0.302	0.314	0.996	0.922	0.888
300	-61	-0.073	-0.174	0.189	0.986	0.926	0.894	-0.082	-0.301	0.312	0.996	0.925	0.891
301	-60	-0.083	-0.182	0.200	0.985	0.923	0.891	-0.093	-0.308	0.322	0.996	0.926	0.891
302	-59	-0.104	-0.196	0.221	0.986	0.922	0.887	-0.114	-0.318	0.338	0.995	0.926	0.889
303	-58	-0.106	-0.202	0.228	0.986	0.920	0.885	-0.126	-0.325	0.348	0.996	0.923	0.887
304	-57	-0.095	-0.212	0.233	0.985	0.918	0.882	-0.135	-0.333	0.360	0.996	0.922	0.886
305	-56	-0.113	-0.211	0.239	0.984	0.917	0.882	-0.146	-0.342	0.372	0.995	0.923	0.885
306	-55	-0.124	-0.234	0.265	0.984	0.918	0.878	-0.156	-0.361	0.393	0.996	0.924	0.885
307	-54	-0.134	-0.209	0.249	0.985	0.918	0.882	-0.170	-0.350	0.389	0.995	0.926	0.886
308	-53	-0.144	-0.195	0.242	0.983	0.919	0.885	-0.177	-0.342	0.385	0.997	0.927	0.889
309	-52	-0.143	-0.201	0.247	0.985	0.919	0.882	-0.185	-0.341	0.388	0.995	0.925	0.887
310	-51	-0.145	-0.181	0.232	0.985	0.917	0.886	-0.184	-0.324	0.373	0.996	0.922	0.887
311	-50	-0.156	-0.177	0.236	0.985	0.917	0.886	-0.193	-0.315	0.369	0.996	0.918	0.885
312	-49	-0.154	-0.178	0.236	0.984	0.914	0.883	-0.195	-0.308	0.365	0.995	0.916	0.884
313	-48	-0.143	-0.172	0.224	0.984	0.914	0.885	-0.196	-0.304	0.361	0.996	0.914	0.883
314	-47	-0.142	-0.183	0.232	0.985	0.917	0.887	-0.208	-0.297	0.363	0.995	0.912	0.882
315	-46	-0.126	-0.173	0.213	0.983	0.919	0.889	-0.220	-0.286	0.360	0.995	0.910	0.880
316	-45	-0.109	-0.187	0.216	0.984	0.921	0.889	-0.229	-0.287	0.367	0.995	0.907	0.877
317	-44	-0.121	-0.185	0.221	0.984	0.921	0.889	-0.217	-0.285	0.358	0.995	0.911	0.882
318	-43	-0.133	-0.150	0.200	0.983	0.920	0.894	-0.211	-0.265	0.339	0.996	0.914	0.886
319	-42	-0.143	-0.137	0.198	0.983	0.920	0.897	-0.209	-0.255	0.330	0.996	0.917	0.888
320	-41	-0.150	-0.131	0.199	0.984	0.920	0.896	-0.214	-0.255	0.333	0.995	0.920	0.890
321	-40	-0.158	-0.116	0.196	0.983	0.920	0.898	-0.226	-0.243	0.332	0.995	0.923	0.894
322	-39	-0.172	-0.111	0.205	0.983	0.918	0.898	-0.240	-0.240	0.340	0.996	0.922	0.895
323	-38	-0.178	-0.099	0.203	0.984	0.916	0.899	-0.246	-0.226	0.334	0.995	0.921	0.896
324	-37	-0.181	-0.095	0.205	0.984	0.917	0.901	-0.242	-0.223	0.329	0.995	0.921	0.898
325	-36	-0.181	-0.098	0.206	0.984	0.918	0.901	-0.241	-0.220	0.327	0.995	0.920	0.898
326	-35	-0.165	-0.093	0.189	0.985	0.918	0.902	-0.232	-0.205	0.310	0.994	0.919	0.897
327	-34	-0.148	-0.093	0.175	0.985	0.917	0.902	-0.219	-0.197	0.295	0.995	0.916	0.895
328	-33	-0.140	-0.077	0.160	0.984	0.916	0.905	-0.215	-0.187	0.285	0.996	0.915	0.895
329	-32	-0.141	-0.072	0.158	0.984	0.917	0.905	-0.202	-0.184	0.274	0.995	0.915	0.896
330	-31	-0.145	-0.074	0.163	0.983	0.917	0.907	-0.195	-0.188	0.271	0.996	0.914	0.896
331	-30	-0.135	-0.068	0.151	0.983	0.916	0.906	-0.179	-0.189	0.260	0.995	0.915	0.897
332	-29	-0.138	-0.058	0.150	0.983	0.918	0.908	-0.177	-0.183	0.255	0.996	0.917	0.898
333	-28	-0.131	-0.070	0.148	0.983	0.918	0.907	-0.172	-0.189	0.256	0.996	0.920	0.899
334	-27	-0.134	-0.053	0.144	0.982	0.917	0.909	-0.174	-0.180	0.250	0.997	0.922	0.900
335	-26	-0.141	-0.056	0.152	0.983	0.918	0.909	-0.174	-0.183	0.253	0.996	0.924	0.902
336	-25	-0.140	-0.043	0.146	0.984	0.919	0.912	-0.182	-0.166	0.246	0.996	0.923	0.901

337	-24	-0.134	-0.024	0.136	0.984	0.921	0.917	-0.188	-0.151	0.241	0.996	0.921	0.901
338	-23	-0.121	-0.022	0.123	0.984	0.922	0.917	-0.193	-0.144	0.241	0.995	0.918	0.898
339	-22	-0.131	-0.013	0.131	0.985	0.922	0.920	-0.215	-0.134	0.253	0.995	0.914	0.896
340	-21	-0.117	-0.013	0.118	0.984	0.923	0.920	-0.186	-0.130	0.227	0.996	0.918	0.900
341	-20	-0.101	-0.013	0.102	0.983	0.923	0.920	-0.156	-0.126	0.201	0.996	0.920	0.902
342	-19	-0.090	-0.002	0.090	0.983	0.923	0.923	-0.132	-0.121	0.179	0.996	0.922	0.905
343	-18	-0.084	-0.009	0.084	0.983	0.924	0.923	-0.117	-0.125	0.171	0.996	0.925	0.908
344	-17	-0.095	-0.019	0.097	0.983	0.924	0.922	-0.131	-0.128	0.183	0.996	0.926	0.909
345	-16	-0.097	-0.027	0.100	0.983	0.926	0.921	-0.140	-0.129	0.191	0.997	0.929	0.911
346	-15	-0.104	-0.036	0.110	0.983	0.926	0.919	-0.158	-0.130	0.205	0.996	0.930	0.911
347	-14	-0.115	-0.042	0.122	0.984	0.924	0.917	-0.176	-0.128	0.218	0.995	0.933	0.912
348	-13	-0.142	-0.042	0.148	0.985	0.924	0.916	-0.203	-0.128	0.240	0.995	0.933	0.917
349	-12	-0.160	-0.042	0.166	0.984	0.922	0.912	-0.228	-0.122	0.258	0.994	0.933	0.920
350	-11	-0.187	-0.041	0.192	0.987	0.919	0.910	-0.254	-0.118	0.280	0.992	0.932	0.923
351	-10	-0.205	-0.036	0.208	0.990	0.917	0.909	-0.270	-0.110	0.291	0.990	0.929	0.925
352	-9	-0.222	-0.017	0.222	0.994	0.916	0.911	-0.282	-0.094	0.297	0.983	0.928	0.925
353	-8	-1.000	-1.000	-1.000	-1.000	0.916	-1.000	-1.000	-1.000	-1.000	-1.000	0.927	-1.000
354	-7	-1.000	-1.000	-1.000	-1.000	0.914	-1.000	-1.000	-1.000	-1.000	-1.000	0.926	-1.000
355	-6	-1.000	-1.000	-1.000	-1.000	0.912	-1.000	-1.000	-1.000	-1.000	-1.000	0.923	-1.000
356	-5	-1.000	-1.000	-1.000	-1.000	0.911	-1.000	-1.000	-1.000	-1.000	-1.000	0.921	-1.000
357	-4	-1.000	-1.000	-1.000	-1.000	0.909	-1.000	-1.000	-1.000	-1.000	-1.000	0.919	-1.000
358	-3	-1.000	-1.000	-1.000	-1.000	0.908	-1.000	-1.000	-1.000	-1.000	-1.000	0.916	-1.000
359	-2	-1.000	-1.000	-1.000	-1.000	0.908	-1.000	-1.000	-1.000	-1.000	-1.000	0.915	-1.000
360	-1	-1.000	-1.000	-1.000	-1.000	0.909	-1.000	-1.000	-1.000	-1.000	-1.000	0.913	-1.000
361	0	-1.000	-1.000	-1.000	-1.000	0.911	-1.000	-1.000	-1.000	-1.000	-1.000	0.910	-1.000
362	1	-1.000	-1.000	-1.000	-1.000	0.910	-1.000	-1.000	-1.000	-1.000	-1.000	0.904	-1.000
363	2	-1.000	-1.000	-1.000	-1.000	0.910	-1.000	-1.000	-1.000	-1.000	-1.000	0.901	-1.000
364	3	-1.000	-1.000	-1.000	-1.000	0.909	-1.000	-1.000	-1.000	-1.000	-1.000	0.896	-1.000
365	4	-1.000	-1.000	-1.000	-1.000	0.907	-1.000	-1.000	-1.000	-1.000	-1.000	0.893	-1.000
366	5	-1.000	-1.000	-1.000	-1.000	0.905	-1.000	-1.000	-1.000	-1.000	-1.000	0.891	-1.000
367	6	-1.000	-1.000	-1.000	-1.000	0.907	-1.000	-1.000	-1.000	-1.000	-1.000	0.897	-1.000
368	7	-1.000	-1.000	-1.000	-1.000	0.908	-1.000	-1.000	-1.000	-1.000	-1.000	0.901	-1.000
369	8	-1.000	-1.000	-1.000	-1.000	0.908	-1.000	-1.000	-1.000	-1.000	-1.000	0.903	-1.000
370	9	-1.000	-1.000	-1.000	-1.000	0.908	-1.000	-1.000	-1.000	-1.000	-1.000	0.906	-1.000
371	10	-1.000	-1.000	-1.000	-1.000	0.915	-1.000	-1.000	-1.000	-1.000	-1.000	0.911	-1.000
372	11	-1.000	-1.000	-1.000	-1.000	0.920	-1.000	-1.000	-1.000	-1.000	-1.000	0.915	-1.000
373	12	-1.000	-1.000	-1.000	-1.000	0.927	-1.000	-1.000	-1.000	-1.000	-1.000	0.919	-1.000
374	13	-0.251	0.124	0.280	0.991	0.928	0.917	-0.251	0.094	0.268	0.987	0.921	0.940
375	14	-0.218	0.117	0.247	0.986	0.930	0.923	-0.223	0.097	0.243	0.994	0.923	0.944
376	15	-0.189	0.112	0.220	0.984	0.929	0.925	-0.193	0.098	0.217	0.996	0.924	0.946
377	16	-0.156	0.108	0.190	0.983	0.927	0.927	-0.163	0.103	0.193	0.997	0.926	0.946
378	17	-0.144	0.107	0.180	0.982	0.927	0.928	-0.160	0.104	0.191	0.998	0.926	0.944
379	18	-0.137	0.107	0.173	0.983	0.927	0.930	-0.168	0.103	0.197	0.999	0.922	0.942
380	19	-0.128	0.106	0.166	0.982	0.926	0.929	-0.178	0.101	0.204	0.998	0.918	0.939
381	20	-0.120	0.101	0.157	0.981	0.927	0.930	-0.186	0.094	0.209	0.997	0.914	0.935
382	21	-0.131	0.093	0.160	0.982	0.927	0.926	-0.205	0.086	0.222	0.997	0.913	0.936
383	22	-0.150	0.087	0.173	0.982	0.924	0.924	-0.228	0.080	0.241	0.997	0.912	0.939
384	23	-0.172	0.086	0.192	0.982	0.923	0.924	-0.255	0.079	0.267	0.997	0.910	0.938
385	24	-0.188	0.088	0.208	0.982	0.919	0.917	-0.276	0.085	0.288	0.996	0.906	0.938
386	25	-0.184	0.108	0.214	0.982	0.919	0.917	-0.268	0.106	0.288	0.997	0.910	0.939
387	26	-0.181	0.121	0.218	0.983	0.919	0.917	-0.255	0.128	0.286	0.998	0.914	0.939
388	27	-0.169	0.134	0.216	0.983	0.919	0.917	-0.239	0.148	0.281	0.997	0.915	0.938
389	28	-0.161	0.151	0.220	0.982	0.919	0.915	-0.226	0.171	0.283	0.996	0.915	0.937
390	29	-0.146	0.161	0.218	0.982	0.917	0.914	-0.218	0.192	0.290	0.997	0.910	0.932
391	30	-0.133	0.177	0.221	0.983	0.914	0.910	-0.210	0.214	0.300	0.996	0.906	0.928
392	31	-0.129	0.193	0.233	0.984	0.913	0.908	-0.210	0.239	0.318	0.996	0.900	0.922
393	32	-0.124	0.203	0.238	0.984	0.911	0.906	-0.205	0.254	0.326	0.996	0.896	0.918
394	33	-0.125	0.209	0.243	0.983	0.910	0.906	-0.204	0.255	0.327	0.996	0.896	0.920
395	34	-0.122	0.212	0.245	0.983	0.910	0.905	-0.204	0.259	0.329	0.997	0.897	0.920
396	35	-0.122	0.224	0.255	0.984	0.907	0.903	-0.203	0.267	0.335	0.996	0.897	0.921

397	36	-0.134	0.239	0.274	0.983	0.909	0.900	-0.214	0.273	0.347	0.997	0.896	0.919
398	37	-0.154	0.255	0.298	0.983	0.911	0.898	-0.229	0.287	0.367	0.996	0.896	0.917
399	38	-0.168	0.265	0.314	0.985	0.914	0.893	-0.245	0.294	0.383	0.996	0.894	0.914
400	39	-0.195	0.272	0.335	0.983	0.915	0.889	-0.260	0.297	0.395	0.995	0.893	0.912
401	40	-0.183	0.279	0.333	0.983	0.914	0.886	-0.256	0.298	0.393	0.995	0.892	0.912
402	41	-0.181	0.283	0.336	0.984	0.912	0.885	-0.248	0.304	0.393	0.995	0.891	0.911
403	42	-0.182	0.292	0.344	0.984	0.909	0.882	-0.248	0.313	0.399	0.996	0.892	0.910
404	43	-0.178	0.298	0.348	0.984	0.908	0.879	-0.244	0.320	0.402	0.996	0.892	0.910
405	44	-0.175	0.294	0.342	0.983	0.906	0.879	-0.236	0.314	0.393	0.997	0.891	0.912
406	45	-0.175	0.279	0.329	0.985	0.903	0.880	-0.232	0.301	0.379	0.996	0.891	0.916
407	46	-0.170	0.273	0.321	0.983	0.901	0.880	-0.221	0.297	0.371	0.996	0.891	0.917
408	47	-0.165	0.263	0.310	0.984	0.899	0.880	-0.213	0.290	0.360	0.997	0.890	0.919
409	48	-0.172	0.271	0.321	0.984	0.897	0.877	-0.222	0.290	0.365	0.996	0.891	0.920
410	49	-0.180	0.261	0.317	0.983	0.895	0.875	-0.230	0.284	0.365	0.996	0.891	0.920
411	50	-0.195	0.259	0.324	0.983	0.894	0.871	-0.249	0.284	0.378	0.996	0.890	0.920
412	51	-0.211	0.264	0.338	0.983	0.892	0.868	-0.257	0.288	0.386	0.995	0.889	0.918
413	52	-0.192	0.259	0.322	0.985	0.892	0.872	-0.239	0.283	0.371	0.995	0.889	0.918
414	53	-0.187	0.254	0.316	0.984	0.892	0.873	-0.230	0.280	0.362	0.996	0.887	0.918
415	54	-0.180	0.269	0.324	0.984	0.892	0.873	-0.214	0.286	0.357	0.996	0.887	0.918
416	55	-0.181	0.259	0.316	0.983	0.891	0.873	-0.212	0.281	0.352	0.996	0.886	0.918
417	56	-0.162	0.260	0.306	0.984	0.889	0.875	-0.212	0.279	0.351	0.996	0.884	0.917
418	57	-0.153	0.245	0.289	0.983	0.887	0.875	-0.211	0.270	0.342	0.997	0.883	0.917
419	58	-0.137	0.233	0.270	0.985	0.883	0.876	-0.209	0.251	0.326	0.996	0.882	0.920
420	59	-0.142	0.211	0.254	0.985	0.884	0.877	-0.209	0.241	0.318	0.996	0.883	0.921
421	60	-0.161	0.205	0.261	0.984	0.885	0.877	-0.215	0.234	0.318	0.996	0.885	0.924
422	61	-0.165	0.216	0.272	0.984	0.886	0.877	-0.224	0.242	0.329	0.995	0.887	0.924
423	62	-0.166	0.216	0.272	0.985	0.889	0.879	-0.220	0.247	0.331	0.995	0.889	0.925
424	63	-0.180	0.225	0.288	0.985	0.887	0.878	-0.230	0.252	0.341	0.996	0.888	0.923
425	64	-0.185	0.218	0.286	0.984	0.885	0.879	-0.239	0.250	0.346	0.995	0.885	0.922
426	65	-0.193	0.221	0.294	0.985	0.882	0.877	-0.243	0.257	0.354	0.997	0.883	0.919
427	66	-0.186	0.210	0.281	0.983	0.880	0.876	-0.241	0.249	0.346	0.996	0.880	0.920
428	67	-0.164	0.230	0.283	0.983	0.879	0.874	-0.225	0.267	0.349	0.996	0.877	0.917
429	68	-0.171	0.245	0.298	0.982	0.877	0.873	-0.220	0.275	0.352	0.996	0.876	0.917
430	69	-0.151	0.248	0.290	0.983	0.873	0.869	-0.204	0.275	0.342	0.996	0.875	0.916
431	70	-0.146	0.269	0.306	0.983	0.876	0.869	-0.198	0.290	0.351	0.996	0.875	0.914
432	71	-0.136	0.279	0.310	0.984	0.876	0.867	-0.191	0.296	0.352	0.997	0.875	0.913
433	72	-0.134	0.291	0.320	0.983	0.874	0.865	-0.185	0.304	0.355	0.995	0.873	0.913
434	73	-0.133	0.302	0.330	0.983	0.873	0.862	-0.188	0.316	0.368	0.997	0.872	0.910
435	74	-0.129	0.308	0.334	0.984	0.873	0.861	-0.183	0.317	0.366	0.997	0.872	0.910
436	75	-0.125	0.320	0.344	0.984	0.870	0.861	-0.184	0.325	0.374	0.996	0.873	0.910
437	76	-0.127	0.321	0.345	0.984	0.869	0.860	-0.184	0.330	0.378	0.996	0.873	0.911
438	77	-0.127	0.313	0.338	0.984	0.867	0.859	-0.186	0.321	0.371	0.996	0.873	0.913
439	78	-0.131	0.303	0.330	0.984	0.866	0.858	-0.187	0.317	0.369	0.995	0.873	0.913
440	79	-0.156	0.297	0.336	0.984	0.866	0.856	-0.203	0.314	0.374	0.995	0.871	0.911
441	80	-0.177	0.311	0.358	0.984	0.865	0.853	-0.213	0.324	0.388	0.996	0.870	0.908
442	81	-0.184	0.316	0.366	0.982	0.864	0.849	-0.218	0.332	0.397	0.997	0.869	0.906
443	82	-0.173	0.322	0.365	0.983	0.864	0.847	-0.215	0.341	0.403	0.996	0.865	0.901
444	83	-0.175	0.319	0.364	0.982	0.861	0.846	-0.217	0.336	0.400	0.996	0.862	0.899
445	84	-0.170	0.320	0.362	0.984	0.859	0.844	-0.215	0.338	0.401	0.996	0.859	0.896
446	85	-0.165	0.313	0.353	0.984	0.855	0.843	-0.218	0.329	0.395	0.996	0.855	0.894
447	86	-0.147	0.308	0.341	0.984	0.856	0.844	-0.214	0.320	0.385	0.996	0.857	0.897
448	87	-0.157	0.313	0.350	0.982	0.857	0.844	-0.219	0.315	0.384	0.995	0.863	0.901
449	88	-0.148	0.302	0.337	0.984	0.858	0.847	-0.225	0.297	0.373	0.996	0.867	0.906
450	89	-0.134	0.297	0.326	0.983	0.860	0.849	-0.225	0.290	0.367	0.996	0.870	0.910
451	90	-0.123	0.293	0.317	0.983	0.860	0.850	-0.208	0.283	0.351	0.997	0.870	0.911
452	91	-0.113	0.298	0.319	0.983	0.861	0.850	-0.191	0.286	0.344	0.997	0.869	0.911
453	92	-0.097	0.297	0.312	0.983	0.860	0.852	-0.170	0.289	0.335	0.996	0.869	0.910
454	93	-0.090	0.299	0.312	0.984	0.860	0.852	-0.148	0.289	0.325	0.995	0.868	0.909
455	94	-0.094	0.298	0.312	0.983	0.857	0.851	-0.139	0.298	0.329	0.996	0.868	0.909
456	95	-0.091	0.290	0.304	0.983	0.855	0.850	-0.130	0.298	0.325	0.996	0.868	0.908

457	96	-0.082	0.285	0.296	0.984	0.852	0.849	-0.114	0.294	0.316	0.997	0.866	0.908
458	97	-0.077	0.276	0.286	0.985	0.850	0.848	-0.098	0.286	0.302	0.996	0.867	0.908
459	98	-0.088	0.279	0.292	0.984	0.850	0.849	-0.104	0.282	0.301	0.996	0.867	0.909
460	99	-0.093	0.283	0.297	0.985	0.850	0.848	-0.107	0.279	0.299	0.996	0.867	0.910
461	100	-0.098	0.286	0.302	0.984	0.850	0.848	-0.122	0.281	0.306	0.996	0.867	0.909
462	101	-0.092	0.294	0.308	0.983	0.848	0.846	-0.119	0.289	0.313	0.995	0.865	0.907
463	102	-0.079	0.310	0.320	0.984	0.843	0.842	-0.106	0.309	0.327	0.996	0.863	0.904
464	103	-0.070	0.325	0.332	0.984	0.842	0.839	-0.104	0.315	0.332	0.996	0.859	0.900
465	104	-0.084	0.338	0.348	0.983	0.840	0.836	-0.115	0.328	0.348	0.995	0.856	0.897
466	105	-0.080	0.347	0.356	0.982	0.839	0.834	-0.104	0.338	0.354	0.997	0.857	0.896
467	106	-0.074	0.346	0.354	0.982	0.842	0.834	-0.098	0.338	0.352	0.995	0.860	0.900
468	107	-0.065	0.362	0.368	0.984	0.843	0.834	-0.092	0.354	0.366	0.997	0.864	0.901
469	108	-0.064	0.376	0.381	0.983	0.844	0.833	-0.093	0.365	0.377	0.997	0.866	0.903
470	109	-0.049	0.374	0.377	0.984	0.845	0.833	-0.088	0.362	0.373	0.996	0.867	0.903
471	110	-0.050	0.361	0.365	0.983	0.845	0.836	-0.091	0.355	0.366	0.996	0.866	0.903
472	111	-0.043	0.363	0.365	0.982	0.845	0.836	-0.088	0.354	0.364	0.995	0.865	0.902
473	112	-0.030	0.367	0.368	0.983	0.846	0.836	-0.086	0.355	0.365	0.996	0.865	0.901
474	113	-0.038	0.377	0.379	0.982	0.845	0.833	-0.089	0.359	0.370	0.997	0.865	0.900
475	114	-0.045	0.374	0.377	0.983	0.843	0.832	-0.087	0.362	0.372	0.995	0.862	0.900
476	115	-0.030	0.373	0.374	0.983	0.842	0.832	-0.073	0.362	0.369	0.996	0.861	0.899
477	116	-0.034	0.372	0.374	0.984	0.841	0.831	-0.067	0.362	0.368	0.995	0.858	0.898
478	117	-0.036	0.369	0.370	0.985	0.841	0.830	-0.071	0.354	0.361	0.995	0.858	0.896
479	118	-0.049	0.367	0.370	0.984	0.840	0.828	-0.077	0.356	0.364	0.996	0.859	0.894
480	119	-0.077	0.377	0.385	0.984	0.840	0.825	-0.087	0.369	0.379	0.996	0.859	0.891
481	120	-0.082	0.374	0.383	0.985	0.842	0.824	-0.090	0.372	0.383	0.996	0.861	0.890
482	121	-0.056	0.374	0.378	0.983	0.842	0.824	-0.082	0.387	0.395	0.996	0.863	0.890
483	122	-0.044	0.378	0.381	0.985	0.842	0.824	-0.081	0.397	0.405	0.995	0.867	0.892
484	123	-0.030	0.380	0.381	0.985	0.841	0.823	-0.079	0.401	0.409	0.994	0.869	0.892
485	124	-0.017	0.381	0.381	0.983	0.841	0.822	-0.069	0.402	0.407	0.995	0.872	0.893
486	125	-0.013	0.384	0.384	0.983	0.841	0.822	-0.058	0.415	0.419	0.994	0.872	0.891
487	126	0.006	0.394	0.394	0.982	0.839	0.818	-0.034	0.426	0.427	0.995	0.873	0.891
488	127	0.007	0.396	0.396	0.983	0.839	0.816	-0.021	0.428	0.429	0.996	0.873	0.890
489	128	0.010	0.407	0.407	0.983	0.837	0.811	-0.009	0.439	0.439	0.996	0.872	0.886
490	129	0.010	0.416	0.416	0.984	0.836	0.806	0.004	0.446	0.446	0.996	0.869	0.880
491	130	-0.003	0.431	0.431	0.983	0.833	0.801	0.009	0.459	0.459	0.996	0.865	0.873
492	131	0.004	0.444	0.444	0.983	0.832	0.795	0.033	0.474	0.475	0.996	0.858	0.863
493	132	0.018	0.450	0.450	0.983	0.833	0.795	0.039	0.489	0.490	0.995	0.862	0.862
494	133	0.029	0.462	0.463	0.982	0.838	0.792	0.031	0.502	0.503	0.996	0.868	0.865
495	134	0.047	0.468	0.471	0.984	0.843	0.793	0.034	0.514	0.516	0.995	0.875	0.866
496	135	0.056	0.475	0.478	0.983	0.846	0.793	0.034	0.528	0.529	0.995	0.881	0.867
497	136	0.055	0.468	0.472	0.985	0.845	0.791	0.031	0.527	0.528	0.995	0.881	0.865
498	137	0.069	0.454	0.459	0.984	0.844	0.790	0.046	0.521	0.523	0.996	0.879	0.863
499	138	0.057	0.456	0.460	0.984	0.843	0.788	0.046	0.532	0.534	0.997	0.877	0.858
500	139	0.054	0.470	0.474	0.983	0.842	0.783	0.048	0.549	0.551	0.997	0.873	0.852
501	140	0.079	0.480	0.486	0.983	0.845	0.780	0.065	0.565	0.569	0.995	0.875	0.848
502	141	0.089	0.501	0.509	0.984	0.845	0.775	0.070	0.589	0.593	0.995	0.879	0.845
503	142	0.104	0.516	0.527	0.985	0.847	0.771	0.080	0.613	0.618	0.995	0.880	0.839
504	143	0.122	0.525	0.539	0.983	0.846	0.768	0.091	0.630	0.636	0.995	0.879	0.834
505	144	0.118	0.538	0.551	0.983	0.846	0.766	0.083	0.642	0.647	0.995	0.880	0.832
506	145	0.134	0.532	0.549	0.984	0.846	0.765	0.083	0.644	0.649	0.995	0.880	0.832
507	146	0.148	0.543	0.563	0.982	0.844	0.762	0.085	0.656	0.662	0.996	0.878	0.829
508	147	0.152	0.533	0.554	0.984	0.844	0.764	0.091	0.657	0.663	0.996	0.876	0.827
509	148	0.178	0.519	0.548	0.982	0.845	0.766	0.134	0.635	0.649	0.996	0.870	0.821
510	149	0.192	0.514	0.549	0.982	0.843	0.765	0.160	0.626	0.646	0.997	0.864	0.817
511	150	0.206	0.491	0.532	0.983	0.843	0.769	0.184	0.601	0.629	0.994	0.857	0.814
512	151	0.226	0.479	0.530	0.983	0.844	0.771	0.199	0.595	0.627	0.996	0.855	0.813
513	152	0.231	0.476	0.530	0.983	0.845	0.770	0.196	0.585	0.617	0.997	0.854	0.814
514	153	0.255	0.452	0.519	0.983	0.845	0.775	0.206	0.561	0.597	0.995	0.855	0.816
515	154	0.276	0.429	0.510	0.984	0.845	0.778	0.208	0.539	0.577	0.996	0.853	0.818
516	155	0.293	0.416	0.509	0.985	0.846	0.782	0.224	0.514	0.561	0.995	0.859	0.828

517	156	0.302	0.395	0.497	0.985	0.849	0.789	0.242	0.491	0.548	0.995	0.868	0.840
518	157	0.307	0.368	0.479	0.983	0.850	0.796	0.237	0.468	0.524	0.995	0.877	0.854
519	158	0.311	0.368	0.482	0.984	0.851	0.798	0.232	0.450	0.506	0.996	0.885	0.865
520	159	0.323	0.354	0.479	0.984	0.850	0.801	0.252	0.433	0.501	0.996	0.885	0.869
521	160	0.324	0.341	0.471	0.985	0.849	0.805	0.265	0.416	0.493	0.996	0.883	0.872
522	161	0.338	0.309	0.458	0.983	0.847	0.810	0.281	0.390	0.481	0.996	0.883	0.876
523	162	0.341	0.294	0.450	0.983	0.847	0.814	0.291	0.372	0.472	0.996	0.883	0.880
524	163	0.368	0.296	0.473	0.983	0.851	0.816	0.300	0.368	0.474	0.994	0.882	0.880
525	164	0.390	0.294	0.488	0.985	0.856	0.819	0.307	0.368	0.479	0.995	0.882	0.879
526	165	0.392	0.283	0.484	0.984	0.860	0.822	0.298	0.360	0.467	0.995	0.881	0.879
527	166	0.419	0.284	0.507	0.983	0.865	0.825	0.308	0.354	0.469	0.995	0.880	0.879
528	167	0.433	0.278	0.514	0.983	0.867	0.827	0.334	0.346	0.481	0.996	0.882	0.882
529	168	0.494	0.276	0.566	0.983	0.871	0.826	0.379	0.341	0.509	0.996	0.883	0.883
530	169	0.507	0.283	0.580	0.984	0.872	0.826	0.399	0.337	0.522	0.994	0.886	0.885
531	170	0.521	0.275	0.589	0.984	0.872	0.827	0.417	0.328	0.531	0.996	0.885	0.886
532	171	0.514	0.286	0.588	0.984	0.869	0.828	0.416	0.330	0.531	0.996	0.882	0.886
533	172	0.524	0.276	0.592	0.984	0.865	0.828	0.420	0.325	0.531	0.995	0.877	0.886
534	173	0.533	0.282	0.603	0.982	0.862	0.826	0.425	0.319	0.531	0.996	0.873	0.888
535	174	0.540	0.266	0.602	0.984	0.859	0.826	0.432	0.304	0.528	0.996	0.872	0.891
536	175	0.574	0.257	0.629	0.984	0.856	0.824	0.472	0.285	0.551	0.995	0.872	0.894
537	176	0.591	0.252	0.642	0.983	0.853	0.822	0.491	0.268	0.559	0.996	0.873	0.898
538	177	0.598	0.258	0.651	0.983	0.850	0.820	0.512	0.262	0.575	0.996	0.875	0.901
539	178	0.636	0.289	0.698	0.983	0.851	0.817	0.527	0.284	0.599	0.995	0.874	0.900
540	179	0.673	0.317	0.744	0.983	0.854	0.812	0.545	0.315	0.629	0.995	0.874	0.896
541	180	0.698	0.340	0.776	0.983	0.853	0.810	0.553	0.342	0.650	0.996	0.873	0.892
542	181	0.716	0.356	0.800	0.984	0.856	0.809	0.564	0.361	0.669	0.996	0.873	0.889
543	182	0.749	0.401	0.849	0.985	0.862	0.802	0.583	0.409	0.712	0.995	0.875	0.882
544	183	0.770	0.441	0.887	0.984	0.865	0.795	0.594	0.455	0.748	0.996	0.878	0.875
545	184	0.787	0.488	0.926	0.983	0.868	0.784	0.601	0.511	0.788	0.997	0.881	0.867
546	185	0.813	0.517	0.964	0.985	0.871	0.776	0.613	0.551	0.825	0.996	0.883	0.860
547	186	0.862	0.535	1.014	0.985	0.872	0.767	0.652	0.572	0.867	0.995	0.883	0.854
548	187	0.885	0.543	1.038	0.984	0.873	0.761	0.678	0.593	0.901	0.995	0.885	0.848
549	188	0.906	0.552	1.061	0.983	0.874	0.754	0.705	0.612	0.933	0.995	0.886	0.842
550	189	0.936	0.581	1.102	0.984	0.875	0.745	0.722	0.642	0.966	0.996	0.886	0.834
551	190	0.958	0.585	1.123	0.984	0.876	0.741	0.725	0.664	0.984	0.996	0.887	0.828
552	191	0.980	0.594	1.146	0.984	0.877	0.735	0.735	0.677	0.999	0.996	0.886	0.823
553	192	0.989	0.591	1.152	0.982	0.878	0.733	0.735	0.686	1.005	0.996	0.882	0.818
554	193	1.004	0.583	1.162	0.983	0.880	0.732	0.740	0.680	1.005	0.996	0.880	0.815
555	194	1.051	0.536	1.180	0.985	0.880	0.727	0.768	0.644	1.002	0.996	0.880	0.815
556	195	1.075	0.510	1.190	0.986	0.883	0.725	0.787	0.612	0.997	0.996	0.881	0.815
557	196	1.093	0.471	1.191	0.984	0.885	0.727	0.804	0.577	0.990	0.996	0.879	0.815
558	197	1.115	0.440	1.199	0.984	0.888	0.725	0.833	0.536	0.991	0.996	0.883	0.817
559	198	1.155	0.417	1.228	0.984	0.894	0.720	0.890	0.496	1.019	0.995	0.888	0.817
560	199	1.189	0.389	1.251	0.983	0.896	0.715	0.942	0.452	1.045	0.995	0.891	0.817
561	200	1.224	0.354	1.275	0.983	0.902	0.711	0.986	0.409	1.068	0.996	0.894	0.817
562	201	1.257	0.333	1.300	0.984	0.907	0.705	1.034	0.387	1.104	0.997	0.896	0.816
563	202	1.281	0.332	1.323	0.983	0.911	0.699	1.079	0.370	1.140	0.996	0.899	0.817
564	203	1.315	0.307	1.351	0.983	0.916	0.694	1.121	0.340	1.172	0.996	0.904	0.818
565	204	1.335	0.285	1.365	0.984	0.919	0.689	1.168	0.316	1.210	0.996	0.909	0.818
566	205	1.367	0.301	1.399	0.983	0.918	0.682	1.198	0.328	1.242	0.996	0.909	0.815
567	206	1.412	0.323	1.448	0.983	0.918	0.671	1.237	0.342	1.283	0.995	0.907	0.811
568	207	1.444	0.319	1.479	0.984	0.915	0.663	1.263	0.348	1.310	0.996	0.906	0.807
569	208	1.466	0.349	1.507	0.984	0.912	0.658	1.289	0.368	1.341	0.997	0.905	0.803
570	209	1.496	0.411	1.551	0.983	0.917	0.648	1.342	0.435	1.411	0.997	0.909	0.796
571	210	1.544	0.462	1.612	0.983	0.920	0.638	1.410	0.492	1.493	0.996	0.914	0.786
572	211	1.587	0.515	1.668	0.984	0.917	0.625	1.484	0.551	1.583	0.996	0.914	0.773
573	212	1.604	0.533	1.691	0.984	0.921	0.622	1.536	0.588	1.644	0.995	0.912	0.762
574	213	1.631	0.538	1.718	0.984	0.919	0.613	1.574	0.600	1.685	0.996	0.905	0.750
575	214	1.656	0.556	1.747	0.984	0.917	0.608	1.608	0.622	1.724	0.996	0.894	0.737
576	215	1.674	0.568	1.767	0.984	0.914	0.602	1.659	0.640	1.778	0.996	0.878	0.721

577	216	1.707	0.568	1.800	0.985	0.916	0.598	1.705	0.636	1.820	0.996	0.867	0.709
578	217	1.737	0.512	1.811	0.984	0.917	0.599	1.707	0.580	1.802	0.996	0.873	0.711
579	218	1.768	0.479	1.832	0.985	0.920	0.597	1.706	0.540	1.790	0.996	0.874	0.709
580	219	1.792	0.439	1.845	0.984	0.921	0.598	1.708	0.490	1.777	0.997	0.872	0.706
581	220	1.822	0.409	1.867	0.984	0.925	0.594	1.750	0.458	1.809	0.997	0.879	0.706
582	221	1.835	0.390	1.876	0.984	0.926	0.595	1.806	0.438	1.858	0.997	0.890	0.713
583	222	1.844	0.373	1.882	0.984	0.930	0.595	1.847	0.421	1.895	0.996	0.897	0.713
584	223	1.854	0.350	1.887	0.983	0.933	0.595	1.879	0.396	1.921	0.997	0.900	0.705
585	224	1.897	0.353	1.930	0.983	0.936	0.585	1.918	0.404	1.960	0.997	0.909	0.698
586	225	1.937	0.375	1.973	0.984	0.936	0.578	1.970	0.416	2.014	0.997	0.917	0.689
587	226	1.966	0.411	2.008	0.984	0.940	0.569	2.032	0.439	2.079	0.997	0.927	0.673
588	227	2.020	0.449	2.070	0.984	0.944	0.556	2.103	0.465	2.154	0.996	0.938	0.656
589	228	2.070	0.498	2.129	0.984	0.942	0.542	2.177	0.549	2.245	0.996	0.940	0.640
590	229	2.136	0.560	2.208	0.984	0.940	0.526	2.258	0.646	2.349	0.996	0.932	0.620
591	230	2.192	0.625	2.280	0.984	0.934	0.512	2.343	0.733	2.454	0.996	0.916	0.600
592	231	2.250	0.666	2.346	0.984	0.928	0.498	2.421	0.793	2.547	0.996	0.901	0.583
593	232	2.284	0.632	2.370	0.983	0.927	0.498	2.448	0.759	2.562	0.997	0.904	0.584
594	233	2.309	0.607	2.388	0.984	0.928	0.496	2.480	0.723	2.583	0.996	0.908	0.583
595	234	2.337	0.575	2.407	0.983	0.925	0.494	2.504	0.691	2.598	0.996	0.913	0.585
596	235	2.376	0.556	2.440	0.984	0.926	0.491	2.529	0.659	2.614	0.997	0.917	0.587
597	236	2.426	0.542	2.486	0.985	0.928	0.483	2.563	0.634	2.640	0.996	0.923	0.586
598	237	2.467	0.516	2.521	0.984	0.931	0.477	2.595	0.607	2.665	0.996	0.925	0.587
599	238	2.523	0.502	2.573	0.985	0.933	0.465	2.633	0.579	2.696	0.996	0.925	0.584
600	239	2.558	0.512	2.608	0.983	0.934	0.457	2.661	0.586	2.725	0.997	0.925	0.582
601	240	2.610	0.582	2.674	0.985	0.931	0.445	2.712	0.656	2.790	0.996	0.928	0.573
602	241	2.649	0.658	2.730	0.984	0.930	0.432	2.762	0.730	2.857	0.997	0.926	0.563
603	242	2.694	0.707	2.786	0.984	0.926	0.423	2.812	0.788	2.921	0.996	0.923	0.552
604	243	2.761	0.769	2.867	0.984	0.919	0.408	2.861	0.851	2.985	0.997	0.918	0.545
605	244	2.827	0.782	2.933	0.984	0.914	0.396	2.900	0.881	3.031	0.998	0.915	0.540
606	245	2.889	0.795	2.996	0.984	0.910	0.386	2.937	0.909	3.075	0.997	0.911	0.535
607	246	2.945	0.811	3.054	0.984	0.904	0.377	2.977	0.932	3.120	0.997	0.908	0.531
608	247	3.012	0.805	3.118	0.983	0.909	0.363	3.029	0.927	3.168	0.997	0.912	0.526
609	248	3.073	0.781	3.171	0.984	0.917	0.348	3.077	0.908	3.208	0.997	0.922	0.523
610	249	3.142	0.755	3.232	0.984	0.927	0.328	3.129	0.880	3.250	0.997	0.931	0.519
611	250	3.221	0.740	3.305	0.983	0.936	0.306	3.184	0.856	3.297	0.997	0.928	0.515